

UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS



**Software interactivo para la educación medioambiental de la  
población Salvadoreña  
SIEMPOSAL**

PRESENTADO POR:

**DINORA DEL CARMEN DELEÓN GÓMEZ  
LIGIA ASTRID HERNÁNDEZ BONILLA  
MIGUEL ÁNGEL LÓPEZ DEL CID  
ROBERTO ANTONIO ZELAYA**

PARA OPTAR AL TÍTULO DE:

**INGENIERO DE SISTEMAS INFORMÁTICOS**

CIUDAD UNIVERSITARIA, FEBRERO DE 2009

**UNIVERSIDAD DE EL SALVADOR**

RECTOR :

**MSc. RUFINO ANTONIO QUEZADA SÁNCHEZ**

SECRETARIO GENERAL :

**LIC. DOUGLAS VLADIMIR ALFARO CHÁVEZ**

**FACULTAD DE INGENIERÍA Y ARQUITECTURA**

DECANO :

**ING. MARIO ROBERTO NIETO LOVO**

SECRETARIO :

**ING. OSCAR EDUARDO MARROQUÍN HERNÁNDEZ**

**ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS**

DIRECTOR :

**MSc. CARLOS ERNESTO GARCÍA GARCÍA**

UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

Trabajo de Graduación previo a la opción al Grado de:

**INGENIERO DE SISTEMAS INFORMÁTICOS**

Título :

**Software interactivo para la educación medioambiental de  
la población Salvadoreña  
SIEMPOSAL**

Presentado por :

**DINORA DEL CARMEN DELEÓN GÓMEZ  
LIGIA ASTRID HERNÁNDEZ BONILLA  
MIGUEL ÁNGEL LÓPEZ DEL CID  
ROBERTO ANTONIO ZELAYA**

Trabajo de Graduación Aprobado por:

Docente Director :

**ING. ELMER ARTURO CARBALLO RUIZ**

San Salvador, Febrero de 2009

Trabajo de Graduación Aprobado por:

Docente Director :

**ING. ELMER ARTURO CARBALLO RUIZ**



# *Agradecimientos*

Agradezco primeramente a mi Padre Celestial, **YAVÉ, DIOS TODOPODEROSO** por que a Él le ha placido darme vida, fuerza y todo lo necesario para poder llegar hasta el final de esta carrera, agradezco a Dios, por no dejarme sola en los momentos más difíciles, por guiarme en toda mi carrera, por permitirme haber sido parte de la creación del videojuego que dejamos como legado a la UES, le agradezco por todas sus bendiciones materiales y espirituales derramadas sobre mi vida y ahora dedico este triunfo al Creador del sistema más complejo que existe, el cielo, tierra, mar y del universo en general. Le alabo por sus infinitas bondades hacia mi.

**A JESUCRISTO EL HIJO DE DIOS** por ser el Salvador de mi vida, mi abogado, el mediador entre Dios y mi persona, por estar siempre conmigo.

**AL ESPIRITU SANTO** por ser mi guía en todo momento, por ser mi consolador, mi intercesor, le agradezco por iluminar mi camino.

**A MI FAMILIA MATERIAL** por toda su ayuda incondicional, dedico este triunfo:

**A MIS PAPÁS:** José Carlos Deleón Crespín y Martha Gómez de Deleón, por todo su amor, cariño y apoyo incondicional en todo momento, les agradezco por creer en mi, por esperar con paciencia el día en que finalizara esta carrera.

**A MIS HERMANOS,** Verónica, Elmer y Marisol por todo su apoyo y en especial a Verónica por comprenderme, por escucharme, por esforzarme a no desmayar en mi camino, gracias Vero por tu apoyo.

**A MIS SOBRINOS,** Elizabeth, Samuel, Moisés y Karlita, porque en los momentos en los cuales me encontraba triste por alguna circunstancia de la vida, recordarlos me llenaba de felicidad. Recordar sus sonrisas, los momentos felices que he vivido con ellos me ayudaban a tomar nuevos ánimos, me ayudaba a olvidar lo que me pasaba.

**A MI FAMILIA ESPIRITUAL** por todo el amor, apoyo espiritual que me han dado durante estos años, en especial al Hno. Pedro Peraza y al Hno. Carlos Aguilar. A todos les agradezco infinitamente.

**A MIS COMPAÑEROS DE TRABAJO DE GRADUACIÓN** por ser personas llevaderas, apacibles, con las cuales he compartido tiempo de mi vida, y juntos hemos superado todos los retos y obstáculos para lograr nuestro objetivo. Quiero agradecer de manera especial a

Roberto por ser una gran persona, un gran amigo, gracias Robert, por transmitirme tus conocimientos.

**A DOÑA MELVA ZELAYA**, por permitirme entrar y permanecer en su casa durante el desarrollo de este trabajo.

**A LOS DOCENTES** de la Escuela de Ingeniería de Sistemas Informáticos por transmitirnos el conocimiento necesario para poder hoy ser parte de los profesionales de nuestro país.

**AL ING. ELMER CARBALLO**, Director de este proyecto, por su apoyo, su comprensión y paciencia durante el desarrollo de este proyecto.

**A ROBERTO BONILLA** por compartir sus conocimientos conmigo, pues me han servido de mucho, gracias Roberto, por todo.

**A ROSALBA SANDOVAL** por todo su apoyo al grupo, pues ha sido de mucha ayuda para la finalización de este trabajo de graduación.

Finalmente, quiero agradecer a todas aquellas personas familiares y amigos que me ayudaron a llegar hasta el final de esta carrera,

..... a todos muchas gracias.

*Dinora del Carmen Delcín Gómez*

# **AGRADECIMIENTOS**

## **A Dios:**

Por su amor, misericordia y la fortaleza que me ha dado en los momentos de adversidad.

## **A mi madre:**

Por darme los dos tesoros más valiosos de mi vida: El amor y la educación.

## **A mis compañeros de trabajo de graduación:**

Dinora, Roberto y Miguel, por el esfuerzo dedicado a este proyecto, gracias por el conocimiento, apoyo y la comprensión que me han brindado durante todo el proceso de formación profesional.

## **A mis hermanos:** Hazzel, Aracely, Henry y Rosalba

Por su amor, apoyo y amistad. Me han mostrado que la perseverancia y el esfuerzo son el camino para lograr el éxito.

## **Al Ing. Elmer Arturo Carballo:**

Por brindarnos su apoyo y conocimientos para el desarrollo de este proyecto.

Ligia Astrid Hernández Bonilla

## AGRADECIMIENTOS

En primer lugar agradezco a Dios por permitirme terminar mi etapa académica, durante todos estos años dándome fuerza, fortaleza, voluntad, paciencia, constancia y perseverancia al logro de esta meta.

A mi madre Zoila Andrea del Cid que me apoyo de gran manera en todos los aspectos que se requieren para poder llegar a mi meta académica, su apoyo fue el más importante ya que su amor, apoyo, confianza, comprensión y consejos fueron una base que me alimento para seguir en este objetivo que hoy se ha cumplido. Y nada mas se puede pedir de una madre.

Por eso este logro y agradecimiento es primeramente para Dios y a mi madre.

Agradezco a la Universidad de El Salvador, por abrirme su puerta, para darme la oportunidad de llegar a ser un profesional. Ya que ella da una apertura a las generaciones pasadas actuales y las futuras, la oportunidad de que muchos logremos nuestro sueño de ser profesionales.

A mi asesor Ing. Elmer Arturo Carballo que fue una persona excelente profesionalmente y personalmente, por su asesoría, consejos y su comprensión durante nuestro trabajo. Así como a nuestro observador Ing. Rigoberto flores por su buen criterio profesional. A nuestro coordinador Rubén Acencio por su comprensión y su buena coordinación.

A mi grupo de trabajo de graduación que a pesar de todas las dificultades logramos llegar hasta el final logrando el objetivo propuesto.

A mis amigos que cuando estuve grave de enfermedad me hicieron compañía y estuvieron conmigo Mario Alberto Acosta García, Pavel Salomón murga y Edwin Giovanni Paredes (Largo), Familia de don Mardo, niña Tere e hijos, Gerber y familia.

A mi generación de mis primeros años de universidad, Mario Acosta, Alcides Henríquez Contreras, Raúl Beltrán guzmán, Roger Dacio, Ing. Balmore, Gloria Sigüenza, Luz, Edwin (chino viejo), Ronaldo (chino), Ronaldo (pepe), Carlos Osorio, Carlos Cuadra, Julio (babanin), el Lic. Jaime, Marvin (Rambo), Carlos palacios, Henry Contreras y Jorge Alberto Jiménez moreno. Los cuales compartimos buenos momentos y compañía.

Además a Laura Ayala, Ana Miriam Méndez, Noemí, Rosalba Guerra y a mi gran recordada Kerys Lucila López orellana. Por su compañía en estos años de estudio.

A mis amigos actuales más allegados Manuel de Jesús coreas, Adán Mauricio Romero, Henry Cañengues (triple), Ernesto Zetino (Burro) a Leo que me dieron su amistad y apoyo cuando los necesitaba y a los amigos de fútbol que llegan al polideportivo. Con los cuales compartimos buenos momentos de camarería. Y compartimos muchas cosas más.

Francamente hay muchas personas que en el transcurso de mi estudio académico me dieron su amistad y compañía, a ellas también les dedico. El colocar todos sus nombres no me ajustará esta página pero cuando lean estas líneas sabrán que también es para ellas.

Les dedico este logro a mi familia hermanos Francisco Antonio Guerrero del Cid, Ana Graciela guerrero del Cid, Guadalupe Guerrero del Cid, sobrinos Víctor Alexander Díaz guerrero, Lorena Anabell Valle Guerrero, Claudia Carolina Aguilar guerrero, Marlon y Daniel aguilar.

Miguel Ángel López del Cid

## Agradecimientos:

A mi familia sin cuyo soporte, paciencia, amor y consuelo, el peso cotidiano sería mucho más difícil de cargar.

A los ingenieros Elmer Arturo Carballo, Pedro Vásquez, Patricia Estrada, Elizabeth de Pocasangre y Emerson Padilla quienes me enseñaron no solamente conocimientos indispensables de ingeniería, si no también he aprendido de ellos que la humildad, la virtud base de los triunfadores y el sentido humano no deben ser opacados por la burocracia del sistema de educación superior, a ellos gracias y que Dios les bendiga siempre.

A mi Terruño El Salvador, que a pesar que un día di por olvidado, sin tierra ni desazón me extendió sus brazos, otorgándome un hogar y retornándome a la identidad que daba por perdida.

Y por último y más importante, al Creador, quien a pesar que yace escondido a mis ojos, se ha compadecido de mi, sacándome de la oscuridad y pena, angustia y necesidad, mostrándome a través de la luz de este mundo que todo es posible para aquel que cree, y a pesar que la magnitud de la grandeza del mero propósito de la vida es insostenible dentro de la mente humana, solo con Él lo amargo y lo dulce tienen tanto sentido, solo con Él un hombre como yo ha logrado escalar un tramo más en la montaña de la sociedad, basta deleitar mi ego en el hecho que este logro, por común que sea para algunos, únicamente he podido alcanzarlo porque a Él le complace. Es en ese sentido de agradecimiento y sentimiento satisfacción por el fruto de un trabajo de varios años que deseo plasmar en este libro las palabras que siempre desee decir en el lenguaje del alma al final de mi carrera:

בורך אתה אדני ובורך אתה יהושוע המשיח

Roberto Antonio Zelaya,

San Salvador, lunes 2 de Febrero, MMIX



## CONTENIDO

|  |            |
|--|------------|
| <b>INTRODUCCION .....</b>  | <b>I</b>   |
| <b>OBJETIVOS.....</b>  | <b>III</b> |
| <b>IMPORTANCIA Y JUSTIFICACIÓN.....</b>                                    | <b>IV</b>  |
| IMPORTANCIA.....   | IV         |
| JUSTIFICACION.....   | VII        |
| <b>ALCANCES Y LIMITACIONES.....</b>  | <b>X</b>   |
| ALCANCES .....   | X          |
| LIMITACIONES .....   | X          |
| <b>CAPITULO I: ESTUDIO PRELIMINAR .....</b>                                | <b>1</b>   |
| 1.1. ANTECEDENTES.....   | 2          |
| 1.1.1. <i>Origen de la Educación Ambiental en El Salvador.....</i>         | 2          |
| 1.1.2. <i>Instituciones Co-responsables de la Educación Ambiental.....</i> | 4          |
| 1.1.3. <i>Marco Legal.....</i>   | 7          |
| 1.1.4. <i>Aulas Informáticas en Centros Escolares de El Salvador.....</i>  | 9          |
| 1.1.5. <i>Software Interactivo.....</i>                                    | 10         |
| 1.2. SOLUCION DEL PROBLEMA .....   | 15         |
| 1.2.1. <i>Reconocimiento del problema .....</i>                            | 15         |
| 1.2.2. <i>Formulación del problema .....</i>                               | 18         |
| 1.2.3. <i>Planteamiento del problema .....</i>                             | 18         |
| 1.2.4. <i>Análisis del problema.....</i>                                   | 19         |
| 1.2.5. <i>Búsqueda de soluciones.....</i>                                  | 29         |
| 1.2.6. <i>Selección de alternativa de solución.....</i>                    | 30         |
| 1.3. FACTIBILIDADES DEL PROYECTO.....                                      | 31         |
| 1.3.1. <i>Factibilidad técnica .....</i>                                   | 32         |
| 1.3.2. <i>Factibilidad económica .....</i>                                 | 45         |
| 1.3.3. <i>Factibilidad operativa.....</i>                                  | 55         |
| 1.3.4. <i>Factibilidad ambiental.....</i>                                  | 59         |
| 1.3.5. <i>Conclusiones factibilidades del proyecto .....</i>               | 61         |
| 1.4. RESULTADOS ESPERADOS.....   | 61         |
| 1.4.1. <i>Operativos .....</i>   | 61         |
| 1.4.2. <i>Tácticos.....</i>  | 62         |



|  |            |
|--|------------|
| 1.4.3. <i>Estratégicos</i> .....   | 63         |
| <b>CAPITULO II: SITUACIÓN ACTUAL</b> .....   | <b>64</b>  |
| <b>2. SITUACION ACTUAL</b> .....   | <b>65</b>  |
| 2.1. ANALISIS DE LA SITUACION ACTUAL .....   | 65         |
| 2.1.1. <i>Diagrama de Enfoque de Sistemas</i> .....  | 68         |
| 2.1.2. <i>Descripción de los componentes del Enfoque de Sistemas</i> .....                             | 69         |
| 2.2. DESCRIPCION DE PROCESOS ACTUALES. ....  | 74         |
| 2.2.1. <i>Diagrama jerárquico de procesos.</i> .....   | 74         |
| 2.2.2. <i>Procesos</i> .....   | 75         |
| 2.2.3. <i>Simbología utilizada en los diagramas de procesos</i> .....                                  | 79         |
| 2.2.4. <i>Diagramas de procesos</i> .....  | 80         |
| 2.3. DIAGNOSTICO.....  | 91         |
| 2.3.1. <i>Problemas o deficiencias del programa de educación ambiental actual</i> .....                | 91         |
| 2.3.2. <i>Análisis FODA ponderado del programa actual de concientización ambiental del MARN.</i> ..... | 92         |
| 2.3.3. <i>Objetivos y pasos para analizar la Matriz FODA Ponderada:</i> .....                          | 93         |
| 2.4. PROPUESTA DE SOLUCIÓN.....  | 95         |
| 2.4.1. <i>Justificación de la propuesta de solución</i> .....  | 96         |
| 2.4.2. <i>Creación de un videojuego</i> .....  | 96         |
| 2.4.3. <i>¿Por qué Programación Orientada a Objetos?</i> .....   | 101        |
| 2.4.4. <i>Historia del Videojuego de Solución Propuesta</i> .....                                      | 102        |
| 2.4.5. <i>Temática y Alcance del Videojuego</i> .....  | 125        |
| <b>CAPITULO III: DETERMINACIÓN DE REQUERIMIENTOS</b> .....   | <b>130</b> |
| <b>3. DETERMINACION DE REQUERIMIENTOS</b> .....  | <b>131</b> |
| 3.1. REQUERIMIENTOS DE DESARROLLO.....   | 131        |
| 3.1.1. <i>Software de desarrollo.</i> .....  | 131        |
| 3.1.2. <i>Hardware</i> .....   | 135        |
| 3.1.3. <i>Recurso humano</i> .....   | 136        |
| 3.2. REQUERIMIENTOS OPERATIVOS.....  | 137        |
| 3.2.1. <i>Generales</i> .....  | 137        |
| 3.2.2. <i>Software</i> .....   | 139        |
| 3.2.3. <i>Hardware</i> .....   | 139        |
| 3.2.4. <i>Recurso humano</i> .....   | 140        |



|  |            |
|--|------------|
| 3.3. REQUERIMIENTOS DE INFORMACION.....  | 141        |
| 3.3.1. <i>Requerimientos funcionales</i> .....                                 | 141        |
| 3.3.2. <i>Requerimientos no funcionales</i> .....                              | 142        |
| <b>CAPITULO IV: ANÁLISIS DEL SOFTWARE.....</b>                                 | <b>144</b> |
| <b>4. ANALISIS DEL SOFTWARE.....</b>   | <b>145</b> |
| 4.1. FUNCIONES DEL SOFTWARE .....  | 145        |
| 4.1.1. <i>Funciones de Inicialización</i> .....                                | 145        |
| 4.1.2. <i>Funciones de Entrada</i> .....                                       | 146        |
| 4.1.3. <i>Funciones de Procesamiento</i> .....                                 | 146        |
| 4.1.4. <i>Funciones de Salida</i> .....  | 147        |
| 4.1.5. <i>Funciones de finalización</i> .....                                  | 147        |
| 4.2. MODELO DE CASOS DE USO .....  | 147        |
| 4.2.1. <i>Diagrama de casos de uso</i> .....                                   | 150        |
| 4.2.2. <i>Clasificación de los Casos de Uso del Diagrama de Contexto</i> ..... | 152        |
| 4.2.3. <i>Descripción de Casos de Uso</i> .....                                | 153        |
| 4.2.4. <i>Diagramas de Secuencia del Sistema (DSS)</i> .....                   | 174        |
| 4.2.5. <i>Contratos de operación</i> .....                                     | 187        |
| 4.3. MODELO DEL DOMINO .....   | 201        |
| 4.3.1. <i>Modelo de Dominio del Videojuego</i> .....                           | 202        |
| 4.3.2. <i>Organización del Modelo del Dominio en Paquetes</i> .....            | 203        |
| 4.4. DICCIONARIO DE DATOS DE LAS CLASES CONCEPTUALES.....                      | 205        |
| 4.5. CONCLUSION DE LA FASE DE ANÁLISIS.....                                    | 208        |
| <b>CAPITULO V: DISEÑO DEL SOFTWARE.....</b>                                    | <b>209</b> |
| <b>5. DISEÑO DEL SOFTWARE.....</b>   | <b>210</b> |
| 5.1. DISEÑO GLOBAL .....   | 211        |
| 5.1.1. <i>Estándares para el diseño de software</i> .....                      | 211        |
| 5.1.2. <i>Estándares de interfaz de salida</i> .....                           | 211        |
| 5.1.3. <i>Estándares de interfaz de Entrada</i> .....                          | 215        |
| 5.1.4. <i>Estándares de menús</i> .....  | 219        |
| 5.1.5. <i>Estándares de Código Fuente</i> .....                                | 221        |
| 5.1.6. <i>Estándares de Recursos Multimedia</i> .....                          | 232        |
| 5.1.7. <i>Patrones de Arquitectura</i> .....                                   | 233        |
| 5.1.8. <i>Diagrama Jerárquico del Videojuego</i> .....                         | 238        |

---





|   |            |
|---|------------|
| 5.2. DISEÑO DETALLADO .....   | 241        |
| 5.2.1. Descripción de Casos de Usos Reales (con diseño de interfaces) ..... | 241        |
| 5.2.2. Diagramas de Interacción .....                                       | 261        |
| 5.2.3. Integración de Patrones GRASP .....                                  | 272        |
| 5.2.4. Diagramas de Estado .....  | 284        |
| 5.2.5. Diagrama de Componentes .....  | 295        |
| 5.2.6. Diagrama de Clases y Paquetes.....                                   | 299        |
| <b>CAPITULO VI: PROGRAMACIÓN Y PRUEBAS DEL SOFTWARE .....</b>               | <b>304</b> |
| <b>6. DESARROLLO DE SIEMPOSAL .....</b>                                     | <b>305</b> |
| 6.1. SUB-ETAPAS DE DESARROLLO DE SIEMPOSAL.....                             | 305        |
| 6.1.1. Toma de fotografía. ....   | 305        |
| 6.1.2. Creación de Texturas. ....   | 306        |
| 6.1.3. Modelado y Animación. ....   | 306        |
| 6.1.4. Codificación e Integración de Modelos y Sonidos .....                | 307        |
| 6.2. AMBITO DE DESARROLLO .....   | 310        |
| 6.2.1. Condiciones tecnológicas .....                                       | 310        |
| 6.2.2. Configuración del entorno de desarrollo .....                        | 312        |
| 6.2.3. Metodología de trabajo del grupo de desarrollo.....                  | 312        |
| 6.2.4. Directorios y archivos .....   | 315        |
| 6.3. PRUEBAS DEL SOFTWARE .....   | 318        |
| 6.3.1. Generalidades de las pruebas realizadas .....                        | 318        |
| 6.3.2. Recorridos .....   | 320        |
| 6.3.3. Revisiones Formales .....  | 321        |
| 6.3.4. Caja negra.....  | 323        |
| 6.3.5. Caja blanca .....  | 333        |
| 6.3.6. Conclusiones de las pruebas realizadas a SIEMPOSAL .....             | 335        |
| <b>CAPITULO VII: DOCUMENTACIÓN Y PLAN DE IMPLANTACIÓN .....</b>             | <b>336</b> |
| <b>7. DOCUMENTACIÓN EXTERNA.....</b>  | <b>337</b> |
| 7.1. DOCUMENTACIÓN DEL SOFTWARE.....  | 337        |
| 7.2. PLAN DE IMPLANTACIÓN DEL SOFTWARE.....                                 | 337        |
| <b>CONCLUSIONES .....</b>   | <b>338</b> |
| <b>RECOMENDACIONES .....</b>  | <b>339</b> |
| <b>BIBLIOGRAFÍA .....</b>   | <b>340</b> |

---



---

|   |            |
|---|------------|
| LIBROS .....  | 340        |
| TRABAJOS DE GRADUACION .....  | 341        |
| SITIOS WEB.....   | 341        |
| <b>GLOSARIO DE TERMINOS.....</b>  | <b>343</b> |
| <b>ANEXOS.....</b>  | <b>346</b> |
| ANEXO No. 1: ORGANIGRAMA DEL MINISTERIO DE MEDIO AMBIENTE Y RECURSOS<br>NATURALES .....   | 347        |
| ANEXO No. 2: ORGANIGRAMA DE LA DIRECCION GENERAL DE PARTICIPACION<br>CIUDADANA (MARN) .....                                     | 348        |
| ANEXO No. 3: ORGANIGRAMA DEL MINISTERIO DE EDUCACION .....  | 349        |
| ANEXO No. 4: CÁLCULO DE LOS VALORES ACTUALES NETOS (VAN) Y TASA<br>INTERNA DE RETORNO (TIR).....                                | 350        |
| ANEXO No. 5: INSTRUMENTOS DE INVESTIGACION.....   | 351        |
| ANEXO No. 6: ESTUDIANTES POR GRUPO ETARIO .....   | 355        |
| ANEXO No. 7: GRAFICO ACEPTACION DE VIDEOJUEGOS .....  | 355        |
| ANEXO No. 8: REQUERIMIENTOS DE HARDWARE PARA EL SOFTWARE DE<br>DESARROLLO .....   | 356        |
| ANEXO No. 9: REQUERIMIENTOS DE HARDWARE PARA LA OPERACIÓN DEL<br>VIDEOJUEGO .....   | 360        |
| ANEXO No. 10: PLANEACIÓN DE RECURSOS.....   | 361        |
| ANEXO No. 11: CAPACITACIÓN Y DISTRIBUCIÓN DIRECTA DEL VIDEOJUEGO A<br>TRAVÉS DE CDS (NO INCLUYE MANUAL DE USUARIO IMPRESO)..... | 368        |

---



## INTRODUCCION

Este proyecto, debe su razón de ser a un problema que afecta a todos, la contaminación ambiental, por la cual se planteó la siguiente interrogante. ¿Qué puede hacerse para permitir un medioambiente sostenible en El Salvador?. La respuesta del Estado ha sido la creación de organismos pro-medioambiente como el Ministerio del Medio Ambiente y Recursos Naturales (MARN), que se encarga de concientizar a la población por medio de programas educativos, no obstante, sus objetivos en la mayoría de casos no alcanzan a cumplirse debido a factores como el bajo presupuesto que se destina este tipo de programas, metodologías poco interactivas, no existe seguimiento de logros, entre otros.

Se considera que no habrá soluciones reales mientras no exista educación y cultura medioambiental en todos los niveles y modalidades posibles, ante este reto, se requiere del replanteamiento de los procesos educativos en su conjunto y nuevos enfoques, métodos, conocimientos y nuevas relaciones entre los distintos agentes educativos. Es por esto, que se pretende desarrollar un software que promueva la concientización de la problemática ambiental que afronta El Salvador y a su vez expanda el alcance del actual programa de Educación Ambiental del MARN.

El presente documento es un consolidado que describe el desarrollo del “Software Interactivo para la Educación Medioambiental de la Población Salvadoreña”, en este se presenta la investigación preliminar que expone la necesidad de educar a los jóvenes en materia ambiental, también se describen videojuegos que han tenido éxito para promover cambios de conducta en las personas. Un punto esencial que se aborda en el estudio preliminar es el estudio de factibilidades, realizado desde los puntos de vista: tecnológico, económico, operativo y medioambiental, donde se estableció la viabilidad de realizar el Videojuego Ambiental.

Posteriormente, se analiza la situación actual del Programa Nacional para el Reciclaje donde por medio del Enfoque de Sistemas se presentan las entradas, salidas, medio ambiente, fronteras y recursos del sistema actual; para posteriormente realizar una descripción detallada de las actividades que se ejecutan en la Unidad de Educación Ambiental mediante diagramas de procesos.

Como otro apartado se presenta la descripción de necesidades de la contraparte por medio de los requerimientos de desarrollo, operativos y de información. Partiendo de estos, se define la funcionalidad que el software debe realizar. En el Capítulo IV: Análisis del Software, se muestra la conceptualización de la aplicación desarrollada por medio de los casos de uso esenciales y diagramas de secuencia de sistemas siguiendo el estándar UML 2.0.



El diseño del software se presenta en dos apartados importantes dentro del Capítulo V, estos son: el global, donde se plantea la arquitectura de desarrollo, estándares y el plan de pruebas, posteriormente se presenta el diseño detallado donde por medio de casos de uso reales, diagramas de interacción, de estados y clases se presenta la aplicación de forma tal que sea comprensible para los programadores de la solución. Las metodologías y pruebas realizadas en la etapa de programación se presentan en el capítulo VI.

A continuación, se le presenta al lector este documento que le servirá de guía para la creación de un videojuego utilizando el modelo de ciclo de vida en cascada y el enfoque orientado a objetos.



## **OBJETIVOS**

### **OBJETIVO GENERAL**

- Desarrollar un software interactivo que apoye la educación ambiental de la población Salvadoreña para fomentar actitudes que promuevan la conservación del medio ambiente.

### **OBJETIVOS ESPECÍFICOS**

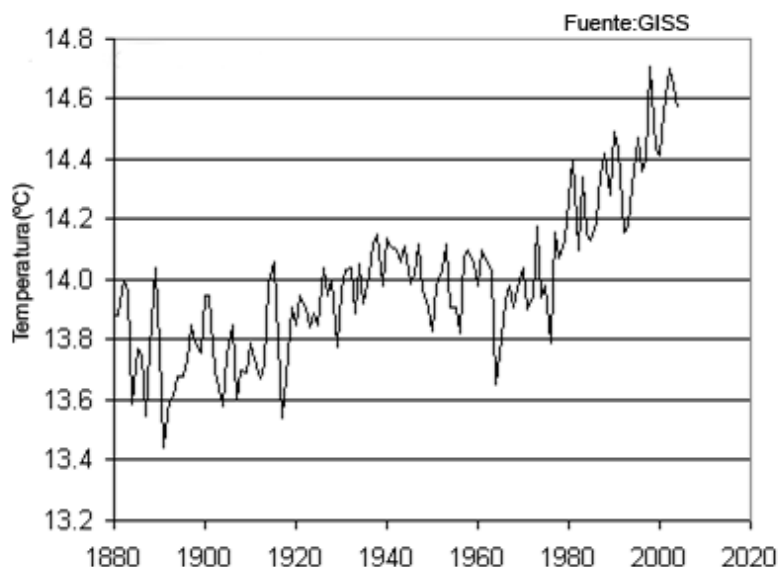
- Identificar y establecer los requerimientos, los elementos e interrelaciones que permitan construir el Software Interactivo para la Educación Medioambiental de la Población Salvadoreña.
- Diseñar la interfaz del Software Interactivo para la Educación Medioambiental de la Población Salvadoreña, de tal forma que se aprovechen las características adictivas de los videojuegos e influir en los jugadores por medio de mensajes de concientización ambiental.
- Codificar el software de acuerdo al diseño establecido, aplicando métodos, procedimientos, estándares y herramientas que provee la ingeniería de software para el desarrollo de una aplicación con métricas de calidad.
- Elaborar los manuales de instalación, usuario y técnico del software pro-medioambiente, así como también un demo que muestre al usuario el manejo operativo del videojuego y le permita familiarizarse con los controles de manipulación.
- Realizar el plan de implementación del software que permita instalar y capacitar al personal del Ministerio de Educación (MINED) y Ministerio de Medio Ambiente y Recursos Naturales en el adecuado uso de la aplicación informática.



## IMPORTANCIA Y JUSTIFICACIÓN

### IMPORTANCIA

A lo largo de este documento se hace énfasis a lo importante y necesario que es el concientizar a la población salvadoreña, sobre los daños que sus hábitos pueden ocasionar al medioambiente. Es innegable el hecho que debido a las acciones indiscriminadas del hombre, el deterioro del ecosistema es cada vez más acentuado, esto se percibe en fenómenos globales como el cambio en patrones meteorológicos (fenómeno Niño y la Niña), el aumento en la temperatura promedio global y otras problemáticas ambientales. La siguiente figura<sup>1</sup> muestra de forma gráfica el impacto a través de los años del alza en la temperatura.



Al tener una población concientizada, el gobierno se beneficiara con ahorros, ya que en un medio ambiente saludable, menos personas padecieran de enfermedades relacionadas a la contaminación ambiental (como las enfermedades comunes de las vías respiratorias) las que generan gastos en las instituciones de salud como el Hospital Bloom, Rosales, Instituto Salvadoreño del Seguro Social (ISSS) y Unidades de Salud. El beneficio no se limita a ahorros adicionalmente contribuye a mejorar la calidad de vida de los Salvadoreños.

El videojuego desarrollado aborda temas referentes a la correcta separación de desechos sólidos (aluminio, plástico, papel, biodegradables, otros.) para su reciclaje, concientización de respeto hacia especies en extinción, limpieza de los mantos acuíferos en el país, todo esto, promoverá los buenos hábitos en la población relacionados al ecosistema, mejorando la calidad de

<sup>1</sup>Gráfico tomado del sitio del Instituto para Políticas de la Tierra <http://www.earth-policy.org/Indicators/Temp/2004.htm>



aprendizaje y minimizando costos, pues se sabe que es mejor y más factible económicamente, el invertir dinero en prever catástrofes.

En entrevistas con los responsables de la Unidad de Educación Ambiental del MARN<sup>2</sup>, se manifestó que el desarrollo de material didáctico de este tipo sería de gran influencia en niños y jóvenes, constituyendo un importante aporte para los programas de educación ambiental que se desarrollan en convenio con el Ministerio de Educación, facilitando la labor del docente y reduciendo el tiempo en que los estudiantes asimilan la información.

El impacto del software en el medio ambiente será significativo, pues en base a los estudios realizados por la Universidad de California (UCLA) en 59 jóvenes, se determinó que los videojuegos pueden reducir en un 77% los malos hábitos<sup>3</sup>, modificando el comportamiento de sus usuarios de forma positiva y revirtiendo tendencias perjudiciales para el ser humano. Aprovechando las características benéficas de los videojuegos se formarán conductas a favor del uso equilibrado de los recursos naturales. Por lo cual se espera que se cambien las malas conductas hacia el medio ambiente y de esta forma se pueda revertir la degradación de los recursos naturales.

Desarrollar un software que concientice sobre las problemáticas ambientales es de trascendencia debido a que la degradación ambiental impone altos costos económicos y sociales, ya que afectan negativamente la salud, así como a la esperanza de vida, y en consecuencia reduce la recuperación de la inversión del capital humano y la tasa de crecimiento económico, debido a un aumento acelerado de los costos de salud por enfermedades respiratorias y gastrointestinales.

Además según el libro verde<sup>4</sup> de FUSADES, la degradación ambiental reduce la productividad de las tierras y aumentan los costos de producción, dañando la infraestructura económica y aumentando la vulnerabilidad del país ante los desastres naturales; también contribuye al incremento del valor de la canasta básica debido a la escasez de recursos naturales y pérdidas monetarias considerables a causa del cambio climático. La educación ambiental merece atención especial debido a que existen evidencias de que es la más efectiva para reducir los costos de la degradación ambiental de una nación.<sup>5</sup>

En la medida que exista una población estudiantil más educada en materia ambiental, se reducirán los costos económicos que implica, esto es urgente tratarlo, ya que mediante el estudio de campo se determinó que existe una deficiencia en el conocimiento de instituciones y métodos que velan por la preservación de los recursos naturales, solamente el 16.7% de los estudiantes en los niveles de cuarto grado a universitario conocen la correcta distribución de los desechos sólidos.

---

<sup>2</sup> Gerente de la Unidad de Educación Ambiental, Lic. Blanca Alarcón.

<sup>3</sup> Artículo, Los Videojuegos Cambian: tiempo de considerar 'Juegos Serios', Revistas para el desarrollo de la Educación, Pág 4. [http://www.netaid.org/documents/DEJ\\_article-Games\\_and\\_Development\\_Education-June05.pdf](http://www.netaid.org/documents/DEJ_article-Games_and_Development_Education-June05.pdf)

<sup>4</sup> Libro Verde de FUSADES versión electrónica <http://www.marn.gob.sv/CD1/Gestion/Nacional/Estado/desafio.html>

<sup>5</sup> Ibid, Pág. 20



El hecho de concientizar a un gran número de la población sobre el buen trato del Ecosistema del que forma parte, generará ahorros al gobierno ya que se espera:

Que se disminuyan los focos de contaminación cuando en el país se implemente la correcta separación de desechos sólidos y biodegradables, para su posterior reciclaje, dando como resultados, beneficios económicos; se aumente el turismo cuando todos en conjunto mantengamos ríos y playas limpias; se reducirá el riesgo de existir un desbalance en el medio ambiente al perderse un eslabón en las cadenas alimenticias, cuando se informe a las personas sobre la flora y fauna en peligro de extinción; se disminuyan gastos por enfermedades causadas por la contaminación del aire y del agua. Aspectos como estos, permitirá acortar costos en áreas como la salud, para poder ser invertidos en otras vitales como la educación y fondos para la vejez.

Cabe mencionar que en el parque Nacional Walter Deninger en La Libertad, existían cerca de 2,000 monos hace diez años, así como gacelas y venados, hoy la población es nula debido a incendios forestales causados por personas sin escrúpulos y cazadores que, debido a lo extenso del terreno y pocos recursos monetarios para vigilancia, han cazado indiscriminadamente a muchos animales hasta extinguirlos del parque<sup>6</sup>, se hace mención de esto, ya que una población educada respecto a la importancia de mantener las especies (ya que la cadena natural alimenticia de muchos animales beneficia directamente al ser humano) ininterrumpidas, es vital para la salud de todos.

En síntesis, el impacto económico radica en el hecho que a menores problemas medioambientales, menor inversión sería necesaria para evitar enfermedades y catástrofes o situaciones caóticas como la escasez del agua y alto precio de combustible basado en petróleo fósil.

---

<sup>6</sup> Información proporcionada por el guardabosques del Parque Walter Deninger, Febrero 15 2008.





## JUSTIFICACION

La necesidad de educar a la población Salvadoreña sobre el medioambiente es evidente y se manifiesta en los altos índices de contaminación y degradación ambiental reportados por instituciones nacionales e internacionales, algunos datos que evidencian esta situación son:

- Actualmente, El Salvador se encuentra en el segundo lugar entre los países Latinoamericanos con bajos índices de Desempeño Ambiental (EPI 2008), es uno de los 14 países del mundo (Según el Tribunal Centroamericano del agua) con mayor problema de escasez de agua.
- La producción de basura que actualmente es de 2,347.26 ton/día, del total de 182 municipios del país, solo el 69% de estos y el 63% de la población tiene servicio de recolección de basura<sup>7</sup>.
- Solamente el 3%, aproximadamente está cubierto de vegetación original y el 75% de los suelos están erosionados (Informe GEO 2002).
- La contaminación del aire, provoca enfermedades Infecciosas Respiratorias Agudas (IRAS), las cuales se cuantifican en el 80% de las consultas en los últimos tres años (Ministerio de Salud Pública y Asistencia Social, MSPAS).

El Ministerio de Medio Ambiente y Recursos Naturales es el ente encargado de difundir la información ambiental a la población Salvadoreña, pero esto representa grandes costos para el Estado. Proyectos como Guardianes ambientales han promovido la educación ambiental en 750,000 estudiantes del sistema educativo a un costo de \$713,539.<sup>8</sup>, en un período de cuatro años. Se realizan otros programas de concientización, pero se sabe que existe una carrera contra el tiempo, ya que según investigaciones realizadas por FUSADES, el costo anual de la degradación ambiental para el año 1997 era en promedio equivalente al 3.5% del PIB, valor monetario que oscila desde los 300 a 400 millones de dólares<sup>9</sup>; actualmente, expertos del Programa Salvadoreño de Investigación sobre Medio Ambiente (PRISMA), consideran que en base a las experiencias en otros países, el gasto aproximado por degradación ambiental actualmente ronda los \$500,000 millones anuales<sup>10</sup> cifra equivalente al 5% del PIB anual.

Actualmente, el MARN está a cargo de un programa de concientización ambiental, un trabajo en conjunto con el gobierno de la República y el Ministerio de Educación, dicho programa tiene una duración de seis meses y comenzó a ejecutarse el 5 de Mayo del presente año. En

<sup>7</sup> Informe sobre conferencia Mundial Sobre Reducción de Desastres ,<http://www.unisdr.org/eng/mdgs-drr/national-reports/El-Salvador-report.pdf>

<sup>8</sup> Informes Guardianes Ambientales hasta 2007, <http://www.fonaes.gob.sv/download/guardianes4.pdf>

<sup>9</sup> Libro Verde ,FUSADES,1997, <http://www.marn.gob.sv/CD1/Gestion/Nacional/Estado/desafio.html>

<sup>10</sup> <http://www.elsalvador.com/vertice/2003/130703/deportada.html>



entrevistas sostenidas con técnicos del MARN, se expresó la necesidad de educar a la población salvadoreña sobre el buen trato al medioambiente, haciendo uso de todas las herramientas o métodos posibles, a través de las cuales se logre concientizar sobre el cuidado del medio ambiente. Los oficiales y técnicos del MARN se mostraron entusiastas en promover su propaganda medioambientalista a través de un método, como ellos lo describieron ‘poco convencional’, al referirse a una demostración de tipo vertical slice<sup>11</sup> (corte vertical) del videojuego que les fue mostrada por los estudiantes desarrolladores.

Con los sistemas convencionales para diseminar la información a los estudiantes y ciudadanía en general, se necesita capacitar al recurso humano para brindar charlas, lo cual incurre en salarios, costos de impresión y elaboración de recurso didáctico, incurriendo también en otros gastos. También ha sido difícil llegar a toda la población estudiantil y darle continuidad al proceso de sensibilización, ya que por falta del recurso económico solamente se han realizado pruebas piloto en el área metropolitana. El desarrollo e implantación de este software reduciría el tiempo en que la información ambiental llegue a los estudiantes, ya que el videojuego podrá ser distribuido por medio de la intranet del MINED a las aulas informáticas<sup>12</sup>, por lo que fácilmente se diseminará la información ambiental en la población estudiantil y en la ciudadanía debido al efecto multiplicador. Además se reducirá el costo de capacitación del recurso humano ya que el personal que coordina las aulas informáticas son personas con amplio conocimientos en esta materia, igualmente los estudiantes, están familiarizados con la tecnología y con los videojuegos. Estas aseveraciones tienen fundamento en los datos obtenidos en la investigación de campo en diversas instituciones educativas, que revelan que el 83% de los estudiantes<sup>13</sup> tienen una predilección por lo videojuegos, más de la mitad de estudiantes juegan frecuentemente, por lo cual, la concientización será oportuna y continua. El software será tan intuitivo que cualquier persona con mínimos conocimientos en informática pueda utilizarlo.

Se tendrá un sistema de evaluación mediante puntajes, en el cual se verá reflejado el grado de aprendizaje sobre el correcto uso de los recursos naturales. Además, se aprovechara al máximo la tecnología con la que cuenta el sistema educativo de El Salvador.

La educación ambiental trae consigo una mejor calidad y aumento de la esperanza de vida, para la población Salvadoreña, mejor percepción e inversión por parte de la comunidad internacional y mejor salud en la ciudadanía.

---

<sup>11</sup> Vertical Slice en el contexto de software, se refiere a una demostración de software que no es el software terminado, y solo representa un pequeño porcentaje de este, sin embargo contiene o muestra las características del software en su totalidad.

<sup>12</sup> Método de distribución del software avalado por Lic. Julio Castellanos, Encargado de la INTRANET, Vice Ministerio de Tecnología, MINED, 2008.

<sup>13</sup> Anexo 7: Gráfico aceptación de videojuegos, Pág. 355



Por todo lo anteriormente expuesto, es de trascendencia contribuir al mejoramiento del sistema actual de educación ambiental, apoyando instituciones y programas que tienen como objetivo única preservar el medio ambiente de El Salvador.

El software será distribuido por el Ministerio de Medio Ambiente y Recursos Naturales (MARN), a la ciudadanía en general que cuente con un computador que cumpla con los requerimientos de hardware necesario para su operatividad. Por lo cual, el desarrollo de este software es justificable ya que brindará los siguientes beneficios tangibles:

- Se promoverá el buen uso de los recursos naturales en aproximadamente 812,68014 estudiantes del sistema educativo de El Salvador, a través de las Aulas Informáticas, en un periodo de 5 años.
- De acuerdo a los datos publicados en el VI Censo de Población V vivienda 2007, se tiene que existen 153,283 hogares que cuentan con un computador, haciendo un total de 630,969 personas que en sus hogares podrían implementar el software. De esta población no se tiene el número exacto de los que no son estudiantes, pero podrían ser alcanzados por este proyecto
- Apoyará el Sistema de Información Ambiental (SIA) del Ministerio de Medio Ambiente y Recursos Naturales (MARN), que pone a la disposición de los 14 departamentos de El Salvador, información medioambiental.

---

<sup>14</sup> Ver Sección Factibilidad Económica, Pág. 45



## **ALCANCES Y LIMITACIONES**

### **ALCANCES**

- El software solo será un recurso informativo referente a temáticas medio ambientales y podrá ser usado en todas las disciplinas e instituciones que promuevan la educación sobre el medio ambiente, prevención y mejoramiento de las condiciones medio ambientales actuales de nuestro país, así como también para individuos que lo soliciten al MARN.
- El software será diseñado para personas con edades de 7 años en adelante, debido a que es necesario para la manipulación del videojuego el dominio de la lectoescritura, que en promedio se da inicio a esta edad, a pesar de contener narraciones e instrucciones audibles en castellano.
- El proyecto comprenderá las fases de determinación de requerimientos, análisis del software, diseño de la solución, construcción y pruebas del software.
- Se elaboraran los manuales de instalación, configuración y técnico del software pro-medioambiente; así como un demo automatizado que muestre al usuario el manejo de dicho juego y pueda éste, familiarizarse con los controles de manipulación. También se presentara el respectivo plan de implantación para el MARN y MINED

### **LIMITACIONES**

- La información que maneje el software estará basada en los datos proporcionados por el Ministerio de Educación y Ministerio de Medio Ambiente y Recursos Naturales.
- El software únicamente educara sobre las principales problemáticas ambientales y el grado de concientización que el jugador adquiera, se verá influenciado por factores psicológicos y sociales.
- La implantación del sistema y todos los gastos que esta conlleve, será responsabilidad del MARN y MINED.



# **CAPITULO I: ESTUDIO PRELIMINAR**



## 1.1. ANTECEDENTES

### 1.1.1. Origen de la Educación Ambiental en El Salvador

En los últimos años la población se ha conmovido con desastres naturales que han afectado al país, como en el caso del huracán Mitch que dejó daños estimados en 388 millones de dólares equivalentes al 6.4% del Producto Interno Bruto Nacional<sup>15</sup>, este tipo de fenómenos naturales incrementan su magnitud debido al calentamiento global originado por los niveles de gases invernadero en la atmósfera<sup>16</sup>.

A nivel internacional, una de las primeras organizaciones en preocuparse por los problemas ambientales ha sido la Organización de las Naciones Unidas (ONU), quien en 1972 creó el Programa de las Naciones Unidas para el Medio Ambiente (PNUMA) y adoptó la siguiente definición de educación ambiental (EA): "Es la educación que pretende brindar a los individuos, elementos necesarios para realizar un análisis crítico de las condiciones de su medio ambiente, permitiendo identificar los principales problemas ambientales y buscar la participación en la solución a los mismos."

Debido a las recomendaciones de este organismo internacional, la Dirección de Patrimonio Natural del Ministerio de Educación tomó interés en la EA, creando a finales de la década de los años 80 un departamento en esta área, con lo cual se logró la capacitación de técnicos de instituciones públicas y privadas.

En el periodo de 1994 y 1999, inició el Proyecto Protección de Medio Ambiente Salvadoreño (PROMESA), que incorporó el componente de Educación Ambiental, a nivel formal e informal. Las principales actividades de este proyecto se enmarcaron en cuatro estrategias: apoyo a la reforma curricular, elaboración de materiales educativos, capacitación de maestros y proyectos escolares. Paralelamente, el Fondo Iniciativa de las Américas de El Salvador (FIAES) y el Fondo Ambiental de El Salvador (FONAES) han financiado desde la década de los años 90 diversos proyectos relacionados con la sensibilización mediante actividades ambientalistas, charlas, reforestación, ferias y otras acciones destinadas a concientizar a la población.

Hasta 1991, El Salvador no contaba con ninguna instancia gubernamental encargada del tema ambiental, ni con legislaciones relacionadas, siendo la única referencia cercana a la

---

<sup>15</sup> COEN, Comité de Emergencia Nacional, [http://www.reliefweb.int/library/documents/paho\\_hurr\\_7elsalv.pdf](http://www.reliefweb.int/library/documents/paho_hurr_7elsalv.pdf).

<sup>16</sup> Burbujas de aire encontradas en hielo milenario de la antártica muestran que son los niveles más altos en 600,500 años. <http://data.giss.nasa.gov/gistemp/graphs/>



Educación Ambiental, la enunciada en la Constitución Política<sup>17</sup> referente a impartir enseñanza sobre la protección de los recursos naturales.

En ese año se conformó la Secretaría Ejecutiva del Medio Ambiente (SEMA), dependencia del Órgano Ejecutivo, responsable de coordinar, y velar por el cumplimiento de las políticas y estrategias sobre el medio ambiente Salvadoreño.

SEMA, incluyó en su estructura la Coordinación de Educación Ambiental, y en 1994, presentó la Estrategia Nacional del Medio Ambiente, documento en el que se propuso la estrategia para educar ambientalmente a los salvadoreños, la cual contemplaba: Incorporar la dimensión ambiental en el Sistema Educativo Nacional.

En 1997 se creó el Ministerio de Medio Ambiente y Recursos Naturales con el objetivo de velar por el cumplimiento de leyes ambientales que pretenden preservar los recursos naturales de El Salvador, tal es el caso de la Ley del Medio Ambiente donde se establece la necesidad de llevar la EA a la probación en general. El país se ha suscrito a nivel regional e internacional, a diversos tratados, protocolos y convenios enfocados en preservar el medio ambiente.

El Cuadro1.1 presenta algunos convenios adoptados por El Salvador que promovieron la integración de la EA como un eje transversal de la educación formal.

| Año  | Convenio – tratados   |
|------|---|
| 1998 | Ratificación del Protocolo de Kioto (PK)  |
| 2001 | Convención de Estocolmo sobre Compuestos Orgánicos Persistentes, POPs   |
| 2000 | Suscripción a los Objetivos del Milenio (ODM) en la Cumbre del Milenio celebrada en la sede de las Naciones Unidas. El objetivo número siete es mantener la sostenibilidad ambiental. El Protocolo de Cartagena sobre Bioseguridad de la Biotecnología. |
| 2002 | Acuerdo entre los Estados Miembros y los Miembros Asociados de la Asociación de Estados del Caribe para la Cooperación Regional en Materia de Desastres Naturales.  |
| 2005 | 18 de Febrero, se firmo el Tratado de Libre Comercio suscrito entre República Dominicana, Centroamérica y Los Estados Unidos de América conocido como DR-CAFTA con convenios ambientales.   |

Cuadro1.1: Convenios adquiridos por El Salvador

<sup>17</sup>Art. 60 “En todos los centros docentes, públicos o privados, civiles o militares, será obligatoria la enseñanza de la historia nacional, el civismo, la moral, la Constitución de la República, los derechos humanos y la conservación de los recursos naturales”, Constitución Política 1983



Los tratados, convenios y políticas ambientales, comprometen al gobierno a:

- Incorporar los principios del desarrollo sostenible en las políticas y los programas nacionales e invertir la pérdida de recursos del medio ambiente.
- Reducir a la mitad, el porcentaje de personas que carezcan de acceso a agua potable.
- Apoyo a iniciativas en la implementación de prácticas ambientales sustentables y responsables, a través de la concientización y diseminación de información en las escuelas, comunidades y empresas.
- Promover acciones de voluntariado en las comunidades para educar y sensibilizar a la población, con interferencia directa en las asociaciones y órganos representativos, escuelas, parques, etc.
- Apoyo a proyectos de búsqueda y formación en el área ambiental; promoción de concursos internos o locales que estimulen el debate y la concientización individual sobre el medio ambiente y la importancia de la colaboración de cada uno.

En concordancia con los compromisos asumidos a nivel regional e internacional, el gobierno ha incorporado instrumentos jurídicos como la Política Nacional de Educación Ambiental (PNEA), donde se exponen las estrategias a seguir para concientizar sobre la problemática ambiental a la población Salvadoreña a nivel formal, no formal e informal.

### **1.1.2. Instituciones Co-responsables de la Educación Ambiental**

En La Política Nacional de Educación Ambiental se establece que: “El Ministerio de Medio Ambiente y Recursos Naturales y Ministerio de Educación son las entidades líderes, responsables de la Educación Ambiental<sup>18</sup>.”

Actualmente existen convenios entre ambas instituciones gubernamentales, para fortalecer esta disciplina a nivel formal.

#### **1.1.2.1. Ministerio de Medio Ambiente**

Este Ministerio, tiene como objetivo principal promover la protección del medio ambiente y el uso racional de los recursos naturales a través del desarrollo de políticas, estrategias, marco legal y otros instrumentos que propicien cambios de comportamiento en la sociedad.

---

<sup>18</sup> Lineamientos Estratégicos para Implementar la PNEA, PNEA, Pág.15.





### Misión

Dirigir una gestión ambiental efectiva a través de políticas y normas que en forma participativa y transparente faciliten el desarrollo sostenible de la sociedad Salvadoreña.

### Visión

Establecer las bases para armonizar el desarrollo social y económico con el manejo sostenible de los recursos naturales.

### Organización

El MARN en su estructura organizativa<sup>19</sup> tiene 6 direcciones generales que son: Asuntos Jurídicos, Cooperación y Asuntos Internacionales Ambientales, Inspectoría Ambiental, Patrimonio Natural, Gestión Ambiental y Participación Ciudadana; además cuentan con el apoyo de las gerencias de Auditoría Interna, Comunicaciones, Administrativa y Financiera Institucional; El Fondo Ambiental de El Salvador (FONAES) y el Servicio Nacional de Estudios Territoriales (SNET), son entes adscritos a este Ministerio.

La Dirección General de Participación Ciudadana es la que incorpora en su estructura organizativa<sup>20</sup> la Gerencia de Educación Ambiental, unidad encargada de llevar a la población información ambiental a través de los sectores: campesino, organizaciones no Gubernamentales, comités ambientales empresariales, comités ambientales departamentales, municipalidades, hombres y mujeres líderes a nivel local y otros involucradas en la gestión ambiental.

Esta gerencia realiza las siguientes acciones:

- Coordinación Interinstitucional
- Consulta pública.
- Capacitaciones en gestión ambiental
- Poner a la disposición de la ciudadanía el Centro de Información y Documentación Ambiental

### Responsabilidades en cuanto a Educación Ambiental Formal

- Apoyar al Ministerio de Educación en sensibilizar y capacitar a Supervisores, Directores, Técnicos y Docentes para incorporar la cultura ambientalista.
- Brindar, dentro de sus capacidades, asistencia técnica en la planificación y seguimiento de planes, programas, proyectos y acciones ambientalistas.

---

<sup>19</sup> Ver Anexo No. 1: Organigrama MARN, Pág. 347.

<sup>20</sup> Ver Anexo No. 2: Organigrama de la Dirección de Participación Ciudadana, Pág. 348.



- Promover la aplicación y desarrollo de métodos de enseñanza aprendizaje que conlleven a la práctica de conservación y recuperación del Medio Ambiente en la comunidad educativa.
- Establecer planes, programas, proyectos y acciones que promuevan la participación conjunta del MARN–MINED y las comunidades rurales y urbanas, en actividades destinadas a la conservación y recuperación de los recursos naturales.
- Propiciar la divulgación de información, documentos técnicos, legales y didácticos, que lleven a la conservación del medio ambiente.

### **1.1.2.2. Ministerio de Educación**

Esta institución establece las políticas y los lineamientos para dotar al sector educativo de un servicio de calidad, su objetivo es mejorar los servicios educativos mediante el fortalecimiento del desempeño de directores y docentes, la introducción de tecnologías educativas y la transferencia de recursos financieros a los centros educativos.

#### Misión

Contribuir y promover el desarrollo integral de la persona en sus dimensiones moral, individual y social, garantizando una educación de calidad.

#### Visión

Ser modelo internacional como institución pública de alto rendimiento que en equipo, trabaje en un proceso de mejora continua que garantice el pleno acceso de la población salvadoreña a servicios educativos de excelencia, que le habiliten para la vida y propicien el desarrollo integral del país.

#### Organización

En cuanto a este Ministerio su estructura organizativa<sup>21</sup> es grande, por lo cual nos centraremos en describir las Direcciones, Gerencias y Unidades que tienen relevancia en este estudio.

Bajo la Gerencia de Programas Complementarios de la Dirección Nacional de Educación El MINED ha incorporado la Unidad de Educación Ambiental y Gestión de Riesgos, encargada de impulsar los mecanismos orientados al fortalecimiento del tema ambiental como eje transversal en el plan curricular de educación y que se le dé prioridad a la dimensión ambiental en los planes y programas educativos.

---

<sup>21</sup> Ver Anexo No 3: Organigrama de Ministerio de Educación, Pág. 349.



El Viceministerio de Tecnología adscrito al Ministerio de Educación es el encargado de llevar las TIC a los centros escolares por medio de diversos programas, específicamente es de importancia mencionar el programa Aulas Informáticas que beneficia a algunos centros escolares, proporcionándoles centros de computo para la comunidad estudiantil.

#### Responsabilidades en cuanto a Educación Ambiental

- Fortalecer el aprendizaje y la práctica de actitudes y comportamientos orientados a conservar y recuperar el medio ambiente a través del sistema educativo nacional.
- Designar en su interior la Unidad responsable que supervise, coordine y de seguimiento a las políticas, planes, programas, proyectos y acciones ambientalistas internas y externas que vele por el cumplimiento de las normas ambientales; así como asegurar la necesaria coordinación interinstitucional en la gestión ambiental educativa.
- Facilitar la participación del personal técnico para la ejecución y seguimiento de las políticas, planes, programas y proyectos.
- Fortalecer dentro del servicio de las Unidades Móviles y del Programa de Bibliotecas Rodantes (cajas viajeras y bibliobús), la difusión de la Educación Ambiental.
- Apoyar al Ministerio de Medio ambiente y Recursos Naturales, en la divulgación de información, documentos técnicos, legales y didácticos.

El MINED ha creado en su estructura organizativa la Unidad Ambiental, encargada de impulsar los mecanismos orientados al fortalecimiento del tema ambiental como eje transversal en el plan curricular de educación.

#### **1.1.3. Marco Legal**

En materia de Educación Ambiental la labor que realiza el Ministerio de Medio Ambiente y Recursos Naturales (MARN) se rige en la Constitución de la República de El Salvador y La Ley de Medio Ambiente.

La Constitución de la República de El Salvador en su artículo 117, dispone literalmente que: “Es deber del Estado proteger los recursos naturales, así como la diversidad e integridad del medio ambiente para garantizar el desarrollo sostenible y declara de interés social la protección, conservación, aprovechamiento racional y restauración de los recursos naturales.”



En el artículo 4 de La Ley de Medio Ambiente, se especifica que “Se declara de interés social la protección y mejoramiento del medio ambiente. Las instituciones públicas o municipales, están obligadas a incluir, de forma prioritaria en todas sus acciones, planes y programas, el componente ambiental. El Gobierno es responsable de introducir medidas que den una valoración económica adecuada al medio ambiente acorde con el valor real de los recursos naturales, asignando los derechos de explotación de los mismos de forma tal que el ciudadano al adquirirlos, los use con responsabilidad y de forma sostenible”.

De conformidad con el artículo 2, literales h) y m) de la citada Ley, la gestión pública del medio ambiente debe ser compartida por las distintas instituciones del Estado. Los artículos 4, 6 y 7 de la Ley del Medio Ambiente, establecen que el Ministerio de Educación debe, incluir de forma prioritaria en todas sus acciones, planes y programas el componente ambiental.

El MARN y el MINED, están comprometidos<sup>22</sup> a buscar mecanismos de coordinación, planificación y operación, a fin de incorporar la dimensión ambiental en las prácticas para la obtención de cualquier título académico, promover y desarrollar investigación tecnológica.

En cuanto a la información que el MARN disemina en la población Salvadoreña, existen normativas en el reglamento de normas técnicas de control interno con respecto a los sistemas de información y comunicación, donde se define que los sistemas de información deben procesar información confiable oportuna, suficiente y pertinente (Art. 38); además la información debe apegarse a los objetivos institucionales, ajustándose a sus características y necesidades (Art. 35)

Los procesos realizados por ambas instituciones referentes a la educación ambiental son regulados por La política Nacional de Educación Ambiental (PNEA), donde se establece que el material didáctico debe ser actualizado, contextualizado y propicio de ser usado por los estudiantes de los diferentes niveles educativos.<sup>23</sup>

En ninguna de estas leyes se hace referencia explícita al desarrollo de software informático, solamente de manera implícita a las características que la información llevada a la población por medio del software debe cumplir.

En las Aulas Informáticas no existe ninguna ley que prohíba el uso de videojuegos educativos, el Ministerio de Medio Ambiente por medio de la Gerencia de Medio Ambiente ha expresado estar a favor del desarrollo de este software y no encuentran ningún impedimento de carácter legal para su realización e implementación, por el contrario basados en la PNEA sección V literal B inciso 4 y 5 manifiestan que este proyecto es avalado por dicha política.

---

<sup>22</sup> Aseveración fundamentada en los artículos 39, 40 y 41 de la Ley del Medio Ambiente El Salvador.

<sup>23</sup> La Política Nacional de Educación Ambiental, Aprobada en el 2006, Pág. 15, [http://www.fonaes.gob.sv/download/politica\\_nac.pdf](http://www.fonaes.gob.sv/download/politica_nac.pdf)



#### 1.1.4. Aulas Informáticas en Centros Escolares de El Salvador

Desde hace más de una década en el país se comenzó a integrar la tecnología con la educación, una de las primeras instituciones en incursionar en esta área fue FutureKids, empresa dedicada a brindar capacitación a docentes, cursos, certificaciones y llevar las TIC a las escuelas.

Esta empresa proporciona desde los años 90 software a las instituciones educativas salvadoreñas, sobre todo a las privadas, desde la creación de las aulas informáticas conocidas anteriormente como Centro de Recurso y Aprendizaje (CRA), el software interactivo es más accesible para la población estudiantil de pocos recursos económicos.

Actualmente las aulas informáticas, consisten de un conjunto de equipos y herramientas de tecnología, comunicaciones, audiovisuales y otros con el objetivo de apoyar el proceso de enseñanza-aprendizaje en el sistema educativo a nivel de educación básica y media.

Este conjunto de equipos incluye: equipo de informática (servidores, computadores personales, impresores, accesorios), redes de datos, enlace a Internet, infraestructura física, aire acondicionado, proyectores de cañón, etc.

Las Aulas Informáticas forman parte del programa CONECTATE que ha sido impulsado por el Viceministerio de Tecnología del Ministerio de Educación en el contexto del plan 2021. En cada centro escolar se ha construido un aula especial donde se han hecho instalaciones eléctricas, polarizaciones, instalaciones de aires acondicionados para brindar un mejor ambiente al equipo que se instala, siendo acomodadas 20 a 25 computadoras con programas informáticos básicos.

El software que se proporciona a cada aula informática<sup>24</sup> es el siguiente:

- Enciclopedia Encarta
- Micronet
- Matemática y reto matemático
- Tablas de multiplicar
- Naturales y ejercicios de procesos
- Geografía con Pipo

Los docentes de cada Aula Informática son los responsables de descargar software educativo u otro material de libre distribución que refuerce el aprendizaje de las diferentes asignaturas. El Ministerio de Educación no cuenta con un registro estadístico de las aplicaciones educativas que se usan en las aulas informáticas.

---

<sup>24</sup> Información Proporcionada por la Unidad de asistencia técnica -pedagógica de la dirección tecnológica educativa



### 1.1.5. Software Interactivo

La interacción, según el Diccionario de la Real Academia Española, significa “la acción que se ejerce recíprocamente entre dos o más objetos”, entonces, con interactividad, se hace referencia al proceso cognitivo por el cual el usuario comprende y cambia su entendimiento, sus perspectivas y estructura de pensamiento.

Se entenderá como software interactivo, a los programas informáticos creados con la finalidad de que haya interacción del usuario con la aplicación, este término no incluye a los paquetes donde el usuario es solo un observador pasivo. La interactividad en este tipo de aplicaciones informáticas, no se reduce a la simple manipulación del usuario, es decir pulsar botones o pinchar con el ratón para acceder a otra ventana, sino más bien, hace alusión a el software al que el usuario puede preguntar, poner sus parámetros, opiniones y a su vez la aplicación se ejecute.

Los niveles de interactividad según Orozco<sup>25</sup> son:

- Primer nivel la interactividad significa fundamentalmente “selección” de una cierta información dentro de un amplio universo de posibilidades informativas.
- En un segundo nivel, la interactividad se complejiza incorporando asociaciones, pero es todavía una actividad esencialmente comandada por la computadora.
- Es hasta un tercer nivel que la interactividad es conducida por el propio usuario, de acuerdo a los propósitos particulares que tenga, en este nivel el usuario puede modificar.
- Para el cuarto nivel de interactividad ya es posible una actividad realmente creativa, de diseño, que permite también producir, se dan los escenarios simulados, de actividades posibles en secuencias de acciones, de intercambios no sólo con la pantalla, y que en el caso de los videojuegos, participan también como jugadores en la competencia por llegar primero a las metas.

Según esta clasificación los videojuegos se encuentran en el mayor nivel de interactividad.

---

<sup>25</sup>Orozco Gómez Guillermo ,III Congreso Educare de Aprendiendo con Videojuegos,Pág.10-14,  
[http://www.aurora.ufsc.br/artigos/orozco\\_congresso\\_educarede.pdf](http://www.aurora.ufsc.br/artigos/orozco_congresso_educarede.pdf)



### 1.1.5.1. Videojuegos

Un videojuego (VJ) es un juego para computador y, por extensión, para videoconsolas, que consiste en un entorno virtual programado, para que los jugadores puedan conseguir un objetivo<sup>26</sup>.

#### Breve Historia

En 1952, A.S. Douglas escribió su primera tesis para optar por un doctorado en la Universidad de Cambridge sobre la interacción Humano-Computadora. Douglas creó su primer juego gráfico, una versión electrónica de Tic-Tac-Toe (equis-cero). El juego fue programado en una computadora de tubo al vacío, una EDSAC, la cual implementaba un tubo de rayos catódicos para desplegar datos en una pantalla.

William Higinbotham creó el primer juego de video en la historia del mundo en 1958, su juego llamado "Tenis para Dos" fue creado y jugado utilizando un osciloscopio de laboratorio de la Brookhaven National. En 1962, Steve Russel inventó SpaceWar! (guerra espacial!) y fue el primer juego creado para ser usado en una computadora.

En 1967, Ralph Baer escribió su primer juego de video que se podía jugar en un televisor, un juego llamado Chase. En 1971, Nolan Bushnell junto con Ted Dadey, crearon el primer juego arcade<sup>27</sup>, basados en la versión de Steve Russell de Spacewar!. El juego arcade *Pong* fue creado por Nolan Bushnell con la ayuda de Al Alcorn un año después en 1972. Nolan Bushnell y Ted Dadey fueron los empresarios iniciadores de Atari Computers ese mismo año. Atari re-lanzó el juego Pong para uso doméstico en televisores.

En 1972, la primera consola de juegos llegó al Mercado, exclusiva para juegos en televisores, La Odyssey, fue lanzada por Magnavox y diseñada por Ralph Baer. La Odyssey contenía 12 juegos pre-programados.

En 1976, Fairchild lanzó la primera consola programable casera, llamada Sistema de Entretenimiento Fairchild y luego fue renombrada a Channel F. Este fue uno de los primeros sistemas en utilizar el recién inventado microprocesador, producto de Robert Noyce y exclusivo de la Corporación de Semiconductores Fairchild que permitía que los juegos no fueran limitados por el número de interruptores y compuertas lógicas.

El 17 de junio de 1980, los juegos "Asteroids" y "Lunar Lander" de Atari, fueron los dos primeros videojuegos en ser registrados con derechos de autor.

---

<sup>26</sup> <http://www.arcadia.co.cr/Default.aspx?tabid=633>

<sup>27</sup> Arcade: Utiliza monedas para su funcionamiento.



Las computadoras principales para juegos hicieron su aparición a principio de los años 1980s, en 1982 apareció la Commodore 64, Apple II (mejorada, ya que existía desde 1977) y la computadora ZX Spectrum al igual que las Atari 800, por 15 años la Apple II tuvo un total de 20,000 programas diseñados para esta, convirtiéndola en la computadora de 8 bits con el mayor software hasta entonces.

### Consolas de Tercera Generación (1985–1989)

En 1984, las computadoras ya competían con las consolas de juegos, ya que las prestaciones y aumento en ventas de juegos hacían que las personas utilizaran sus PCs para ejecutar videojuegos.

En 1985, el Mercado de consolas de video fue revivido con la aparición de la consola de 8 bits NES (Nintendo Entertainment System, Figura 1.1) junto con Super Mario Bros. Esta consola se convirtió en todo un éxito y marco el preámbulo para las consolas de cuarta generación de los 90s.



Figura 1.1: Consola de 4ta. Nintendo Entertainment System.

Consolas como Sega Génesis, y luego PlayStation entre las más populares, hicieron su aparición junto con dispositivos de mano como el famoso gameboy. Las empresas desarrolladores de juegos se proliferaron y la competencia se vio en aumento, con la limitante de los profesionales desarrolladores y programadores de los juegos de video. Ya en los años 90 existía en los Estados Unidos, universidades con carreras exclusivas de programación 2D y 3D para juegos comerciales.

Hoy en día, y con la tecnología de 64 bits a disposición de los ingenieros de juegos de video, consolas con grandes prestaciones han hecho su aparición en el mercado, estas son las consolas de Séptima generación, entre las cuales se tiene Nintendo Wii, Playstation 3, Gamebox Micro y Xbox-360<sup>28</sup>. (Esta última, es la considerada idónea para la programación del juego de video aquí propuesto, por ser exclusivamente programable en visual C# 2005-2008 y permitir crear versiones para PC o Xbox fácilmente).

### Los Videojuego y la Salud

Packy & Marlon (Figura 1.2) es un juego diseñado con la intención de servir de utilidad a adolescentes diabéticos, eligiendo comidas saludables, en el juego, el usuario toma el rol de un

---

<sup>28</sup> Texto tomado y traducido exclusivamente para la presentación de este anteproyecto desde el los sitio web:  
[http://inventors.about.com/library/inventors/blcomputer\\_video\\_games.htm](http://inventors.about.com/library/inventors/blcomputer_video_games.htm),  
[http://en.wikipedia.org/wiki/History\\_of\\_computer\\_and\\_video\\_ga](http://en.wikipedia.org/wiki/History_of_computer_and_video_ga)





personaje que debe controlar su diabetes, comprobando el nivel de glucosa en sangre, inyectándose insulina mientras intenta liberar un campamento de verano de ratas y ratones que roban las medicinas necesarias para los diabéticos.

El estudio fue realizado a 59 pacientes de 8 a 16 años. A 31 le dieron el juego de Packy & Marlon y a 28 le dieron un juego que no tenía nada que ver con el tratamiento de la diabetes. Los padres rellenaban un cuestionario cada 3 meses. Los resultados de los niños que jugaron al Packy & Marlon demostraron la eficiencia del juego.

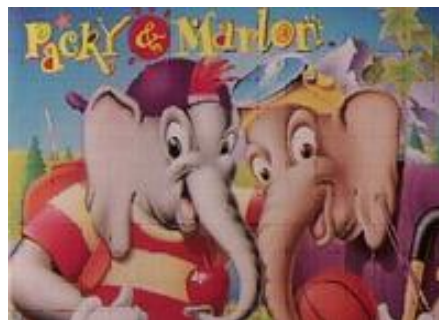


Figura 1.2: Packy & Marlon

Había aumentado el nivel de comunicación con los padres y de autocontrol por parte del paciente, mientras que el nivel de visitas de urgencia debido a la diabetes había disminuido en un 77%<sup>29</sup>.

### Los Videojuego y la Educación

Existen actualmente estudiosos de las bondades de los videojuegos que afirman de acuerdo a sus investigaciones que son herramientas idóneas para la enseñanza, Gros afirma: “Desde el punto de vista intelectual, la complejidad de la mayor parte de los juegos de ordenador actuales permiten desarrollar no sólo aspectos motrices sino, sobre todo, procedimientos tales como las habilidades para la resolución de problemas, la toma de decisiones, la búsqueda de información, la organización, etc. Desde el punto de vista afectivo, los juegos ejercen una importante motivación y pueden utilizarse para el trabajo de aspectos relativos a la autoestima.”<sup>30</sup>

Se han realizado estudios que revelan la gran efectividad de los videojuegos como metodología de enseñanza alguno de estos son:

La ELSPA (“The European Leisure Software Publishers Association”) ha publicado un informe titulado “Aprendizaje sin límites: En donde un colegio de Nottinghamshire, se incorporó el juego Neverwinter Nights en el plan de estudio, lo que disparó el porcentaje de éxito en las principales materias a un 94%, frente a la media nacional de 22%<sup>31</sup>, este estudio se realizó en 1,500 alumnos.

<sup>29</sup> Lieberman, videojuego para la Educación diabetes juvenil: Resultados de un ensayo controlado, Informática Médica, 1997, Pág. 77-89.

<sup>30</sup> Begoña Gros Salvat, La dimensión socioeducativa de los videojuegos, Pág. 8, <http://www.uib.es/depart/gte/edutec-e/Revelec12/gros.html>

<sup>31</sup> ELSPA, Aprendizaje ilimitado de informática y videojuegos en el aprendizaje del paisaje, Pág 30, [www.elspa.com/assets/files/u/unlimitedlearningtheroleofcomputerandvideogamesint\\_344.pdf](http://www.elspa.com/assets/files/u/unlimitedlearningtheroleofcomputerandvideogamesint_344.pdf)



### Videojuego para el Cambio de Actitud

En el ámbito internacional, se han desarrollado diferentes videojuegos para promover el cambio de conductas en diferentes temáticas, algunos de estos se presentan a continuación. *Power Up*, el desarrollo de este videojuego (Figura1.3) se realizó en San José California, 200 niños y adolescentes de 12 a 16 años actuaron de asesores para los investigadores de IBM durante su desarrollo.

A través de mundos virtuales se alienta a aprender sobre principios de la ingeniería, mientras manejan por escarpadas montañas, para construir torres solares o buscan piezas en oscuros depósitos de chatarra para reparar turbinas de viento. También se aprende sobre la conservación de la energía de acuerdo a las opciones que seleccionen para llevar a cabo sus misiones. El juego también tiene personajes que representan un corte transversal diverso de la población y sirven de modelos de roles, actúan con



Figura1.3: Videojuego Power Up

*Food Force*, videojuego creado por el Programa Mundial de Alimentos de la ONU, el escenario del videojuego es una isla ficticia llamada Sheyland, devastada por una guerra. El juego consiste en repartir víveres en zonas afectadas por el conflicto desde un avión que lanza las provisiones y desde camiones que se desplazan en caminos destruidos y riesgosos para llegar a la población hambrienta. En medio del viaje, los jugadores, que serían los voluntarios de Naciones Unidas que llevan a cabo la operación humanitaria, se encuentran con las mismas dificultades que en la realidad, como puentes intransitables, caminos de puro fango o minados y pasos tomados por grupos rebeldes. El objetivo siempre es socializar de forma equitativa los alimentos entre las víctimas del conflicto.

Con los resultados expuestos anteriormente, se puede concluir que los videojuegos podrían considerarse como elemento potencial, que, en el marco de ciertas y diversas regulaciones, ofrecen al usuario la oportunidad de entrar a nuevas formas de diversión que estimulan, fortalecen las áreas de desarrollo, aprendizaje, desenvolvimiento social,



Figura1.4: FoodForce. Videojuego de la ONU

efectivo, cognitivo y psicomotor, por lo que merece un estudio de los mismos, con el propósito de extraer sus bondades y ofrecerlos a la masa poblacional que demanda nuevas formas de entretenimiento, ocio, juego y diversión en el contexto tecnológico actual.



## 1.2. SOLUCION DEL PROBLEMA

### 1.2.1. Reconocimiento del problema

Cuando alguien recorre el país, puede observar claramente el deterioro que sufre el medio ambiente, debido a los malos hábitos poblacionales, entre los problemas ambientales más evidentes se pueden mencionar la deforestación que implica una serie de cambios ecológicos como por ejemplo: La pérdida de la biodiversidad, cambios de temperatura, pérdidas de hábitats y nichos ecológicos, migración de especies y, lo más delicado, la presión que se ejerce sobre las especies hacia su extinción.

El aire, se encuentra altamente contaminado por gases de efecto invernadero (CO<sub>2</sub>), plomo, óxido de azufre, óxido nítrico, ácido nítrico, que con la lluvia ácida son arrastrados hacia el suelo, creando problemas en los diferentes ecosistemas como por ejemplo: acidez en los suelos, deterioro en edificios y casas.

Los mantos acuíferos de nuestro país se encuentran en su mayoría contaminados por aguas negras provenientes de las grandes ciudades, provocando la disminución de especies marítimas, turismo y consumo del vital líquido.

Ante este problema, surge la necesidad de lograr cambios de actitud en el niño y el joven con respecto a los problemas ambientales que influyen negativamente en la vida misma, esto solo es posible por medio de la educación ambiental.

#### 1.2.1.1. Factores que intervienen en el problema

Según las entrevistas<sup>32</sup> con especialistas en Educación Ambiental (Técnicos del MARN) los factores que influyen en la problemática que afronta la educación ambiental son:

- **Recurso humano:** Personal encargado de impartir las temáticas ambientales en el sistema de educación Nacional, muchas veces no lleva la teoría a la práctica.
- **Recurso didáctico:** El material de apoyo utilizado actualmente para impartir las clases no es innovador.
- **Recurso tecnológico:** No se usa toda la tecnología disponible en ciertos centros escolares para profundizar en la enseñanza de niños y jóvenes.
- **Programa de asignatura de Ciencia Salud y Medio Ambiente:** Es reciente la incorporación de temáticas ambientales como: Reciclaje y consumo responsable.
- **Tiempo:** El periodo en el que se refuerza el conocimiento ambiental no es constante.

---

<sup>32</sup>Ver Anexo No. 5: Instrumentos de investigación, Pág. 351.



- **Metodología de enseñanza:** La obsolescencia o el estancamiento de las metodologías se convierte en un paradigma que afecta a la enseñanza-aprendizaje rápido y fácil por parte de los educandos, siendo la metodología una de las situaciones claves para impartir las clases, esta situación se vuelve un obstáculo en ocasiones.
- **Recurso pedagógico:** En ocasiones el enfoque utilizado por los maestros no es del todo desarrollado para que los alumnos comprendan los contenidos, sino para que se memoricen los mismos, esto implica que después de un tiempo lo aprendido es olvidado por qué no se profundizó en la comprensión y peor aún no se lleva a la práctica lo recibido.
- **Infraestructura:** Aulas, centros de computo y centros escolares donde se imparten las clases.
- **Grupo familiar:** El no involucramiento del grupo familiar en forma positiva en el proceso enseñanza-aprendizaje se vuelve un problema que afecta en la educación del alumno, por esta falta de atención el educando reduce el rendimiento académico.
- **El estudiante:** Los alumnos resultan ser el problema principal cuando estos no tienen una atención personalizada y el apoyo necesario, desmotivándolos a la hora del aprendizaje.

El cuadro 1.2, presenta de forma detallada las relaciones entre los factores involucrados en la problemática ambiental, la lectura de este cuadro se realiza de izquierda a derecha, a continuación se ejemplifica su lectura con el elemento "Recurso didáctico":

- "Recurso didáctico" proporciona ayuda en el soporte del "Recurso Tecnológico".
- "Recurso didáctico" proporciona apoyo en aprendizaje de conocimientos al "programa de la asignatura Ciencia Salud y medio ambiente"
- "Recurso didáctico" proporciona soporte a la metodología de enseñanza



|   | Recurso Didáctico                    | Recurso Tecnológico              | Recurso Humano  | Programa de asignatura de Ciencia Salud y Medio Ambiente | Tiempo             | Metodología de Enseñanza    | Recurso Pedagógico       | Infraestructura            | Grupo Familiar                          | Estudiante  |
|---|--------------------------------------|----------------------------------|---|--|--------------------|-----------------------------|--------------------------|----------------------------|---|---|
| <b>Recurso Didáctico</b>  | Actualización                        | Ayuda al Soporte                 | Material de Apoyo                                       | Apoya el aprendizaje de los contenidos                   | Ahorro             | Soporte                     | Complementa              | Equipa                     | Concientización en los hogares          | Medio de Aprendizaje  |
| <b>Recurso Tecnológico</b>                                      | Se complementan en la enseñanza      | Actualiza                        | Herramienta de apoyo                                    | Facilita su enseñanza                                    | Ahorro             | Nuevos métodos de enseñanza | Complementa              | Equipa                     | Concientización en los hogares          | -Nuevas herramientas de apoyo<br>-Motivación al aprendizaje<br>- Innovación en el aprendizaje |
| <b>Recurso Humano</b>   | Utiliza                              | Utiliza                          | Capacitación Cooperativa                                | Desarrolla Aplica  | Administra         | Aplicación                  | Aplicación               | Utiliza Mantenimiento      | Informa Sugerencias                     | Enseñanza   |
| <b>Programa de asignatura de Ciencia Salud y Medio Ambiente</b> | Utiliza                              | Utiliza                          | Provee una guía de contenidos                           | Retroalimenta Mejora                                     | Se actualiza       | Cumple objetivos            | Utiliza                  |                            | Concientiza                             | Se desarrolla   |
| <b>Tiempo</b>   | Se optimiza                          | Se optimiza                      | Asignado a tareas                                       | Redistribuye contenido                                   |                    | Planifica                   | Aumenta la Eficiencia    |                            |   | Periodo de Aprendizaje  |
| <b>Metodología de Enseñanza</b>                                 | Uso apropiado                        | Mejores resultados               | - Facilita enseñanza<br>-Herramienta Técnicas y métodos | Mejor desarrollo   | Mejor organización | Actualiza                   | Complementa              |                            | Guía para servir de apoyo al estudiante | -Facilidad en Aprendizaje<br>-Aprendizaje sistemático<br>- Aprendizaje organizado             |
| <b>Recurso Pedagógico</b>                                       | Utilización para una mejor enseñanza | Utilización para mejor enseñanza | Facilita la Enseñanza                                   | Aplicación   | Utiliza            | Complemento                 | Actualiza                |                            | Ayuda a la estimulación de la enseñanza | Estimulación Motivación   |
| <b>Infraestructura</b>  | Almacena                             | Almacena Protege                 | Ambiente de trabajo apropiado                           | Medio para ser impartido                                 |                    | Medio para su desarrollo    | Medio para su desarrollo |                            | Creación de ambiente de seguridad       | -Ambiente apropiado<br>-Ofrece seguridad<br>Motivación  |
| <b>Grupo Familiar</b>   | Utiliza                              | Utiliza                          | Informa   | Aprende  |                    | -Apoyo -utiliza             | Utiliza                  | Ayuda a su mantenimiento   | Concientizar sobre el medio ambiente    | -Apoyo<br>-Motivación<br>-Estimulación  |
| <b>Estudiante</b>   | Utiliza                              | Utiliza                          | Comunicación  | Aprende temas  | Realiza buen uso   | Utiliza                     | Obtiene conocimiento     | Utiliza y da Mantenimiento | Comunica conocimiento                   | Apoyo mutuo   |

Cuadro 1.2: Matriz de Interrelación de las variables involucradas en la Educación Ambiental.



## 1.2.2. Formulación del problema

La figura 1.5 presenta el estado actual y deseado de la problemática de educación ambiental.



Figura 1.5: Diagrama de la caja negra.

## 1.2.3. Planteamiento del problema

### 1.2.3.1. Descripción del estado actual de la problemática:

- Poco impacto de los programas ambientales actuales
- Programas con poco alcance.
- Recurso Humano no aplica las metodologías de forma adecuada.
- Metodologías poco motivadoras.
- Deficiencia en conocimientos generales sobre el cuidado del medio ambiente.
- Estudiantes desinteresados por los problemas ambientales.

### 1.2.3.2. Descripción del estado deseado:

- Uso de herramientas tecnológicas para la concientización de la población salvadoreña.
- Aplicación de metodología innovadora.
- Mayor alcance de población estudiantil.
- Motivación del estudiante por aprender sobre el medio ambiente.
- Estudiantes con conocimientos necesarios para el correcto cuidado de los recursos naturales.
- Estudiantes interesados por los problemas ambientales.
- Difusión de la información en el núcleo familiar y comunidad.



El problema se sintetiza de la siguiente manera:

“Necesidad de implementar una técnica/herramienta de relativamente bajo costo, cuyo alcance sea superior al alcance del programa actual, de naturaleza innovadora y que capte la atención a través de un proceso interactivo, intuitivo y continuo para la educación ambiental de la población salvadoreña”

## 1.2.4. Análisis del problema

### 1.2.4.1. Variables de Entradas:

- Individuo No concientizado con respecto a actitudes y hábitos diarios que promuevan un medioambiente sostenible

### 1.2.4.2. Variables de Salida:

- Individuo concientizado con respecto a actitudes y hábitos diarios que promuevan un medioambiente sostenible

### 1.2.4.3. Variables de Solución:

- Variables de reestructuración del programa actual
- Técnicas y métodos innovadores para la generación de la Técnica/herramienta solución.
- Medios y formas de distribución masiva de dicha técnica.

### 1.2.4.4. Restricciones:

- Elementos que se implementen deben de estar de acuerdo a la realidad que vive la institución y el país (disponibilidad de recursos para aplicar nueva herramienta.)
- Enseñanza que implemente la herramienta debe de estar dentro de los requisitos dictados por leyes que rigen a la educación superior, universidad y facultad. (No utilizar material ofensivo para ciertos sectores de la población)

### 1.2.4.5. Criterios:

- Excelencia académica, calidad de herramienta solución, reutilización de herramienta a largo plazo.



#### **1.2.4.6. Análisis causa-efecto**

##### Instituciones Pro-Medio Ambiente

Estas se encargan de proteger al medio ambiente y de llevar una educación ambiental a la población. Las causas que no permiten alcanzar los objetivos propuestos por estas instituciones son las siguientes:

- **Existen pocas instituciones y la población las desconoce:** Mediante el estudio de campo realizado, se ha logrado determinar que un 50.5% población, no sabe que existen estas instituciones.
- **Presupuesto reducido:** Se asigna un presupuesto estatal reducido para la educación ambiental \$200,000 anuales.
- **Poco involucramiento del sector privado y público:** El Ministerio de Medio Ambiente y Recursos Naturales, tiene alianza con diferentes instituciones públicas y privadas, como alcaldías, centros escolares, etc., con las cuales hacen convenios, y transmiten los lineamientos que a través de los cuales, se inician actividades para proteger el medio ambiente, luego estas tienen que crear unidades ambientales para poder darle seguimiento a los proyectos, es decir son las instituciones de contrapartida del MARN las que se encargan de diseminar la información.

##### Planes, Programas y Proyectos Ambientales

Los programas actuales no llegan a toda la población y no alcanzan los objetivos trazados debido a:

- **Presupuesto asignado es mínimo:** El presupuesto asignado para el proyecto medio ambiental es escaso. Esto impide que las instituciones competentes puedan tener un alcance nacional, y llegar a todos los lugares de nuestro país.
- **Poco impacto en la población:** Los proyectos medio ambientales, tiene poco impacto en la población en general y especialmente en los estudiantes debido al poco interés de parte de la población para proteger al medio ambiente, pero esto se debe a que la población no se encuentra totalmente informada.

##### Recurso Tecnológico

La explotación de los recursos tecnológicos no es completa, puesto que se está perdiendo la oportunidad de transmitir información a la población sobre las problemáticas que en nuestro país y en todo el mundo existe, es decir, que se podría usar este recurso para





apoyar a la educación ambiental a través de software educativos, algunas de las causas para este problema son:

- **Falta de explotación de Recursos Tecnológicos como Herramientas para apoyar Métodos y Técnicas:** A cada aula informática se le ha asignado: veinte computadoras, una impresora, un televisor, una cámara digital, un proyector de cañón y otro de acetatos, con todos estos recursos tecnológicos los únicos software que se tiene actualmente en funcionamiento en las aulas informáticas son: la Enciclopedia Encarta, Micronet, Matemática y reto matemático, Tablas de multiplicar, Naturales y ejercicios de procesos, Geografía con Pipo los cuales sirven de apoyo a nivel investigativo, tal situación reafirma el hecho que se está desaprovechando una oportunidad de explotar los recursos para mejorar la metodología de enseñanza – aprendizaje, a través de la implementación de un software interactivo (video juego) que esté orientado a la educación ambiental.
- **Falta de cultura informática en el área medio ambiental:** La cultura informática en el área ambiental es nula, ya que en las aulas informáticas, como ya se dijo, la enseñanza se orienta a otras áreas de estudio, pero no al área de educación medioambiental.
- **Inexistencia de software interactivos-educativos (Video juegos Educativos) ambientalistas:** El tema del medio ambiente se puede ver en un eje transversal en todas las materias impartidas a los estudiantes de nuestro país, pero a pesar de eso, no existe un software que de una manera atractiva y divertida enseñe a los educandos en la temática de educación ambiental o que al menos les sirva de apoyo en la materia de Ciencias Salud y Medio Ambiente o en las temáticas del medio ambiente que se abordan en las demás materias.
- **Utilización nula de los recursos tecnológicos para educar a la población en la temática del medio ambiente:** Se tienen herramientas de software educativos que apoyan al alumno en sus tareas, pero ninguno está hecho para transmitir la educación ambiental y que esté acorde al plan curricular del MINED. El contenido de estas herramientas actualmente no satisface los objetivos de las materias impartidas, hay aplicativos que poseen que nunca llegan a ser utilizados y existen otros que no han sido tomados en cuenta e incluidos, por lo tanto no se posee una aplicación que de soporte total o parcial a los temas relacionados al medio ambiente. En base a estadísticas se ha podido establecer que los estudiantes asimilan el 22% con los métodos tradicionales de



enseñanza, y por medio de los video juegos logran asimilar el 94%, esto porque el usuario interactúa con el software, lo cual lo pasa de un estado pasivo a un activo durante el juego. Esto nos permite decir que si se utilizaran métodos como el de video juego, se tendría un mejor aprendizaje en los estudiantes.

### Metodología de Enseñanza y Técnica

La educación es la base para la formación de una cultura ambiental más respetuosa del patrimonio natural y facilita las posibilidades de lograr un desarrollo sostenible y una calidad de vida cada vez mejor, especialmente a lo relacionado a la vivienda, salud y educación. Por lo mismo, la temática ambiental debe de formar parte de la educación en todos sus niveles y modalidades. Las metodologías de enseñanza y técnicas utilizadas para transmitir la educación ambiental en la población en general y en especial en la población estudiantil son convencionales, por lo que las metodologías se convierten en un paradigma que afecta a la enseñanza-aprendizaje rápida y fácil por parte de la población y de los educandos, siendo la metodología una de los criterios para transmitir el conocimiento, la información, se vuelve un obstáculo en el proceso de educación ambiental. Sus causas principales son las siguientes:

- **Deficiencia en el enfoque pedagógico:** En la población estudiantil, el enfoque utilizado por los maestros no es del todo desarrollado para que los alumnos comprendan los contenidos, sino para que se memoricen los mismos, esto implica que después de un tiempo lo aprendido es olvidado por que no se profundizo en la comprensión, esto radica en el mal enfoque pedagógico al impartir las clases<sup>33</sup>. Por otra parte, el MINED recientemente está trabajando en los planes curriculares, para integrar temas que aborden la problemática ambiental y estos serán implementados hasta el próximo año. Por el momento la enseñanza se transmite por medio de métodos tradicionales.
- **Requiere un nivel educativo:** Esto se refiere a que en el programa de estudio, del MINED, se asignan determinados temas de acuerdo al grado, esto limita el conocimiento en los primeros ciclos. Se debería transmitir la educación ambiental desde los primeros años de estudio, pues mediante estudios psicológicos, en la edad temprana, se forman los hábitos en las personas, esto ayudará a formar desde pequeños a los niños y no esperar hasta que estos sean adolescentes, ya con hábitos impregnados en su personalidad. Esto debe hacerse con metodologías apropiadas, que entretengan a los pequeños, que les

---

<sup>33</sup> Datos obtenidos por observación directa en un centro educativo



diviertan a parte de educarles en el tema del medio ambiente, como un video juego que les eduque en el área ambiental.

- **Materiales poco llamativos e innovadores:** En la población estudiantil, la metodología generalmente utilizada consiste en impartir la clase oral o escrita en el pizarrón sin más atributos atractivos que despierten el interés de los educandos y faciliten el proceso de enseñanza aprendizaje. Se poseen recursos en las aulas informáticas que actualmente no son usados y explotados al máximo como herramientas o software adecuados al proceso de enseñanza - aprendizaje<sup>34</sup> de la educación ambiental. Por otra parte, en la demás población, el MARN es el que se encarga en informar sobre esta temática y lo hace mediante los servicios prestados a la población: General especializada, Hemeroteca, Consulta, Tesis, Audiovisuales, Mapoteca.
- **No se logra captar la atención de la población en general y especialmente de los jóvenes:** Para la población estudiantil, la metodología de enseñanza-aprendizaje, se basa en los métodos y técnicas tradicionales sin innovar en nuevos mecanismos de enseñanza que llamen la atención de los educandos y que de una manera permita que los contenidos se comprendan con mayor facilidad al atraer el interés de los mismos. En cuanto al nivel de captación de la población en general se puede decir que no se logran los objetivos de manera permanente, por hacerlo mediante recursos inadecuados, poco llamativos y aburridos, pues en su gran mayoría, se hace por medio de boletines informativos, folletería, enciclopedias, etc., esto parece aburrido y tedioso para la población en general. Pues en nuestra cultura, no se cultiva suficientemente la lectura.
- **No toda la población tiene acceso a la información:** Esto se refiere a que por problemas de canalización de la información, gran parte de la población, desconoce los servicios que el MARN ofrece en el Centro de Información con el que cuenta, pues como ya se ha recalcado, la falta de recursos humanos y económicos, no posibilita llevar la información a toda la población a nivel nacional. Es decir, el alcance de los proyectos de información ambiental es reducido.

---

<sup>34</sup> Para mejor aclaración referirse a la espina de los recursos tecnológicos



### Recurso Humano

Actualmente, no se cuenta con suficiente recurso humano que esté totalmente capacitado en las áreas de informática y medio ambiente, es decir, no se incrementa el conocimiento de los maestros en el uso de nuevas herramientas didácticas que apoyen a la enseñanza.

- **Falta de la cultura informática en profesores:** Los maestros no están acostumbrados al manejo del equipo computacional por eso se les hace muy difícil operar un computador y otros recursos tecnológicos (cámara digital, cañón, impresoras, etc.). Actualmente se está desarrollando una cultura informática lo que trae como consecuencia la no utilización total de la tecnología de las aulas informáticas como un medio didáctico.
- **Falta de capacitación en los profesores:** Las capacitaciones a los maestros sobre el uso del computador y demás equipo existente, no son continuas, por lo que los profesores no se actualizan con los avances informáticos que se dan a diario, es decir, al personal no se capacita de manera periódica y planeada para apoyarse en nuevas metodologías de aprendizaje atractivas al estudiante. Las capacitaciones se dan con la idea que estos utilicen los recursos que se les han proporcionado. Anteriormente solo se les daba asistencias y capacitaciones pedagógicas tradicionales. Recientemente los profesores han sido capacitados en el área medio ambiental, para que éstos puedan transmitir el conocimiento a los estudiantes a través de los nuevos programas de estudio, que serán implementados en su totalidad, a partir del siguiente año.
- **Inexistencia de personal destinado al desarrollo de software educativo a la medida:** Ni el MINED, ni el MARN, cuentan con un área destinada y especializada en el desarrollo de Software Educativos; el MINED solo tiene una unidad que desarrolla software de gestión administrativa. No ha nacido la iniciativa de crear sus propias herramientas de software a la medida que incluyan los contenidos programáticos actualizados.
- **Poco recurso humano para la demanda estudiantil:** La demanda estudiantil sobrepasa la capacidad de la planta docente, esto se puede observar en las aulas de las escuelas públicas donde se encuentran alrededor de 40 a 55 alumnos para un solo profesor, el cual no alcanza a atender al alumnado de manera personalizada.



- **Resistencia al cambio en los maestros:** Los maestros se restringen al uso limitado de la computadora, estos no se dedican a explorar el equipo como herramienta de apoyo a sus clases<sup>35</sup>.

### Información

El Ministerio de Medio Ambiente y Recursos Naturales, transmite la información por medio de documentos, libros especializados en Medio Ambiente y Recursos Naturales, publicaciones, revistas, folletos, boletines, videos, CD'S, cassettes, rotafolios, mapas, y etc. A pesar de esto, la información no llega a toda la población, las causas se describen a continuación:

- **Poca difusión por los medio de comunicación:** La información relacionada a la temática del medio ambiente, plasmada en los planes, programas y proyectos ambientales, en la mayoría de los casos no llega para quedarse en el subconsciente de la población, debido a la poca promoción y publicidad que a esta temática se le da a través de los medios de comunicación. Esto da como resultado un porcentaje alarmante de la ciudadanía en general que desconoce que existen instituciones que protegen al medio ambiente, personas desinteresadas en el cuidado del medio ambiente.
- **No llega a toda la población salvadoreña:** El alcance actual de promoción de folletería y capacitaciones medioambientalistas solo se reduce a unas cuantas personas a la vez debido al poco presupuesto, la institución carece de los recursos económicos para llevar a cabo campañas masivas ya sea por medio de la utilización de medios impresos, radio o televisión. Este limitante hace que la mayoría de la población carezca de los conocimientos fundamentales para el cuidado del medioambiente.
- **Uso de medios convencionales:** La folletería y capacitación de persona a persona, como se ha mencionado antes, solo tiene un alcance limitado, y muchas personas no se encuentran interesadas en leer folletos o tratados del cuidado medioambiental, simplemente no llama la atención. En el caso de capacitaciones personales de maestros y voluntarios, va a depender del individuo capacitado, en que tanto va a transmitir su conocimiento a las demás personas y que tan preciso ese conocimiento será, es decir, ¿se transmitirá

---

<sup>35</sup> Información proporcionada por el personal de tecnología, observaciones a la hora de capacitar en los centros educativos



completamente la información? o debido a factores humanos, ¿se olvidará mencionar puntos críticos a las personas (alumnos, niños o voluntarios).

### Estudiantes

Los alumnos resultan ser el problema principal cuando estos no tienen una atención personalizada y el apoyo necesario, desmotivándolos a la hora del aprendizaje, las causas que influyen en este problema se exponen a continuación:

- **Falta de educación personalizada:** Debido a que existen demasiados alumnos en una sola aula los maestros no pueden brindar una atención personalizada a estos y que no se pueden cubrir todas las dudas que estos puedan tener siendo algunas de estas importantes e imprescindibles sin percatarse de ello, esta situación lleva al retraso de más de un alumno en el proceso enseñanza-aprendizaje.
- **Falta de herramientas de apoyo para el aprendizaje:** La falta de una herramienta educativa (software adicional a la enciclopedia Encarta) ocasiona que se siga con el mismo mecanismo y no se implementen nuevas herramientas como un software que ayude a mejorar los promedios de los alumnos y minimice la reprobación de los estudiantes.
- **Falta de motivación en aprendizaje:** Los estudiantes se ven afectados por diversos factores como su medio ambiente, relación familiar; etc., los cuales pueden desmotivarlos durante el proceso de enseñanza. En general, la misma metodología les causa aburrimiento.

### Grupo Familiar

El no involucramiento del grupo familiar en forma positiva en el proceso enseñanza-aprendizaje se vuelve un problema que afecta en la educación del alumno, por esta falta de atención el educando reduce el rendimiento académico. Las causas que llevan a este problema son las siguientes:

- **Falta de apoyo y desinterés en el aprendizaje del niño:** el alumno no recibe apoyo por los miembros de su familia a la hora de realizar sus tareas en el hogar, y es cuando este necesita ayuda adicional por qué no se encuentra con el profesor y le podrían surgir ciertas dudas que necesiten ser resueltas de inmediato para que el proceso de aprendizaje no se detenga.



- **Malas influencias:** el medio en el que se desarrolla el grupo familiar puede traer malas consecuencias para el estudiante, si los miembros de la familia se inclinan por vicios o conductas extremas que causan estrés en el educando, esto hará que incidan en conductas psicológicas de desinterés y ansiedad en el mismo.
- **Falta de comunicación con maestros:** si el padre no está en constante comunicación con el maestro y no existe una coordinación para identificar los avances y las deficiencias en el aprendizaje del alumno hace que este sea más lento, la comunicación constante refuerza en el grupo familiar los contenidos con dificultad.
- **Falta de herramientas en el hogar para apoyar el aprendizaje:** la no existencia de equipo computacional u otros medios didácticos para apoyar al alumno en el hogar conlleva al círculo vicioso de la enseñanza tradicional que no lleva al aprendizaje rápido del alumno.

Dado los principales factores que intervienen en este sistema de educación, se procede a profundizar sobre el tema para afinar y determinar más a fondo la naturaleza de la problemática por medio del diagrama causa y efecto presentado en la figura 1.6.

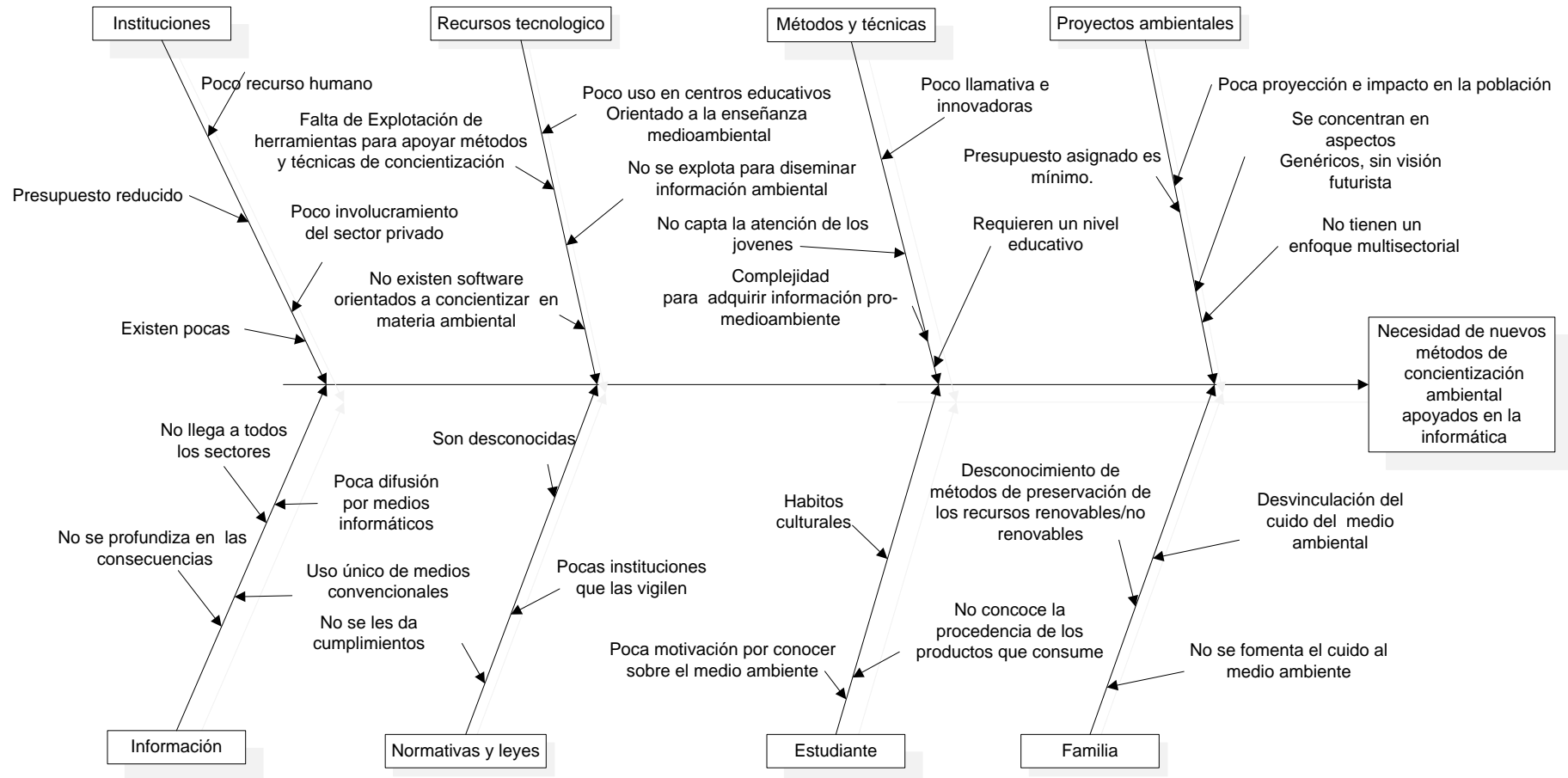


Figura1.6: Diagrama causa-efecto





### 1.2.5. Búsqueda de soluciones

En base al análisis de la problemática ambiental que afronta El Salvador, entrevistas con los técnicos del MARN y Gerente de la Unidad de Educación Ambiental se han propuesto una serie de alternativas que podrían solventar las diferentes problemáticas planteadas en la figura 1.6, estas soluciones han sido formuladas por el grupo de desarrollo por medio de la técnica de lluvia de ideas:

- **Integración de nuevos contenidos ambientales al programa de Ciencias Salud y Medio Ambiente:** Esta solución consiste en incrementar el número de horas y temáticas que se incluye en los programas de educación para los diferentes niveles de educación. Incluye el proceso de reforma de contenidos, impresión de material didáctico, distribución de material y capacitación de docentes.
- **Integrar la televisión educativa en los centros escolares:** Se le presentan constantemente videos ambientalistas, por medio de tareas se puede llevar un control de lo aprendido por el estudiante. Es necesario que se elabore el material educativo y se capacite a los docentes sobre las formas de retroalimentar y aplicar esta metodología.
- **Contratación y capacitación de personal adicional para cada departamento del país:** Esta alternativa de solución implica el proceso de reclutamiento, selección, inducción y capacitación de nuevo personal para cada zona departamental del país.
- **Desarrollo de un software interactivo para concientizar a la población estudiantil sobre el cuidado del medio ambiente:** Esta alternativa propone la creación de un software que llegue a los niños por medio de las aulas informáticas de MINED, donde se les enseñe de una forma divertida e innovadora sobre las siguientes temáticas: Reforestación, especies Salvadoreñas en extinción, consumo responsable de productos dañino al medio ambiente y preservación de mantos acuíferos.
- **Mayor capacitación al personal existente:** Esta alternativa consiste en identificar las fortalezas, amenazas, debilidades y oportunidades del personal, en base a este análisis elaborar y llevar a cabo planes de capacitación adecuados para solventar deficiencias en el personal.



### 1.2.6. Selección de alternativa de solución

La asignación de los valores de ponderación para cada criterio mostrados en el siguiente cuadro, fue el resultante de las directrices proporcionadas por el Gerente de Educación Ambiental del MARN.

| Código       | Criterios           | Ponderación |
|--------------|---------------------|-------------|
| C1           | Costo               | 40%         |
| C2           | Alcance             | 40%         |
| C3           | Temáticas abordadas | 20%         |
| <b>Total</b> |                     | <b>100%</b> |

| Código       | Mínimos vitales   | Ponderación |
|--------------|---|-------------|
| M1           | Que profundice en los métodos de preservación del medio ambiente. | 30%         |
| M2           | Que capte la atención de los jóvenes                              | 40%         |
| M3           | Poca proyección e impacto en la población                         | 30%         |
| <b>Total</b> |   | <b>100%</b> |

Cuadro 1.3: Ponderación de Criterios de Solución y Mínimos Vitales

Las soluciones propuestas a la Unidad de Educación Ambiental, se presentan en el cuadro 1.4, la metodología para la selección la alternativa de solución fue la siguiente:

- Se coordinó una reunión con el Gerente de Educación Ambiental, con el propósito de evaluar cada una de las alternativas.
- Si se tratase de un criterio de solución (C1 por ejemplo), la nota más baja (cero) indicaría que la alternativa de solución propuesta no satisface dicho criterio; mientras que la nota más alta (% asignado a cada criterio) indicaría que la alternativa de solución satisface por completo al criterio.
- En caso de ser un mínimo vital (M1 por ejemplo), la nota más baja (cero) indicaría que la alternativa de solución propuesta no ataca en lo absoluto dicho mínimo vital; mientras que la nota más alta (% asignado a cada mínimo) indicaría que la alternativa de solución ataca por completo al mínimo vital. Los resultados se muestran en el cuadro 1.4.
- La aceptación de la alternativa aceptabilidad se dará si esta posee como mínimo un 80% de cumplimiento de criterios y un 80% de cumplimiento de mínimos vitales, es decir que aborde las causas principales del problema previamente diagnosticado por medio del análisis causa-efecto.



| Posible solución   | C1 | C2 | C3 | CC | M1 | M2 | M3 | CM |
|--|----|----|----|----|----|----|----|----|
| Integración de nuevos contenidos ambientales al programa de Ciencias Salud y Medio Ambiente:                           | 10 | 35 | 19 | 64 | 15 | 10 | 20 | 45 |
| Integrar la televisión educativa en los centros escolares  | 10 | 30 | 20 | 60 | 10 | 25 | 30 | 65 |
| Contratación y capacitación de personal adicional para cada departamento del país                                      | 8  | 10 | 5  | 23 | 5  | 12 | 15 | 32 |
| Desarrollo de un software interactivo para concientizar a la población estudiantil sobre el cuidado del medio ambiente | 20 | 40 | 20 | 80 | 20 | 40 | 30 | 90 |
| Mayor capacitación al personal existente   | 20 | 12 | 5  | 37 | 15 | 12 | 10 | 37 |

Cuadro 1.4: Porcentaje de cumplimiento de los mínimos vitales para las soluciones propuestas

Mediante esta evaluación de alternativas se ha llegado a determinar que el desarrollo de un software interactivo es la que más se ajusta a los recursos actuales del Programa Nacional para el Reciclaje y con esta se aborda las principales causas del poco interés de la población por el cuidado del medio ambiente. El criterio que se evaluó mejor para esta alternativa es su alcance, ya que por medio de este tipo de software se puede llegar a más población estudiantil a un menor costo y principalmente esta metodología tiene como fuerte atraer a todo tipo de personas.

Por tanto el “Desarrollo de un software interactivo para concientizar a la población estudiantil sobre el cuidado del medio ambiente”, cubre en un 80% los criterios de selección y en un 90% se abordan las tres principales problemáticas que afronta el programa actual.

### 1.3. FACTIBILIDADES DEL PROYECTO

El estudio de factibilidad respalda el desarrollo del Software Interactivo para la Educación Medioambiental de la Población Salvadoreña, los aspectos tomados en cuenta se dividen en cuatro áreas: técnica, económica, operativa y ambiental.



### 1.3.1. Factibilidad técnica

Se determina si se cuenta con los recursos técnicos y tecnológicos para efectuar las actividades o procesos que requiere el proyecto, analizando tres aspectos: software, hardware y recurso humano.

#### 1.3.1.1. Software

Es indispensable para el desarrollo del videojuego contar con las categorías de software siguientes: sistema operativo, lenguaje de programación, APIs gráfica, programa de modelado 3D, editor de imágenes, lenguaje para efectos gráficos y utilitarios.

La selección del software de desarrollo se realizó por medio de una evolución técnica aplicando la siguiente metodología<sup>36</sup>:

- Se definen macro-criterios
- Para cada macro-criterio se definen sub-criterios según se considere necesario.
- La importancia de cada macro-criterio y sub-criterios se establece en forma porcentual.
- El resultado final debe ser una estructura jerárquica de criterios y porcentajes en base a un proceso top-down.
- Posteriormente se realiza la calificación de cada sub-criterio en el rango de 0-10.
- Se totaliza macro-criterio.

#### Sistema Operativo

Para seleccionar el sistema operativo se realizó una evaluación técnica considerando características referentes a: *Atributos de calidad del software, compatibilidad con el hardware y soporte técnico.*

1- **Calidad del Sistema Operativo**<sup>37</sup>: Grado en que el sistema operativo cumple con los atributos de calidad siguientes:

- **Eficiencia:** Nivel en que se requiere recursos de ordenador para realizar las funciones con la precisión necesaria, en este criterio se evalúan: tiempo de respuestas y utilización de recursos (memoria, procesador).

<sup>36</sup> García Carlos Ernesto, Gerencia Informática, Informatik, Pág. 79.

<sup>37</sup> Conceptos y atributos en base a la Norma ISO-9126: Calidad en la Industria del Software.



- **Portabilidad:** Se necesita que el software posea alta facilidad de instalación, adaptabilidad y coexistencia (capacidad del software de funcionar en paralelo con otras aplicaciones).
  - **Adaptabilidad:** La capacidad del sistema para adaptarse a medio ambientes diferentes a aquellos para los cuales fue diseñado. El sistema operativo debe poder adaptarse a esta evolución de hardware.
  - **Robustez:** Capacidad del Sistema para continuar operando en circunstancias extremas (captura de datos inválidos, excesiva carga de trabajo).
- 2- **Compatibilidad de Hardware:** El sistema operativo debe poseer controladores de variedad de dispositivos de hardware, en este aspecto se evalúan los sub-criterios:
- **Compatibilidad con dispositivos:** Capacidad de funcionar correctamente con dispositivos (tarjetas de video) de diversos fabricantes.
  - **Actualizaciones de controladores:** Si permite actualización de controladores, así como la disponibilidad de desarrollo de controladores de los fabricantes para este sistema operativo.
- 3- **Soporte Técnico:** El sistema debe contar con mecanismos (páginas web, foros de discusión) y soporte del fabricante en caso de surgir problemas en su operación cotidiana, en este criterio se miden las sub-características:
- **Ayuda:** El fabricante del sistema debe proveer ayuda ya sea telefónica o en línea para solventar problemas.
  - **Documentación:** Mide la disponibilidad de páginas web, foros que existan para solventar dudas o problemas sobre el sistema operativo.
- 4- **Costo y Disponibilidad:** Toma en cuenta la factibilidad de costeo del sistema operativo, por medio de los siguientes sub-criterios.
- **Disponibilidad en el mercado:** Que sea de fácil obtención en el mercado de software en nuestro país.
  - **Factibilidad de costo:** Evalúa la disponibilidad de adquisición en base a la cantidad monetaria a pagar en concepto de licencia de uso.



El cuadro 1.5 presenta los sistemas operativos considerados para la evaluación por ser los más usados a nivel mundial de acuerdo a la Figura 1.7 publicada por la Global Market Share Statistics<sup>38</sup>.

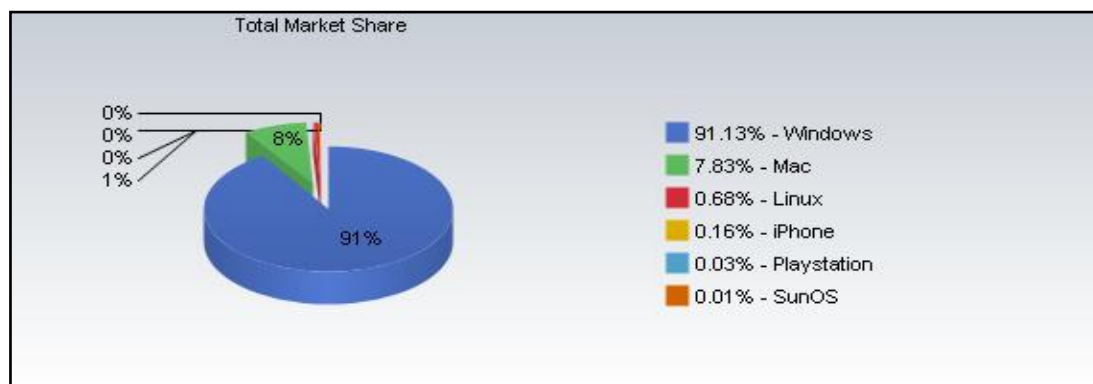


Figura 1.7: Sistemas operativos usados a nivel mundial

| Macro criterio /Criterio          | Ponderación | Calificación <sup>39</sup> |               |            |            |
|-----------------------------------|-------------|----------------------------|---------------|------------|------------|
|                                   |             | Windows XP                 | Windows Vista | Linux      | Mac OS     |
| <b>Calidad del software</b>       | <b>35%</b>  | <b>7.9</b>                 | <b>6.8</b>    | <b>8.7</b> | <b>8.4</b> |
| Eficiencia                        | 10%         | 8                          | 7             | 9          | 9          |
| Portabilidad                      | 5%          | 9                          | 6             | 7          | 7          |
| Adaptabilidad                     | 10%         | 8                          | 7             | 8          | 8          |
| Robustez                          | 10%         | 7                          | 7             | 10         | 8          |
| <b>Compatibilidad de hardware</b> | <b>35%</b>  | <b>8.6</b>                 | <b>8</b>      | <b>6.4</b> | <b>5.3</b> |
| Compatibilidad con dispositivos   | 20%         | 9                          | 8             | 6          | 4          |
| Actualizaciones de controladores  | 15%         | 8                          | 8             | 7          | 7          |
| <b>Soporte técnico</b>            | <b>15%</b>  | <b>9</b>                   | <b>8.7</b>    | <b>7</b>   | <b>7.6</b> |
| Ayuda (línea, telefónica)         | 10%         | 9                          | 9             | 7          | 9          |
| Documentación(web, foros)         | 5%          | 9                          | 8             | 7          | 5          |
| <b>Costo y disponibilidad</b>     | <b>15%</b>  | <b>8.2</b>                 | <b>4.8</b>    | <b>9.3</b> | <b>7.7</b> |
| Disponibilidad en el mercado      | 2%          | 10                         | 10            | 5          | 6          |
| Factibilidad de costo             | 13%         | 8                          | 4             | 10         | 8          |
| <b>Total</b>                      | <b>100%</b> | <b>8.3</b>                 | <b>7.2</b>    | <b>7.7</b> | <b>7.1</b> |

Cuadro 1.5: Evaluación de Sistemas Operativos.

<sup>38</sup> <http://marketshare.hitslink.com/report.aspx?qprid=8>

<sup>39</sup> En base a Requerimientos de hardware de los Sistemas Operativos, experiencia y estudios como: Linux as a Trusted Operating System: A Secure and Cost-Effective Choice for Government Systems [http://t1d.www-03.cacheibm.com/industries/government/doc/content/bin/GI202466\\_TrustedLinux.pdf](http://t1d.www-03.cacheibm.com/industries/government/doc/content/bin/GI202466_TrustedLinux.pdf), Comparación de los sistemas operativos [http://t1d.www-03.cacheibm.com/industries/government/doc/content/bin/GI202466\\_TrustedLinux.pdf](http://t1d.www-03.cacheibm.com/industries/government/doc/content/bin/GI202466_TrustedLinux.pdf), entre otros.



La puntuación se multiplica por el porcentaje asignado a cada sub-criterio y la sumatoria de estas se refleja en la puntuación total. En esta evaluación se evidencia que Windows y Linux son los Sistema Operativos económicamente con mejor puntaje ya que Linux es de distribución libre y la licencia de Windows ya ha sido previamente adquirida.

Conclusiones evaluación sistema operativo: En base a los criterios técnicos y económicos considerados para la evaluación del sistema operativo, se estableció la valoración cuantitativa, resultando con mayor puntuación el sistema operativo Windows XP SP2. Por lo anteriormente expuesto consideramos que este es el que mejor se adecua a las necesidades del proyecto.

### Lenguaje de desarrollo

Realizar la mejor selección del lenguaje de desarrollo permitirá desarrollar una aplicación en el menor tiempo y con la calidad deseada. Para ello se tomarán los siguientes criterios:

1. **Entorno integrado de desarrollo y compatibilidad:** Evalúa la existencia de entornos integrados de desarrollo para el respectivo lenguaje de programación que cumpla con los siguientes sub-criterios:
  - **Auto complementación:** La facilidad de poder automáticamente complementar la sintaxis del lenguaje.
  - **Depuración en tiempo de ejecución:** Evalúa la inclusión de esta técnica que mejorar el rendimiento de los sistemas de programación, por medio de un lenguaje intermedio portable y optimizado.
  - **Extensiones de lenguaje:** La disponibilidad de configuraciones de otras herramientas, que permitan extender las utilidades del entorno integrado de desarrollo, contribuye con la automatización del desarrollador.
  - **Integración de APIs gráfica:** Que permita el uso de librerías para el manejo de la tarjeta gráfica.
  
2. **Orientación a objetos:** Mide la capacidad que tiene el lenguaje de programación de aplicar los siguientes conceptos:
  - **Herencia:** Debe permitir la creación de clases que se relacionan entre sí, formando una jerarquía de clasificación. Los objetos deben heredar las propiedades y el comportamiento de todas las clases a las que pertenecen.



- **Polimorfismo:** Debe permitir comportamientos diferentes, asociados a objetos distintos, pueden compartir el mismo nombre, al llamarlos por ese nombre se utilizará el comportamiento correspondiente al objeto que se esté usando.
  - **Abstracción:** Evalúa la capacidad del lenguaje de programación de facilitar que cada objeto sirva como modelo de un "agente" abstracto que puede realizar trabajo, informar y cambiar su estado, y "comunicarse" con otros objetos en el sistema sin revelar cómo se implementan estas características.
  - **Encapsulamiento:** Permite proteger y englobar a los datos y la funcionalidad de dichos datos en una estructura que permite tenerlos asociados.
3. **Facilidad de aplicación:** Mide el tiempo invertido en el aprendizaje del lenguaje de programación así como las líneas de código por función a desarrollar.
- **Curva de aprendizaje del lenguaje:** Facilidad de aprender la semántica y sintaxis del lenguaje de programación.
  - **Tiempo de desarrollo de aplicaciones:** Evalúa la cantidad de líneas de código necesarias para realizar una función.
4. **Costo:** Toma en cuenta los precios de los entornos de desarrollo para estos lenguajes o la existencia de versiones libres, se evalúan principalmente los siguientes atributos. Disponibilidad en el mercado y factibilidad de pago.

Este criterio no se evaluará en el cuadro 1.6 debido a que todos los lenguajes de desarrollo evaluados, cuentan con un IDE gratuito. Los lenguajes de desarrollo que se evaluarán serán C++, C#, Visual Basic y Java como se presenta, ya que son orientados a objetos siendo los más usados en la actualidad por desarrolladores profesionales de videojuegos multiplataforma.<sup>40</sup>

---

<sup>40</sup> <http://www.enterate.unam.mx/Articulos/2004/agosto/vidjava.htm>





| Criterio/Sub-Criterio <sup>41</sup>                | %           | CALIFICACION <sup>42</sup> |            |            |            |
|--|-------------|----------------------------|------------|------------|------------|
|  |             | C++                        | C#         | VB .Net    | Java       |
| <b>IDE</b>   | <b>35%</b>  | <b>8</b>                   | <b>8.3</b> | <b>7.6</b> | <b>7</b>   |
| Auto complementación                               | 15%         | 10                         | 10         | 10         | 9          |
| Depuración en ejecución.                           | 5%          | 10                         | 10         | 10         | 10         |
| Extensiones de lenguaje                            | 10%         | 8                          | 9          | 8          | 8          |
| Integración con Apis Gráficas                      | 5%          | 10                         | 10         | 7          | 6          |
| <b>Orientación a Objetos</b>                       | <b>35%</b>  | <b>10</b>                  | <b>9.7</b> | <b>9.7</b> | <b>10</b>  |
| Herencia   | 10%         | 10                         | 9          | 9          | 10         |
| Polimorfismo                                       | 10%         | 10                         | 10         | 10         | 10         |
| Abstracción  | 5%          | 10                         | 10         | 10         | 10         |
| Encapsulamiento                                    | 10%         | 10                         | 10         | 10         | 10         |
| <b>Facilidad de aplicación</b>                     | <b>30%</b>  | <b>7</b>                   | <b>9</b>   | <b>9</b>   | <b>8</b>   |
| Curva de aprendizaje del lenguaje                  | 15%         | 7                          | 9          | 9          | 8          |
| Tiempo de desarrollo de aplicaciones <sup>43</sup> | 15%         | 7                          | 9          | 9          | 8          |
| <b>Total</b>                                       | <b>100%</b> | <b>8.4</b>                 | <b>9</b>   | <b>8.7</b> | <b>8.3</b> |

Cuadro 1.6: Cuadro comparativo de calificación de lenguajes de programación.

### Conclusiones evaluación lenguaje de desarrollo:

El lenguaje de desarrollo seleccionado es C#, por haber obtenido una puntuación de 9 al considerar los aspectos técnicos principales; el entorno integrado de desarrollo se utilizará Visual Studio en su versión Express Edition, tomando en cuenta las características que lo hacen el más idóneo para el desarrollo de la solución propuesta.

### Modelado 3D.

Los criterios que se tomarán en cuenta para evaluar el software de modelado 3D son:

- **Creación de primitivas y Animación esquelética:** Creación de formas básicas (esferas, cubos, toroides, etc.), además debe permitir la creación de huesos que permiten la realización de animaciones de los modelos 3D.
- **Herramientas de optimización:** Evalúa la integración de herramientas que permitan reducir la utilización de recursos de los modelos manteniendo el realismo de estos.

<sup>41</sup> En base a Documentación desarrollador 5 estrella, Estrella 1, Introducción Microsoft .Net, Diapositiva 68 en adelante

<sup>42</sup> Ibid

<sup>43</sup> En base a líneas de código por función evaluadas con el software Resource Standard Metrics C ,C++ C# Java 7.50



- **Interfaz de usuario integrada:** Poder disponer de un conjunto de herramientas que permitan hacer un trabajo más intuitivo, contribuye a la mejora del trabajo.
- **Formatos de modelos:** Capacidad de exportar e importar a diferentes formatos compatibles con otras aplicaciones.
- **Costo:** Evalúa la factibilidad de costeo de las aplicaciones evaluadas.

En el cuadro siguiente se presenta la evaluación de criterios para elegir el software de modelado 3D.

| Criterio                     | Ponderación | CALIFICACION <sup>44</sup> |            |               |            |
|------------------------------|-------------|----------------------------|------------|---------------|------------|
|                              |             | fragMotion                 | Milkshape  | 3D Max Studio | Maya 8.0   |
| Animación                    | 25%         | 10                         | 7          | 10            | 10         |
| Herramientas de Optimización | 15%         | 5                          | 10         | 7             | 9          |
| Interfaz Integrada           | 10%         | 10                         | 10         | 10            | 10         |
| Formato de Modelos           | 25%         | 9                          | 9          | 8             | 10         |
| Costo                        | 25%         | 7                          | 7          | 5             | 4          |
| <b>Total</b>                 | <b>100%</b> | <b>8.3</b>                 | <b>8.3</b> | <b>7.8</b>    | <b>8.3</b> |

Cuadro 1.7: Cuadro comparativo de calificación de programa de modelado 3D

#### Conclusiones evaluación modelado 3D:

De la puntuación obtenida de estas aplicaciones se deduce que Milkshape, fragMOTION integran mejor las funcionalidades necesarias para el modelado 3D por lo que es factible usar ambos.

#### Editor de imágenes.

Es necesario contar con un editor de grafico que cumpla con siguientes criterios:

- **Optimización de imágenes:** Equilibrio entre las imágenes atractivas y tamaño de imagen.
- **Edición y exportación a diferentes formatos de imágenes:** Es necesario contar con una herramienta que permita crear y editar imágenes de mapas de bit, targa<sup>45</sup>, raw<sup>46</sup>, png, entre otros formatos necesarios para la creación de terrenos y efectos de texturizado como transparencias, sombras, etc.

<sup>44</sup> La experiencia de modelado e información de característica provistas en la ayuda de cada aplicación.

<sup>45</sup> Permite guardar imágenes monocromáticas (2 bits) y con diferentes niveles de profundidad de color (8, 16, 24 y 32 bits), utilizando o no una paleta gráfica.

<sup>46</sup> Es un formato de archivo digital de imágenes que contiene la totalidad de los datos.



- **Incorporación de efectos vectoriales:** El realismo y realce de las ilustraciones, se logran a través de efectos vectoriales como los degradados de contorno, trazos discontinuos, esto a través de combinación de colores.
- **Herramientas de dibujo:** La disponibilidad de variedad de herramientas de dibujo, transformación, filtros, efectos, etc.
- **Interfaz de usuario amigable:** Poder disponer de un conjunto de fácil accesibilidad a herramientas, que permitan hacer un trabajo más intuitivo y agilizar la creación de texturas.

Se evalúan editores de imágenes considerados profesionales y que permiten formato targa, png y Raw (usados en la creación de terrenos), estos se evalúan en el cuadro 1.8

| Criterio               | Ponderación | CALIFICACION <sup>47</sup> |               |              |             |
|------------------------|-------------|----------------------------|---------------|--------------|-------------|
|                        |             | Gimp                       | Photoshop CS2 | Corel Draw 8 | FreeHand MX |
| Optimización           | 20%         | 7                          | 9             | 8            | 8           |
| Formatos de imágenes   | 15%         | 5                          | 9             | 9            | 7           |
| Efectos                | 20%         | 8                          | 9             | 9            | 8           |
| Herramientas de dibujo | 20%         | 7                          | 10            | 8            | 9           |
| Interfaz integrada     | 10 %        | 5                          | 10            | 9            | 9           |
| Costo                  | 15%         | 10                         | 5             | 4            | 5           |
| <b>Total</b>           | <b>100%</b> | <b>7.1</b>                 | <b>8.7</b>    | <b>7.9</b>   | <b>7.7</b>  |

Cuadro1.8: Evaluación editor de imágenes

Conclusiones evaluación herramienta de edición de imagen:

La aplicación a utilizar es Photoshop CS2 debido a que tiene más herramientas de dibujo, efectos vectoriales y una interfaz más integrada.

API Gráfica.

Una API (Application Programming Interface) o Interfaz de Programación de Aplicaciones, es un conjunto de funciones que realizan tareas específicas facilitando la vida al programador. Al hablar de API gráfica se refiere a un set de funciones para inicializar por ejemplo modos gráficos, realizar copiado de datos de la memoria del computador a la tarjeta de video, etc. Los criterios de selección para esta API se establecen de acuerdo a la funcionalidad que permiten:

<sup>47</sup> Experiencia en técnica de texturizado de integrantes del equipo de trabajo y artículos de diseñadores 2D y 3D profesionales como: <http://www.leighvanderbyl.com/pdf/texturing.pdf>



- Manejo de Gamepad, mouse y teclado: Permite la comunicación con periféricos.
- Manejo de modelos 3D: Incorpora instrucciones para manejar modelos 3D
- Cargar Imágenes: Carga de imágenes en diferentes formatos (png, jpg, etc.)
- Cargar Sonido: Carga de formatos de sonido como wav, mp3, ogg, etc.
- Uso de Fuentes: Permite el uso de fuentes TrueType<sup>48</sup>.

En el cuadro 1.9 se evalúa las siguientes APIs Gráficas OpenGL y DirectX que son las actualmente soportadas por la mayoría de tarjetas gráficas<sup>49</sup>.

| Criterio                    | Ponderación | CALIFICACION <sup>50</sup> |            |
|-----------------------------|-------------|----------------------------|------------|
|                             |             | DirectX                    | OpenGL     |
| Manejo de Gamepad y teclado | 20%         | 8                          | 6          |
| Manejo de gráficos 3D       | 20%         | 10                         | 10         |
| Manejo de gráficos 2D       | 20%         | 10                         | 10         |
| Cargar sonido               | 20%         | 9                          | 5          |
| Uso de Fuentes              | 20%         | 9                          | 8          |
| <b>Total</b>                | <b>100%</b> | <b>9.2</b>                 | <b>7.8</b> |

Cuadro 1.9: Evaluación API Gráfica

### Lenguaje para efectos gráficos

En función de la APIs Gráfica seleccionada se usara el siguiente lenguaje para la creación de efectos: Se usara el lenguaje HLSL<sup>51</sup> de alto nivel que fue desarrollado por Microsoft para permitir a los programadores que usan su API DirectX desarrollar efectos gráficos complejos.

### Librería XNA 2.0

Es un conjunto de utilidades y *frameworks* que permiten desarrollar fácilmente juegos en C# para las plataformas Windows XP, Windows Vista y XBox 360.

<sup>48</sup> Formato estándar de fuentes tipográficas

<sup>49</sup> [www.sig.upv.es/asignaturas/IMS/Tema01.InformaticaGráfica.pdf](http://www.sig.upv.es/asignaturas/IMS/Tema01.InformaticaGráfica.pdf)

<sup>50</sup> En base a la experiencia del grupo en desarrollo de videojuegos y artículos de profesionales como: Richard's Corner <http://www.xmission.com/~legalize/d3d-vs-opengl.html>

<sup>51</sup> High Level Shader Language



### Utilitarios

Además del software para desarrollo, se necesitará de otros programas, los cuales se especifican a continuación:

**Microsoft Office 2003:** Se usará para crear los documentos a presentar en cada etapa del proyecto, principalmente por su facilidad de uso y compatibilidad con el sistema operativo de desarrollo.

**Poseidón CE 3 para UML:** La aplicación de esta herramienta se realizará en la fase de análisis de requerimientos y diseño para la creación de diagramas UML. Se usará principalmente por su interfaz integrada, soporte de UML 2.0 y facilidad de uso.

#### **1.3.1.2. Hardware**

En esta sección se especifica los requisitos mínimos y recomendados de hardware del equipo informático usado para el desarrollo.

#### Características del Hardware necesario para el desarrollo del videojuego

En base a los requerimientos mínimos y recomendados de cada una de las aplicaciones (Sistema Operativo, lenguaje de desarrollo, API gráfica, Herramienta de edición de imágenes, modelado 3D y utilerías.) a usar en el desarrollo del videojuego se establecen las características mínimas y recomendadas de hardware que debe poseer el equipo informático de desarrollo.

| Hardware  | Mínimo                            | Recomendado  |
|---|-----------------------------------|--|
| Procesador  | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1 GHz   |
| Memoria RAM   | 320 Mb                            | 512 Mb   |
| Disco Duro  | 1.8 GB                            | 5.8 GB libres  |
| Unidad de CD o DVD  | Unidad CD o DVD                   | Unidad CD-RW / DVD   |
| Pantalla  | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits  |
| Teclado   | Teclado Microsoft o compatible    | Teclado Microsoft o compatible                                       |
| Ratón   | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |
| Tarjeta de video con soporte DirectX 9.0c y Shader Model 2. | Integrada                         | Independiente con 128 MB RAM y soporte para Shaders 2.0 en adelante. |

Cuadro 1.10: Características del equipo necesario para desarrollo.



Equipo disponible para el desarrollo del videojuego

**Computadoras:**

El equipo de desarrolladores cuenta con 4 computadoras que tienen las características expuestas en el cuadro 1.11.

| Equipo | Características   |
|--------|---|
| A      | <p><b>Procesador:</b> AMD Athlon 64 X2 Dual Core Processor 4800+2.51 GHz<br/> <b>Memoria RAM:</b> 1GB<br/> <b>Disco duro:</b> 300GB<br/> <b>Monitor:</b> Dell E176FP Black 17" LCD,1280x1024 pixels<br/> <b>Video:</b> ATI Rodeón 155x Series de 250 MB de video RAM, para shadders<sup>52</sup> de efectos 3.0<br/> <b>Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps</p>  |
| B      | <p><b>Procesador:</b> AMD Turion 64 ML-34 y 1MB de caché L2 (1.8 GHZ)<br/> <b>Memoria RAM:</b> 512 MB de memoria de sistema DDR, a 333 MHZ<br/> <b>Disco duro:</b>80 GB (4.200 rpm)<br/> <b>Monitor:</b> HP WXGA BrightView de alta definición de 15,4 pulgadas, 1280 x 800 píxeles<br/> <b>Video:</b> ATI RADEON XPRESS 200M IGP con 128 MB de memoria DDR (dedicada)<br/> <b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps</p> |
| C      | <p><b>Procesador:</b> Intel Celeron 3.2 GHz<br/> <b>Memoria RAM:</b> 1GB<br/> <b>Disco duro:</b> 160 GB<br/> <b>Video:</b> GeForce 7800 GTX 256<br/> <b>Monitor:</b> Monitor CRT Compaq MV7540e de 17 pulgadas, 1280 x 1024 píxeles<br/> <b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps</p>  |
| D      | <p><b>Procesador:</b> Intel Core Duo T2300 (1660 MHz)<br/> <b>Memoria RAM:</b>1GB<br/> <b>Disco duro:</b>80GB<br/> Monitor: Dell 14.1 inch 16:10 ,1440 x 900 pixels<br/> <b>Video:</b> Intel Graphics Media Accelerator 950 soporte pixel Shadder 3.0<br/> <b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps</p>  |

Cuadro 1.11: Equipo de Hardware (computadoras)

Periféricos de entrada/salida

En el cuadro 1.12 y 1.13 se describen las características de los periféricos de entrada y salida con que se cuenta para el desarrollo del proyecto.

<sup>52</sup> Shadder: Sombreador, termino entre desarrolladores de sistemas gráficos computaciones que hace referencia a un archivo de extensión .fx (de efectos) en donde se escribe código (hibrido entre Java-C# y C++), Ver el glosario en la Pág. 345.



| Cantidad | Equipo    | Características   |
|----------|-----------|---|
| 2        | Impresora | Canon ip-1800, Hasta 4800 x 1200 dpi, 20ppm en negro y 16 ppm en color. |
| 3        | Bocinas   | Genéricas.  |

Cuadro 1.12: Periféricos Salida

| Cantidad | Equipo           | Características          |
|----------|------------------|--------------------------|
| 1        | Gamepad Xbox 360 | Interfaz USB             |
| 1        | Micrófono        | Genérico                 |
| 4        | Mouse            | Genérico                 |
| 2        | Teclado          | Genérico, Inglés/Español |

Cuadro1.13: Periféricos Entrada

### 1.3.1.3. Recurso Humano

En el cuadro 1.14 y 1.15 se presenta el perfil académico mínimo y deseable que debe satisfacer el grupo de personas involucradas en la creación del videojuego.

| Administrador de proyectos informáticos   |              |                    |               |                     |
|---|--------------|--------------------|---------------|---------------------|
| <b>Funciones:</b> Coordinar todas las actividades de proyecto del grupo con el Objetivo de alcanzar las metas de cada etapa.  |              |                    |               |                     |
| <b>Nivel académico:</b> Ingeniero en Sistemas Informáticos  |              |                    |               |                     |
| <b>Características:</b><br>Líder en motivación así como negocio y solución de conflictos.<br>Capacidad de análisis y abstracción.<br>Creativo e innovador.<br>Alto sentido de ética profesional, responsable. |              |                    |               |                     |
| Conocimientos y habilidades   | Nivel mínimo | Experiencia mínima | Nivel deseado | Experiencia deseada |
| <b>Análisis y Diseño de sistema</b>   |              |                    |               |                     |
| Experiencia en análisis y diseño de sistemas.   |              | 2 años             |               | 3 años              |
| Dominio de tecnologías de análisis y diseño orientadas a objetos  | Intermedio   | 1 año              | Avanzado      | 2 años              |
| Dominio de UML 2.0  | Intermedio   | 1 año              | Avanzado      | 3 años              |
| <b>Programación.</b>  |              |                    |               |                     |
| Conocimiento de programación orientada a objetos.   | Intermedio   | 1 año              | Avanzado      | 2 años              |
| Experiencia en desarrollo de sistema para la plataforma Windows.  | Intermedio   | 1 año              | Avanzado      | 3 años              |
| Experiencia en control de calidad de software.  | Intermedio   | 1 año              | Avanzado      | 2 años              |
| <b>Administración de Proyectos informáticos</b>   |              |                    |               |                     |
| Experiencia en dirección y control de proyectos de desarrollo de sistemas.  |              | 2 años             |               | 4 años              |

Cuadro1.14: Nivel de conocimiento del coordinador de proyectos



| <b>Analistas Programadores de videojuegos</b>   |                     |                           |                      |                            |
|---|---------------------|---------------------------|----------------------|----------------------------|
| <b>Funciones:</b> Establecimiento de requerimientos, análisis y diseño del software, programación del videojuego. |                     |                           |                      |                            |
| <b>Características:</b><br>Responsabilidad y trabajo en grupo.<br>Apasionado por los videojuegos e innovador.     |                     |                           |                      |                            |
| <b>Conocimientos y habilidades</b>  | <b>Nivel mínimo</b> | <b>Experiencia mínima</b> | <b>Nivel deseado</b> | <b>Experiencia Deseada</b> |
| <b>Análisis y Diseño de sistema</b>   |                     |                           |                      |                            |
| dominio de tecnologías de análisis y diseño orientadas a objetos  | Intermedio          | 1 año                     | Avanzado             | 2 años                     |
| dominio de UML 2.0  | Intermedio          | 1 año                     | Avanzado             | 3 años                     |
| Manejo de la herramienta de productividad Visio 2003.   | Básico              | 6 meses                   | Avanzado             | 1 año                      |
| <b>Programación.</b>  |                     |                           |                      |                            |
| Conocimientos sobre Apis Gráficas para plataforma PC y XBOX-360   | Intermedio          | 1 año                     | Avanzado             | 3 años                     |
| Conocimientos de matemáticas 3D   | Intermedio          | 1 año                     | Avanzado             | 2 años                     |
| Programación de algoritmos de inteligencia artificial.  | Básico              | 1 año                     | Intermedio           | 2 años                     |
| Experiencia previa en creación de aplicaciones.   | Intermedio          | 6 meses                   | Avanzado             | 1 año                      |
| Dominio de Visual C# y programación orientada a objetos.  | Intermedio          | 1 año                     | Avanzado             | 2 años                     |
| Dominio de creación de terrenos, sprites, animación de mesh, manejo de cámaras y transformaciones en XNA 1.1.     | Básico              | 1 año                     | Avanzado             | 3 años                     |
| <b>Modelado 3D</b>  |                     |                           |                      |                            |
| Manejo de fragMOTION o Milkshape 3D   | Intermedio          | 1 año                     | Avanzado             | 2 años                     |
| Dominio de técnicas de texturizado.   | Intermedio          | 6 meses                   | Avanzado             | 1 año                      |
| Manejo de Photoshop CS2   | Intermedio          | 6 meses                   | Avanzado             | 1 año                      |
| Manejo de técnicas para crear modelos de bajos vértices.  | Intermedio          | 2 meses                   | Avanzado             | 6 meses                    |
| Manejo de técnicas para la animación esquelética.   | Intermedio          | 6 meses                   | Avanzado             | 1 año                      |

Cuadro 1.15: Perfil de conocimiento de analista programador

El recurso humano con el que se cuenta para el desarrollo del proyecto es el siguiente:





| Número | Nivel académico                                 | Puesto                                    | Responsabilidades   |
|--------|---|---|---|
| 1      | Ingeniero en sistemas Informáticos              | Director                                  | <ol style="list-style-type: none"> <li>1. Dirigir el proyecto</li> <li>2. Coordinar y motivar el equipo del proyecto</li> <li>3. Asesoría al equipo del proyecto.</li> </ol>  |
| 4      | Egresado de la carrera de Sistemas Informáticos | Analistas<br>Diseñadores<br>Programadores | <ol style="list-style-type: none"> <li>1. Levantamiento de requerimientos.</li> <li>2. Diseño del software</li> <li>3. Codificación del Videojuego.</li> <li>4. Realizar pruebas del software.</li> <li>5. Realizar la documentación y manuales pertinentes.</li> </ol> |

Cuadro 1.16: Recuso Humano.

El docente director cuenta con experiencia y conocimientos necesarios para desempeñar el cargo de administrador de proyectos informáticos, el equipo de desarrolladores cuentan con el conocimiento de un analista programador de videojuegos.

#### **1.3.1.4. Conclusiones factibilidad técnica**

Como resultado de este estudio técnico, se determino que actualmente se cuenta con la infraestructura tecnológica (hardware y software) necesario para el desarrollo del Software Interactivo para la Educación Medioambiental de la Población Salvadoreña. Además los integrantes de este proyecto están capacitados (y han creado para otros proyectos de la carrera) en programación 3D, tanto para juegos DirectX como para juegos desarrollados en XNA con calidad comercial. Al mismo tiempo, el equipo desarrollador cuenta con sólidos conocimientos en el manejo de entornos de desarrollo y lenguajes de programación como en el caso de *c#*, que es el lenguaje exclusivo para desarrollo de juegos XNA.

#### **1.3.2. Factibilidad económica**

Seguidamente se presenta el estudio económico realizado para determinar la viabilidad del desarrollo del software propuesto, se determinará si existe equilibrio económico entre los costos intrínsecos del videojuego y los beneficios que se derivarán de este, lo cual permitirá tener un mejor panorama financiero, para comprobar si en efecto, se obtendrán beneficios económicos del proyecto.

Como punto de partida de la factibilidad económica, se presentará un apartado en el que se expone la vida útil del software y la tasa de interés. Estos datos son necesarios para costear todos los gastos en los que se incurrirán durante el desarrollo e implementación del videojuego, para ser comparados con los gastos del sistema actual. Los costos de desarrollo son basados en costos actuales de mercado y sueldos de profesionales, conocimiento empírico



y análisis de actividades de desarrollo de un videojuego.

Los datos necesarios para realizar una comparación entre el sistema actual y el propuesto se obtendrán al estimar costos de inversión inicial, los costos de operación para ambos sistemas en un tiempo de 5 años, para luego obtener el beneficio total.

### 1.3.2.1. Vida útil del software

Para efectos del análisis económico del proyecto, se considero que el sistema tendrá una vida útil de 5 años<sup>53</sup>; sin embargo el tiempo de operación del mismo podrá prorrogarse en función de su capacidad de adaptación y la funcionalidad que presente.

### 1.3.2.2. Tasa de inflación

La tasa de interés a utilizar para los cálculos relacionados al método del valor actual neto es la tasa de inflación anual promedio de acuerdo a los últimos 10 años más la tasa de riesgo.

A continuación se presenta la tasa de inflación que se ha tenido en los últimos años

| Año                     | Tasa de inflación esperada (%) |
|-------------------------|--------------------------------|
| 2000                    | 4.3                            |
| 2001                    | 1.4                            |
| 2002                    | 2.8                            |
| 2003                    | 2.5                            |
| 2004                    | 5.4                            |
| 2005                    | 4.3                            |
| 2006                    | 4.9                            |
| 2007                    | 4.9                            |
| Tasa inflación promedio | 3.8 <sup>54</sup>              |

Cuadro 1.17: Tasa de Inflación anual

Teniendo estos datos, la tasa de interés TMAR se calcula de la siguiente forma:

$$\text{TMAR}^{55} = \text{tasa de inflación promedio} + \text{tasa de riesgo}$$

$$\text{TMAR} = 3.8\% + 5.0\%$$

$$\text{TMAR} = 8.8\%$$

<sup>53</sup> Tiene relación a posibles reformas a los proyectos ambientales por cambio de Gobiernos.

<sup>54</sup> Datos Publicados por el Banco Central de Reserva [www.bcr.com.sv](http://www.bcr.com.sv)

<sup>55</sup> Ver fuente en libro de Baca Urbina, Evaluación de Proyectos, también [http://www.itson.mx/dii/mconant/materias/ingeco/cap31\\_33.htm](http://www.itson.mx/dii/mconant/materias/ingeco/cap31_33.htm)



### **1.3.2.3. Costo de operación del actual Programa Nacional de Reciclaje.**

El objetivo principal de este proyecto es fomentar el manejo adecuado y responsable de los desechos sólidos, mediante la práctica cotidiana de separación. Promoviendo en los aproximadamente 20 mil niños y niñas de primero y segundo ciclo de nivel básico, involucrados en esta actividad, el cambio de actitudes y comportamientos mediante la ejecución de acciones concretas. El programa se ejecutará en 113 Centros Educativos de 11 municipios del Gran San Salvador; en Santa Tecla y Antiguo Cuscatlán del departamento de La Libertad.

El programa tiene un equivalente a 2 mil cuarenta horas clase, interactiva y de seguimiento. Finalmente se persigue conformar un comité de medio ambiente por cada Centro Educativo participante y el cual podrá estar conformado por docentes, estudiantes y padres de familia, para que a través de esto pueda generarse la sostenibilidad y seguimiento a dicho programa.

El alcance de este proyecto es de 20,000 estudiantes, a un costo promedio de \$20.00 (cada periodo de 6 meses se desembolsará \$200,000, teniendo en un año un costo de \$400,000), en resumen se tiene:

| <b>Rubro</b>                  | <b>Cantidad</b> |
|-------------------------------|-----------------|
| Costo del sistema actual      | \$400,000.00    |
| Alcance de estudiantes        | 20,000          |
| Costo promedio por estudiante | \$20.00         |

Cuadro 1.18: Detalle del sistema actual.

### **1.3.2.4. Costo del software propuesto SIEMPOSAL.**

Los datos necesarios para obtener el costo total del nuevo sistema son: Costos de inversión inicial, costos de operación en 5 años. Se obtendrá el costo promedio por estudiante, para luego hacer un análisis económico formal y más a fondo utilizando el método de Valor Actual Neto para determinar la viabilidad económica.

#### *Inversión Inicial*

La inversión inicial está relacionada específicamente, al costo de desarrollo del Software Interactivo para la Educación Medioambiental de la Población Salvadoreña y al costo respectivo para su operación en el primer año de implantación.



Para el cálculo del costo de inversión inicial del videojuego, se tiene que el sistema educativo salvadoreño, cuenta con 25,000 computadoras, distribuidas en sus 2,141<sup>56</sup> centros escolares que están provistos de un aula informática, con las cuales se atiende a más de 646,000 estudiantes, pero debido a que no se cuenta con el dato exacto de computadoras que tienen tarjeta de video recomendada para la ejecución del videojuego, se realizará la estimación de la compra de 25,000 tarjetas de video a un costo de \$15.00 cada una (precio de mayoreo), correspondientes al número de computadoras que el sistema educativo tiene, para la factible operación del videojuego.

El costo de los equipos, actualización y ensamblaje es despreciado ya que el Centro de Reacondicionamiento de computadoras, el cual funciona en la sede del ITCA de Zacatecoluca<sup>57</sup> tiene previsto brindar apoyo para la sostenibilidad del parque tecnológico del MINED gratuitamente, por lo que solo se tendría el costo de compra de tarjetas de video, como se muestra en el siguiente cuadro.

| Rubro   | Cantidad | Costo unitario (\$) | Monto(\$)  |
|---|----------|---------------------|------------|
| Tarjetas de video de 128 MB que soporten DirectX 9.0 Shader Model 2.0 32 bit. | 25,0000  | 15.00               | 375,000.00 |

Cuadro 1.19: Costos de tarjetas de video, para la operación del software.

A continuación se presenta el consolidado del costo de inversión inicial que se tendrá al implementar el videojuego

| Rubro   | Monto(\$)                |
|---|--------------------------|
| Costo de desarrollo del software  | 234,078.15 <sup>58</sup> |
| Capacitación y distribución directa del Software a través de CDs (no incluye Manual de Usuario impreso) | 2,484.60 <sup>59</sup>   |
| Tarjetas de video de 128 MB que soporten DirectX 9.0 Shader Model 2.0 de 32 bit                         | 375,000.00               |
| Total   | 611,562.75               |

Cuadro 1.20: Costos de inversión inicial del software interactivo.

<sup>56</sup> Datos publicados en la Memoria de labores, 2006-2007 del Plan Nacional de Educación 2021, del Ministerio de Educación, página 45-47

<sup>57</sup> lbd, página 47

<sup>58</sup> En base al costo estimado en la planeación de recursos, ver Anexo No 10, Pág. 361.

<sup>59</sup> Dato obtenido del calculo del costo de capacitar y distribuir directamente el software por medio de CDs a los 2,141 Centros Educativos que cuenta con un aula informática, en el plan piloto. Anexo No 11, Pág. 368.



### Alcance inicial y Costo Promedio por Estudiantes.

En base a los datos publicados en la **Memoria de labores, 2006-2007 del Plan Nacional de Educación 2021**, del Ministerio de Educación, se tiene que más de 646,000 estudiantes tienen acceso a las aulas informáticas. Estos podrán ser alcanzados por el software propuesto.

Además, en los datos publicados en el **VI Censo de Población y vivienda 2007**, se tiene que existen 153,283 hogares que tienen computadora, haciendo un total de 630,969 personas que en sus hogares podrían implementar el software. De esta población no se tiene el número exacto de los que no son estudiantes, pero podrían ser beneficiados por este proyecto.

Entonces, en base a lo anteriormente dicho, puede decirse que el proceso de concientización al utilizar el videojuego no está limitado a un determinado tiempo, como lo es con el programa actual, sino el estudiante también podrá jugar el videojuego en su casa, (si cuenta con un equipo).

El cálculo del costo promedio por estudiante (CPE), el primer año, se realiza mediante la siguiente fórmula:

$$\text{Costo promedio por estudiante} = \frac{\text{Inversión Inicial del Software}}{\text{No. de estudiantes con acceso a computadora}}$$

$$\text{Costo promedio por estudiante} = \frac{611,562.75}{646,000}, = 0.95$$

En resumen se tiene los datos siguientes para el software desarrollado.

| Rubro                                | Cantidad     |
|--------------------------------------|--------------|
| Costo inicial del software propuesto | \$611,562.75 |
| Alcance inicial de estudiantes       | 646,000      |
| Costo promedio por estudiante        | \$0.95       |

Cuadro 1.21: Detalle del sistema propuesto.

### Costos de Operación.

Los costos de operación en los años restantes, pueden ser calculados en base a los datos estadísticos presentados en la **Memoria de labores, 2006-2007 del Plan Nacional de Educación 2021**, del Ministerio de Educación, que se tiene un incremento anual de 72 aulas informáticas, con un promedio de 18 computadoras cada una, atendiendo a un aproximado de 463 alumnos por centro escolar, para las cuales se tendría un costo de operación anual.



Se considerará que la distribución del software se realizará de forma directa, se estimará la compra, grabación y distribución de un CD para los Centro Escolares a los cuales se les provea un aula informática en los próximos 5 años.

También se está considerando que el Ministerio de Medio Ambiente y Recursos Naturales, realice una convocatoria a los profesores que atenderán estas aulas informáticas, para distribuir y capacitarlos en el uso del videojuego, en un periodo de 3 días, siendo estos asistidos por 2 capacitadores con un costo de \$ 60.00 por día, cada uno. Esta capacitación, se realizará por zonas o rutas: occidental, central y oriental.

Además, para garantizar la funcionalidad del videojuego en las nuevas aulas informáticas, se estimará el costo de la compra de tarjetas de video.

En el siguiente cuadro se presenta el detalle de los costos anuales al implementarse el videojuego, para atender a 33,336 estudiantes.

| <b>Costo de capacitación y distribución directa del software</b>  |                  |
|---|------------------|
| <b>Rubro</b>  | <b>Monto(\$)</b> |
| Capacitación y Distribución del software al Ministerio del Medio Ambiente y Recursos Naturales MARN.                | 360              |
| Precio de adquisición de 72 CDs, (\$0.10 c/u):  | 7.20             |
| Precio de Grabación de CDs ( a \$0.30 por unidad X 72 unidades)   | 21.60            |
| Precio de 72 cajas de distribución ( a \$0.20 por unidad X 2,141 unidades)  | 14.40            |
| Costo por combustible   | 170.00           |
| Precio de 1,296 Tarjetas de video de 128 MB que soporten DirectX 9.0 Shader Model 2.0 de 32 bit (1,296 por \$15.00) | 19,440           |
| <b>Total</b>  | <b>20,013.20</b> |

Cuadro 1.22: Costo anual de operación del software propuesto.

En resumen se tiene como costos de operación en los años restantes, los datos presentados en la tabla siguiente.

| <b>Rubro</b>                                    | <b>Cantidad</b> |
|---|-----------------|
| Costo de operación anual del software propuesto | \$20,013.20     |
| Alcance de estudiantes por año                  | 33,336          |
| Costo promedio por estudiante                   | \$0.60          |

Cuadro 1.23: Detalle del software propuesto.



El costo del incremento en promedio de las 72 aulas informáticas equipadas con 18 computadoras con sus respectivas tarjetas de video, en el país es de \$20,013.20 anual, con lo cual se atenderá un total de 33,336 estudiantes.

*Beneficios Tangibles.*

En el sistema actual se tienen costos de operación de \$400,000.00 al año, para alcanzar a 20,000 estudiantes, teniendo que el costo promedio por estudiante (CPE) alcanzado es de \$20.00.

Con el software propuesto, se tiene un costo de operación de \$20,013.20 al año con un alcance de 33,336 estudiantes, teniendo un costo promedio por estudiante (CPE) alcanzado de \$0.60.

Para efectos de comparación se obtendrá la proporción del costo que se tendría en el sistema propuesto para atender a 20,000 estudiantes.

$$\begin{aligned}
 \text{Costo anual de} & & \text{costo promedio por} & & \text{No. de estudiantes} \\
 \text{operación del sistema} & = & \text{estudiante} & \times & \text{alcanzados en} \\
 \text{propuesto} & & & & \\
 & = & 0.60 * 20,000 & & \\
 & = & \$ 12,000 & & 
 \end{aligned}$$

El cálculo del beneficio, se obtendrá con la siguiente fórmula, considerando un alcance de 20,000 estudiantes al año en ambos sistemas.

$$\begin{aligned}
 \text{Beneficio Anual} & = & \text{costo de operación} & - & \text{costo de operación del} \\
 & & \text{del sistema actual} & & \text{software propuesto.} \\
 & = & \$400,000.00 - \$12,000.00 & & \\
 & = & \$ 388,000.00 & & 
 \end{aligned}$$

Se tendría un ahorro anual para el software propuesto respecto al sistema actual correspondiente a \$388,000.00 y para efectos de comparación, también se obtendrá para el sistema propuesto, la proporción correspondiente de la inversión inicial para atender a 20,000 estudiantes como se muestra a continuación.



$$\begin{aligned}
 \text{Proporción de la inversión inicial del sistema propuesto para 20,000 estudiantes} &= \frac{\text{Número de estudiantes alcanzados en el sistema actual}}{\text{Número de estudiantes alcanzados con el sistema propuesto}} \times \text{Inversión inicial del sistema propuesto correspondiente a 33,336} \\
 &= \\
 &= \$18,933.83
 \end{aligned}$$

En el siguiente cuadro se muestra el resumen de los datos.

| Descripción del flujo                 | Vida Útil del Software Propuesto (Años) |               |               |               |               |               |
|---------------------------------------|---|---------------|---------------|---------------|---------------|---------------|
|                                       | 0                                       | 1             | 2             | 3             | 4             | 5             |
| Costos anuales del software propuesto | \$18,933.83                             | \$ 12,000     | \$ 12,000     | \$ 12,000     | \$ 12,000     | \$ 12,000     |
| Beneficios anuales                    | \$0.00                                  | \$ 388,000.00 | \$ 388,000.00 | \$ 388,000.00 | \$ 388,000.00 | \$ 388,000.00 |
| Utilidad Anual                        | (\$18,933.83)                           | \$376,000.00  | \$376,000.00  | \$376,000.00  | \$376,000.00  | \$376,000.00  |

Cuadro 1.24: Flujo de efectivo del software a cinco años.

Método del Valor Presente Neto de Costos en ambos sistemas.

Es el valor monetario que resulta de la suma de costos de operación obtenidos (al atender a 20,000 estudiantes en ambos sistemas durante los 5 años de vida útil) a la inversión inicial.

El valor presente neto se obtiene de la siguiente fórmula:

$$VPN = P + A * (P/A, i, n)$$

Donde:

- VPN = Valor Presente Neto
- P = Inversión Inicial.
- A = Costo anual.
- i = Tasa mínima aceptable de rendimiento o tasa de descuento que se aplica para llevar a valor presente de 8.8%
- n = Vida Útil del sistema = 5 años
- (P/A, i, n) = Factor de valor presente P, para una serie de pagos iguales A.
- (P/A, 8.8%, 5) = 3.992710037





Realizando los cálculos se tiene:

### **Sistema Actual**

*VPN = Valor Presente Neto*

*P = Inversión Inicial = \$ 400,000.00.*

*A = Costo anual del sistema actual = \$400,000.00*

$$\begin{aligned} VPN &= P + A(P/A, i, n) \\ &= 400,000.00 + 400,000.00(3.992710037) \\ &= 1,997,084.01 \end{aligned}$$

### **Sistema Propuesto (Videojuego)**

*VPN = Valor Presente Neto*

*P = Inversión Inicial = \$18,933.83*

*A = Costo anual del sistema actual \$12,000.00*

$$\begin{aligned} VPN &= P + A(P/A, i, n) \\ &= 18,933.83 + 12,000.00(3.992710037) \\ &= 66,846.35 \end{aligned}$$

Por lo que el beneficio en el presente es:

$$\begin{aligned} \text{Beneficio} &= \text{Valor Presente Neto del Sistema Actual} - \text{Valor Presente Neto del Sistema Propuesto} \\ &= 1,997,084.01 - 66,846.35 \\ &= \$ 1,930,237.66 \end{aligned}$$

Como se ha observado, al implementar el Videojuego como herramienta de concientización aplicado a 20,000 estudiantes se tendría un beneficio total de \$ 1,930,237.66.

Además, en el sistema actual solo se está cubriendo la temática del reciclaje, a diferencia del software propuesto que está cubriendo la temática del reciclaje y otras temáticas medioambientales. Esto pone en desventaja al sistema actual ya que con el sistema propuesto se puede tener un alcance mayor en cuanto a temáticas a un costo menor, es decir, a un costo menor se está impartiendo no solo una temática, sino cinco temáticas.



### Beneficios intangibles de SIEMPOSAL

También se tendrían los siguientes beneficios intangibles al implementar software.

- Reducción de los tiempos de aprendizaje sobre temas medio ambientales.
- Incremento del alcance geográfico de la educación ambiental de El Salvador.
- Incrementar los conocimientos ambientales.
- Apoyar la formación de valores positivos en la personalidad de los jóvenes.
- Enfocar las aptitudes del estudiante en proteger y mejorar el medio ambiente.
- Crea nuevas pautas de conductas ambientales.
- Promueve la participación en proyectos ambientales
- Incremento de la satisfacción de aprendizaje en el estudiante.
- Retroalimentación de los temas pro-medioambiente
- Aumento de la motivación y la atención en temáticas ambientales.
- Disminución de la deforestación, ya que en el país, solo el 3%, aproximadamente está cubierto de vegetación original y el 75% de los suelos están erosionados (Informe GEO 2002)
- Disminución de la emisión de CO<sub>2</sub> y emisiones contaminantes para el aire, las cuales provocan enfermedades Infecciosas Respiratorias Agudas (IRAS), las que han provocado el 80% de las consultas en los últimos tres años (Ministerio de Salud Pública).
- Reducir el vertido de desechos en los ríos.
- Aumento del respeto por la flora y fauna, ya que 719 especies de vida silvestre están en peligro de extinción.
- Disminuir la producción de basura que actualmente es de 2,347.26 ton/día, de la cual solo 1% son botaderos-comporteras.
- Apoya la mitigación del cambio climático.
- Aumentar Calidad del agua.
- Aumento del índice de esperanza de vida de la población que actualmente es de 0.74



### **1.3.2.5. Conclusiones factibilidad económica**

La propuesta de implantación de un videojuego interactivo es económicamente factible, ya que el valor actual neto es positivo, esto implica la aceptación sobre la inversión en el videojuego propuesto, por lo que se sugiere que las autoridades correspondientes implementen este nuevo método poco convencional en sus programas de capacitación y concientización de la población para con el medioambiente. El sistema genera un beneficio neto de \$1,930,237.66 calculados para el presente a partir de los costos y beneficios iniciales y de operación de ambos sistemas para 5 años. El costo de implantar el videojuego es de \$66,846.35, en contraste con el costo del sistema actual que es de 1,997,084.01. Esto nos indica que al comparar ambas inversiones en su equivalente al valor presente, la inversión que tendría el nuevo sistema es menor, lo que genera un beneficio estimado para los siguientes cinco años de vida útil del software propuesto. Por lo tanto, al observar los resultados obtenidos de la aplicación de las distintas técnicas de evaluación de la viabilidad económica se puede afirmar que el software propuesto es económicamente factible, ya que existe un ahorro económico, se tiene un mayor alcance de estudiantes, un mayor alcance en las temáticas abordadas y además permitirá que el estudiante no solo aprenda en las aulas informáticas, sino que aún en su casa podría continuar el aprendizaje, dando como resultado una mayor concientización en cuanto al medioambiente.

### **1.3.3. Factibilidad operativa**

Este estudio se realizó con el fin de establecer, si se pondrá en marcha el software propuesto, aprovechando los beneficios que ofrece a los ciudadanos y estudiantes que hagan uso del videojuego.

#### **1.3.3.1. Aceptación del software**

El uso de nuevos métodos para llegar a la población de El Salvador es necesaria, según lo ha manifestado el Ministerio de Medio Ambiente y Recursos Naturales, por lo cual ha exteriorizó la aceptación del videojuego como una estrategia innovadora para concientizar sobre todo a la Juventud del país.

El Ministerio de Educación por medio de las aulas informáticas, pretende promover el uso de cualquier aplicación con fines educativos, por lo cual se cuenta con el apoyo de dicha institución para difundir el videojuego en la población estudiantil.



Se ha comprobado por medio del estudio de campo que el 83% de los estudiantes gusta de los videojuegos (Figura 1.8).

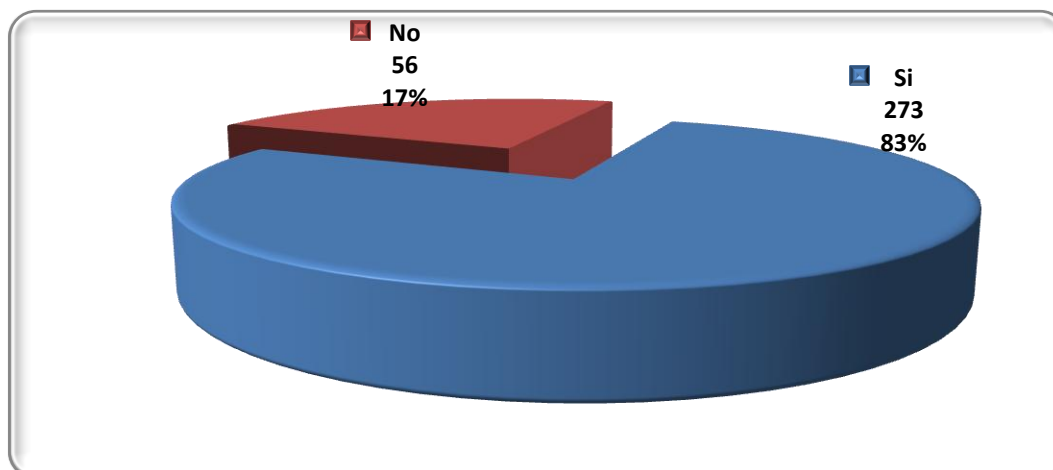


Figura 1.8: Opinión sobre los videojuegos educativos.

Este es un indicador del grado de aceptación que tendrá el videojuego en el sistema educativo de El Salvador, la población estudiantil considera que la implantación de esta herramienta para educar es una estrategia excelente (Figura 1.9.). Esto indica que se tendrá una gran aceptación de los usuarios finales.

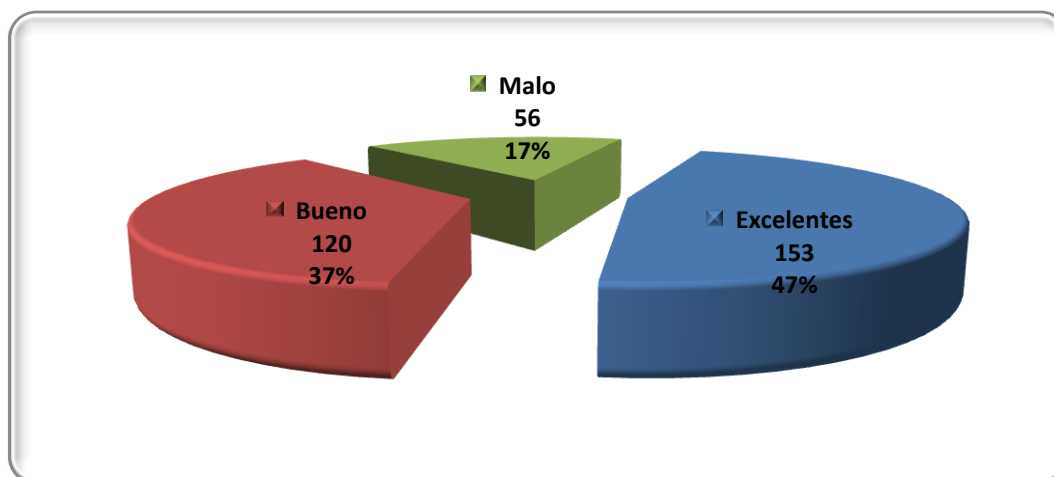


Figura 1.9: Opinión sobre los videojuegos educativos

El diseño del software será intuitivo, cualquiera sin mayor conocimiento en informática podrá usarlo, lo que se traduce en un videojuego de fácil manejo y comprensión.

Se tiene la ventaja que la mayoría de estudiantes ya han manipulado previamente videojuegos, por lo cual se considera que estos usuarios cuentan con la experiencia necesaria para operar la aplicación.



### 1.3.3.2. Volúmenes de datos

Las operaciones que se estiman almacenan datos son las siguientes:

| Operación             | Variables         | Tipo de dato | Tamaño   |
|-----------------------|-------------------|--------------|----------|
| Almacenaje de puntaje | Ecodólares        | Doble        | 32 bytes |
|                       | Ecopuntos         | Entero       | 16 bytes |
| Nivel de Vida         | Vida              | Entero       | 16 bytes |
|                       | Número de vidas   |              |          |
| Avance de Niveles     | Nivel             | Entero       | 16 bytes |
| Nuevo Jugador         | Nombre jugador    | Cadena(30)   | 60 bytes |
|                       | descripción       | Cadena(30)   | 60 bytes |
|                       | Contraseña        | Entero       | 16 bytes |
|                       | Nivel de oxígeno  | Cadena(8)    | 16 bytes |
|                       | Rapidez           | Doble        | 32 bytes |
|                       | Personaje elegido | Entero       | 16 bytes |
|                       | Posición inicial  | Vector 3     | 96 bytes |
| Guardar Posición      | OtraPosición      | Vector 3     | 96 bytes |

Cuadro 1.25: Operaciones del Video Juego.

En base al Cuadro 1.25, son necesarios (1Kb) aproximadamente para almacenar datos de partida por usuario, considerando que en todo el sistema educativo de El Salvador hay 25 alumnos por cada computadora<sup>60</sup>, se espera como máximo tener esta cantidad de usuarios PC, por lo que en total se almacenará 375 Kb, aproximadamente 0.4 Mb lo cual es mínimo.

### 1.3.3.3. Software para la operación del videojuego

Para el funcionamiento del videojuego se necesita el siguiente software:

| Software  | Disponibilidad                                  | Espacio Requerido en disco |
|---|---|----------------------------|
| Microsoft .Net Framework 2.0 SP1                    | Gratis  | 185 Mb                     |
| Microsoft Visual C++ 2005                           | Gratis  | 5.07 Mb                    |
| Bibliotecas XNA 2.0 Runtime                         | Gratis, compatibles con PCs x86, x64 y XBOX-360 | 2.05 Mb                    |
| DirectX 9.0c End-User Runtime                       | Gratis  | 61.44 Mb                   |
| L&H TTS 3000 Español y Microsoft Speech 4.0 Runtime | Gratis  | 2 MB                       |
| Recursos de videojuego                              |   | 970 Mb                     |
| Total de espacio                                    |   | 1225.56 Mb                 |

Cuadro 1.26: Software para la ejecución del videojuego<sup>61</sup>

Se necesita de 1,225.96 Mb, aproximadamente 1.2<sup>62</sup> Gb espacio en disco duro, para que el Videojuego opere correctamente.

<sup>60</sup> Informe CONECTATE, MINED, Pág. 13.

<sup>61</sup> Ver Anexo No.8: Requerimientos de hardware para el software de desarrollo, Pág. 356.

<sup>62</sup> No incluye el espacio en disco utilizado por el sistema operativo con el que cuente la máquina en la que se ejecute el videojuego.



En base al software necesario para la operación del videojuego se establecen las características mínimas y recomendadas de hardware que debe cumplir el equipo informático para instalar y ejecutar el videojuego.

| Hardware        | Mínimo                            | Recomendado                |
|-----------------|-----------------------------------|----------------------------|
| Procesador      | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1.8 GHz |
| Memoria         | 256 Mb                            | 512 Mb                     |
| Disco Duro      | 1.2 GB                            | 2 GB                       |
| Tarjeta gráfica | Integrada con Shader Model 2.0    | Shader Model 2.0           |
| Pantalla        | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits    |
| Teclado y rato  | Microsoft o compatible            | Microsoft o compatible     |

Cuadro 1.27: Características del hardware para la operación del videojuego

El ministerio de educación (MINED) en sus aulas informáticas dispone de hardware con las características<sup>63</sup> siguientes:

| Cantidad | Descripción  |
|----------|--|
| 19,750   | <b>Procesador:</b> Pentium IV<br><b>Memoria RAM:</b> 256 MB<br><b>Disco duro:</b> 60GB<br><b>Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps  |
|          | <b>Procesador:</b> Pentium III<br><b>Memoria RAM:</b> 128 MB<br><b>Disco duro:</b> 60GB<br><b>Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps |
| 5,250    | <b>Procesador:</b> Pentium IV<br><b>Memoria RAM:</b> 512 MB<br><b>Disco duro:</b> 60GB<br><b>Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps  |

Cuadro 1.28: Hardware disponible para la operación del Videojuego.

#### 1.3.3.4. Periféricos

Todas las computadoras de las Aulas Informáticas tienen Mouse y teclado. Actualmente el MINED, desconoce la cifra exacta de computadoras que cuentan con tarjeta gráfica, no obstante, el software será diseñado para ejecutarse en PC con requerimientos mínimos

#### 1.3.3.5. Conclusión factibilidad operativa.

Con el estudio operativo realizado se determinó que se cuenta con las capacidades de hardware y software necesario para soportar el incremento de almacenamiento anual en disco, así como con los conocimientos y aceptación necesarios para manipular el videojuego.

<sup>63</sup> Información proporcionada por el departamento de Infraestructura Tecnológica MINED



### 1.3.4. Factibilidad ambiental

Para el funcionamiento óptimo del software a desarrollar, sin que este dañe o perjudique al exterior, es necesario cumplir con las siguientes normas ambientales:

#### 1.3.4.1. Ruido

Se recomiendan que se mantengan los niveles de sonido para los efectos sonoros incorporados en el videojuego en un rango de 55 dB, para no provocar ningún tipo de efecto estresante y la comunicación en el entorno donde este se juegue, no se vea afectado, esto es debido a que el oído humano es capaz de detectar variaciones de presión acústica comprendidas entre 20  $\mu$ Pa (nivel umbral de percepción) y 200  $\mu$ Pa (nivel umbral del dolor), o lo que es lo mismo, entre 0 dB y 140 dB.

| Niveles en decibelios dB | Tipo de comunicación        |
|--------------------------|-----------------------------|
| 140                      | Umbral de dolor             |
| 100                      | Comunicación casi imposible |
| 80                       | Hay que gritar              |
| 50                       | Comunicación posible        |
| 20                       | Comunicación fácil          |
| 0                        | Umbral de audición          |

Cuadro 1.29: Niveles de audición

De acuerdo a lo especificado en el cuadro 1.29, el ambiente de trabajo debe ser moderadamente silencioso como para no molestar y distraer a los demás usuario que se encuentren en el aula informática. El ruido de fondo no debe sobrepasar los 55 dB.

#### 1.3.4.2. Seguridad Física

Se refiere a la protección física, en cuanto a operaciones de mantenimiento preventivo ante un peligro, riesgo o daño.

Los cables de los equipos pueden constituir una fuente de peligros y, en cualquier caso, siempre constituyen un estorbo; por lo tanto, se utilizará protectores como objetivo de evitar dichos problemas.

#### 1.3.4.3. Iluminación

La iluminación del entorno grafico del juego se considera estándar para monitores VGA y SVGA, se recomienda utilizar tecnología de despliegue de gráficos de plasma o monitores



planos, ya que el voltaje y por ende el consumo eléctrico es mucho menor que los monitores de raster<sup>64</sup> convencional (tubo).

#### **1.3.4.4. Temperatura**

El procesamiento en el CPU y GPU<sup>65</sup> es igual y genera la misma temperatura que cualquier aplicación, ya se limita a la capacidad en GHz propia de cada equipo, no obstante se recomienda instalar la computadora en lugares ventilados y sin objetos que obstruyan los fans de enfriamiento ya que eventualmente, por la termodinámica, la temperatura se expandirá a otros objetos cercanos y creará un ambiente desagradable para el usuario. Al mismo tiempo, un equipo que se mantenga frío tiene una mayor vida útil que aquel expuesto a temperaturas altas constantes, por lo que se recomienda un rango de temperatura que oscile entre los 18°C y 24°C.

#### **1.3.4.5. Ergonomía**

Ya que, dependiendo del lugar y la autoridad mentora que asigne a los usuarios un tiempo de juego (puede ser en aulas virtuales en la escuela o en la comodidad de una habitación en un hogar del país), se recomienda mantener una postura alineada, pues el usuario, puede pasar horas sin estar consiente de ello al involucrarse en la historia y manipulación del videojuego y esto a la larga, puede ocasionar problemas quiroprácticos y/o estrés a la columna vertebral y músculos aledaños. El uso de un teclado ergonómico es recomendable a pesar de no ser necesario en la manipulación del videojuego, no obstante, se aconseja el uso de un control de juegos dedicado (gamepad) compatible con la tecnología XBOX-360, ya que su ergonomía permiten a los usuarios no causar estrés a sus manos, pues estas se acoplan alrededor del control de forma natural, al mismo tiempo, añade a la experiencia del usuario, especialmente aquellos controles con motores de vibración incorporados.

#### **1.3.4.6. Conclusión de la factibilidad ambiental**

Si bien, parece una paradoja el utilizar electricidad para utilizar un software que permita a las personas concientizarse sobre el buen trato y sustentabilidad del medioambiente, esta es un mal necesario, no obstante, las técnicas y recomendaciones arriba mencionadas, permiten minimizar el uso eléctrico, así como también, malestares musculares, daños a tendones si son

---

<sup>64</sup> Raster: Píxeles dibujados en pantalla por un haz de electrones que se disparan desde un acelerador de electrones o tubo desde la parte trasera de un monitor hasta la pantalla.

<sup>65</sup> GPU: Graphics Processing Unit, Unidad de Procesamiento de Gráficos.





acatadas por los usuarios y 'ruido' térmico, sonoro e iluminativo al lugar que contiene la computadora cliente. La cantidad de intrusión del software en contexto al ambiente que lo contiene no es mayor que otras aplicaciones de uso diario, exceptuado la naturaleza adictiva propia de este, y va a depender de las personas como mentores, docentes y padres de familia el delimitar los tiempos de uso del software.

### **1.3.5. Conclusiones factibilidades del proyecto**

A nivel técnico se cuenta con el software, hardware y el recurso humano necesario para realizar el proyecto.

Económicamente el proyecto es factible ya que representa ahorros anuales de \$388,000.00 con respecto a metodologías tradicionales, teniendo un beneficio total en función del valor presente de \$1,930,237.66.

Los usuarios (estudiantes) de este software cuentan con conocimientos básicos de computación, lo que permitirá que fácilmente se adapten a este.

La aplicación del videojuego en las aulas informáticas tiene el aval de las máximas autoridades del Ministerio de Medioambiente, lo cual asegura que este software será distribuido e instalado en los centros escolares del país.

Existen las condiciones ambientales en las aulas informáticas que son necesarias para que el uso del software no influya perjudicialmente a su entorno.

Se garantizará que el videojuego no exponga al usuario a altos niveles de ruido, temperatura y uso moderado del recurso del ordenador, así como los contrastes de color de las pantallas.

## **1.4. RESULTADOS ESPERADOS**

### **1.4.1. Operativos**

- Aumento del nivel y calidad de concientización ambiental en los estudiantes mediante la técnica de inmersión del individuo en un ambiente virtual agradable, familiar y seguro, libre del estrés que a veces se siente en grupos sociales, en el caso de salón de clases o eventos especiales de educación ambiental.
- Entorno virtual que refleje la realidad y localidades propias del país, a las que muchos estudiantes encontrarían familiar.



- Efectos especiales aplicados a cada objeto y propios de juegos de séptima generación como iluminación realista, detalles fidedignos de sus contrapartes del mundo real, movimientos reales, interacciones entre los objetos y el usuario.
- Paisajes, ciudades, recursos naturales de El Salvador
- Niebla, fuego, humo, reflexiones y transparencias.
- Cielos simulados con nubes y distintos movimientos.
- Lo que muestra la cámara ayuda a descubrir los enigmas del juego, pues proporciona pistas de hacia dónde ir o que pared golpear.
- Efectos de agua, cascadas de gran diseño, con reales reflejos de árboles y arbustos son de categoría real.
- Animaciones individualizadas, tanto de ataque como de defensa o de muerte, especialmente las del protagonista quien tiene una gran variedad de combos y movimientos distintos para cada enemigo en los niveles en donde se enfrenta a criaturas del futuro (una metáfora al medioambiente hostil que puede existir en el planeta si no se hace nada en la actualidad para permitir un medioambiente sustentable y sano).
- Cada nivel contiene una serie de check points<sup>66</sup>, que son puntos hasta donde queda guardado el juego, para cuando el personaje muere poder empezar desde el último check point, avances en el juego, en la vida y el poder acumulado.
- Las melodías son capaces de ambientar perfectamente cada momento de la aventura.
- La Inteligencia artificial logra conquistar al jugador a través de todo el juego.
- Los paisajes son una emulación de la realidad.
- Los menú, toda la información necesaria en las opciones de Menú
- Estadísticas de usuarios del software
- Estadísticas de índices medio ambientales, Puntaje del jugador, Mensajes de alerta, entre otros.

#### 1.4.2. Tácticos

- Eficiencia en la enseñanza de temáticas ambientales
- Reducción en el tiempo necesario para retroalimentar los temas ambientales.
- Existencia de un control de aprendizaje sobre temáticas ambientales, por medio de la bitácora del jugador.
- Apoyo a la metodología de enseñanza

---

<sup>66</sup> Puntos de Control



- La naturaleza adictiva de los videojuegos permitirán “bombardear” al usuario del software de manera constante con mensajes positivos, permitiendo un proceso de concientización fluido y casi inconsciente en el estudiante usuario.

### 1.4.3. Estratégicos

- Al implementar el software se tendría un ahorro total de \$ 1,930,237.66<sup>67</sup> durante los cinco años de vida útil del software.
- Mayor alcance de la población, actualmente el Programa de Concientización Ambiental del MARN alcanzan a 113 centros escolares y aproximadamente 20,000 estudiantes de primer y segundo ciclo, esto debido al presupuesto limitado de \$200,000.00 por seis meses. Con la implementación del videojuego se puede alcanzar a una población de 646,000 estudiantes inicialmente, con un incremento anual de 33,336 estudiantes, esto debido a que se tiene la apertura de nuevas aulas informáticas en cada año, haciendo un total de 812,680 estudiantes con acceso a las aulas informáticas en un periodo de 5 años. Indudablemente, los resultados positivos en el trato del medioambiente tendrían una mejoría notable a corto, mediano y largo plazo.
- Mayor alcance en las temáticas, debido a que el actual Programa Nacional “Educación Ambiental para el Reciclaje” solo aborda el tema de desechos sólidos. Con la implementación del videojuego, se tiene el alcance de cinco temáticas las cuales se abordarían mediante videos o mediante la interacción directa del jugador en el videojuego. Es decir, se puede impartir cinco temáticas al precio de una, y a un costo menor, a una población mayor de estudiantes.
- El estudiante no se limitaría a recibir esta información una vez de parte de profesores y charlas ambientalistas, si no que recibiría esta información de forma constante y periódica cada vez que utilice el software, ya sea en su centro escolar o en su casa (si cuenta con computadora en su casa), divirtiéndose y aprendiendo a la vez. Además con el videojuego se podría alcanzar no solo a los estudiantes, sino también aquellas personas que cuentan con un computador que cumplan con los requerimientos de operación para el videojuego, en sus hogares.

---

<sup>67</sup> Ver Factibilidad Económica, Pág. 53



## **CAPITULO II: SITUACIÓN ACTUAL**



## 2. SITUACION ACTUAL

### 2.1. ANALISIS DE LA SITUACION ACTUAL

El Ministerio de Medioambiente y Recursos Naturales (MARN) tiene como objetivo promover la participación de los diferentes sectores de la población salvadoreña en programas destinados a la protección y prevención del deterioro ambiental<sup>68</sup>.

Las líneas fundamentales de acción de esta institución son:

- Fortalecimiento de las capacidades institucionales y sectoriales de la gestión ambiental, orientado a la solución de la problemática.
- Promover la plena participación de mujeres y hombres con equidad en el desarrollo sustentable, bajo el marco Constitucional de los acuerdos regionales e internacionales.
- Educación Ambiental.

Dando énfasis a este último objetivo, eje central de este documento, actualmente, la institución junto con el apoyo del gobierno central, es protagonista de uno de los programas de concientización ciudadana más fuerte en el país. El 5 de marzo de 2008, el Presidente de la República, acompañado del Ministro de Medio Ambiente y Recursos Naturales, realizó el lanzamiento del Programa Nacional de “Educación Ambiental para el Reciclaje”.

El objetivo principal de este proyecto es fomentar el manejo responsable de los desechos sólidos, mediante la práctica cotidiana de separación, promoviendo en aproximadamente veinte mil estudiantes de primero y segundo ciclo de nivel básico, el cambio de actitud y comportamiento mediante la ejecución de acciones concretas. El programa se está ejecutando en 113 Centros Educativos de 11 municipios de San Salvador; Santa Tecla y el departamento de La Libertad.

Entre las actividades planificadas se encuentran la implementación de una campaña interactiva de concientización ambiental, mediante una clase impartida por tres personajes denominados: Eco Ambiente, Latita y Súper Plástico; la cual impulsa en los estudiantes el aprendizaje sobre separación, recuperación y comercialización de plástico y aluminio.

---

<sup>68</sup> Artículo No. 10 de la Ley del Medio Ambiente de El Salvador.



El programa tiene un equivalente a dos mil cuarenta horas clases interactivas y de seguimiento. Finalmente se persigue conformar un comité de medio ambiente por Centro Educativo participante, los principales integrantes son docentes, estudiantes y padres de familia.

El Programa Nacional, recibe el respaldo oficial del Plan de Gobierno País Seguro<sup>69</sup>, con énfasis en promover el cuidado del Medio Ambiente por medio de las siguientes acciones estratégicas: la concientización e involucramiento de la sociedad civil, la empresa privada, las comunidades y los estudiantes con respecto al manejo adecuado de los desechos sólidos. Los 11 municipios en los cuales se desarrolla este proyecto son: Ciudad Delgado, Apopa, Ayutuxtepeque, San Salvador, Soyapango, Ilopango, San Marcos, Mejicanos, Cuscatancingo, La libertad y Antiguo Cuscatlán. Se espera recuperar aproximadamente unas 7 mil libras de plástico y 4 mil quinientas de aluminio mensualmente.

Las que serán retiradas por una empresa recolectora para ser trasladado a las empresas recicladoras para su comercialización, esos fondos servirán para fines netamente educativos. El proyecto tendrá una duración de 6 meses y la inversión económica del mismo asciende a más de doscientos mil dólares, sin incluir el aporte técnico del Ministerio de Educación.

Este proyecto planeado para 6 meses alcanza a veinte mil niños, el número de potenciales “guardianes del medioambiente” es corto, comparado con el número estimado de la población estudiantil en el área metropolitana que se estima es de aproximadamente 337,709 estudiantes<sup>70</sup>.

El reducido alcance de estudiantes del proyecto actual se debe a factores principalmente económicos y recurso humano capacitado. No obstante dicha limitante, el factor humano es una fortaleza propia del programa en ejecución, principalmente, por la capacidad técnica especializada.

Uno de los principales temas dentro del programa actual es la gestión de desechos sólidos que es parte de las campañas de concientización y educación ambiental, el solo hecho de educar a la población con respecto a esta actividad y por ende su puesta en práctica de buenos hábitos de reciclaje, involucra un ahorro cuantitativo en gastos de manipulación de desechos sólidos, mejora en la salud poblacional y limpieza medioambiental, eliminando focos de contaminación y daños a largo plazo al medioambiente.

---

<sup>69</sup> <http://www.rree.gob.sv/sitio/sitio.nsf/0/9152999b18c314a706256eb600692813?OpenDocument>

<sup>70</sup> Anexo No. 6: Estudiantes por grupos etarios, Pág. 355.



Si bien existen campañas de educación sobre el correcto reciclaje de desechos sólidos, la investigación realizada, no ha generado cifras sobre contenedores de reciclado en el área metropolitana.

La figura 2.1 muestra como se distribuye actualmente los gastos y aprestamiento de servicios en el área de recolección de desechos sólidos, esta estructura de compartición de servicios no ha variado mucho desde 2001<sup>71</sup>.

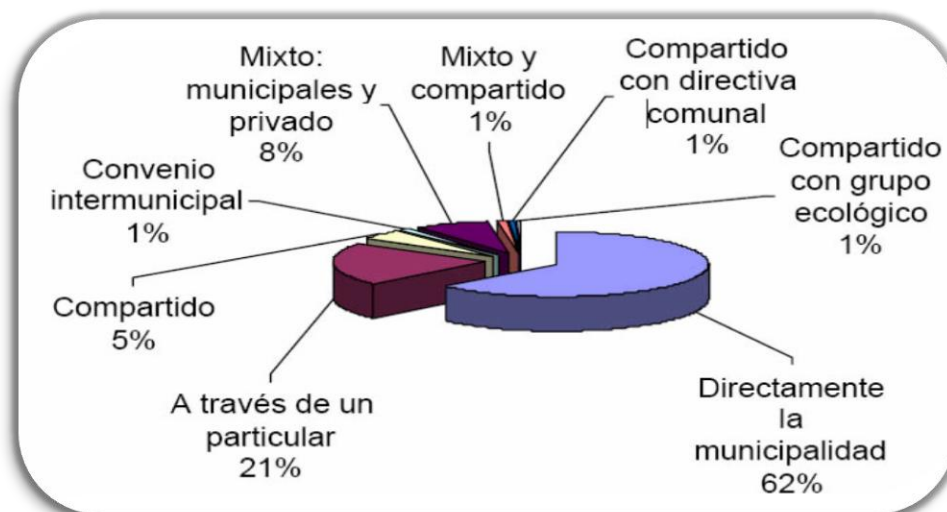


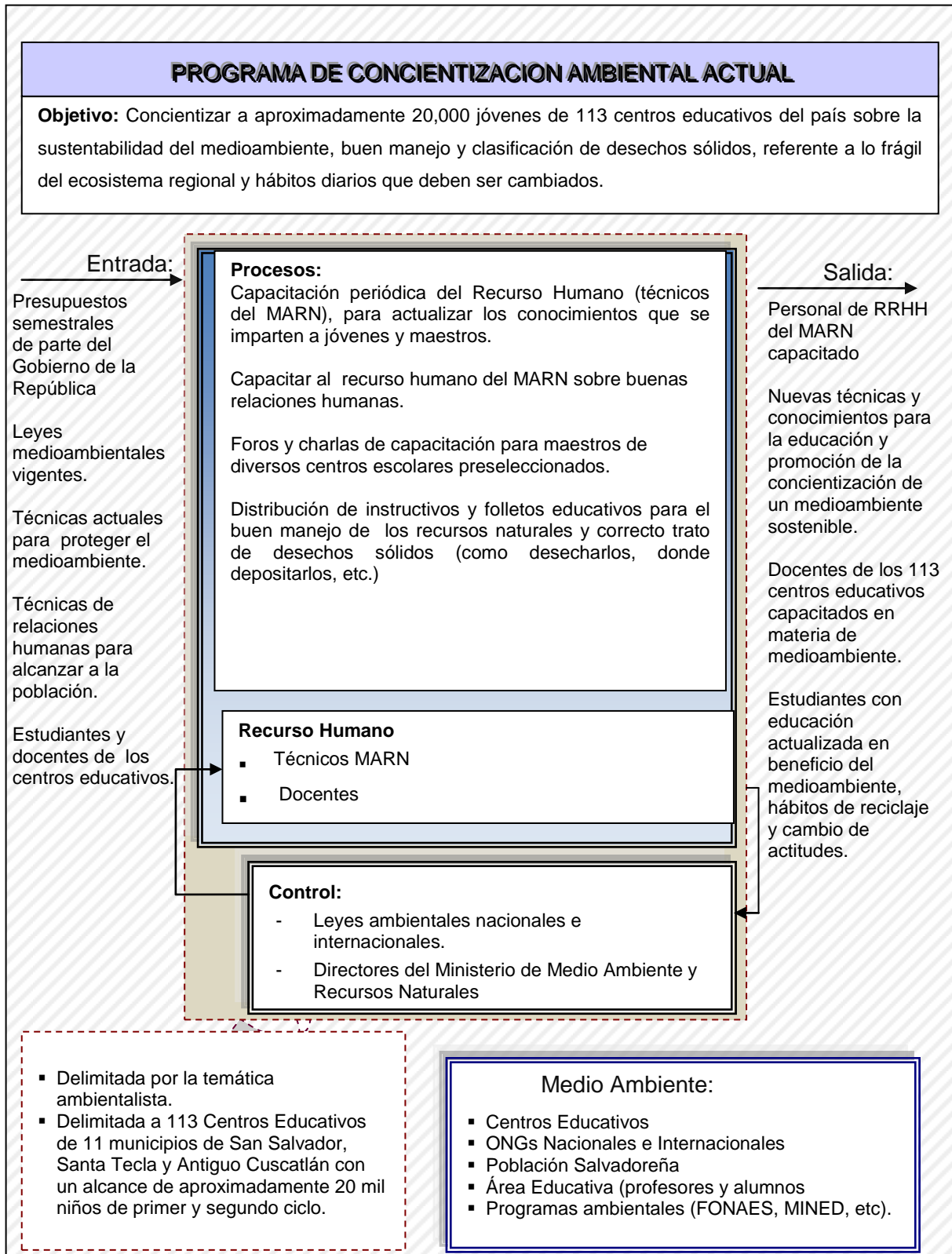
Figura 2.1: Prestatarios, servicio de recolección de desechos sólidos

Como puede observar el 62% de los municipios brinda el servicio directamente. El resto lo hace a través de un particular o mediante mezclas de ambos modelos de gestión. La contratación de servicios particulares de recolección y transporte, no se hace a nivel formal.

<sup>71</sup> [www.mspas.gob.sv/p\\_salud\\_ambiental5.asp](http://www.mspas.gob.sv/p_salud_ambiental5.asp)



### 2.1.1. Diagrama de Enfoque de Sistemas







## **2.1.2. Descripción de los componentes del Enfoque de Sistemas**

### **2.1.2.1. Objetivo:**

En lo general, se pretende alcanzar a un número seleccionado de instituciones y estudiantes de primer y segundo ciclo para educarles sobre el buen trato de los recursos naturales, reciclaje de desechos sólidos y concientización ambiental. En lo específico, se pretende formar el recurso humano que requiere el país en el campo científico-tecnológico, en el ámbito ecológico, que con función social contribuya al desarrollo económico y mejora en el nivel de vida a través de la conservación del medio ambiente y racionalización de los recursos naturales.

### **2.1.2.2. Entradas**

- Presupuestos semestrales de parte del Gobierno de la República.
- Leyes medioambientales vigentes.
- Técnicas actuales para el buen trato al medioambiente.
- Estrategias de información para alcanzar a la población.

### **2.1.2.3. Procesos**

- Capacitación periódica del personal de Recurso Humano (Técnicos del MARN), para actualización de conocimientos a impartir a los jóvenes y maestros.
- A medida que nuevas técnicas de concientización ciudadana emergen, el equipo de técnicos ambientales del MARN se actualiza sobre estas técnicas o métodos para mejorar su alcance y concientización ciudadana para los programas educativos ecológicos vigentes.
- Capacitación al personal de recurso humano del MARN sobre buenas relaciones humanas.
- Ya que el trabajo de educación ambiental involucra a pocos técnicos expertos y un gran número de ciudadanos es necesario que cuando impartan su conocimiento sean capaces de lidiar con las personas asignadas a su cargo, por lo tanto, una buena capacidad de relaciones interpersonales es necesario.
- Foros y charlas de capacitación para maestros de diversos centros escolares preseleccionados.
- Los foros y charlas consisten en la selección de un lugar focal en donde las personas a quienes está dirigido el programa se aglomeran y escuchan las



charlas de los técnicos del MARN, la mayoría de veces estas charlas están acompañadas de eventos especiales, como la aparición de las mascotas del MARN, figuras celebres como alcaldes, el presidente de la república, la primera dama, etc.

- Las charlas son programadas con anticipación y es responsabilidad del MARN hacer llegar a las instituciones seleccionadas las fechas de estos eventos.
- La razón de cada charla, es educar a los maestros y estudiantes sobre el buen uso de desechos sobre lo necesario que es trabajar juntos para crear un medioambiente sustentable.
- Distribución de instructivos y folletos educativos para el buen manejo de los recursos naturales, instructivos y material educativo impreso, para el correcto trato de desechos sólidos (como desecharlos, donde depositarlos, etc.)
- Usualmente, este procesos consiste en proveer material educativo en la materia para que los docentes se capaciten y los estudiantes aprenden sobre la ecología de su alrededor. La mayoría de veces los panfletos son distribuidos en el mismo lugar de las charlas, en otras ocasiones los técnicos del MARN llegan a las instituciones a distribuir el material, también las personas interesadas pueden llegar a las oficinas del MARN a recoger copias del material educativo.

#### **2.1.2.4. Relaciones entre los componentes**

Las interacciones entre los componentes del sistema son elementales para la realización de su objetivo, entre estas están:

##### Técnico MARN – Profesor

- Esta es una de las principales relaciones, ya que es donde el Profesor adquiere su mayor conocimiento relacionado con la educación ambiental en contacto directo con el Técnico asignado, esta relación es necesaria para adquirir las capacidades que le permitirán desenvolverse como promotor de la concientización ambiental al transmitir a los estudiantes el conocimiento adquirido.
- Esta interacción se da mediante charlas impartidas en donde el Técnico del MARN juega el papel de un proveedor o fuente de información y orientación para el profesor de primer o tercer ciclo; Luego se llevan a cabo evaluaciones en donde el Técnico cuantifica los conocimientos adquiridos por el maestro al tener



una perspectiva clara del estado cognoscitivo de este con respecto al tema.

- Es durante esta relación que también se puede proveer el material impreso que sirve de soporte para la educación impartida por el técnico.

#### Técnico MARN – Estudiante

- Esta relación es considerada especial, ya que sirve como una preparación para la educación ambiental en los estudiantes, se le da seguimiento por medio de las interacciones profesor-estudiante. Al inicio, puede ser un poco confuso para los niños, debido a que es un tema nuevo para muchos, no obstante, la atención que se le da en los eventos especiales, les crea curiosidad y motivación para aprender más al respecto.

#### Profesor –Estudiante

- Esta relación es indispensable para la capacitación del estudiante, no puede este componente existir, si el otro no está, ya que el alumno es parte de la clase y esta es parte del alumno al mismo tiempo otro componente que juega un papel importante en esta relación es el docente, que de faltar, el subsistema no existiría.
- Estos dos elementos forman un subsistema imprescindible para el objetivo del sistema en su totalidad. Ya que al dar seguimiento a la educación ambiental, es ahí en donde se comienza a concientizar al estudiante sobre su medioambiente, y en última instancia, son ellos quienes harán un verdadero cambio con respecto a la sustentabilidad de ecosistema cuyos beneficios (cuantificables) están fuera del alcance de este anteproyecto.

#### Estudiante-Clase de Discusión-Docente

- Este subsistema tiene las mismas características que la relación estudiante-clase Teórica-docente.

#### Estudiante-trabajo de campo-Docente

- Este subsistema tiene las mismas características que la relación estudiante-clase teórica-docente, al ser una tarea evaluada en la mayoría de casos, con la diferencia que los estudiantes son organizados por los maestros y directores de instituciones para formar parte de eventos especiales en donde pueden observar



directamente los problemas medioambientales o estudiar especies de animales interesantes, creando en el estudiante una memoria visual sobre su ecosistema.

#### Estudiante-Evaluaciones-Docente

- El objetivo de esta relación es de cuantificar o cualificar los conocimientos adquiridos por el estudiante en determinado momento o periodo dentro del tiempo del proceso educativo medioambiental.

#### Estudiante-Material de Apoyo

- Esta relación tiene como finalidad permitir al estudiante dar seguimiento a su educación ambiental en casa o en horarios extracurriculares para reforzar el conocimiento adquirido en clase.

#### **2.1.2.5. Salidas**

- Personal de RRHH del MARN capacitado.
- Nuevas técnicas, conocimientos para la educación y promoción de la concientización de un medioambiente sostenible.
- Docentes de los 113 centros educativos capacitados en materia de medioambiente.
- Estudiantes dentro de los veinte mil seleccionados, con educación actualizada en beneficio del medioambiente, hábitos de reciclaje y cambio de actitudes hacia el medio que les rodea.

#### **2.1.2.6. Control**

##### Leyes ambientales nacionales e internacionales.

Las leyes vigentes tanto nacionales como internacionales sirven para controlar el material educativo distribuido, seleccionar los conocimientos que se van a impartir en los procesos de concientización ambiental y actualizar cualquier información obsoleta que pudiera existir en el programa de educación ambiental del MARN.

##### Directores del Ministerio de Medio Ambiente y Recursos Naturales

Estas entidades dentro de la organización monitorean constantemente el desempeño de sus técnicos y verifican si los objetivos se están alcanzando, tienen el control del sistema en su totalidad, el funcionamiento, gestión económica, operativa y alcance recae casi exclusivamente



sobre estas personas. No obstante, el Gobierno de la Nación efectúa chequeos periódicos o hace uso de reportes para tener un control sobre sus programas ambientalistas.

#### **2.1.2.7. Frontera**

La frontera del sistema es física al igual que conceptual, la primera es debido a que solo las instituciones seleccionadas por el programa se convierten en el alcance del sistema, y la segunda porque solo los estudiantes de primer y segundo ciclo son incluidos en las campañas y eventos del MARN.

La frontera se delimita por lo siguiente:

- La temática ambientalista (solo se tocan temas a favor del medioambiente).
- Solo 113 Centros Educativos de 11 municipios de San Salvador, Santa Tecla y Antiguo Cuscatlán son parte del proyecto del MARN como sistema, con un alcance de aproximadamente veinte mil niños de primer y segundo ciclo.

#### **2.1.2.8. Entorno**

- Centros Educativos no seleccionados por el programa actual.
- Estudiantes de tercer ciclo, bachillerato y educación superior.
- ONGs Nacionales e Internacionales.
- Programas vigentes dirigidos por instituciones como FONAES, MINED, entre otras.
- Población Salvadoreña en general.
- Área Educativa (profesores que no forma parte del programa actual).



## 2.2. DESCRIPCIÓN DE PROCESOS ACTUALES.

En esta sección se detallan los procesos que se realizan actualmente en el Programa Nacional de Educación Ambiental para el Reciclaje.

### 2.2.1. Diagrama jerárquico de procesos.

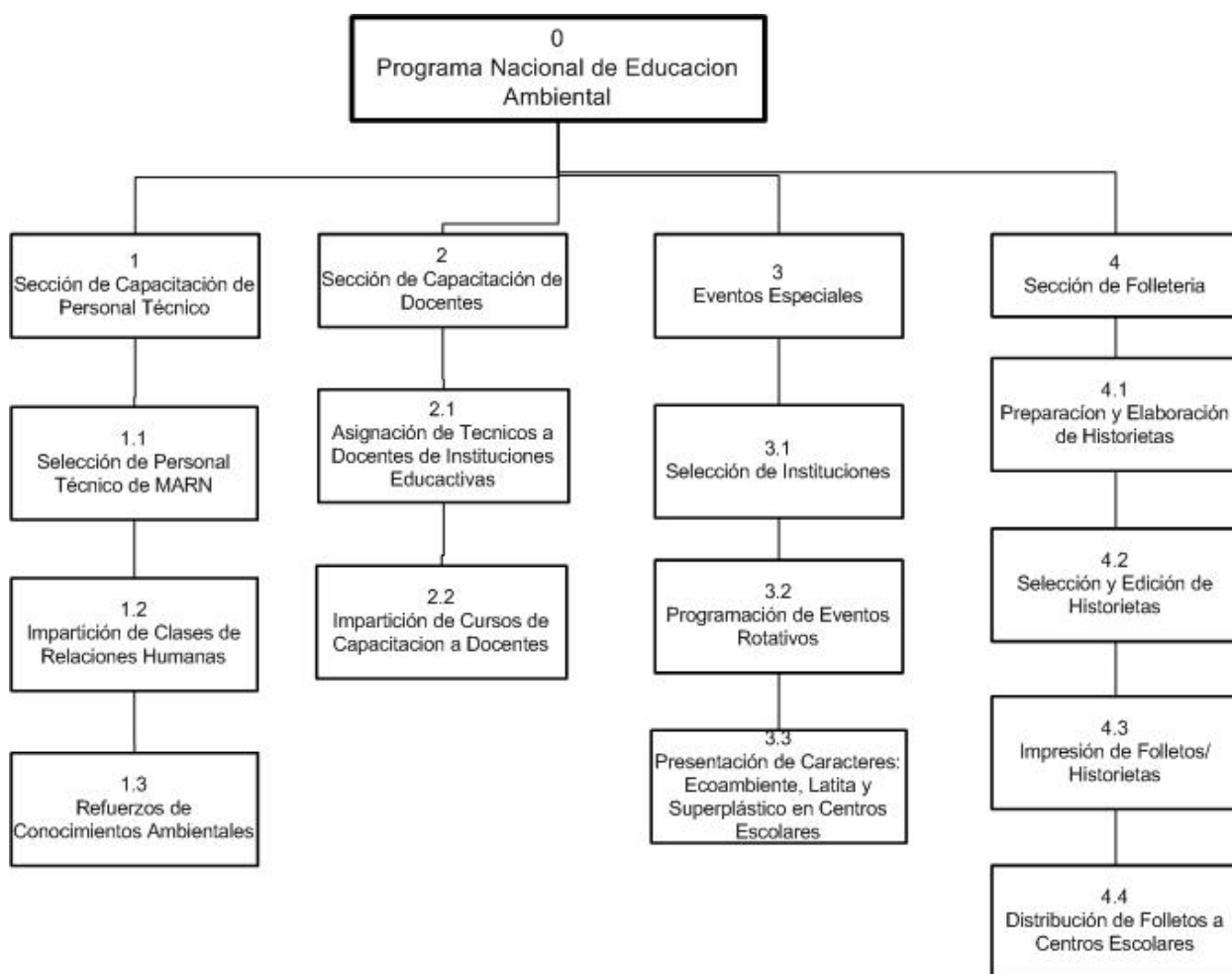


Figura 2.3: Diagrama jerárquico de procesos



## 2.2.2. Procesos

La figura 2.3 presenta la jerarquía de los procesos del programa bajo estudio, la descripción de estos se realiza a continuación.

### 2.2.2.1. Nivel 0: Programa Nacional de Educación Ambiental

El programa de Educación Ambiental<sup>72</sup> actual, está a cargo de una empresa consultora contratada por El Ministerio del Medio Ambiente y Recursos Naturales MARN, especialmente para dicho programa. Ha iniciado sus procesos de concientización ciudadana el 5 de marzo de 2008, como iniciativa del Gobierno de El Salvador en conjunto con ONGs extranjeras y nacionales, el MARN y el MINED. El Actual programa, a cargo del Gerente del Departamento de Educación Ambiental, tiene como objetivo la concientización ambiental de la población salvadoreña, alcanzado hasta el momento a aproximadamente 20,000 estudiantes de 113 centros educativos en San Salvador, Santa Tecla y Antiguo Cuscatlán del departamento de La Libertad. El Departamento de Educación Ambiental esta bajo la dirección del departamento de Participación Ciudadana del MARN a cargo del Arq. Jorge Alberto Domínguez.

El programa de Educación Ambiental Nacional tiene como objetivo a mediano plazo concientizar a los jóvenes de primero a sexto grado sobre el medioambiente que les rodea y los recursos naturales de los cuales este último está constituido, creando en los estudiantes un sentimiento de participación en la sustentabilidad del medioambiente a través de prácticas diarias de clasificación de desechos sólidos, buen trato a la flora, fauna e impartición de conocimiento sobre conservación de mantos acuíferos, reforestación y disminución de la contaminación atmosférica. El objetivo a largo plazo es concientizar a la población salvadoreña en general sobre dicha gestión. Nota se utilizará el formato **Pn.m** para diferenciar los números de procesos de los números de temas para esta sección.

#### P1. Sección de capacitación de personal técnico

A pesar de que el actual programa está a cargo de una empresa privada, el MARN provee el recurso humano técnico que estará a cargo de capacitar a los docentes de los diferentes centros educativos sobre la gestión medioambiental.

**P1.1 Selección de personal técnico de MARN:** Este proceso se lleva a cabo en determinadas fechas, dependiendo de la necesidad de capacitación de empleados del MARN, quienes, bajo

---

<sup>72</sup> Ver Participación Ciudadana <http://www.marn.gob.sv/?fath=4&categoria=88>



la dirección de la Gerencia de Educación Ambiental, son seleccionados para su participación en el programa.

**P1.2 Impartición de clases de relaciones humanas:** Este proceso se lleva a cabo seguido de la selección del personal, consiste en:

- Charlas sobre Relaciones interpersonales entre técnicos y docentes.
- Planeación de Metodología para interacciones e impartición de conocimientos a los docentes.
- Planeación de temas y conocimientos a impartir

**P1.3 Refuerzos de conocimientos ambientales:** La sección de Capacitación de Personal Técnico incluye refuerzos temáticos, este proceso consiste en:

- Evaluación personal de los conocimientos de la gestión ambiental de cada empleado seleccionado.
- Enumerar cualquier duda o inquietud que tengan los empleados seleccionados
- Enumerar actualizaciones en técnicas emergentes de la gestión del conocimiento ambientalista.
- Establecer reuniones para reforzar los conocimientos de cada técnico basados en las inquietudes del personal técnico y actualizaciones técnicas de la gestión conocimiento ambientalista.
- Reforzar conocimientos ambientales.

## P2. Sección de capacitación de docentes

Se lleva a cabo después de que la dirección de Participación ciudadana cuenta con un grupo de técnicos para la impartición de conocimientos y está conformada por los siguientes procesos:

**P2.1 Asignación de técnicos a docentes de instituciones educativas:** Se lleva a cabo periódicamente y rotativamente durante todo el ciclo de vida del programa actual de concientización ambiental, consiste en:

- Seleccionar diferentes centros escolares de entre los 113 pre-seleccionados.
- Enviar un delegado a cada centro escolar y programar visita de técnicos, determinando que docentes serán parte de las charlas de capacitación.
- Para cada centro escolar, uno o dos técnicos son asignados a uno o varios





docentes con los cuales entablaran charlas de capacitación en las fechas y horas acordadas.

**P2.2 Impartición de cursos de capacitación a docentes:** Se lleva a cabo en las fechas y horas acordadas entre la institución y el departamento de Educación Ambiental de MARN. Consiste en:

- Clases expositoras
- Impartición de conocimientos sobre la gestión ambiental del país a través de charlas.
- Distribución de folletos para docentes y material de apoyo para los estudiantes que el docente estará a cargo en clases normales, el material de apoyo consiste en historietas infantiles cuya temática principal es el medioambiente, para lectura en clase o el hogar.

### P3. Eventos especiales

Para la promoción de las charlas y la concientización ambiental, se llevan a cabo eventos especiales a diario, rotando los centros educativos durante todo el ciclo de vida de programa actual. Los eventos especiales consisten en la presentación de caracteres, charlas directas con los estudiantes, piñatas, rifas y distribución de historietas o folletería pro medioambiente. Los procesos que permiten dichos eventos son:

**P3.1 Selección de instituciones:** Consiste en seleccionar diferentes centros escolares, dependiendo de la ruta a seguir por los presentadores, de la última vez que se visitó un determinado centro escolar y de la disponibilidad de la institución en determinada fecha.

**P3.2 Programación de eventos rotativos:** Cada semana se programa la visita a diferentes centros educativos preseleccionados. Consiste en contactar al director/a de cada centro escolar, solicitando su aval para llegar a la escuela en una hora y fecha determinada, reprogramando visitas dependiendo de la disponibilidad de aulas o aval de las autoridades de los centros educativos.

**P3.3 Presentación de caracteres:** Ecoambiente, Latita y Superplástico en centros escolares se lleva a cabo en la hora y fecha programada, consiste en:

- Preparación de aulas y coordinación con Estudiantes y docentes.



- Llegada de Personajes disfrazados en un panel con los logos del Programa de Educación Ambiental, MARN y MINED
- Acto de presentación y bienvenida de personajes.
- Dinámicas y juegos interactivos entre personajes y estudiantes.
- Descanso y aperitivos
- Charla ambientalista de parte de los personajes.
- Entrega de historietas, vejigas, gorras, camisetas y figurines de personajes.
- Despedida de personajes
- Los personajes se retiran en panel especial para añadir al realismo.

#### P4. Sección de folletería

Incluye los procesos de diseño, edición, impresión y distribución de material de apoyo para docentes y estudiantes referentes a la gestión ambiental.

**P4.1 Preparación y elaboración de historietas:** Este proceso se lleva a cabo dependiendo de la demanda e inventario de material de apoyo. Consiste en el empleo de profesionales artistas y escritores para crear historias con personajes especiales que apelen a los niños para concientizarles sobre el medioambiente. Al mismo tiempo se elabora material técnico y folletos para docentes y menor medida, para los técnicos de MARN.

**P4.2 Selección y edición de historietas:** Dada una propuesta, el departamento de Educación Ambiental de MARN puede aceptar, rechazar o pedir modificaciones al material presentado. Cuando una propuesta ha sido aceptada, se elabora un estudio para determinar la cantidad de historietas o folletos que deberán ser impresos y se da seguimiento al siguiente proceso.

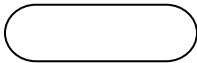
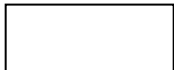
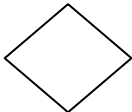





**P4.3 Impresión de folletos e historietas:** Este proceso se lleva a cabo después que un determinado material de apoyo ha sido aprobado por la dirección del Programa de Educación Ambiental y está a cargo del la empresa consultora.

**P4.4 Distribución de folletos a centros escolares:** Este proceso se lleva a cabo durante las charlas Técnico-Docente y durante los eventos especiales, consiste en almacenar en cajas el material a distribuir, las cajas son transportadas en los paneles en que llegan los personajes disfrazados (para eventos especiales) o por los técnicos del MARN (para charlas Técnico-Docente)



Los procesos anteriormente descritos conforman el Programa Nacional de Educación Ambiental actual el cual es el enfoque de este estudio. Se describen los procesos de más bajo nivel que se llevan a cabo para alcanzar los objetivos del programa actual de Educación Ambiental Nacional. Los siguientes símbolos describen los utilizados para representar dichos procesos.

### 2.2.3. Simbología utilizada en los diagramas de procesos

| Símbolo   | Descripción  |
|---|--|
|    | Indica el inicio o fin de los procesos.  |
|    | Indica la realización de un proceso.   |
|   | Representa a las diferentes decisiones que deben tomarse en el desarrollo del proceso. |
|  | Representa un documento que entra o sale del proceso.                                  |
|  | Simboliza una nota aclaración  |
|  | Representa el flujo que tienen las diferentes actividades dentro del proceso.          |
|  | Representa un medio de almacenamiento de datos   |
|  | Representa un proceso predefinido.   |

Cuadro 2.1: Notación empleada para la descripción de procesos



## 2.2.4. Diagramas de procesos

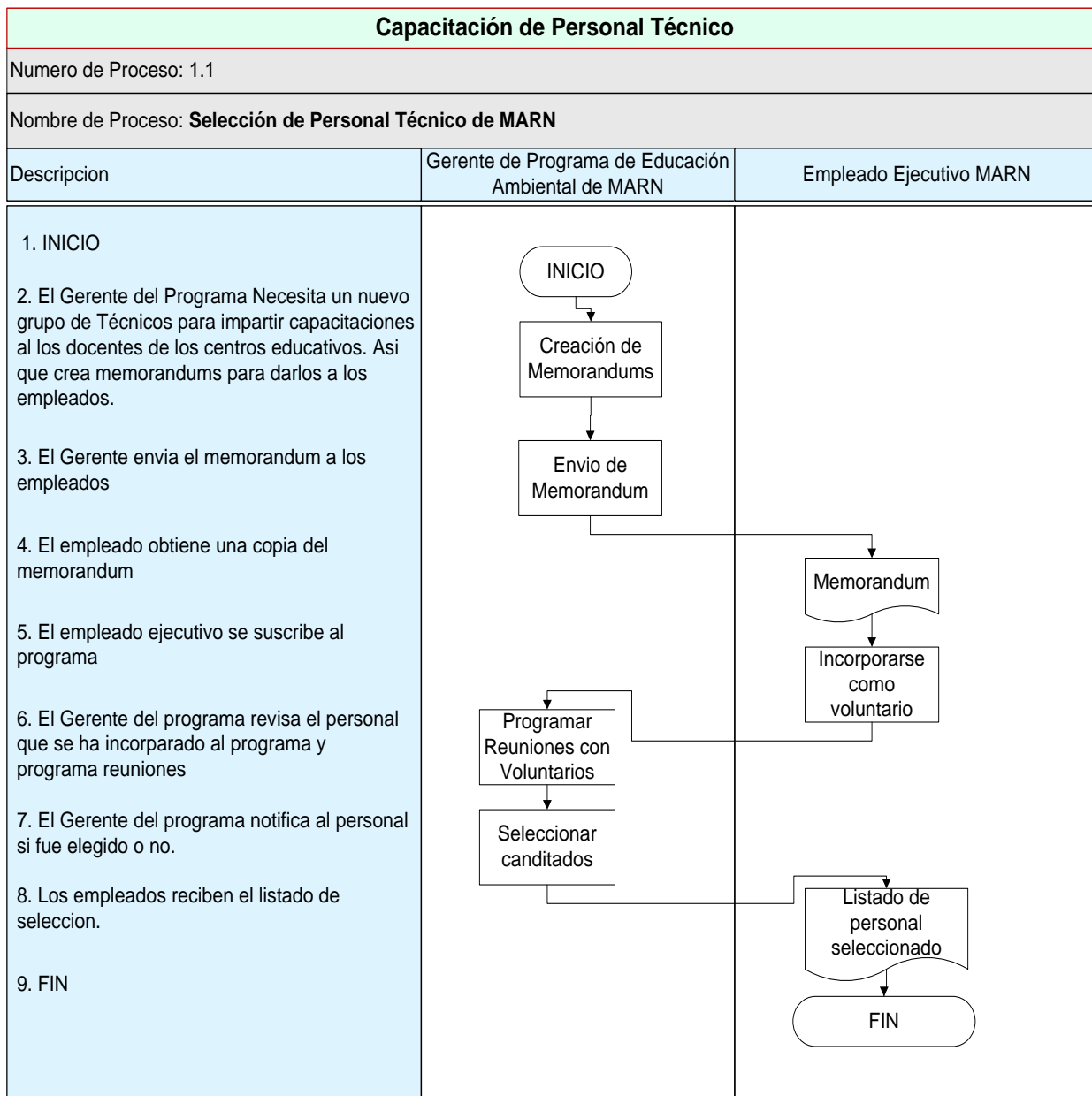


Figura 2.4: Procesos selección de personal técnico del MARN.

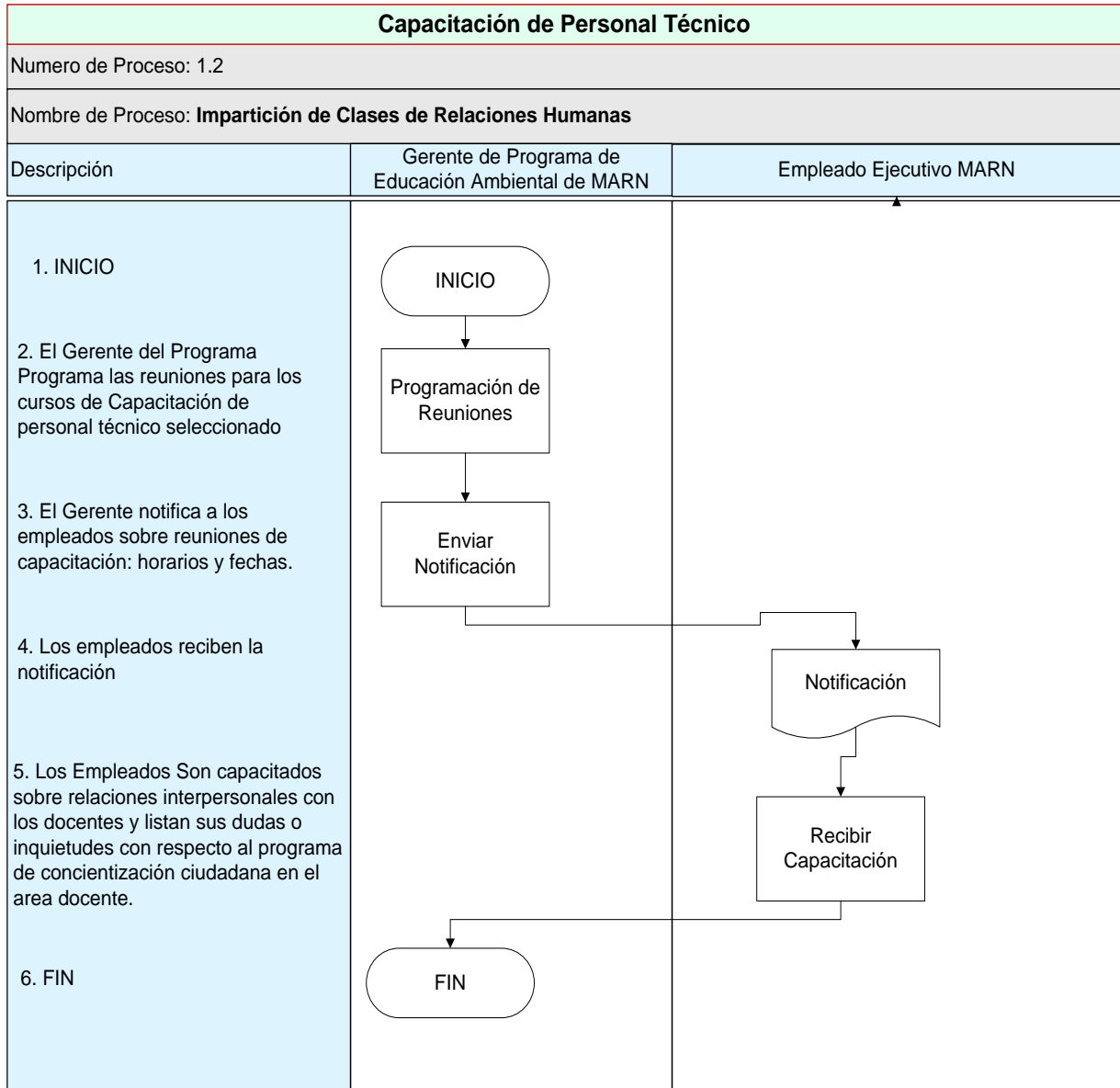


Figura 2.5: Procesos impartición de clases de relaciones humanas.

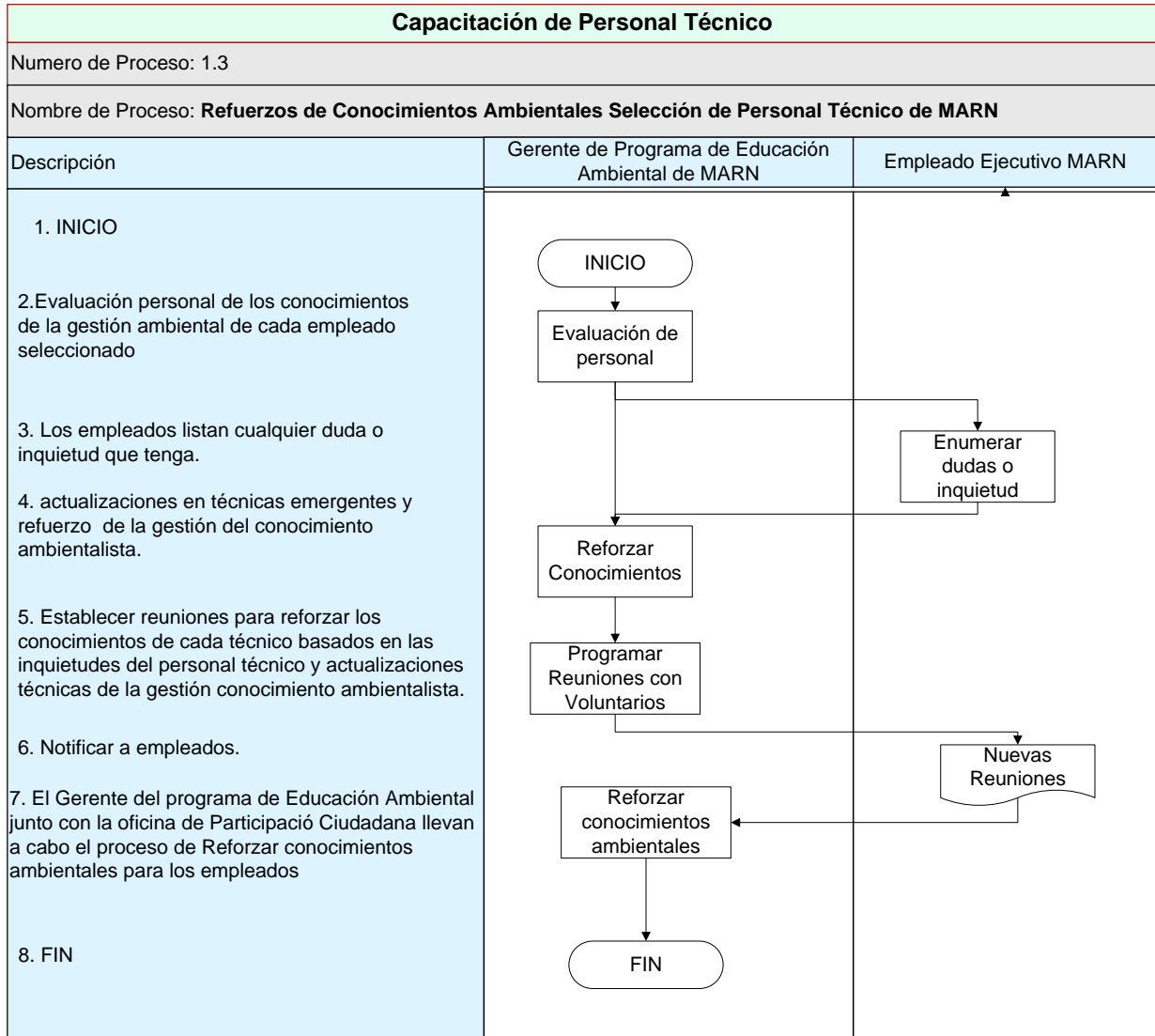


Figura 2.6: Procesos refuerzos de conocimientos ambientales.

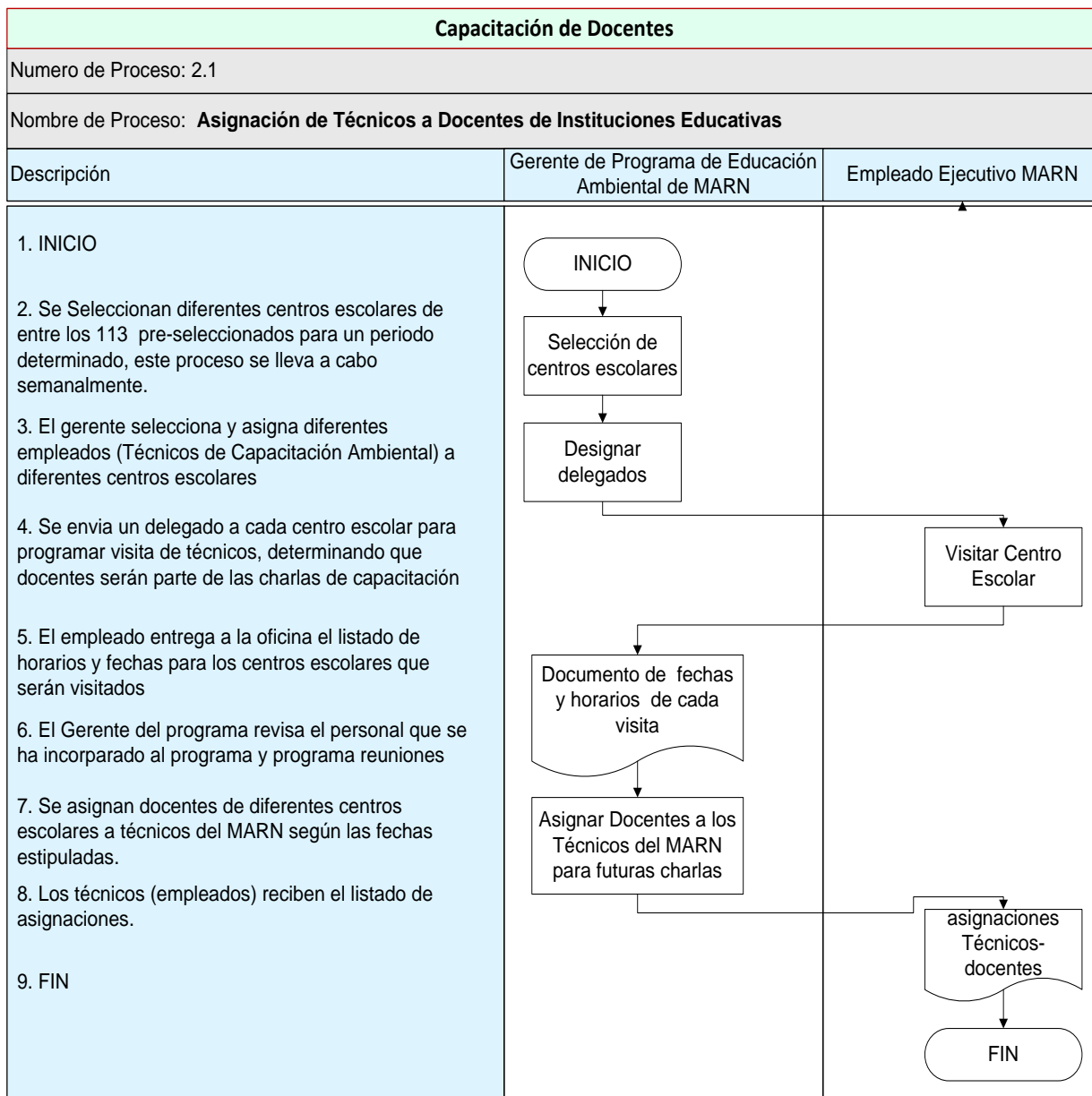


Figura 2.7: Asignación de técnicos a docentes de instituciones educativas.

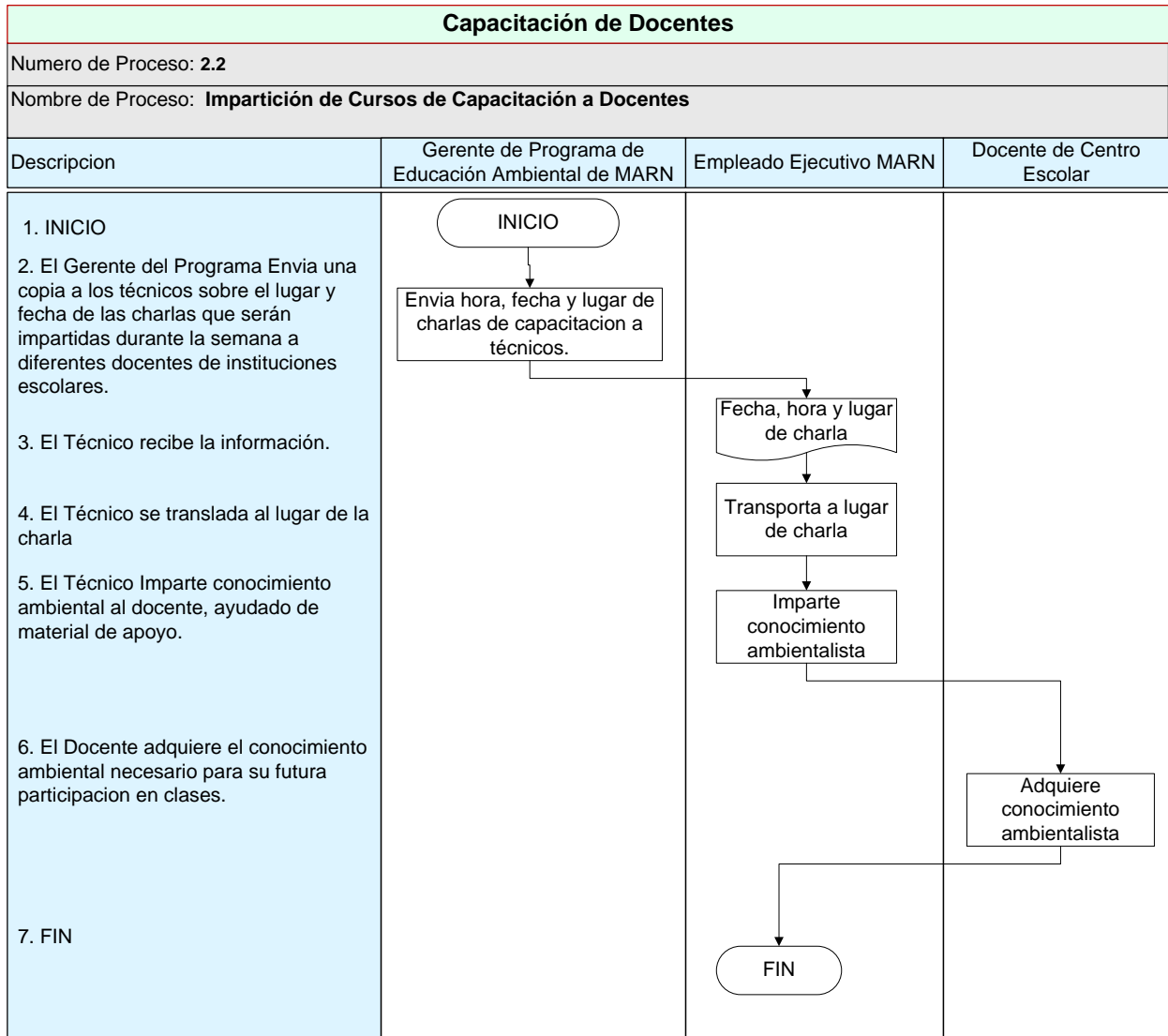


Figura 2.8: Procesos Impartición de cursos de capacitación a docentes.



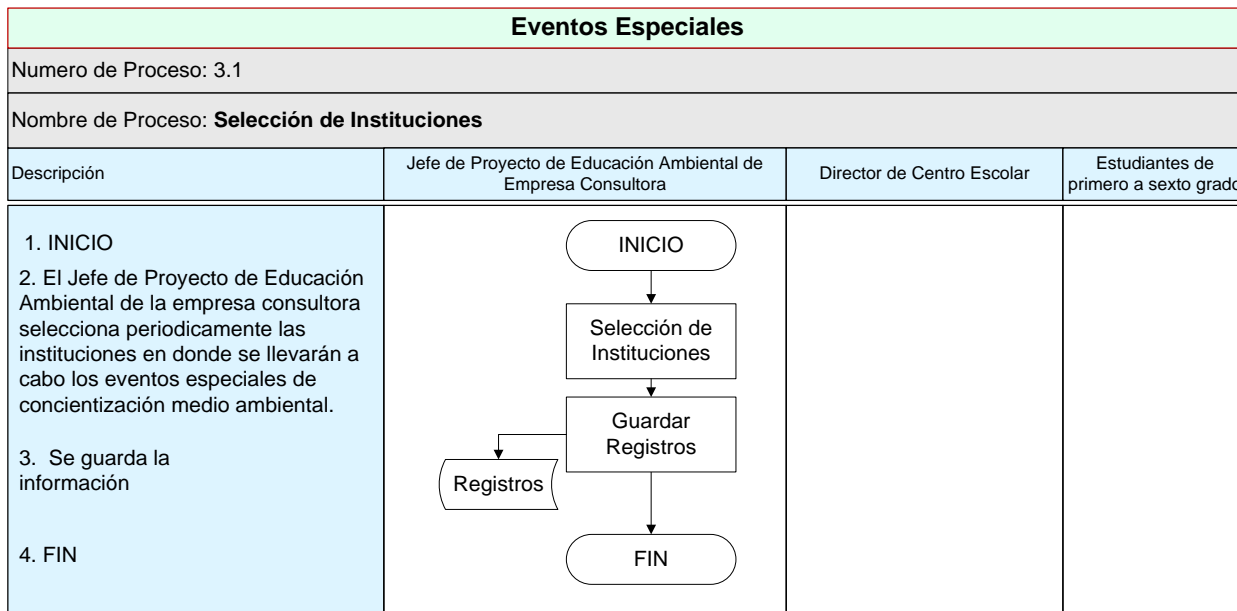


Figura 2.9: Procesos selección de instituciones.

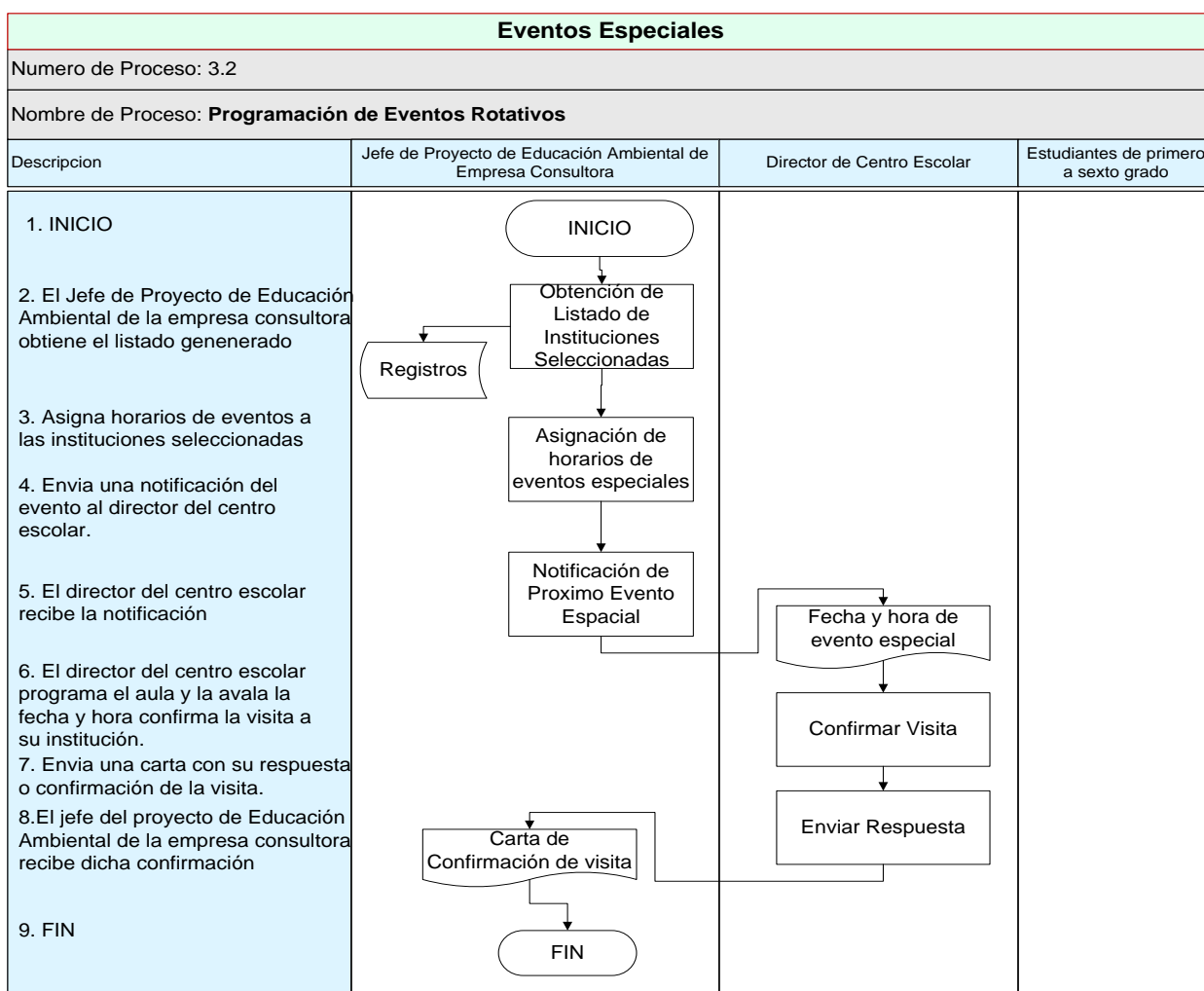


Figura 2.10: Procesos programa de eventos rotativos.

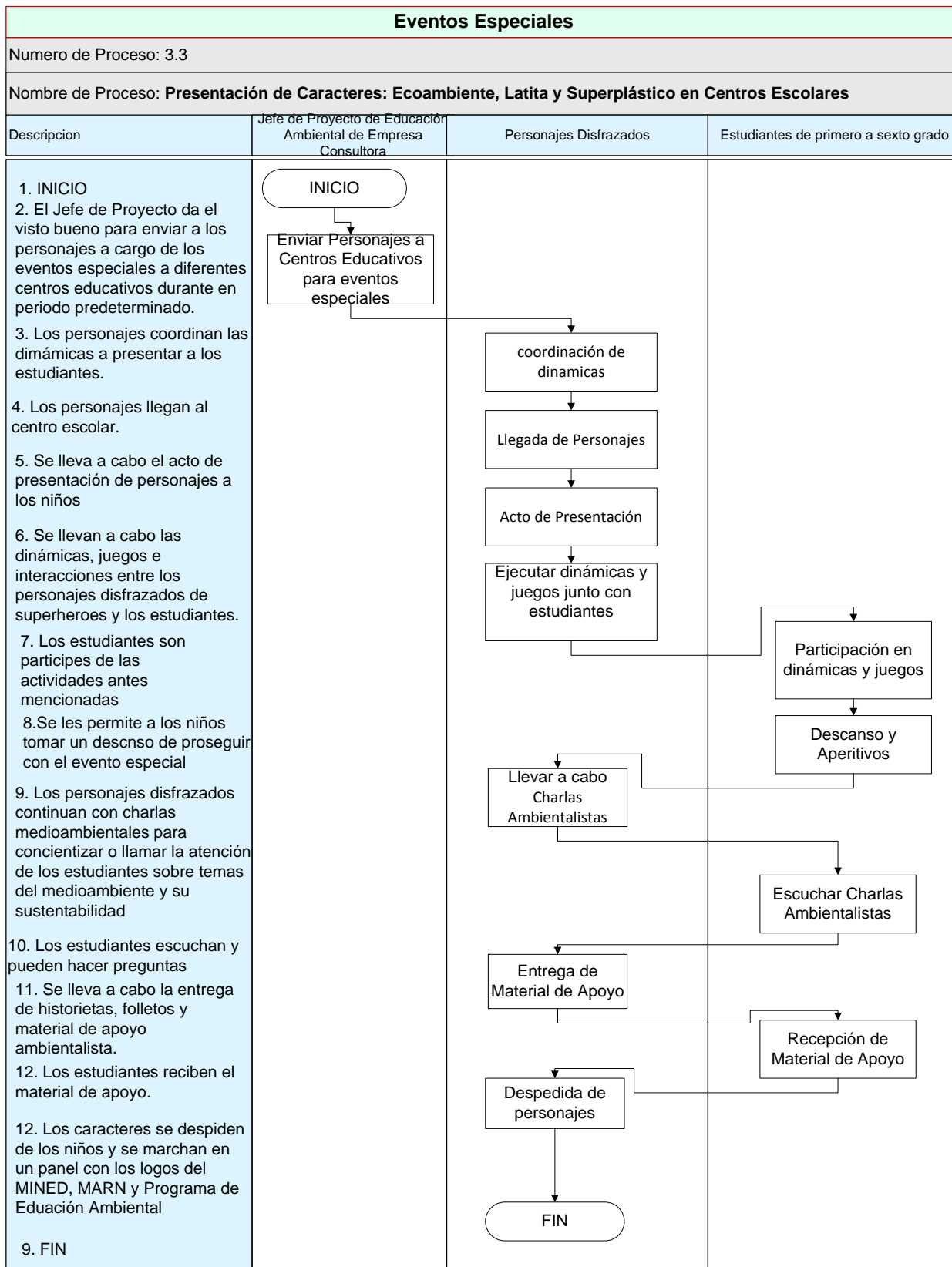


Figura 2.11: Procesos presentación de caracteres.

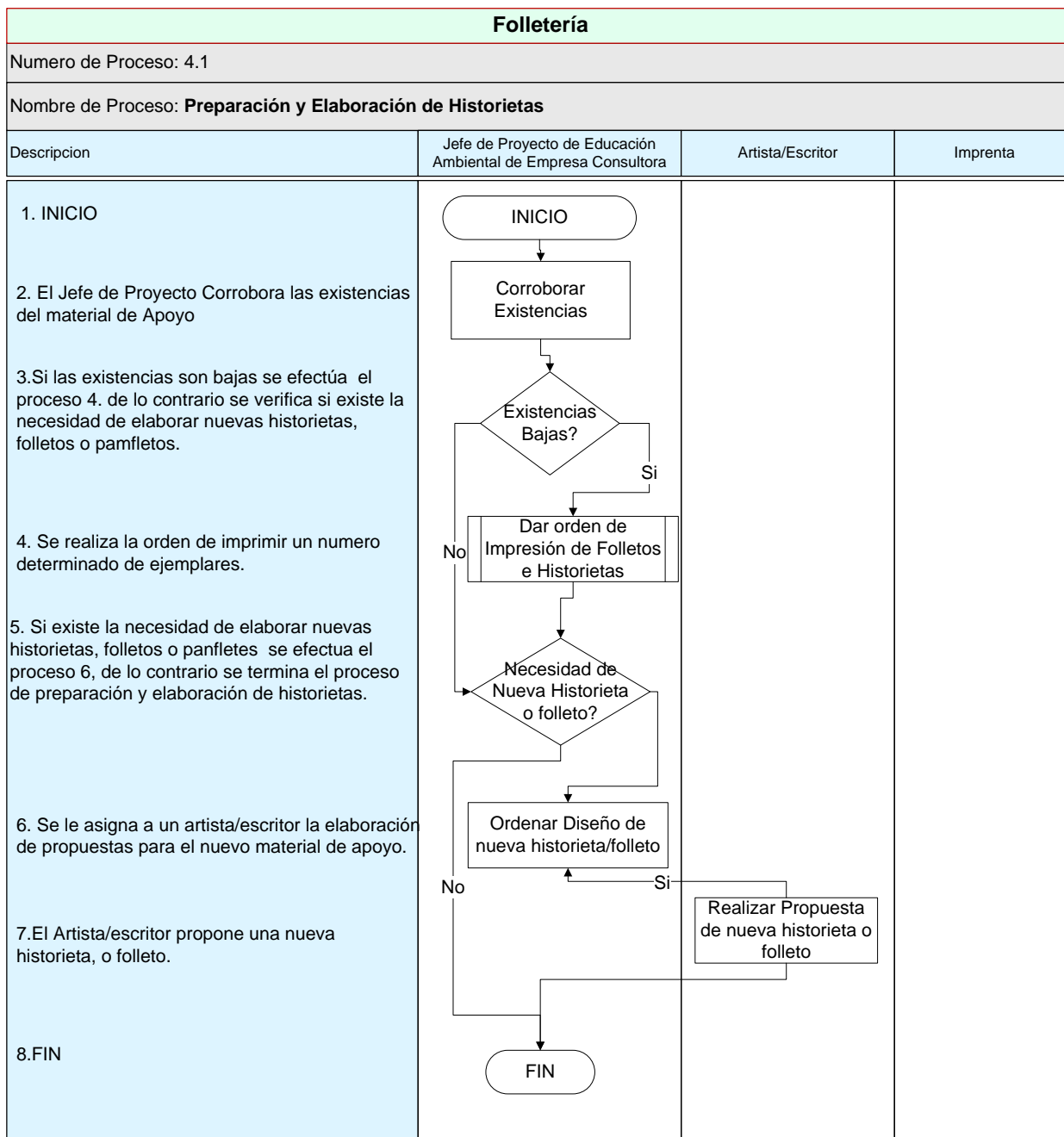


Figura 2.12: Procesos preparación y elaboración de historietas.

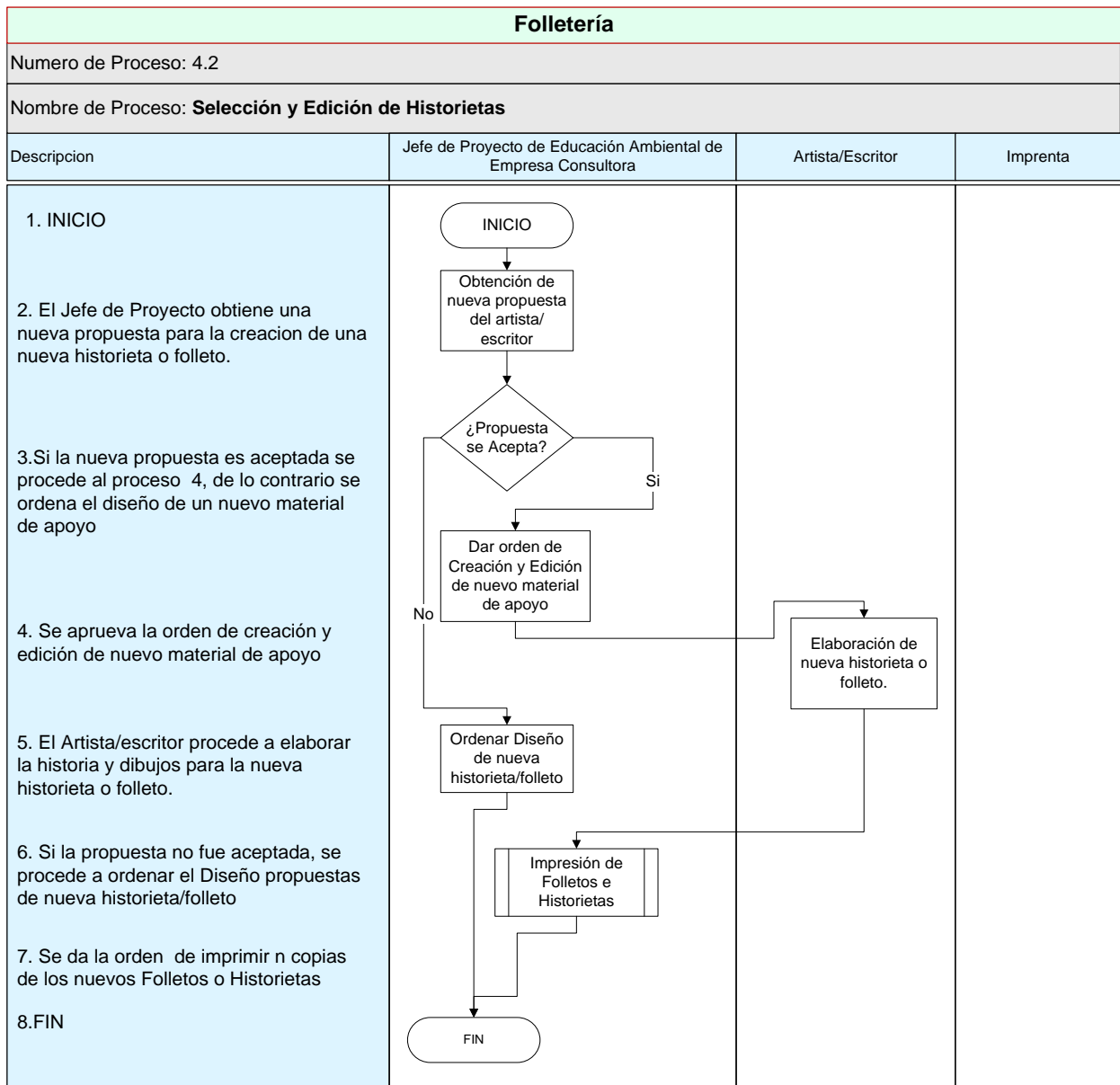


Figura 2.13: Procesos selección y edición de historietas.

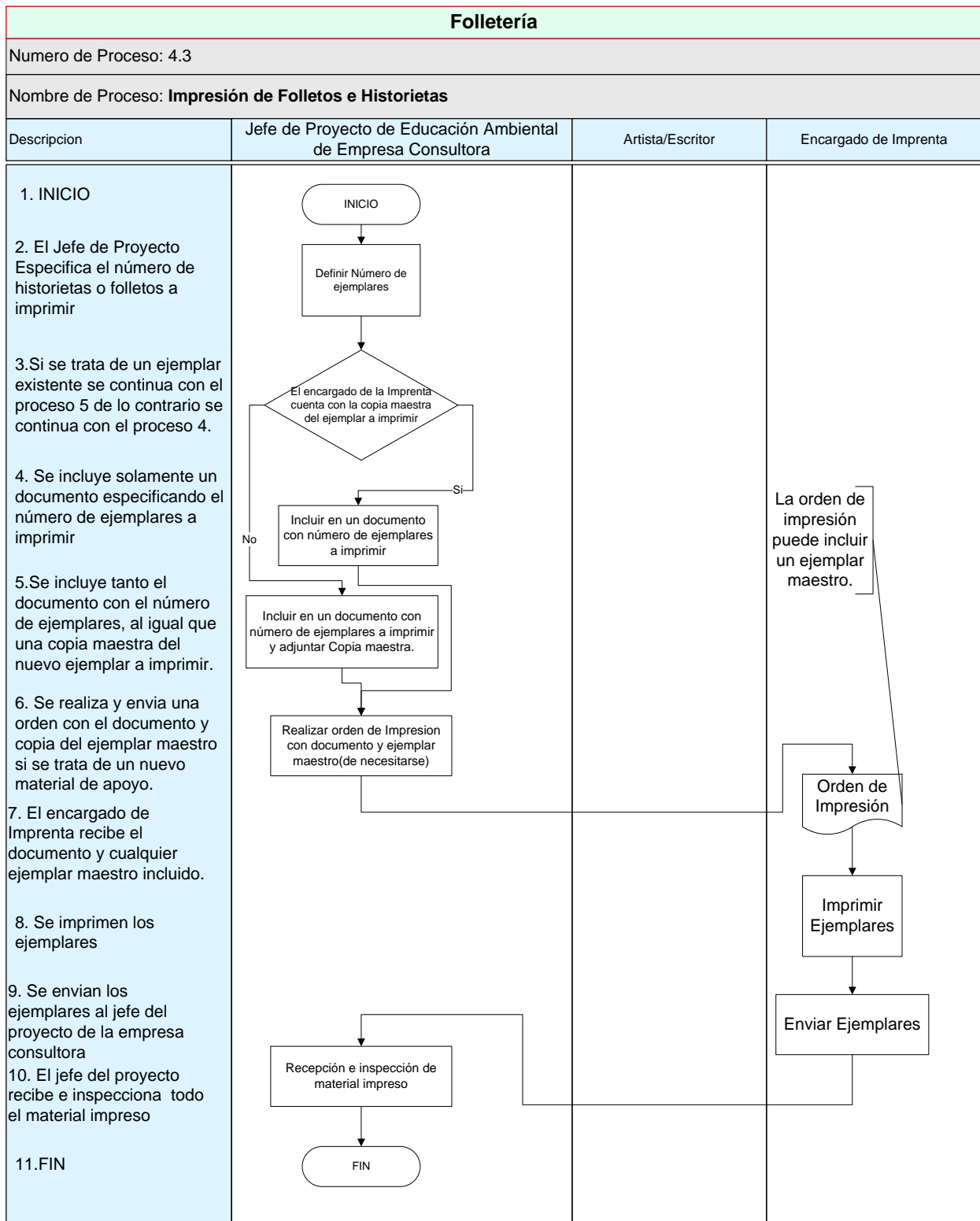


Figura 2.14: Procesos impresión de folletos e historietas.

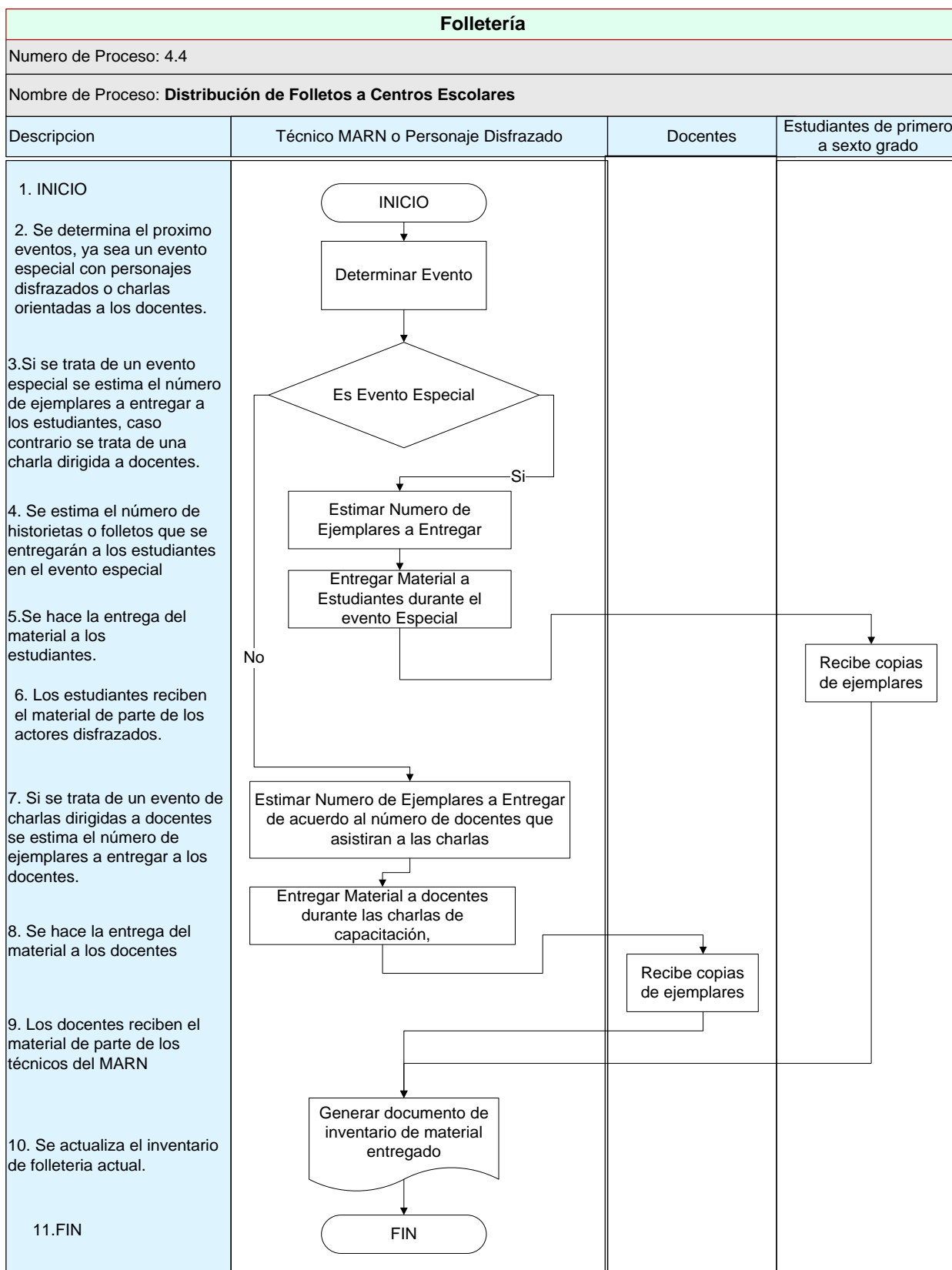


Figura 2.15: Procesos distribución de folletos a centros escolares.



## **2.3. DIAGNOSTICO.**

El actual Programa de Concientización Ambiental del MARN, tiene una serie de deficiencias las cuales no son fácilmente visibles en los diagramas de procesos anteriores y enfoque de sistemas presentado en este capítulo, estos se resumen en las áreas siguientes: Instituciones Pro-Medio Ambiente, recursos tecnológicos, métodos y técnicas, proyectos ambientales, información, normativas y leyes, estudiante y familia. Para el diagnostico del problema se ha optado por utilizar la Matriz FODA ponderada como herramienta para visualizar de manera estructurada las causas de las deficiencias del problema actual. Antes de proceder al análisis FODA, es necesario enmarcar los temas para las fortalezas, oportunidades, debilidades y amenazas; luego asignarles sus respectivos pesos de forma analítica.

### **2.3.1. Problemas o deficiencias del programa de educación ambiental actual**

El programa actual está sujeto a un presupuesto fijo de \$200,000<sup>73</sup> lo que conlleva a una delimitación en su alcance de 113 centros escolares, aproximadamente 20 mil estudiantes de primero a sexto grado. Al mismo tiempo, este presupuesto depende de la disponibilidad del gobierno de la República y/o organismos internacionales a favor de la sustentabilidad del medioambiente. Por ahora, no se puede cuantificar el beneficio o ingresos que se pueden esperar del programa, tampoco se puede garantizar que los estudiantes asimilen y/o pongan en práctica el conocimiento que se les está impartiendo. Como otra limitante el periodo de tiempo en que se lleva a cabo el actual programa (6 meses por \$200,000 aportados al proyecto). Si bien existen deficiencias también hay factores que aportan al éxito del programa, estos son:

- El factor humano, el conocimiento es transmitido de técnicos a docentes, de técnicos a estudiantes y de docentes a estudiantes.
- La calidad de información proporcionada por los técnicos de la entidad, por su experiencia en la materia.
- La disponibilidad de las instituciones de trabajar en conjunto con una organización estatal.

A continuación, se presentara de forma ordenada los temas aquí expuestos a través del análisis FODA ponderado.

---

<sup>73</sup> Según datos proporcionados por la Gerencia de la Unidad de Educación Ambiental.



### 2.3.2. Análisis FODA ponderado del programa actual de concientización ambiental del MARN.

| Fortalezas  | Debilidades  |
|---|--|
| 1. Factor Humano<br>2. Temas a impartir son fidedignos por tratarse de la máxima autoridad medioambiental del país.                           | 3. Alcance de número de estudiantes capacitados en proporción directa al presupuesto disponible.   |
| Amenazas  | Oportunidades  |
| 4. Se educa al estudiante pero no se garantiza que el niño/adolescente va a acatar las normas sugeridas por las autoridades de medioambiente. | 5. Presupuesto asignado depende en gran porcentaje a entes internacionales.<br>6. Periodos de tiempo de capacitación no continuo, limitado al presupuesto recurso humano |

Cuadro 2.2: Matriz FODA y Matriz FODA Ponderada<sup>74</sup>

| Matriz de evaluación de factores internos |   |              |      |             |
|---|---|--------------|------|-------------|
|   |   | Calificación | Peso | Ponderación |
| <b>Fortalezas</b>                         | 1. Factor Humano  | 2            | 0.3  | 0.6         |
|   | 2. Temas a impartir son fidedignos por tratarse de la máxima autoridad medioambiental del país. | 2            | 0.1  | 0.2         |
| <b>Debilidades</b>                        | 5. Presupuesto asignado depende en gran porcentaje a entes internacionales.                     | 1            | 0.1  | 0.1         |
|   | Total   | 5            | 0.5  | 0.9 ¥       |

Cuadro 2.3: Evaluación de factores internos.

<sup>74</sup> Pesos y calificación asignados a partir de observación directa, sentido común y analítico, datos obtenidos por entrevistas e investigaciones bibliográficas y de campo, un estudio más a fondo sería necesario para la exactitud de pesos y calificaciones, este análisis FODA de pesos ponderados es de naturaleza preliminar y solo sirve para tener una estimación de la situación actual del programa de capacitación medioambiental que se está llevando a cabo a partir del 5 de marzo, de 2008, ver sección de factibilidad económica para una descripción detallada de dicho programa





| Matriz de evaluación de factores externos |   |              |      |             |
|---|---|--------------|------|-------------|
|   |   | Calificación | Peso | Ponderación |
| Oportunidades                             | 3. Alcance Limitado de número de estudiantes capacitados en proporción directa al presupuesto disponible.         | 3            | 0.3  | 0.9         |
|   | 4. No se garantiza que el niño/adolescente va a acatar las normas sugeridas por las autoridades de medioambiente. | 1            | 0.1  | 0.1         |
| Amenazas                                  | 6. Periodos de tiempo de capacitación no continuo, limitado al presupuesto recurso humano y factores externos.    | 1            | 0.1  | 0.1         |
|   | Total   | 5            | 0.5  | 1.1 §       |

Cuadro 2.4: Evaluación de factores externos.

### 2.3.3. Objetivos y pasos para analizar la Matriz FODA Ponderada:

El análisis aquí presentado esta basado en investigaciones previas, comentarios de autores y bibliografía de consulta<sup>75</sup>.

Sobre el análisis FODA, Steiner<sup>76</sup> plantea lo siguiente: “El análisis FODA es un paso crítico en el proceso de planeación. Examinar correctamente oportunidades y peligros (amenazas) futuros de una empresa, y relacionarlo en un estudio imparcial con las potencialidades (fortalezas) y debilidades de la misma representa una enorme ventaja”.

Los Pasos a seguir para el Análisis FODA ponderado son los siguientes:

- Listar las Fortalezas, Oportunidades, Debilidades y Amenazas, de la entidad o situación que es el enfoque de estudio.
- Limitar el análisis FODA a una identificación de Fortalezas y Debilidades (Factores Internos), y de Oportunidades y Amenazas en el entorno (Factores Externos), es decir, tener un “listado”, sin preparar y utilizar la Matriz FODA que es el instrumento para la preparación de estrategias.
- Asignar una calificación total de 5 para los factores internos y para los factores externos.

<sup>75</sup> Balbino Sebastián, Manual para Formulación Evaluación y Ejecución de Proyectos, 4ª. Edición, Grafic S.A. de C.V. Pág.60

<sup>76</sup> George A. Steiner, Professor Emeritus, Profesor e Investigador de la Escuela de Administración de Empresas de la Universidad UCLA. Reconocido a nivel mundial y asesor de políticas administrativas y estratégicas de la oficina del gobierno de Estados Unidos, email: george.steiner@anderson.ucla.edu



- Asignar calificaciones individuales a cada factor interno y a cada factor externo basado en el grado en que este se considera afecta a la situación en estudio. La suma de calificaciones no debe exceder 5 para los factores internos y externos.
- Asignar los pesos a cada factor de acuerdo al conocimiento adquirido por investigaciones previas, cada peso representa en qué medida influencia al estado de la situación en su totalidad. La sumatoria de todos los pesos no debe exceder 1.0.
- Multiplicar cada calificación por su respectivo peso y calcular la ponderación para cada factor y comparar el resultado de ponderación de Internos y Externos con la media. La media es calculada dividiendo el total de calificación entre cuatro, para fortalezas, oportunidades, debilidades y amenazas.

En este caso los valores obtenidos son los siguientes:

- Media  $10/4 = 2.5$
- Factores Internos: 0.9
- Factores Externos: 1.1

Por lo tanto:

- ¥ Se necesitan fortalecer las debilidades para mejorar los factores Internos, ya que resultado de evaluación es de 0.9, por debajo de la media.
- § El peso ponderado de 1.1 resulta ser menor que la media, se necesita por lo tanto enfocar la atención del programa actual de capacitación medioambientalista hacia los factores externos, ya que está por debajo del promedio que debería de ser.

Si bien, en el análisis de arriba, el factor humano tiene un gran peso y calificación sobre el programa actual, no se supera a la media de 2.5 en la evaluación de factores internos (fortalezas-debilidades), por lo tanto, haciendo referencia a la debilidad de un “Alcance Limitado de número de estudiantes capacitados en proporción directa al presupuesto disponible”, es necesario incrementar este alcance, no obstante, depende directamente del presupuesto asignado y del periodo de tiempo estipulado dentro de los factores externos, ya que tanto el presupuesto, como autoridades a cargo del programa han definido este periodo, debido también factores económicos , periodo escolar y a estudios que han llevado a esa fijación de tiempo.



El actual Programa de Concientización Ambiental del MARN, tiene una serie de deficiencias las cuales no son fácilmente visibles en los diagramas de procesos anteriores y enfoque de sistemas presentado en este capítulo, estos se resumen en las áreas siguientes: Instituciones Pro-Medio Ambiente, recursos tecnológicos, métodos y técnicas, proyectos ambientales, información, normativas y leyes, estudiante y familia.

Para el diagnóstico del problema se ha optado por utilizar la Matriz FODA ponderada como herramienta para visualizar de manera estructurada las causas de las deficiencias del problema actual.

Antes de proceder al análisis FODA, es necesario enmarcar los temas para las fortalezas, oportunidades, debilidades y amenazas; luego asignarles sus respectivos pesos de forma analítica.

## 2.4. PROPUESTA DE SOLUCIÓN.

Dado que tanto los factores internos como externos expuestos en el análisis anterior son claramente la causa directa de la ineficiencia o problema del programa actual, es necesario implementar una solución que solviente dicha problemática. Entre estos factores se pueden mencionar:

- A.** Se educa al estudiante pero no se garantiza que el niño/adolescente va a acatar las normas sugeridas por las autoridades de medioambiente.
- B.** Alcance limitado de número de estudiantes capacitados en proporción directa al presupuesto disponible.

Para solventar el problema **A.** es necesario implementar una herramienta que garantice o aumente la probabilidad que un niño o adolescente acate las normas sugeridas por las autoridades del medioambiente, a través de un método que se acople a los jóvenes de forma natural, es decir que el niño o adolescente tienda a utilizar la herramienta por sí solo y no bajo la presión de adultos.

Para solventar el problema **B.** es necesario implementar una herramienta que minimice los costos asociados de tal forma que el alcance del programa de educación ambiental y por ende la concientización ambiental de niños y jóvenes aumente considerablemente.

Ya expuesta la necesidad, se propone el uso de **un juego de video de séptima generación** que funcione como herramienta diseminadora de conocimiento ambiental, que sea altamente portable y fácil de proporcionar a los centros educativos nacionales, privados y a cualquier persona interesada en ello.



### 2.4.1. Justificación de la propuesta de solución

Debido a que el 83% de estudiantes entrevistados respondieron “Si” a la pregunta sobre si le gustan los videojuegos y dado que 67.4% respondió que utilizan videojuegos a diario o frecuentemente, se toma como parámetro para desarrollar la herramienta propuesta (videojuego). Entre las características de los videojuegos se tienen<sup>77</sup>:

- Efectos de Sonido
- Efectos Gráficos
- Pantallas de Fondo
- Configuraciones durante el juego
- Rapidez de juego
- Rapidez de Avance en los niveles
- Utilización del sentido de humor
- Opciones de control
- Dinámica de juego
- Características de ganar o perder
- Desarrollo de caracteres
- Entre otras.

A continuación se describirá la estructura básica de un videojuego para luego describir la solución propuesta, la cual ha sido avalada por las respectivas autoridades del Ministerio del Medio Ambiente y Recursos Naturales de El Salvador.

### 2.4.2. Creación de un videojuego

Un videojuego consta de diferentes elementos esenciales para considerarse como tal, entre estos elementos se pueden mencionar la historia, personajes y niveles que le conforman. La creación de cada elemento son procesos que no difieren mucho en la mayoría de casos de otros procesos de creación como el de una película o el relato de una novela complementados con el proceso de desarrollo de software. Una vez que se haya realizado el planteamiento del videojuego, es necesario proceder al análisis de este, al diseño, desarrollo y a su implementación de manera formal.

---

<sup>77</sup> Tomado del libro CyberPsychology & Behavior, The Structural Characteristics of Video Games: A Psycho-Structural Analysis (CiberPsicología y Comportamiento, Las Características Estructurales de los Videojuegos: Un Analisis Psico-Estructural) de Richard T.A. WOOD Ph.D. de la Division de Psicología de la Universidad de Nottingham Trend, UK. Una cita electrónica de este libro puede ser visto en línea en la siguiente dirección: <http://www.liebertonline.com/doi/abs/10.1089/109493104322820057>



### **2.4.2.1. La historia**

Lo primero que se necesita hacer antes de crear un videojuego es escribir la historia que se desea contar. Puede darse el caso que el videojuego que se desee desarrollar no necesite una historia que lo envuelva, en tal situación, no es necesario llevar a cabo este aspecto.

En cuanto al desarrollo de la historia no existe una metodología que permita crear historias claramente. Hay varios aspectos sobre la manera de escribir, hablar, mostrar cosas que provoquen en el usuario tensión, felicidad, curiosidad, relajamiento, etc. Es ese aspecto emocional que la presente propuesta debe de aprovechar para llegar a los jóvenes y concientizarles con respecto a su medioambiente.

La historia debe de ser progresiva, con un sentido de anticipación para lo que puede suceder después, debe de hacer un uso inteligente de sonidos y música que incite en el usuario el estado emocional necesario para abrir y mantener el contexto de la escena que se esté mostrando.

Una historia debe de tener las siguientes partes bien definidas:

- **Situación:** Es la introducción de la historia donde se sitúa al jugador en la época y le presenta la historia en la que se va a envolver.
- **Desarrollo:** Es la trama de la historia que debe ir avanzando paralelamente a los niveles que vaya completando el jugador.
- **Desenlace:** El jugador espera que después de haber dedicado parte de su tiempo a completar el videojuego el final de éste le proporcione un refuerzo positivo para aventurarse en un nuevo videojuego.

La calidad de la historia va a depender en gran manera de la imaginación del creador y del grado en que esta sea apelante al jugador.

### **2.4.2.2. Los personajes**

Una vez desarrollada la historia del juego será mucho más fácil plantear los personajes, este debe cumplir con los requisitos que lo hagan especial para el desarrollo del juego. Esto añade al valor agregado de la historia y cumple mejor con la apelación del jugador.

Entre los personajes se pueden distinguir a cuatro tipos: 1) el personaje principal, 2) los adversarios, 3) los personajes que enriquecen la historia con sus interacciones y 4) objetos o ítems. El personaje principal debe de poder distinguirse claramente de los demás y tener una animación con mayor detalle. Los adversarios deben de haberlos de varios tipos. La dificultad con que son eliminados debe de ser fácilmente reconocible según el aspecto que tenga.



Los objetos deben de ser fácilmente identificables y poseer alguna animación que los haga distinguibles del fondo del juego, se tiene que decidir que aporta el obtenerlo.

La creación y edición de niveles está íntimamente ligado al tipo de videojuego que se desea implementar, ya sea un juego 2D o 3D, un juego de aventura, de disparar, si se trata de desempeñar un papel específico o no, es necesario diseñar los niveles acordes a la historia y avance de la misma en el videojuego. Existen muchas técnicas para diseñar niveles, si se trata de un juego en dos dimensiones se implementa la técnica más común de división de la pantalla en mosaicos para la interacción con el entorno. De tratarse de un juego en tres dimensiones, se implementan diferentes técnicas como la detección de colisiones con los objetos, caracteres y fondo 3D.

### **2.4.2.3. La lógica del juego**

Una vez que se tengan la historia, caracteres, recursos y niveles se presenta en pantalla la colección de estos elementos del videojuego. La lógica que sigue el juego se basa en un bucle, conocido por los desarrolladores de este tipo de software como *game loop* o bucle del juego, que se va repitiendo a lo largo de la ejecución del programa hasta que el jugador decida salir de la aplicación. En ese *game loop* se realiza una actualización lógica de las posiciones y estados de los personajes y demás elementos para luego actualizar la pantalla con esa nueva información. El siguiente diagrama muestra la estructura básica de los juegos de video de séptima generación, en donde toda la estructura está contenida dentro de un inicio y un fin de la aplicación.

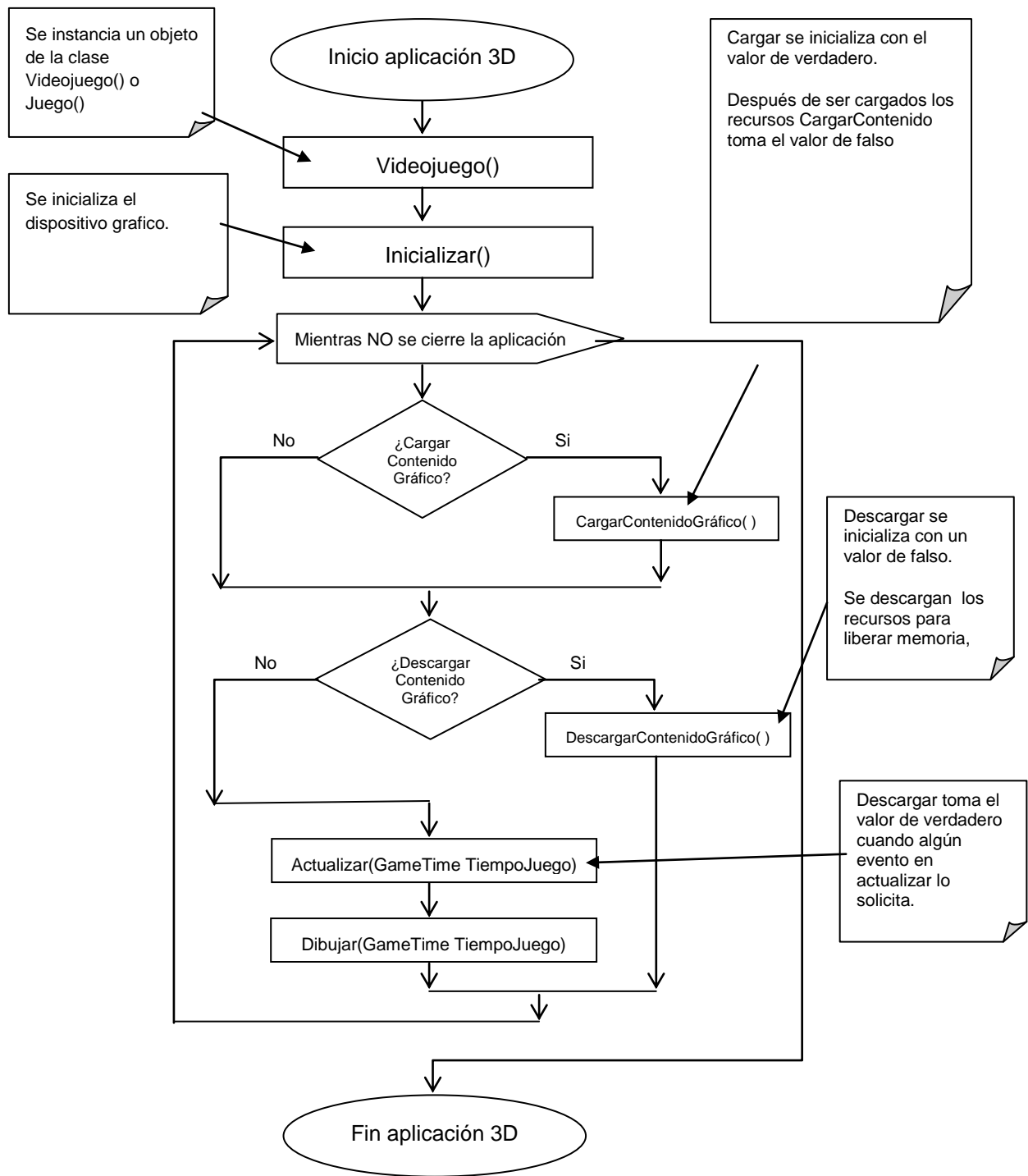


Figura 2.16: Estructura básica de una aplicación de juego en tres dimensiones.



Como se observa en la figura 2.16 al iniciar la aplicación 3D, se instancia un objeto de la clase Videojuego(), posteriormente se inicializa el dispositivo grafico para poner en modo 3D el despliegue de pantalla. Al entrar al ciclo se cargan al menos una vez los recursos necesarios: modelos, sonidos, efectos especiales, iluminación, texturas, etc.

Durante la actualización lógica Actualizar (Gametime TiempoJuego) se comprueban las capturas de objetos, se gestionan las colisiones, se comprueba la temporización de la aplicación. Después de esta actualización lógica se deben de mostrar los cambios a través de la interfaz de pantalla Dibujar (Gametime TiempoJuego). Este ciclo se repite hasta que descargar toma el valor de verdadero para liberar memoria y descargar los recursos del dispositivo.

#### **2.4.2.4. El control del tiempo**

El control del tiempo es una de las partes fundamentales que tenemos que tener en cuenta a la hora de desarrollar un videojuego. Este es uno de los aspectos que muchas veces se descuida cuando se diseñan videojuegos. Dependiendo de la escena o fluidez de las acciones de los caracteres u otros objetos, es necesario implementar contadores que estén relacionados con la velocidad con que se dibuja cada cuadro en pantalla. Si no se implementa una variable temporizadora que sea función del dibujado en pantalla, se pueden tener secuencias muy rápidas en equipos de altas prestaciones o secuencias muy lentas en equipos de bajas prestaciones.

#### **2.4.2.5. Detección de colisiones**

En un espacio virtual generado por computadoras no existen límites, excepto la memoria de video o RAM de la tarjeta madre. No obstante, con trucos de programación recursiva se puede dar la impresión de mundos infinitos o relativamente inmensos. Cualquier movimiento en este mundo virtual está sujeto a transformadas matriciales de traslación, escalación y/o rotación. No existe un método (en la actualidad) para detectar a un objeto 3D y reaccionar a su posición espacial de forma automática. No obstante existen métodos para detectar dicha interacción programáticamente, utilizando funciones trigonométricas relacionadas con la dirección, ángulos, vectores normales y cajas invisibles que engloben a determinado objeto para detectar si se está cerca o si se comparten las coordenadas de posición de este y del carácter principal, al haber detectado la posición se hace uso de otro algoritmo que permita reaccionar a la presencia de determinado objeto, ya sea deteniendo el avance, rebotando o cambiando la dirección. A estas técnicas o métodos de detección y





reacción de colisiones se le conoce como *Detección de Colisiones*.

Todas las características funcionales de un videojuego antes mencionadas pueden entenderse durante la etapa de diseño y desarrollo de la solución propuesta, implementado un enfoque orientado a objetos. Existen herramientas disponibles para diseñadores y programadores en las cuales, se puede directamente generar el código necesario para clases con sus atributos y métodos a partir de los diagramas de clases propios de modelos orientados a objetos.

### 2.4.3. ¿Por qué Programación Orientada a Objetos?

La programación orientada a objetos integra las estructuras de datos y las funciones o métodos asociadas a ellas. Las funcionalidades se traducen en colaboraciones entre objetos que se realizan dinámicamente y las estructuras del programa no se ven amenazadas por un cambio en el mismo. Uno de los principales motivos de utilizar programación orientada a objeto (POO)s es la adecuación de este modelo a la programación de videojuegos. Se puede distinguir perfectamente los objetos que componen el videojuego y las funciones o capacidades asociadas a ellos los que nos permite que este enfoque sea perfecto para crear una abstracción mediante clases. Las características fundamentales por las que se aplica la POO en este proyecto se detallan a continuación.

#### 2.4.3.1. Características de la Orientación a Objetos

Entre las características de la orientación a objetos se tienen:

- **Abstracción:** Las clases son la abstracción de un conjunto de objetos del mismo tipo.
- **Encapsulamiento:** Permite proteger, englobar a los datos y la funcionalidad de dichos datos en una estructura que permite tenerlos asociados.
- **Ocultación de datos:** Esta estructura permite tener varios tipos de datos. Entre ellos están los públicos y los privados ocultos al usuario.
- **Generalización:** Una clase puede abstraerse en una clase más general y viceversa. Esto nos permite tener varios tipos de clases hijos de una clase padre según un discriminante con un comportamiento común o individual.
- **Polimorfismo:** Permite funciones con el mismo nombre que se diferencien, por ejemplo, en el tipo de parámetros, que tengan comportamientos totalmente diferentes.



- **Clases y objetos:** Un objeto es una instancia de una clase o lo que es lo mismo, es una clase inicializada. El objeto está próximo al mundo real mientras que la clase es una abstracción de éste.
- **Herramientas:** Son cada una de las funcionalidades que aporta una clase para trabajar sobre tipos de datos como pueden ser sus atributos.
- **Atributos:** Los atributos de un objeto son aquellas variables que definen dicho objeto. Las funciones de la clase deben de permitir trabajar con ellos para realizar las modificaciones pertinentes o simplemente para ser consultados ya que son la parte fundamental de dicho objeto.

Todas estas propiedades y características de la programación orientada a objetos la hacen ideal para la programación de videojuegos. Este tipo de programación sobre un modelo de ciclo de vida en cascada permite realizar un desarrollo incremental. UML es todo lo que se necesita para enfrentar la construcción de un software de calidad.

#### **2.4.4. Historia del Videojuego de Solución Propuesta**

A continuación se presenta la historia aprobada por las autoridades del MARN en la cual se involucrará al jugador.



## Historia

*Año: 3008*






### Caracteres que componen la trama de la historia

Los diferentes caracteres que envuelven la historia del videojuego se clasifican según el rol que desempeñan en la trama, así como la interacción que el jugador tiene con estos:

| PERSONAJE                         | ROL EN EL JUEGO   |
|-----------------------------------|---|
| <b>Caracteres principales</b>     | Héroes en la trama del videojuego, su misión principal es preservar la vida del planeta Tierra.   |
| <b>Especies en extinción</b>      | Son animales con los cuales el jugador puede interactuar ya sea por medio de su manipulación o medio de interacción de diálogos.  |
| <b>Enemigos medio ambientales</b> | Seres que atacan a los héroes y personas que con su conducta amenazan la supervivencia de especies animales.  |
| <b>Guías</b>                      | Personas que proporcionan información sobre las misiones u obstáculos que se deben de emprender en los diferentes niveles.  |
| <b>Peatones</b>                   | En su visita al planeta Tierra los caracteres principales se encuentran con personas en las calles de San Salvador, con las que puede interactuar por medio de inteligencia artificial. |
| <b>Vendedores</b>                 | Comerciantes que brindan productos a los súper héroes para su consumo.  |

Cuadro 2.5: Clasificación de personajes.



| CARACTERES PRINCIPALES   |  |   |
|--|--|---|
| DESCRIPCION  | OBJETIVOS  | CARACTERES  |
| <p><b>Latita:</b><br/>Famosa por sus argollas de poder y especialista en la correcta clasificación de metales.</p>   | <p>Motivar a los usuarios a realizar acciones de reciclaje de metales.<br/>Informar a los usuarios por medio de clases interactivas sobre la problemática ambiental actual.<br/>Mostrar el proceso de comercialización de latas.</p> |    |
| <p><b>Ecoambiente:</b><br/>Lanza esferas de hidrogeno y pasa la mayoría del tiempo preocupándose para que nadie interrumpa el frágil ecosistema de nuestro medioambiente.</p>                                  | <p>Promover en los usuarios el sentido de respeto por el medio ambiente.<br/>Informar sobre el proceso de reciclaje de plástico, aluminio y otros.<br/><br/>Motivar la conservación de la flora y fauna Salvadoreñas.</p>            |    |
| <p><b>Superplástico:</b><br/>Experto en la clasificación de plásticos, tiene la capacidad de lanzar hilos de plastioxigen</p>  | <p>Concientizar sobre la reutilización del plástico en los usuarios del videojuego.</p>  |   |
| ESPECIES EN EXTINCIÓN  |  |   |
| DESCRIPCION  | OBJETIVOS  | CARACTERES  |
| <p><b>La Abronia Montecristoi</b><br/>Es una especie en extinción que debe de salir de un lugar inhóspito, lleno de cazadores y trampas creadas por tala de árboles.</p>                                       | <p>Concientizar sobre el daño que ejerce la caza de especies a nuestro medio ambiente.<br/><br/>Que el jugador experimente el temor que sienten los animales al ser hacechados por personas indiscriminadas.</p>                     |  |
| <p><b>El venado:</b><br/>Esta amigable especie puede ser detectada en muchos lugares del parque, pero Ecoambiente debe tener cuidado de no arrollarlos cuando maneje el ecopickup todo terreno del parque.</p> | <p>Fomentar el respeto a las especies en extinción.</p>  |  |



|  |   |  |
|--|---|--|
| <p><b><i>Harpahaliaetus solitarius</i></b><br/>Es una especie en extinción a la cual Ecoambiente tiene que ayudar devolviendo los huevos que se le han caído del nido, si Ecoambiente se tarda mucho en reposicionar los huevos en el nido, la hembra de esta especie lo puede confundir con un depredador y atacarlo.</p> | <p>Inculcar el respeto hacia las diferentes especies y sus crías.</p> |  |
|--|---|--|


**ENEMIGOS MEDIO AMBIENTALES**

| DESCRIPCION  | OBJETIVOS  | CARACTERES |
|--|--|------------|
| <p><b><i>Cuasimodo:</i></b><br/>Medio humano, medio bestia, los cuasimodos respiran monóxido de carbono, su única debilidad es el oxígeno.</p> | <p>Representa la maldad de la contaminación, se pretende que los usuarios tomen conciencia de lo monstruoso que es destruir el medio ambiente.</p> |            |
| <p><b><i>Cazadores</i></b><br/>Gente indiscriminada que se dedica a matar animales, sin medir sus consecuencias.</p>                           | <p>Concientizar sobre el daño que ocasionan los cazadores a el medio ambiente.</p>   |            |



| GUÍAS  |  |            |
|--|--|------------|
| DESCRIPCION  | OBJETIVOS  | CARACTERES |
| <p><b>Elmer:</b><br/>El ingeniero en tecnología Positrónica del cuartel general, él le provee a los súper héroes dispositivos necesarios para su viaje al pasado</p>   | <p>Informar a los caracteres principales sobre los articulos que se debe llevar al pasado</p>                  |            |
| <p><b>Prof. Josh:</b><br/>El profesor Josh es un carácter que nunca se ha visto, es una de las incógnitas de la historia, él es el constructor de la máquina del tiempo en la cual los superhéroes viajan a San Salvador del pasado, nuestro presente. El superhéroe que culmine todas las misiones... ¿podrá verle?</p> |  |            |
| <p><b>Freddy</b><br/>Es un agente del Ministerio del Medioambiente y Recursos Naturales a cargo de la Administración del Parque Nacional Montecristo, el recibe a los visitantes del parque y les provee información necesaria</p>   | <p>Dar la bienvenida a los visitantes del parque Montecristo y orientar sobre las normas que deben seguir.</p> |            |



| GUÍAS  |   |   |
|--|---|---|
| <p><b>Josefa:</b><br/>Una agente del Ministerio del Medioambiente y Recursos Naturales a cargo de la Eco-cabaña de Información del Parque Nacional Montecristo. Ella es la encargada de hacerle saber a Ecoambiente sobre las misiones ambientalistas en las que él puede ayudar.</p>  | <p>Orienta sobre las misiones que el usuario puede realizar en Montecristo.</p>                   |    |
| <p><b>Biólogo del Parque</b><br/>El biólogo es un experto en flora y fauna, y puede beneficiar a cualquiera que responda correctamente a sus preguntas.</p>  | <p>Evalúa el conocimiento adquirido por los visitantes del parque Montecristo</p>                 |    |
| Vendedores   |   |   |
| DESCRIPCION  | OBJETIVOS   | CARACTERES  |
| <p><b>La Niña Menche:</b><br/>Es una comerciante de la micro empresa que tiene chalets en varias escuelas de la capital, así como en el área del Salvador del Mundo y el Centro Capitalino. Comprando en el chalet de la niña Menche, nuestros superhéroes pueden saciar su hambre y sed. La niña menche podría hacerle un porcentaje de rebaja de ecodólares a sus productos para el superhéroe que sea buen cliente.</p> | <p>Su finalidad es de ofertar todo tipo de productos ya sean biodegradables o no degradables.</p> |  |

Cuadro 2.6: Descripción de personajes





*Nuestros superhéroes: Ecoambiente, Latita y Superplástico son exploradores estudiantes de la Universidad Global del planeta Terra armonía, un centro de educación superior universal de exploración intergaláctica para todas las especies avanzadas de diferentes partes de la galaxia Abell 2029, a 3 millones años luz de nuestra vía Láctea. Su misión es monitorear todos aquellos rincones del universo donde pueda existir alguna forma de vida, y cómo cada una de las especies dominantes de cada planeta influye en la sustentabilidad de sus ecosistemas. Mientras se encontraban en la Vía Láctea, tomaron interés en el único sistema de estrellas capaz de sustentar la vida biológica. El sistema solar, cuyo sistema planetario poseía un único planeta que albergaba a especies parecidas a las de los exploradores, con la excepción de nunca haber desarrollado poderes especiales.*



Figura 2.17: La nave intergaláctica de nuestros superhéroes en camino a la tierra para estudiar más de cerca la causa de su deterioración ambiental.

*Mientras estudiaban al planeta tierra se dieron cuenta que estaba a punto de perecer todo aquello que sustentaba la vida como se conocía, y de convertirse en la casa de un nuevo tipo de especies radio-biológicas, cuya aberración genética amenazaba la propia existencia de su especie natal, el ser humano.*

### **Misión conjunta Nivel 1**

*Los superhéroes llegan a la tierra, a un lugar conocido como San Salvador, en donde la atmosfera está llena de monóxido de carbono, la altitud ha aumentado varios metros debido a terremotos consecuencia de las presiones tectónicas de la cuenca de Caribe a raíz del aumento en los niveles de los océanos, producto del derretimiento de glaciares masivos en los polos. Estas catástrofes, consecuencia directa del calentamiento global producido por los gases de efecto invernadero y tala indiscriminada de árboles son las condiciones ideales para los cuasimodos, estos atacan a nuestros superhéroe a su llegada. Los cuasimodos son entes semihumanos y semibestias, producto de una guerra nuclear masiva durante los eventos globales de 2094.*

*Los cuasimodos respiran monóxido de carbono, su única debilidad es el oxígeno. Ecoambiente, Latita y Superplástico serán capaces de lanzar, respectivamente: esferas de oxígeno, círculos de oxígeno e hilos de Plastioxígeno, una aleación entre las*





*moléculas de oxígeno y polihidrocarburos.*



Figura 2.18: El Salvador del Mundo en el futuro



Figura 2.19: Vista antineblina Tóxica

*Un cuasimodo puede ser eliminado con tan solo un golpe directo de cualquiera de los artefactos arriba mencionados. Así mismo, cualquier superhéroe, a pesar de contar con un traje y casco protector, no podrá sobrevivir a los ataques recursivos de plasma radiactivo que vomitan los cuasimodos. La computadora biológica incorporada a su traje les mantendrá informado del soporte de vida a cada instante durante un ataque, cuando el traje especial no pueda soportar más y los niveles de vida se encuentren en porcentajes críticos, la computadora biológica emitirá alarmas audibles, unos cuantos vómitos radiactivos mas podrían acabar con la vida de un superhéroe.*



Figura 2.20: Eliminación de los cuasimodos

*Cada superhéroe está equipado con diferentes dispositivos para hacer frente a formas de vida hostiles en su recorrido por la galaxia, entre estos, el Eco-Traje, que entre sus muchas características incluye una computadora biología dotada de sensores que permiten al superhéroe hacer uso de visión Anti-Gas, niveles de vida y de oxígeno así como alertas audibles que le proveen información detallada en cada momento. La siguiente figura muestra la visión Anti-Gas, que permite discriminar entre las partículas de gases tóxicos que ofuscan la visión de nuestros superhéroes y permite reconstruir imágenes de vida biológica, terrenos y objetos del entorno.*



Figura 2.21: Ecoambiente entregando la esmeralda de berilo

*Ellos necesitan encontrar el escondite de los últimos humanos del planeta, ahí se encontrarán con el Profesor Josh, un científico descendiente de los pocos sobrevivientes de la especie humana. Él junto con su ayudante Elmer les proveen instrucciones a los superhéroes antes de viajar al pasado, al año 2009 para salvar a la raza humana evitando la tercera guerra mundial y deterioración medioambiental por medio de la promoción de un medioambiente sustentable. El viaje en el tiempo es posible gracias a la energía potencial extraída de la esmeralda de berilo que los superhéroes han encontrado, permitiendo energizar la máquina del tiempo que el profesor Josh ha construido.*



**Año: 2009**

*La historia de los protectores del medioambiente se divide en tres, tras viajar en el tiempo, dependiendo de cada superhéroe y sus especialidades.*

## **ECOAMBIENTE:**

*Ecoambiente es el más joven de los superhéroes, nació en el planeta Praeclarus Ambientem, luego se mudó a Terrarmonia para concluir sus estudios superiores de exploración galáctica.*

### **El Salvador del Mundo**



Figura 2.22: Superhéroe Ecoambiente

*Los superhéroes llegan en su nave a la zona de El Salvador del Mundo en San Salvador, El Salvador; Al aterrizar, la econave espacial se transforma en un Eco-panel del MARN para que su origen distante se mantenga en secreto.*

*Latita y Superplástico se separan de Ecoambiente para realizar tareas ambientalistas mientras que Ecoambiente decide quedarse en el área para investigar y encontrar las oficinas del MARN.*

*El conduce el Eco-panel en busca de dicha institución. Tendrá que respetar las señales de tráfico, dar paso a peatones y manejar cortésmente para llegar a su destino. El Eco-panel solo cuenta con unos cuantos galones de combustible, si pasa mucho tiempo manejando perderá la reserva y no podrá alcanzar su objetivo.*



Figura 2.23: Ecopanel

*Si Ecoambiente encuentra las oficinas del MARN, se le informará sobre su misión preliminar y tendrá acceso a todo tipo de información medioambiental a través de los videos disponibles en la Videoteca de la institución. Como bonos extra, se le otorgará una determinada cantidad de Ecodólares para que pueda reabastecer de combustible al Eco-panel. Cuando Ecoambiente se acerque a una gasolinera para adquirir combustible, se le dará la opción de reabastecerse con Diesel, Súper, regular, Etanol y Biodiesel.*

*El Eco-panel está diseñado para cualquier tipo de combustible, no obstante, si se selecciona un combustible que sea 'amigable' con el medioambiente, se le otorgará extra ecopuntos.*



Figura 2.24: Las figuras anteriores muestran a Latita en la zona del Salvador del Mundo en el año 2009

Más adelante en la historia, *Ecoambiente* podrá manejar un vehículo energizado con baterías cargadas con energía solar.

Su misión consiste en mostrar a las demás personas la forma correcta de clasificar desechos sólidos para su posterior reciclaje. Él podrá recoger desechos sólidos en diferentes áreas, buscar depósitos exclusivos para cada desecho y de esa forma obtener *Ecopuntos* esenciales para que le sea asignada su próxima misión. Al mismo tiempo, *Ecoambiente* podrá interactuar con diferentes personas dentro de la historia, comprar comestibles y encontrar objetos especiales. Cuando se encuentre caminando por la zona, tendrá que tener cuidado al cruzar la calle, de lo contrario podría ser investido por algún conductor si este no se percata del peligro.

Al igual que los humanos, *Ecoambiente* necesita recuperar su energía después de caminar o correr por largos periodos de tiempo, por lo tanto, es necesario que consuma alimentos que puede comprar en chalets o maquinas dispensadoras de golosinas y bebidas.



Figura 2.25: Comprando en el chalet de la niña Menche.





Figura 2.26: Comprando de una maquina dispensadora de golosinas.

*La escena del área del Salvador del Mundo y su gestión medioambiental terminan cuando Ecoambiente haya encontrado las oficinas del Ministerio del Medio Ambiente y Recursos Naturales para instalar los depósitos de desechos sólidos en la plaza del Salvador del Mundo y alcanzado una cierta cantidad de Ecopuntos para pasar a su siguiente misión.*

### **MonteCristo**

*Cuando Ecoambiente haya culminado su misión en la zona del Salvador del Mundo, este es trasladado al Parque Nacional Montecristo.*



Figura 2.27: Mapa del área de Monte Cristo que Ecoambiente tendrá que administrar durante el nivel 3.



| LEYENDA:  |                                     |
|---|-------------------------------------|
|    | Centro de Administración del Parque |
|    | Ecocabaña                           |
|    | Árbol centenario                    |
|    | Orquidario                          |
|    | Mirador                             |
|   | Rio                                 |
|  | Calle de tierra                     |
|  | Calle empedrada                     |

Figura 2.28: Simbología utilizada en el Parque Nacional Montecristo

*El guardabosques del parque le pedirá a Ecoambiente cuidar ciertos aspectos del parque, cuando todas las tareas se hayan realizado con éxito podrá avanzar hasta el siguiente nivel. El superhéroe podrá tomar una caminata virtual por el parque, en donde interactuará con ciertas especies en extinción y disfrutará de un ambiente natural. El superhéroe podrá entablar diálogos con los guardabosques y llevar a cabo actividades especiales como “Identifica la orquídea”, “encuentra a la especie en extinción”, “apaga el incendio”, “Regrésale el Huevo al Harpyhaliaetus solitarius”, entre otros.*

*La primera tarea de Ecoambiente dentro del parque es obtener las llaves del Eco-Pickup y el Dispositivo de Posicionamiento Global (GPS), el biólogo que se encuentra en el Casco puede proveer estos artículos si se contesta acertadamente a 3 de sus preguntas, luego estará listo para aceptar cualquier prueba que la encargada de la Info-cabaña le presente.*



Figura 2.29: Biólogo realizando preguntas a Ecoambiente



Figura 2.30: Encargada de la Info-cabaña

Las tres pruebas que Ecoambiente debe llevar a cabo para completar el nivel son:

### 1. **Recuperar la Abronia Montecristo**

*Esta especie en extinción se ha salido del parque y se encuentra en una zona donde cazadores sin escrúpulos pretenden cazarla, por lo tanto es necesario que ella llegue hasta Ecoambiente para que este la lleve de regreso al parque.*

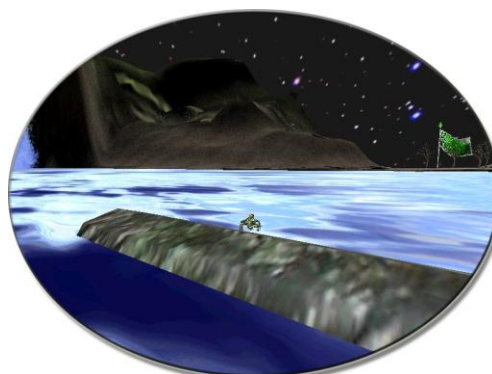


Figura 2.31: Recuperando la Abronia Montecristo



## 2. **Regrésale los huevos al Harpyhaliaetus solitarius**

Esta misión empieza cuando la encargada de la Info-cabaña le informa a Ecoambiente que unos ecoturistas han divisado unos huevos no muy lejos del camino, nuestro superhéroe debe de retornar los huevos al nido justo arriba, en una de las ramas más altas de un árbol centenario. Ecoambiente debe regresar los huevos antes que el águila solitaria se dé cuenta que este los lleva. Si Ecoambiente es descubierto antes de retornarlos al nido, le espera una mala experiencia con una de las aves en extinción propias del parque.



Figura 2.32: Ecoambiente y águila solitaria

## 3. **Incendios Forestales**

La *Penelopina nigra* acaba de terminar de empollar a sus crías y todos los guardabosques se encuentran en dicho acontecimiento, por lo tanto Ecoambiente debe de vigilar el parque en caso que algún incendio ponga en peligro la flora y la fauna. Durante esta misión se le otorga un planeador para permitir apagar incendios desde el aire si estos han crecido fuera de control. Si Ecoambiente detecta un incendio antes que se expanda, puede apagarlos usando el Eco-Hidratador en spray.



Figura 2.33: Eco- Planeador y Eco-Hidratador para sofocar incendios

Otras actividades dentro del Parque:

### 1. **Orquidario (Jardín de los Cien Años)**

Como se sabe, el parque también tiene una de las más grandes colecciones de orquídeas en El Salvador, algunas de ellas en peligro de extinción. En cualquier momento





mientras se encuentre en Monte Cristo, Ecoambiente puede visitar el orquideario para aprender sobre la flora y llevar a cabo actividades para recolectar extra ecopuntos.



Figura 2.34: Jardín de los cien años

## 2. Casco

En la sección de especies en peligro de extinción del casco, nuestro superhéroe podrá aprender sobre la fauna propia de Monte Cristo. Es aquí donde el podrá obtener el conocimiento necesario para responder a las preguntas del biólogo y obtener el GPS y llaves de Eco-pickup 4x4 del MARN.



Figura 2.35: Casco Monte Cristo

## 3. Limpieza de desechos

Algunos visitantes del parque no observan las normas y reglas para mantener limpio las áreas de turistas, por lo tanto, es necesario que Ecoambiente recoja y clasifique correctamente estos desechos para servir como ejemplo a los que no protegen nuestro ecosistema.

## 4. Consumo de Fruta

Si Ecoambiente no utiliza el Eco-pickup 4x4 del MARN, él se cansará y no podrá usar toda su energía, sin embargo, él podrá recoger fruta debajo de ciertos árboles

La Misión en el Parque Nacional Monte Cristo terminará cuando se hayan pasado todas las pruebas y actividades especiales.



## El Rio Lempa y los huevos de tortuga

Cuando *Ecoambiente* haya culminado su misión en el Parque Monte Cristo, será transportado hasta la presa hidroeléctrica 5 de Noviembre. Es desde ahí donde el superhéroe se desplazará corriente abajo para limpiar desde un vehículo acuático toda basura de las aguas. En su travesía por el Rio Lempa, nuestro héroe se encontrará con muchos obstáculos y retos mientras limpia, tanto como pueda, las aguas de desechos sólidos. La misión para la concientización de descontaminación de las aguas termina cuando *Ecoambiente* haya llegado al estero la Puntilla en la Costa del Sol. En donde como bonos, el superhéroe tendrá que defender unos huevos de tortuga en la playa contra 3 diferentes depredadores: Un perro, una garza tigre y un humano. *Ecoambiente* tendrá que esconderse a como dé lugar de los ladrones de huevos, y sin ser detectado ( un buen escondite puede ser entre las rocas), les disparará esferas de hidroxígeno (agua viscosa) para ahuyentarlos.

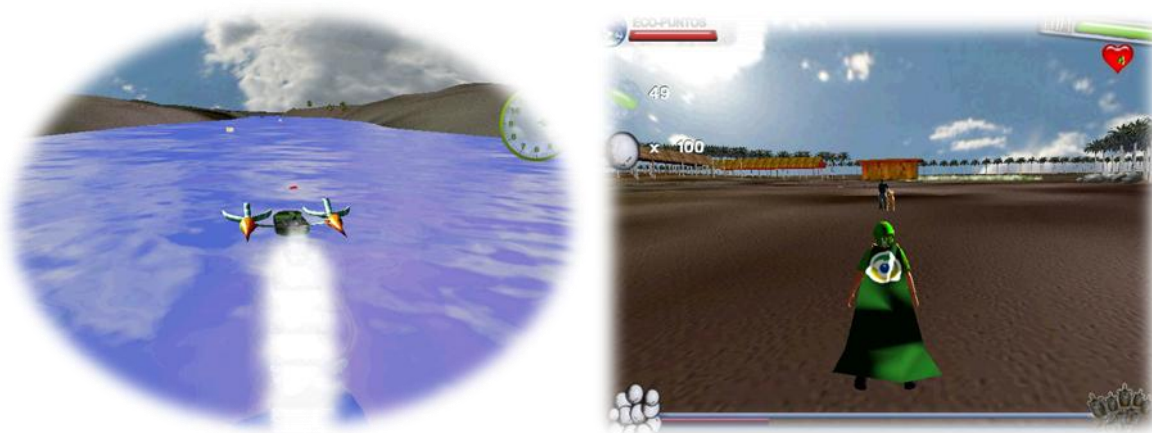


Figura 2.36: Limpiando el Rio Lempa (izquierda), Cuidando los huevos de la Tortuga Carey (derecha)

Mientras limpia el Rio Lempa, *Ecoambiente* tiene que evitar salir del agua, de lo contrario, perderá tiempo mientras empuja su *Eco-Lancha*. Posteriormente, mientras cuida los huevos de tortuga hasta que nazcan, el puede usar los ecobinoculares mientras se esconde y lanza esferas de hidroxígeno para ahuyentar a los ladrones de huevos.

El nivel de vida de *Ecoambiente* disminuirá con cada mordida del perro o picada de gavita, al mismo tiempo, perderá el nivel si el ser humano le descubre.

Si *Ecoambiente* logra cuidar los huevos de tortuga y si evita que no se roben muchos (tiene que permitir que se gestionen al menos 100 tortuguitas), se le otorgarán extra ecopuntos y terminara su misión.



Figura 2.37: Superhéroe Latita

## LATITA:

*El planeta madre de Latita es Metallum, un planeta en donde los metales componen el 90% de la superficie. Muchos alienígenas tienen buenas relaciones comerciales con Metallum, ya que muchos componentes necesarios para sus tecnologías de viaje interestelar requieren muchos metales, como aluminio, oro, cobre, estaño entre otros. La demanda es tan grande, que de no promover el correcto reciclado de estos, dentro de poco Metallum dejaría de existir ya que el uso indiscriminado de su recurso primario terminaría con el planeta. Al igual que Ecoambiente, ella también es estudiante residente de la Universidad de Terrarmonia.*

*Latita podrá interactuar con otros caracteres en El Salvador del pasado. Esta iteración consistirá en establecer diálogos con los niño y jóvenes, realizar compras de comestibles en los chalets de las instituciones, interactuar con los otros superhéroes, así como llevar a cabo actividades como aplastar latas, recoger desechos*

*sólidos, comprar y consumir comestibles, clasificar los desechos sólidos en tres categorías: Plásticos, Aluminios y Otros.*

### Clasificación de desechos sólidos

*Durante su primer visite a la Escuela Medioambiental, latita podrá enseñar a recoger y clasificar desechos sólidos que pueda encontrar en la escuela, cuando haya depositado correctamente los objetos o basura desechada, y llenado los contenedores, el Eco-camión recolector de desechos llegará a la escuela y recogerá las bolsas dentro de la Eco-Estación del MARN, de esa forma, ella podrá pasar al siguiente nivel.*



Figura 2.38: Depósitos para basura



Figura 2.39: Ecoestación

### El Centro de San Salvador

*Ecoambiente, Superplástico y Latita llegarán en el Eco-panel al Centro de la Capital. En “El Centro”, Latita tendrá la misión de recolectar una determinada cantidad de Ecodólares a través de la recolección de todo tipo de metales y latas de aluminio para llevarlas a un centro de canje que tendrá que encontrar. Mientras se encuentre buscando latas, Latita tendrá que tener mucho cuidado al cruzar las calles, ya que si lo hace sin cuidado, podrá ser atropellada y perder niveles de vida, tendrá acceso, al mismo tiempo, podrá comprar comestibles e interactuar con algunos ciudadanos de la capital. Latita podrá adquirir mayor cantidad de ecodólares si aplasta los metales antes de entregarlos al centro de reciclaje.*



Figura 2.40: Latita recolectando latas y otros desechos sólidos en la plaza General Gerardo Barrios (izquierda).





La misión termina cuando nuestra Súper Heroína haya llevado cierta cantidad de metales a los centros de canje y logre acumular 100 ecodólares.

### La Ecogóndola

Este nivel está basado en el tiempo; Latita, utilizando los poderes de su Ecopulseras, debe de lanzar todos los desechos sólidos metálicos en el contenedor de metales. Un brazo mecánico hará más difícil esta tarea a medida que el tiempo avance. Cada vez que Latita falle en su puntería perderá la oportunidad de tirar el objeto metálico dentro de su contenedor, así mismo, si ella bota objetos que no son de metal, sus ecopuntos disminuirán y el tiempo se hará más corto. Al tirar objetos de metal de forma consecutiva se le añadirá extra tiempo.

En este nivel Latita tendrá la oportunidad de recolectar tantos ecopuntos como el tiempo y su puntería permitan, permitiéndole posicionarse en una posición de puntajes globales alta.



Figura 2.41: Ecogondola

Este nivel termina cuando Latita haya obtenido los ecopuntos necesarios para avanzar. La figura anterior (ejemplo preliminar) muestra a Latita en una fábrica de reciclaje disparando círculos de poder con sus Ecopulseras para botar los objetos de metal en su respectivo contenedor.

### Regreso a Eco-Clases

En este nivel, Latita se presentará frente a sus alumnos dentro de un salón de clases en donde responderá a sus preguntas, si las contesta sin fallar podrá considerarse como un ejemplo a los demás al haber culminado sus misiones en El Salvador. Para prepararse para responder todas las preguntas, Ecoambiente podría aprender desde las computadoras y libros del CIDOC dentro de las instalaciones del MARN y enseñarle a Latita a aclarar cualquier duda antes de someterse a las preguntas de sus alumnos.



Figura 2.42: Clases interactivas

## SUPERPLÁSTICO:



Figura 2.43: Superhéroe Superplástico

*Superplástico es originario de Plasticus, una luna artificial que orbita alrededor del planeta Metallum, el hogar de Latita. Plasticus funciona como bahía de acoplamiento para cruceros eléctricos, ya que la alta electricidad estática de las especies de esta civilización, genera muchos problemas al aterrizar directamente en Metallum. De esa forma, el aislamiento eléctrico de la luna artificial le provee a los cruceros un lugar donde esperar para obtener las materias primas metálicas para los generadores hipereléctricos de su planeta de origen. Superplástico, al igual que Latita, ha perfeccionado sus técnicas de reciclaje, en este caso para los plásticos, ya que su uso es indispensable para la sobrevivencia de su luna. Junto a Ecoambiente y Latita. Superplástico está por terminar su último curso en la Universidad de Terrarmonia, siendo esta aventura la actividad necesaria para que él pueda obtener su título de Guardian Intergalactico de Recursos Renovables y Sustentables con Especialidad en la manipulación molecular de polihidrocarburos.*

### Concientización de los usos del plástico

*La primera misión de Superplástico en el Centro de San Salvador es, al igual que sus amigos, recoger y clasificar los desechos sólidos que encuentre. A diferencia de los otros, el debe de guardar todo desechos sólido de plástico para colaborar con la encargada de la tienda Plastilandia, al costado oriente de la catedral, a construir juguetes de plástico para los niños, sin embargo, ya que existe escases de plástico debido a que pocas personas están clasificando los desechos sólidos de plástico, es aquí donde nuestro súper héroe de plástico puede ayudar, al recolectar todo el plástico que pueda y llevarlo de regreso a la tienda, en donde la dueña se*



encargará de reciclar y generar el plástico necesario para crear diversos juguetes.



Figura 2.44: Depositando desechos sólidos en los contenedores instalados al costado de la plaza Libertad (izquierda). Entrando a Plastilandia (derecha)

### El Laberinto y el Castillo

Después que Superplástico haya superado el nivel anterior, será transportado a un inmenso laberinto en donde tendrá que encontrar la salida antes que su tiempo termine. En diferentes secciones del laberinto el podrá recolectar relojes para comprar más tiempo, así mismo, Toto, el estudiante, le ofrecerá información importantísima sobre el medioambiente a nuestro súper héroe, y le echará una mano para salir rápido del confuso lugar.

Si Superplástico logra encontrar la salida, alguien lo estará esperando en la corte de un Castillo.



Figura 2.45: Superplástico hablando con Toto en el laberinto de plástico





Figura 2:46: Superplástico en la entrada del Castillo al final del laberinto

## ***Desenlace de la Historia***

*Cada superhéroe aportará a la sustentabilidad del medio ambiente cuando hayan terminado sus respectivas misiones. El avance de cada jugador puede reflejarse en la sección de 'Bitácora', el jugador con mayores estrellas será el mejor guerrero medioambientalista!*

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.

**Ecodólares:** Necesarios para realizar compras

**Ecopuntos:** Al alcanzar ciertos Ecopuntos, el usuario puede intercambiarlos por extra vidas o por Ecodólares.

**Vidas:** El usuario inicia con 3 vidas, el usuario puede perder una vida al:

- Perder el nivel
- Saltar de un edificio
- Ser atropellado por un automóvil
- Ahogarse
- Ser alcanzado por suficiente vomito radioactivo lanzado por los cuasimodos
- No comer durante un largo periodo de tiempo





### 2.4.5. Temática y Alcance del Videojuego

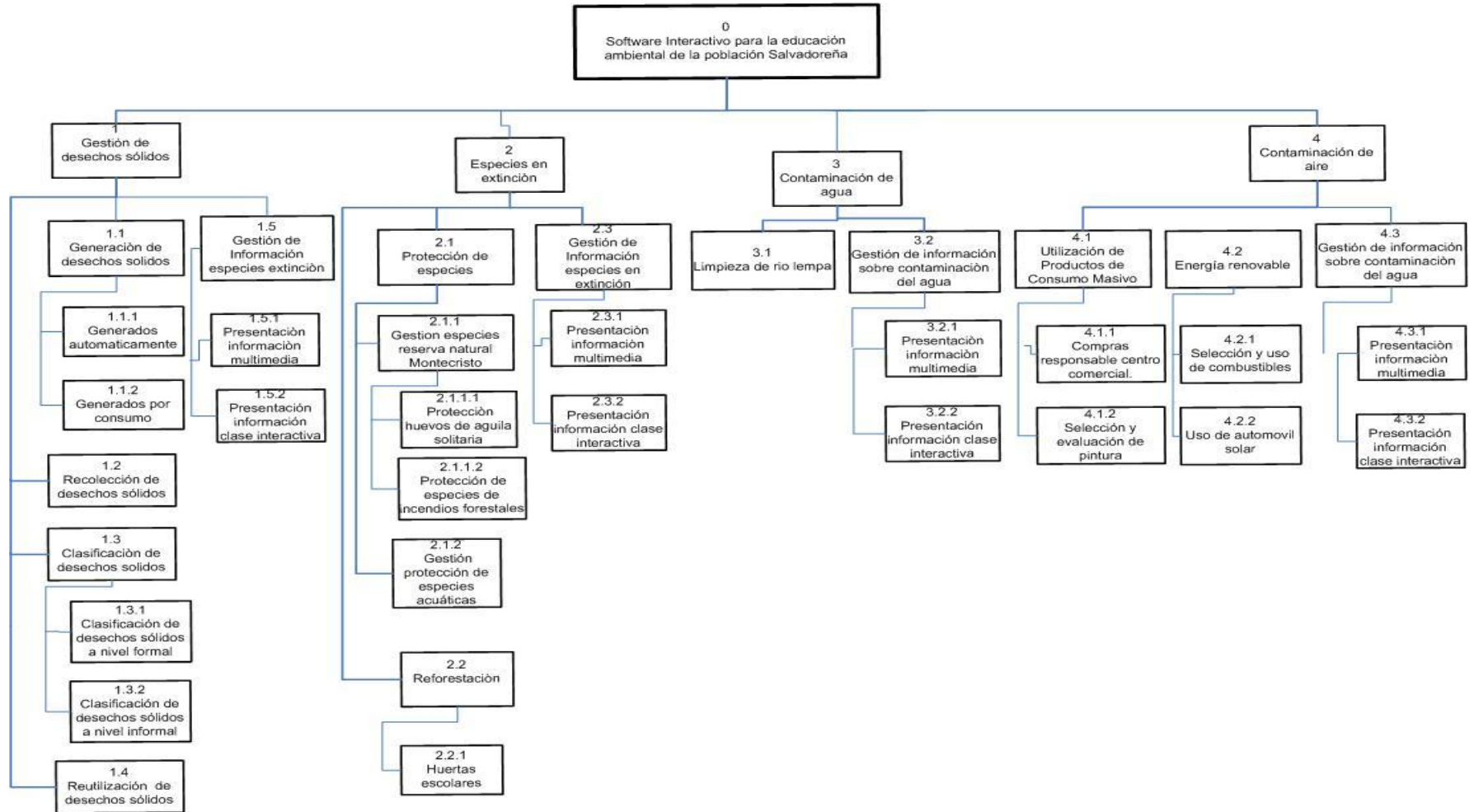


Figura. 2.47: Temática del videojuego



Un sistema ya sea informático o de software, tiene un alcance definido y un número finito de elementos que lo componen. Así mismo, un videojuego educativo como en este caso, necesita delimitar exactamente su alcance y temática. Ya que este software está orientado a la educación ambiental, su área de enfoque será exclusivamente relacionada con el medioambiente. Habiendo delimitado el área, ahora se procede a definir las diferentes temáticas y elementos que componen a cada una de estas (Ver Figura 2.47).

#### **2.4.5.1. Temática**

La temática que envolverá a cada nivel determinará el escenario que el software presentará al jugador, se abordarán 5 temas en total:

- Clasificación de Desechos Sólidos
- Utilización de Energía Renovable
- Utilización de Productos de Consumo Masivo que no dañan el medioambiente
- Especies en Extinción
- Contaminación de Recursos Acuíferos
- Reforestación

##### Clasificación de Desechos Sólidos

Este tema, a diferencia de los demás, se incluirá en todos los niveles para los tres caracteres, a excepción del Nivel Uno (En el futuro), ya que es el tema de mayor interés para el nuevo programa de Educación Ambiental de El Salvador. Esta temática se desarrollará de dos diferentes formas a lo largo del juego: la formal y la no formal.

##### Implementación Formal:

Se lleva a cabo exclusivamente dentro de dos centros escolares que se incluirán en la historia para los niveles 2 y 4 para el carácter **Latita**. Los desechos sólidos se clasificarán en: **Aluminio, Plástico y Otros**.

##### Implementación No Formal:

Se lleva a cabo exclusivamente en el área del Centro de la Capital y el área del Salvador del Mundo. Aquí los desechos sólidos se clasificarán en cinco, al igual que el programa en gestión a cargo del MARN: **Aluminio, Plástico, Papel, Material Peligroso y Materiales No Reciclables**.



### Utilización de Energía Renovable

Se abordará este tema a través de las siguientes situaciones:

- Cuando el jugador reabastezca su vehículo de combustible: Tendrá la opción de seleccionar Gasolina Súper, Regular, Diesel, Etanol y Biodiesel.
- Cuando el jugador tenga los recursos suficientes dentro de la historia para convertir al Eco-Panel en un vehículo impulsado por energía solar o hidrogeno.

### Utilización de Productos de Consumo Masivo que no daña el medioambiente (Incluye Contaminación del Aire)

Se abordará este tema a través de las siguientes situaciones:

- Cuando el jugador se encuentre realizando la tarea de pintar una pared. Tendrá la opción de seleccionar entre pinturas a base de resina, agua, látex o aerosoles.
- Cuando el usuario se encuentre adquiriendo productos de Consumo Masivo en el supermercado.

### Especies en Extinción

El énfasis que se hará a cada especie en extinción (incluye flora y fauna), dependerá de la especie. Los diferentes escenarios donde aparecerán las especies en extinción según la historia del videojuego son:

- El parque Nacional Montecristo (Nivel 3 de Ecoambiente)
- Los huertos de las Escuelas para los niveles 2 y 4 de Latita.
- La escena de protección a los huevos de tortugas Carey (Extensión del Nivel 4 de Ecoambiente).
- Información a través de Multimedia basada en los catalogo informativos del CIDOC (Centro de Información y Documentación Ambiental)

El cuadro 2.7 y cuadro 2.8, muestran las especies<sup>78</sup> de flora y fauna en extinción propias de El Salvador que serán incluidas en el videojuego.

---

<sup>78</sup> Listado Oficial de Especies de Fauna Silvestre en Peligro de Extinción en El Salvador, Diario oficial, jueves 29 abril 2004.

*Fauna.*

| No. | Nombre común                          | Nombre científico         |
|-----|---------------------------------------|---------------------------|
| 1   | Tacuazín de agua                      | Chironectes minimus       |
| 2   | Mono araña                            | Ateles geoffroyi          |
| 3   | Murciélago mexicano nariz De cerdo    | Choeronycteris mexicana   |
| 4   | Murciélago sureño de nariz Grande     | Leptonycteris curasoae    |
| 5   | Murciélago alado blanco               | Diaemus youngi            |
| 6   | Murciélago de Underwood con gorra     | Eumops underwoodi         |
| 7   | Ratón de cola crestada                | Habromys lophurus         |
| 8   | Rata de madera mexicana               | Neotoma mexicana          |
| 9   | Nutria                                | Lontra longicaudis        |
| 10  | Taira                                 | Eira barbara              |
| 11  | Puma                                  | Puma concolor mayensis    |
| 12  | Margay, ocelote                       | Leopardus pardalis        |
| 13  | Tigrillo                              | Leopardus wiedii          |
| 14  | Cuche de monte                        | Tayassu tajacu            |
| 15  | Venadito rojo                         | Mazama americana          |
| 16  | Ballena azul                          | Balaenoptera musculus     |
| 17  | Ballena                               | Balaenoptera physalus     |
| 18  | Pato real                             | Cairina moschata          |
| 19  | Pato enmascarado                      | Penelope purpurascens     |
| 20  | Pajuil, Chacha negra                  | Penelopina nigra          |
| 21  | Hocofaisán, pahuil, pajuil            | Crax rubra                |
| 22  | Codorniz ocellada, codorniz           | Cyrtonyx ocellatus        |
| 23  | Anhinga americana, pato aguja         | Anhinga anhinga           |
| 24  | Pájaro músico                         | Botaurus pinnatus         |
| 25  | Avetorito americano, Garzita de tular | Ixobrychus exilis         |
| 26  | Garza tigre gorjinuda y jorjora       | Tigrisoma mexicanum       |
| 27  | Garza cucharón, charancuaco           | Cochlearius cochlearius   |
| 28  | Ibis cariblanco, Ibis                 | Plegadis falcinellus      |
| 29  | Espátula rosada, garza morena         | Platalea ajaja            |
| 30  | Zopilote rey, rey zope                | Sarcoramphus papa         |
| 31  | Milano cabecigris, gavilán            | Leptodon cayanensis       |
| 32  | Águila solitaria, águila              | Harpyhaliaetus solitarius |
| 33  | Polla de agua                         | Laterallus ruber          |
| 34  | Pájaro marimba                        | Aramus guarauna           |

Cuadro 2.7: Fauna en extinción



## Flora

| No. | Nombre común          | Nombre científico                   |
|-----|-----------------------|-------------------------------------|
| 1   | Helecho Arborescente  | Schaeropteris horrida               |
| 2   | Zamia                 | Zamia herrerae                      |
| 3   | Llorón                | Saurauia waldheimii                 |
| 4   | Manune                | Cordia collococca                   |
| 5   | Gallito               | Billbergia pallidiflora             |
| 6   | Copal santo           | Bursera graveolens                  |
| 7   | Palo de pólvora       | Capparis discolor                   |
| 8   | Campanilla            | Ipomoea ophioides                   |
| 9   | Cuculmecca            | Dioscorea salvadorensis             |
| 10  | Encino                | Quercus esesmilensis                |
| 11  | Platanillo de Montaña | Heliconia collinsiana var. Velutina |
| 12  | Funera                | Dalbergia calderonii                |
| 13  | Orquídea              | Amparoa beloglossa                  |
| 14  | Orquídea              | Beadlea miradorensis                |
| 15  | Mezcal                | Ulmus mexicana                      |
| 16  | Guayacán              | Guaiacum sanctum                    |
| 18  | Majagua               | Hampea reynae                       |

Cuadro 2.8: Flora en extinción.

### Contaminación de Recursos Acuíferos

Se abordará este tema a través de las siguientes situaciones:

- Cuando el jugador se encuentre en el nivel 4 de Ecoambiente, corriente abajo en el Río Lempa, limpiando el agua de basura que encuentre durante la travesía.
- Dentro del parque Montecristo, en el nivel 3 de Ecoambiente, se sancionará al jugador si tira basura en ríos y nacimientos de agua del parque.

### Reforestación.

- Se desarrollará esta temática durante los escenarios del parque Montecristo, mediante la gestión de obtención de leña, incendios forestales e información de guías turísticos del MARN.
- Como extensión del nivel 3 de Ecoambiente, a través de un escenario sobre la reforestación.



## **CAPITULO III: DETERMINACIÓN DE REQUERIMIENTOS**



### 3. DETERMINACION DE REQUERIMIENTOS

Para Ian Sommerville requerimiento es “Una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste”.<sup>79</sup>

Según la IEEE “Es una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal”<sup>80</sup>. Analizando las definiciones anteriores, la determinación de requerimientos permite establecer y detallar las capacidades y/o restricciones que se deben cumplir, ya sea derivados de una necesidad de usuario identificada, o bien, estipuladas en un contrato, estándar u otro documento formal. Los requerimientos del Software Interactivo para la Educación Medioambiental de la Población Salvadoreña se dividen en tres áreas que son: desarrollo, operativas e informáticos.

#### 3.1. REQUERIMIENTOS DE DESARROLLO.

Los requerimientos de desarrollo establecen las características de software, hardware y recurso humano necesarios para desarrollar el videojuego.

##### 3.1.1. Software de desarrollo.

Son todas las aplicaciones que conformaran el ambiente de desarrollo del videojuego como: sistema operativo, lenguaje de desarrollo y utilerías que permitirán ejecutar aplicaciones o dispositivos específicos.

###### 3.1.1.1. Sistema operativo

Para el desarrollo del proyecto se requiere de un sistema operativo que sea compatible con las herramientas de desarrollo a emplear, MICROSOFT WINDOWS XP SP2 se selecciona para el desarrollo del proyecto por ser el sistema operativo que cumple con los requisitos de las herramientas a utilizar; principalmente por cumplir con las características siguientes:

- Calidad de software
- Compatibilidad de hardware
- Soporte técnico
- Costos

<sup>79</sup> Sommerville, Ingeniería del Software, Pearson Educación, S.A., Madrid, 2005, Pág. 108

<sup>80</sup> IEEE, Standards Collection: Software Engineering, Standard 610.12-1900, IEEE,1993.



### 3.1.1.2. Lenguaje de desarrollo.

El desarrollo de un videojuego requiere de un lenguaje de programación que permita codificar los resultados de la etapa de diseño del sistema, de una manera rápida, confiable y con la versatilidad suficiente para cubrir las expectativas de los usuarios. El lenguaje seleccionado debe ser soportado por la plataforma de desarrollo elegida (MICROSOFT WINDOWS XP). A continuación se describen con más detalles las características en base a las cuales se selecciono Visual Studio C# como lenguaje de programación idóneo para el desarrollo del software propuesto.

| a) Compatibilidad con otras herramientas de desarrollo  |  |
|---|--|
| Característica  | Justificación  |
| a.1) Es necesario utilizar un lenguaje y entorno de desarrollo que permitan trabajar en conjunto con otras herramientas.                      | a.1) VC# EE como parte del entorno de desarrollo Visual Studio .NET, permite una integración sin precedentes con otras herramientas como Paint, aplicaciones Adobe, herramientas de modelado, documentos de texto, XML, XAML, entre otras, una característica común es que, si el archivo generado por alguna herramienta es arrastrado a la ventana de solución del entorno .NET. |
| a.2) Necesidad de alternar entre el diseño de recursos multimedia, texto, etc. y la integración programática de estos al proyecto o solución. | a.2) El entorno de desarrollo Visual Studio .NET se ejecuta en hilos que no afectan a otras herramientas de desarrollo, permitiendo cambiar de aplicación sin necesidad de cerrar el entorno y reconociendo automáticamente si algún recurso ha sido modificado por la herramienta en que se creo, actualizando el recurso en tiempo real  |

Cuadro 3.1: Compatibilidad con otras herramientas de desarrollo

| b) Ser un lenguaje orientado a objetos para facilidad de la documentación del diseño Orientado a Objetos que es el implementado para este proyecto.     |   |
|---|---|
| Criterio  | Justificación   |
| b.1) No debe de existir una demora entre la documentación y las clases, métodos y estructuras así como la lógica que se implementa en el código fuente. | b.1) VC# es un lenguaje de alto nivel exclusivamente orientado a objetos, al mismo tiempo, bajo el entorno .NET, se pueden crear con poco esfuerzo, esquemas de clases, funciones y estructuras que componen un proyecto o solución a través de asistentes. |

Cuadro 3.2: Lenguaje orientado a objetos





| c) Debe de permitir alcanzar los objetivos y requerimientos planteados en el diseño preliminar de SIEMPOSAL, así como su compatibilidad plena con el sistema operativo anfitrión. |  |
|---|--|
| Característica  | Justificación  |
| c.1) Es indispensable alcanzar los objetivos planteados para el software.   | c.1) VC#, a través de las bibliotecas DirectX, permite generar aplicaciones 3D robustas, limitadas solamente a las prestaciones del Hardware donde se ejecute una aplicación escrita en este lenguaje  |
| c.2) Compatibilidad plena con el sistema operativo anfitrión  | c.2) Las bibliotecas empaquetadas del Framework 2.0 (parte de SP2), ya residentes en el sistema operativo, permiten a las aplicaciones escritas en código manejable (VC#), utilizar libremente estos recursos, aumentando la compatibilidad y ligereza de los ejecutables compilados. Al mismo tiempo y dado que VC# Express es el lenguaje designado y recomendado para el desarrollo de videojuegos, por los creadores de las bibliotecas de XNA <sup>81</sup> , por su flexibilidad y facilidad de uso, las bibliotecas XNA añaden un valor agregado ya que XNA es un "Wrapper" <sup>82</sup> de las bibliotecas DirectX. |

Cuadro 3.3: Compatibilidad plena con el sistema operativo

| d) Debe de ser conocido por la comunidad desarrolladora o programadores   |  |
|---|--|
| Criterio  | Justificación  |
| d.1) Es necesario que el lenguaje sea conocido por muchos desarrolladores para futuras mejoras o mantenimiento. | d.1) A pesar que no se cuenta con información referente al número de programadores y desarrolladores que manejan y conocen VC#, se sabe que junto con Visual Basic conforman el lenguaje de mayor aceptación para tutoriales en línea, existen una gran cantidad de comunidades de desarrolladores en C# cuyos sitios expresan la proliferación del conocimiento para todos aquellos que desean conocerlo. Cabe mencionar que un programador experto en C++ o Visual Basic, no tardaría mucho tiempo en convertirse un experto en C#, ya que el material de apoyo gratuito para su aprendizaje es enorme y de fácil acceso a través de internet. |

Cuadro 3.4: Lenguaje conocido por los desarrolladores

<sup>81</sup> XNA, conjunto de bibliotecas SDK para el desarrollo, pero no limitado a estos, de juegos 2D y 3D para las plataformas Windows y consolas XBOX y XBOX 360.

<sup>82</sup> "Wrapper" (del inglés Envoltorio) Se le llama así a bibliotecas SDK generadas a partir de otras bibliotecas existentes, creando nuevos métodos y clases a partir de estas.



| e) El lenguaje a implementar debe permitir minimizar el tiempo de desarrollo sin afectar la calidad del producto final.                                   |   |
|---|---|
| Criterio  | Justificación   |
| e.1) Debido al corto tiempo para la creación de este proyecto, es necesario una herramienta que permita programar la solución en el menor tiempo posible. | e.1) Cuando se codifica en VSC# dentro del entorno VS.NET (también se puede usar aplicaciones alternas o un bloc de notas para escribir el código y bajo MS-DOS generar archivos los ejecutables) se tiene acceso a la característica <i>Intellisense</i> (ver glosario de términos) que permite agilizar la tarea de programar.  |
| e.2) El tiempo corto de etapas de codificación de la solución no debe de afectar la calidad del producto final.   | e.2) VSC# es un lenguaje híbrido, que tiene la productibilidad de Visual Basic 7.0 así como el “poder crudo” y estabilidad de Visual C++ <sup>83</sup>  |
| e.3) El tiempo de desarrollo debe ser menor comparado con otros lenguajes de vanguardia.  | e.3) Una de las características que aumentan la eficiencia de codificar en VSC# es que muchas gestiones de memoria, liberación y re-uso de recursos, entre otras, en muchos lenguajes se deja como tarea para el programador, sin embargo VSC# automáticamente libera todo recurso cuando las clases u objetos asociados a este se han vuelto obsoletas, ya sea en tiempo de ejecución o al cerrar la aplicación. Esto se realiza a través de un “Garbage Collector <sup>84</sup> ” (Recolector de Basura) como es conocido por la comunidad mundial de desarrolladores en este lenguaje, este algoritmo residente en memoria, es el responsable de gestionar todo recurso, liberando al programador de la tarea de hacerlo el mismo. Si bien la gestión de memoria minimiza el tiempo para los programadores, también existen otras razones como la facilidad de manipular las tareas o hilos que van a ser procesadas por el(los) microprocesador(es), capacidad de actualización de código escrito en VC# 2003 y conversiones a versiones posteriores como 2005 y 2008, entre otras. |

Cuadro 3.5: Minimizar el tiempo de desarrollo

### 3.1.1.3. Modelado 3D.

La creación de modelos 3D demanda de un software especializado para esta actividad por lo cual su selección se baso en características fundamentales para agilizar y optimizar los recursos en tres dimensiones creados.

El software seleccionado fue fragMOTION, por ser una herramienta sencilla que permite la creación de primitivas para construir modelos, esqueletos para animación y tener un entorno integrado e intuitivo.

<sup>83</sup> Según las palabras de Scott Wiltamutb Gerente del grupo de desarrollo de Visual C# NET, Foreword Pag. Índice xxi Inside C# by Tom Archer y Andrew Whitechapel, Microsoft Press 2002.

<sup>84</sup> Inside C# by Tom Archer y Andrew Whitechapel, Microsoft Press 2002 Pág. 662 en adelante.



#### **3.1.1.4. Edición de Imágenes**

La edición de imágenes requiere que el software seleccionado permita trabajar con imágenes mapa de bits, efectos vectoriales y soporte la exportación a diversos formatos estándares usados en técnicas de efectos. La aplicación que mejor se adecua a estas características es Photoshop CS2.

#### **3.1.1.5. Gestión de Información**

No se ha seleccionado para el desarrollo del software ningún gestor de bases de datos disponibles en el mercado, debido a que por la naturaleza del proyecto, es factible almacenar la información en archivos de texto en los cuales se guardará la información pertinente de cada usuario. Es por lo tanto, los archivos de texto los que mejor se adaptan a las necesidades de desarrollo del software, tomando en cuenta la factibilidad de puesta en marcha, las características más destacadas que se requieren para el almacenamiento de los datos y que son necesarias incluyen, costo, estabilidad, velocidad, seguridad y otros aspectos que serán utilizados en el proceso de desarrollo e implementación del sistema.

### **3.1.2. Hardware**

El campo de los videojuegos es extenso, existen de todo tipo y para todas las edades y recursos de hardware. Antes de proseguir con la definición de requerimientos del Hardware utilizado en el desarrollo, es necesario mencionar que estos se basan en los requerimientos de Hardware necesarios para la instalación del software a utilizar en este proyecto.<sup>85</sup> Es necesario explicar el término de *frame* o “cuadro” en el contexto de dibujado de pixeles en la pantalla de un monitor. En un juego típico 3D ya sea Open GL o DirectX, cada elemento 3D que se visualiza en pantalla se encuentra en un buffer de video, si se cuenta con suficiente memoria de video, se almacenan más elementos y por lo tanto, el tiempo para convertir y presentar estos objetos en pantalla se hace menor, permitiendo tener mas *frames* por segundo. En una tarjeta ATI Radeon de la serie 1550, con 64 MB de RAM, por ejemplo, un juego 3D típico se estaría dibujando a aproximadamente 70 fs (frames por segundo), mientras que en una tarjeta de video integrada sin memoria de video propia (pero la comparte con el RAM), ese mismo juego puede dibujar sus frames a unos 5 fs., lo que generaría una baja en la calidad del juego y experiencia de usuario.

Otro termino necesario de mencionar es “shader”, del inglés “sombreador”, que realmente es un archivo de texto con extensión .fx en donde se escriben comandos en un

---

<sup>85</sup> Anexo 8: Requerimientos de hardware para el software de desarrollo, Pág. 356.



lenguaje híbrido entre JAVA y C#, que especifican los efectos que se están implementando en algunos elementos del videojuego, como sombreado, efectos especiales de transformaciones, opacidad de colores, reflexiones, efectos de fuego, agua, humo, etc. A la fecha de la creación de este documento, la mayoría de tarjetas en el mercado soportan shaders, desde versiones 1.1 hasta 3.0 para juegos de séptima generación. En el desarrollo de este videojuego, se requiere el uso de una tarjeta de video, ya sea integrada o independiente, que soporte shaders 2.0.

De acuerdo a los requerimientos mínimos y recomendados de funcionamiento herramientas a emplear, para el desarrollo del software se necesita que el equipo informático cumpla al menos con las características presentadas en el cuadro 3.6.

| Hardware  | Mínimo                            | Recomendado  |
|---|-----------------------------------|--|
| Procesador  | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1 GHz   |
| Memoria RAM   | 320 Mb                            | 512 Mb   |
| Disco Duro  | 1.8 GB                            | 5.8 GB libres  |
| Unidad de CD o DVD  | Unidad CD o DVD                   | Unidad CD-RW / DVD   |
| Pantalla  | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits  |
| Teclado   | Teclado Microsoft o compatible    | Teclado Microsoft o compatible                                       |
| Ratón   | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |
| Tarjeta de video con soporte DirectX 9.0c y Shader Model 2. | Integrada                         | Independiente con 128 MB RAM y soporte para shaders 2.0 en adelante. |

Cuadro 3.6: Requerimientos mínimos y recomendados

### 3.1.3. Recurso humano

Es necesario para la realización de este videojuego contar con 1 director de proyectos y 4 analistas programadores.

Las tecnologías que se deben conocer para la creación de videojuegos son esencialmente programas para modelado 3D (fragMOTION, Milkshape y 3D Max), así como un lenguaje de programación que integre fácilmente las librerías XNA (creadas para el desarrollo de videojuegos).



## 3.2. REQUERIMIENTOS OPERATIVOS

Los requerimientos operativos que surgen en un proyecto, están orientados a identificar las condiciones en las que deben de estar los equipos para la correcta instalación, buen funcionamiento y ejecución del software planteado.

### 3.2.1. Generales

#### 3.2.1.1. Desempeño

Se requiere que el software tenga un buen desempeño una vez instalado en cualquier computador que cuente con las características mínimas de hardware y que tenga instalado las aplicaciones previamente descritas, este se puede medir en términos del tiempo de respuesta, carga inicial, uso de memoria RAM, y uso de CPU.

Para realizar estas pruebas se utilizarán en primer momento los equipos informáticos utilizados para el desarrollo del software, para comprobar los tiempos antes mencionados; después se hará uso de equipos informáticos existentes dentro de las aulas informáticas para comprobar los tiempos dentro del entorno de trabajo real.

Para la obtención de los datos que seguidamente se describirán, se realizaron pruebas en dos equipos de cómputo utilizados para el desarrollo del software. La prueba consistió en ejecutar una demostración del videojuego un número de 10 veces en cada equipo y anotar los resultados de tiempos de carga y ejecución, uso de memoria RAM y porcentaje de CPU utilizado por el software. La ecuación utilizada para cada promedio de desempeños ( $Desempeño_n$ ) es:

$$\left( \left[ \frac{\sum_{n=1}^{10} (Desempeño_n \text{ Máquina1})}{10} \right] + \left[ \frac{\sum_{n=1}^{10} (Desempeño_n \text{ Máquina2})}{10} \right] \right) / 2$$

Donde n es el número de prueba de desempeño, luego se sumó el desempeño promedió de cada máquina y se dividió entre el número de máquina (2).

En el cuadro 3.7 se detallan los resultados esperados de las pruebas de desempeño a ser realizadas para el sistema:



| Característica                          | Descripción  |
|---|--------------|
| Tiempo promedio de respuesta a eventos  | 0.7 segundos |
| Tiempo promedio de carga del videojuego | 15 segundos  |
| Promedio de uso de RAM                  | 30 MB        |
| Porcentaje de uso de CPU                | 40%          |

Cuadro 3.7: Desempeño esperado del software.

### **3.2.1.2. Manejo de errores.**

Se requiere que el software este preparado para responder a errores cometidos por los usuarios en la entrada de datos y que pueda notificar a estos cual ha sido la causa y la mejor manera de solucionarlo.

### **3.2.1.3. Interfaz de usuario.**

Se requiere que la interfaz de usuario del sistema sea en extremo intuitiva para que el usuario pueda familiarizarse con ella en el menor tiempo posible; y que sea lo suficientemente flexible para poder adaptarse a los cambios que sufra durante la vida útil que el sistema tenga. Se deberá crear un estándar de desarrollo para la interfaz que permita optimizar el desarrollo de la misma.

### **3.2.1.4. Disponibilidad**

Debido a la plataforma Windows, el software aquí evaluado, puede correrse desde cualquier computadora que cumpla con los requisitos establecidos en los requerimientos de operación.

### **3.2.1.5. Seguridad**

La seguridad del software estará basada en el control de partidas del jugador y en el control de las tareas permitidas/asignadas a los usuarios. Para ello se requerirá del nombre de usuario y contraseña para el inicio de sesión para llevar un control del puntaje y de las actividades que este realice. Además cada usuario podrá tener acceso a eliminar su propia cuenta. Los avances y la información de cada jugador solo podrán ser accedidos si un jugador se autentica, no se permitirá las manipulaciones de esta información por parte de personas ajenas a una cuenta.



### 3.2.2. Software

Para la instalación y ejecución del videojuego es necesario contar con un computador que tenga instalado el siguiente esquema de software.

- **Sistema operativo:** Microsoft Windows XP con SP2 es el que deberá estar instalado en el ordenador donde se ejecutará el videojuego.
- Las siguientes librerías serán de utilidad para la operación del videojuego.
- **Framework .net 2.0:** Provee las clases necesarias para la integración de la aplicación con el sistema operativo, gestión de memoria y métodos necesarios para la ejecución de programas desarrollados con la plataforma .net.
- **DirectX 9.0c redistribuible:** Es una amplia colección de APIs creadas y recreadas para facilitar las complejas tareas relacionadas con multimedia.
- **Microsoft Speech 4.0 Runtime:** Motor de texto a voz, necesario para el reconocimiento y transformación de cadenas de texto a audio.
- **L&H TTS 3000 Español:** Sintetizador de voz por software.
- **Microsoft XNA Framework Redistributable 2.0:** Provee las librerías que contienen los espacios de nombre, clases y métodos necesarios para ejecutar juegos de video desarrollados en XNA.
- **Runtime de C++ 2005:** Provee las librerías necesarias para la ejecución del videojuego..

### 3.2.3. Hardware

En base al software necesario para la operación del videojuego se establecen las características mínimos y recomendados de hardware que debe cumplir el equipo informático para instalar y ejecutar el videojuego.

| Hardware           | Mínimo   | Recomendado  |
|--------------------|--|--|
| Procesador         | Procesador Intel / Athlon 600 MHz                    | Procesador Pentium 1.8 GHz                           |
| Memoria            | 256 Mb   | 512 Mb   |
| Disco Duro         | 1.2 GB   | 2 GB   |
| Tarjeta gráfica    | Integrada con soporte DirectX 9.0 y Shader Model 2.0 | Soporte a DirectX 9.0 Shader Model 2.0 (recomendado) |
| Unidad de CD o DVD | Unidad CD o DVD                                      | Unidad CD-RW / DVD                                   |
| Pantalla           | SVGA 640 x 480 256 Colores                           | SVGA 1024 x 768 32 Bits                              |
| Teclado y rato     | Microsoft o compatible                               | Microsoft o compatible                               |

Cuadro 3.8: Características del hardware para la operación del videojuego



### Requerimientos Opcionales de hardware para funcionamiento del videojuego

- GamePad Xbox 360
- Control de Juegos Genérico
- Sistema de Altavoces

#### 3.2.4. Recurso humano

Los conocimientos requeridos para la manipulación correcta del videojuego son:

| PERFIL DE CONOCIMIENTOS DE USUARIOS DEL VIDEOJUEGO |              |               |
|--|--------------|---------------|
| Conocimiento Requerido                             | Nivel Mínimo | Nivel deseado |
| Lectura y Escritura                                | -            | -             |
| Manejo de entorno de Windows XP                    | Básico       | Medio         |
| Instalación/desinstalación de programas.           | Ninguno      | Avanzado      |
| Manejo de Mouse y teclado.                         | Básico       | Avanzado      |
| Manejo de Gamepad                                  | Ninguno      | Avanzado      |

Cuadro 3.9: Conocimientos Requeridos para manipular el videojuego<sup>86</sup>

Actualmente en los centros escolares se imparten estos conocimientos de computación desde primer grado a bachillerato, siendo este uno de los objetivos primordiales de la existencia de las aulas Informáticas.

El perfil de conocimientos que cumplen los estudiantes por nivel académico, según el contenido de la asignatura Ciencia de la Computación son:

| PERFIL DE CONOCIMIENTOS DE LOS USUARIOS DE LAS AULAS INFORMATICAS |               |                |              |
|---|---------------|----------------|--------------|
| Conocimiento Requerido  | Primero-Sexto | Séptimo-Noveno | Bachillerato |
| Lectura y Escritura   | -             | -              | -            |
| Manejo de entorno Windows XP                                      | Básico        | Medio          | Avanzado     |
| Manejo de paquetes de oficina                                     | Básico        | Medio          | Avanzado     |
| Configuración del entorno de Windows XP                           | Ninguno       | Avanzado       | Avanzado     |
| Instalación/desinstalación de programas.                          | Ninguno       | Avanzado       | Avanzado     |
| Manejo de Mouse y teclado.  | Básico        | Avanzado       | Avanzado     |

Cuadro 3.10: Conocimientos estudiantes de las aulas informáticas<sup>87</sup>

<sup>86</sup> Basado en estudios psicomotores del adolescente: Juegos psicomotores Lic. Edith Beatriz Burgos, <http://psicopedagogias.blogspot.com/2008/02/el-desarrollo-ldico.html>

<sup>87</sup> Información proporcionada por la Unidad de Programas Complementarios, MINED.





Además el 83% de los estudiantes ya ha manipulado un videojuego, por lo que se garantiza que se tienen conocimientos mínimos para operar este tipo de aplicaciones.

En base a lo expuesto anteriormente los usuarios cuentan con los conocimientos requeridos para manipular correctamente y a provechar al máximo la diversión de los videojuegos.

### **3.3. REQUERIMIENTOS DE INFORMACION**

#### **3.3.1. Requerimientos funcionales**

- El software deberá tener una interfaz amigable que facilite la interacción con el usuario, por medio del uso primordial de imágenes, texto y sonido en la aplicación, esto debido a que a los usuarios finales se le necesita llamar su atención, en su mayoría jóvenes de 7 a 18 años, para que el proceso de concientización ambiental se lleve a cabo.
- La interfaz debe explotar el factor de entretenimiento para alcanzar el objetivo de la enseñanza, por medio de escenarios virtuales reales y ficticios.
- El software deberá ofrecer niveles de dificultad a medida que se avance a niveles superiores para estimular el nivel de motivación de los jóvenes, al hacer el juego más interesante.
- El software tendrá la capacidad de almacenar información en archivos de texto, cuyos datos serán generados mediante la interacción del jugador con el software. Al culminar un determinado nivel, el software almacenará de forma automatizada la información respectiva para cada jugador.
- El software deberá permitir a un usuario crear una cuenta de sesión, para almacenar su avance y acciones dentro del juego para una futura evaluación de parte del docente a cargo o para referencia personal.
- El software presentará un listado de opciones por medio del cual el jugador podrá tener acceso a las funcionalidades de la aplicación, como también podrá tomar decisiones en cuanto al tipo de carácter a utilizar y otras funciones opcionales.
- El software deberá permitir a los usuarios eliminar su cuenta de sesión del juego.
- El software desplegará información referente al medio ambiente, de acuerdo a la realidad de nuestro país, en cuanto al alto grado de deforestación, manejo



adecuado de desechos sólidos, contaminación de aguas, contaminación del aire y extinción de especies.

- La aplicación admitirá el modo de juego One Player en dos perspectivas, primera y tercera persona.
- El videojuego categorizará a los personajes por buenos y malos, cada uno tendrá sus respectivas animaciones<sup>88</sup>.
- Los personajes podrán interactuar con el jugador, mediante diálogos predefinidos en el juego.
- En las escenas en las cuales el software transmita información medio ambiental al jugador, ya sea por medio de un video o una narrativa, el jugador no podrá interactuar.
- Los personajes permitidos por el software serán tres, Eco-ambiente, Latita y Superplástico, de los cuales el jugador podrá escoger el que mejor le simpatice como su súper héroe.
- La aplicación deberá permitir al jugador personalizar configuraciones de juego, entre estas: resolución de pantalla, ventana o pantalla completa, velocidad de juego, combinación de entradas de interfaz de usuario.
- La aplicación deberá presentar los controles que el jugador podrá utilizar para interactuar con el juego.
- El software deberá permitirle al jugador cargar el avance que previamente haya sido guardado, de no existir un juego guardado, el software le indicaría al jugador que no existe ninguna sesión guardada. (En el caso que no se halla guardado ningún avance de parte del jugador, desde que haya iniciado la sesión.)

### 3.3.2. Requerimientos no funcionales

- El software deberá funcionar en las versiones de Windows por ser esta la plataforma el sistema operativo con el cual cuenta se van a implementar.
- Se permitirá al usuario ciertas actividades no relacionadas con la concientización del medioambiente, propias de centros escolares como por ejemplo: la interacción con otros caracteres virtuales, etc. Ofreciendo el sentimiento de realidad en niveles que se lleven a cabo en centros escolares virtuales incluidos en el juego.

---

<sup>88</sup> Ejemplo de animaciones: caminar, saltar, correr, hablar, volar.



- Por la naturaleza del software no será escalable, es decir no se aumentará el tamaño de los componentes del mismo ni se desarrollará nuevas funciones. Sin embargo, se podrían actualizar la información estadística medio ambiental por medio de actualizaciones a un archivo de texto incluido.
- Debido a la naturaleza del proyecto, el software no dispondrá de soporte técnico por parte del equipo de desarrollo, por lo que se debe proporcionar adjunto al software, un manual de usuario completo y efectivo, que incluya guías de instalación, configuración, operación, de forma que se pueda reducir la necesidad de soporte técnico personalizado.
- Para facilitar la instalación y operación del video juego se requiere la presentación de la siguiente documentación:
  - Manual técnico del software
  - Demo automatizado que muestre al usuario el manejo de dicho juego y le permita familiarizarlos con los controles de manipulación.
  - Plan de implantación.



# **CAPITULO IV: ANÁLISIS DEL SOFTWARE**



## 4. ANALISIS DEL SOFTWARE.

### 4.1. FUNCIONES DEL SOFTWARE

Las funciones del software definen lo que este debe hacer, por lo que se deberán identificar y listar en grupos cohesivos y lógicos.

Las funciones deben clasificarse en grupos dependiendo de su existencia dentro del sistema si pueden ser observadas por los actores o no; de acuerdo a lo establecido en la siguiente tabla.

| Tipo           | Función   |
|----------------|---|
| <b>Externa</b> | Debe realizarse, y el usuario debería saber que se ha realizado.  |
| <b>Interna</b> | Debe realizarse, aunque no es visible para los usuarios. Esto se aplica a muchos servicios técnicos subyacentes, como guardar información en un mecanismo persistente de almacenamiento. Las funciones ocultas a menudo se omiten (erróneamente) durante el proceso de obtención de los requerimientos. |

Cuadro 4.1: Clasificación de las funciones del sistema.

#### 4.1.1. Funciones de Inicialización

| Ref. # | Función  | Categoría |
|--------|--|-----------|
| R 1.1  | Cargar recursos de cada nivel                    | Interna   |
| R 1.2  | Cargar recursos de inicio de la aplicación       | Interna   |
| R 1.3  | Cargar recursos de opciones principales de juego | Interna   |

Cuadro 4.2: Funciones de Inicialización



### 4.1.2. Funciones de Entrada

| Ref. # | Función   | Categoría |
|--------|---|-----------|
| R 2.1  | Crear cuenta de jugador   | Externa   |
| R 2.2  | Iniciar sesión  | Externa   |
| R 2.3  | Eliminar cuenta de jugador  | Externa   |
| R 2.4  | Gestionar opciones principales de videojuego.                         | Externa   |
| R 2.5  | Configurar controles de interfaz humana.                              | Externa   |
| R 2.6  | Configurar velocidad  | Externa   |
| R 2.7  | Configurar modo de video  | Externa   |
| R 2.8  | Cargar juego almacenado   | Externa   |
| R 2.9  | Seleccionar personaje   | Externa   |
| R 2.10 | Controlar personaje   | Externa   |
| R 2.11 | Gestionar objetos recolectados, comprados y especiales de videojuego. | Externa   |
| R 2.12 | Gestionar Sub opciones de cada nivel.                                 | Externa   |
| R 2.13 | Seleccionar temática de información medioambiental                    | Interna   |

Cuadro 4.3: Funciones de Entrada

### 4.1.3. Funciones de Procesamiento

| Ref. # | Función  | Categoría |
|--------|--|-----------|
| R 3.1  | Almacenar datos de usuarios y avances de videojuego. | Externa   |
| R 3.2  | Validar usuario                                      | Externa   |
| R 3.3  | Actualizar valores de recursos de videojuego         | Interna   |
| R 3.4  | Gestionar sonidos                                    | Externa   |
| R 3.5  | Gestionar video                                      | Externa   |
| R 3.6  | Gestionar texturas 2D                                | Interna   |
| R 3.7  | Aplicar efectos especiales                           | Externa   |
| R 3.8  | Gestionar información medioambiental                 | Externa   |
| R 3.9  | Gestionar ecopuntos del jugador                      | Externa   |
| R 3.10 | Gestionar el avance del juego                        | Externa   |
| R 3.11 | Gestionar nivel de vida                              | Externa   |
| R 3.12 | Procesar inteligencia artificial                     | Externa   |
| R 3.13 | Detectar colisiones con objetos del mundo virtual    | Externa   |
| R 3.14 | Guardar Sesión                                       | Externa   |

Cuadro 4.4: Funciones de Procesamiento



#### 4.1.4. Funciones de Salida

| Ref. # | Función  | Categoría |
|--------|--|-----------|
| R 4.1  | Mostrar ecopuntos  | Externa   |
| R 4.2  | Mostrar nivel de vida  | Externa   |
| R 4.3  | Mostrar sesiones almacenadas   | Externa   |
| R 4.4  | Mostrar bitácora de juegos para cada jugador.  | Externa   |
| R 4.5  | Mostrar objetos recolectados, comprados y especiales.  | Externa   |
| R 4.6  | Demostrar poderes de personajes  | Externa   |
| R 4.7  | Generar en pantalla un certificado digital como estímulo a los jugadores que hayan terminado todos los niveles | Externa   |
| R 4.8  | Reproducir sonidos de diálogo de interacción con el mundo virtual  | Externa   |
| R 4.9  | Reproducir video   | Externa   |
| R 4.10 | Mostrar información medioambiental   | Externa   |
| R 4.11 | Pausar juego   | Externa   |
| R 4.12 | Dibujar pantalla de espera para siguiente nivel  | Externa   |
| R 4.13 | Dibujar valores de recursos de opciones principales de video juego   | Externa   |
| R 4.14 | Dibujar escenarios de cada nivel   | Externa   |
| R 4.15 | Mostrar Demo   | Externa   |

Cuadro 4.5: Funciones de Salida.

#### 4.1.5. Funciones de finalización.

| Ref. # | Función   | Categoría |
|--------|---|-----------|
| R 5.1  | Destruir recursos de cada nivel de juego            | Interna   |
| R 5.2  | Destruir recursos de introducción                   | Interna   |
| R 5.3  | Destruir recursos de opciones principales del juego | Interna   |
| R 5.4  | Salir de Videojuego.                                | Externa   |

Cuadro 4.6: Funciones de finalización

## 4.2. MODELO DE CASOS DE USO

Aclaración de términos<sup>89</sup>:

**Sistema:** a) Es regularmente un grupo de elementos interdependientes o interrelacionados que conforman un todo unificado. b) Un grupo de cuerpos bajo la influencia de fuerzas externas. c) Un grupo de objetos o fuerzas naturales relacionados entre sí.

<sup>89</sup> Según el Diccionario Merriam Webster de la lengua Inglesa <http://www.merriam-webster.com/dictionary/> y el Bureau of Justice Assistance of the United States (Oficina de Asistencia Jurídica del Gobierno de Los Estados Unidos) [http://www.ojp.usdoj.gov/BJA/evaluation/glossary/glossary\\_i.htm](http://www.ojp.usdoj.gov/BJA/evaluation/glossary/glossary_i.htm)



**Software:** Es algo utilizado o asociado con hardware. a) Agrupaciones de programas, procedimientos y documentación asociada con un sistema y específicamente con un sistema computacional. b) Materiales utilizados para equipo audiovisual.

**Sistema Informático:** Un sistema de presentación, almacenamiento y colección organizada de datos implementado para la toma de decisiones, reporte de progresos y para programas de planeación y evaluación. Puede ser manual, computarizado o una mezcla de ambos.

Si bien, según las definiciones anteriores, un software es un sistema computarizado; un sistema informático puede ser un sistema computarizado o manual. Por lo tanto el sistema aquí descrito no es un sistema informático, no obstante, cuando se haga mención de la palabra 'software' o 'sistema', se estará haciendo referencia al sistema solución para el programa actual de Educación Ambiental Nacional.

El software permite al usuario interactuar de forma directa con el mundo virtual que se le presenta al jugador. Muchas funciones son directamente dependientes de las entradas y respuestas del usuario, otras son acciones derivadas de dichas interacciones. Para visualizar mejor esta interacción jugador-videojuego, y debido a la naturaleza orientada a objetos de este proyecto, se hará uso de las técnicas de diseño UML 2.0, basadas en las técnicas promovidas por el experto en UML Craig Larman<sup>90</sup>.

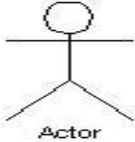

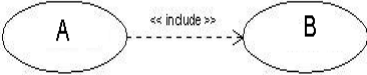
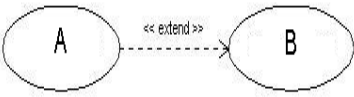
La simbología utilizada para la creación de los diagramas de caso de uso es la que se presenta en el cuadro 4.7.

---

<sup>90</sup> UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos, Craig Larman. Pearson Prentice Hall 2003





| SIMBOLO   | SIGNIFICADO  |
|---|--|
| <p style="text-align: center;"><b>ACTOR</b></p>        | <p>Un actor es un usuario del sistema. Esto incluye usuarios humanos y otros sistemas computacionales. Un actor usa un Caso de Uso para ejecutar una porción de trabajo de valor para el negocio. El conjunto de casos de uso al que un actor tiene acceso define rol en el sistema y el alcance de su acción.</p> |
| <p style="text-align: center;"><b>CASO DE USO</b></p>  | <p>Un Caso de Uso es una representación de una unidad discreta de trabajo realizada por un usuario (u otro sistema) usando el sistema en operación. Se ejecuta en su totalidad o no se ejecuta nada, devolviendo algo de valor al usuario</p>  |
| <p style="text-align: center;"><b>&lt;&lt;ESTEREOTIPO&gt;&gt;</b></p>   | <p>Se usan para definir tipos de objetos o elementos, significado de relación entre casos de uso que pueden ser de inclusión, extensión o de uso.</p>  |
| <p style="text-align: center;"><b>INCLUIR</b></p>     | <p>Un Caso de Uso puede incluir (&lt;&lt;include&gt;&gt;) la funcionalidad de otro como parte de su procesamiento normal, se asume que los casos de uso incluidos se llamarán cada vez que se ejecute el camino base. El caso de uso A incluye la funcionalidad de B.</p>  |
| <p style="text-align: center;"><b>EXTENSIÓN</b></p>  | <p>Un Caso de Uso puede extender (&lt;&lt;extend&gt;&gt;) el comportamiento de otro Caso de Uso; típicamente cuando ocurren situaciones excepcionales. El caso de uso A extiende la funcionalidad del caso de uso B.</p>   |

Cuadro 4.7: Simbología Casos de uso

A continuación, se presenta el diagrama de casos de uso para visualizar la interacción jugador-videojuego.



### 4.2.1. Diagrama de casos de uso

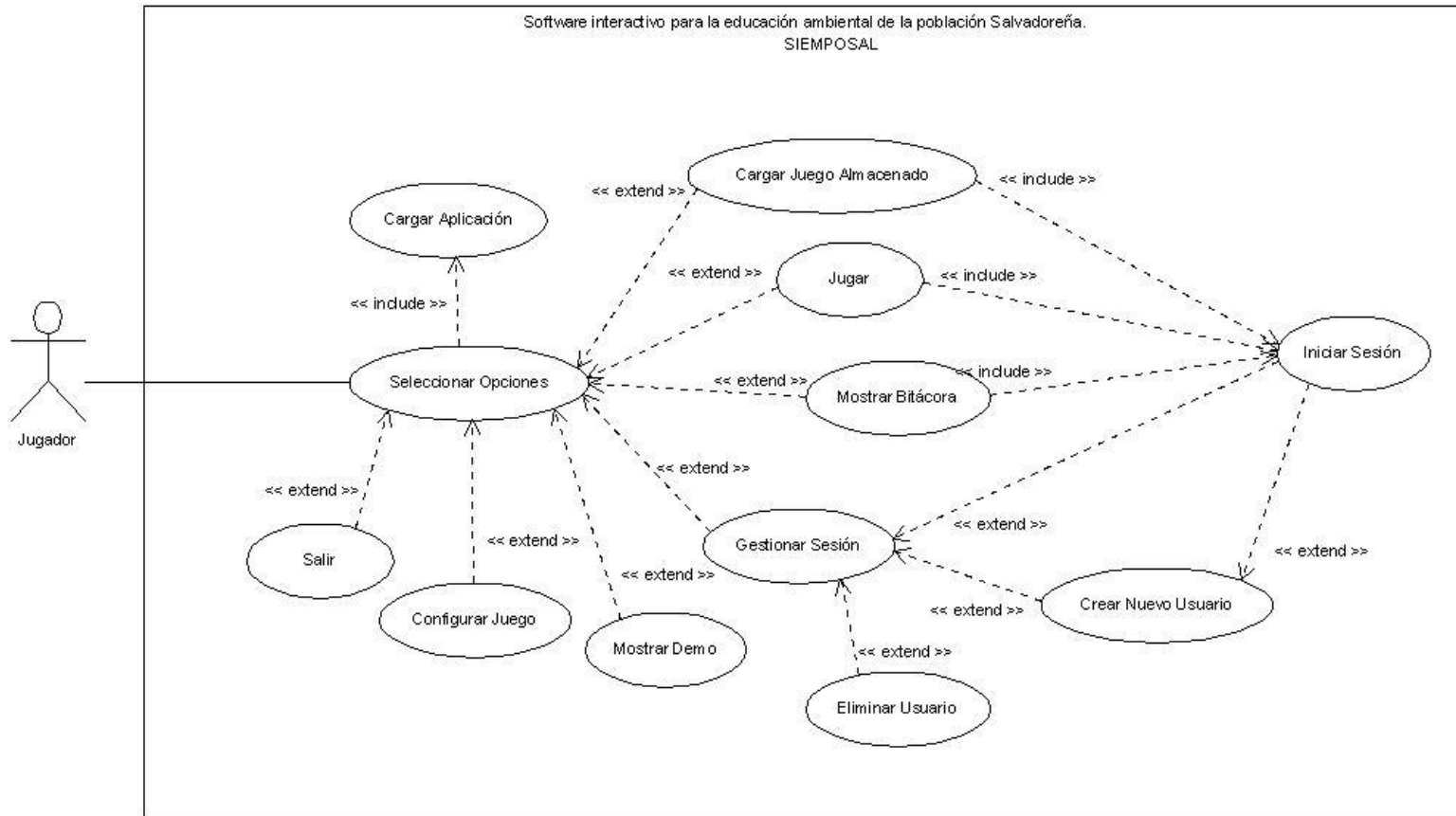


Figura 4.1: Diagrama de Contexto, Caso de uso nivel 0

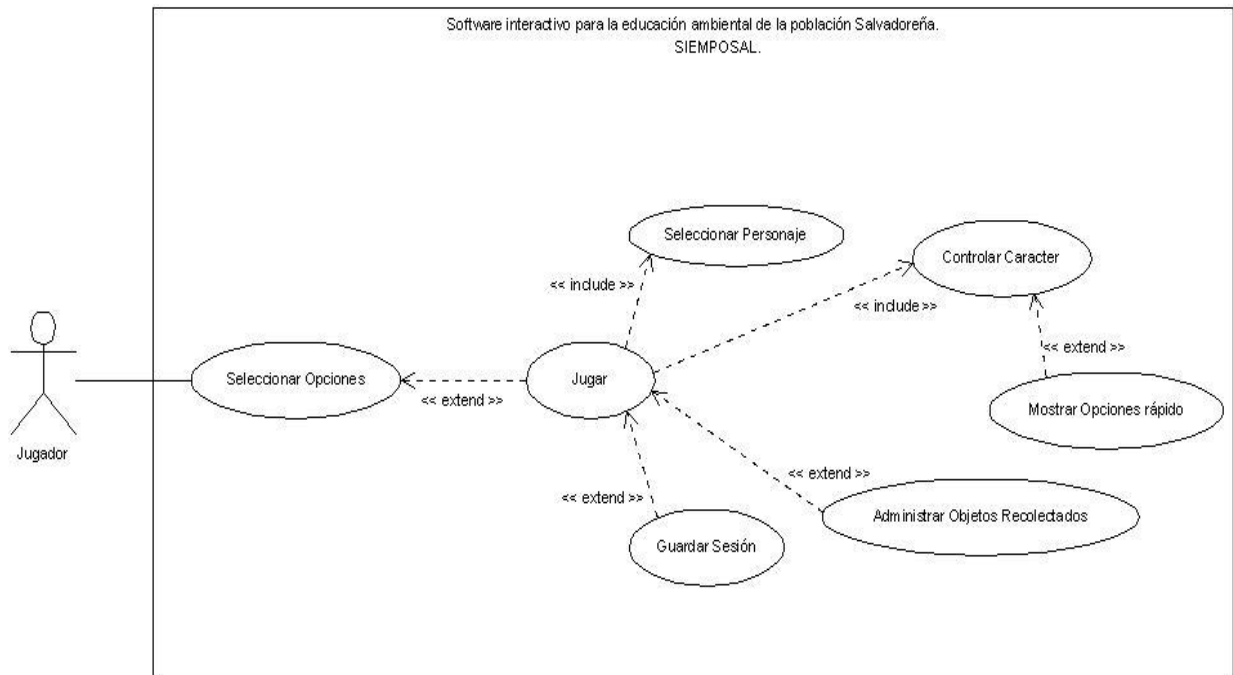


Figura 4.2: Caso de uso nivel 1 Jugar

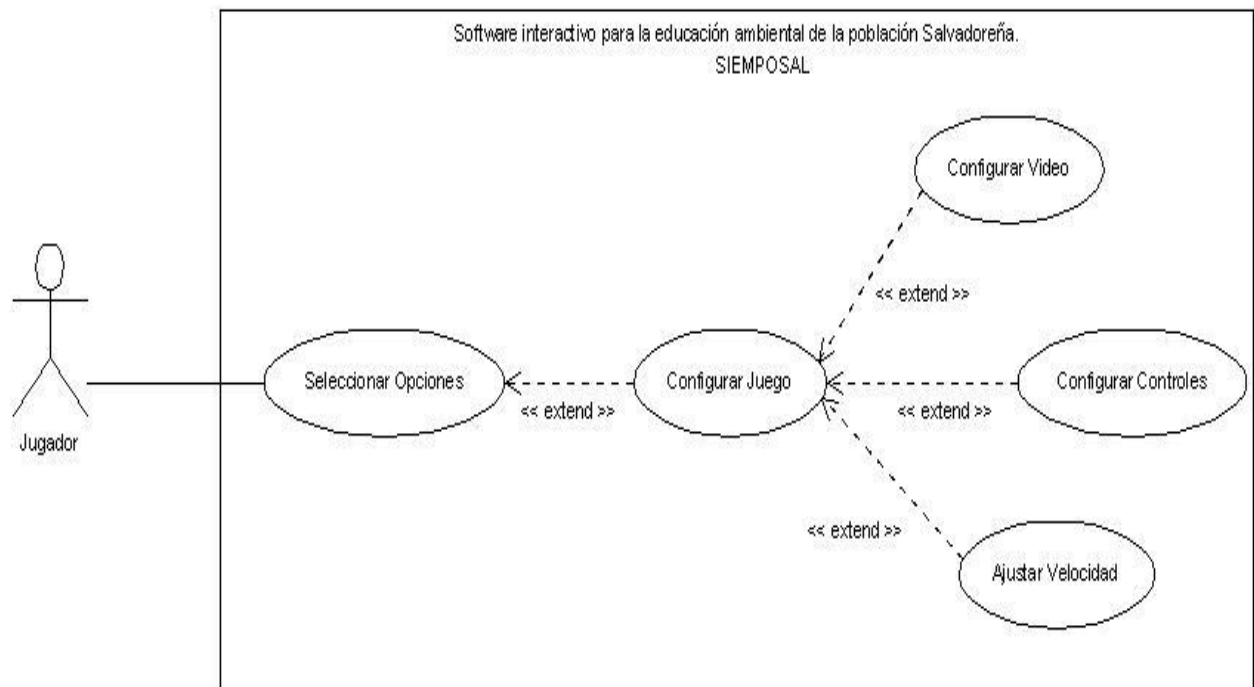


Figura.4.3: Caso de uso nivel 1 configurar juego



### 4.2.2. Clasificación de los Casos de Uso del Diagrama de Contexto.

| Clasificación | Caso de Uso             | Justificación   |
|---------------|-------------------------|---|
| Alta          | Cargar Aplicación       | Corresponde al criterio de clasificación más alto ya que es aquí donde se cargan los recursos básicos del videojuego y se configura el entorno en tres dimensiones.   |
|               | Iniciar Sesión          | Es necesario este caso de uso ya que es necesario el inicio de una sesión para autenticar al usuario y guardar el avance obtenido durante una partida de juego.   |
|               | Crear Nuevo Usuario     | Indispensable para identificar a cada jugador y llevar un seguimiento individual de los avances del videojuego.   |
|               | Seleccionar Opción      | Es el caso de uso a través del cual se lleva a cabo toda “transacción” con el juego de video, es el “portal” a través del cual se tiene que pasar para llegar a cualquier función del software.   |
|               | Jugar                   | Es de gran importancia pues sin este caso de uso no se llevaría a cabo el objetivo principal de este software, que es concientizar a los usuarios en la temática del medioambiente.   |
| Mediana       | Mostrar Demo            | Permite al jugador familiarizarse con los controles. Funciona como un tutorial en vivo, permitiendo al jugador agilizar el aprendizaje del videojuego y por lo tanto la concientización ambiental, que es el objetivo primario del videojuego como sistema. |
|               | Cargar Juego Almacenado | Permite que el usuario tenga un seguimiento en su aprendizaje, y esto contribuye al logro del objetivo perseguido.  |
|               | Mostrar Bitácora        | Efecto mínimo en la arquitectura, no obstante, permite a los usuarios tener una perspectiva de su avance, logros y evaluación del conocimiento adquirido sobre el tema del medioambiente a través del videojuego como sistema.                              |
|               | Configurar Juego        | Permite que el usuario pueda cambiar las opciones/funciones por defecto del videojuego, esto afecta el entorno del videojuego al personalizarse de acuerdo a las preferencias de los jugadores.   |
| Baja          | Eliminar Usuario        | Es un caso de uso opcional, ayudará al mantenimiento de cuentas de usuario.   |
|               | Salir                   | Efecto mínimo en la arquitectura del juego. Permite terminar la aplicación.   |

Cuadro 4.8: Clasificación de casos de uso



### 4.2.3. Descripción de Casos de Uso.

| Caso de Uso:       | Cargar Aplicación (Alto Nivel)  | Código: CU-001 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario-esencial   |                |
| <b>Descripción</b> | Este caso de uso comienza desde el momento en que el jugador inicia el Videojuego. En esta fase se cargan todos los recursos necesarios para el funcionamiento del software, también presenta información introductoria, en la que se dan a conocer las instituciones para las cuales se está elaborando el software. |                |

| Caso de Uso:                 | Cargar Aplicación (Expandido)   | Código: CU-001 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Cargar los recursos necesarios para inicializar las funciones del software.   |                |
| <b>Resumen:</b>              | El jugador inicia el juego, el software prepara el entorno para el mundo virtual y presenta una introducción en la cual se despliegan los logos de las instituciones a cargo del programa actual de Educación Ambiental y logotipo de dicho programa. |                |
| <b>Referencias cruzadas:</b> | R1.2, R 5.2   |                |

#### Curso Normal de los Eventos

| Acción de los actores   | Respuesta del sistema  |
|---|--|
| 1. Este caso de uso comienza cuando el jugador ejecuta el juego |  |
|   | 2. El software carga los recursos necesarios para la configuración inicial del sistema.        |
|   | 3. El software carga el despliegue de la información de introducción.                          |
| 4. El jugador visualiza el despliegue de la introducción.       |  |
|   | 5. Terminada la introducción se llama a <b>“Seleccionar Opciones”</b> . Ver caso de uso CU-002 |

#### Curso Alterno A:

|  |
|--|
| 4. El jugador decide saltar la presentación de la introducción y le indica al sistema su decisión.                             |
| 5. El software cancela el proceso de “Desplegar Introducción” y llama a <b>“Seleccionar Opciones”</b> . Ver caso de uso CU-002 |



| Caso de Uso:       | Seleccionar Opciones (Alto Nivel)  | Código: CU-002 |
|--------------------|--|----------------|
| <b>Actores:</b>    | Jugador  |                |
| <b>Tipo:</b>       | Primario - Esencial  |                |
| <b>Descripción</b> | Consiste en presentar al usuario varias opciones a través de las cuales el usuario puede acceder a las funciones del software. |                |

| Caso de Uso:                 | Seleccionar Opciones (Expandido)  | Código: CU-002 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al jugador navegar entre las diferentes opciones que el software ofrece, y poder tener acceso a sus funciones.   |                |
| <b>Resumen:</b>              | El jugador decide que acción tomar, para empezar a interactuar con el software y así Iniciar una partida de juego, configurar el juego, cargar partidas de juego almacenadas previamente, visualizar la bitácora de avance de cada jugador, gestionar sesión, acceder a ver un demo y salir del videojuego. |                |
| <b>Referencias cruzadas:</b> | R 1.3, R 2.4, R 4.14, R 5.3   |                |

**Curso Normal de los Eventos**

| Acción de los actores   | Respuesta del sistema  |
|---|--|
| 1. Este caso de uso comienza cuando el jugador cancela el despliegue de la introducción o al terminar de desplegarse la introducción.   |  |
|   | 2. El sistema proporciona las opciones que permiten: Jugar, Configurar el videojuego, Cargar Juego Almacenado, Mostrar Bitácora, Gestionar Sesión, Mostrar Demo y Salir. |
| 3. El jugador selecciona la opción deseada. <ul style="list-style-type: none"> <li>a. Si selecciona Jugar, ver Caso de Uso: <b>"Jugar"</b>, CU-003.</li> <li>b. Si selecciona Configurar Juego, ver Caso de Uso: <b>"Configurar juego"</b>, CU-009.</li> <li>c. Si selecciona Cargar Juego Almacenado, ver Caso de Uso: <b>"Cargar Juego Almacenado"</b>, CU-013</li> <li>d. Si selecciona Mostrar Bitácora, ver Caso de Uso: <b>"Mostrar Bitácora"</b>, CU-014.</li> <li>e. Si selecciona Gestionar Sesión, ver Caso de Uso: <b>"Gestionar Sesión"</b>, CU-015.</li> <li>f. Si selecciona Mostrar Demo, ver Caso de Uso: <b>"Mostrar Demo"</b>, CU-019.</li> <li>g. Si selecciona Salir, ver Caso de Uso: <b>"Salir"</b>, CU-020.</li> </ul> | 4. El sistema carga los recursos de la opción seleccionada por el usuario.   |



| Caso de Uso:       | Jugar (Alto Nivel)  | Código: CU-003 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario - Esencial   |                |
| <b>Descripción</b> | Para jugar el usuario debe previamente iniciar una sesión, para la cual debe haber creado una cuenta de usuario. Una vez autenticado el usuario puede iniciar una partida de juego. Cada partida de juego será almacenada automáticamente al finalizar el nivel o al pasar por ciertos obstáculos o “check points”. |                |

| Caso de Uso:                 | Jugar (Expandido)   | Código: CU-003 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al jugador iniciar una partida donde no existe ningún avance previo  |                |
| <b>Resumen:</b>              | El jugador inicia una sesión, de no existir una cuenta para determinado jugador, se le da la opción de crear una cuenta. Después de haber iniciado sesión el jugador puede seleccionar su carácter favorito, el cual determinara los diferentes recursos y niveles que se le presentarán. El juego permitirá almacenar el avance del jugador. |                |
| <b>Referencias cruzadas:</b> | R 1.1, R 2.13, R 3.3, R 3.4, R 3.5, R 3.6, R 3.7, R 3.8, R 3.9, R 3.10, R 3.12, R 3.13, R 3.14, R 4.1, R 4.2, R 4.3, R 4.7, R 4.8, R 4.9, R 4.10, R 4.11, R 4.12, R 4.13, R 4.15, R 5.1   |                |

**Curso Normal de los Eventos**

| Acción de los actores  | Respuesta del sistema  |
|--|--|
|  | 1. El sistema actualiza y dibuja los recursos del nivel asociado con el personaje seleccionado |
| 2. El jugador inicia la interacción con el videojuego, (Jugar) y al mismo tiempo podrá seleccionar entre las siguientes opciones. <ul style="list-style-type: none"> <li>a. Si el jugador desea que el carácter camine, corra, salte, etc., se ejecutará <b>“Controlar Carácter”</b>. Ver caso de uso CU-005.</li> <li>b. Si el jugador desea tener acceso a las opciones de menú rápido, se ejecutará <b>“Mostrar Opciones Rápidas”</b>. Ver caso de uso CU-006. T</li> <li>c. Si el jugador desea realizar alguna acción con los objetos recolectados, ya sea ubicarlos en sus lugares respectivos si se trata de desechos sólidos, o consumir alimentos, o tomar objetos para cumplir con las misiones asignadas, deberá ejecutar el caso de uso “Administrar Objetos Recolectados”. Ver caso de uso CU-007.</li> </ul> |  |



|  |  |
|--|--|
| 3. Si el jugador mediante la interacción a avanzado hasta determinado checkPoint |  |
|  | 4. El sistema ejecuta la función para “ <b>Guardar Sesión</b> ”, Ver caso de uso CU-008. |
| 5. El jugador ha pasado todas las pruebas de un nivel (finalizado el nivel)      |  |
|  | 6. El sistema ejecuta la función “ <b>CargarRecursos</b> ”                               |

| Caso de Uso:       | Seleccionar Carácter (Alto Nivel)   | Código: CU-004 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario - Esencial   |                |
| <b>Descripción</b> | Este caso de uso se ejecuta cuando el usuario ha seleccionado la opción “Jugar” desde las opciones principales. Se le presentan diferentes caracteres que este podrá controlar a lo largo de su sesión de juego. Previo a seleccionar un carácter, el jugador debe haber iniciado una sesión. |                |

| Caso de Uso:                 | Seleccionar Carácter (Extendido)  | Código: CU-004 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al jugador seleccionar de entre los caracteres disponibles el que utilizará a través de la sesión de juego iniciada.   |                |
| <b>Resumen:</b>              | El jugador tendrá la oportunidad de seleccionar un carácter sobre el cual tendrá el control durante el juego. Cada carácter tendrá una variante en la historia que se desenlazara a medida que el usuario avance. |                |
| <b>Referencias cruzadas:</b> | R 2.9   |                |

| Curso Normal de los Eventos           |  |
|---------------------------------------|--|
| Acción de los actores                 | Respuesta del sistema  |
|                                       | 1. El sistema del videojuego presenta al usuario tres caracteres interactivos entre los cuales puede seleccionar uno |
| 2. El jugador Selecciona un carácter. |  |
|                                       | 3. El sistema carga los recursos del carácter seleccionado y la historia asociada a este.                            |
|                                       | 4. El sistema dibuja los recursos del carácter seleccionado y la historia asociada a este                            |





|  |
|--|
| <b>Curso Alterno A:</b>  |
| 2. El jugador decide cancelar la función de Seleccionar Carácter.                      |
| 3. El sistema retorna al usuario a las opciones principales.                           |
| <b>Curso Alterno B:</b>  |
| 1. El sistema verifica que el jugador no ha iniciado sesión.                           |
| 2. El sistema ejecuta el caso de uso <b>“Iniciar Sesión”</b> , ver caso de uso CU-016. |
| 3. El sistema ejecuta los pasos del curso normal de este caso de uso.                  |

| Caso de Uso:       | Controlar Carácter (Alto Nivel)  | Código: CU-005 |
|--------------------|--|----------------|
| <b>Actores:</b>    | Jugador  |                |
| <b>Tipo:</b>       | Primario - Esencial  |                |
| <b>Descripción</b> | Este caso de uso ocurre durante toda la sesión de juego de un determinado nivel de la aplicación 3D, le permite al usuario realizar acciones como correr, saltar, caminar, etc., e interactuar con los personajes y mundo virtual que comprenden la historia del videojuego, a través de la gestión de entradas del usuario predefinidas en las funciones de configuración de control de interfaz humana del software. |                |

| Caso de Uso:                 | Controlar Carácter (Extendido)  | Código: CU-005 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al usuario controlar el carácter seleccionado e interactuar con personajes y escenarios virtuales.   |                |
| <b>Resumen:</b>              | El jugador, a través de los controles de interfaz, podrá manipular al carácter e indicar las acciones que este tome en situaciones en donde se interactúa con personajes y mundo virtual. |                |
| <b>Referencias cruzadas:</b> | R 2.10, R 3.14, R 4.7   |                |



| Curso Normal de los Eventos  |   |
|--|---|
| Acción de los actores  | Respuesta del sistema   |
|  | 1. El sistema carga los recursos relacionados a controlar carácter.   |
| 2. El jugador realiza determinadas acciones a través de las cuales solicita al sistema, mediante sus acciones, que realice ciertos procesos entre estas pueden mencionarse: caminar, correr, saltar, hablar, entre otras acciones. |   |
|  | 3. El sistema del videojuego “escucha” las entradas de datos o acciones ejecutadas por el jugador, mediante los controles de interfaz humana.   |
|  | 4. El sistema filtra los eventos de entrada de datos dependiendo del contexto en que se encuentra y ejecuta la(s) función(es) correspondientes. |
| 5. El jugador recibe las respuestas del software en función de sus acciones con el carácter.   |   |
| Curso Alternativo A:   |   |
| 1. El jugador no realiza ninguna acción ni solicitud al sistema  |   |
| 2. El sistema del videojuego “escucha” los eventos de entrada de datos del usuario.  |   |



| Caso de Uso:       | Mostrar Opciones Rápidas (Alto Nivel)  | Código: CU-006 |
|--------------------|--|----------------|
| <b>Actores:</b>    | Jugador  |                |
| <b>Tipo:</b>       | Secundario - Opcional  |                |
| <b>Descripción</b> | Este caso de uso se lleva a cabo cuando el jugador decide realizar ciertas acciones durante el juego. Entre las acciones entre las que puede seleccionar están: pausar juego, mostrar diferentes sub opciones. |                |

| Caso de Uso:                 | Mostrar Opciones Rápidas (Expandido)  | Código: CU-006 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al jugador que en ciertos momentos del juego tenga acceso a un conjunto de sub opciones, para ejecutar acciones durante el juego, entre estas se puede mencionar: pausar una partida de juego, desplegar sub opciones.   |                |
| <b>Resumen:</b>              | El jugador en función de las acciones previamente realizadas o del momento en el q se encuentre puede pausar el videojuego, esta opción le permitirá pasar el videojuego a un estado de espera, hasta que el jugador reanuda la partida, podrá tener acceso a las sub opciones. |                |
| <b>Referencias cruzadas:</b> | R 2.12  |                |

**Curso Normal de los Eventos**

| Acción de los actores                         | Respuesta del sistema   |
|---|---|
|   | 1. El sistema carga los recursos relacionados a Mostrar Opciones Rápidas.   |
|   | 2. El sistema muestra una interfaz en la que le muestra una lista de opciones entre las que el usuario puede elegir.    |
| 3. El jugador selecciona la opción que desea. |   |
|   | 4. El sistema ejecuta la acción solicitada por el jugador y retorna el control a <b>"Jugar"</b> ver caso de uso CU-003. |



|                     |   |                       |
|---------------------|---|-----------------------|
| <b>Caso de Uso:</b> | <b>Administrar Objetos Recolectados (Alto Nivel)</b>  | <b>Código: CU-007</b> |
| <b>Actores:</b>     | Jugador   |                       |
| <b>Tipo:</b>        | Primario - Esencial   |                       |
| <b>Descripción:</b> | Este caso de uso se lleva a cabo cuando el jugador decide administrar (comer, utilizar, depositar, manipular, etc.) |                       |

|                              |  |                       |
|------------------------------|--|-----------------------|
| <b>Caso de Uso:</b>          | <b>Administrar Objetos Recolectados (Expandido)</b>  | <b>Código: CU-007</b> |
| <b>Actores:</b>              | Jugador  |                       |
| <b>Propósito:</b>            | Permitir al jugador administrar los objetos recolectados, a lo largo de los diferentes niveles.  |                       |
| <b>Resumen:</b>              | Permite al usuario iterar sobre los diferentes objetos recolectados, como comestibles, objetos especiales, desechos sólidos recolectados, entre otros. |                       |
| <b>Referencias cruzadas:</b> | R 2.11, R 4.6  |                       |

| <b>Curso Normal de los Eventos</b>   |  |
|--|--|
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>   |
| 1. Este caso de uso inicia cuando el jugador desea ver los objetos recolectados o cuando se encuentre en ciertas posiciones en el mundo virtual. |  |
|  | 2. El sistema muestra los objetos recolectados, dependiendo del contexto, al mismo tiempo muestra los controles para la navegación y selección de estos. |
| 3. El jugador navega por los objetos y selecciona los objetos para un uso en particular.   |  |
|  | 4. El sistema realiza la acción asociada con el objeto.  |
| 5. El jugador, indica al sistema que desea finalizar la administración de los objetos recolectados.  |  |
|  | 6. El sistema vuelve el control a <b>“Jugar”</b> ver caso de uso CU-003.   |



|                     |  |                       |
|---------------------|--|-----------------------|
| <b>Caso de Uso:</b> | <b>Guardar Sesión (Alto Nivel)</b>   | <b>Código: CU-008</b> |
| <b>Actores:</b>     | Jugador  |                       |
| <b>Tipo:</b>        | Primario - Esencial  |                       |
| <b>Descripción:</b> | Este caso de uso se ejecuta cuando el jugador, a través de las acciones en el videojuego, alcanza un grado de avance, es decir, ha superado obstáculos o ha finalizado un nivel. |                       |

|                              |  |                       |
|------------------------------|--|-----------------------|
| <b>Caso de Uso:</b>          | <b>Guardar Sesión (Extendido)</b>  | <b>Código: CU-008</b> |
| <b>Actores:</b>              | Jugador  |                       |
| <b>Propósito:</b>            | Guardar las sesiones de usuario, cada vez que este ha pasado por checkPoint (Obstáculos) o ha finalizado el nivel.   |                       |
| <b>Resumen:</b>              | El sistema, a través de las acciones y avance del usuario, determina si éste ha pasado por obstáculos en el videojuego y ha cumplido los requisitos para pasar al siguiente nivel. |                       |
| <b>Referencias cruzadas:</b> | R 3.1, R 3.11, R 3.15  |                       |

| <b>Curso Normal de los Eventos</b> |  |
|------------------------------------|--|
| <b>Acción de los actores</b>       | <b>Respuesta del sistema</b>   |
|                                    | 1. El sistema guarda el avance obtenido por el jugador.  |
|                                    | 2. Si el jugador, ha pasado y superado los obstáculos del videojuego, el sistema le indica que se ha guardado el avance.   |
|                                    | 3. Si el jugador ha finalizado nivel, el sistema muestra una interfaz en la que le indica que avanzará al siguiente nivel y carga los recursos correspondientes al nivel inmediato superior. |



| Caso de Uso:       | Configurar Juego (Alto Nivel)   | Código: CU-009 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Secundario  |                |
| <b>Descripción</b> | Este caso de uso se lleva a cabo cuando el jugador ha indicado al sistema del videojuego que desea hacer ciertas configuraciones, este le presenta una lista de opciones de configuración las cuales son: video, controles y velocidad. |                |

| Caso de Uso:  | Configurar Juego (Extendido)   | Código: CU-009 |
|---|--|----------------|
| <b>Actores:</b>   | Jugador  |                |
| <b>Propósito:</b>   | Permitir al usuario modificar y personalizar diferentes opciones del videojuego.                                   |                |
| <b>Resumen:</b>   | El jugador tiene acceso a las configuraciones de modo de video, configuraciones de controles y velocidad de juego. |                |
| <b>Referencias cruzadas:</b>  | R 2.7, R 2.5, R 2.6  |                |
| <b>Curso Normal de los Eventos</b>  |  |                |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                |
|   | 1. El sistema del videojuego presenta al usuario las opciones de configuración para el videojuego                  |                |
| 2. El jugador Selecciona la opción que desea configurar: <ul style="list-style-type: none"> <li>a. Si selecciona Configurar Video, ver Caso de Uso: <b>“Configurar Video”</b>, CU-010.</li> <li>b. Si selecciona Configurar Controles, ver Caso de Uso: <b>“Configurar Controles”</b>, CU-011.</li> <li>c. Si selecciona Ajustar Velocidad, ver Caso de Uso: <b>“Ajustar Velocidad”</b>, CU-012.</li> </ul> |  |                |
|   | 3. El sistema carga los recursos de la opción seleccionada por el usuario.   |                |
| <b>Curso Alterno A:</b>   |  |                |
| 2. El jugador no selecciona ninguna opción y sale sin realizar ninguna configuración al videojuego.   |  |                |
| 3. El sistema retorna al usuario a las opciones principales.  |  |                |



| Caso de Uso:       | Configurar Video (Alto Nivel)  | Código: CU-010 |
|--------------------|--|----------------|
| <b>Actores:</b>    | Jugador  |                |
| <b>Tipo:</b>       | Secundario   |                |
| <b>Descripción</b> | Este caso de uso lleva a cabo cuando el jugador ha indicado al sistema del videojuego que desea configurar el video. |                |

| Caso de Uso:  | Configurar Video (Extendido)  | Código: CU-010 |
|---|---|----------------|
| <b>Actores:</b>   | Jugador   |                |
| <b>Propósito:</b>   | Permitir al usuario modificar el tamaño de la ventana   |                |
| <b>Resumen:</b>   | El jugador tiene acceso a las configuraciones de modo de video (pantalla completa o ventana) para personalizar el tamaño de esta. |                |
| <b>Referencias cruzadas:</b>  | R 2.7   |                |
| <b>Curso Normal de los Eventos</b>  |   |                |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>  |                |
|   | 1. El sistema del videojuego presenta las opciones a configurar.  |                |
| 2. El jugador Selecciona entre las Opciones.<br>a. Pantalla Completa<br>b. Ventana                  |   |                |
|   | 3. El sistema solicita confirmar el cambio de configuración de video.   |                |
| 4. Confirma el cambio de configuración de video.  |   |                |
|   | 5. Actualiza y muestra los cambios efectuados en la configuración de video.   |                |
| <b>Curso Alternativo A:</b>   |   |                |
| 2. El jugador no selecciona ninguna opción y sale sin realizar ninguna configuración al videojuego. |   |                |
| 3. El sistema retorna al usuario a las opciones principales.  |   |                |



| Caso de Uso:       | Configurar Controles (Alto Nivel)   | Código: CU-011 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Secundario  |                |
| <b>Descripción</b> | Este caso de uso lleva a cabo cuando el jugador ha indicado al sistema que desea personalizar la funcionalidad de los controles del videojuego. |                |

| Caso de Uso:                 | Configurar Controles (Extendido)  | Código: CU-011 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Permitir al jugador personalizar las funcionalidad de los controles de interfaz de entrada para realizar funcionalidades o acciones con el carácter, como por ejemplo: caminar correr, saltar, etc.   |                |
| <b>Resumen:</b>              | El jugador accede a la opción configurar controles, donde puede cambiar las funcionalidades de los controles de interfaz del dispositivo de entrada, para personalizar las funciones. Al realizar la reconfiguración, automáticamente estos controles quedan definidos para cada partida del juego. |                |
| <b>Referencias cruzadas:</b> | R 2.5   |                |

**Curso Normal de los Eventos**

| Acción de los actores   | Respuesta del sistema   |
|---|---|
|   | 1. El sistema del videojuego presenta los controles y las funciones asignadas por defecto. Estos pueden ser modificados por el usuario asignando nuevos valores a los controles de interfaz de entrada. (adelante, atrás, izquierda, derecha, saltar, correr, lanzar, recoger). |
|   | 2. El sistema presenta las opciones disponibles de configuración que pueden ser habilitadas para una función seleccionada.  |
| 3. El jugador selecciona la nueva configuración que desea asignar al control de interfaz, de acuerdo a sus preferencias y confirma su almacenamiento. |   |
|   | 4. El sistema guarda la configuración realizada y finaliza el caso de uso.  |

**Curso Alterno A:**

|   |
|---|
| 3. El jugador no selecciona ninguna función y sale del caso de uso. |
| 4. El sistema retorna al usuario a las opciones principales.        |





|                     |  |                       |
|---------------------|--|-----------------------|
| <b>Caso de Uso:</b> | <b>Configurar Velocidad (Alto Nivel)</b>   | <b>Código: CU-012</b> |
| <b>Actores:</b>     | Jugador  |                       |
| <b>Tipo:</b>        | Secundario   |                       |
| <b>Descripción</b>  | Este caso de uso lleva a cabo cuando el jugador ha indicado al sistema del videojuego que desea configurar la velocidad del video. |                       |

|   |  |                       |
|---|--|-----------------------|
| <b>Caso de Uso:</b>   | <b>Configurar Velocidad (Extendido)</b>  | <b>Código: CU-012</b> |
| <b>Actores:</b>   | Jugador  |                       |
| <b>Propósito:</b>   | Permitir al jugador personalizar la velocidad del juego.   |                       |
| <b>Resumen:</b>   | Este caso de uso se ejecuta cuando el jugador desea ajustar la velocidad del videojuego, en este caso de uso es donde puede cambiar la rapidez con que se realizan las funciones, estas son almacenadas para cada partida. |                       |
| <b>Referencias cruzadas:</b>  | R 2.6  |                       |
| <b>Curso Normal de los Eventos</b>  |  |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                       |
|   | 1. El sistema del videojuego presenta las velocidades de las que el jugador puede disponer.  |                       |
| 2. El jugador selecciona una nueva velocidad para el videojuego                             |  |                       |
|   | 3. El sistema actualiza la nueva velocidad para el videojuego.   |                       |
| <b>Curso Alternativo A:</b>   |  |                       |
| 2. El jugador no realiza ningún cambio de velocidad y decide cancelar configurar velocidad. |  |                       |
| 3. El sistema retorna a las opciones principales.   |  |                       |



| Caso de Uso:       | Cargar Juego Almacenado (Alto Nivel)  | Código: CU-013 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario - Esencial   |                |
| <b>Descripción</b> | Este caso de uso ocurre cuando el usuario (con una cuenta existente), ha seleccionado la opción "Cargar Juego Almacenado" desde las opciones principales. |                |

| Caso de Uso:  | Cargar Juego Almacenado (Extendido)   | Código: CU-013  |
|---|---|---|
| <b>Actores:</b>   | Jugador   |   |
| <b>Propósito:</b>   | Permitir al usuario la recuperación de una de diez partidas de juego, previamente almacenadas y permitirle continuar el juego desde el punto en el que fue almacenado en un momento dado. |   |
| <b>Resumen:</b>   | El sistema muestra las últimas diez partidas que se hallan almacenado. Se efectúa la carga de la partida que el usuario elija.  |   |
| <b>Referencias cruzadas:</b>  | R 2.8, R 3.11, R 4,4  |   |
| <b>Curso Normal de los Eventos</b>  |   |   |
| <b>Acción de los actores</b>  |   | <b>Respuesta del sistema</b>  |
|   |   | 1. El sistema del videojuego presenta al usuario los recursos de <b>"cargar juego almacenado"</b> , permitiéndole visualizar las últimas 10 partidas guardadas. |
| 2. El jugador selecciona una de las diez partidas.  |   |   |
|   |   | 3. El sistema carga la partida seleccionada   |
| <b>Curso Alternativo A:</b>   |   |   |
| 2. El jugador no selecciona ninguna partida y decide cancelar la función de cargar juego almacenado |   |   |
| 3. El sistema retorna al usuario a las opciones principales.  |   |   |
| <b>Curso Alternativo B:</b>   |   |   |
| 1. El sistema determina que el jugador no ha iniciado sesión  |   |   |
| 2. El sistema ejecuta la función de iniciar sesión (ver caso de uso: <b>CU-016</b> )                |   |   |
| 3. El sistema ejecuta los pasos del curso normal de este caso de uso.                               |   |   |



| <b>Curso Alterno C:</b>   |
|---|
| 2. El jugador selecciona una partida equivocada y decide elegir hacer una nueva elección. |
| 3. El sistema muestra nuevamente el listado de partidas almacenadas.                      |
| 4. El jugador selecciona otra partida de juego almacenada.                                |

| <b>Caso de Uso:</b> | <b>Mostrar Bitácora (Alto Nivel)</b>   | <b>Código: CU-014</b> |
|---------------------|--|-----------------------|
| <b>Actores:</b>     | Jugador  |                       |
| <b>Tipo:</b>        | Primario - Esencial  |                       |
| <b>Descripción</b>  | Este caso de uso ocurre cuando el usuario (con una cuenta existente), ha seleccionado la opción "Mostrar Bitácora" desde las opciones principales. |                       |

| <b>Caso de Uso:</b>          | <b>Mostrar Bitácora (Extendido)</b>  | <b>Código: CU-014</b> |
|------------------------------|--|-----------------------|
| <b>Actores:</b>              | Jugador  |                       |
| <b>Propósito:</b>            | Permitir al usuario visualizar su avance en los diferentes niveles del juego.  |                       |
| <b>Resumen:</b>              | El jugador puede tener una perspectiva global de su avance y aprendizaje a través de la interacción de este con el videojuego. |                       |
| <b>Referencias cruzadas:</b> | R 4,5  |                       |

| <b>Curso Normal de los Eventos</b>  |   |
|-------------------------------------|---|
| <b>Acción de los actores</b>        | <b>Respuesta del sistema</b>  |
|                                     | 1. El sistema del videojuego presenta al usuario información de su historial de juego y nivel de puntos acumulado |
| 2. El jugador Selecciona "regresar" |   |
|                                     | 3. El sistema finaliza y retorna al jugador a las opciones principal.   |

| <b>Curso Alterno A:</b>  |
|--|
| 1. El sistema determina que el jugador no ha iniciado sesión                           |
| 2. El sistema ejecuta la función de Gestionar sesión (ver caso de uso: <b>CU-015</b> ) |
| 3. El sistema ejecuta los pasos del curso normal de este caso de uso.                  |



|                     |   |                       |
|---------------------|---|-----------------------|
| <b>Caso de Uso:</b> | <b>Gestionar Sesión (Alto Nivel)</b>  | <b>Código: CU-015</b> |
| <b>Actores:</b>     | Jugador   |                       |
| <b>Tipo:</b>        | Primario - Esencial   |                       |
| <b>Descripción</b>  | La gestión de sesión será ejecutada, para administrar las cuentas de usuario (crear y eliminar cuentas), como también para llevar un registro de las sesiones de juego de cada usuario. |                       |

|                              |   |                       |
|------------------------------|---|-----------------------|
| <b>Caso de Uso:</b>          | <b>Gestionar Sesión (Expandido)</b>   | <b>Código: CU-015</b> |
| <b>Actores:</b>              | Jugador   |                       |
| <b>Propósito:</b>            | Permitir la administración de las cuentas de usuario (crear y eliminar) como también permitirá la autenticación de los jugadores.                         |                       |
| <b>Resumen:</b>              | El sistema muestra al usuario las opciones de <b>“Gestionar Sesión”</b> las cuales son: Iniciar sesión, crear nuevo usuario y Eliminar cuenta de usuario. |                       |
| <b>Referencias cruzadas:</b> | R 2.1, R 2.2, R 2.3, R 3.2  |                       |

**Curso Normal de los Eventos**

| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |
|---|--|
|   | 1. El sistema muestra al jugador las opciones de <b>“Gestionar Sesión”</b>   |
| 2. El jugador selecciona la opción que desea: <ul style="list-style-type: none"> <li>a. Si el usuario selecciona Iniciar Sesión, ver caso de uso: <b>“Iniciar Sesión”</b>, CU-016.</li> <li>b. Si el usuario selecciona Crear Nuevo Usuario, ver caso de uso <b>“Crear Nuevo Usuario”</b>, CU-017.</li> <li>c. Si el usuario seleccionar Eliminar Usuario, ver caso de uso: <b>“Eliminar Usuario”</b>, CU-018.</li> </ul> |  |
|   | 3. El sistema carga los recursos para la opción seleccionada por el usuario. |

**Curso Alterno A:**

|   |
|---|
| 2. El jugador no selecciona ninguna opción y decide cancela <b>“Gestionar Sesión”</b> . |
| 3. El sistema retorna a <b>“Seleccionar Opciones”</b> (o menú principal) CU-002.        |



| Caso de Uso:       | Iniciar Sesión (Alto Nivel)   | Código: CU-016 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario-esencial   |                |
| <b>Descripción</b> | Este caso de uso permite autenticar al jugador que selecciona una de las siguientes opciones principales: Cargar Juego Almacenado, Jugar, Mostrar Bitácora. Se le permitirá al jugador autenticarse mediante dos modalidades: 1) Desde Gestionar Sesión o desde una de las 3 opciones mencionadas. Esto le permite al sistema llevar un registro de las acciones que el jugador realiza en el videojuego. |                |

| Caso de Uso:                 | Iniciar Sesión (Expandido)  | Código: CU-016 |
|------------------------------|---|----------------|
| <b>Actores:</b>              | Jugador   |                |
| <b>Propósito:</b>            | Autenticar al usuario con la finalidad de llevar un registro de las acciones realizadas por este durante la interacción con el videojuego en las opciones de cargar Juego Almacenado, Jugar y Mostrar Bitácora. |                |
| <b>Resumen:</b>              | El sistema autentica al jugador antes de cargar un Juego Almacenado, Jugar y Mostrar Bitácora. Esto le permitirá al sistema tener un control individual de cada jugador.  |                |
| <b>Referencias cruzadas:</b> | R 2.2, R 3.2  |                |

**Curso Normal de los Eventos**

| Acción de los actores  | Respuesta del sistema  |
|--|--|
| 1. Este caso de uso se ejecuta cuando el jugador inicia sesión, mediante la opción: Gestionar Sesión; o cuando el jugador, sin haber iniciado una sesión, elige una de las siguientes opciones principales: Cargar Juego Almacenado, Jugar y Mostrar Bitácora. |  |
|  | 2. El sistema solicita el nombre de usuario (login) y contraseña (password). |
| 3. El jugador introduce su nombre de usuario (login) y su contraseña (password)  |  |
|  | 4. El sistema valida los datos del usuario e inicializa la sesión.           |

**Curso Alternativo A:**

|   |
|---|
| 3. El jugador introduce nombre de usuario o contraseña incorrectos. |
| 4. El sistema muestra al usuario un mensaje de "datos incorrectos". |
| 5. El sistema solicita al jugador introducir nuevamente los datos   |



|                     |   |                       |
|---------------------|---|-----------------------|
| <b>Caso de Uso:</b> | <b>Crear Nuevo Usuario (Alto Nivel)</b>   | <b>Código: CU-017</b> |
| <b>Actores:</b>     | Jugador   |                       |
| <b>Tipo:</b>        | Primario  |                       |
| <b>Descripción</b>  | Este caso de uso ocurre cuando el usuario ha seleccionado la opción “Crear Nuevo Usuario” desde el caso de uso <b>“Gestionar Sesión”</b> . El sistema pide al usuario el ingreso de Nombre de Usuario (login) y una contraseña (password) para crear una nueva cuenta en donde se almacenará el avance del jugador. |                       |

|                              |   |                       |
|------------------------------|---|-----------------------|
| <b>Caso de Uso:</b>          | <b>Crear Nuevo Usuario (Expandido)</b>  | <b>Código: CU-017</b> |
| <b>Actores:</b>              | Jugador   |                       |
| <b>Propósito:</b>            | Permitir al jugador crear una nueva cuenta para llevar un control de las acciones de cada jugador. También tener una cuenta de usuario le permitirá almacenar el avance obtenido en las partidas de juego que realice en los diferentes niveles, esto le permitirá recuperar una partida de juego desde el punto en el que haya guardo el avance. |                       |
| <b>Resumen:</b>              | El jugador tendrá la oportunidad de crear una nueva cuenta para diferenciar sus registros de otros usuarios y continuar con el avance del videojuego en sesiones posteriores.   |                       |
| <b>Referencias cruzadas:</b> | R 2.1   |                       |

**Curso Normal de los Eventos**

| Acción de los actores  | Respuesta del sistema   |
|--|---|
|  | 1. El sistema del videojuego le indica al usuario que introduzca un nombre de usuario (login) y una contraseña (password) para crear una nueva cuenta |
| 2. El jugador introduce un Nombre de Usuario y una contraseña para el nuevo usuario a crear. |   |
|  | 3. El sistema genera un nuevo expediente con el nombre del usuario y contraseña.  |

**Curso Alterno A:**

- 2. El jugador introduce un Nombre de Usuario existente.
- 3. El sistema le muestra un mensaje en el que le indica que ese usuario ya existe y que introduzca uno diferente.

**Curso Alterno B:**

- 2. El jugador no introduce datos de usuario y decide cancela la Función de “Crear Nuevo Usuario”
- 3. El sistema retorna al usuario a las opciones principales.



|                     |  |                       |
|---------------------|--|-----------------------|
| <b>Caso de Uso:</b> | <b>Eliminar Usuario (Alto Nivel)</b>   | <b>Código: CU-018</b> |
| <b>Actores:</b>     | Jugador  |                       |
| <b>Tipo:</b>        | Primario - Esencial  |                       |
| <b>Descripción</b>  | Este caso de uso ocurre cuando el usuario ha seleccionado la opción "Eliminar Usuario". Solo tendrá acceso a esta función un usuario con una cuenta existente. |                       |

|  |  |                       |
|--|--|-----------------------|
| <b>Caso de Uso:</b>  | <b>Eliminar Usuario (Extendido)</b>  | <b>Código: CU-018</b> |
| <b>Actores:</b>  | Jugador  |                       |
| <b>Propósito:</b>  | Eliminar cuentas de usuarios obsoletas o indeseables.  |                       |
| <b>Resumen:</b>  | El jugador, después de haber iniciado una sesión, podrá eliminar su cuenta de usuario, accediendo a la opción de Eliminar Usuario. |                       |
| <b>Referencias cruzadas:</b>                                 | R 2.3  |                       |
| <b>Curso Normal de los Eventos</b>                           |  |                       |
| <b>Acción de los actores</b>                                 | <b>Respuesta del sistema</b>   |                       |
|  | 1. El sistema del videojuego le indica al usuario si está seguro de eliminar su cuenta   |                       |
| 2. El jugador indica que "Si"                                |  |                       |
|  | 3. El sistema elimina la cuenta y lleva al usuario a las opciones principales.   |                       |
| <b>Curso Alternativo A:</b>                                  |  |                       |
| 2. El jugador indica que "No"                                |  |                       |
| 3. El sistema retorna al jugador a las opciones principales. |  |                       |



| Caso de Uso:       | Mostrar Demo (Alto Nivel)   | Código: CU-019 |
|--------------------|---|----------------|
| <b>Actores:</b>    | Jugador   |                |
| <b>Tipo:</b>       | Primario - Esencial   |                |
| <b>Descripción</b> | Este caso de uso se lleva a cabo cuando el jugador seleccionado la opción "Mostrar Demo". El sistema le presenta una demostración del juego en la que se le muestra una partida de juego corta que se ejecuta automáticamente, como una vista previa de lo que hace el sistema. |                |

| Caso de Uso:  | Mostrar Demo (Expandido)   | Código: CU-019 |
|---|--|----------------|
| <b>Actores:</b>   | Jugador  |                |
| <b>Propósito:</b>   | Mostrar al jugador una partida de juego corta en la que se le presenta la funcionalidad del videojuego.  |                |
| <b>Resumen:</b>   | Antes de iniciar una sesión por primera vez, el jugador puede tener una vista previa del funcionamiento del videojuego, para lo cual debe ejecutar la opción "Mostrar Demo", esta opción le mostrará escenas del videojuego en las que represente la funcionalidad básica del videojuego y de esta forma el jugador pueda familiarizarse con el sistema. |                |
| <b>Referencias cruzadas:</b>  | R 4.16   |                |
| <b>Curso Normal de los Eventos</b>  |  |                |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                |
|   | 1. El sistema carga los recursos del videojuego relacionados a la carga del demo.  |                |
|   | 2. El sistema dibuja o presenta la demostración del funcionamiento del software.   |                |
| 3. El jugador ve la demostración y regresar a las opciones principales.     |  |                |
| <b>Curso Alternativo A:</b>   |  |                |
| 3. El jugador decide no ver el demo y decide cancelar la demostración.      |  |                |
| 4. El sistema cancela la demostración y retorna a las opciones principales. |  |                |





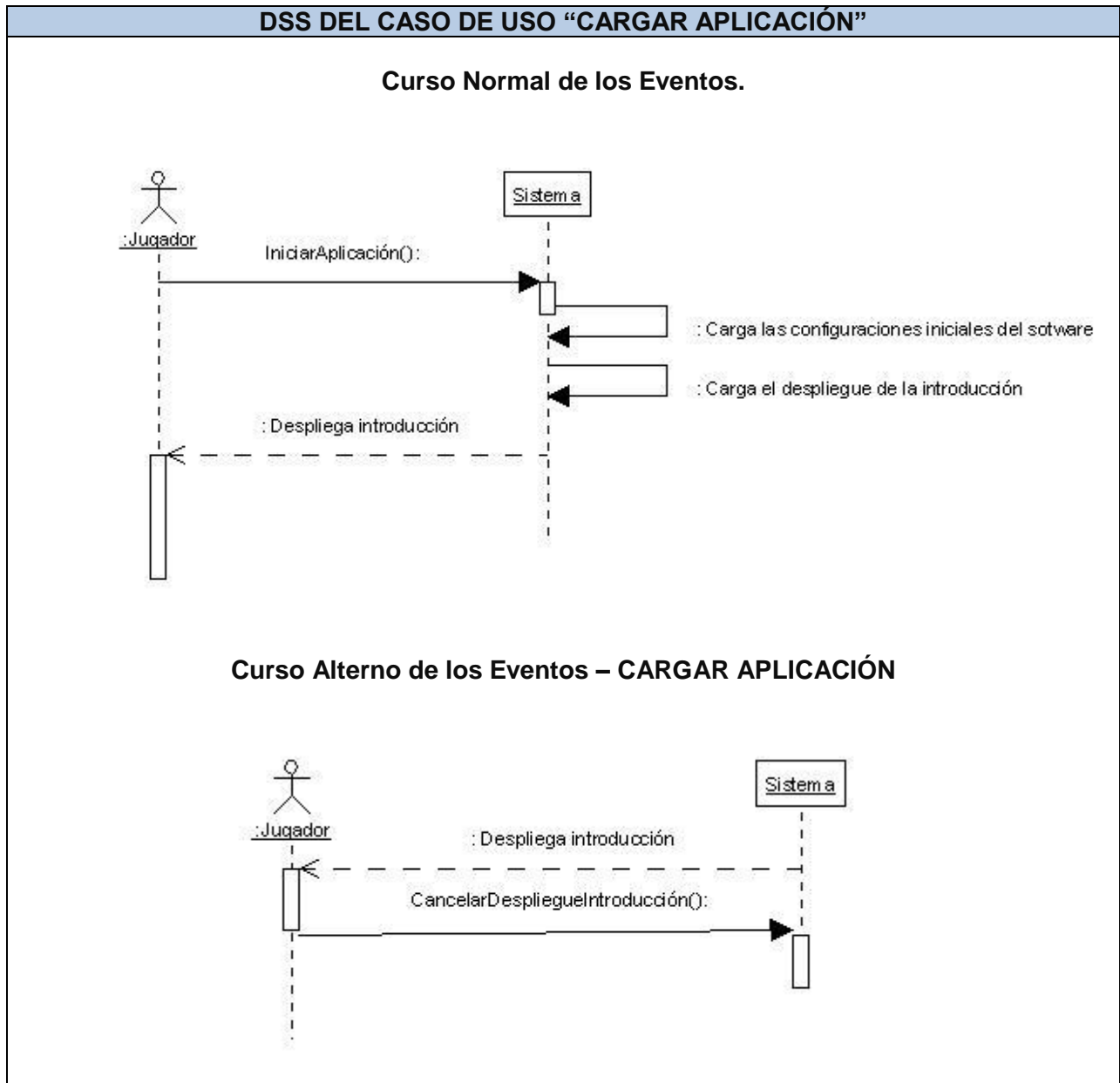
|                     |   |                       |
|---------------------|---|-----------------------|
| <b>Caso de Uso:</b> | <b>Salir (Alto Nivel)</b>   | <b>Código: CU-020</b> |
| <b>Actores:</b>     | Jugador   |                       |
| <b>Tipo:</b>        | Primario - Esencial   |                       |
| <b>Descripción</b>  | Este caso de uso ocurre cuando el usuario ha seleccionado la opción "Salir" desde las opciones principales. |                       |

|                                    |  |                       |
|------------------------------------|--|-----------------------|
| <b>Caso de Uso:</b>                | <b>Salir (Extendido)</b>   | <b>Código: CU-020</b> |
| <b>Actores:</b>                    | Jugador  |                       |
| <b>Propósito:</b>                  | Cerrar el videojuego y terminar la sesión de juego.  |                       |
| <b>Resumen:</b>                    | Este caso de uso se ejecuta cuando el usuario indica al sistema del videojuego que desea cerrar la aplicación. |                       |
| <b>Referencias cruzadas:</b>       | R 5.4  |                       |
| <b>Curso Normal de los Eventos</b> |  |                       |
| <b>Acción de los actores</b>       | <b>Respuesta del sistema</b>   |                       |
|                                    | 1. El sistema cierra la aplicación y descarga los recursos asociados a la aplicación.                          |                       |
| 2. El jugador sale del videojuego. |  |                       |

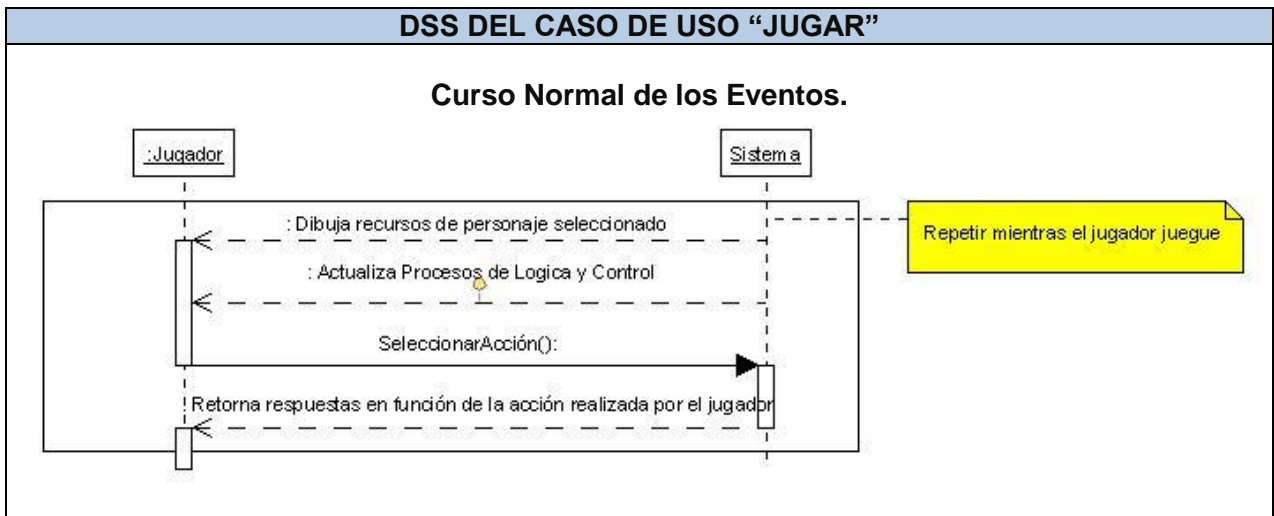
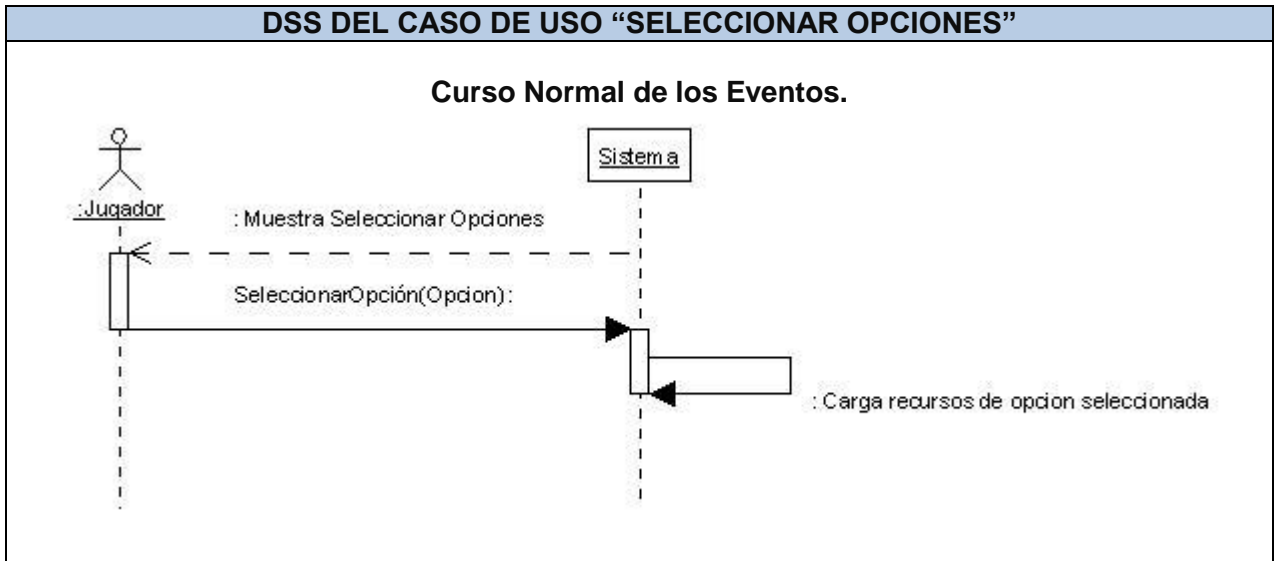


#### 4.2.4. Diagramas de Secuencia del Sistema (DSS) <sup>91</sup>

Un diagrama de secuencia de un sistema muestra gráficamente los eventos que fluyen de los actores al sistema. Permite visualizar fácilmente cualquier interacción Actor-Sistema y muestra las funciones en su versión preliminar, asociadas a dicha interacción.



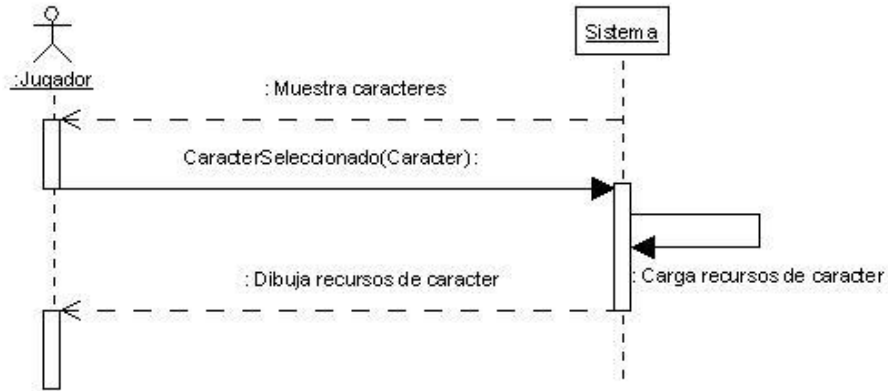
<sup>91</sup> De acuerdo a formato presentado en Systems Analysis and Design in a Changing World, Fourth Edition Por John W. Satzinger, JACKSON, BURD, Robert B. Jackson, Stephen D. Burd, SATZINGER - Course Technology Ptr (2006) – Cap 7.





**DSS DEL CASO DE USO “SELECCIONAR CARACTER”**

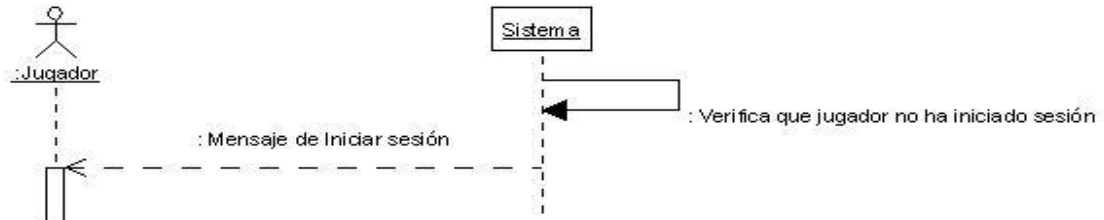
**Curso Normal de los Eventos.**

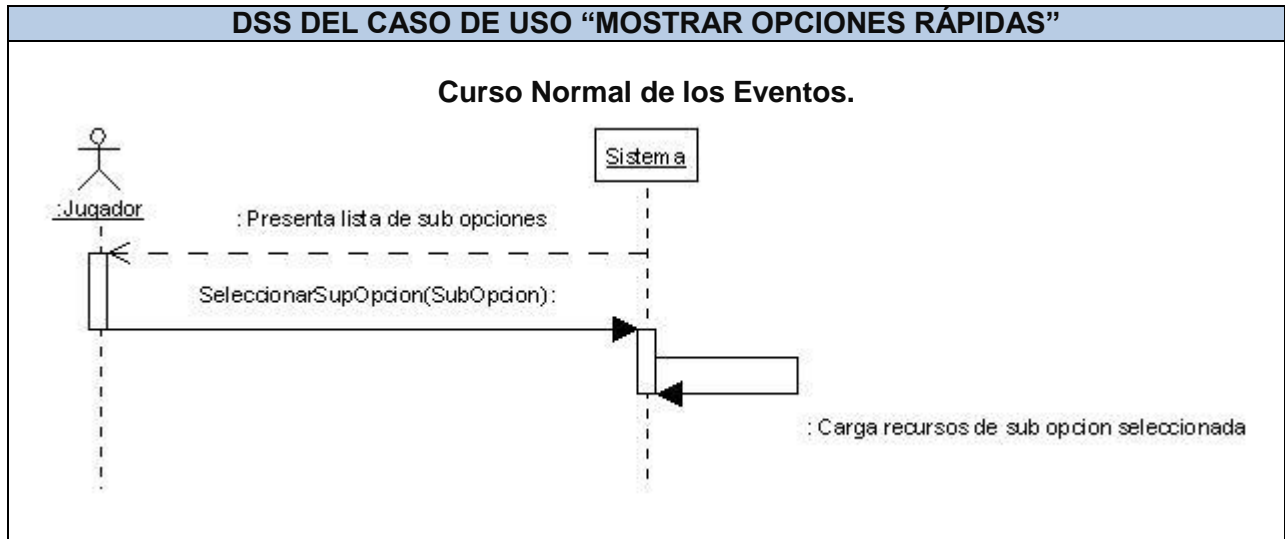
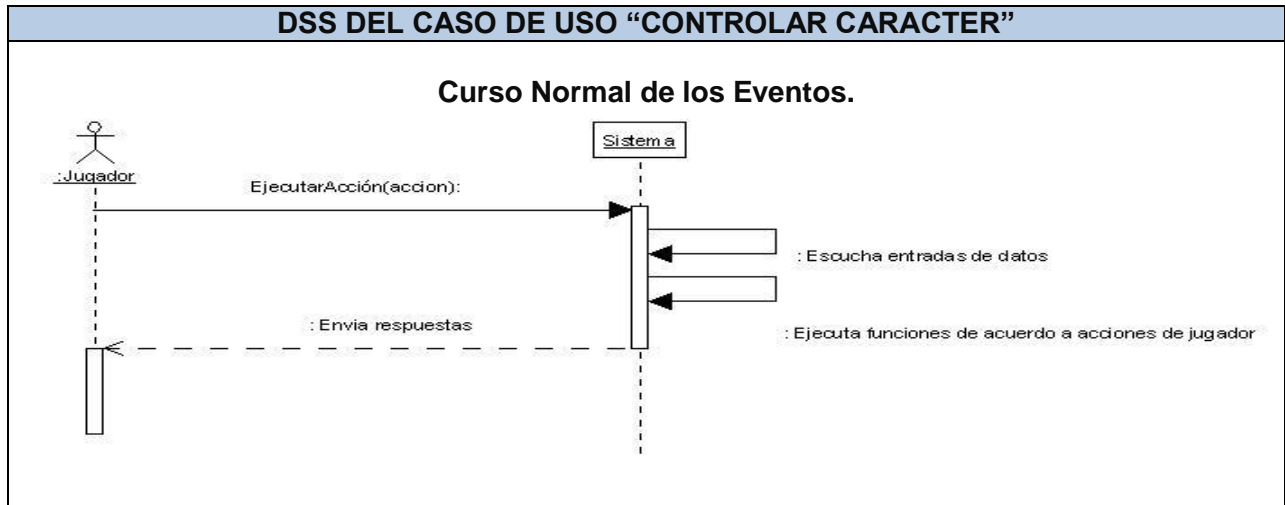


**Curso Alternativo A de los Eventos – SELECCIONAR CARACTER.**



**Curso Alternativo B de los Eventos – SELECCIONAR CARACTER.**

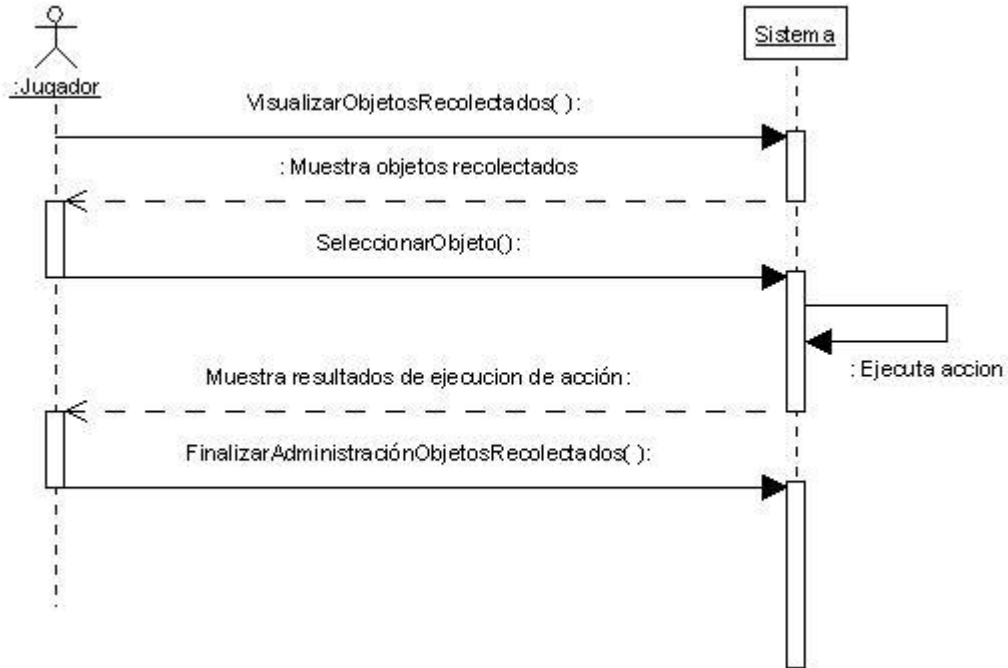






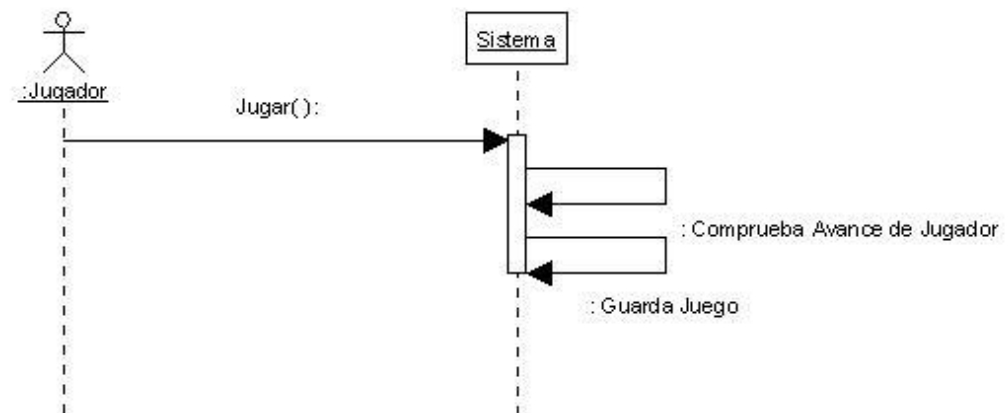
**DSS DEL CASO DE USO “ADMINISTRAR OBJETOS RECOLECTADOS”**

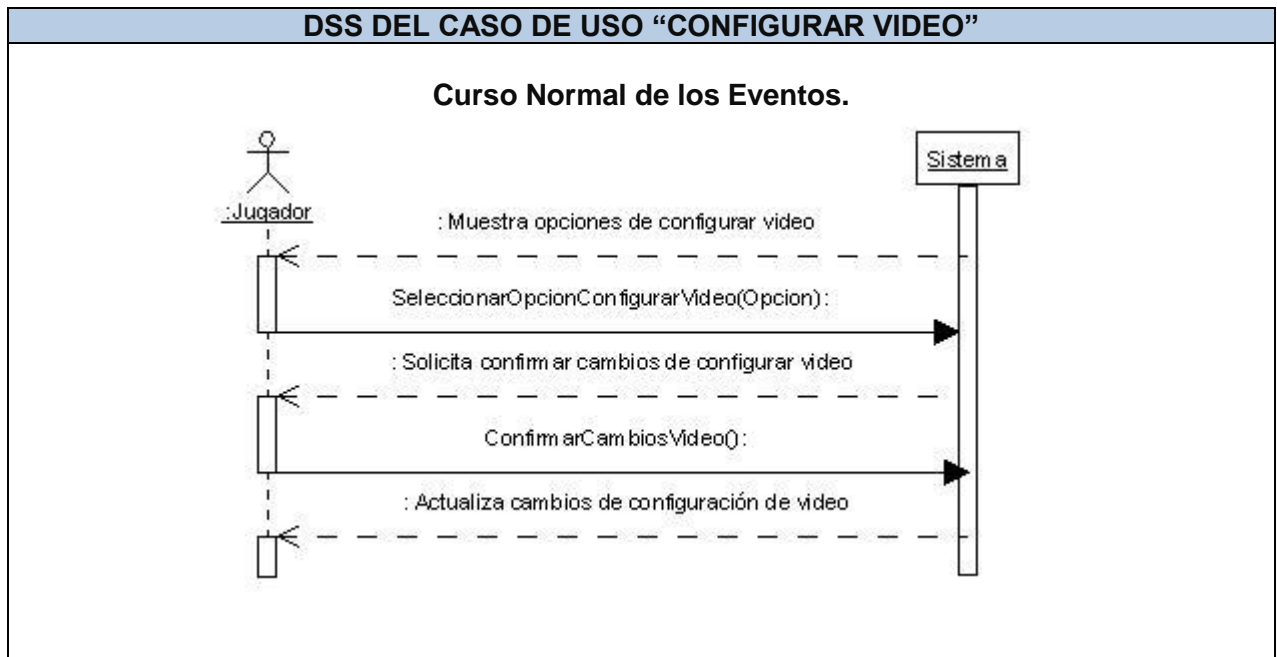
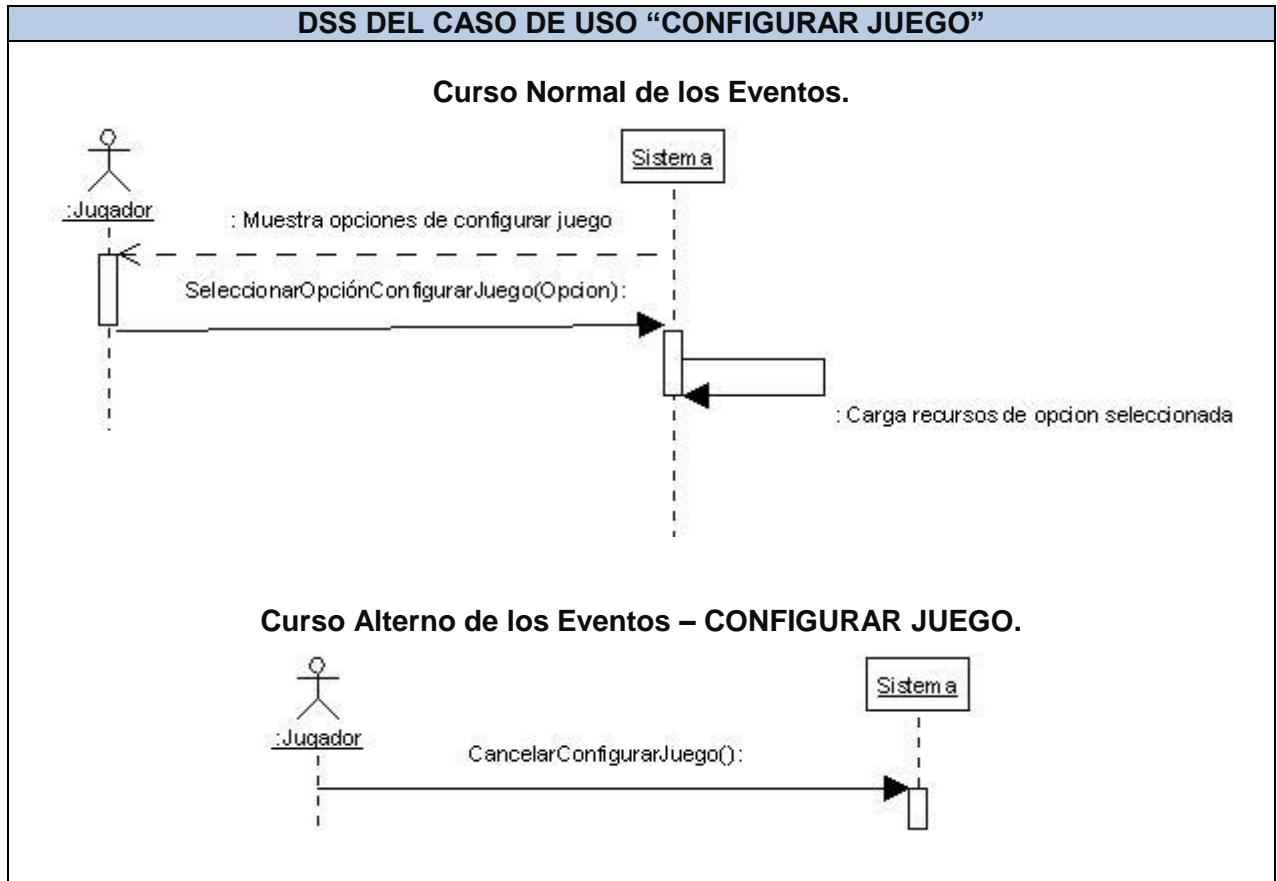
**Curso Normal de los Eventos.**



**DSS DEL CASO DE USO “GUARDAR SESIÓN”**

**Curso Normal de los Eventos.**





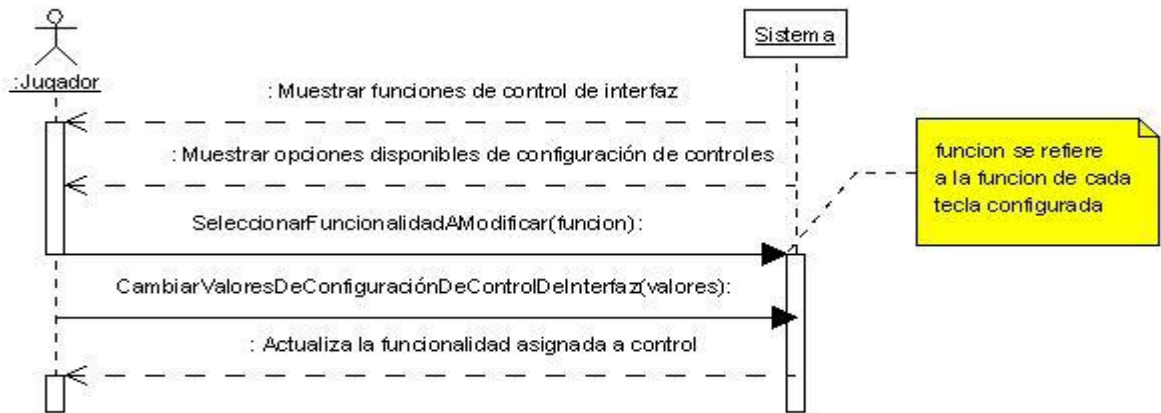


**Curso Alterno de los Eventos – CONFIGURAR VIDEO.**



**DSS DEL CASO DE USO “CONFIGURAR CONTROLES”**

**Curso Normal de los Eventos.**



**Curso Alterno de los Eventos – CONFIGURAR CONTROLES.**

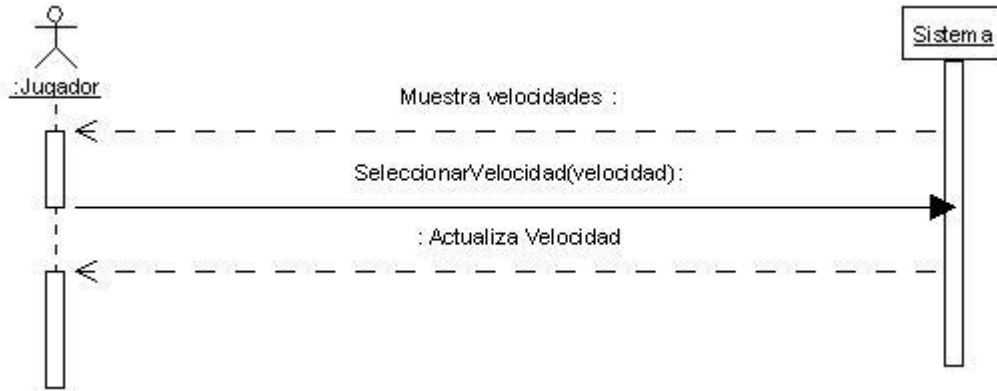




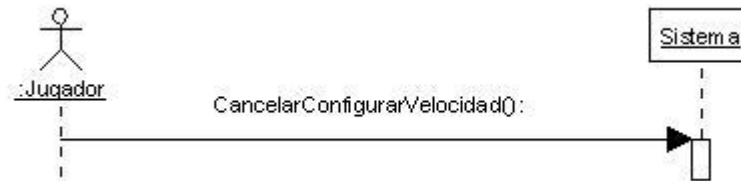


**DSS DEL CASO DE USO "CONFIGURAR VELOCIDAD"**

**Curso Normal de los Eventos.**

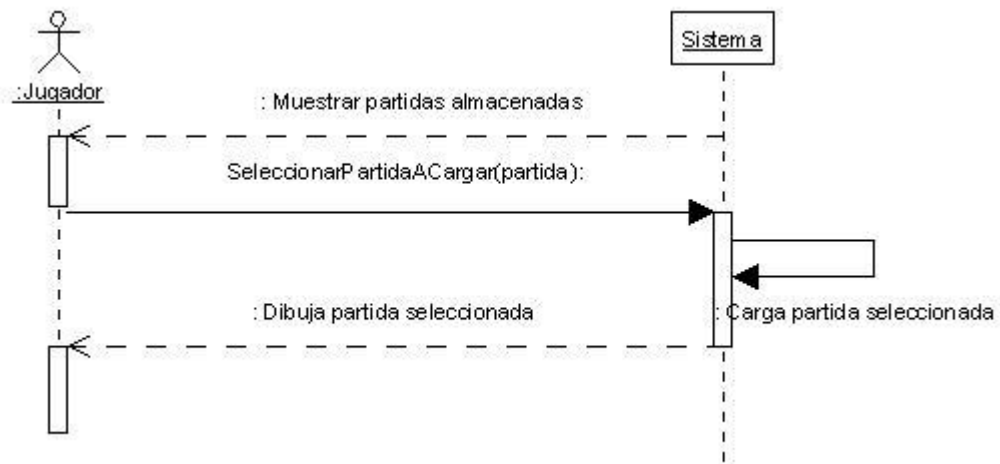


**Curso Alternativo de los Eventos – CONFIGURAR VELOCIDAD.**



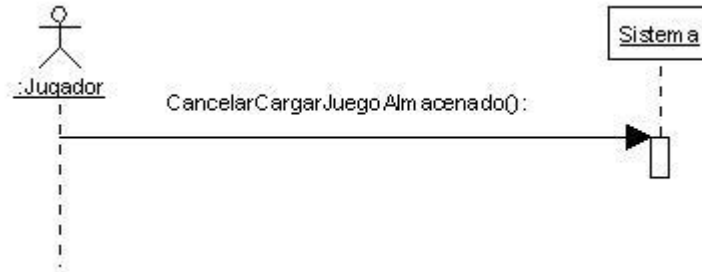
**DSS DEL CASO DE USO "CARGAR JUEGO ALMACENADO"**

**Curso Normal de los Eventos.**

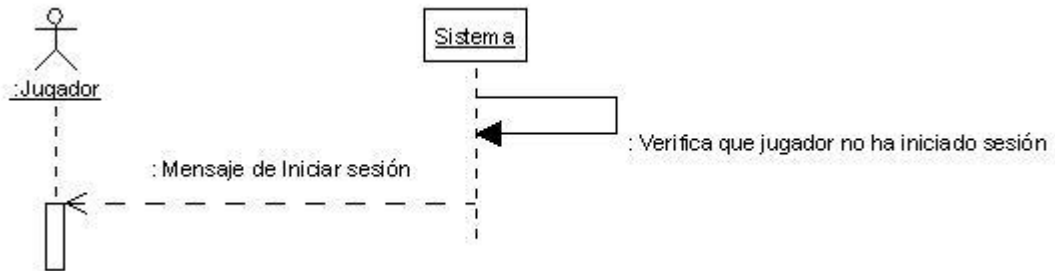




**Curso Alternativo A de los Eventos – CARGAR JUEGO ALMACENADO.**



**Curso Alternativo B de los Eventos – CARGAR JUEGO ALMACENADO.**

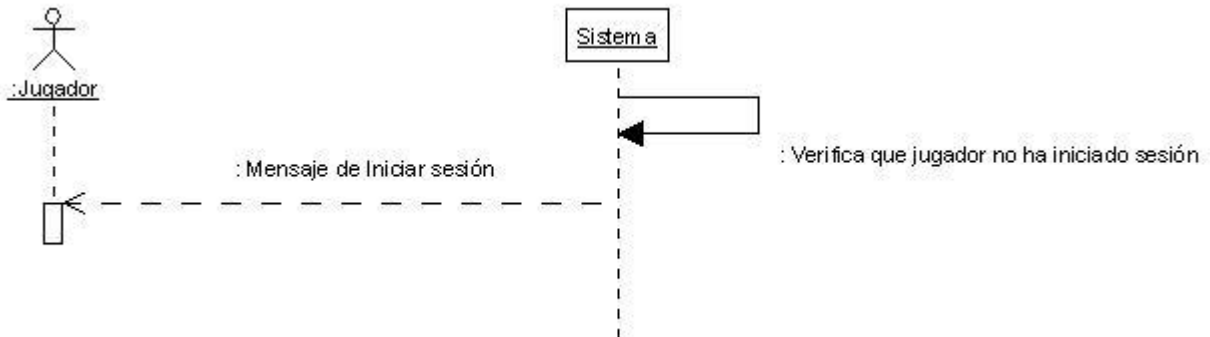


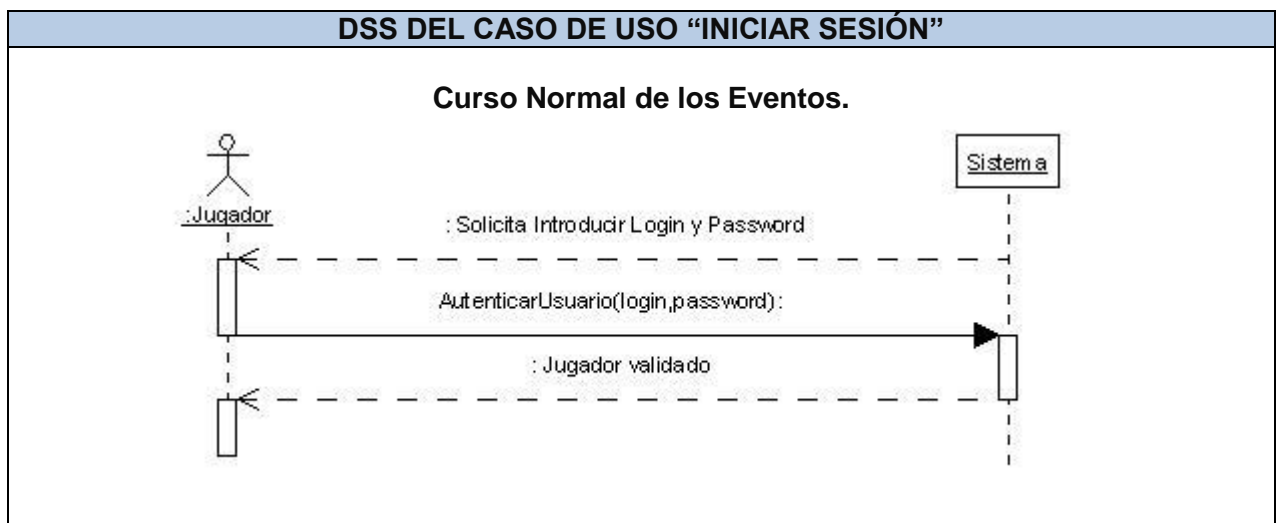
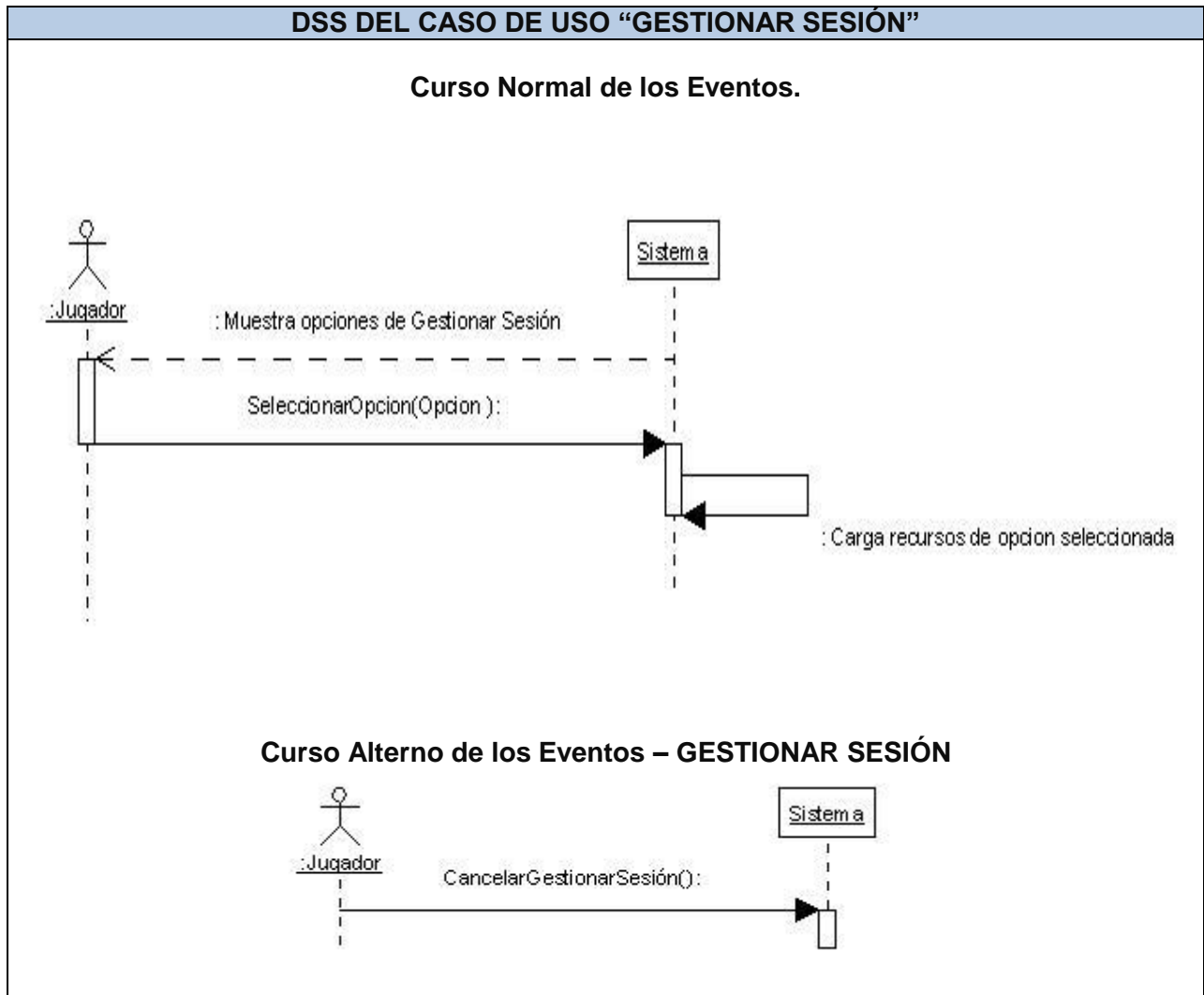
**DSS DEL CASO DE USO “MOSTRAR BITÁCORA”**

**Curso Normal de los Eventos.**



**Curso Alternativo de los Eventos –MOSTRAR BITÁCORA**





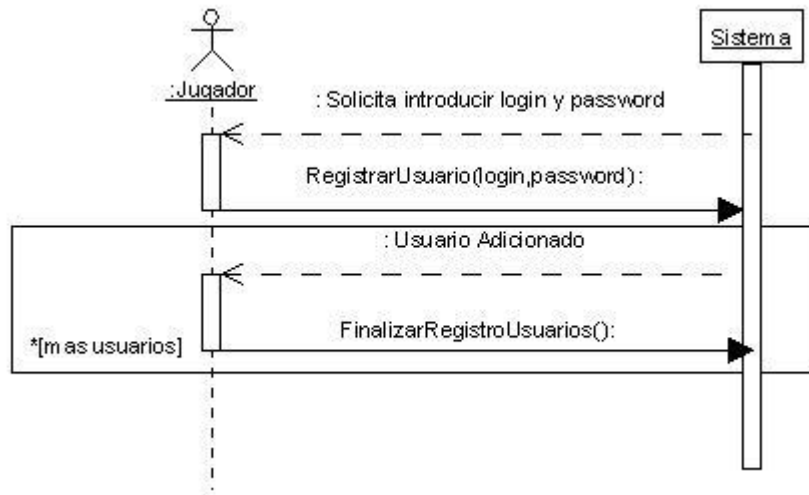


**Curso Alterno de los Eventos – INICIAR SECIÓN**

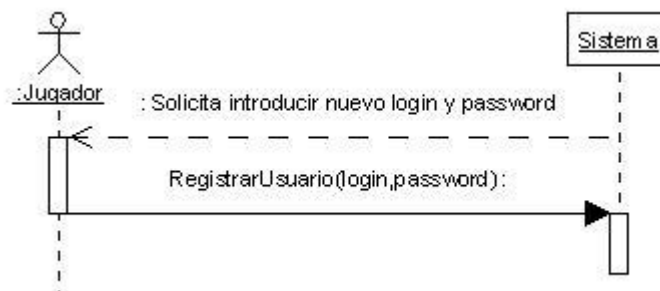


**DSS DEL CASO DE USO “CREAR NUEVO USUARIO”**

**Curso Normal de los Eventos.**



**Cursos Alterno A de los Eventos – CREAR NUEVO USUARIO.**



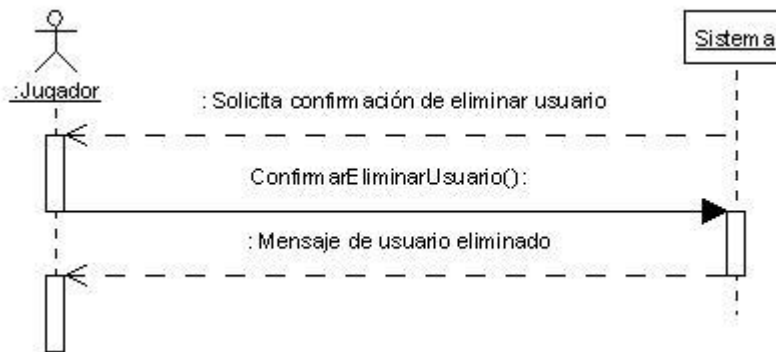


**Cursos Alternos B de los Eventos – CREAR NUEVO USUARIO.**

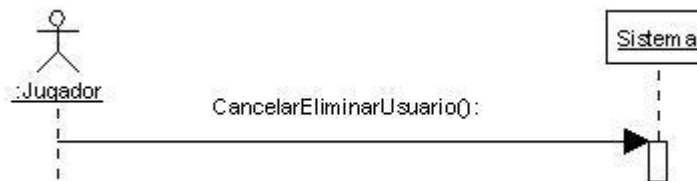


**DSS DEL CASO DE USO “ELIMINAR USUARIO”**

**Curso Normal de los Eventos.**

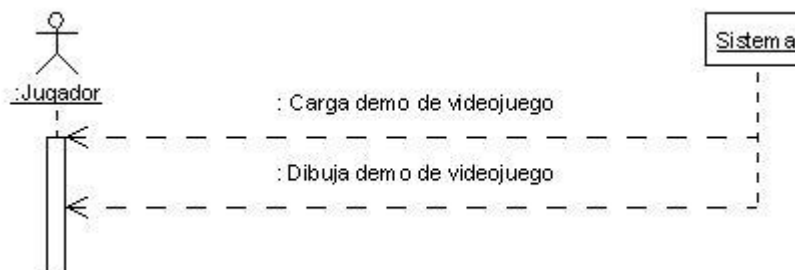


**Alternos de los Eventos – ELIMINAR USUARIO.**



**DSS DEL CASO DE USO “MOSTRAR DEMO”**

**Curso Normal de los Eventos.**





### Curso Alterno de los Eventos – MOSTRAR DEMO



### DSS DEL CASO DE USO “SALIR”

#### Curso Normal de los Eventos.





### 4.2.5. Contratos de operación <sup>92</sup>

Una vez se tienen las Operaciones del Sistema identificadas en los Diagramas de Secuencia, se describe mediante contratos el comportamiento esperado del sistema en cada operación.

Un Contrato es un documento que describe qué es lo que se espera de una operación. Tiene una redacción en estilo declarativo, enfatizando en el qué más que en el cómo. Lo más común es expresar los contratos en forma de precondiciones y postcondiciones en torno a cambios de estado sin especificar como se ha llegado a tal condición. Se puede escribir un contrato para un método individual de una clase software (como aquí se presenta), o para una operación del sistema completa. Un Contrato de Operación del Sistema describe cambios en el estado del sistema cuando una operación del sistema es invocada.

| Nombre del contrato          | Nombre del contrato | Código |  |
|------------------------------|---------------------|--------|--|
| <b>Responsabilidades:</b>    |                     |        |  |
| <b>Tipo:</b>                 |                     |        |  |
| <b>Referencias cruzadas:</b> |                     |        |  |
| <b>Notas:</b>                |                     |        |  |
| <b>Excepciones</b>           |                     |        |  |
| <b>Salida:</b>               |                     |        |  |
| <b>Precondiciones:</b>       |                     |        |  |
| <b>Poscondiciones:</b>       |                     |        |  |

Cuadro 4.9: Formato de los Contratos de Operación

<sup>92</sup> De acuerdo a formato presentado en la sección "Comportamientos de los sistemas: Contratos" de UML y PATRONES de Craig Larman, Cap., 14, Pág. 145.



#### 4.2.5.1. Contratos para Caso de Uso Iniciar Sesión

| Nombre                       | IntroducirLoginyPassword()  | Código | CO1 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    |   |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 2.2, R 3.2  |        |     |
| <b>Notas:</b>                | El diseño de la pantalla debe de ser atractiva y de sentido común para niños de 6 años en adelante. Las entradas deben de ser a través de teclado, mouse y control de videojuegos.                                  |        |     |
| <b>Excepciones</b>           | El usuario cancela <i>IntroducirLoginyPassword()</i>  |        |     |
| <b>Salida:</b>               | Pantalla de entrada de datos.   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ El jugador tiene que poseer una cuenta activa guardada en disco.</li> <li>✓ El jugador no ha iniciado sesión aun.</li> </ul>   |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanció una pantalla de entrada de datos.</li> <li>✓ Se creó un cuadro de entrada para el login.</li> <li>✓ Se creó un cuadro de entrada para el password.</li> </ul> |        |     |

| Nombre                       | AutenticarUsuario(string login,string password)   | Código | CO2 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Autenticar los datos del Jugador.   |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 2.2, R 3.2  |        |     |
| <b>Notas:</b>                |   |        |     |
| <b>Excepciones</b>           | El jugador cancela <i>AutenticarUsuario(login, password)</i> , El jugador introduce datos erróneos, el jugador no posee cuenta de usuario.  |        |     |
| <b>Salida:</b>               |   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ El jugador tiene que poseer una cuenta activa guardada en disco.</li> <li>✓ El jugador no ha iniciado sesión aun.</li> <li>✓ El jugador tiene que introducir un login y password correctos.</li> <li>✓ El jugador tiene que aceptar iniciar sesión.</li> </ul> |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se obtuvieron del archivo de sesión los datos del jugador.</li> <li>✓ Se Autentica antes de Validar al jugador para iniciar su sesión.</li> </ul>  |        |     |





| Nombre                       | ValidarJugador()   | Código | CO3 |
|------------------------------|--|--------|-----|
| <b>Responsabilidades:</b>    | Validar sesión de Jugador.   |        |     |
| <b>Tipo:</b>                 | Sistema  |        |     |
| <b>Referencias cruzadas:</b> | R 2.2, R 3.2   |        |     |
| <b>Notas:</b>                | Solo mostrar la pantalla de entrada de datos al jugador si la excepción de datos no validos se presenta, de lo contrario seguir esta la secuencia lógica descrita en este contrato.  |        |     |
| <b>Excepciones</b>           | El jugador no es validado con su login y/o password  |        |     |
| <b>Salida:</b>               |  |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ El jugador debe haber introducido valores para el login y el password.</li> <li>✓ El login y password introducidos deben de ser correctos.</li> <li>✓ Se obtuvieron del archivo de sesión los datos del jugador.</li> </ul>   |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignó un valor a la variable interna gestión de jugador.</li> <li>✓ Se crearon tres instancias de la clase Caracter().</li> <li>✓ Se instanciaron las imágenes de fondo.</li> <li>✓ Se cargaron los sonidos de dialogo de los caracteres.</li> <li>✓ Se presentaron los caracteres para que el usuario elija.</li> <li>✓ Se validó la sesión del jugador</li> </ul> |        |     |

**4.2.5.2. Contratos para Caso de Uso Seleccionar Personaje**

| Nombre                       | MostrarCaracteres()   | Código | CO4 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Permitir al usuario elegir uno de tres caracteres para iniciar su nueva partida. Permitir obtener la información necesaria para cargar recursos de juego. |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 2.9   |        |     |
| <b>Notas:</b>                | Debe de ser claro para el jugador que puede cancelar la función aquí descrita si lo desea.  |        |     |
| <b>Excepciones</b>           | El jugador cancela la función <i>MonstrarCaracteres()</i>   |        |     |
| <b>Salida:</b>               | Caracteres 3D, planeta tierra, nave en órbita, texturas de controles.   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ La sesión debe haber sido validada.</li> </ul>   |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó un carácter.</li> <li>✓ Se asoció el carácter seleccionado al jugador.</li> </ul>                  |        |     |



| Nombre                       | CaracterSeleccionado(Entero Caracter)   | Código | CO5 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Enviar al sistema un valor asociado a un carácter seleccionado para gestionar los recursos necesarios al nivel inicial asignado a cada carácter.                        |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 2.13, R 3.5, R 3.8, R 4.13, R   |        |     |
| <b>Notas:</b>                |   |        |     |
| <b>Excepciones</b>           |   |        |     |
| <b>Salida:</b>               | Caracteres 3D, planeta tierra, nave en órbita, texturas de controles.   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó un carácter.</li> <li>✓ Existe una asociación del carácter seleccionado con el jugador.</li> </ul>               |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron recursos para determinar la temática del nivel a jugar.</li> <li>✓ Se instanció la pantalla de espera.</li> </ul> |        |     |

#### 4.2.5.3. Contratos para Caso de Uso Cargar Aplicación

| Nombre                       | CargarRecursos()  | Código | CO6 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Enviar al sistema un valor asociado a un carácter seleccionado para gestionar los recursos necesarios al nivel inicial asignado a cada carácter.  |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 1.1   |        |     |
| <b>Notas:</b>                | La pantalla de espera podría variar en cada nueva partida de acuerdo a una aleatoriedad predeterminada.   |        |     |
| <b>Excepciones</b>           |   |        |     |
| <b>Salida:</b>               | Pantalla de espera  |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó un carácter.</li> <li>✓ Se instanció la pantalla de espera.</li> <li>✓ Se asignaron estados a los principales switches booleanos para determinar la ruta lógica del sistema.</li> </ul>  |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se mostró la pantalla de espera mientras se cargan los recursos.</li> <li>✓ Se instanciaron los recursos (objetos 3D) acordes al carácter seleccionado.</li> <li>✓ Se preparó el escenario virtual para el nivel del juego relativo al carácter previamente seleccionado.</li> </ul> |        |     |



#### 4.2.5.4. Contratos para Caso de Uso Jugar

| Nombre                       | DibujarNivel(GraphicsDevice dispositivographics, GameTime contadorInterno)  | Código | CO7 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Dibujar la escena virtual, gestionar los algoritmos de inteligencia artificial, actualizar valores matriciales, Seleccionar Respuestas de Caracteres Virtuales e interacción Jugador-Carácter-Mundo Virtual   |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 4.15, R 4.14, R4.11, 4.10,4.7, 4.6, 4.2,4.3   |        |     |
| <b>Notas:</b>                | Eventos Recursivos, corriendo a una velocidad aproximada de 30 -60 cuadros por segundo  |        |     |
| <b>Excepciones</b>           | El usuario cancela la partida   |        |     |
| <b>Salida:</b>               | Mundo y objetos virtuales 3D  |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanciaron los recursos (objetos 3D) acordes al carácter seleccionado.</li> <li>✓ Se preparó el escenario virtual para el primer del juego relativo al carácter previamente seleccionado.</li> </ul>                                |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se ha instanciado el nuevo nivel</li> <li>✓ Se dibujó el nuevo nivel</li> <li>✓ Se volvió al menú principal de seleccionar opciones (cuando el usuario fracasase en su avance en el nivel dado o deseado terminar la partida)</li> </ul> |        |     |

| Nombre                       | Jugar()  | Código | CO8 |
|------------------------------|--|--------|-----|
| <b>Responsabilidades:</b>    | Escuchar eventos del jugador a través del teclado, mouse y/o control de juegos. Responder a las entradas del usuario. Gestionar lógica de juego  |        |     |
| <b>Tipo:</b>                 | Sistema  |        |     |
| <b>Referencias cruzadas:</b> | R 4.9, 4.12, R 2.10, R 2.11,   |        |     |
| <b>Excepciones</b>           | El usuario cancela la partida  |        |     |
| <b>Salida:</b>               | Entradas de usuario, Interacción gráfica, Mensajes visuales, sonidos.  |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanciaron los recursos (objetos 3D) acordes al carácter seleccionado.</li> <li>✓ Se preparó el escenario virtual para el primer del juego relativo al carácter previamente seleccionado.</li> </ul>                 |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se crearon las instancias de gestión lógica del videojuego.</li> <li>✓ Se instanció un nuevo nivel (cuando el usuario culmine la partida dada)</li> <li>✓ Si se desea pausar, SeleccionarOpciones fue cargada.</li> </ul> |        |     |



#### 4.2.5.5. Contratos para Caso de Uso Configurar Juego

| Nombre                       | MostrarOpcionesConfigurarJuego()   | Código | CO9 |
|------------------------------|--|--------|-----|
| <b>Responsabilidades:</b>    | Mostrar al usuario las opciones para configurar el juego.  |        |     |
| <b>Tipo:</b>                 | Sistema  |        |     |
| <b>Referencias cruzadas:</b> | R 3.6  |        |     |
| <b>Notas:</b>                | Esta operación la puede realizar cualquier jugador, no es necesario iniciar sesión.  |        |     |
| <b>Excepciones</b>           | Ninguna  |        |     |
| <b>Salida:</b>               | Presentación de opciones de configuración de juego (video, controles y velocidad).   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se invoco a la operación SeleccionarOpcion(Opción)</li> <li>✓ Se instanció el despliegue de opciones</li> </ul> |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se presentaron opciones de configuración</li> </ul>   |        |     |

| Nombre                       | SeleccionaOpcionConfigurarJuego(Opcion)   | Código | C10 |
|------------------------------|---|--------|-----|
| <b>Responsabilidades:</b>    | Permitir la selección de una opción de configuración(Video, controles y velocidad )   |        |     |
| <b>Tipo:</b>                 | Sistema   |        |     |
| <b>Referencias cruzadas:</b> | R 13.6  |        |     |
| <b>Notas:</b>                | Sin notas.  |        |     |
| <b>Excepciones</b>           | Jugador cancela la selección de opciones.   |        |     |
| <b>Salida:</b>               | Opción de configuración seleccionada.   |        |     |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se presentaron opciones de configuración</li> </ul>  |        |     |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó una opción de configuración del juego.</li> <li>✓ Se creó una instancia de video (creación de instancia)</li> <li>✓ Se creó una instancia de controles (creación de instancia)</li> <li>✓ Se creó una instancia de velocidad(creación de instancia)</li> </ul> |        |     |



| Nombre                       | CancelarConfigurarJuego()  | Código | CO11 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Permitir la cancelación de la presentación de opciones de configuración del juego. |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 4.14   |        |      |
| <b>Notas:</b>                | Curso alternativo de Configurar Juego.   |        |      |
| <b>Excepciones</b>           | Ninguno  |        |      |
| <b>Salida:</b>               | Retorno a las opciones principales.  |        |      |
| <b>Precondiciones:</b>       | ✓ Se instanció el despliegue de opciones de configuración                          |        |      |
| <b>Poscondiciones:</b>       | ✓ Se instanció el despliegue de opciones principales.                              |        |      |

#### 4.2.5.6. Contratos para Caso de Uso Configurar Video

| Nombre                       | MostrarOpcionesConfigurarVideo()   | Código | CO12 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Obtener la información de la configuración actual de video<br>Permitir al jugador elegir el atributo a configurar.   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 4.14, R 2.7  |        |      |
| <b>Notas:</b>                | Debe de ser claro para el jugador que puede cancelar la función aquí descrita si lo desea.   |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarVideo()</i>   |        |      |
| <b>Salida:</b>               | Atributos modificados de video.  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó una opción de configuración del juego.</li> <li>✓ Se creó una instancia de video (creación de instancia)</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de video a configurar.</li> </ul>              |        |      |



| Nombre                       | SeleccionarOpcionConfigurarVideo (Resolucion)   | Código | CO13 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir la asignación de valores a los atributos de pantalla.  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 4.14. R 2.7   |        |      |
| <b>Notas:</b>                | Solamente se configura un atributo (resolución) del cual automáticamente se cargan el resto de atributos de video.  |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarVideo()</i>  |        |      |
| <b>Salida:</b>               | Valor asignado a Resolución   |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de video a configurar.</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron valores a los atributos de video</li> </ul>   |        |      |

| Nombre                       | ConfirmarCambiosVideo(Resolucion)   | Código | CO14 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir la asignación de valores a los atributos de pantalla.  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 2.7   |        |      |
| <b>Notas:</b>                | Solamente se configura un atributo (resolución) del cual automáticamente se cargan el resto de atributos de video.  |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarVideo()</i>  |        |      |
| <b>Salida:</b>               | Valor asignado a Resolución   |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de video a configurar.</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron valores a los atributos de video</li> </ul>   |        |      |



**4.2.5.7. Contratos para Caso de Uso Configurar Control**

| Nombre                       | MostrarOpcionesConfigurarControl()   | Código | CO15 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Obtener la información de la configuración actual de los controles de entrada.<br>Permitir al jugador elegir el atributo a configurar.   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 4.14. R 2.5  |        |      |
| <b>Notas:</b>                | Debe de ser claro para el jugador que puede cancelar la función aquí descrita si lo desea.   |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarControl()</i>   |        |      |
| <b>Salida:</b>               | Atributos modificados de controles.  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó una opción de configuración del juego.</li> <li>✓ Se creó una instancia de Controles (creación de instancia)</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de Controles a configurar.</li> </ul>              |        |      |

| .Nombre                      | SeleccionarOpcionConfigurarControl()  | Código | CO16 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir la asignación de valores a los atributos de pantalla.  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 4.14. R 2.5   |        |      |
| <b>Notas:</b>                |   |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarControl()</i>  |        |      |
| <b>Salida:</b>               | Valor asignado a cada Control   |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de los controles a configurar.</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron valores a los atributos de Control</li> </ul>   |        |      |



| Nombre                       | ConfirmarCambiosControl()  | Código | CO17 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Permitir la asignación de valores a los atributos de pantalla.   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 4.14. R 2.5  |        |      |
| <b>Notas:</b>                |  |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarControl()</i>   |        |      |
| <b>Salida:</b>               | Valor asignado a Resolución  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos del Control a configurar.</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron valores a los atributos de Control.</li> </ul>   |        |      |

#### 4.2.5.8. Contratos para Caso de Uso Configurar Velocidad

| Nombre                       | MostrarOpcionesConfigurarVelocidad()   | Código | CO18 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Obtener la información de la configuración actual de la velocidad del juego.   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 4.14. R 2.6  |        |      |
| <b>Notas:</b>                | Debe de ser claro para el jugador que puede cancelar la función aquí descrita si lo desea.   |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarVelocidad()</i>   |        |      |
| <b>Salida:</b>               | Velocidad modificada.  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó una opción de configuración de velocidad.</li> <li>✓ Se creó una instancia de Gestión de Velocidad (creación de instancia)</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de velocidad para configurar.</li> </ul>                         |        |      |





| Nombre                       | SeleccionarOpcionConfigurarVelocidad()  | Código | CO19 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir la asignación de valores a los atributos de pantalla.  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 2.6   |        |      |
| <b>Notas:</b>                |   |        |      |
| <b>Excepciones</b>           | El jugador cancela la configuración <i>CancelarConfigurarVelocidad()</i>  |        |      |
| <b>Salida:</b>               | Valor asignado a la Velocidad   |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de la velocidad para configurarla.</li> </ul> |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se asignaron valores a la velocidad.</li> </ul>  |        |      |

#### 4.2.5.9. Contratos para Caso de Uso Administrar Objetos Recolectados

| Nombre                       | MostrarObjetosRecolectados()  | Código | CO20 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Mostrar al usuario las los objetos adquiridos   |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 2.11, R 4.6, R 4.14   |        |      |
| <b>Notas:</b>                | Esta operación la puede realizar un jugador al indicarle así al sistema a través de un evento de entrada.   |        |      |
| <b>Excepciones</b>           | Ninguna   |        |      |
| <b>Salida:</b>               | Presentación de objetos recolectados en forma de modelo e información concerniente (Los objetos recolectados pueden ser comprados u obtenidos en el transcurso del juego).  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se invoco a la operación MostrarObjetosRecolectados</li> </ul>   |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se presentaron los Objetos Recolectados</li> <li>✓ Se selecciono uno o varios objetos recolectados</li> <li>✓ Se utilizó o consumió uno o varios objetos recolectados</li> </ul> |        |      |



**4.2.5.10. Contratos para Caso de Uso Controlar Caracter**

| Nombre                       | EscucharEventos()  | Código | CO21 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario Controlar el Caracter principal a través de eventos de entrada   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 2.10   |        |      |
| <b>Notas:</b>                |  |        |      |
| <b>Excepciones</b>           | Ninguna  |        |      |
| <b>Salida:</b>               |  |        |      |
| <b>Precondiciones:</b>       | ✓ Se llevo a cabo un evento de entrada   |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se Registró una nueva posición del Caracter.</li> <li>✓ Se Registró un cambio en la animación del Caracter.</li> <li>✓ Se Registró un cambio el comportamiento del Caracter.</li> </ul> |        |      |

| Nombre                       | ActualizarCaracter()  | Código | CO22 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario observar el cambio entre el entorno y el carácter debido a sus entradas para controlarlo  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 3.3, R 3.14,  |        |      |
| <b>Notas:</b>                |   |        |      |
| <b>Excepciones</b>           | Ninguna   |        |      |
| <b>Salida:</b>               | La posición y comportamiento del carácter es actualizada en una instancia en el tiempo  |        |      |
| <b>Precondiciones:</b>       | ✓ Se llevo a cabo un evento de entrada  |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se Actualizó la una nueva posición registrada del Caracter.</li> <li>✓ Se Actualizó un cambio en la animación del Caracter.</li> </ul> |        |      |



#### 4.2.5.11. Contrato para Caso de Uso Cargar Juego Almacenado

| Nombre                       | CargarJuego ()  | Código | CO23 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario cargar un juego almacenado  |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 3.3, R 2.8  |        |      |
| <b>Notas:</b>                |   |        |      |
| <b>Excepciones</b>           | Ninguna   |        |      |
| <b>Salida:</b>               | Se presenta la partida almacenada en un juego almacenado.   |        |      |
| <b>Precondiciones:</b>       | ✓ Se seleccionó la opción de cargar un juego almacenado.  |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se mostró una lista de juegos almacenados.</li> <li>✓ Se Seleccionó un juego almacenado.</li> <li>✓ Se Cargó un juego almacenado.</li> </ul> |        |      |

#### 4.2.5.12. Contrato para Caso de Uso Mostrar Bitácora

| Nombre                       | Mostrar Bitácora()  | Código | CO24 |
|------------------------------|---|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario mirar el avance de su juego   |        |      |
| <b>Tipo:</b>                 | Sistema   |        |      |
| <b>Referencias cruzadas:</b> | R 4.5   |        |      |
| <b>Notas:</b>                |   |        |      |
| <b>Excepciones</b>           | Ninguna   |        |      |
| <b>Salida:</b>               | Historial del jugador con respecto a puntos adquiridos, objetos recolectados, nivel de avance en cada nivel y niveles terminados.   |        |      |
| <b>Precondiciones:</b>       | ✓ Se seleccionó la opción de Mostrar Bitácora desde el menú de opciones principales.  |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanció una ventana de presentación de la bitácora.</li> <li>✓ Se presentaron al usuario la información de bitácora e información de avance.</li> </ul> |        |      |



#### 4.2.5.13. Contrato para Caso de Uso Crear Usuario

| Nombre                       | CrearUsuario()   | Código | CO25 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario crear una nueva cuenta de usuario  |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 2.1  |        |      |
| <b>Notas:</b>                |  |        |      |
| <b>Excepciones</b>           | El usuario no es validado.   |        |      |
| <b>Salida:</b>               | Validación de Usuario y creación de nueva cuenta.  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó la opción de Crear Nuevo Usuario desde el menú de opciones principales.</li> </ul>  |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanció una ventana para entrada de datos.</li> <li>✓ Se introdujeron los datos del nuevo usuario.</li> <li>✓ Se valida al usuario</li> <li>✓ Se crea una nueva cuenta.</li> </ul> |        |      |

#### 4.2.5.14. Contrato para Caso de Uso Eliminar Usuario

| Nombre                       | EliminarUsuario()  | Código | CO26 |
|------------------------------|--|--------|------|
| <b>Responsabilidades:</b>    | Permitir al Usuario eliminar una cuenta de usuario   |        |      |
| <b>Tipo:</b>                 | Sistema  |        |      |
| <b>Referencias cruzadas:</b> | R 2.2, R 2.3   |        |      |
| <b>Notas:</b>                |  |        |      |
| <b>Excepciones</b>           | El usuario cancela la función de eliminar su cuenta.   |        |      |
| <b>Salida:</b>               | Eliminación de Cuenta de Usuario.  |        |      |
| <b>Precondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se seleccionó la opción de Eliminar Usuario desde el menú de opciones principales.</li> </ul>   |        |      |
| <b>Poscondiciones:</b>       | <ul style="list-style-type: none"> <li>✓ Se instanció una ventana para la presentación de eliminar usuario.</li> <li>✓ Se introdujeron los datos del usuario.</li> <li>✓ Se validó al usuario</li> <li>✓ Se eliminó la cuenta de usuario.</li> </ul> |        |      |



### 4.3. MODELO DEL DOMINO

El Modelo de Dominio (o Modelo Conceptual) es una representación visual de los conceptos u objetos del mundo real significativos para un problema o área de interés. Representa clases conceptuales del dominio del problema. Muestra en forma gráfica conceptos del mundo real, no de los componentes de software.

Una clase conceptual puede ser una idea o un objeto físico (símbolo, definición y extensión). El modelo de dominio se representa en UML con un Diagrama de Clases en los que se muestra:

- Conceptos u objetos del dominio del problema: clases conceptuales
- Asociaciones entre las clases conceptuales
- Atributos de la clase conceptuales

En el Modelo de Dominio no se muestra comportamiento. Las clases conceptuales pueden tener atributos pero no métodos. Un modelo de dominio explica a los desarrolladores los conceptos significativos en el dominio del problema; es una de las herramientas más importante durante el análisis orientado objeto<sup>93</sup> (el cual es el empleado en este proyecto).

Características:

- No hay cronología
- No se diferencia entre dentro y fuera del sistema
- Es global, no por caso de uso
- No es completo: esquemático, las asociaciones están resumidas.

---

<sup>93</sup> El modelo del dominio es el artefacto más importante a crear durante la etapa de análisis orientado a objetos. UML y Patrones de Craig Larman Edición 2001, Cap. 9, pág. 85



### 4.3.1. Modelo de Dominio del Videojuego.

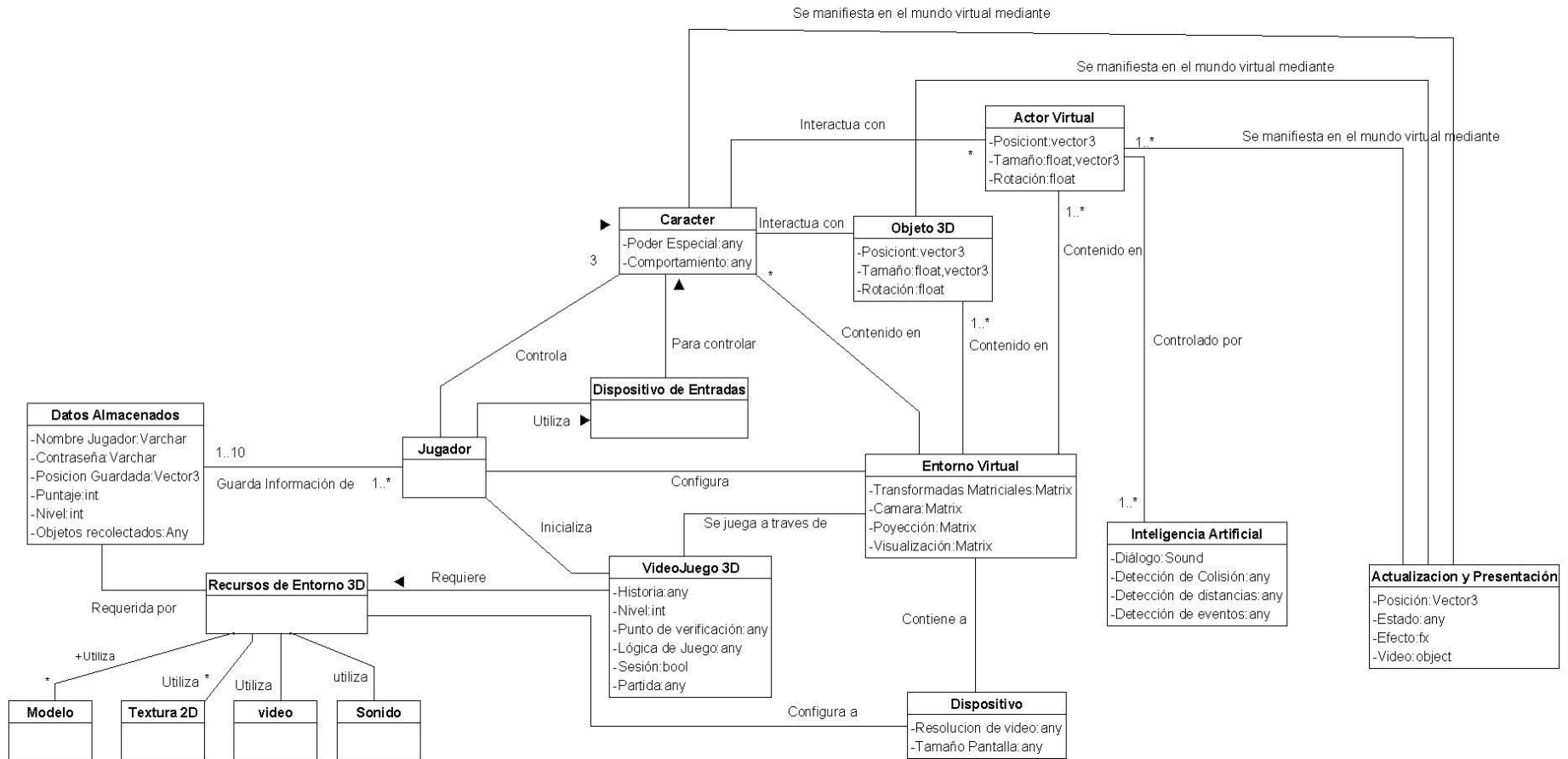


Figura 4.4: Modelo de dominio



### 4.3.2. Organización del Modelo del Dominio en Paquetes

Los diagramas de paquetes en UML 2.0 simplifican diagramas de clases más complejos, se pueden agrupar las clases (así como las clases conceptuales de un modelo de dominio) en paquetes. Un paquete es una colección lógicamente relacionada con elementos del modelado UML. Los paquetes se representan como archivos o carpetas y pueden ser utilizados para encapsular cualquier diagrama UML.<sup>94</sup>

Los paquetes pueden representar directamente la estructura lógica de los archivos de código de programas orientados a objetos como C#, C++, Visual Basic entre otros, ya que el *namespace*<sup>95</sup> de cada archivo de código, es un paquete lógico que encierra una o varias clases. Las clases que comparten el mismo *namespace* pueden comunicarse directamente entre sí, e instanciar objetos de otras clases dentro del mismo paquete lógico.

La siguiente figura encapsula al modelo de dominio del sistema propuesto en cinco paquetes principales:

- P1. Manipulación y Control
- P2. Recursos
- P3. Elementos 3D
- P4. Elementos 2D
- P5. Entorno
- P6. Gestión Lógica.
- P7. EfectosDeParticulas

---

<sup>94</sup> Jung Woo, Diagramas UML: <http://www.cs.umbc.edu/courses/undergraduate/CMSC445/Fall06/UML%20Diagrams.ppt>

<sup>95</sup> namespace o paquete lógico, encapsula un número de clases y estructuras para que estas puedan comunicarse entre sí durante el desarrollo, pruebas y ejecución de un sistema software.



### 4.3.2.1. Encapsulamiento en paquetes

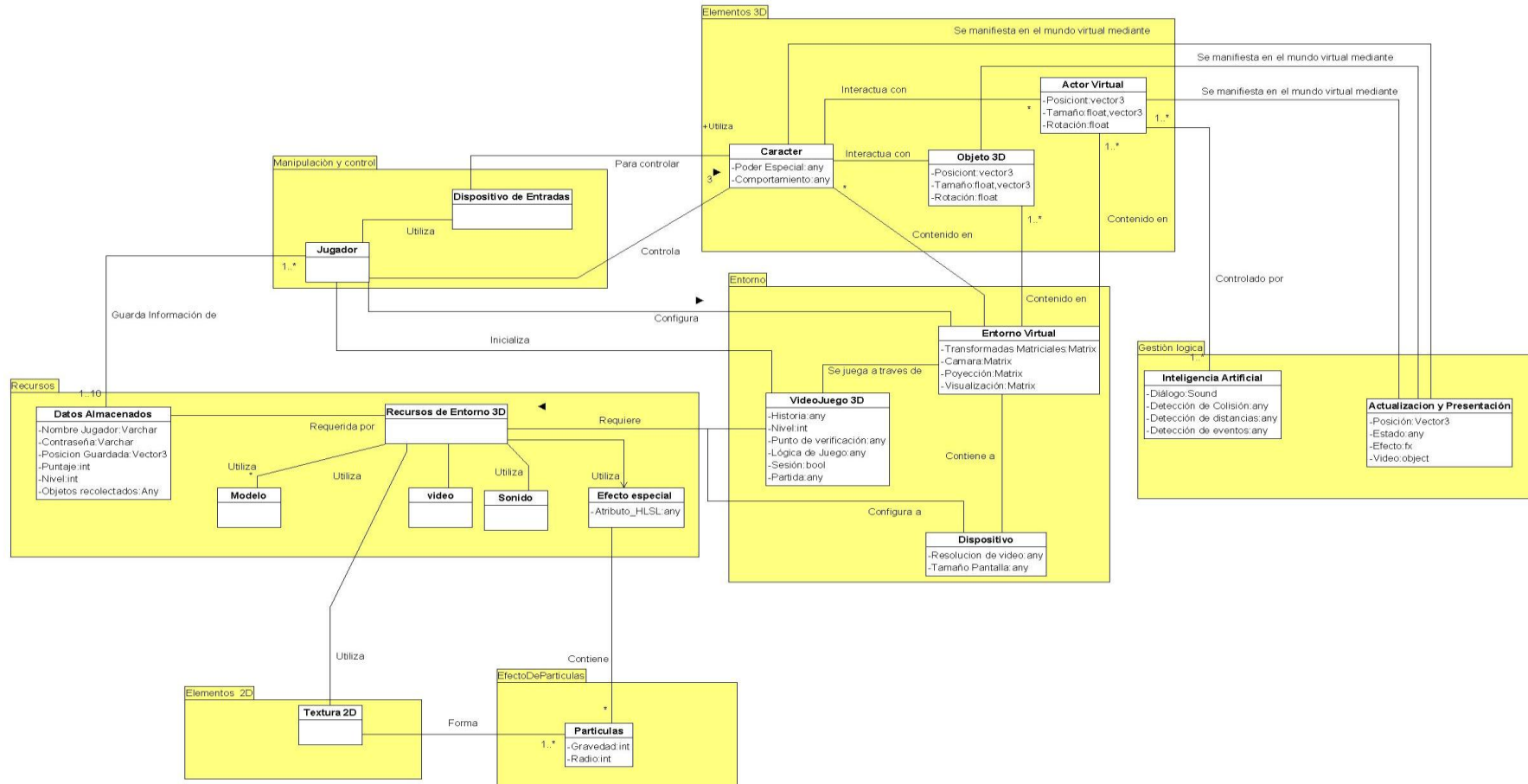


Figura 4.5: Encapsulamiento en paquete





La que se ha generado una representación de mayor generalidad del modelo del dominio de software propuesto, el siguiente paso es mostrar el videojuego en su totalidad como un paquete que encapsula a todas las clases conceptualizadas hasta el momento para el diseño del videojuego. La siguiente figura muestra dicho encapsulamiento:

### DIAGRAMA DE PAQUETES DE SIEMPOSAL

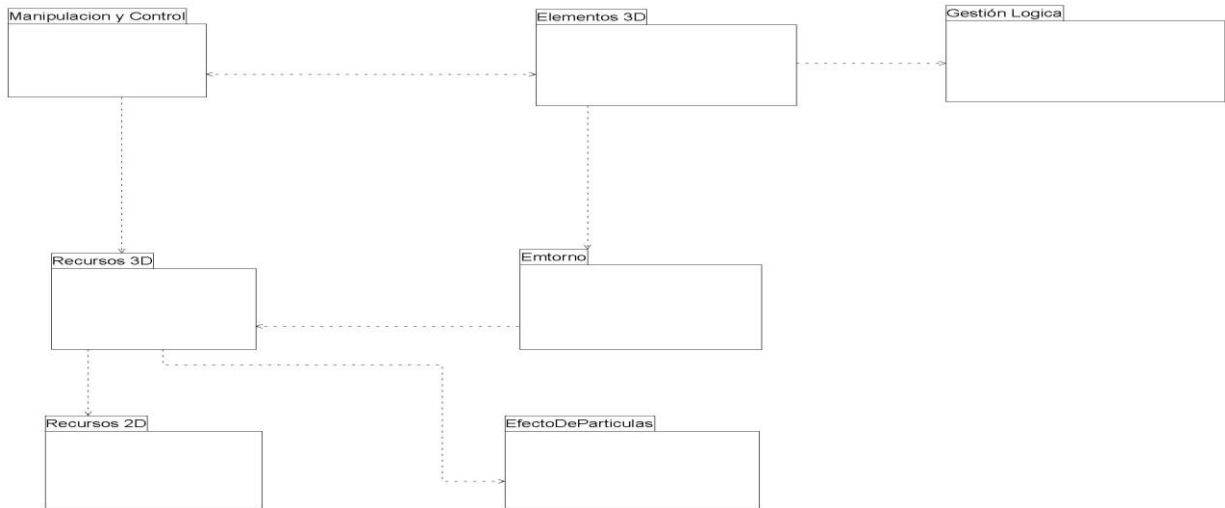


Figura 4.6: Paquetes de SIEMPOSAL

### 4.4. DICCIONARIO DE DATOS DE LAS CLASES CONCEPTUALES

| Clase                       | Atributo             | Tipo de datos | Descripción   |
|-----------------------------|----------------------|---------------|---|
| Gestión de DatosAlmacenados | Nombre Jugador       | VarChar       | Valor que representa el nombre del jugador que inicia una sesión de juego.                              |
|                             | Contraseña           | VarChar       | Cualquier valor tipo cadena de caracteres que representa la clave del jugador.                          |
|                             | Posición Guardada    | Vector3       | Variable de estructura de vector con los componentes x, y, z que representan un punto en el espacio 3D. |
|                             | Puntaje              | int           | Valor del avance de puntaje otorgado al jugador en una sesión de juego dada.                            |
|                             | Nivel                | int           | Valor del avance del nivel de jugador.  |
|                             | Objetos recolectados | Estructura    | Estructura de varios tipos de objetos que el jugador a recolectado, etc.                                |



| Clase           | Atributo               | Tipo de datos | Descripción  |
|-----------------|------------------------|---------------|--|
| Jugador         | Sin especificar        |               |  |
| Carácter        | Poder Especial         | Estructura    | Conjunto de valores de varios tipos que identifican a un determinado carácter de los demás, las diferentes combinaciones de elementos de la estructura determinan las capacidades de cada carácter para volar, desplazarse más rápido, rotar, fuerza, etc. |
|                 | Comportamiento         | Estructura    | Conjunto de valores de diferentes tipos que dictan un comportamiento único para cada carácter.   |
| Actor Virtual   | Posición               | Vector3       | Variable de estructura de vector con los componentes x, y, z que representan un punto en el espacio 3D.  |
|                 | Tamaño                 | Flotante      | Determina el tamaño o escala de cada actor virtual en el entorno 3D  |
|                 | Rotación               | Flotante      | Variable que especifica la rotación del actor virtual, la rotación puede ser sobre el eje Y, sobre el eje X, sobre el eje Z, o sobre un eje arbitrario especificado.   |
| Objeto 3D       | Posición               | Vector3       | Variable de estructura de vector con los componentes x, y, z que representan un punto en el espacio 3D.  |
|                 | Tamaño                 | Flotante      | Determina el tamaño o escala de cada objeto 3D en el entorno 3D  |
|                 | Rotación               | Flotante      | Variable que especifica la rotación del objeto 3D, la rotación puede ser sobre el eje Y, sobre el eje X, sobre el eje Z, o sobre un eje arbitrario especificado.   |
| Entorno Virtual | Transformada Matricial | Matrix        | Estructura que representa cualquier transformada aplicada al entorno virtual, la transformación puede ser de escalación, rotación, traslación, y una combinación de estas.   |
|                 | Cámara                 | Matrix        | Estructura matricial para manipular una cámara virtual manipulada por el jugador a través de sus entradas y control del personaje principal.   |
|                 | Proyección             | Matrix        | Estructura matricial para controlar el aspecto, ángulo y alcance de visualización del entorno a través de la cámara.   |
|                 | Visualización          | Matrix        | Permite enfocarse en determinadas coordenadas cartesianas del espacio 3D, usualmente el personaje principal basado en la posición de la cámara virtual y la posición del objeto o posición de enfoque.   |



| Clase   | Atributo                | Tipo de datos          | Descripción   |
|---|-------------------------|------------------------|---|
| <b>Gestión de Inteligencia Artificial</b>         | Diálogo                 | Sound                  | Variable tipo sound que almacena y emite los sonidos y diálogos de los actores virtuales y carácter principal.  |
|   | Detección de Colisión   | Estructura             | Conjunto de objetos de gestión de detección de colisiones, entre estos se tienen: Arreglos índices, arreglos de vértices, variables dobles y flotantes para almacenar ángulos, distancias y vectores de posición, velocidad y dirección, así como variables booleanas o switches que se activan cuando existe una colisión con un determinado objeto del entorno virtual. |
|   | Detección de Distancias | Flotante               | Guarda la distancia entre objetos, actores virtuales y el personaje principal a través de la ecuación de Pitágoras.   |
|   | Detección de Eventos    | Array                  | Arreglo de booleanas que actúan como switches que se encienden y apagan cuando un determinado evento ocurre o cuando una condición se cumple.   |
| <b>Actualización y Presentación.</b>              | Posición                | Vector3                | Variable de estructura de vectores que guarda un valor actualizado de la posición de los objetos 3d, actores virtuales y personaje principal.   |
|   | Estado                  | Estructura             | Guarda valores actualizados para el estado de presentación del dispositivo, como por ejemplo, efectos especiales, filtros isotrópicos, empañado, nitidez y otros efectos de presentación.   |
|   | Efecto                  | Effect                 | Estructura tipo Effect determinada por un archivo de efectos en donde se especifican múltiples valores dinámicos como iluminación, transformación, filtros, color y posición de luz entre otros.  |
| <b>Dispositivo de Entradas</b>                    | <i>Sin especificar</i>  | <i>Sin especificar</i> | El dispositivo de entrada puede ser el teclado, el mouse, un gamepad o un control genérico, un joystick o una combinación de estos.   |
| <b>Gestión de Carga de Recursos de Entorno 3D</b> | <i>Sin especificar</i>  | <i>Sin especificar</i> | Esta clase conceptual será la encargada de cargar todos los recursos necesarios para la creación del entorno virtual y su población con objetos 3D y actores virtuales.   |
| <b>Modelo</b>                                     | Modelo                  | Model                  | Estructura tipo Model que permite guardar en memoria y presentar todo tipo de modelo 3D cargado de un archivo.  |
| <b>Sprite</b>                                     | SpriteBatch             | Texture                | Objeto tipo Texture para dibujar texturas 2D o simular explosiones, humo, fuego, rayos, etc.  |



| Clase                  | Atributo              | Tipo de datos          | Descripción  |
|------------------------|-----------------------|------------------------|--|
| <b>Efecto Especial</b> | HLSL                  | Estructura             | Estructura de código HLSL que permite especificar una multitud de efectos aplicados a diferentes elementos que conforman el entorno virtual.   |
| <b>Videojuego</b>      | Historia              | <i>Sin especificar</i> | Relata la historia en la cual el jugador será envuelto.  |
|                        | Nivel                 | Int                    | Especifica el nivel de avance durante el juego.  |
|                        | Punto de Verificación | <i>Cualquiera</i>      | Indica que el jugador ha alcanzado un estado que le califica para pasar de nivel.  |
|                        | Lógica                | Estructura             | Consiste en la estructura o arquitectura lógica del juego en su totalidad. Compuesta de operaciones de inicialización, actualización, presentación en pantalla y liberación de recursos. |
|                        | Sesión                | Bool                   | Se le asigna un valor de 'verdadero' solamente si el usuario ha sido validado en la gestión de inicio de sesión.   |
|                        | Partida               | Estructura             | Objeto tipo estructura en donde se asignan diferentes valores como el nivel de juego, nivel de vida, objetos recolectados, avance, entre otros.  |
| <b>Dispositivo</b>     | Resolución de video   | Arreglo de flotantes   | Se guardan los valores del número de pixeles, que se han configurado para la presentación del entorno 3D, ya sea pantalla completa o Ventana   |
|                        | Tamaño pantalla       | Arreglo de flotantes   | Se guardan los valores del tamaño de pantalla cuando se implementa una ventana.  |

Cuadro 4.10: Diccionario de datos del modelo conceptual

#### 4.5. CONCLUSION DE LA FASE DE ANÁLISIS.

En esta fase de análisis se ha dado prioridad al conocimiento de los requerimientos, conceptos y operaciones potenciales relacionadas al software aquí descrito. En la mayoría de casos la investigación y análisis se caracterizan<sup>96</sup> por centrarse en cuestiones concernientes al *qué*: cuales son los procesos, conceptos, etc. y no al *cómo*.

<sup>96</sup> Craig Larman, sobre análisis de sistemas, UML y Patrones: Cap. 15, pág. 161.



## **CAPITULO V: DISEÑO DEL SOFTWARE**



## 5. DISEÑO DEL SOFTWARE

Existe en la actualidad innumerables libros sobre el “análisis y diseño”, hoy más que nunca hay mucha información de diferentes autores sobre el tema y su “orientación a objetos”. El término compuesto automáticamente indica que el autor no comprende en realidad el propósito del análisis y diseño de un sistema.

No existe realmente una disciplina o actividad llamada “Análisis y Diseño”. Además, una distinción clara es absolutamente necesaria entre esas actividades para cualquier desarrollo de software de grandes dimensiones.<sup>97</sup>

En el capítulo anterior se ha presentado de forma, tanto textual como gráfica, la fase de *análisis del sistema propuesto*, y se ha caracterizado por ilustrar de forma clara, implementando las reglas del UML 2.0 el **qué** se desea hacer, sin especificar el **cómo**. En síntesis se ha presentado las especificaciones del software en forma primitiva, no obstante es fundamental para esta etapa de diseño en donde se refinará cada aspecto y se presentará su forma final que servirá como “planos arquitectónico” del software en su fase de desarrollo.

En este capítulo, se desarrolla la fase de *diseño* que es donde se muestra la arquitectura del software compuesta por la plataforma de diseño. Como se ha mencionado, es necesario hacer y tener una distinción clara entre las fases de análisis y diseño, para exponer estas diferencias, el siguiente cuadro muestra las características de cada una de ellas.

|   | Fase de Análisis Cap. IV  | Fase de Diseño Este Capitulo  |
|---|---|---|
| <b>Realizado por</b>  | Analista de Sistemas  | Jefe de Programación<br>(o Arquitecto de software o desarrollador en cargo)   |
| <b>Salidas de la Fase</b>   | Especificaciones del software:<br>Todo sobre <i>el qué</i> hará el software propuesto, nada sobre <i>el cómo</i> se va a construir. | Arquitectura del Software: Plataforma de desarrollo seleccionada, diseño del programa/código fuente, diseño de interfaces de usuario, manejo de datos, estándares de desarrollo, etc. |
| <b>Audiencia que lee, comprende y acepta las salidas u objetivos de la Fase</b> | 1. Usuarios Inversionistas/patrocinadores<br>2. Jefe de programación<br>(o Arquitecto de software o desarrollador en cargo)         | Programadores y especialistas técnicos.   |

Cuadro 5.1: Objetivo del análisis y diseño.

<sup>97</sup> Extracto de: "Analysis and Design" Considered Harmful (análisis y diseño que es considerado peligroso) Frequent pairing of different disciplines violates fundamental life-cycle principle.(El apareamiento frecuente de diferentes disciplinas viola principios fundamentales del ciclo de vida de proyectos) © Conrad Weisert, Information Disciplines, Inc. February, 1999



Una de las causas más comunes del fracaso de un proyecto<sup>98</sup> es la falta de especificaciones del sistema completas, no-ambiguas y comprensibles.

## 5.1. DISEÑO GLOBAL

### 5.1.1. Estándares para el diseño de software

Antes de proseguir, es necesario aclarar el término estándar. Según el diccionario de la Real Academia Española, un Estándar es:

(Del ingl. standard).

1. adj. Que sirve como tipo, modelo, norma, patrón o referencia.
2. m. Tipo, modelo, patrón, nivel. Estándar de vida.

Según el diccionario Webster de la lengua Inglesa:

**“Un estándar es:** una forma para determinar como una cosa debe de ser. Aplica a cualquier regla, principio o medida definitiva establecida por una autoridad”.

Un estándar para el diseño de software se puede definir por lo tanto como una regla establecida y a la cual todo el personal involucrado en el diseño y desarrollo de un software debe de regirse. El uso de estándares en la fase de diseño facilita la comprensión del sistema y su desarrollo para todos los entes involucrados, minimizando ambigüedades y estableciendo el uso de términos, formatos, operaciones comunes de diseño y desarrollo. Es por lo tanto indispensable establecer dichas reglas en esta etapa para parametrizar<sup>99</sup> y estandarizar todos los mecanismos y procesos que conlleven al diseño y desarrollo del proyecto actual. Entre los estándares que a continuación serán descritos se encuentran:

- Estándares de interfaz de salida.
- Estándares de interfaz de entrada.
- Estándares de menús.
- Estándares de código fuente.
- Estándares de gestión de información.

### 5.1.2. Estándares de interfaz de salida

La razón detrás de la implementación de estándares de interfaz de salida es el factor humano, ya que se sabe que la forma en que este interactúa con el software es de vital

---

<sup>98</sup> La otra causa más común es la falta de suficiente detalle en la planeación del proyecto que permita detectar tempranamente sobre- requerimientos o escases de estos.

<sup>99</sup> Jerga informática que significa: estructurar de forma acorde a las necesidades (o parámetros requeridos). No se encuentra aun en el Diccionario de la Real Academia Española.



importancia para su adaptación, aprendizaje y utilización del producto final. La interfaz de salida del software propuesto es descomunal en el ámbito de sistemas convencionales, ya que el hecho de tratarse de un juego de video, la interfaz de salida es desde donde se contará una historia, el jugador se involucrará e interactuará con el mundo virtual que se le presenta y tendrá a su disposición todos los recursos e información necesarios para que su experiencia con el sistema 3D sea amena y motivadora para avanzar a niveles superiores de juego.

En la actualidad existen diferentes estándares para las pantallas de salida de videojuegos, y dependiendo de la historia que se cuente, de los recursos disponibles para el jugador y del entorno del juego ya sea 2D o 3D, esta interfaz variará dependiendo de las necesidades propias del juego de video. Lo importante no es implementar un estándar idéntico a los más comunes de la industria, sino que de diseñar un estándar que permita al jugador del software propuesto disfrutar al máximo de su experiencia e involucrarse en la historia que le envuelve. Un videojuego consta de diferentes pantallas, como por ejemplo la pantalla de configuración, un menú principal, pantalla de objetos recolectados, pantalla de carga de juegos almacenados y la pantalla principal de juego que es donde el usuario interactúa directamente con el entorno virtual presentado, en esta última, un juego típico 3D incluye la información que se considera de importancia para el jugador, ver siguiente figura.



Figura 5.1: Estándar de interfaz de salida





### 5.1.2.1. Entorno Virtual Escenario 1

Como estándar de interfaz de salida para el entorno virtual se implementará el siguiente esquema, aplica a todas las pantallas en donde el estado del jugador no interactúa con ningún objeto o actor virtual del juego:



Figura 5.2: Estándar de interfaz de salida sin interacción con objetos.

### 5.1.2.2. Entorno Virtual Escenario 2

Se implementará el siguiente esquema para todas las pantallas en donde el estado del jugador se encuentra interactuando con algún objeto.

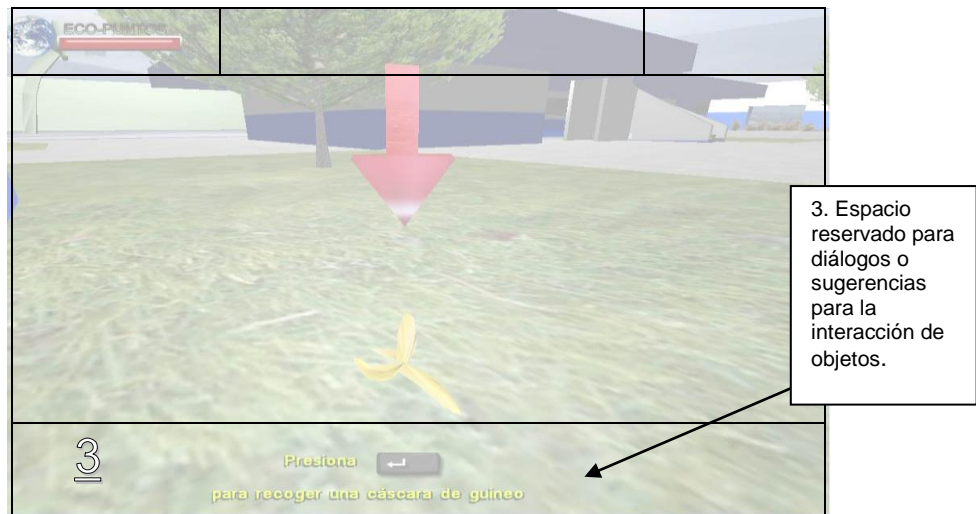


Figura 5.3: Estándar de interfaz de salida con interacción



### 5.1.2.3. Entorno Virtual Escenario 3

Se implementará el siguiente esquema para todas las pantallas en donde se requiere una determinada interacción entre el usuario y objetos estáticos



Figura 5.4: Estándar de interfaz de salida interacción entre usuarios y objetos estáticos.

### 5.1.2.4. Entorno Virtual Escenario 4

Se implementará el siguiente esquema para todas las pantallas en donde se requiere una determinada interacción entre el jugador, un vendedor y artículos en venta según la historia del juego.



Figura 5.5: Estándar de interfaz de salida selección de objetos.



### 5.1.2.5. 5.1.2.5 Entorno Virtual Escenario 5

Se implementará el siguiente esquema para todas las pantallas en donde se requiere una determinada interacción entre el jugador y el consumo o utilización de artículos .



Figura 5.6: Estándar de interfaz de salida.

### 5.1.3. Estándares de interfaz de Entrada

Los juegos de video pueden utilizar diferentes interfaces de entrada; hardware hecho especialmente para un determinado tipo de juego o una combinación de varias interfaces. Para este software se implementaran 4 tipos de interfaces de usuario:

Un teclado, para la versión destinada a las PCs de las aulas informáticas de instituciones educativas del país.

Un Mouse, para el mismo propósito del literal anterior.

Un Gamepad para la versión XBOX-360 (Opcional)

Un Joystick Genérico para versiones PC cuyos usuarios opcionalmente deseen usar.(Opcional)

#### 5.1.3.1. Estándares para el teclado:

La funcionalidad de las teclas podrá ser personalizada por el jugador, de acuerdo a sus preferencias.



Figura 5.7: Teclado - Estándar de interfaz de entrada.

La funcionalidad asignada por defecto a cada tecla se lista a continuación:

**1. Mayúsculas:** Reservada para eventos interactivos como:

- Seleccionar artículos en venta.
- Visión Nocturna (para niveles en escenarios futuros).
- Consumir comestible, utilizar objeto especial.

**2. W, A, S, D:** Dirección adelante, atrás, izquierda, derecha.

**3. F12:** Alternar entre primer persona y vistas en tercera persona.

**4. Insertar:** Cambiar a modo de consumo de comestibles.

**5. Control:** Reservada para eventos de lanzamiento de objetos

- Disparar esferas de oxígeno (para niveles en escenarios futuros).
- Lanzar objetos recolectados.
- Lanzar pelota de basketball (para niveles en escenarios escolares).
- Patear pelota de football (para niveles en escenarios escolares).

**6. Barra Espaciadora:** Saltar, nadar.

**7. Enter:** Reservada para acciones de interacción.

- Recoger desechos sólidos.
- Establecer conversaciones con actores virtuales.



- Cambiar a modo de compra de artículos.
- Abrir puertas, etc.

**8. Teclas *Direccionales*:** Reservada para las acciones siguientes.

- Ángulos de rotación.
- Eje “Y “ y eje “X” desde la perspectiva del jugador.
- Iterar sobre artículos en venta.
- Iterar sobre objetos recolectados.
- Iterar sobre comestibles y
- Navegar por opciones de menús.

**9. *Atrás*:** Salir de pantalla de submenús (durante el juego).

**10. *F1*:** Entrar a pantalla de submenús (durante el juego).

Estos han sido los valores iniciales y configuración de teclas programadas, no obstante, el usuario tendrá la opción de reconfigurar cualquier tecla que no esté reservada por el software para personalizar su experiencia de respuesta en el juego.

**5.1.3.2. *Estándares para el Mouse:***

El mouse no podrá ser reconfigurado, y se utilizará para la selección de objetos, menús y ángulos de rotación en los ejes Y y X. Será utilizado opcionalmente en conjunto con el teclado para las versiones de PC del videojuego.

Botón Izquierdo:

- Disparar esferas de oxígeno (para niveles en escenarios futuros)
- Lanzar objetos recolectados
- Acelerar vehículo
- Recoger desechos sólidos
- Seleccionar opciones de menú

**Botón *Derecho*:** Reservado para acciones de interacción.

- Arrancar motores del ecobote en el río Lempa



### 5.1.3.3. Estándares del GamePad: (Dispositivo opcional).



Figura 5.8: GamePad - Estándar de interfaz de entrada (Opcional).

1. **Pad:** Reservada para eventos interactivos como:

- Iterar sobre artículos en venta,
- Iterar sobre objetos recolectados,
- Iterar sobre comestibles.
- Navegar por opciones de menús.

2. **Palanca Izquierda:** Adelante, atrás, izquierda, derecha.

3. **Back:** Salir de pantalla de submenús (durante el juego).

4. **Start:** Entrar a pantalla de submenús (durante el juego).

5. **Botones:** Reservados para eventos de lanzamiento de objetos y para acciones de interacción.

**Botón A:** Saltar, nadar.



**Botón B:** Sin especificar, se dejará sin función hasta que el desarrollo amerite su programación.

Botón X:

- Recoger desechos sólidos
- Establecer conversaciones con actores virtuales
- Cambiar a modo de compra de artículos.
- Abrir puertas, etc.
- Seleccionar artículos en venta
- Consumir comestible, utilizar objeto especial.

**Botón Y:** Alternar entre Primer Persona y Vistas en tercera persona.

**6. Palanca Derecha:** Ángulos de Rotación, eje Y y eje X desde la perspectiva del jugador.

7. Gatillo izquierdo: Corre

8. Gatillo derecho:

- Disparar esferas de oxígeno (para niveles en escenarios futuros)
- Lanzar objetos recolectados
- Lanzar pelota de basketball (para niveles en escenarios escolares)
- Patear pelota de football (para niveles en escenarios escolares)

**9. Hombro Izquierdo:** Visión Nocturna (para niveles en escenarios futuros)

**10. Hombro Derecho:** Utilizar objeto especial.

#### **5.1.3.4. Estándares del Joystick Genérico:**

Igual que el GamePad.

#### **5.1.4. Estándares de menús**

Cada menú de selección presentado al jugador (incluyendo el menú principal) se presentará de la siguiente manera:



Figura 5.9: Estándar de interfaz de menús

Cuando se haya seleccionado una opción, pero todavía no se haya aceptado, se hará énfasis aumentando el tamaño, cambiando el color y desplazando la posición hacia la derecha de la posición inicial de la opción actual.



Figura 5.10: Estándar de interfaz de menús





### 5.1.5. Estándares de Código Fuente

La codificación del software se realizará utilizando Visual C#. Como se ha descrito, este es un lenguaje exclusivamente orientado a objetos, por lo tanto los estándares harán uso de tales características.

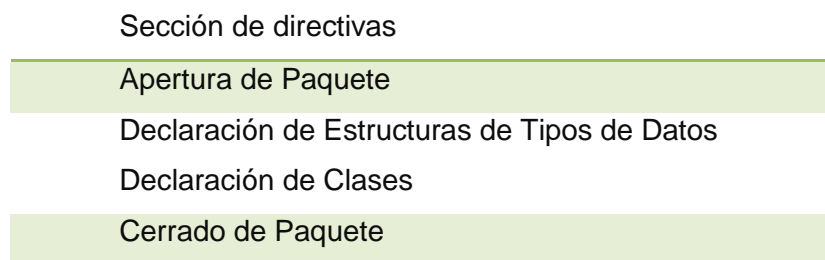
Entre los elementos del lenguaje de programación y código que se estandarizará se mencionan:

- Encapsulamiento o paquetes (namespace)
- Estructuras de clases
- Características de objetos y variables
- Nombres de objetos y variables
- Características y estructuras de funciones.
- Utilización de bibliotecas predefinidas
- Seccionamiento y ocultación de código.
- Documentación del Código

#### 5.1.5.1. Estándar de Encapsulamiento o paquetes (namespace)

El tipo de encapsulamiento o empaquetado a implementar es lógico, es decir, que muchas clases y estructuras podrán ser contenidas en archivos diferentes, físicamente separados, no obstante, si comparten el mismo paquete las clases contenidas podrán comunicarse entre sí e instanciar objetos de estas.

El formato a seguir para cada archivo de código es el siguiente:



Cuadro 5.2: Estructura de código fuente

Las siguientes figuras muestran gráficamente el formato implementado

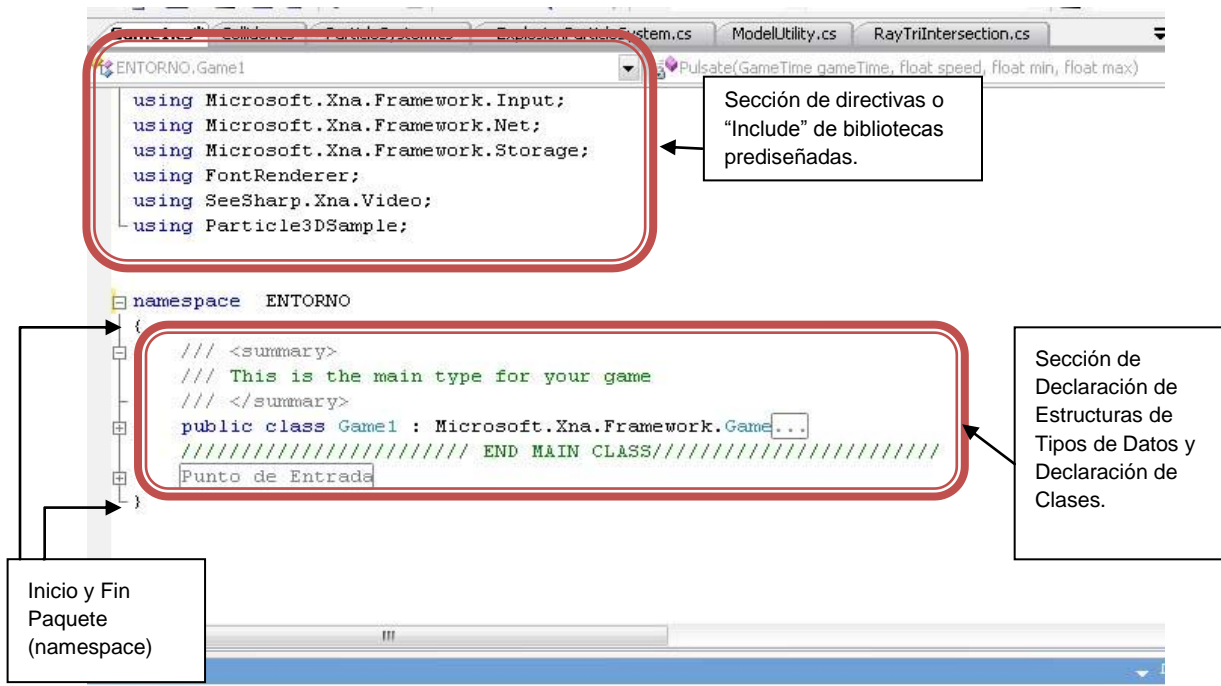


Figura 5.11: Estándar de código fuente, estructura de paquetes

La figura anterior muestra el esquema que se seguirá para cada paquete lógico, en la sección de declaración de estructuras de tipos de datos, podrán incluirse solamente una o más clases, solamente o más estructuras de tipos de datos, o una combinación de ambas.

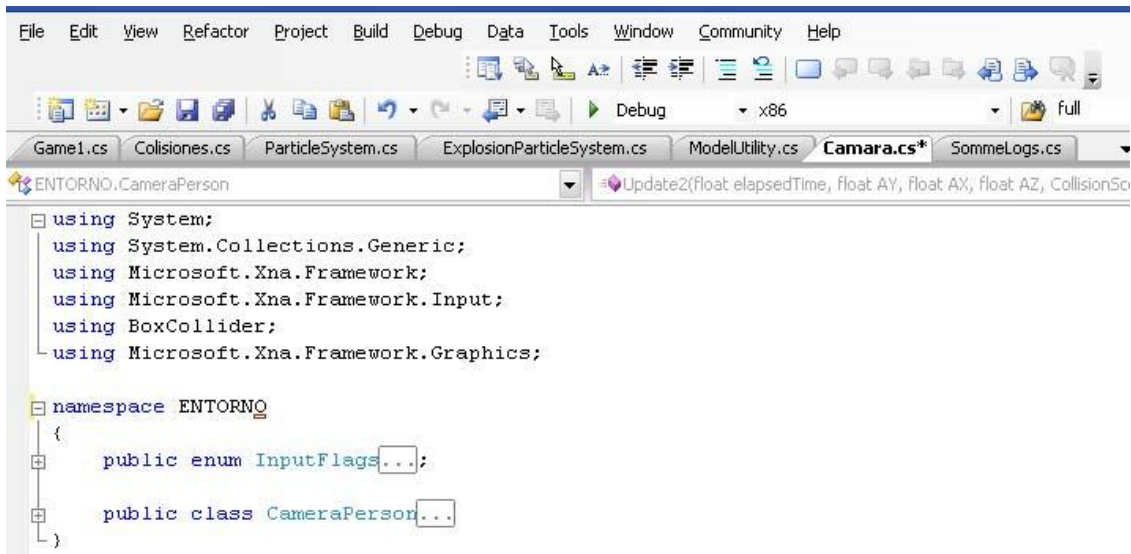


Figura 5.12: Estándar de código fuente, estructura de paquetes



En las figuras anteriores se presentan dos archivos de código diferentes: Camara.cs y Game1.cs, como ambos archivos comparte el mismo paquete “ENTORNO”, las estructuras y clases que contiene podrán ser conocidas por todas las clases que contenga cada archivo.

### 5.1.5.2. Estándar para Estructuras de clases

Para cada clase que componga el código fuente de la solución se seguirá el siguiente estándar:

- Nombre de la clase (apertura) y herencia si aplica.
- Sección de variables globales
- Constructor de la clase
- Declaración de métodos
- Cerrado de la clase

Cuadro 5.3: Estructura de clases.

La siguiente figura muestra gráficamente el formato implementado

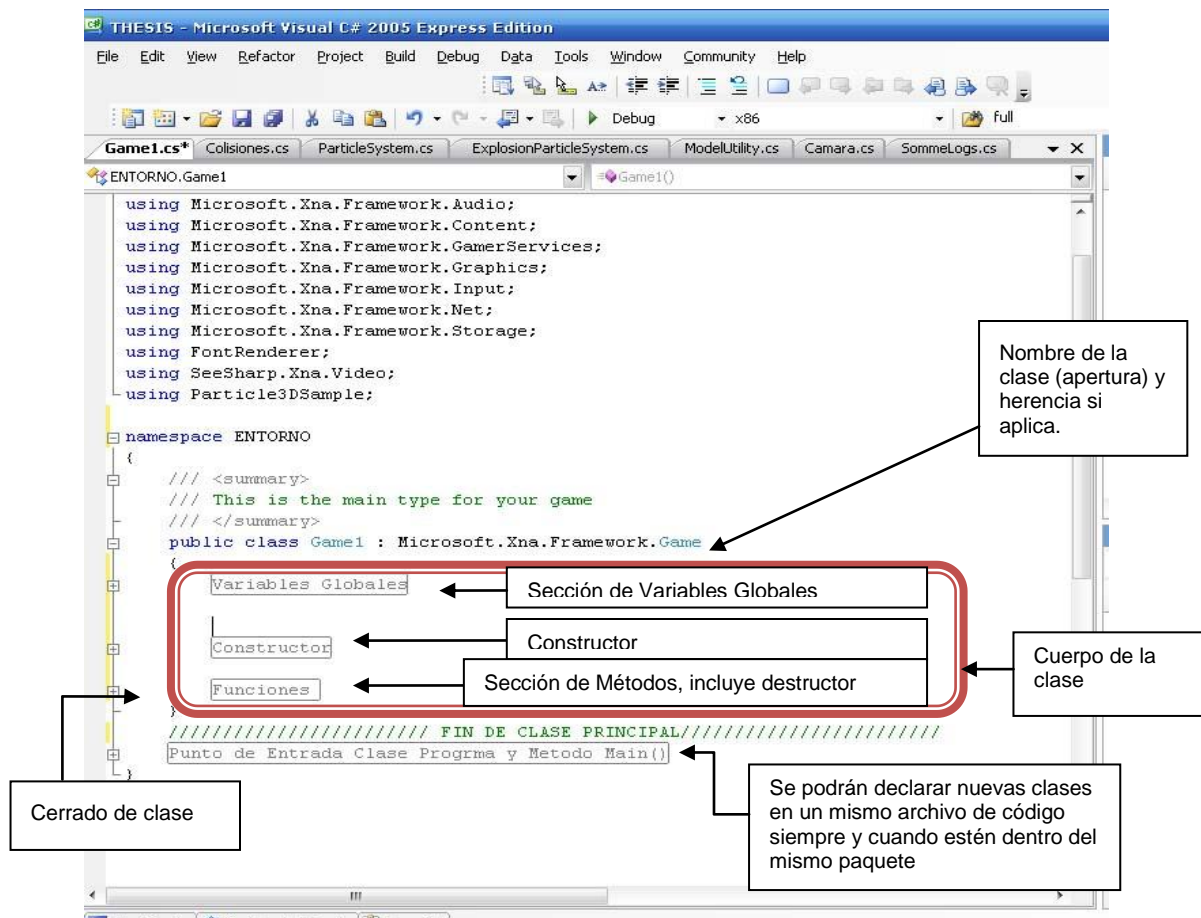


Figura 5.13: Estándar de código fuente, estructura de clases.



**5.1.5.3. Características de objetos y variables: Estándares para cada objeto o variable**

| Tipo de Objeto/Variable                | Posicionamiento  | Ejemplo  |
|--|--|--|
| <b>Global</b>                          | Siempre se declararán o inicializarán antes del constructor de la clase  | <pre> ... Vector3 Posicion; Mesh Modelo = new Mesh(); //Constructor public Game1() {...</pre>  |
| <b>Privadas Globales</b>               | Siempre se declararán o inicializarán antes del constructor de la clase pero después de todas las variables públicas que utilice la clase. Puede o no incluir la palabra reservada <i>private</i> , ya que de no preceder dicha palabra a la variable se considerará como privada. | <pre> ... public Matrix RotacionY; Matrix Rotacion; private Matrix NuevaRotacion;  //Constructor public Game1() {...</pre>   |
| <b>Públicas Globales:</b>              | Siempre se declararán o inicializarán antes del constructor de la clase y antes de toda variable privada que utilice la clase.   | <pre> ... public Matrix RotacionY; Matrix Rotacion; private Matrix NuevaRotacion;  //Constructor public Game1() {...</pre>   |
| <b>Variables Globales de Funciones</b> | Siempre se declararán o inicializarán Después de la definición de la función y encabezaran todo código que forme parte de la función.  | <pre> void RenderizarNave (GameTime gameTime,Matrix viewMatrix, Matrix projectionMatrixSOMEE) { float VelocidadNave; float Temporizador; Vector2 NuevaPosicionHumo = PosicionSprite;</pre>   |
| <b>Parámetros de funciones</b>         | Cuando se enví parámetros a funciones, se re declararan de acuerdo a un nombre que permita identificarlas y al mismo tiempo diferenciarlas de las variables que están siendo enviadas a la función.  | <pre> Llamada a funcion: Pitagoras3D(PosicionX, PosicionY, PosicionZ); Variables dentro de funcion: void Pitagoras3D(float PosX, float PosY, float PosZ) { float DistanciaLinear = (float)Math.Sqrt(Math.Pow(PosX - MatrizCaracter[0].Translation.X, 2) + Math.Pow(PosY - MatrizCaracter[0].Translation.Y, 2)+ Math.Pow(PosZ - MatrizCaracter[0].Translation.Z, 2)); }</pre> |

Cuadro 5.4: Tipos de variables y estándares.



#### 5.1.5.4. Nombres de objetos y variables

Los nombres que se utilizarán deberán hacer uso de la nemotecnia para facilitar comprender el objetivo o razón de cada variable u objeto creado.

El número de caracteres debe de mantenerse dentro de los 32 caracteres para cada nombre, a pesar que C# permite hasta 231 caracteres para la declaración de nombres de variables, se sabe que nombres cortos permiten un mejor manejo de variables y fácil comprensión.

Así mismo cuando la variable u objeto esté compuesto de dos o más palabras, cada palabra tendrá la primer letra en mayúsculas para su fácil lectura (*a excepción de las variables booleanas que funcionan como interruptores*).

Cada Nivel dentro del videojuego contendrá variables booleanas que determinen cuando se dibuja un nivel y se pasa al siguiente, su comportamiento es similar a un grupo de interruptores que, dependiendo de su estado en un periodo de tiempo, determinaran que nivel o acción especial se llevará a cabo. A estas variables globales, para diferenciar de las demás y fácil visualización dentro del código, se dejaran en mayúsculas a todos los caracteres que componen su nombre y si se trata de un nombre compuesto, se utilizará un guión bajo entre cada palabra. Por ejemplo, la variable que dibuja el nivel uno se llamará NIVEL\_UNO, la variable que activa la introducción se llamará INTRODUCCION.

Al nombre de cada variable se le añadirá tres caracteres que representen el tipo de variable, si es una estructura o una instancia de clase (objeto que necesita la palabra reservada **new** para inicializar) utilizara tres letras mayúsculas al final AAA, y si se trata de una variable simple, estructura o arreglo, se utilizarán tres letras, la primera de ella mayúscula: *Aaa*. Por ejemplo, una variable tipo flotante que funciona como contador, se llamará: **ContadorFlt** y una variable tipo estructura que guarde los valores de los vectores de dirección, intensidad y color de luz se le llamara: **LuzStr**.

A continuación se explican algunos nombres de las variables simples y las estructuras.



| Tipo                           | Uso Variable Simple  | <p align="center"><b>Formato Palabras</b></p> <p><b>Mayúscula:</b> inicio de Palabra que compone el nombre<br/> <b>Minúsculas:</b> cuerpo del palabra</p> <p align="center"><b>Formato Tipo</b></p> <p><b>Últimos 3 caracteres:</b> tipo de variable<br/>                     Primer letra mayúscula y últimas 2 minúsculas.</p> |
|--------------------------------|--|--|
| float                          | Determina el tamaño de la escala del texto que se dibuja         | AaaaaAaaaaXxx  |
|                                | Ejemplo: EscalaTextoFit  |  |
| string                         | Guarda los mensajes de texto que se presentarán durante el juego | AaaaAaaaAaaaaaaaXxx  |
| Ejemplo: TextoTodoPropositoStr |  |  |

Cuadro 5.5: Nombres de variables simples

| Tipo                       | Uso Estructura e instancias de Clases.  | <p align="center"><b>Formato Palabras</b></p> <p><b>Mayúscula:</b> inicio de Palabra que compone el nombre<br/> <b>Minúsculas:</b> cuerpo del palabra</p> <p align="center"><b>Formato Tipo</b></p> <p><b>Últimos 3 caracteres:</b> tipo de variable<br/>                     Las tres letras mayúsculas.</p> |
|----------------------------|---|---|
| Vector3D                   | Guarda los componenetes x,y,z de posiciones, velocidades y aceleraciones de objetos 3D. | AaaaAaaaa...XXX   |
|                            | Ejemplo: TierraPosicionV3D  |   |
| Modelo3D                   | Clase para la gestion de presentación y manipulación de objetos 3D Estaticos.           | AaaaaaAaAaaaaXXX  |
| Ejemplo: MaquinaDeSodasM3D |   |   |

Cuadro.5.6: Nombres de estructuras y objetos instanciados



| Tipo | Uso Variables Simples Booleanas<br>(Caso especial para interruptores para gestión de niveles) | Formato Palabras<br>Todas Mayúsculas y si se compone de 2 o más palabras se utiliza guion bajo. |
|------|---|---|
| bool | Permite llamar las funciones y objetos para dibujar el menú principal del videojuego.         | <b>AAAA_AAAAA</b>   |
|      | Ejemplo: MENU_PRINCIPAL   |   |
| bool | Permite llamar las funciones y objetos para desplegar la introduccion del video juego.        | <b>AAAAAAA</b>  |
|      | Ejemplo: INTRODUCCION   |   |

Cuadro 5.7: Nombres de booleanas especiales

| Objetivo de la Variable  | Nombre y Formato Estandarizado   |
|--|--|
| Guardar las posiciones x, y, z de una transformada matricial aplicada a un objeto. | <pre>Vector3 PosicionCarroV3D; Vector3 PosicionVendedoraDeComidaV3D;</pre>   |
| Guardar el Valor de una variable de caracteres                                     | <pre>string RespuestaVendedoraDeComidaStr; RespuestaVendedoraDeComidaStr = "No tienes suficiente dinero!";</pre>   |
| Llevar el conteo de un evento progresivo   | <pre>int ContadorDeRespuestasMalasInt; float ContadorIntroduccionFlt;</pre>  |
| Guardar el valor de múltiples posiciones para un arreglo de 50 modelos de árboles. | <pre>Vector3[] PosicionArbolV3D = new Vector3[50]; PosicionArbolV3D [0] = new Vector3(100,2,12); PosicionArbolV3D [1] = new Vector3(130,2,-12);...</pre> |

Cuadro 5.8: Otros ejemplos



### 5.1.5.5. Características y estructuras de funciones.

Al igual que las variables y objetos instanciados, todos los métodos implementados en el desarrollo del software utilizarán la nemotecnia y la distinción de palabras a través de una primera letra mayúscula para sus nombres, tres letras minúsculas se antepondrán al nombre para determinar si es pública (pub) o privada (prv) y tres letras; la primera mayúscula, se añadirán al final para denotar el tipo de retorno. Al mismo tiempo, la palabra reservada `private` se implementara opcionalmente si se trata de una función privada, de lo contrario se utilizara la palabra reservada `public`. Para todas las funciones que no retornen ningún valor se utilizara la palabra reservada `void`, si retorna un valor se utilizara el tipo del valor retornado en lugar de `void`. Cuando se envíen valores por referencia, es necesario incluir la palabra reservada `ref`, aunque es opcional dependiendo de la estructura del código que se esté implementado en C#.

Todas las variables internas a la función deberán de ser declaradas en el encabezado de la función, es decir justo después del nombre y parámetros entrantes (si existen). Ejemplos:

| Descripción de la función   | Nombre y Formato Estandarizado   |
|---|--|
| <ul style="list-style-type: none"> <li>✓ No recibe ningún valor</li> <li>✓ No retorna ningún valor</li> <li>✓ Es privada</li> </ul>                 | <pre>Void prvCalcularDistanciaVoi() {...}</pre>  |
| <ul style="list-style-type: none"> <li>✓ No recibe ningún valor</li> <li>✓ Retorna un valor tipo flotante</li> <li>✓ Es privada</li> </ul>          | <pre>float prvCalcularDistanciaFlt() { float dFlt = DistanciaAFlt - DistanciaBFlt; return dFlt; }</pre>  |
| <ul style="list-style-type: none"> <li>✓ Recibe dos valores</li> <li>✓ Retorna un valor tipo Matrix</li> <li>✓ Es pública</li> </ul>                | <pre>public Matrix pubTransformarMat(Vector3 Pos, float rotacion) { return Matrix.CreateRotationY*Matrix.CreateTranslation(Pos); }</pre>               |
| <ul style="list-style-type: none"> <li>✓ Recibe tres valores, uno por referencia</li> <li>✓ No retorna ningún tipo</li> <li>✓ Es pública</li> </ul> | <pre>public void pubAsignarPosicionVoi(float PosXFlt, float PosZFlt, ref float rotacionFlt) { rotacionFlt = (float)Math.Atan(PosXFlt/PosZFlt); }</pre> |

Cuadro 5.9: Nombres de funciones y estándares





### **5.1.5.6. Utilización de bibliotecas predefinidas**

Para aprovechar el encapsulamiento del lenguaje orientado a objetos, se empacará todo código recursivo que no necesita mantenimiento posterior en bibliotecas dll (*dynamic linking library* o *bibliotecas de enlace dinámico*.) Cuando pertenezca al mismo *namespace*, simplemente se incluirá en el proyecto en la carpeta de referencias, de lo contrario, se utilizará la palabra reservada `using` en el encabezado de cada archivo de código, seguido por el nombre del paquete al que el archivo de la biblioteca pertenece.

### **5.1.5.7. Seccionamiento y ocultación de código.**

Para facilitar la codificación, se implementará la regla de regionalizar el código, para así seccionar los segmentos de códigos que pueden estar compuestos por funciones, variables, clases, conjunto de clases, estructuras, etc. Ya que a medida que aumenten las líneas de código será cada vez más difícil navegar a través de este.

La estructura de la regionalización de código es la siguiente:

```
#region Nombre de la región de código
Todo código que se desee seccionar y ocultar
#endregion
```

Al implementar regiones, se puede fácilmente ocultar el código para facilitar la añadidura de nuevas clases, rutinas, etc.

### **5.1.5.8. Documentación del Código**

Es necesario documentar el código, en especial en secciones de este que no son fáciles de comprender, por lo tanto, la siguiente regla se utilizará para la documentación en los archivos de código fuente:

#### Resumen

Los resúmenes utilizando los caracteres y palabras reservadas: “`///  
/// </summary>`”, se posicionaran justamente antes de la función o clase, estos permitirán durante el desarrollo y mantenimiento, tener una idea clara de lo que hace determinada función o clase. La siguiente figura muestra a la función heredada de `Game`, dentro de `VideoJuego3D` con su respectivo resumen indicado dentro del rectángulo verde.

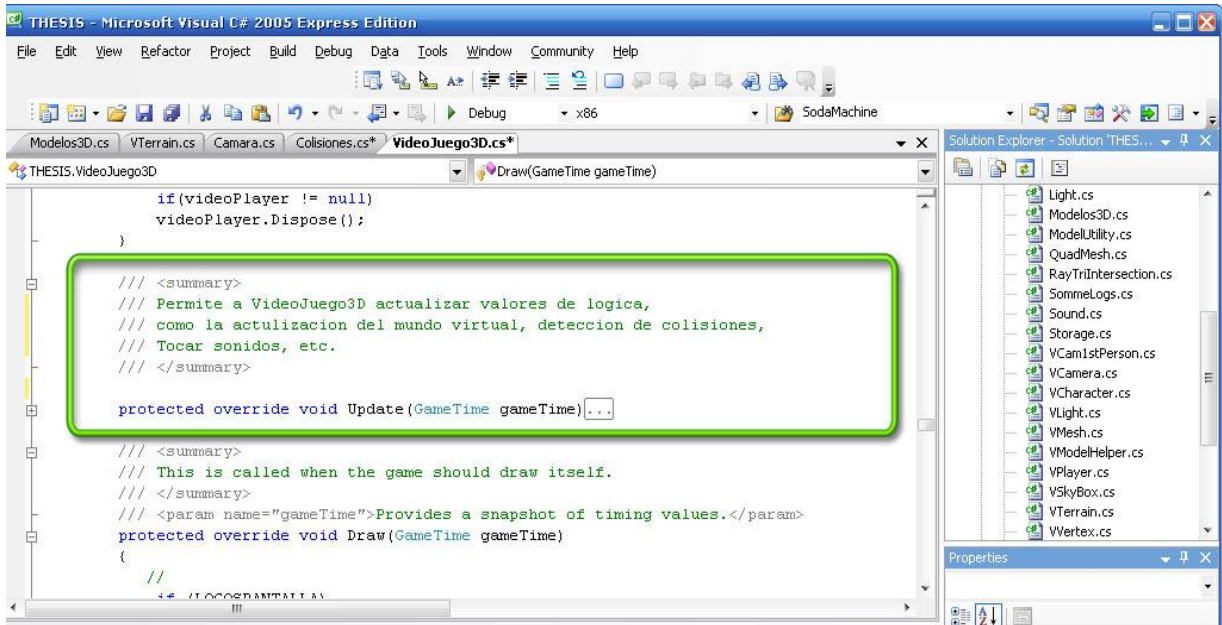


Figura 5.14: Estándar de código fuente, Resúmenes

### Comentarios Paralelos

Los Comentarios Paralelos permiten rápidamente escribir comentarios para cualquier línea de código, el formato a seguir será de comentar las líneas de código a la derecha de este utilizando los caracteres “//”. Ver siguiente figura:

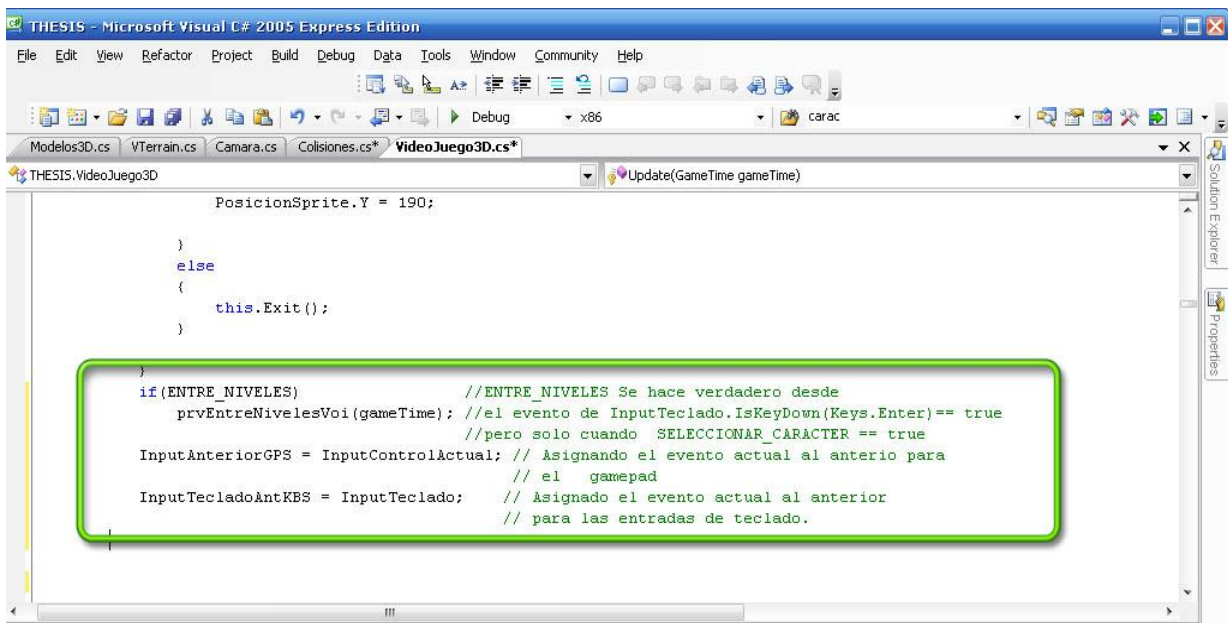


Figura 5.15: Estándar de código fuente, Comentarios Paralelos a la línea



### Comentarios Multilineales

Los Comentarios Multilineales permiten comentar una sección de código para las pruebas o dejar comentarios de documentación interna fácilmente legibles, ver siguiente figura. Utilizando los caracteres reservados “/\*” y “\*/” permite utilizar varias líneas para de comentarios.

```

PPInitialize(AdministradorDispositivo.GraphicsDevice);
ModelCollider = new Collider();
TexturasT2D = new Texturas2D(AdministradorDispositivo, Content, 1);

/*
De aqui en adelante se llaman los modelos
tipo EfectoMapaNormal, ModeloDeEfectoNormalMOD es local
y permite cargar temporalmente el modelo para Efectos
Normales y luego se le asigna al Modelo Tipo ModeloDeEfectoNormalMOD
Para instanciarlo
*/

Model ModeloDeEfectoNormalMOD = Content.Load<Model>("Content/resources/models/StaticModels/Space/earth");
TierraEMN = new EfectoMapaNormal();
TierraEMN.LoadContentNormalMaps(this, ModeloDeEfectoNormalMOD, new Vector4(.2f, .2f, .2f, 1));
AtmosferaEMN = new EfectoMapaNormal();
ModeloDeEfectoNormalMOD = Content.Load<Model>("Content/resources/models/StaticModels/Space/Atmosfera");
AtmosferaEMN.LoadContentNormalMaps(this, ModeloDeEfectoNormalMOD, new Vector4(.2f, .2f, .2f, 1));

ReseteadorEMN = new EfectoMapaNormal();
ModeloDeEfectoNormalMOD = Content.Load<Model>("Content/resources/models/StaticModels/Space/resetter");
ReseteadorEMN.LoadContentNormalMaps(this, ModeloDeEfectoNormalMOD, new Vector4(.2f, .2f, .2f, 1));

LibroEMN = new EfectoMapaNormal();
ModeloDeEfectoNormalMOD = Content.Load<Model>("Content/resources/models/StaticModels/Space/libro");
LibroEMN.LoadContentNormalMaps(this, ModeloDeEfectoNormalMOD, new Vector4(.2f, .2f, .2f, 1));

```

Figura 5.16: Estándar de código fuente, Comentarios Multilineales

#### 5.1.5.9. Estándares de Almacenamiento de Datos

El almacenamiento de datos se llevará a cabo a través de estructuras almacenadas dentro de archivos de extensiones **.sav** y **.bin** que estarán guardados en el folder “mis documentos” del sistema operativo que se esté utilizando, de acuerdo al número de jugadores. Esto para cumplir con los estándares de la industria de videojuegos. Se podrán almacenar un máximo de 10 sesiones de avance por jugador, dándole la opción al usuario de seleccionar cualquiera de las diez ‘slots’ de juego para guardar su avance.

La estructura de datos que se almacenará en cada archivo será la siguiente:

```

[Serializable]
public struct SaveGameData
{
    public string NombreStr; // Nombre de jugador
    public string PasswordStr; // Clave del jugador
    public bool NivelUnoBol; // indica el nivel actual
    public bool NivelDosBol; // indica el nivel actual
    public bool NivelTresBol; // indica el nivel actual
    public bool NivelCuatroBol; // indica el nivel actual
}

```



```

public bool NivelCincoBol; // indica el nivel actual
public int EscenaInt;      // La escena dentro del nivel guardado
public int VidasInt;      // número de vidas
public float NiveldeVidaFlt; // Nivel de vida
public int dificultad;    // Dificultad de juego actual
public Vector3D PosicionV3D; // Posicion del Jugador
public double EcoDolaresDbl; // Cantidad de Ecodolares
public double EcoPuntosDbl; // Cantidad de Ecopuntos recolectados
public struct objetosRecSTR; // nombre y cantidad de objetos
recolectados

    }
    
```

Este patrón será utilizado cada vez que se cree un nuevo jugador y cada vez que se guarde el avance para este. El acceso a los datos almacenados se hará a través de una conexión serializada al archivo, desconectándose de esta al recuperar el último valor del arreglo.

### 5.1.6. Estándares de Recursos Multimedia

Para permitir una mejor experiencia de parte del jugador y para el desarrollo de un videojuego liviano, es necesario estandarizar los recursos multimedia, (cuadro 5.10), en especial, limitar su tamaño para no sobrecargar la memoria de video y minimizar los tiempos de acceso a esta.

| Tipo de Recurso                  | Peso Máximo en MB | Descripción  |
|----------------------------------|-------------------|--|
| Modelo 3D                        | 1.00 MB           | Modelos geométricos compuestos por vértices y polígonos para representar caracteres, edificios, ciudades, etc. Para añadir realismo estos se 'texturizan' con imágenes. Entre los tipos de archivos de modelos que se utilizarán se tienen: archivos .x y .fbx. Ambos tipos son estándares de la industria de videojuegos. |
| Textura de Modelos 3D            | 256.00 KB         | Todas las imágenes que se utilizan para texturizar al modelo. Entre los tipos de imágenes que se utilizarán para este software se tienen: jpg, png, tga y bmp.   |
| Textura 2D                       | 256.00 KB         | Utilizadas para representar una consola de control, nivel de vida, ecodolares, ecopuntos. Los tipos de archivos que se utilizan para las texturas 2D son: jpg y png.   |
| Texturas para efectos especiales | 64 KB             | Texturas base para crear efectos especiales como: lluvia, fauna, rayos, fuego, humo, destellos, entre otros. Se utilizarán exclusivamente los formatos jpg y png.  |



|        |         |  |
|--------|---------|--|
| Sonido | 1.00 MB | Sonidos para efectos y diálogos entre caracteres. Se utilizara exclusivamente el formato .wav de 8 bits a 12,000 kHz a 23 kilobits por segundo.  |
| Video  | 10 MB   | Videos educativos que serán mostrados en los diferentes niveles y áreas especificas, implementado un hilo de procesamiento dedicado y carga en demanda. El único formato que se utilizará es el formato WMV debido a sus características de compactación y minimización de cuadros de video. |

Cuadro 5.10: Estándares de recursos multimedia

### 5.1.7. Patrones de Arquitectura

Los patrones de arquitectura expresan el esquema fundamental de organización para sistemas de software. Proveen un conjunto de subsistemas predefinidos, especifican sus responsabilidades e incluyen reglas y guías para organizar las relaciones entre ellos. Los patrones de arquitectura representan el nivel más alto en el sistema de patrones propuesto en *Pattern Oriented Software Architecture*.<sup>100</sup> Ayudan a especificar la estructura fundamental de una aplicación.

La categorización POSA (**Pattern Oriented Software Architecture**)<sup>101</sup>, establece los patrones arquitectónicos que se adaptan a los diferentes tipos de sistemas existentes como se muestra en el cuadro 5.11.

| Tipos de sistemas      | Patrón arquitectónico                     |
|------------------------|---|
| Sistemas estructurados | Layers<br>Pipes and Filtres<br>Blackboard |
| Sistemas distribuidos  | Broker                                    |
| Sistemas interactivos  | Model-View-Controller                     |
| Sistemas adaptables    | Microkernel<br>Reflection                 |

Cuadro 5.11: Clasificación de patrones de arquitectura de POSA

Debido a la naturaleza interactiva del software propuesto y enfoque orientado a objetos que se ha desarrollado desde la etapa de análisis, se describirá al videojuego como sistema implementando uno de los principales patrones de arquitectura utilizados actualmente, el patrón Modelo-Vista-Controlador.

<sup>100</sup>Arquitectura de Software orientado a Patrones por Frank Buschmann , Regine Meunier , Hans Rohnert , Peter Sommerlad y Michael Stal. Editorial John Wily and Sons LTD. Volumen 1, 2001

<sup>101</sup> Patrones y anti patrones, [http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ\\_3317/default.aspx](http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_3317/default.aspx)



### 5.1.7.1. El patrón Modelo-Vista-Controlador

El **Model-View-Controller** (Modelo-Vista-Controlador o MVC) fue introducido inicialmente en la comunidad de desarrolladores de Smalltalk-80. MVC divide una aplicación interactiva en 3 áreas: *procesamiento*, *salida* y *entrada*. Para esto, utiliza las siguientes abstracciones:

- **Modelo (Model):** Encapsula los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y/o comportamiento de entrada.
- **Vista (View):** Muestra la información al usuario, obtiene los datos del modelo. Pueden existir múltiples vistas del modelo, cada vista tiene asociado un componente controlador.
- **Controlador (Controller):** Reciben las entradas, usualmente como eventos que codifican los movimientos o pulsación de botones del ratón, pulsaciones de teclas, etc. Los eventos son traducidos a solicitudes de servicio (“**service requests**”) para el modelo o la vista. El usuario interactúa con el sistema a través de los controladores.

Las Vistas y los Controladores conforman la interfaz de usuario. Un mecanismo de propagación de cambios asegura la consistencia entre la interfaz y el modelo. La separación del modelo de los componentes vista y del controlador permite tener múltiples vistas del mismo modelo. Si el usuario cambia el modelo a través del controlador de una vista, todas las otras vistas dependientes deben reflejar los cambios. Las vistas, en cambio, recuperan los nuevos datos del modelo y actualizan la información que muestran al usuario. La siguiente figura muestra la estructura de este patrón:

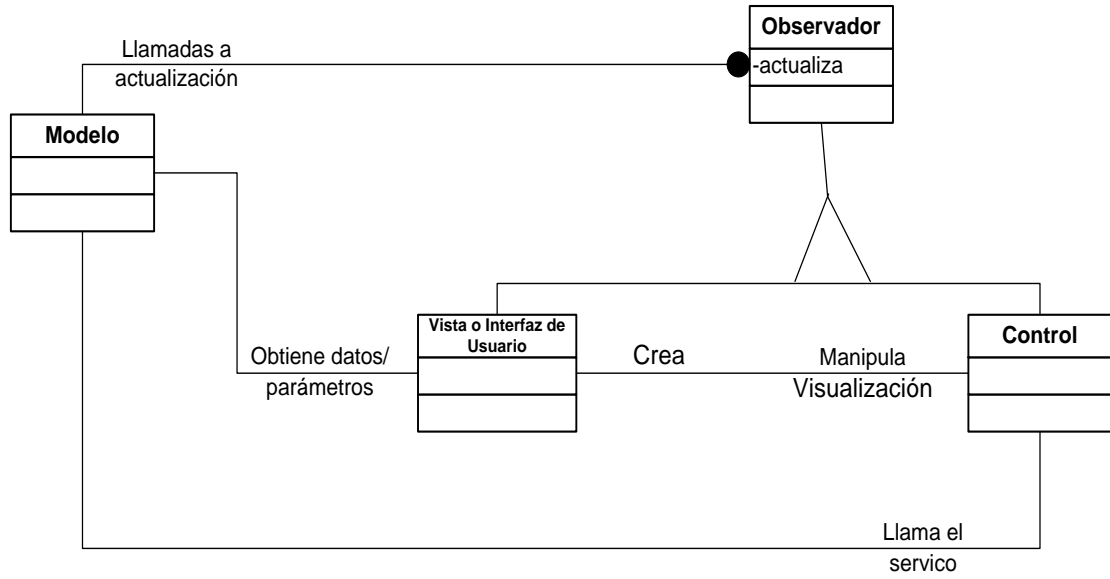


Figura 5.17: Patrón de arquitectura MVC

La figura 5.17 muestra como el usuario controla (Control) el videojuego (Modelo) y visualiza esta actualización a través del monitor (Vista). El Control (teclado, mouse, joystick, gamepad) manipula la visualización (Vista) al enviar los parámetros necesarios a través de una llamada de servicio al modelo (videojuego).

El Modelo (videojuego) interactúa con el Observador (Jugador) a través de Control y Vista. La Vista (Monitor) obtiene los datos necesarios para su actualización del Modelo (videojuego).

Las ventajas de usar este patrón de arquitectura son:

- **Menor acoplamiento**
  - Desacopla las vistas de los modelos
  - Desacopla los modelos de la forma en que se muestran e ingresan los datos
  
- **Mayor cohesión**
  - Cada elemento del patrón está altamente especializado en su tarea (la vista en mostrar datos al usuario, el controlador en las entradas y el modelo en su objetivo de negocio)



### **Las vistas proveen mayor flexibilidad y agilidad**

- Se puede crear múltiples vistas de un modelo
- Se puede crear, añadir, modificar y eliminar nuevas vistas dinámicamente
- Las vistas pueden anidarse
- Se puede cambiar el modo en que una vista responde al usuario sin cambiar su representación visual
- Se puede sincronizar las vistas
- Las vistas pueden concentrarse en diferentes aspectos del modelo

### **Mayor facilidad para el desarrollo de clientes ricos en múltiples dispositivos y canales**

- Una vista para cada dispositivo que puede variar según sus capacidades
- Una vista para la Web y otra para aplicaciones de escritorio

La siguiente figura presenta la arquitectura MVC del videojuego:



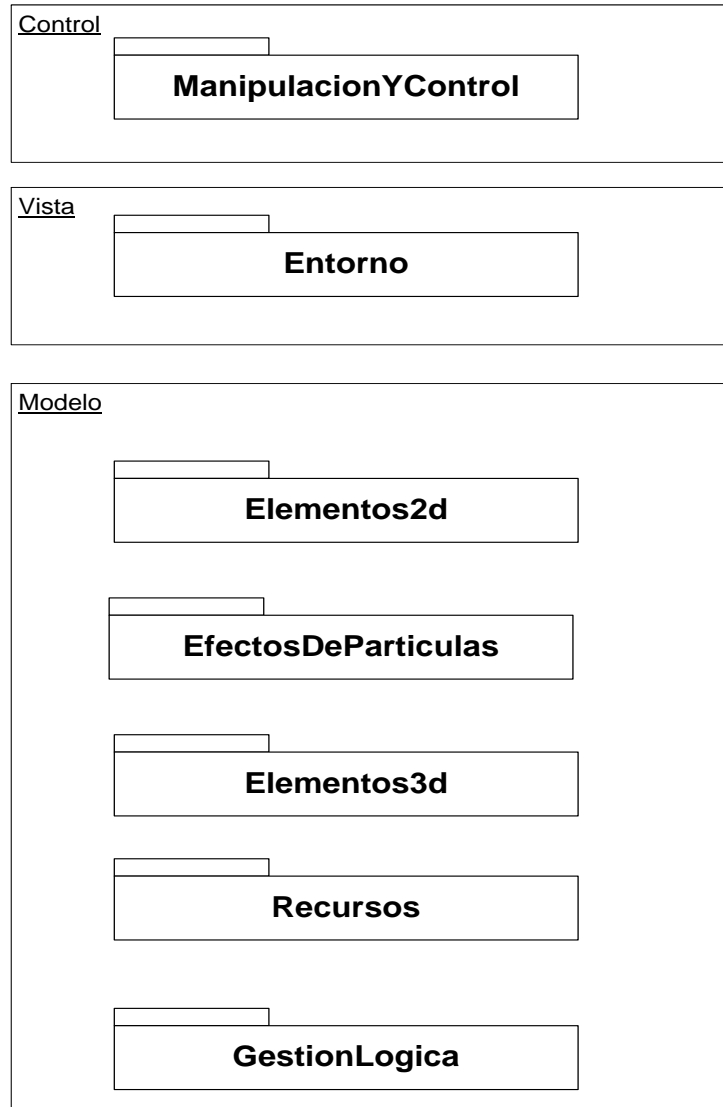


Figura 5.18: Diagrama de arquitectura del videojuego



Para poder visualizar de manera jerárquica las interdependencias de las diferentes secciones del software compuesta por módulos con funciones únicas se implementará un diagrama jerárquico de sistemas para comprender mejor al videojuego como sistema en su totalidad.

### 5.1.8. Diagrama Jerárquico del Videojuego

El videojuego está compuesto de 4 macro módulos para la gestión de recursos, gestión de información, presentación y actualización. Las siguientes figuras muestran la estructura lógica y jerárquica de estos y sus dependencias de forma resumida.

#### 5.1.8.1. Software Interactivo para la Educación Medioambiental de la Población

##### Salvadoreña

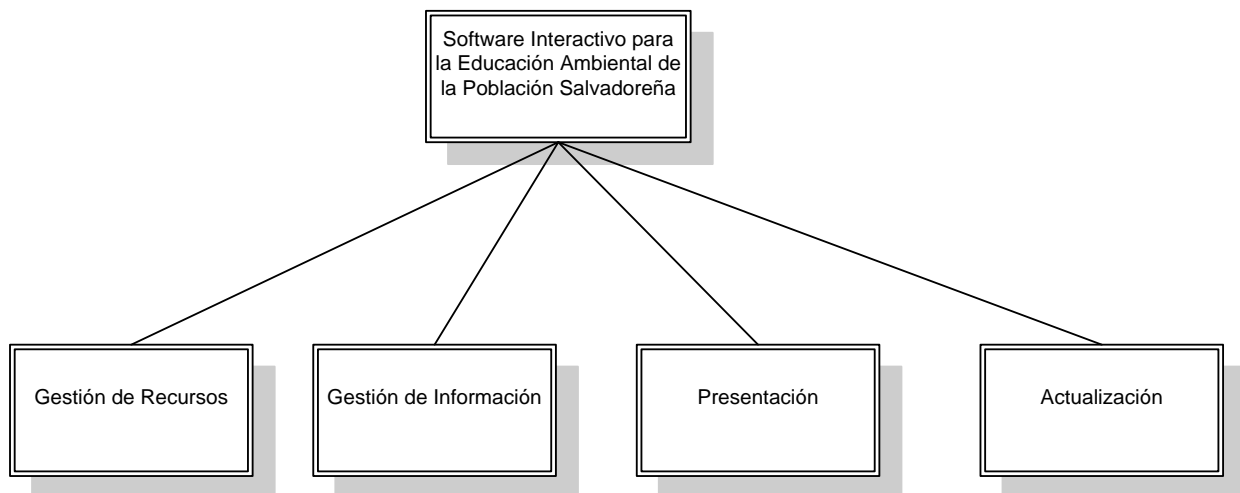


Figura 5.19: Diagrama de arquitectura del videojuego



### 5.1.8.2. Gestión de Recursos

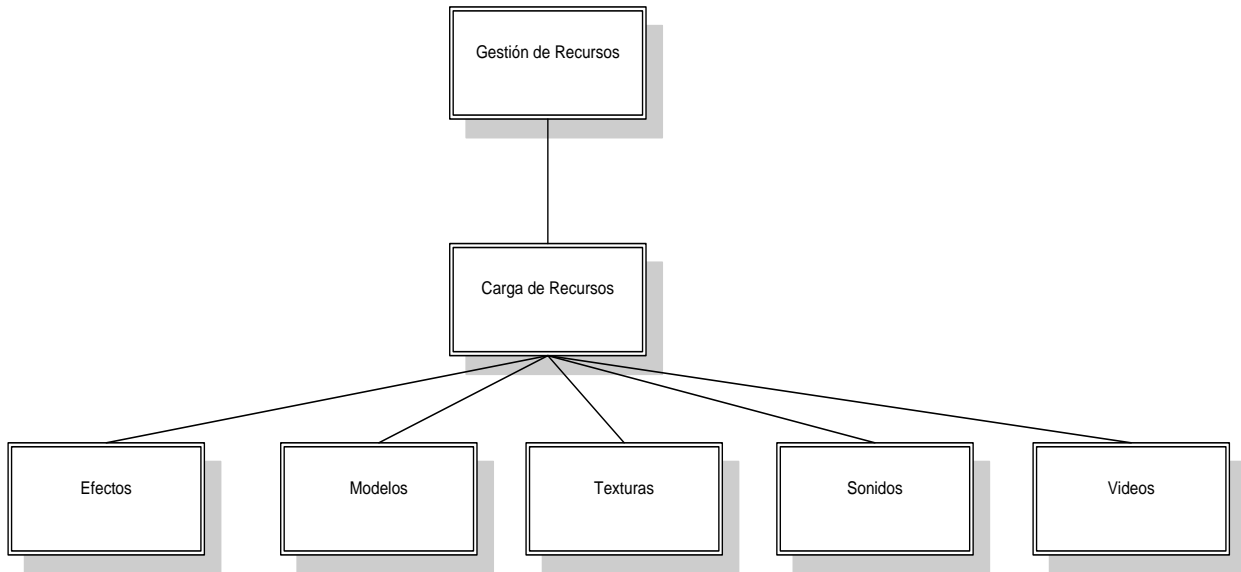


Figura 5.20: Diagrama de arquitectura de Gestión de Recursos

### 5.1.8.3. Gestión de Datos

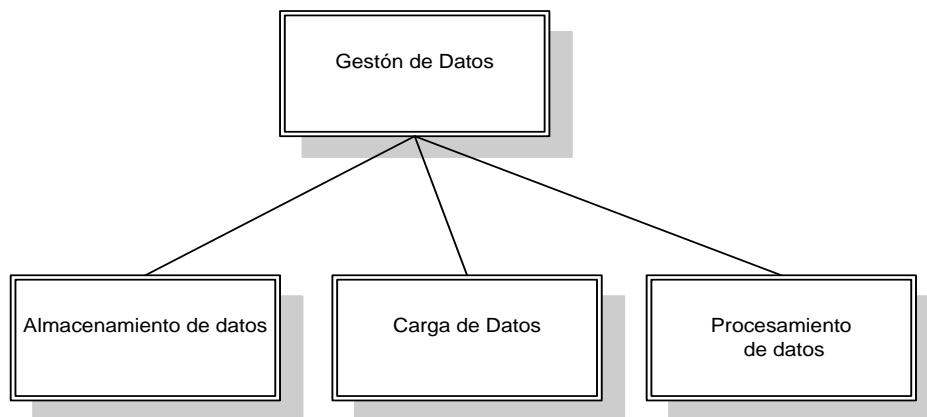


Figura 5.21: Diagrama de arquitectura de Gestión de Datos



### 5.1.8.4. *Presentación*

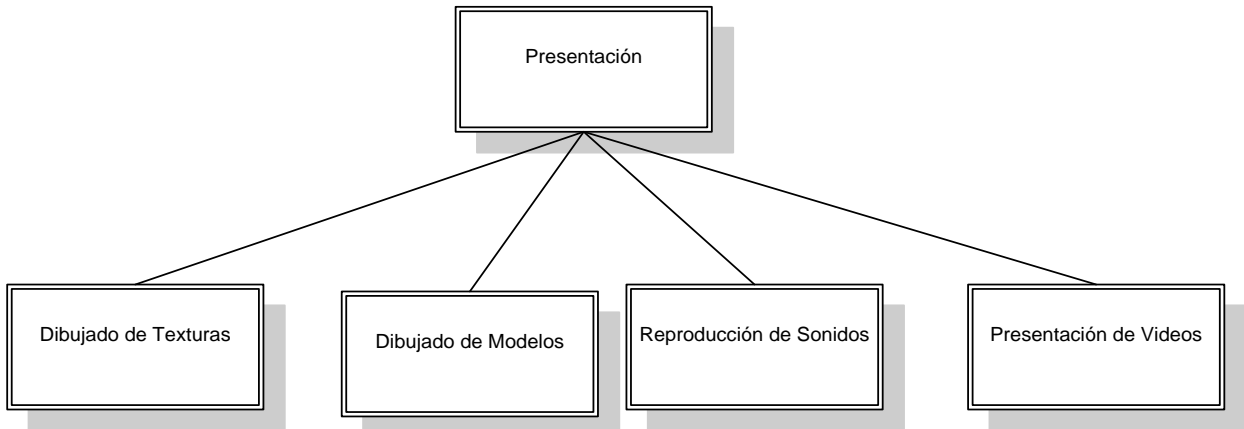


Figura 5.22: Diagrama de arquitectura de Presentación

### 5.1.8.5. *Actualización*

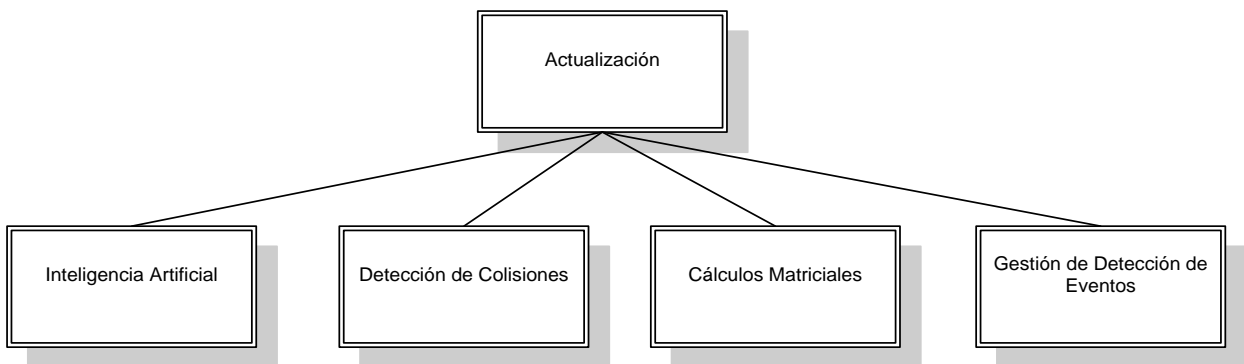


Figura 5.23: Diagrama de arquitectura de Actualización



## 5.2. DISEÑO DETALLADO

### 5.2.1. Descripción de Casos de Usos Reales (con diseño de interfaces)

La definición de los casos de uso reales es una de las primeras actividades dentro de un ciclo de desarrollo. Un caso real describe el diseño concreto del caso de uso a partir de una tecnología particular de entrada y salida, así como su implementación global<sup>102</sup>.

|   |  |                       |
|---|--|-----------------------|
| <b>Caso de Uso:</b>   | <b>Iniciar: Versión 1</b><br><b>Tipo: Primario Real</b>                                | <b>Código: CU-001</b> |
| <b>Actores:</b>   | Jugador  |                       |
| <b>Propósito:</b>   | Cargar los recursos necesarios para inicializar las funciones del software.            |                       |
| <b>Resumen:</b>   | El jugador inicia el juego, el software prepara el entorno para el mundo virtual.      |                       |
| <b>Referencias cruzadas:</b>                                    | R 1.1, R 1.2, R 1.3, R 2.2.  |                       |
| <b>Curso Normal de los Eventos</b>                              |  |                       |
| <b>Acción de los actores</b>                                    | <b>Respuesta del sistema</b>   |                       |
| 1. Este caso de uso comienza cuando el jugador ejecuta el juego |  |                       |
|   | 2. El software carga los recursos necesarios y configuración inicial (R 1.1).          |                       |
|   | 3. El software llama a la siguiente función “Desplegar Introducción <i>i</i> ” (R 1.2) |                       |
| <p><i>i</i>: Pantalla inicial</p>                               |  |                       |

<sup>102</sup> UML y Patrones de Craig Larman, 2001 Prentice Hall. Casos Reales de Uso, Cap. 16, Pág. 163



|   |  |                       |
|---|--|-----------------------|
| <b>Caso de Uso:</b>   | <b>Desplegar Introducción: Versión 1<br/>Tipo: Primario Real</b>   | <b>Código: CU-002</b> |
| <b>Actores:</b>   | Jugador  |                       |
| <b>Propósito:</b>   | Presentar a los usuarios las instituciones para las que se está creando el videojuego.                                       |                       |
| <b>Resumen:</b>   | Se despliegan los logos de las instituciones a cargo del programa actual de Educación Ambiental y logotipo de dicho programa |                       |
| <b>Referencias cruzadas:</b>  | R 1.2, R 1.1, R 2.2.   |                       |
| <b>Curso Normal de los Eventos</b>  |  |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                       |
|   | 1. Presenta la información de introducción ( <i>i</i> ).   |                       |
|   | 2. Terminada la introducción se llamará la pantalla de "Seleccionar Opción".   |                       |
| <b>Cursos Alternos A:</b>   |  |                       |
| 2. El jugador decide saltar la presentación de la introducción ( <i>i</i> ) y le indica al sistema su decisión al presionar la tecla "escape".          |  |                       |
| 3. El software cancela el proceso de "Desplegar Introducción" y llama a la pantalla "Seleccionar Opción" (ver caso de Uso <b>Seleccionar Opciones</b> ) |  |                       |
| <i>i</i> : Pantalla de Introducción   |  |                       |
|   |  |                       |



|                              |   |                       |
|------------------------------|---|-----------------------|
| <b>Caso de Uso:</b>          | <b>Gestionar Sesión: Versión 1</b><br><b>Tipo: Primario Real</b>  | <b>Código: CU-003</b> |
| <b>Actores:</b>              | Jugador   |                       |
| <b>Propósito:</b>            | Permitir al jugador gestionar su cuenta de usuario.   |                       |
| <b>Resumen:</b>              | El sistema muestra al usuario la pantalla de sesión para que este pueda seleccionar una de las tres opciones de sesión, las cuales son: Iniciar sesión, Eliminar cuenta de usuario o crear nuevo usuario. |                       |
| <b>Referencias cruzadas:</b> | R 2.2, R 3.2, CU-004, CU-005, CU-012  |                       |

**Curso Normal de los Eventos**

| Acción de los actores  | Respuesta del sistema   |
|--|---|
| 1. <b>a)</b> Si el usuario selecciona "Iniciar Sesión", Se llama a la pantalla de <b>Inicio de sesión</b> .<br><b>b)</b> Si el usuario seleccionar "Crear Usuario", Se llama a la pantalla de <b>Crear Nueva Cuenta</b> .<br><b>c)</b> Si el usuario seleccionar "Eliminar Usuario", Se llama a la pantalla de <b>Eliminar Usuario</b> . |   |
|  | 2. El sistema Carga los recursos para la opción seleccionada. |

**Cursos Alternos A:**

1. El jugador Cancela **Sesión**.
2. El sistema retorna a **Seleccionar Opción** (o menú principal)





|  |  |                       |
|--|--|-----------------------|
| <b>Caso de Uso:</b>  | <b>Iniciar Sesión : Versión 1<br/>Tipo: Primario Real</b>  | <b>Código: CU-004</b> |
| <b>Actores:</b>  | Jugador  |                       |
| <b>Propósito:</b>  | Autenticar al usuario con la finalidad de llevar un registro de las acciones realizadas por este durante la ejecución del videojuego en las opciones de cargar Juego Almacenado, Iniciar Nuevo Juego, Mostrar Bitácora o Eliminar Usuario. |                       |
| <b>Resumen:</b>  | El sistema autentica al jugador antes de cargar Juego Almacenado, Iniciar Juego Nuevo, Mostrar Bitácora o Eliminar Usuario. Esto le permitirá al sistema tener un control individual de cada jugador.                                      |                       |
| <b>Referencias cruzadas:</b>   | R 2.2, R 3.2   |                       |
| <b>Curso Normal de los Eventos</b>   |  |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>   |                       |
| 1. Este caso de uso comienza cuando el jugador elige en las opciones principales una de las siguientes modalidades: Cargar Juego Almacenado, Iniciar Juego Nuevo, Mostrar Bitácora o Eliminar Usuario. |  |                       |
|  | 2. El sistema solicita (A) el nombre de usuario (login) y (B) contraseña (password) <i>i</i> .   |                       |
| 3. El jugador introduce (A) su nombre de usuario (login) y (B) su contraseña (password)  |  |                       |
|  | 4. El sistema valida los datos del usuario y carga los recursos relacionados a la opción seleccionada por el jugador.  |                       |
| <b>Cursos Alternos A:</b>  |  |                       |
| 3. El jugador introduce nombre de usuario o contraseña incorrecto.   |  |                       |
| 4. El sistema muestra al usuario un mensaje de "datos incorrectos" <i>ii</i> .   |  |                       |
| 5. El sistema solicita al jugador introducir nuevamente los datos (vuelve al paso 2)   |  |                       |





*i*: Pantalla de iniciar sesión





*ii*: mensaje de datos incorrectos



|  |  |                       |
|--|--|-----------------------|
| <b>Caso de Uso:</b>  | <b>Crear Nuevo Usuario : Versión 1<br/>Tipo: Primario Real</b>   | <b>Código: CU-005</b> |
| <b>Actores:</b>  | Jugador  |                       |
| <b>Propósito:</b>  | Permitir al jugador crear una nueva cuenta para almacenar el avance obtenido en las partidas de juego que realice en los diferentes niveles, esto le permitirá reiniciar el juego desde el punto en el que haya jugado y guardo el avance. |                       |
| <b>Resumen:</b>  | El jugador tendrá la oportunidad de crear una nueva cuenta   |                       |
| <b>Referencias cruzadas:</b>   | R 2.1  |                       |
| <b>Curso Normal de los Eventos</b>   |  |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>   |                       |
|  | 1. El sistema del videojuego le indica al usuario que introduzca un nombre de usuario (login) y una contraseña (password) para crear una nueva cuenta <i>i</i>   |                       |
| 2. El jugador introduce un Nombre de Usuario y una contraseña para el nuevo usuario a crear.<br><i>i-a</i> |  |                       |
|  | 3. El sistema genera un nuevo expediente con el nombre del usuario y contraseña.   |                       |



| Curso Alterno A:   |
|--|
| 3. El jugador introduce un Nombre de Usuario existente   |
| 4. El sistema le indica que ese usuario ya existe y que introduzca uno diferente <i>ii</i>   |
| Curso Alterno B:   |
| 3. El jugador Cancela la Función de “Crear Nuevo Usuario” <i>i - b</i>   |
| 4. El sistema retorna al usuario a las opciones principales.   |
| <p><i>i:</i> Pantalla de Crear Nuevo Usuario</p>    |
| <p><i>ii:</i> Pantalla de Crear Nuevo Usuario</p>  |



|   |   |                       |
|---|---|-----------------------|
| <b>Caso de Uso:</b>   | <b>Seleccionar Opciones : Versión 1<br/>Tipo: Primario Real</b>   | <b>Código: CU-006</b> |
| <b>Actores:</b>   | Jugador   |                       |
| <b>Propósito:</b>   | Permitir al jugador navegar entre las diferentes opciones que el software ofrece, y poder tener acceso a sus funciones.   |                       |
| <b>Resumen:</b>   | El jugador decide que acción tomar, para empezar a interactuar con el software y así configurar el entorno, iniciar un nuevo juego, cargar una partida de juego almacenada previamente, visualizar las bitácoras de avance de cada jugador, crear y eliminar usuarios, acceder a un demo, tener un juego rápido y cerrar el videojuego. |                       |
| <b>Referencias cruzadas:</b>  | R 1.3, R 2.4, R 4.14, R 5.3, CU-007   |                       |
| <b>Curso Normal de los Eventos</b>  |   |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>  |                       |
| 1. Este caso de uso comienza cuando el jugador cancela el despliegue de la introducción o al terminar de desplegarse la introducción.   |   |                       |
|   | 2. Proporciona las opciones que permiten: Configurar juego, Cargar Juego, Mostrar Bitácora, Crear Nuevo Usuario, Eliminar Usuario, Iniciar Juego Rápido, Mostrar Demo y Salir.  |                       |
| 3. El jugador selecciona la opción deseada.<br>a. Si selecciona Nueva Misión Juego, ver Caso de Uso: Iniciar <b>Nuevo Juego</b> .<br>b. Si selecciona Configuración, ver caso de uso <b>Configuración</b> .<br>c. Si selecciona Cargar Juego, ver Caso de Uso: <b>Cargar Juego Almacenado</b> .<br>d. Si selecciona Bitácora, ver Caso de Uso: <b>Mostrar Bitácora</b> .<br>e. Si selecciona Sesión, ver caso de Uso <b>Gestionar Sesión</b> .<br>f. Si selecciona Demo, ver Caso de Uso: <b>Mostrar Demo</b> .<br>g. Si selecciona Salir, ver Caso de Uso <b>Salir</b> . |   |                       |



i. Pantalla de Selección de Opciones (Menú Principal)



|  |  |                       |
|--|--|-----------------------|
| <b>Caso de Uso:</b>  | <b>Configurar Juego : Versión 1<br/>Tipo: Opcional Real</b>  | <b>Código: CU-007</b> |
| <b>Actores:</b>  | Jugador  |                       |
| <b>Propósito:</b>  | Permitir al usuario modificar y personalizar diferentes opciones del videojuego.                                   |                       |
| <b>Resumen:</b>  | El jugador tiene acceso a las configuraciones de modo de video, configuraciones de controles y velocidad de juego. |                       |
| <b>Referencias cruzadas:</b>   | R 2.5, R 2.7, R2.6   |                       |
| <b>Curso Normal de los Eventos</b>   |  |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>   |                       |
| 1. El jugador selecciona la opción configurar juego  |  |                       |
|  | 2. El sistema del videojuego presenta al usuario las opciones de configuración para el videojuego <i>i</i> .       |                       |
| 3. El jugador Selecciona la opción que desea configurar:<br>a. Si selecciona Configurar de Video, ver Caso de Uso: Configurar Video.<br>b. Si selecciona Configurar Controles, ver Caso de Uso: Configurar Controles<br>c. Si selecciona Configurar Velocidad, ver Caso de Uso: Configurar Velocidad |  |                       |



|  |   |                       |
|--|---|-----------------------|
| <b>Caso de Uso:</b>  | <b>Configurar Video : Versión 1</b><br><b>Tipo: Opcional Real</b>   | <b>Código: CU-008</b> |
| <b>Actores:</b>  | Jugador   |                       |
| <b>Propósito:</b>  | Permitir al usuario modificar el tamaño de la ventana   |                       |
| <b>Resumen:</b>  | El jugador tiene acceso a las configuraciones de modo de video (pantalla completa o ventana) para personalizar el tamaño de esta. |                       |
| <b>Referencias cruzadas:</b>   | R 2.7   |                       |
| <b>Curso Normal de los Eventos</b>   |   |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>  |                       |
| 1. Este caso de uso comienza cuando el jugador selecciona la opción Configurar Video.            |   |                       |
|  | 2. El sistema del videojuego presenta las opciones a configurar <i>i</i> .  |                       |
| 3. El jugador Selecciona entre las Opciones.<br>a. Resolución, b. Pantalla Completa y c. Ventana |   |                       |



|  |   |
|--|---|
|  | 4. El sistema solicita confirmar el cambio de configuración de video.                 |
| 5. Confirma el cambio de configuración de video.<br><b>b</b> |   |
|  | 6. Actualiza y muestra los cambios efectuados en la configuración de video <b>d</b> . |

**Curso Alterno A:**

|   |
|---|
| 3. El jugador sale (cancela la configuración <b>e</b> ) y no realiza ninguna configuración al videojuego. |
| 4. El sistema retorna al usuario a las opciones principales.  |

*i.* Pantalla de Selección de Opciones (Menú Principal)



|                              |  |                       |
|------------------------------|--|-----------------------|
| <b>Caso de Uso:</b>          | <b>Configurar Controles : Versión 1</b><br><b>Tipo: Opcional Real</b>  | <b>Código: CU-009</b> |
| <b>Actores:</b>              | Jugador  |                       |
| <b>Propósito:</b>            | Permitir al jugador personalizar el dispositivo de entrada para realizar funcionalidades como caminar correr, saltar,  |                       |
| <b>Resumen:</b>              | El jugador accede a la opción configurar controles. Al realizar la reconfiguración, automáticamente estas teclas quedan definidas para cada partida del juego. |                       |
| <b>Referencias cruzadas:</b> | R 2.5  |                       |



| Curso Normal de los Eventos  |   |
|--|---|
| Acción de los actores  | Respuesta del sistema   |
| 1. Este caso de uso inicia cuando el jugador selecciona la opción "Configurar Controles"                 |   |
|  | 2. El sistema del videojuego presenta las funciones para las que puede modificarse los valores de configuración (adelante, atrás, izquierda, derecha, saltar, correr, lanzar, recoger) <i>i</i> . |
| 3. El jugador Selecciona la función para la que desea cambiar la configuración, <b>a</b> .               |   |
|  | 4. El sistema presenta las opciones disponibles de configuración que pueden ser habilitadas para una función seleccionada.  |
| 5. El jugador selecciona la configuración a la que desea cambiar y confirma su almacenamiento <b>b</b> . |   |
|  | 6. El sistema guarda la configuración realizada y finaliza el caso de uso.  |

**Cursos Alternos A:**

3. El jugador no selecciona ninguna función y sale del caso de uso (**c**).

*i*: Pantalla de Configuración de Controles







|  |  |                       |
|--|--|-----------------------|
| <b>Caso de Uso:</b>  | <b>Configurar Velocidad : Versión 1<br/>Tipo: Opcional Real</b>  | <b>Código: CU-010</b> |
| <b>Actores:</b>  | Jugador  |                       |
| <b>Propósito:</b>  | Permitir al jugador personalizar la velocidad del juego.   |                       |
| <b>Resumen:</b>  | El jugador accede a ajustar la velocidad, donde puede cambiar la rapidez con que se realizan las funciones, estas son almacenadas para cada partida. |                       |
| <b>Referencias cruzadas:</b>   | R 2.6  |                       |
| <b>Curso Normal de los Eventos</b>   |  |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>   |                       |
| 1. Este caso de uso inicia cuando el jugador selecciona la opción "Configurar Velocidad"                 |  |                       |
|  | 2. El sistema del videojuego presenta las velocidades de las que el jugador puede disponer: <i>i</i> .   |                       |
| 3. El jugador Selecciona una nueva velocidad <i>a</i> para el videojuego y pulsa "aceptar" <i>b</i>      |  |                       |
|  | 4. El sistema actualiza la nueva velocidad para el videojuego.   |                       |
| <b>Cursos Alternos A:</b>  |  |                       |
| 3. El jugador decide cancelar configurar velocidad: <i>c</i><br>4. El Sistema retorna al menú principal. |  |                       |
| <i>i</i> : Pantalla de Configuración de Velocidad  |  |                       |
|  |  |                       |







|                              |  |                       |
|------------------------------|--|-----------------------|
| <b>Caso de Uso:</b>          | <b>Mostrar Bitácora : Versión 1</b><br><b>Tipo: Opcional Real</b>  | <b>Código: CU-011</b> |
| <b>Actores:</b>              | Jugador  |                       |
| <b>Propósito:</b>            | Permitir al usuario visualizar su avance en los diferentes niveles del juego.  |                       |
| <b>Resumen:</b>              | El jugador puede tener una perspectiva global de su avance y aprendizaje a través de la interacción de este con el videojuego. |                       |
| <b>Referencias cruzadas:</b> | R4.5   |                       |

| <b>Curso Normal de los Eventos</b>  |   |
|---|---|
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>  |
| 1. Este caso de uso comienza cuando el jugador desea ver su historial de avance y para ello elige la opción mostrar bitácora. |   |
|   | 2. El sistema del videojuego presenta al usuario información de su historial de juego y nivel de puntos acumulado: <i>i</i> . |
| 3. El jugador Selecciona "regresar" a   |   |
|   | 4. El sistema finaliza y retorna al jugador al menú de opciones principal.  |

*i*: Pantalla de Bitácora (Avances y Estadísticas del jugador)








|   |  |                       |
|---|--|-----------------------|
| <b>Caso de Uso:</b>   | <b>Eliminar Usuario : Versión 1<br/>Tipo: Primario Real</b>  | <b>Código: CU-012</b> |
| <b>Actores:</b>   | Jugador  |                       |
| <b>Propósito:</b>   | Eliminar cuentas de usuarios obsoletas o indeseables.  |                       |
| <b>Resumen:</b>   | El jugador, después de haber iniciado una sesión, podrá eliminar su cuenta de usuario, accediendo a la opción de eliminación de usuario. |                       |
| <b>Referencias cruzadas:</b>  | R 2.3  |                       |
| <b>Curso Normal de los Eventos</b>  |  |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                       |
| 1. Este caso de uso comienza cuando el jugador elige la opción <b>a</b> "Eliminar Usuario" desde <i>Sesión i</i>                          |  |                       |
|   | 2. El sistema del videojuego le indica al usuario si está seguro de eliminar su cuenta <b>ii</b>   |                       |
| 3. El jugador indica que "Si"   |  |                       |
|   | 4. El sistema elimina la cuenta y lleva al usuario a las opciones principales.   |                       |
| <b>Cursos Alternos A:</b>   |  |                       |
| 3. El jugador indica que No al cancelar Eliminar Usuario <b>ii- c</b>   |  |                       |
| 4. El sistema retorna al jugador a las opciones principales.  |  |                       |
| <p><i>i:</i> Pantalla de sesión</p>                    |  |                       |
| <p><i>ii:</i> pantalla de eliminación de Usuario</p>  |  |                       |



|  |   |                       |
|--|---|-----------------------|
| <b>Caso de Uso:</b>  | <b>Seleccionar Carácter : Versión 1<br/>Tipo: Primario Real</b>   | <b>Código: CU-013</b> |
| <b>Actores:</b>  | Jugador   |                       |
| <b>Propósito:</b>  | Permitir al jugador seleccionar de entre los caracteres disponibles uno que utilizará a través de las sesiones de juego actual y posteriores.                                       |                       |
| <b>Resumen:</b>  | El jugador tendrá la oportunidad de seleccionar un carácter el cual controlará, cada carácter tendrá una variante en la historia que se desenlazara a medida que el usuario avance. |                       |
| <b>Referencias cruzadas:</b>   | R 2.9   |                       |
| <b>Curso Normal de los Eventos</b>   |   |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>  |                       |
|  | 1. El sistema del videojuego presenta al usuario con tres caracteres interactivos que seleccionar. <i>i</i>   |                       |
| 2. El jugador Selecciona un carácter <i>a</i> y acepta <i>b</i>            |   |                       |
|  | 3. El sistema carga el carácter seleccionado y la historia asociada a este.   |                       |
| <b>Cursos Alternos A:</b>  |   |                       |
| 2. El jugador decide cancelar <i>c</i> la función de Seleccionar Carácter. |   |                       |
| 3. El sistema retorna al usuario a las opciones principales.               |   |                       |
| <i>i</i> . Seleccionando el carácter principal                             |   |                       |
|  |   |                       |



|  |   |   |
|--|---|---|
| <b>Caso de Uso:</b>  | <b>Controlar Carácter : Versión 1<br/>Tipo: Primario Real</b>   | <b>Código: CU-014</b>   |
| <b>Actores:</b>  | Jugador   |   |
| <b>Propósito:</b>  | Permitir al usuario controlar el carácter seleccionado e interactuar con personajes y escenarios virtuales.   |   |
| <b>Resumen:</b>  | El jugador, a través de un teclado, podrá controlar al carácter e indicar las acciones que este tomaría en situaciones en donde se interactúa con personajes y mundo virtual. |   |
| <b>Referencias cruzadas:</b>   | R 2.10, R3.14, R4.7   |   |
| <b>Curso Normal de los Eventos</b>   |   |   |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>  |   |
|  | 1. El sistema del videojuego “escucha” los eventos de entrada de datos del usuario. <i>i</i>  |   |
| 2. El jugador presiona determinadas teclas o combinación de estas para caminar, correr, hablar, realizar tareas especiales, hacer efectiva una acción y activar sub menús. |   |   |
|  | 3. El sistema filtra los eventos de entrada de datos dependiendo del contexto en que se encuentra y ejecuta la(s) función(es) correspondientes.                               |   |
| <b>Cursos Alternos A:</b>  |   |   |
| 2. El jugador No presiona ninguna tecla o combinación de estas   |   |   |
| 3. El sistema del videojuego “escucha” los eventos de entrada de datos del usuario indeterminadamente.   |   |   |
| <i>i</i> : Movimientos Básicos   |   |   |
| caminar  | saltar  | correr  |
|   |   |  |



|   |   |                       |
|---|---|-----------------------|
| <b>Caso de Uso:</b>   | <b>Jugar Versión 1<br/>Tipo: Primario Real</b>  | <b>Código: CU-015</b> |
| <b>Actores:</b>   | Jugador   |                       |
| <b>Propósito:</b>   | Permitir al jugador iniciar una partida donde no existe ningún avance previo  |                       |
| <b>Resumen:</b>   | El jugador inicia una sesión, de no existir una cuenta para determinado jugador, se le da la opción de crear una cuenta. Después de haber iniciado sesión el jugador puede seleccionar su carácter favorito, el cual determinara los diferentes recursos y niveles que se le presentarán. El juego permitirá almacenar el avance del jugador. |                       |
| <b>Referencias cruzadas:</b>  | R 1.1, R 2.13, R 3.3, R 3.4, R 3.5, R 3.6, R 3.7, R 3.8, R 3.9, R 3.10, R 3.12, R 3.13, R3.14, R 4.1, R 4.2, R 4.3, R 4.7, R 4.8, R 4.9, R 4.10, R 4.11, R 4.12, R 4.13, R 4.15, R 5.1  |                       |
| <b>Curso Normal de los Eventos</b>  |   |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>  |                       |
|   | 1. El sistema actualiza y dibuja los recursos del nivel asociado con el personaje seleccionado  |                       |
| 2. El jugador inicia la interacción con el videojuego, (Jugar) y al mismo tiempo podrá seleccionar entre las siguientes opciones. <ul style="list-style-type: none"> <li>a. Si el jugador desea que el carácter camine, corra, salte, etc., se ejecutará <b>“Controlar Caracter”</b>. Ver caso de uso CU-005.</li> <li>b. Si el jugador desea tener acceso a las opciones de menú rápido, se ejecutará <b>“Mostrar Opciones Rápidas”</b>. Ver caso de uso CU-006. T</li> <li>c. Si el jugador desea realizar alguna acción con los objetos recolectados, ya sea ubicarlos en sus lugares respectivos si se trata de desechos sólidos, o consumir alimentos, o tomar objetos para cumplir con las misiones asignadas, deberá ejecutar el caso de uso <b>“Administrar Objetos Recolectados”</b>. Ver caso de uso CU-007.</li> </ul> |   |                       |
| 3. Si el jugador mediante la interacción a avanzado hasta determinado checkPoint  |   |                       |
|   | 4. El sistema ejecuta la función para <b>“Guardar Sesión”</b> , Ver caso de uso CU-008.   |                       |
| 5. El jugador ha pasado todas las pruebas de un nivel (finalizado el nivel) figura i.   |   |                       |



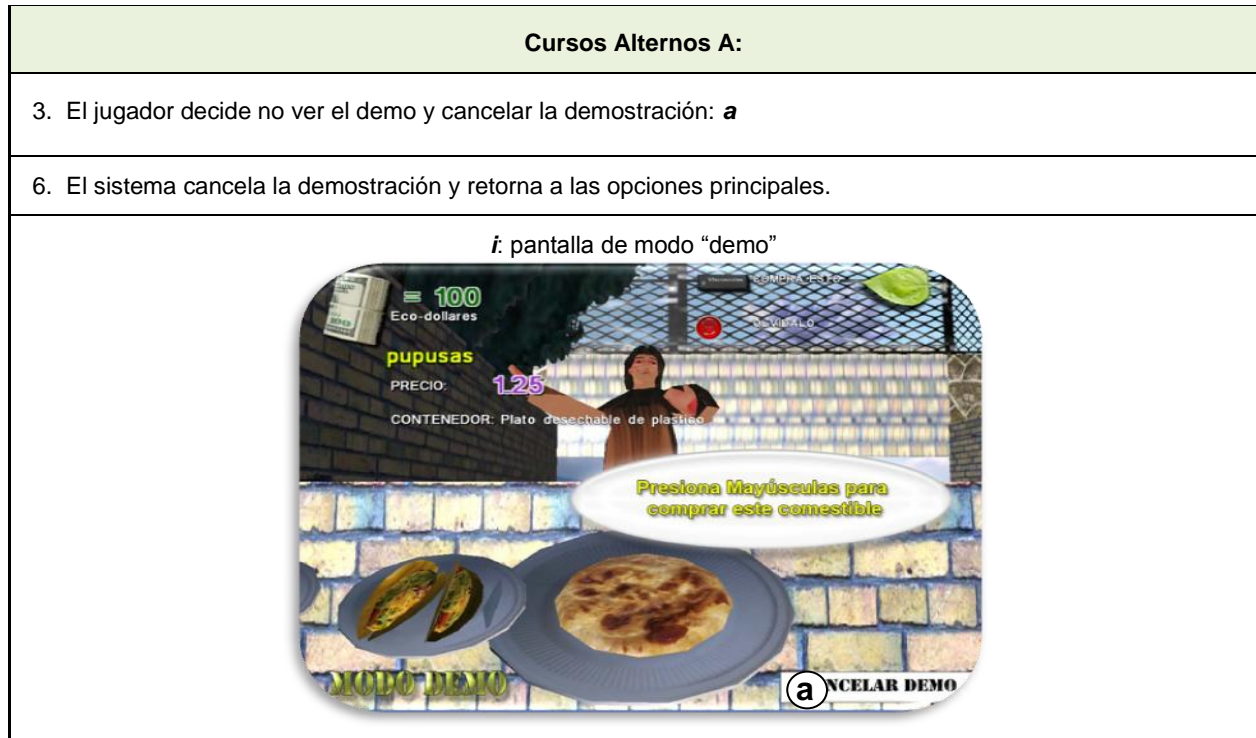
6. El sistema ejecuta la función “**CargarRecursos**”

*i.* Pantalla de felicitación



|  |   |                       |
|--|---|-----------------------|
| <b>Caso de Uso:</b>  | <b>Mostrar Demo : Versión 1<br/>Tipo: Opcional Real</b>   | <b>Código: CU-016</b> |
| <b>Actores:</b>  | Jugador   |                       |
| <b>Propósito:</b>  | Mostrar al jugador una partida de juego corta en la que se le presente la funcionalidad del videojuego.   |                       |
| <b>Resumen:</b>  | Antes de iniciar una sesión por primera vez, el jugador puede tener una vista previa del funcionamiento del videojuego, para lo cual debe ejecutar la opción “Mostrar Demo”, esta opción le mostrará escenas del videojuego que representen la funcionalidad del juego y el jugador pueda de esta manera familiarizarse con el sistema. |                       |
| <b>Referencias cruzadas:</b>   | R4.16   |                       |
| <b>Curso Normal de los Eventos</b>   |   |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>  |                       |
| 1. Este caso de uso comienza cuando el jugador selecciona la opción “Mostrar Demo” |   |                       |
|  | 2. El sistema Carga los recursos del videojuego   |                       |
|  | 3. El sistema dibuja o presenta la demostración del funcionamiento del software: <i>i.</i>  |                       |
| 4. El jugador ve la demostración y regresar a las opciones principales.            |   |                       |





|   |  |                       |
|---|--|-----------------------|
| <b>Caso de Uso:</b>   | <b>Cargar Juego Almacenado : Versión 1<br/>Tipo: Opcional Real</b>   | <b>Código: CU-017</b> |
| <b>Actores:</b>   | Jugador  |                       |
| <b>Propósito:</b>   | Permitir al usuario continuar una partida de juego desde un punto en el que fue almacenado en un momento dado. Esto lo realiza a través de la carga de una partida de varias partidas de juego que el usuario haya almacenado previamente. |                       |
| <b>Resumen:</b>   | El jugador tiene acceso a las últimas 10 “partidas jugadas”, que previamente se hayan guardado al llegar a diferentes “check points” o puntos de chequeo durante una sesión de juego   |                       |
| <b>Referencias cruzadas:</b>  | R2.8, R3.11, R4.4  |                       |
| <b>Curso Normal de los Eventos</b>  |  |                       |
| <b>Acción de los actores</b>  | <b>Respuesta del sistema</b>   |                       |
| 1. Este caso de uso inicia cuando el usuario selecciona la opción “Cargar Juego Almacenado” |  |                       |
|   | 2. El sistema del videojuego presenta al usuario los recursos de “cargar juego almacenado”, permitiéndole visualizar las últimas 10 partidas guardadas <b>a</b>  |                       |



|   |   |
|---|---|
| 3. El jugador Selecciona una de las diez partidas <b>a, b.</b>                |   |
|   | 4. El sistema carga la partida seleccionada |
| <b>Cursos Alternos A:</b>   |   |
| 3. El jugador decide cancelar la función de cargar juego almacenado <b>c.</b> |   |
| 4. El sistema retorna al usuario a las opciones principales.                  |   |
| <p><i>i.</i> pantalla de Cargar Juego</p>                                     |   |

|  |   |                       |
|--|---|-----------------------|
| <b>Caso de Uso:</b>  | <b>Salir (Extendido)</b>  | <b>Código: CU-018</b> |
| <b>Actores:</b>  | Jugador   |                       |
| <b>Propósito:</b>  | Cerrar el videojuego y terminar la sesión de juego.                                   |                       |
| <b>Resumen:</b>  | El usuario ha indicado al sistema del videojuego que desea cerrar la aplicación.      |                       |
| <b>Referencias cruzadas:</b>   | R 5.4   |                       |
| <b>Curso Normal de los Eventos</b>   |   |                       |
| <b>Acción de los actores</b>   | <b>Respuesta del sistema</b>  |                       |
| 1. El jugador seleccionar "Salir" desde las opciones principales. <b>i-a</b> |   |                       |
|  | 2. El sistema cierra la aplicación y descarga los recursos asociados a la aplicación. |                       |





í. Seleccionando la opción **Salir**



## 5.2.2. Diagramas de Interacción

Los diagramas de interacción se elaboran en esta fase de diseño y no se pueden preparar sin antes haber generado:

Un modelo del dominio (conocido como modelo conceptual en UML 1.5), los contratos de operación y los casos de usos reales.

Un diagrama de interacción explica gráficamente las interacciones existentes entre las instancias (y las clases) del modelo de éstas. El punto de partida de las interacciones es el cumplimiento de las poscondiciones de los contratos de operación.

El UML define dos tipos de diagramas de interacción: diagramas de colaboración y diagramas de secuencia; ambos sirven para expresar interacciones semejantes o idénticas de mensaje <sup>103</sup>

### 5.2.2.1. Diagramas de Colaboración

Un diagrama de colaboración es una forma alternativa al diagrama de secuencia para mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas en torno a los objetos y los enlaces entre ellos. La siguiente figura muestra la arquitectura básica de este diagrama.

<sup>103</sup> UML y Patrones de Craig Larman Edición 2001, Cap. 17, págs. 167 - 169



Figura 5.24 Diagrama de colaboración.

En donde Mensaje1 () es enviado a una instancia de la clase A (ClaseAInstancia) y esta al mismo tiempo envía mensajes a una instancia de la clase B (ClaseBInstancia).

A continuación se detallará cada paquete que compone el sistema con su correspondiente diagrama de colaboración, para su fácil comprensión, seguidos por los diagramas de colaboración generados de acuerdo a los contrato de operación.

### 5.2.2.2. Diagrama de Colaboración Manipulación y Control

Manipulación y Control es el paquete que encapsula la funcionalidad o habilidad del videojuego para permitir a un jugador manipular y controlar el caracter principal e interactuar con el mundo virtual.

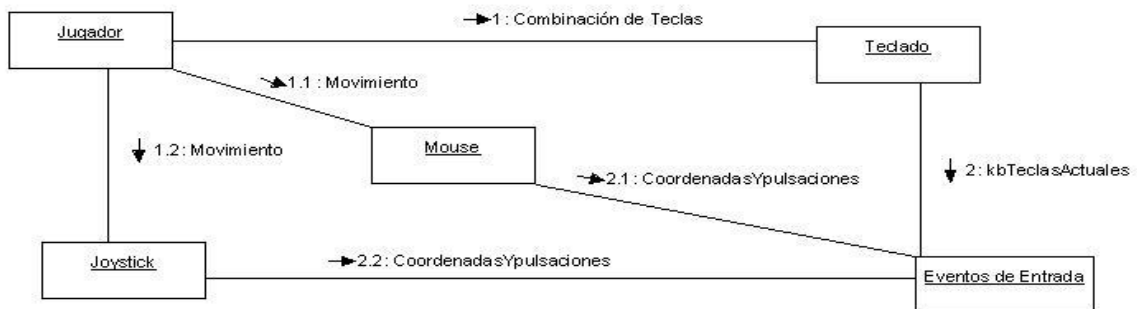


Figura 5.25: Diagrama de Colaboración Manipulación y Control

### 5.2.2.3. Diagrama de Colaboración Elementos 3D

Este paquete encapsula las interacciones entre los diferentes objetos en tres dimensiones así como la gestión de contención y cohesión con el entorno virtual. Va a depender del contexto o avance del nivel y de las posiciones y situaciones en que se encuentre el **Carácter** (carácter principal) para las respuestas de los actores virtuales y Objetos3D.

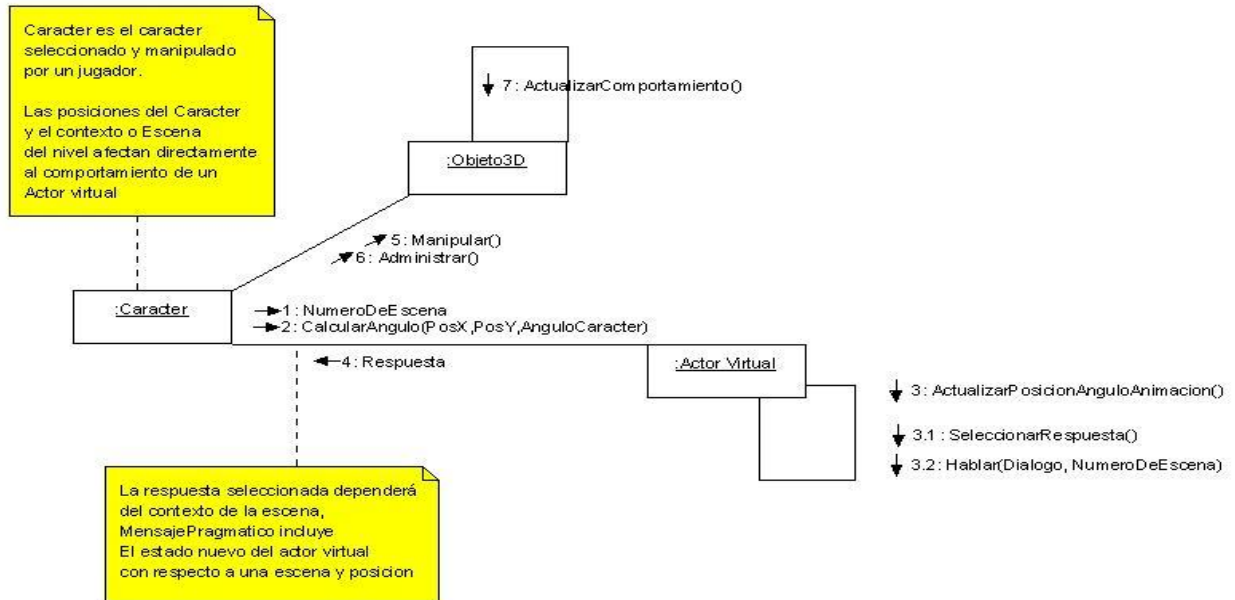


Figura 5.26: Diagrama de Colaboración Elementos 3D

#### 5.2.2.4. Diagrama de Colaboración Recursos

Los recursos son indispensables para el buen funcionamiento del videojuego, su diagrama de colaboración muestra sus interacciones dentro del paquete *Recursos*.

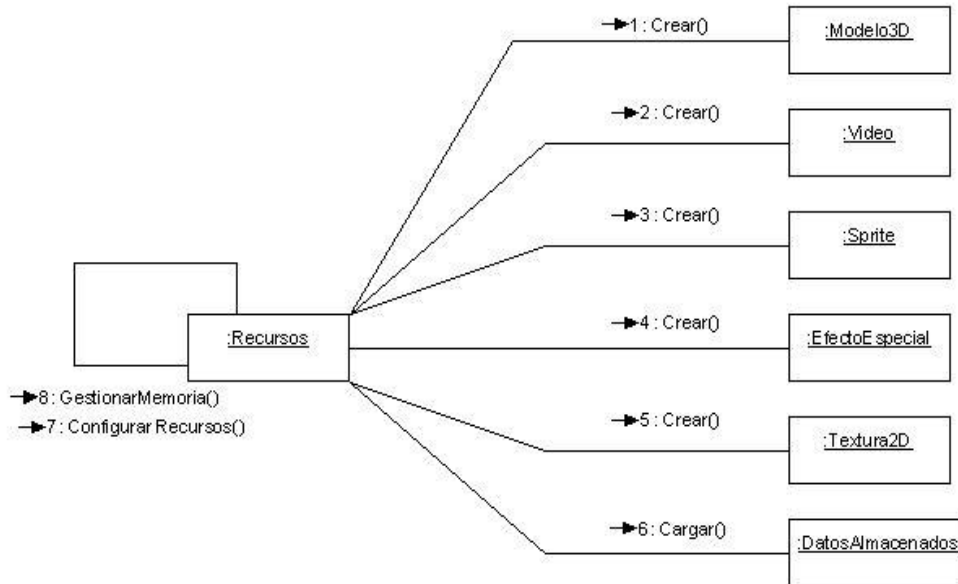


Figura 5.27: Diagrama de Colaboración Recursos



### 5.2.2.5. Diagrama de Colaboración Entorno

El entorno Virtual permite generar la **Vista** del modelo Vista-Modelo-Control (Pág.236), es necesario crear un **Dispositivo** (Clase para el acceso a la tarjeta de video o GPU<sup>104</sup>) para luego crear una **Cámara** y por último el **Entorno Virtual** desde donde la cámara podrá enfocarse

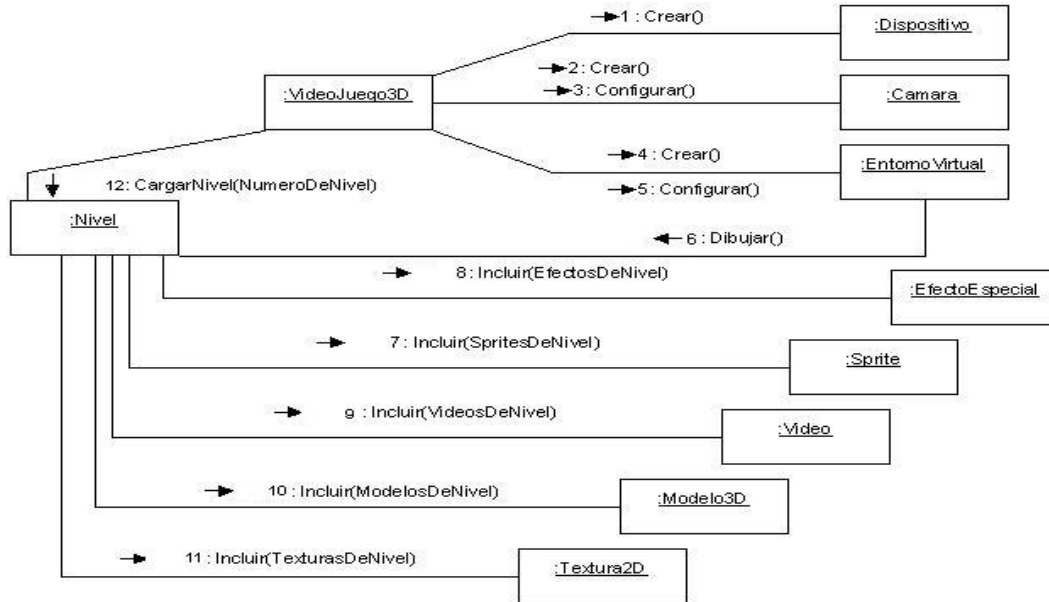


Figura 5.28: Diagrama de Colaboración Entorno

### 5.2.2.6. Diagrama de Colaboración GestionLogica

La gestión lógica es la encargada de las diferentes actualizaciones, progreso del juego, inteligencia artificial, detección de colisiones y todo lo relacionado con las interacciones entre el jugador, el entorno y los actores virtuales.

<sup>104</sup> GPU: Graphics Processing Unit, Unidad de Procesamiento Grafico de las computadoras.

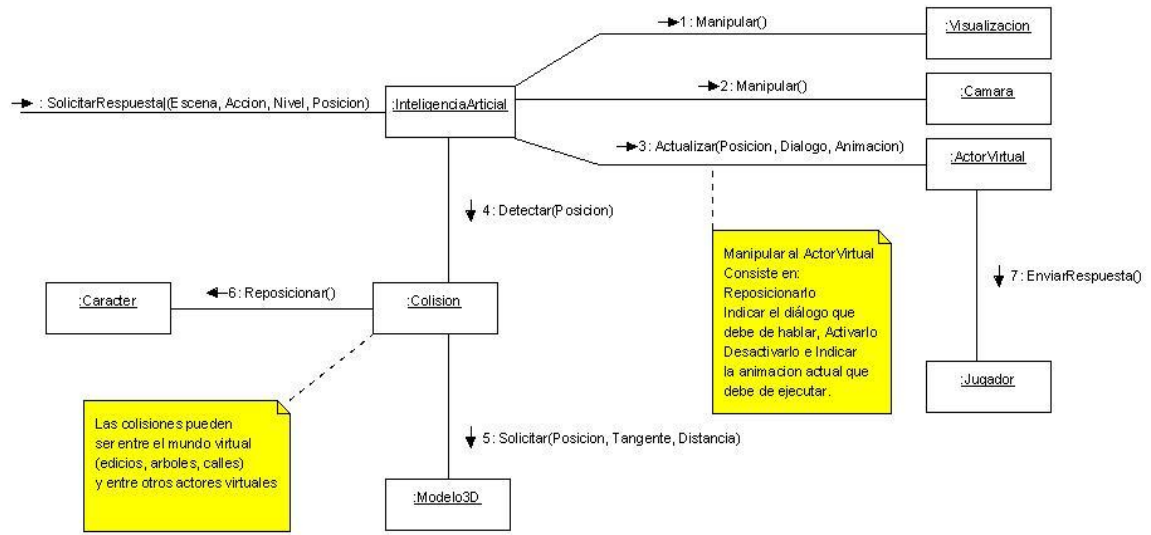


Figura 5.29: Diagrama de Colaboración Gestión Lógica

Hasta ahora se han presentado de forma individual los diagramas de colaboración desde del enfoque de los paquetes que componen al sistema de software. Ahora se presentarán desde un punto de vista de las poscondiciones de los contratos de operación<sup>105</sup>. Como todo desarrollo orientado a objetos, la arquitectura y diseño de un sistema evoluciona y optimiza mediante la visualización de este desde diferentes perspectivas. Como se verá más adelante en los Patrones GRASP<sup>106</sup>, la arquitectura y diseño del videojuego se irá *afinando* hasta alcanzar un diseño final del mismo.

El siguiente formato será implementado para describir la colaboración entre los objetos.

|                           |  |
|---------------------------|--|
| <b>Responsabilidades:</b> | Responsabilidades u objetivos de la colaboración   |
| <b>Referencias</b>        | Referencias a los Contratos de Operación dentro de este mismo documento.                                   |
| <b>Poscondiciones:</b>    | Las poscondiciones del Contrato de Operación en las cuales se basa el respectivo diagrama de colaboración. |

Cuadro 5.12: Formato de colaboración de objetos

<sup>105</sup> El punto de partida de las interacciones es el cumplimiento de las poscondiciones de los contratos de operación. UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos, Craig Larman. Pearson Prentice Hall 2003, Diagramas de Interacción: Cap. 17, pág. 169

<sup>106</sup> **General Responsibility Assignment Software Patterns: Patrones de Software para la Asignación de Responsabilidades Generales.**



**5.2.2.7. Diagrama de Colaboración IntroducirLoginyPassword**

|                           |   |
|---------------------------|---|
| <b>Responsabilidades:</b> | Proporcionar al jugador una pantalla para entrada de datos de sesión.   |
| <b>Referencias</b>        | CO1   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se instanció una pantalla de entrada de datos.</li> <li>✓ Se creó un cuadro de entrada para el login.</li> <li>✓ Se creó un cuadro de entrada para el password.</li> </ul> |

Cuadro 5.13: Diagrama de Colaboración

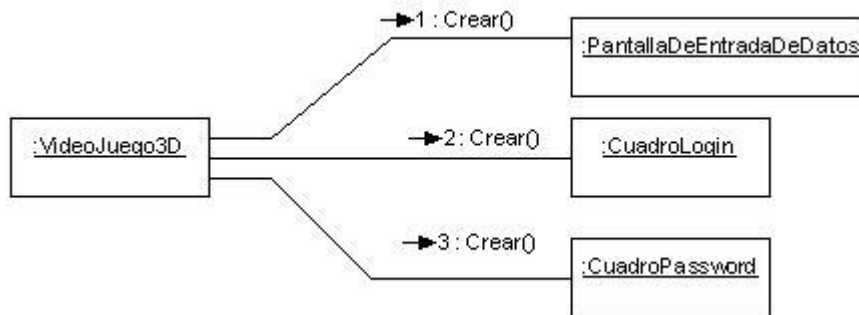


Figura 5.30: Diagrama de Colaboración Introducir Login y Password

**5.2.2.8. Diagrama de Colaboración AutenticarUsuario**

|                           |   |
|---------------------------|---|
| <b>Responsabilidades:</b> | Autenticar los datos del Jugador.   |
| <b>Referencias</b>        | CO2   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se obtuvieron del archivo de sesión los datos del jugador.</li> <li>✓ Se ha Autenticado antes de Validar al jugador para iniciar su sesión.</li> </ul> |

Cuadro 5.14: Autenticar datos del jugador

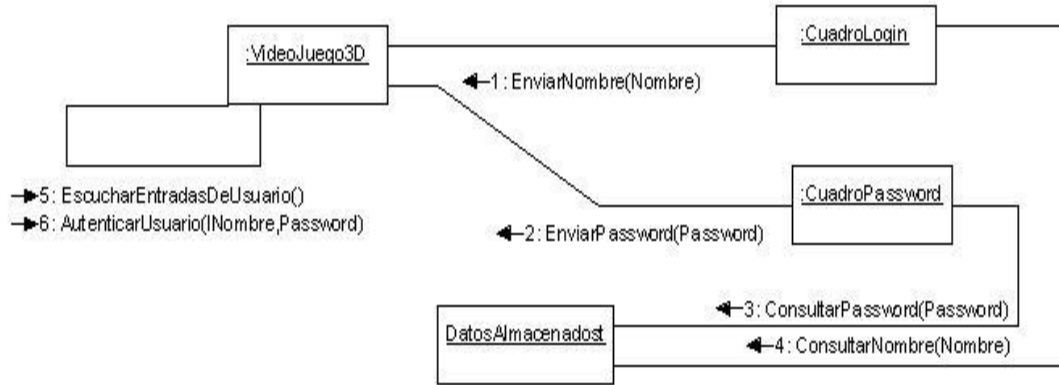


Figura .5.31: Diagrama de Colaboración AutenticarUsuario

### 5.2.2.9. Diagrama de Colaboración ValidarJugador

|                           |  |
|---------------------------|--|
| <b>Responsabilidades:</b> | Validar sesión de Jugador.   |
| <b>Referencias</b>        | CO3,CO26   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se asignó un valor a la variable interna gestión de jugador.</li> <li>✓ Se crearon tres instancias de la clase Caracter().</li> <li>✓ Se instanciaron las imágenes de fondo.</li> <li>✓ Se cargaron los sonidos de dialogo de los caracteres.</li> <li>✓ Se presentaron los caracteres para que el usuario elija.</li> <li>✓ Se validó la sesión del jugador</li> </ul> |

Cuadro 5.15: Validar jugador

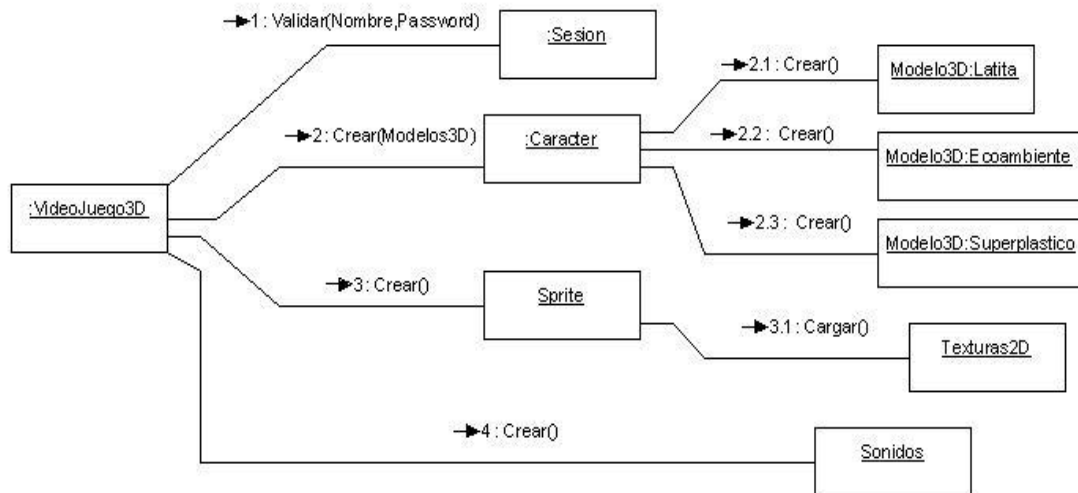


Figura .5.32: Diagrama de Colaboración ValidarJugador



**5.2.2.10. Diagrama de Colaboración MostrarCaracteres y SeleccionarCaracter**

|                           |  |
|---------------------------|--|
| <b>Responsabilidades:</b> | Validar sesión de Jugador.   |
| <b>Referencias</b>        | CO4, CO5   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se seleccionó un caracter.</li> <li>✓ Se asoció el caracter seleccionado al jugador.</li> </ul> |

Cuadro 5.16: Mostrar caracteres y seleccionar carácter

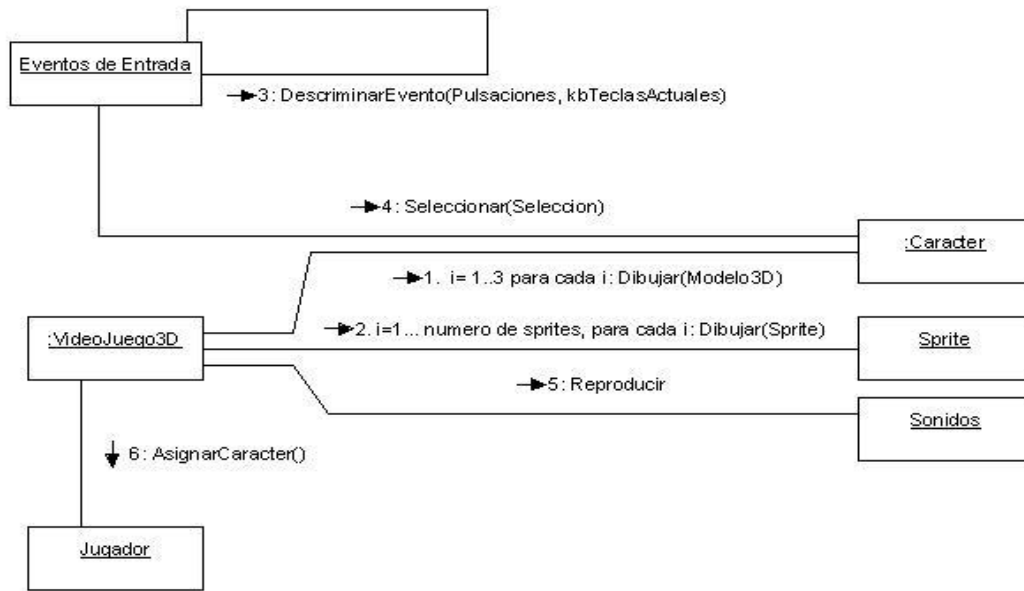


Figura 5.33: Diagrama de Colaboración MostrarCaracteres y SeleccionarCaracter

**5.2.2.11. Diagrama de Colaboración CargarRecursos**

|                           |  |
|---------------------------|--|
| <b>Responsabilidades:</b> | Validar sesión de Jugador.   |
| <b>Referencias</b>        | CO6  |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se mostró la pantalla de espera mientras se cargan los recursos.</li> <li>✓ Se instanciaron los recursos (objetos 3D) acordes al caracter seleccionado.</li> <li>✓ Se preparó el escenario virtual para el primer nivel del juego relativo al carácter previamente seleccionado.</li> </ul> |

Cuadro 5.17: Cargar Recursos



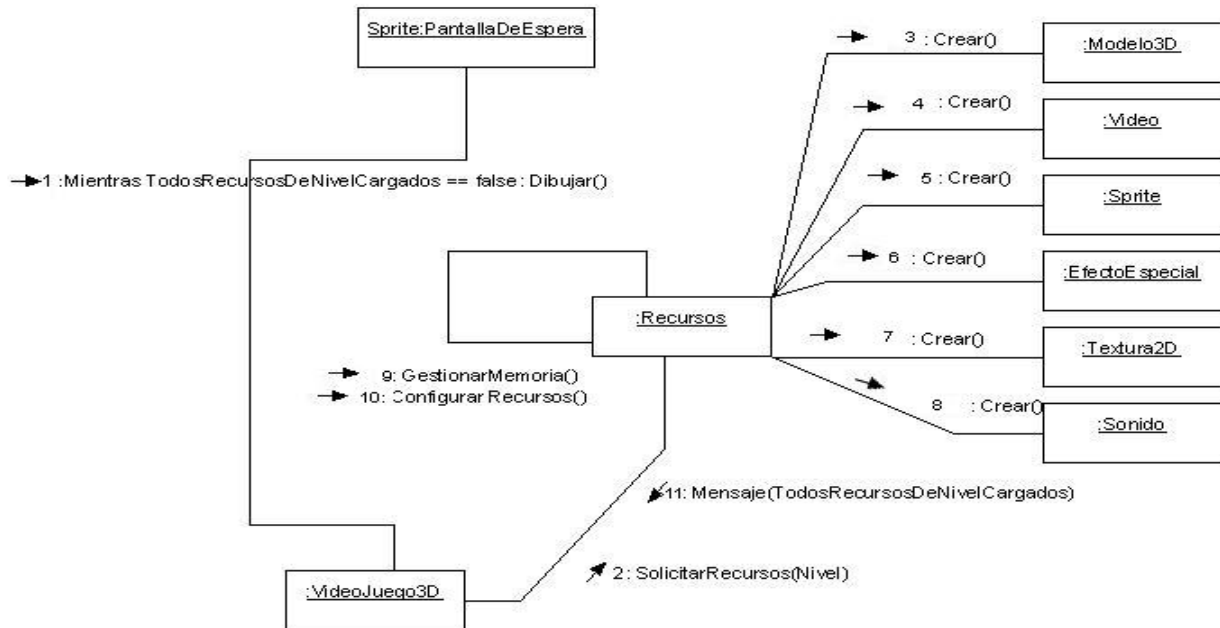


Figura 5.34: Diagrama de colaboración CargarRecurso

### 5.2.2.12. Diagrama de Colaboración DibujarNivel

|                           |   |
|---------------------------|---|
| <b>Responsabilidades:</b> | Dibujar la escena virtual, gestionar los algoritmos de inteligencia artificial, actualizar valores matriciales, Seleccionar Respuestas de Caracteres Virtuales e interacción Jugador-Character-Mundo Virtual  |
| <b>Referencias</b>        | CO7, CO20,CO23  |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se ha instanciado el nuevo nivel</li> <li>✓ Se dibujó el nuevo nivel</li> <li>✓ Se volvió al menú principal de seleccionar opciones (cuando el usuario fracasase en su avance en el nivel dado o deseado terminar la partida)</li> </ul> |

Cuadro 5.18: Dibujar nivel

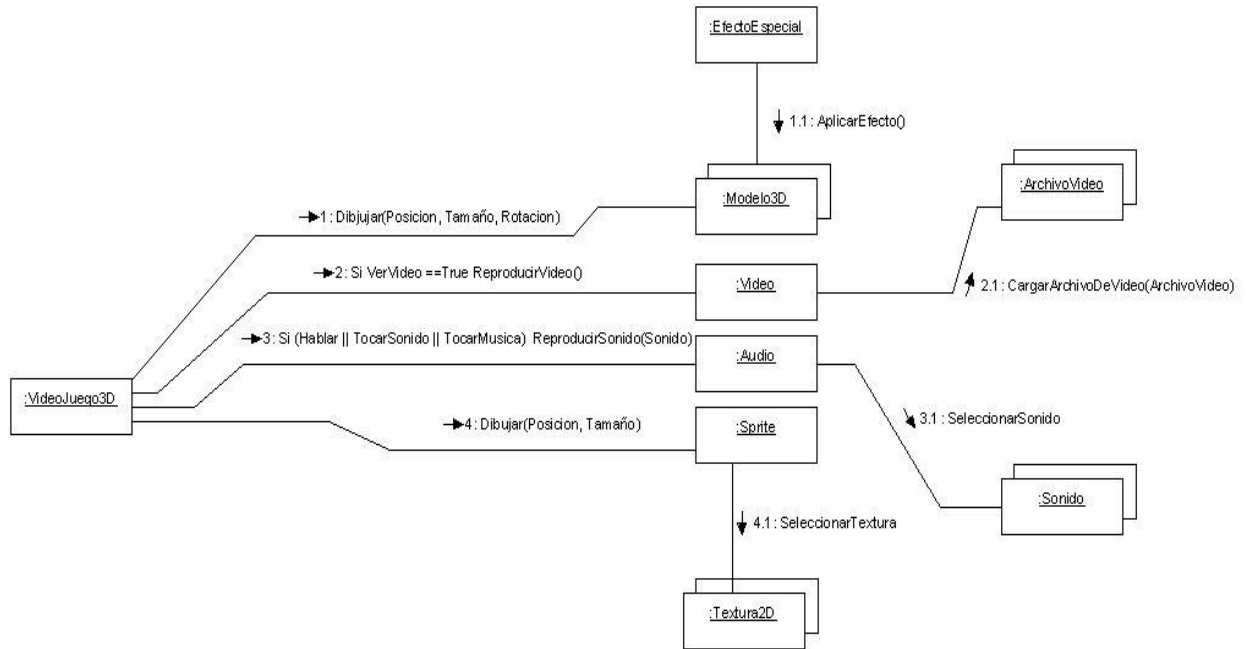


Figura 5.35: Diagrama de colaboración Dibujar nivel

### 5.2.2.13. Diagrama de Colaboración Jugar

|                           |  |
|---------------------------|--|
| <b>Responsabilidades:</b> | Escuchar eventos del jugador a través del teclado, mouse y/o control de juegos. Responder a las entradas del usuario. Gestionar lógica de juego  |
| <b>Referencias</b>        | CO8,CO20,CO21,CO22   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se crearon las instancias de gestión lógica del videojuego.</li> <li>✓ Se instanció un nuevo nivel (cuando el usuario culmine la partida dada)</li> <li>✓ Si se desea pausar, SeleccionarOpciones fue cargada.</li> </ul> |

Cuadro 5.19: Diagrama de Colaboración Jugar

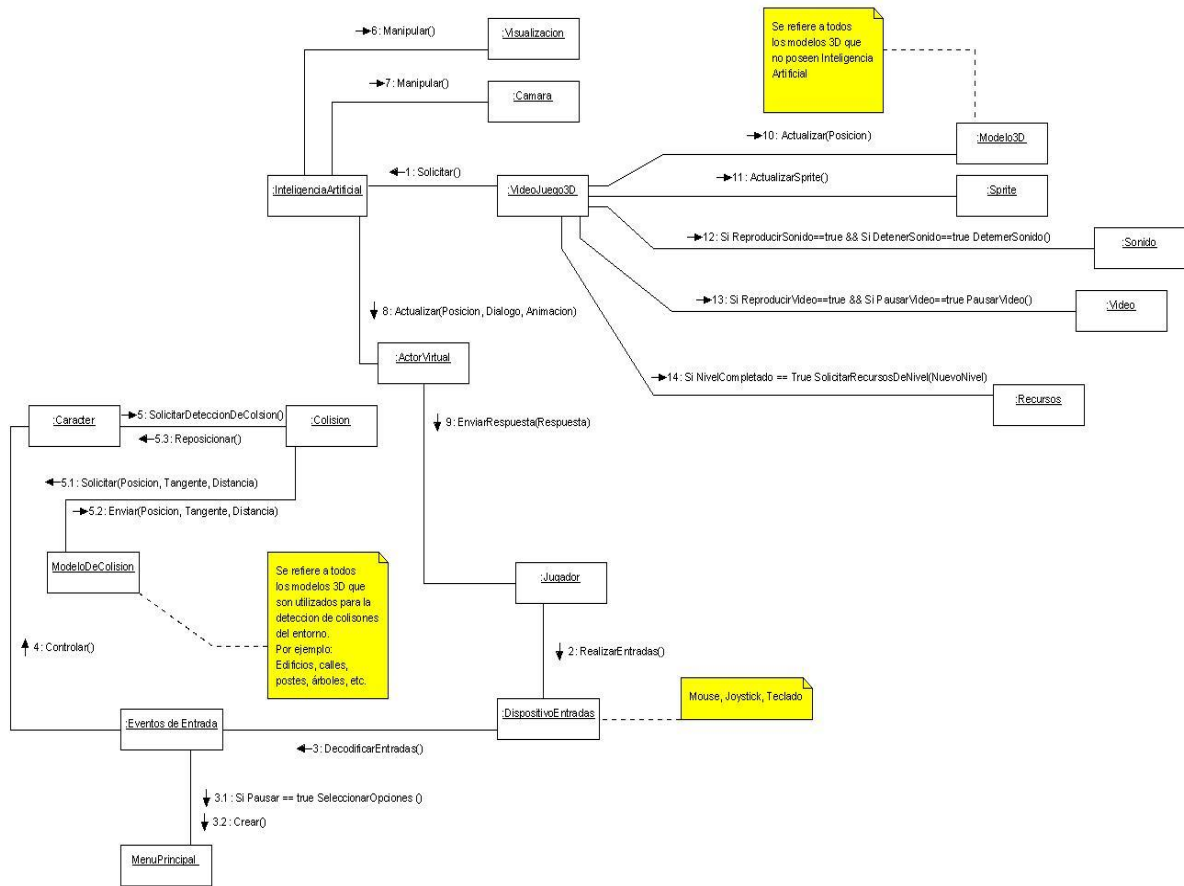


Figura 5.36: Diagrama de colaboración Jugar

**5.2.2.14. Diagrama de Colaboración MostrarOpcionesConfigurarJuego**

|                           |   |
|---------------------------|---|
| <b>Responsabilidades:</b> | Mostrar al usuario las opciones para configurar el juego.<br>Permitir la selección de una opción de configuración (Video, controles y velocidad)  |
| <b>Referencias</b>        | CO9, CO10, CO11, CO12, CO13, CO14, CO15, CO16, CO17, CO18, CO19   |
| <b>Poscondiciones:</b>    | <ul style="list-style-type: none"> <li>✓ Se instanció el despliegue de opciones principales.</li> <li>✓ Se cargo la configuración actual del juego</li> <li>✓ Se desplegaron los atributos de video a configurar</li> <li>✓ Se asignaron valores a los atributos de video</li> <li>✓ Se seleccionó una opción de configuración del juego.</li> <li>✓ Se creó una instancia de video (archivo de video)</li> <li>✓ Se creó una instancia de controles (creación de instancia)</li> <li>✓ Se creó una instancia de velocidad (creación de instancia)</li> </ul> |

Cuadro 5.20: Mostrar Opciones Configurar Juego

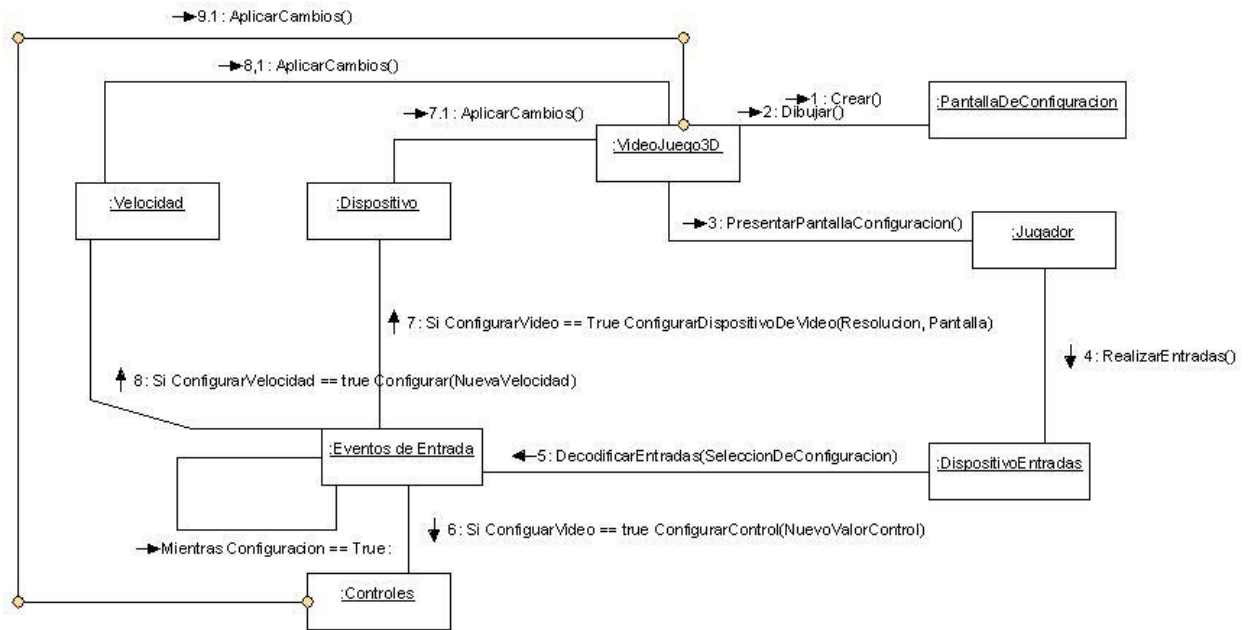


Figura 5.37: Diagrama de colaboración MostrarOpcionesConfigurarJuego

### 5.2.3. Integración de Patrones GRASP<sup>107</sup>

Los patrones GRASP describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y explicable. Las responsabilidades están relacionadas con las obligaciones de un objeto en cuanto a su comportamiento. Básicamente, estas responsabilidades son de los siguientes dos tipos:

- **Conocer:**
  - Conocer los datos privados encapsulados.
  - Conocer los objetos relacionados.
  - Conocer las cosas que puede derivar o calcular.
- **Hacer:**
  - Hacer algo él mismo, como crear un objeto o hacer un cálculo.
  - Iniciar una acción en otros objetos.
  - Controlar y coordinar actividades en otros objetos.

<sup>107</sup> General Responsibility Assignment Software Patterns, Patrones de Software Generales para la Asignación de Responsabilidades



Se pueden destacar 5 patrones principales que son: Experto, Creador, Alta cohesión, Bajo acoplamiento, Controlador. El siguiente cuadro muestra una descripción del problema principal en el desarrollo del diseño y la solución, para cada patrón.

| Nombre del patrón                       | Problema   | Solución  |
|---|--|---|
| <b>Expert - Experto</b>                 | ¿Cuál es un principio general para asignar responsabilidades a los objetos?                  | Asignar una responsabilidad al experto en información – la clase que tiene la información necesaria para la realización de la asignación.   |
| <b>Creator - Creador</b>                | ¿Quién debería ser el responsable de la creación de una nueva instancia de alguna clase?     | Asignar a la clase B la responsabilidad de crear una instancia de clase A si se cumple uno o más de los casos siguientes:<br>B agrega objetos de A<br>B contiene objetos de A<br>B registra instancias de objetos de A<br>B utiliza más estrechamente objetos de A.<br>B tiene datos de inicialización que se pasarán a un objeto de A cuando sea creado (por tanto, B es un Experto con respecto a la creación de A).<br>B es un creador de los objetos A.   |
| <b>Low Coupling - Bajo Acoplamiento</b> | ¿Cómo soportar bajas dependencias, bajo impacto del cambio e incremento de la reutilización? | Asignar una responsabilidad de manera que el acoplamiento permanezca bajo.  |
| <b>High Cohesion - Alta cohesión</b>    | ¿Cómo mantener la complejidad manejable?   | Asignar una responsabilidad de manera que la cohesión permanezca alta.  |
| <b>Controller - Controlador</b>         | ¿Quién debería ser el responsable de gestionar un evento de entrada al sistema?              | Asignar una responsabilidad de recibir o manejar un mensaje de evento del sistema a una clase que representa una de las opciones siguientes:<br>Representa el sistema global, dispositivo o subsistema.<br>Representa un caso de uso en el que tiene lugar el evento del sistema a menudo denominado <nombre del caso de uso> Manejador, <nombre del caso de uso> coordinador, <nombre del caso de uso> Sesión.<br>•Utilizar la misma clase controlador para todos los eventos del sistema en el mismo escenario de caso de uso.<br>•Informalmente, una sesión es una instancia de una conversación con un actor. |

Cuadro 5.21: Cinco Patrones Básicos GRASP



### **5.2.3.1. Descripción de Cada Patrón:**

#### Experto:

Un patrón experto se refiere a las clases de objetos capaces de realizar determinadas tareas. Conocidas como 'Experto'. Por ejemplo, la clase Modelo3D es 'experta' en gestionar todo lo relacionado a un modelo 3D.

#### Creador:

Un patrón que asigna a una clase u objeto la capacidad de 'crear' otros objetos. Por ejemplo, la misma clase Modelo3D, permite instanciar determinados objetos.

#### Alta Cohesión:

Es una medida de cuan relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. En otras palabras una clase con alta cohesión realiza pocas cosas y es fácil de comprender y no realiza un trabajo excesivo o tiene responsabilidades que estén estrechamente relacionadas a su funcionalidad exclusiva. Son fáciles de reutilizar y mantener, no se ven muy afectadas por cambios realizados.

#### Bajo Acoplamiento:

Es una medida de la fuerza con que una clase está conectada a otras clases con las que las conoce y con que recurre a ellas. Una clase con bajo (o débil) acoplamiento no depende de muchas otras. En otras palabras, entre menos dependencia se tiene de otras clases, mejor y más bajo es el acoplamiento.

Para aplicar los patrones GRASP a las clases y objetos que componen al videojuego se tendrá como punto de partida, el modelo de dominio y los diagramas de colaboración para "refinar" el diseño a través de la implementación de los patrones antes mencionados. Para cada diagrama, se aplicara una alta cohesión, bajo acoplamiento, se definirá uno o varios objetos controlador y se señalaran los objetos encargados de la creación de nuevas instancias de clases. El objetivo de este refinamiento es contestar las preguntas de la tabla anterior a través de un diseño de diagramas fácil de entender para los desarrolladores de la aplicación de software aquí descrita.

El siguiente cuadro, se utilizara para explicar en qué grado, cumple un diagrama con los diferentes patrones.



| Nombre del patrón        | Objeto                   | Criterio                    |
|--------------------------|--------------------------|-----------------------------|
| <b>Experto</b>           | Nombre de Clase u Objeto | Porqué se cumple el patrón. |
| <b>Creador</b>           | Nombre de Clase u Objeto | Porqué se cumple el patrón. |
| <b>Bajo Acoplamiento</b> | Nombre de Clase u Objeto | Porqué se cumple el patrón. |
| <b>Alta cohesión</b>     | Nombre de Clase u Objeto | Porqué se cumple el patrón. |
| <b>Controlador</b>       | Nombre de Clase u Objeto | Porqué se cumple el patrón. |

Cuadro 5.22: Cumplimiento de los patrones GRASP

### 5.2.3.2. Diagrama de Colaboración IntroducirLoginyPassword

| Nombre del patrón        | Objeto                        | Criterio  |
|--------------------------|-------------------------------|---|
| <b>Experto</b>           | VideoJuego3D                  | Es Experto en Desplegar una PantallaDeEntradaDeDatos  |
| <b>Creador</b>           | VideoJuego3D                  | Desde aquí se puede crear:<br>✓ Una PantallaDeEntradaDeDatos<br>✓ CuadrosLogin<br>✓ CuadrosPassword |
| <b>Bajo Acoplamiento</b> | CuadroLogin<br>CuadroPassword | Únicamente están conectadas a Videojuego3D y DatosAlmacenados (ver AutenticarUsuario)               |
| <b>Alta cohesión</b>     | CuadroLogin<br>CuadroPassword | Solo dependen de Videojuego3D y cada uno se limita a su función.                                    |
| <b>Controlador</b>       |                               |   |

Cuadro 5.23: Patrones aplicados a introducirLoginyPassword

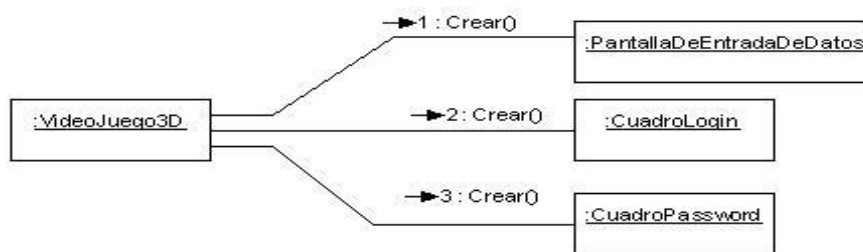


Figura 5.38: Diagrama de colaboración IntroducirLoginPassword



**5.2.3.3. Diagrama de Colaboración AutenticarUsuario**

| Nombre del patrón | Objeto           | Criterio  |
|-------------------|------------------|---|
| Experto           | DatosAlmacenados | Es Experto en Gestión de Almacenamiento de datos. |
| Creador           |                  |   |
| Bajo Acoplamiento |                  |   |
| Alta cohesión     |                  |   |
| Controlador       | VideoJuego3D     | Controla la gestión de Autenticar al Usuario      |

Cuadro 5.24: Aplicación de patrones GRASP a AutenticarUsuario

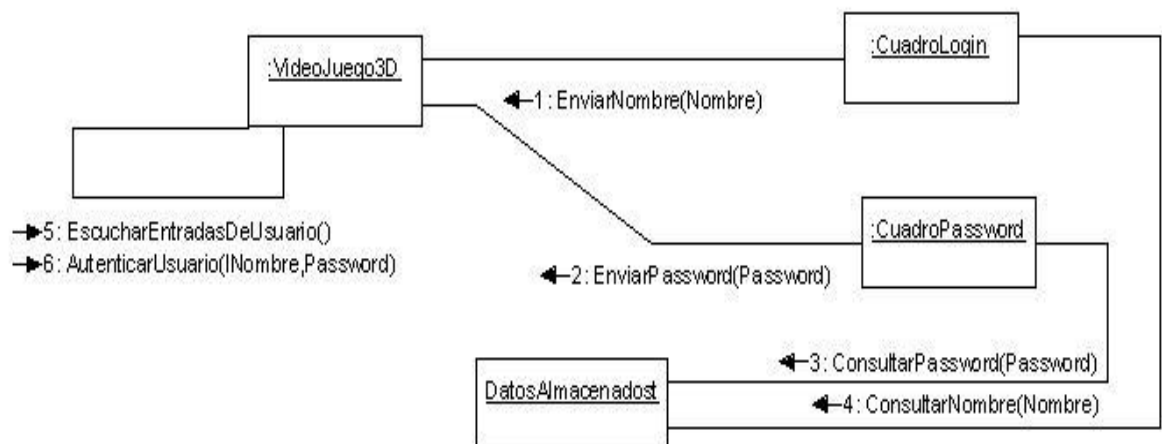


Figura 5.39: Diagrama de colaboración AutenticarUsuario





**5.2.3.4. Diagrama de Colaboración ValidarJugador**

| Nombre del patrón        | Objeto                                  | Criterio  |
|--------------------------|---|---|
| <b>Experto</b>           | Caracter<br>Sprite                      | Es Experto en creación y administración de las instancias de caracteres principales: Latita, Ecoambiente y Superplástico.<br><br>Experto en la gestión de creación y carga de texturas para efectos con sprites |
| <b>Creador</b>           | Caracter<br>Videojuego3D                | Es Creador de los modelos Animados 3D de los caracteres principales: Latita, Ecoambiente y Superplástico.<br><br>Crea los Sonidos a medida que sean necesarios.   |
| <b>Bajo Acoplamiento</b> | Modelo3D<br>Texturas2D<br>Sonidos       | Se limitan a depender de Caracter<br>Se limita a depender de Sprite<br>Se limita a Depender de VideoJuego3D   |
| <b>Alta cohesión</b>     | Sesión<br>Caracter<br>Sprite<br>Sonidos | Todos estos objetos tienen responsabilidades que están limitadas a sus funciones o razón de ser.  |
| <b>Controlador</b>       |   |   |

Cuadro 5.25: Aplicación de patrones GRASP a ValidarJugador

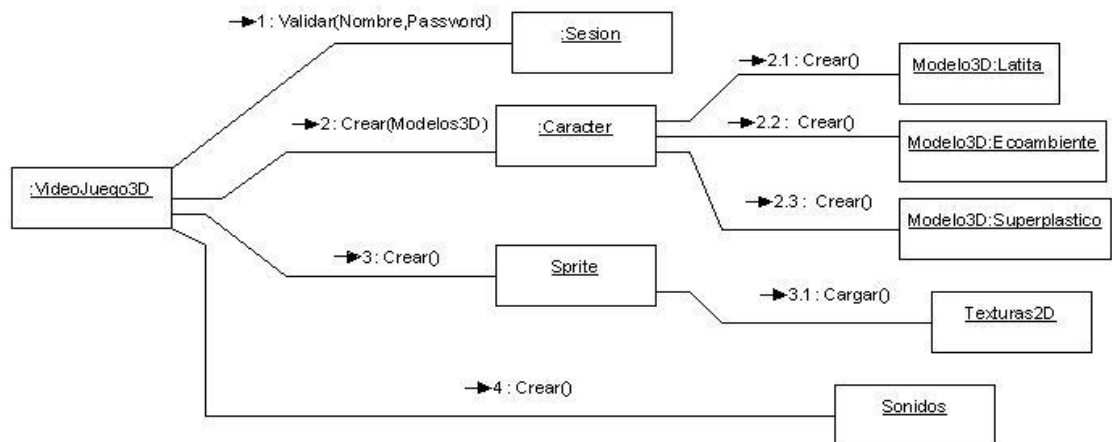


Figura 5.40: Diagrama de colaboración ValidarJugador



**5.2.3.5. Diagrama de Colaboración MostrarCaracteres y SeleccionarCaracter**

| Nombre del patrón        | Objeto                      | Criterio   |
|--------------------------|-----------------------------|--|
| <b>Experto</b>           | VideoJuego3D<br>Sonidos     | Es Experto en Dibujar Caracter y Sprite<br>Experto en Reproducir Sonidos cuando son solicitados por VideoJuego3D |
| <b>Creador</b>           |                             |  |
| <b>Bajo Acoplamiento</b> | EventosdeEntrada            | No depende de muchas clases ni muchos objetos.   |
| <b>Alta cohesión</b>     | Sonidos<br>EventosdeEntrada | Su función es la gestión de sonidos y controlar eventos de entrada   |
| <b>Controlador</b>       | Eventos de Entrada          | Controla los eventos de entrada de usuario.  |

Cuadro 5.26: Aplicación de patrones GRASP a MostrarCaracteres y SeleccionarCaracter

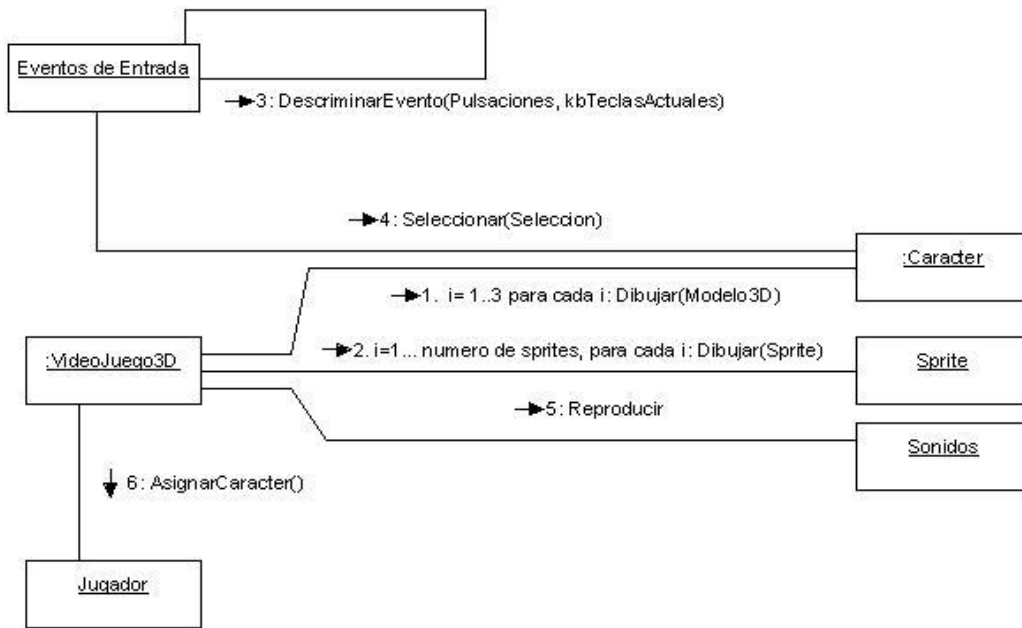


Figura 5.41: Diagrama de colaboración MostrarCaracteres y SeleccionarCaracter



**5.2.3.6. Diagrama de Colaboración CargarRecursos**

| Nombre del patrón | Objeto                   | Criterio  |
|-------------------|--------------------------|---|
| Experto           | Recursos                 | Es Experto en Carga y Creación de Recursos/   |
| Creador           | Recursos<br>Videojuego3D | Instancia los recursos para modelos, video, sprites, efectos especiales, texturas2D y sonido.<br>Crea una PantallaDeEspera. |
| Bajo Acoplamiento | Recursos                 | Solo depende de VideoJuego3D  |
| Alta cohesión     | Recursos                 | Su única función es la gestión de Recursos  |
| Controlador       |                          |   |

Cuadro 5.27: Aplicación de patrones GRASP a CargarRecursos

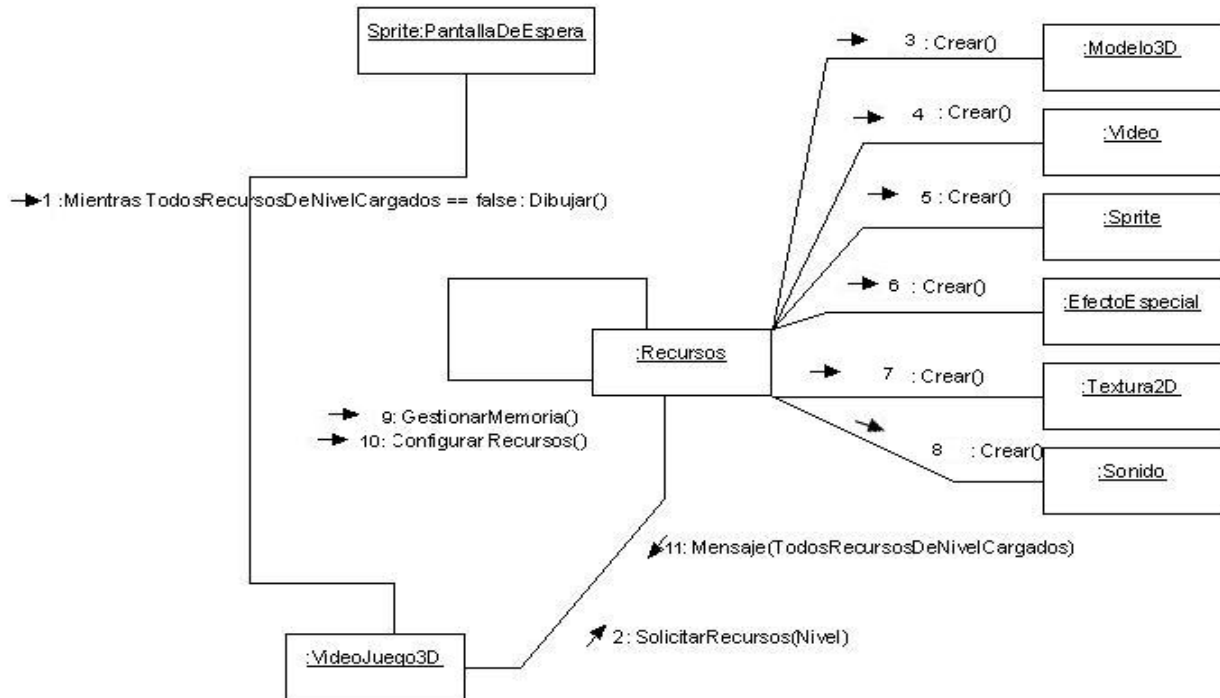


Figura 5.42: Diagrama de colaboración CargarRecursos



**5.2.3.7. Diagrama de Colaboración - DibujarNivel**

| Nombre del patrón        | Objeto  | Criterio  |
|--------------------------|---|---|
| <b>Experto</b>           | VideoJuego3D<br>Modelo3D<br>Video<br>Audio<br>Sprite  | VideoJuego3D es experto en Dibujar y solicitar Reproducción de sonidos y videos<br>Modelos3D es experto en estructuras de Modelos 3D estáticos y animados (ActorVirtual y Caracteres Principales).<br>Video es experto en reproducción de archivos de videos<br>Audio se especializa en reproducción y gestión de sonidos<br>Sprite se encarga presentar efectos de sprites y texturas 2D |
| <b>Creador</b>           | Recursos  | Instancia los recursos para modelos, video, sprites, efectos especiales, texturas2D y sonido.   |
| <b>Bajo Acoplamiento</b> | EfectoEspecial<br>ArchivoVideo<br>Sonido<br>Textura2D | EfectoEspecial Solo depende de Modelo3D<br>Las instancias de ArchivoVideo solo dependen de Video.<br>Las instancias de Sonido solo dependen de Audio<br>Las instancias de Textura2D solo dependen de Sprite   |
| <b>Alta cohesión</b>     | Recursos  | Su única función es la gestión de Recursos  |
| <b>Controlador</b>       | VideoJuego3D  | Controla los eventos asociados con la reproducción de sonidos y videos.   |

Cuadro 5.28: Aplicación de patrones GRASP a DibujarNivel

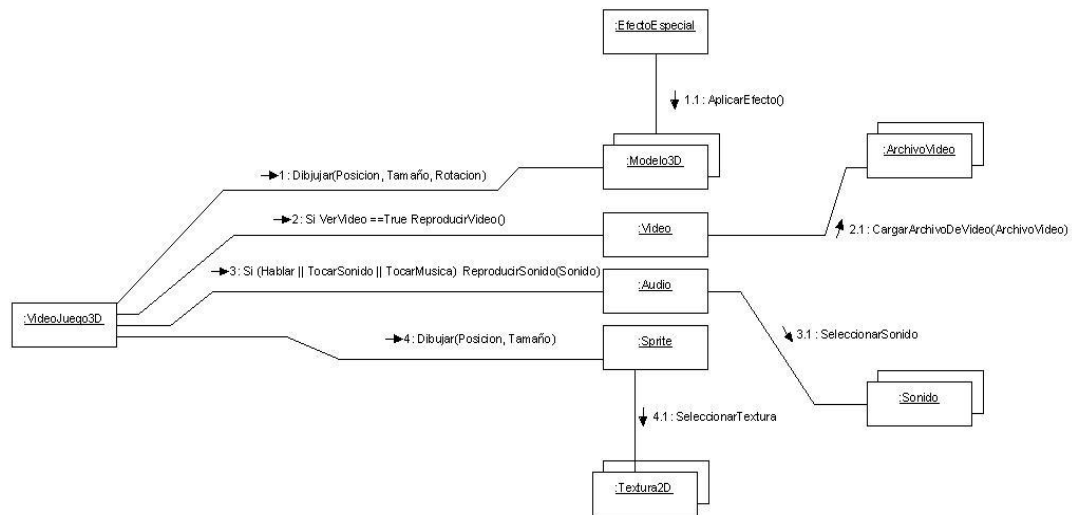


Figura 5.43: Diagrama de colaboración DibujarNivel



**5.2.3.8. Diagrama de Colaboración Jugar**

| Nombre del patrón        | Objeto  | Criterio  |
|--------------------------|---|---|
| <b>Experto</b>           | InteligenciaArtificial  | Especializada en Manipular el objeto Cámara y la Visualización del entorno. Al mismo tiempo, cuando Videojuego lo requiera, le proveerá de inteligencia artificial a los objetos ActorVirtual que son instancias de Modelo3D, actualizando posiciones y diálogos para cada uno. |
|                          | Cámara  | Experto en las funciones de la cámara virtual.  |
|                          | Visualización   | Se encarga exclusivamente de la gestión de visualización del entorno virtual.   |
|                          | Colisión  | Experto en detección de colisiones entre el entorno virtual y el caracter principal.  |
| <b>Creador</b>           |   |   |
| <b>Bajo Acoplamiento</b> | ModeloDeColision  | Depende exclusivamente de Colisión.   |
|                          | Modelo3D<br>Video<br>Audio<br>Recursos<br><br>Cámara<br>Visualización | Dependen únicamente de VideoJuego3D<br><br>Dependen de InteligenciaArtificial   |
| <b>Alta cohesión</b>     | Recursos<br><br>InteligenciaArtificial<br><br>Sprite                  | Su única función es la gestión de Recursos<br>Su única función es la gestión de Inteligencia artificial para los actores virtuales.<br><i>No cumple con el principio debido a su sobre funcionalidad .</i>  |
| <b>Controlador</b>       | InteligenciaArtificial  | Controla las actitudes y comportamiento de los actores virtuales detectando eventos y situaciones aleatorias para 'dar' vida a los actores virtuales.   |
|                          | Eventos de Entrada  | Controla los eventos de entrada de usuario.   |

Cuadro 5.29: Aplicación de patrones GRASP a Jugar

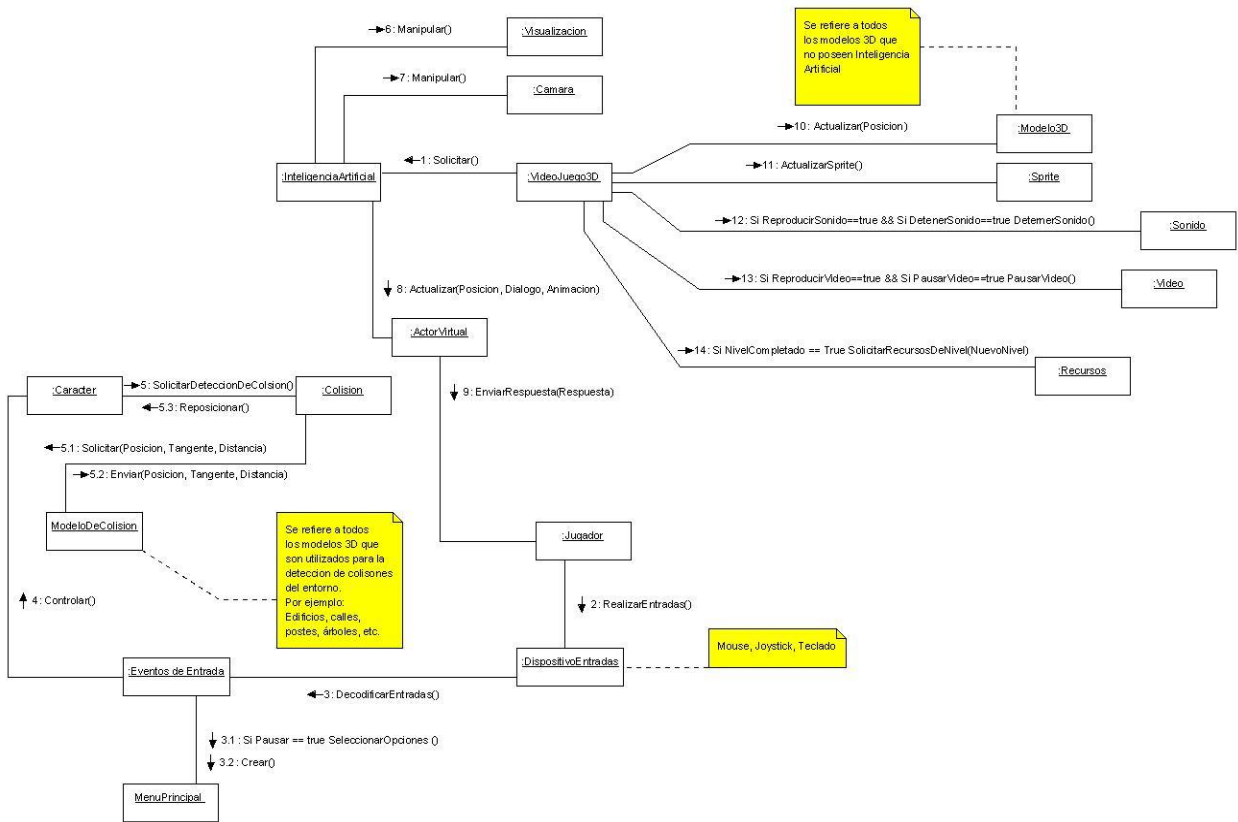


Figura 5.44: Diagrama de colaboración Jugar

### Redefinición

El objeto **Sprite** estará a cargo de dibujar texturas 2D y de efectos especiales; debido a que no cumple con el principio de alta cohesión (tiene doble carga funcional), se redefinirá en dos paquetes.



**5.2.3.9. Diagrama de Colaboración MostrarOpcionesConfigurarJuego**

| Nombre del patrón        | Objeto  | Criterio   |
|--------------------------|---|--|
| <b>Experto</b>           | Velocidad<br>Dispositivo<br>Controles                       | Experto en Configurar Velocidad<br>Experto en la configuración de video y renderizado de Entorno Virtual.<br>Experto en reconfiguración de controles de dispositivos de entrada. |
| <b>Creador</b>           | Videojuego3D  | Crea una PantallaDeEspera.   |
| <b>Bajo Acoplamiento</b> | PantallaDeEspera  | Su creación y dibujado solo depende de Videojuego3D  |
| <b>Alta cohesión</b>     | Velocidad<br>Dispositivo<br>Controles<br>Eventos de Entrada | Se limitan a la funcionabilidad exclusiva para cada uno de estos objetos.  |
| <b>Controlador</b>       | Eventos de Entrada  | Controla los eventos de entrada de usuario.  |

Cuadro 5.30: Aplicación de patrones GRASP a MostrarOpcionesConfigurarJuego

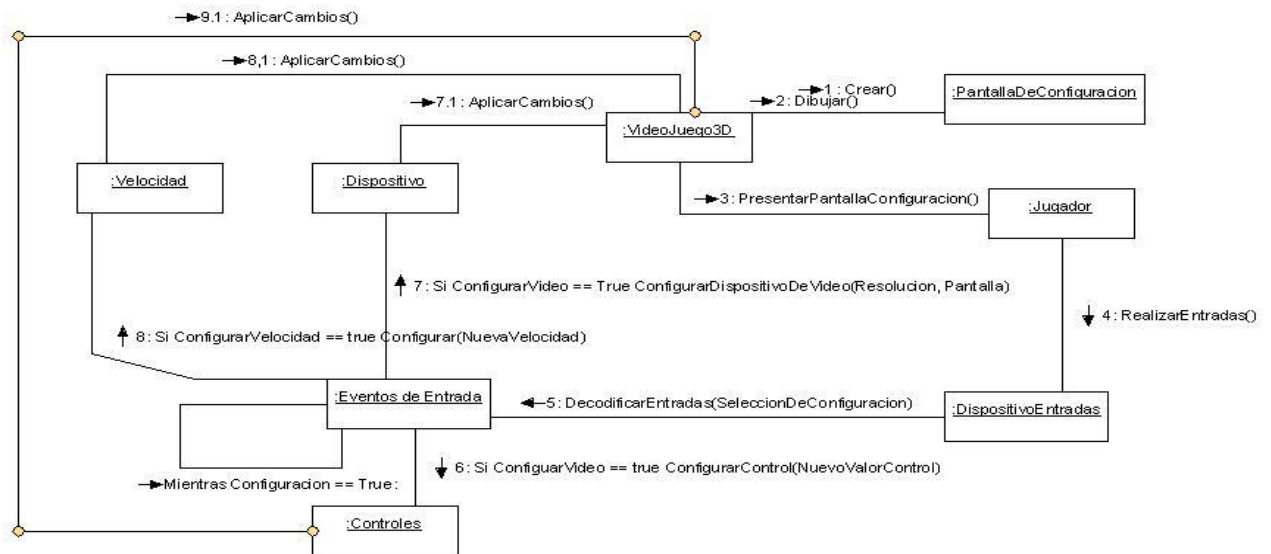


Figura 5.45: Diagrama de colaboración MostrarOpcionesConfigurarJuego



### 5.2.4. Diagramas de Estado

Un diagrama de estado describe los eventos y estados más importantes de un objeto, así como su comportamiento.

Un estado identifica un período de tiempo (no instantáneo) en la vida del objeto durante el cual está esperando alguna operación, tiene cierto comportamiento característico o puede recibir cierto tipo de estímulos.

Un evento es un acontecimiento importante o digno de señalar; la transacción es una relación entre dos estados, indica que, cuando ocurre un evento, el objeto pasa del estado anterior al siguiente. Cuando un estado se encuentra dentro de otros, se conoce como: sub-estados.

Simbología

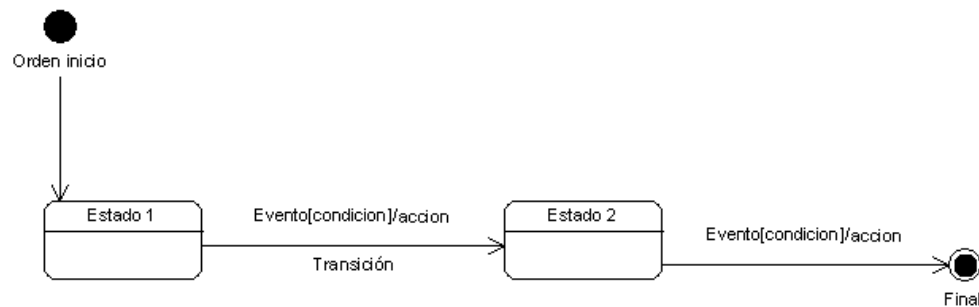


Figura 5.46: Diagrama de Estado

El icono para el estado es un rectángulo de vértices redondeados, un círculo relleno se interpreta como el punto inicial de una secuencia de estados, y una diana representa al punto final. Las flechas representan una transición entre 2 estados, como se muestra en la figura, una transición se puede etiquetar para indicar la condición que se debe evaluar para producir el cambio de estado y la acción que se llevará a cabo si la condición llega a ser cierta.

Los diagramas de estado resultan adecuados para describir el comportamiento de un objeto a través de diferentes casos de uso, sin embargo, no resultan del todo adecuados para describir el comportamiento que incluye a una serie de objetos colaborando entre sí. Por lo tanto, resulta útil combinar los diagramas de estado con los de colaboración.<sup>108</sup>

<sup>108</sup> Craig Larman, UML y Patrones, Prentice Hall, 2002, Pág 379.





### 5.2.4.1. Diagrama de estado Cargar aplicación

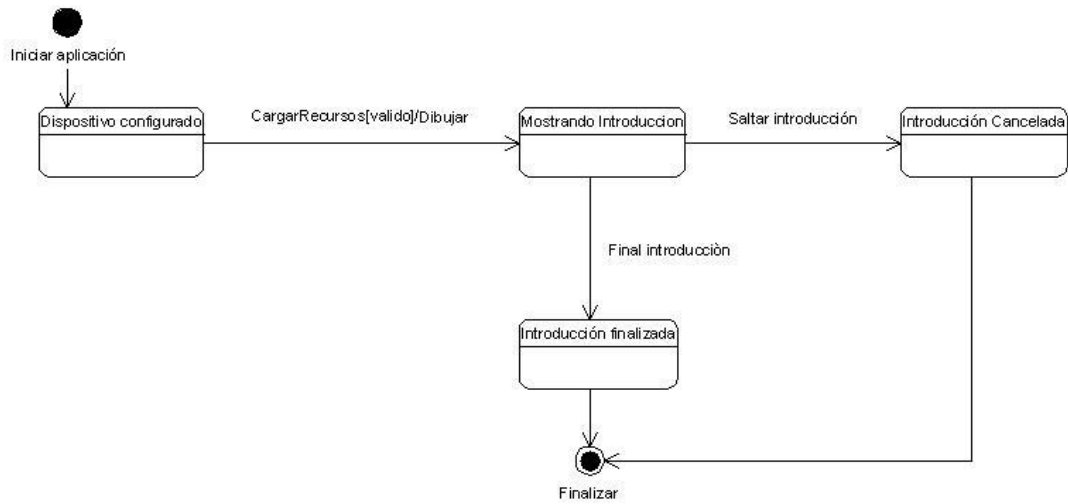


Figura 5.47: Diagrama de estado cargar aplicación

### 5.2.4.2. Diagrama de estado Seleccionar opciones

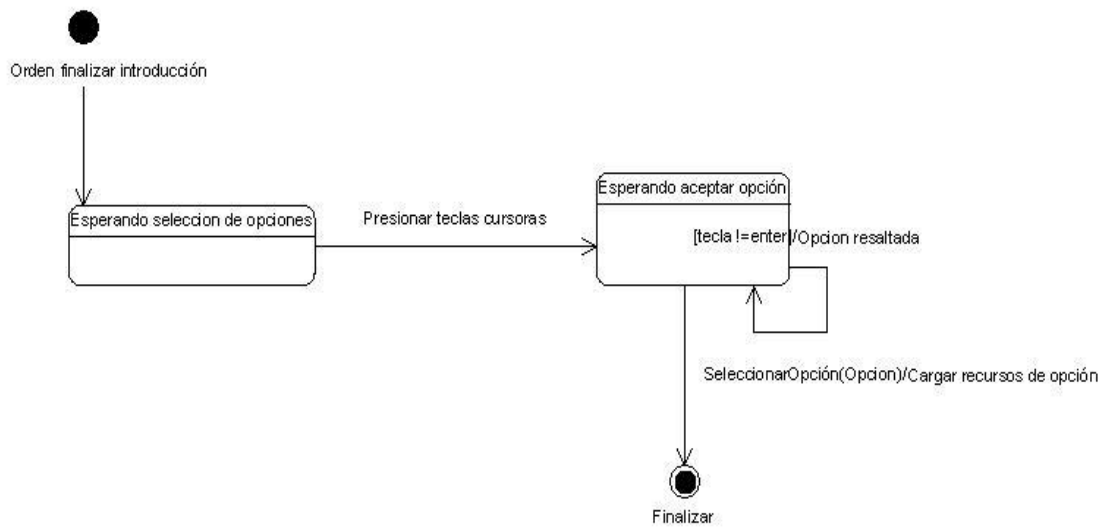


Figura 5.48: Diagrama de estado seleccionar opciones



### 5.2.4.3. Diagrama de estado Iniciar sesión

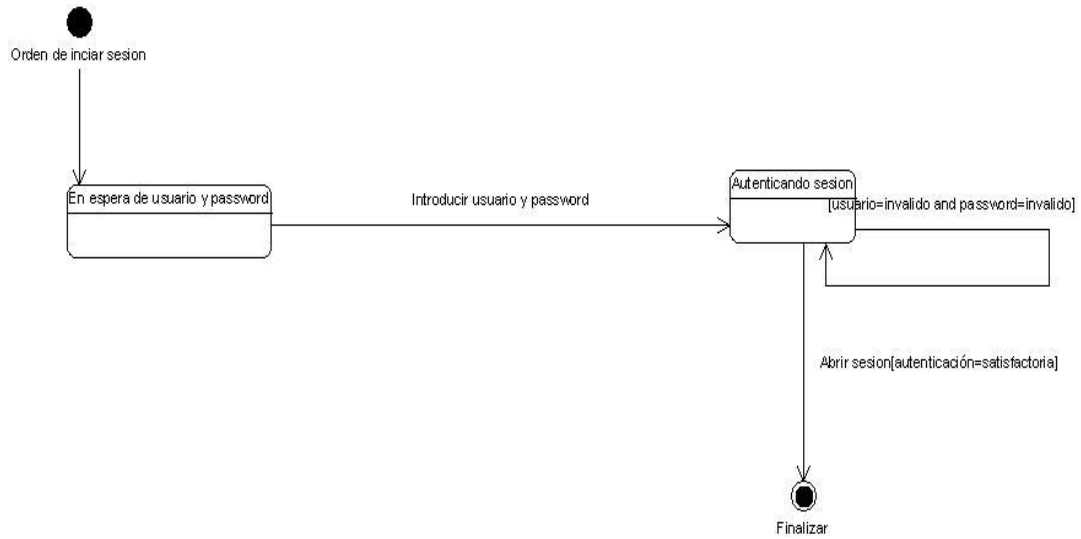


Figura 5.49: Diagrama de estado iniciar sesión

### 5.2.4.4. Diagrama de estado Cargar juego almacenado

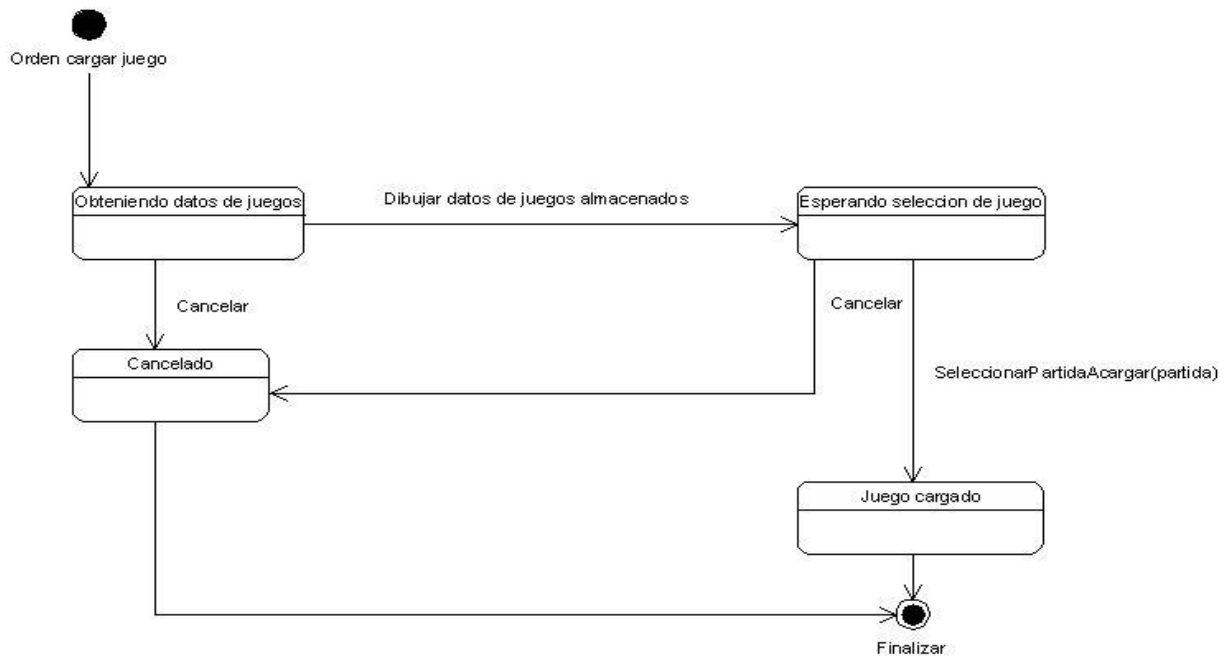


Figura 5.50: Diagrama de estado juego almacenado



### 5.2.4.5. Diagrama de estado *Mostrar bitácora*

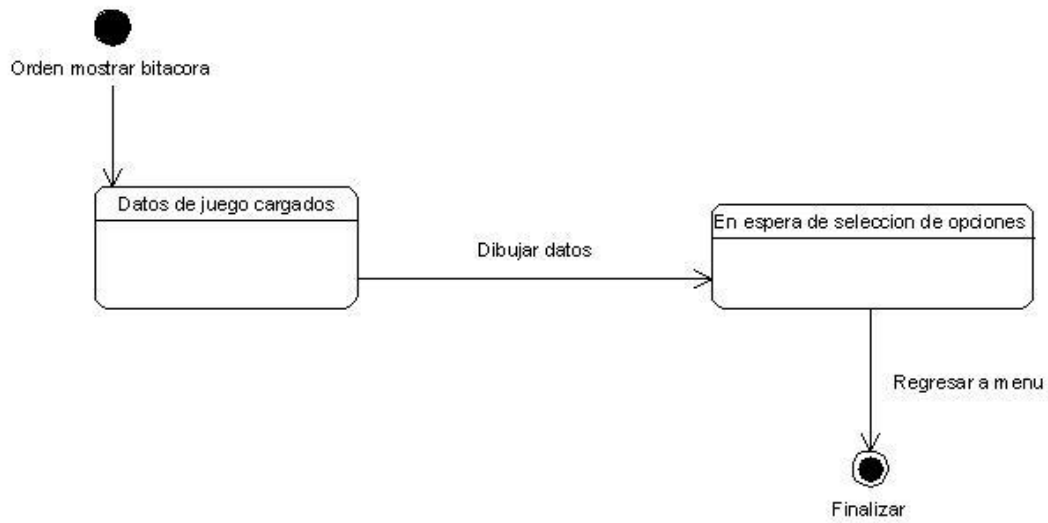


Figura 5.51: Diagrama de estado mostrar bitácora

### 5.2.4.6. Diagrama de estado *Mostrar demo*

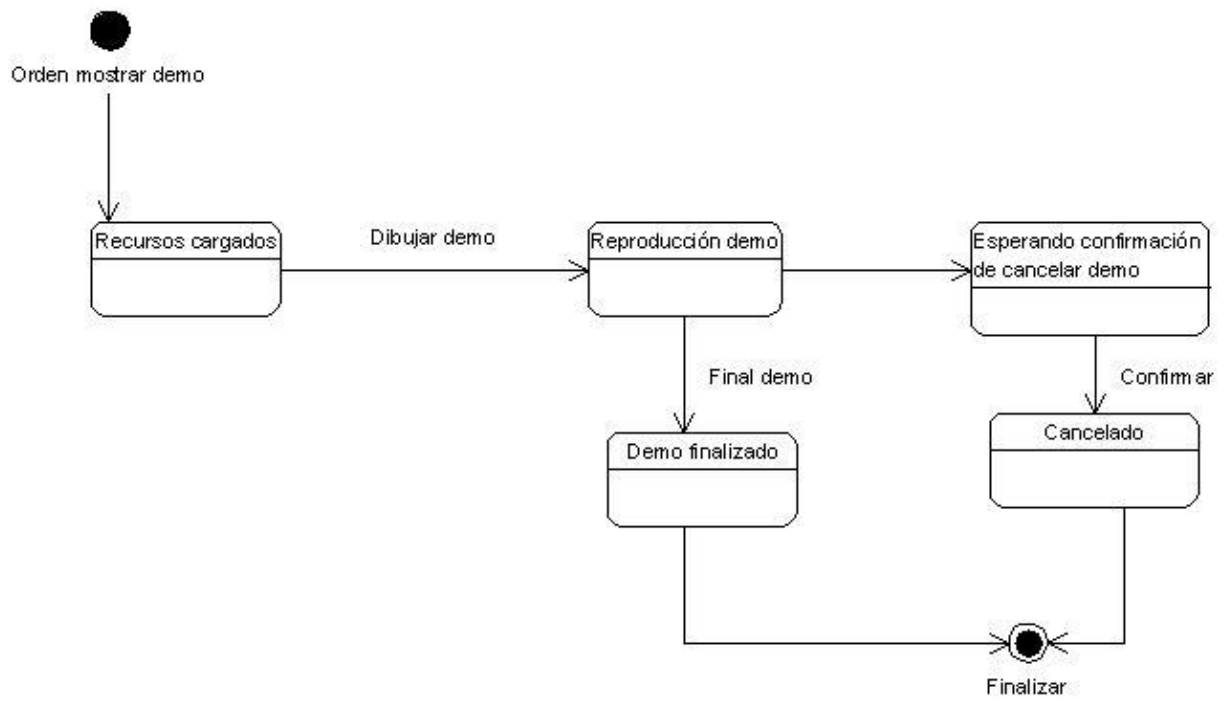


Figura 5.52: Diagrama de estado mostrar demo



### 5.2.4.7. Diagrama de estado Configurar Juego

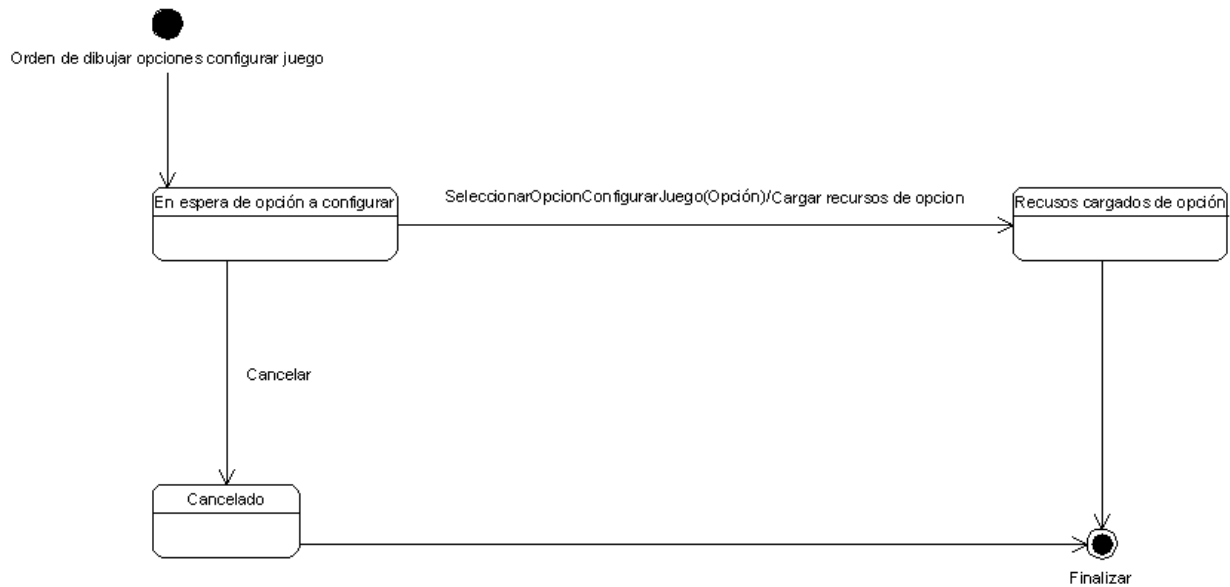


Figura 5.53: Diagrama de estado configurar juego

### 5.2.4.8. Diagrama de estado Configurar controles

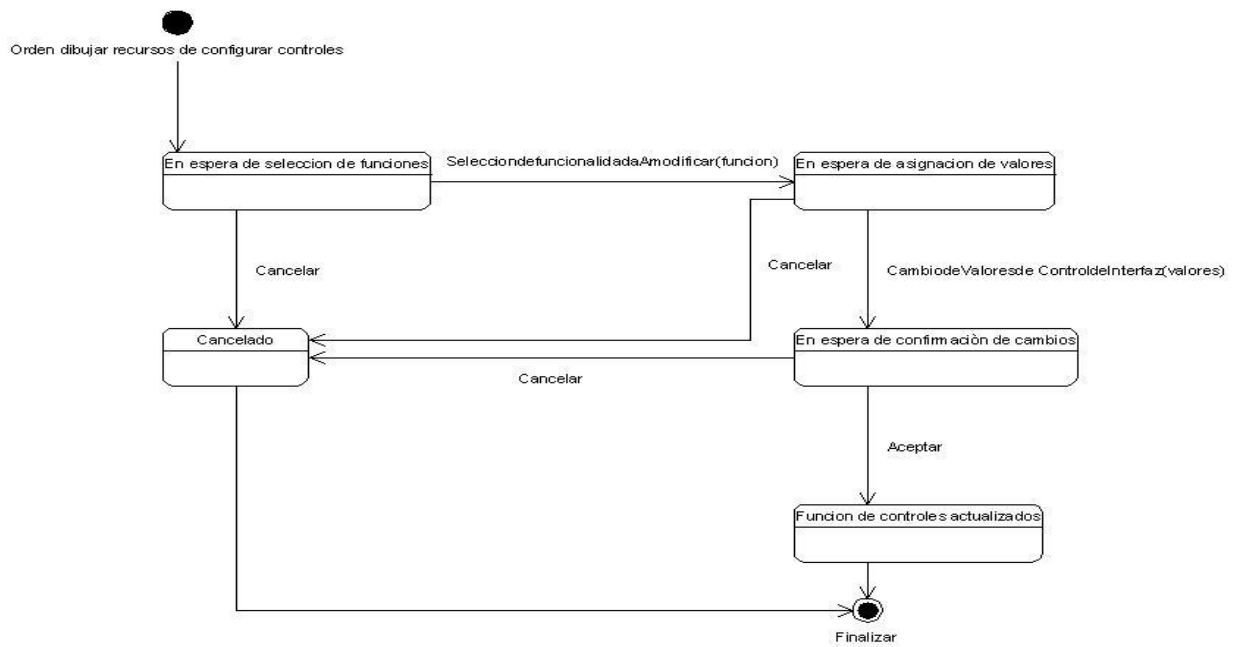


Figura 5.54: Diagrama de estado configurar juego



### 5.2.4.9. Diagrama de estado Configurar video

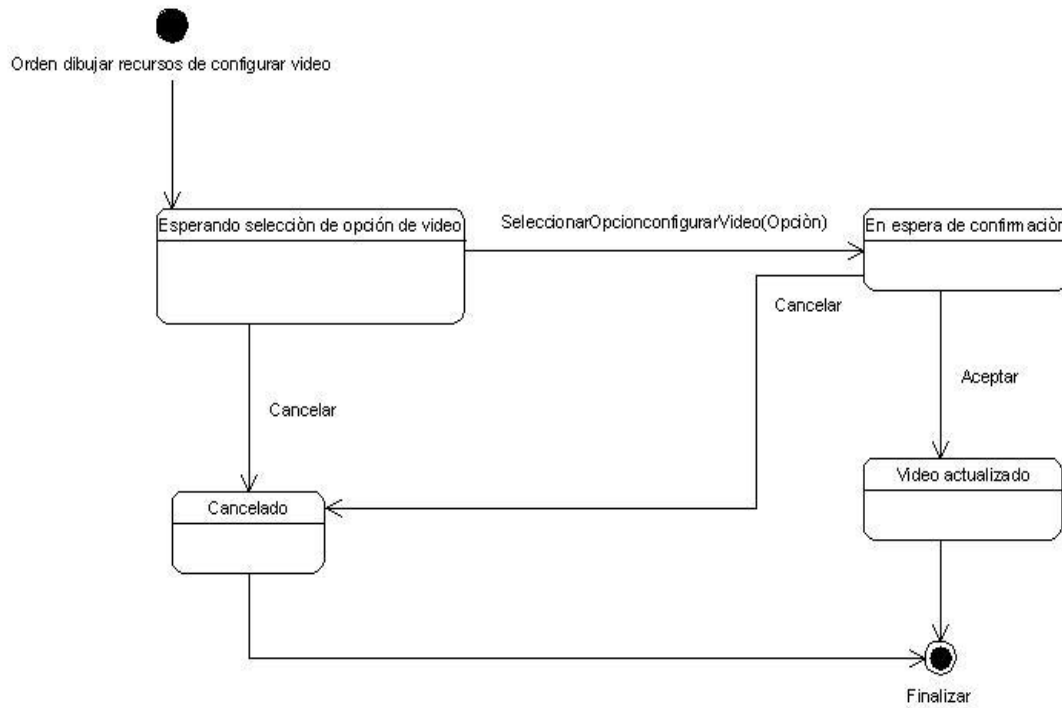


Figura 5.55: Diagrama de estado configurar video

### 5.2.4.10. Diagrama de estado Configurar velocidad

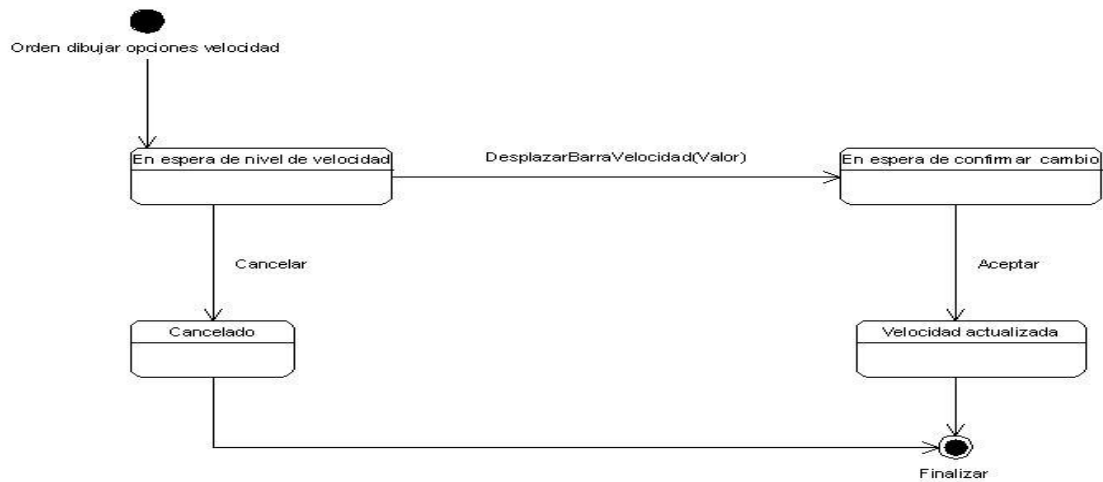


Figura.5.56: Diagrama de estado configurar velocidad



### 5.2.4.11. Diagrama de estado Gestionar sesión

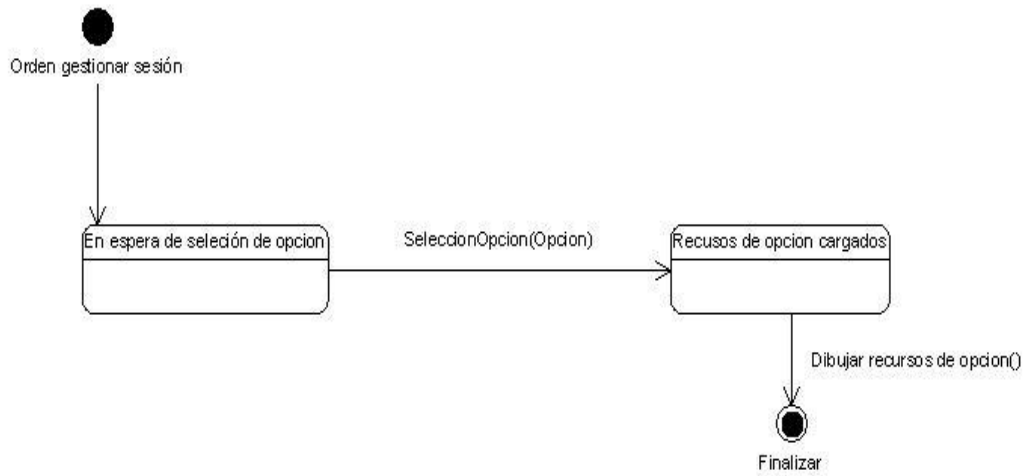


Figura 5.57: Diagrama de estado gestionar sesión

### 5.2.4.12. Diagrama de estado Crear Usuario

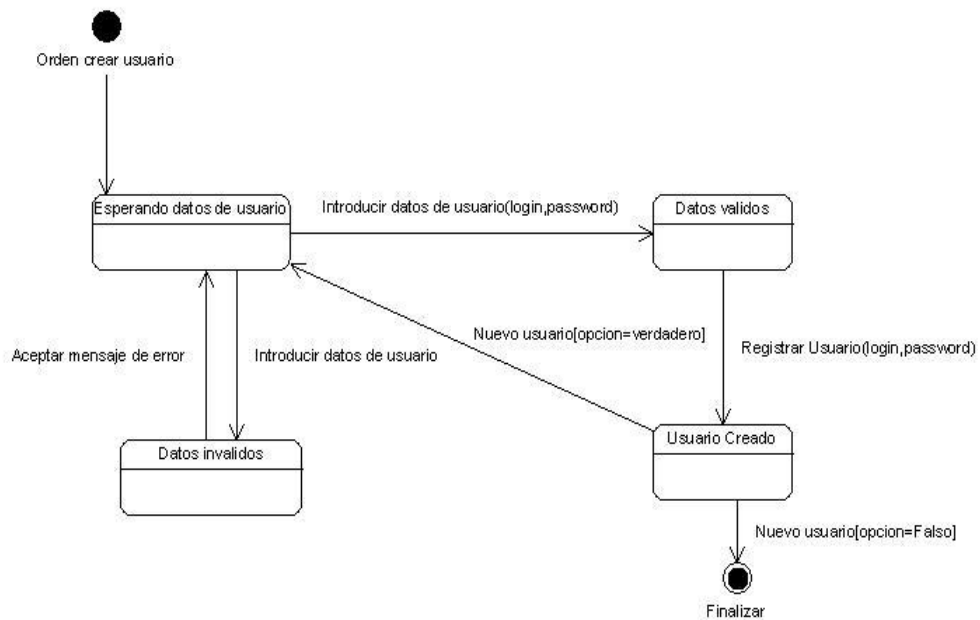


Figura 5.58: Diagrama de estado crear usuario



### 5.2.4.13. Diagrama de estado Eliminar usuario

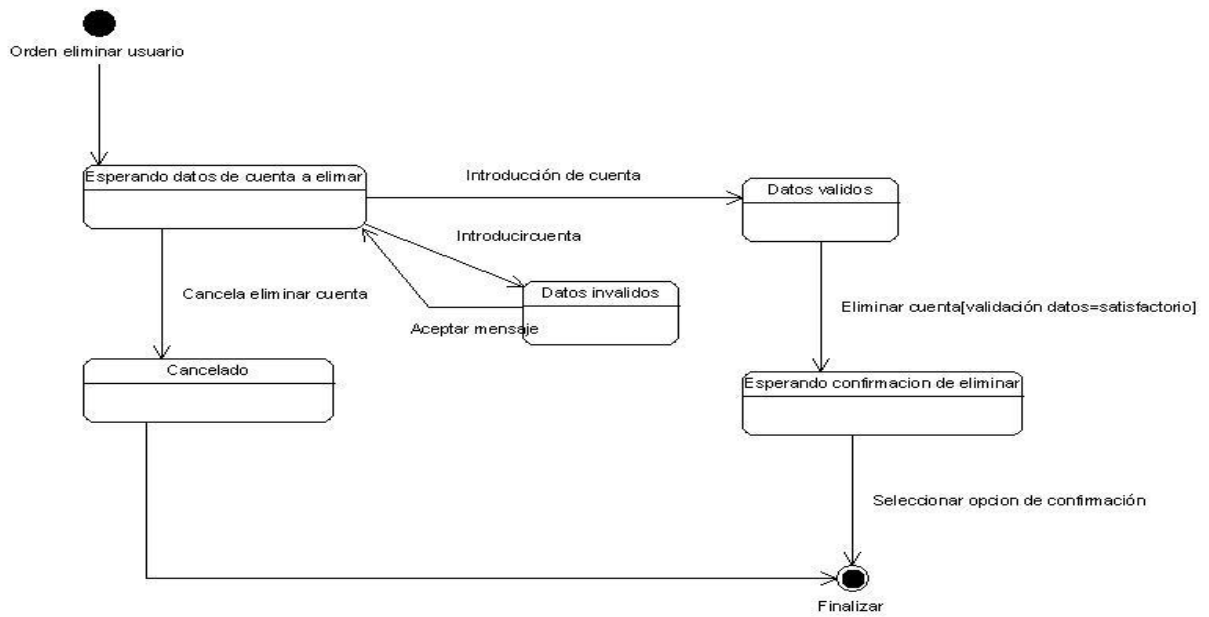


Figura 5.59: Diagrama de estado eliminar usuario

### 5.2.4.14. Diagrama de estado Salir

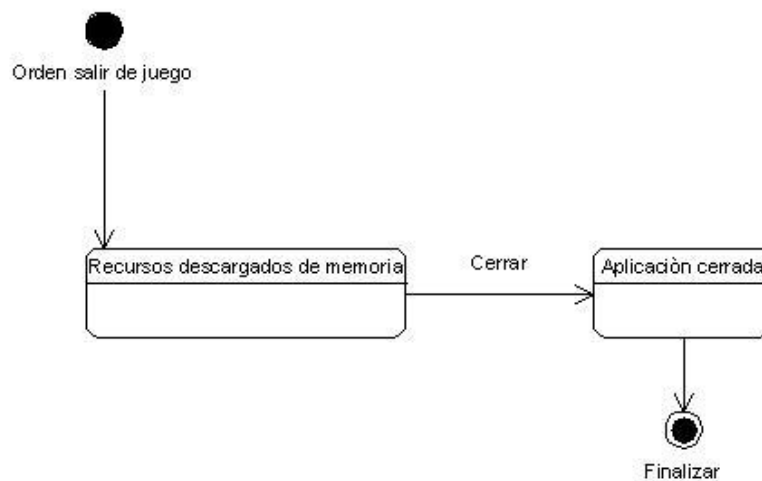


Figura 5.60: Diagrama de estado salir



### 5.2.4.15. Diagrama de estado Jugar

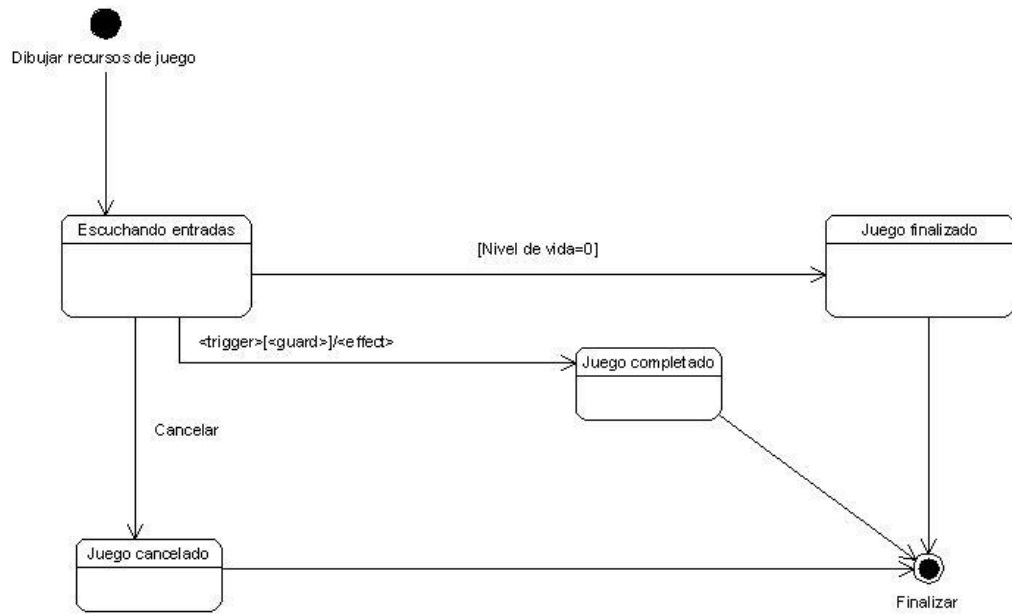


Figura 5.61: Diagrama de estado Jugar

### 5.2.4.16. Diagrama de estado SeleccionarCaracter

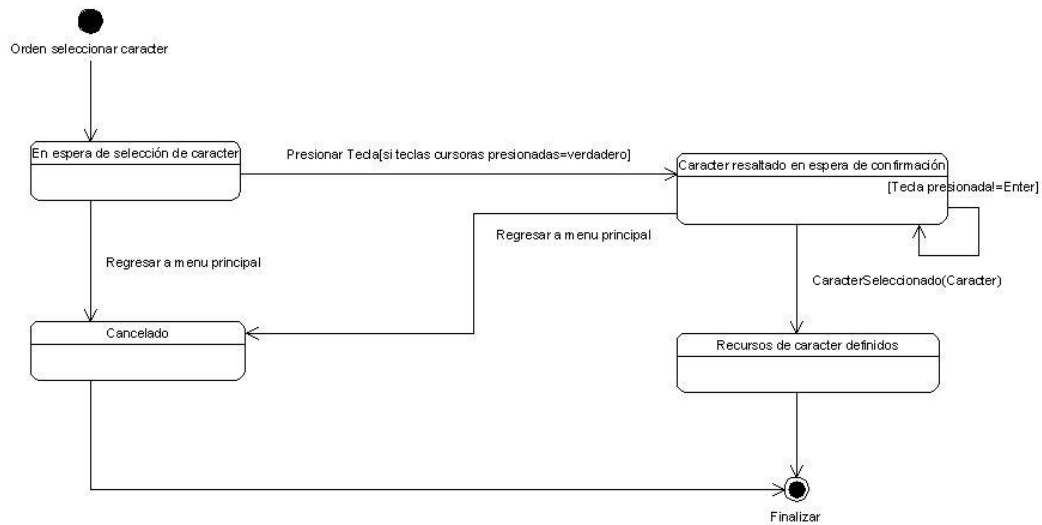


Figura 5.62: Diagrama de estado seleccionar carácter





### 5.2.4.17. Diagrama de estado Controlar carácter

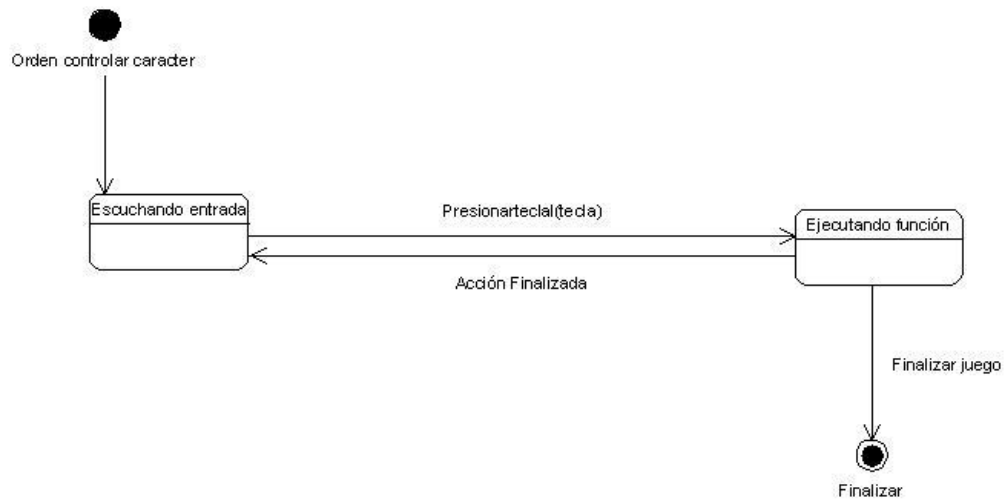


Figura 5.63: Diagrama de estado controlar carácter

### 5.2.4.18. Diagrama de estado Opciones rápidas

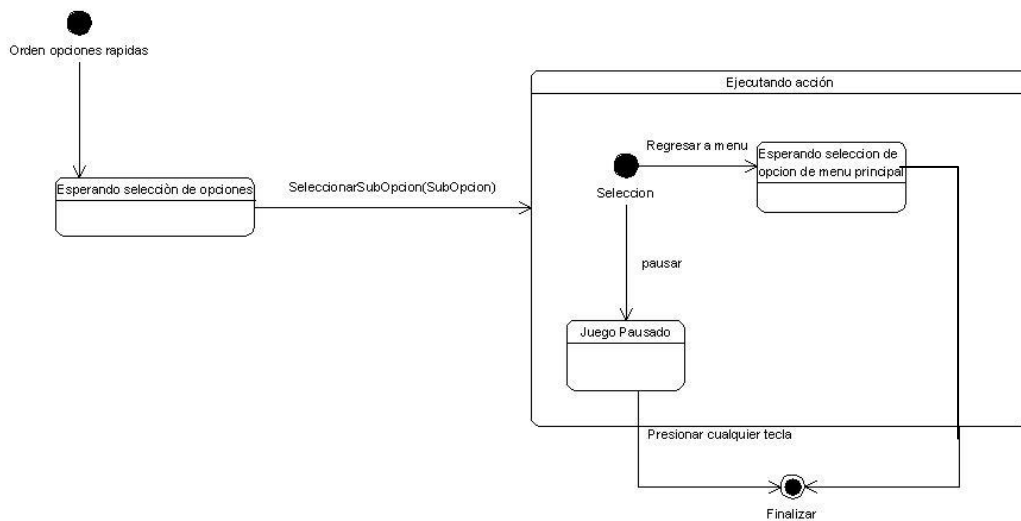


Figura 5.64: Diagrama de estado opciones rápidas



**5.2.4.19. Diagrama de estado Administrar objetos recolectados**

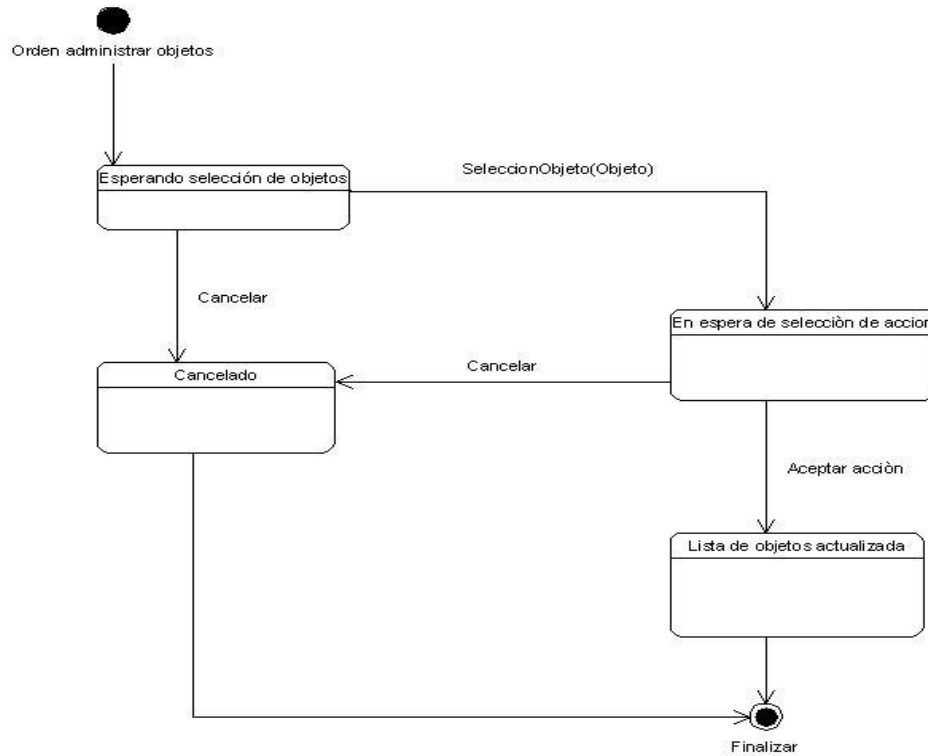


Figura 5.65: Diagrama de estado administrar objetos recolectados

**5.2.4.20. Diagrama de estado Guardar sesión**

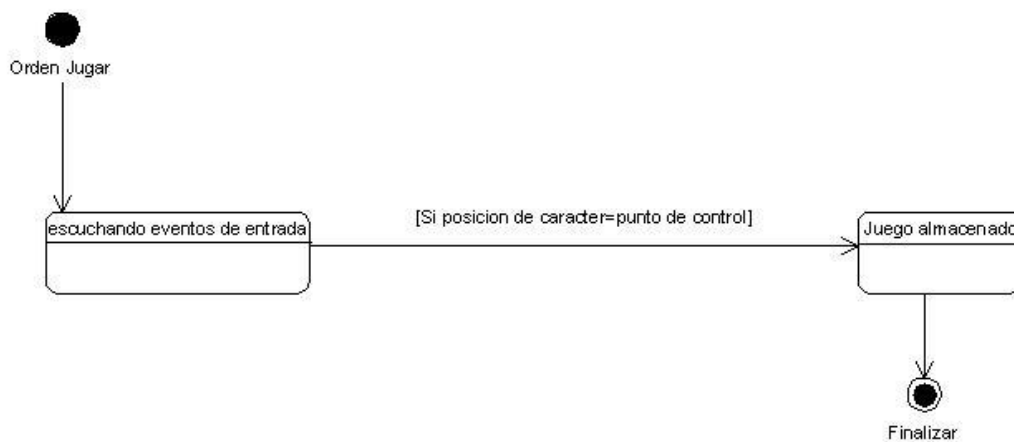


Figura .5.66: Diagrama de estado guardar sesión



### 5.2.5. Diagrama de Componentes

Un componente de Software es una parte física de un sistema y sus relaciones y se encuentra en la computadora, no en la mente del analista. Un componente puede ser por ejemplo: una tabla, un archivo de datos, un ejecutable, una biblioteca de vínculos dinámicos, documentos. Un componente puede ser la implementación de más de una clase.

El diagrama de componentes sirve para que:

- Los clientes puedan ver la estructura del sistema finalizado,
- Los desarrolladores cuenten con una estructura con la cual trabajar en adelante.
- Quienes escriban las notas técnicas y la documentación puedan entender de qué escribirán.
- Que el desarrollador reutilice los componentes.

Un componente debe tener un nombre. Un componente puede contener adornos, valores etiquetados e información adicional. Ej: referencias a las interfaces que realiza.

Un componente posee características similares a una clase: tiene nombre, realiza interfaces, puede participar de relaciones, puede participar en interacciones.

Un componente representa un elemento físico (bits). Las operaciones de un componente solo se alcanzan a través de interfaces. Una interfaz contiene una colección de operaciones y se utiliza para especificar los servicios de una clase o de un componente.

Una interfaz se conecta al componente que la implementa a través de una relación de realización, y al componente que utiliza sus servicios con una dependencia.

Las relaciones de dependencia se utilizan en los diagramas de componentes para indicar que los servicios ofrecidos por otro componente.

Un diagrama de componentes representa las dependencias entre componentes software, incluyendo componentes de código fuente, componentes del código binario, y componentes ejecutables. Un módulo de software se puede representar como componente. Algunos componentes existen en tiempo de compilación, algunos en tiempo de enlace y algunos en tiempo de ejecución, otros en varias de éstas.

Un componente de sólo compilación es aquel que es significativo únicamente en tiempo de compilación del sistema. Un componente ejecutable es un programa ejecutable que es parte o funciona como el sistema mismo.

El diagrama de componente hace parte de la vista física de un sistema, la cual modela la estructura de implementación de la aplicación por sí misma, su organización en componentes. La vista de implementación se representa con los diagramas de componentes.



Los componentes tienen dos características: Empaquetan el código que implementa la funcionalidad de un sistema, y algunas de sus propias instancias de objetos que constituyen el estado del sistema. Los llamados últimos componentes de la identidad, porque sus instancias poseen identidad y estado.

### **Código:**

Un componente contiene el código para las clases de implementación y otros elementos. Un componente de código fuente es un paquete para el código fuente de las clases de implementación. Algunos lenguajes de programación distinguen archivos de declaración de los archivos de método, pero todos son componentes. Un componente de código binario es un paquete para el código compilado. Una biblioteca del código binario es un componente.

Cada tipo de componente contiene el código para las clases de implementación que realizan algunas clases e interfaces lógicas. La relación de realización asocia un componente con las clases y las interfaces lógicas que implementan sus clases de implementación. Las interfaces de un componente describen la funcionalidad que aporta. Cada operación de la interfaz debe hacer referencia eventualmente a un elemento de la implementación disponible en el componente.

La estructura estática, ejecutable de una implementación de un sistema se puede representar como un conjunto interconectado de componentes. Las dependencias entre componentes significan que los elementos de la implementación en un componente requieren los servicios de los elementos de implementación en otros componentes. Tal uso requiere que dichos elementos sean de visibilidad pública.

### **Identidad:**

Un componente de identidad tiene identidad y estado. Posee los objetos físicos que están situados en él. Puede tener atributos, relaciones de composición con los objetos poseídos, y asociaciones con otros componentes. Desde este punto de vista es una clase. Sin embargo la totalidad de su estado debe hacer referencia a las instancias que contiene.

### **Estructura:**

Un componente ofrece un conjunto de elementos de implementación, esto significa que el componente proporciona el código para los elementos. Un componente puede tener operaciones e interfaces. Un componente de identidad es un contenedor físico para las entidades físicas como bases de datos. Para proporcionar manejadores para sus elementos



contenidos, puede tener atributos y asociaciones salientes, que deben ser implementadas por sus elementos de implementación. Este componente se representa con un rectángulo con dos rectángulos más pequeños que sobresalen en su lado izquierdo.

Las operaciones e interfaces disponibles para los objetos exteriores se pueden representar directamente en el símbolo de clase. Estos son su comportamiento como clase. Los contenidos del subsistema se representan en un diagrama separado.

Las dependencias de un componente con otros componentes o elementos del modelo se representan usando líneas discontinuas con la punta de flecha hacia los elementos del proveedor. Si un componente es la realización de una interfaz, se representa con un círculo unido al símbolo del componente por un segmento de línea.

UML define cinco estereotipos estándar que se aplican a los componentes:

- **Executable:** Especifica un componente que se puede ejecutar en un nodo.
- **Library:** Especifica una biblioteca de objetos estática o dinámica.
- **Table:** Especifica un componente que representa una tabla de una base de datos.
- **File:** Especifica un componente que representa un documento que contiene código fuente o datos.
- **Document:** Especifica un componente que representa un documento.

**Los elementos que componen a un diagrama de componente son:** componentes, interfaces, relaciones de dependencia, generalización, asociación y realización.

### **Cómo elaborar un Diagrama de Componentes:**

1. Distribuir las clases del diseño en los componentes que formarán el producto de software
2. Agrupar los componentes relacionados en paquetes
3. Identificar las interfaces y asignarlas a cada componente que debería utilizarlas
4. Asociar las clases que utilizan las interfaces para manipular los componentes.

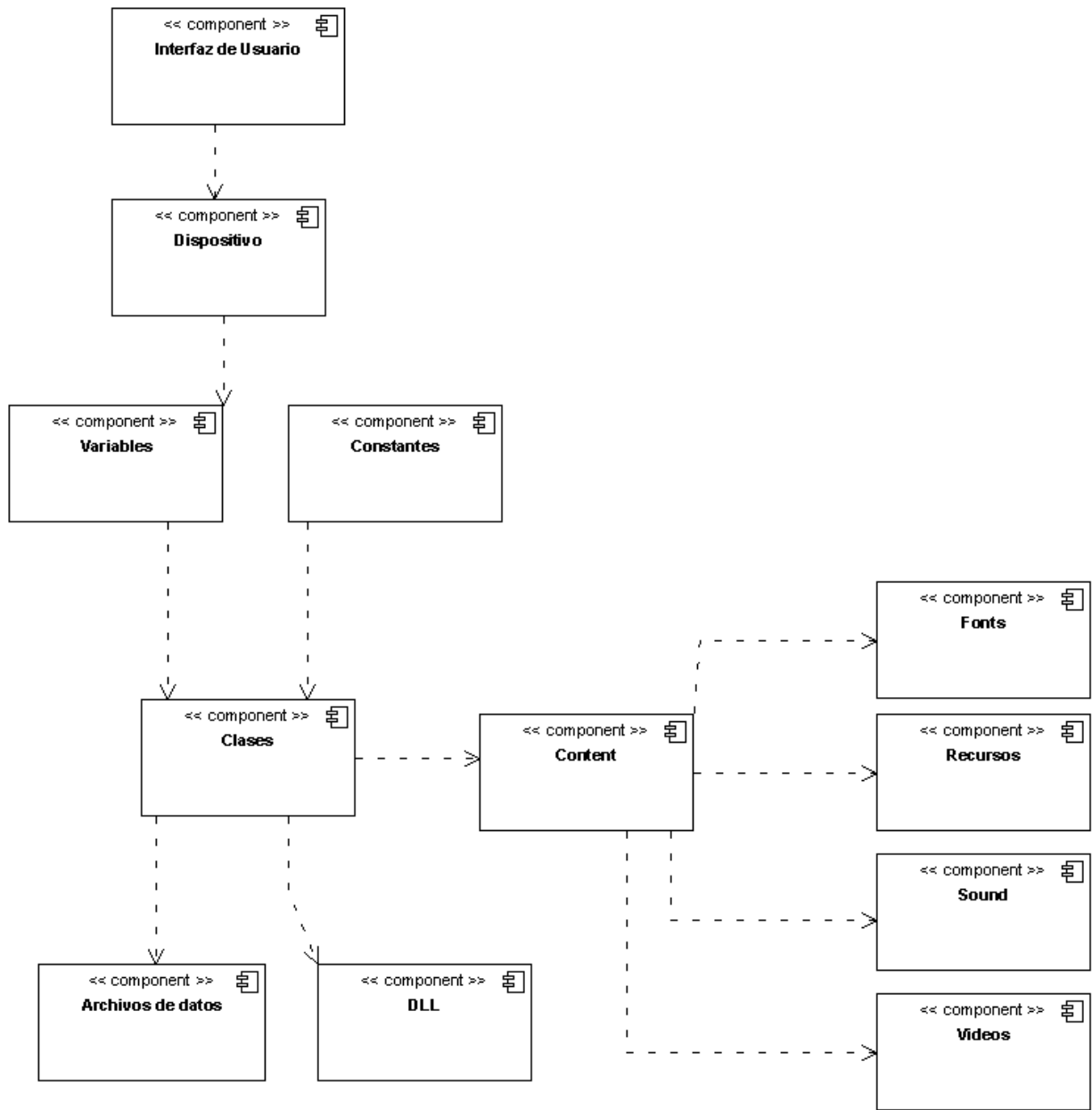


Figura 5.67: Diagrama de componentes del sistema



## 5.2.6. Diagrama de Clases y Paquetes

Uno de los elementos importantes del diseño UML, son los diagramas de clases, que permiten al desarrollador utilizarlos como “plano arquitectónico” para la codificación de la solución. Al mismo tiempo, la encapsulación de estas clases permite un mejor panorama global de la aplicación solución, aportando en gran manera a la modulación de la misma. En esta sección se describirán de forma gráfica tales elementos, comenzando por el diagrama de clases del diseño de la solución.

### 5.2.6.1. Diagrama de Clases del Diseño

Un diagrama de Clases representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas.

Los diagramas de Clases por definición son estáticos, esto es, representan que partes interactúan entre sí, no lo que ocurre cuando<sup>109</sup>.

Para crear y desarrollar un diagrama de clases de diseño, se necesita modelar iterativamente:<sup>110</sup>

- Clases
- Responsabilidades
- Asociaciones
- Relaciones de Herencia
- Asociaciones de Composición

#### Clases

Cada clase, con su nombre, está representada por *atributos* y *métodos*.

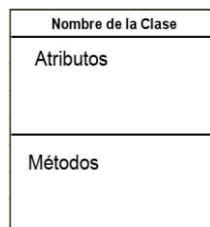


Figura 5.68: Clase UML 2.0

<sup>109</sup> <http://www.osmosislatina.com/lenguajes/uml/compdeploy.htm>

<sup>110</sup> The Object Primer 3rd. Edition, Agile Model Driven Development with UML 2. Ágil Desarrollo a través de Modelos con UML 2 por Scott W. Ambler. Cambridge University Press 2004. Cap. 12.



### Responsabilidades

Las responsabilidades de una clase son sus atributos y métodos.

### Asociaciones

Las asociaciones entre clases son expresadas de la siguiente forma:

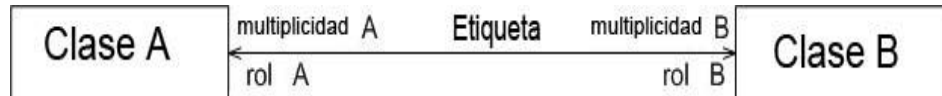


Figura 5.69: Asociación UML 2.0

**A** esta asociada con **B** y viceversa. Al igual que **A**, **B** tiene un rol. La línea con flechas simples en los extremos denota la asociación entre las dos clases. En cada extremo se especifica la multiplicidad de la asociación para **A** y **B**, así mismo, cada asociación tiene una etiqueta que es el nombre de esta.

### Relaciones de Herencia

Las relaciones de herencia son representadas por una flecha rellena, que apunta a la clase padre.

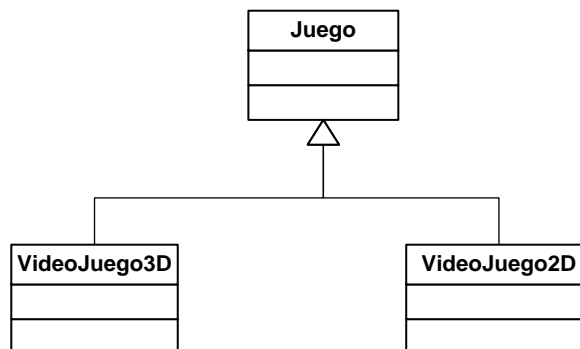


Figura 5.70: Relaciones de Herencia UML 2.0

En este ejemplo, VideoJuego3D y VideoJuego2D, heredan los atributos de Juego (Clase XNA Game)





### Asociaciones de Composición

El diagrama anterior muestra a ModeloEstatico que está compuesto por Modelo3D. Este puede estar compuesto al mismo tiempo por varios Modelo3Ds.

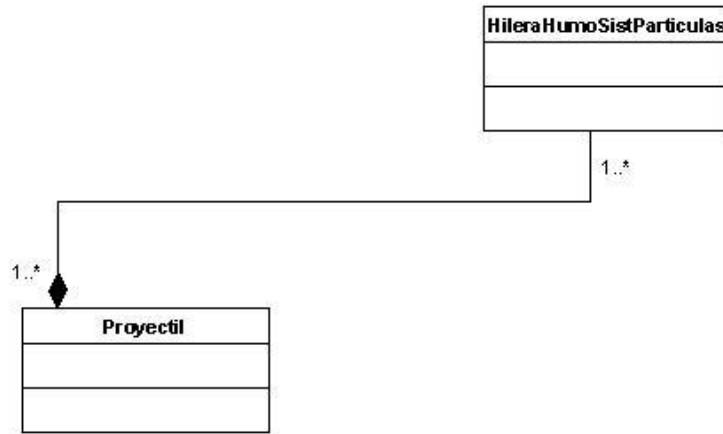


Figura 5.71: Asociaciones de Composición UML 2.0

#### **5.2.6.2. Diagrama de Clases de Diseño**

A continuación se presenta el diagrama de clases de diseño que componen al videojuego. Por razones de espacio, se muestran las clases puras sin atributos ni métodos, para identificar fácilmente las asociaciones entre estas. Seguido de los diagramas de clases de diseño globales, se mostrará cada clase individual junto con sus atributos y métodos.





DIAGRAMA DE CLASES DE DISEÑO PARTE II. (CONTENIDO DE EfectosDeParticulas)

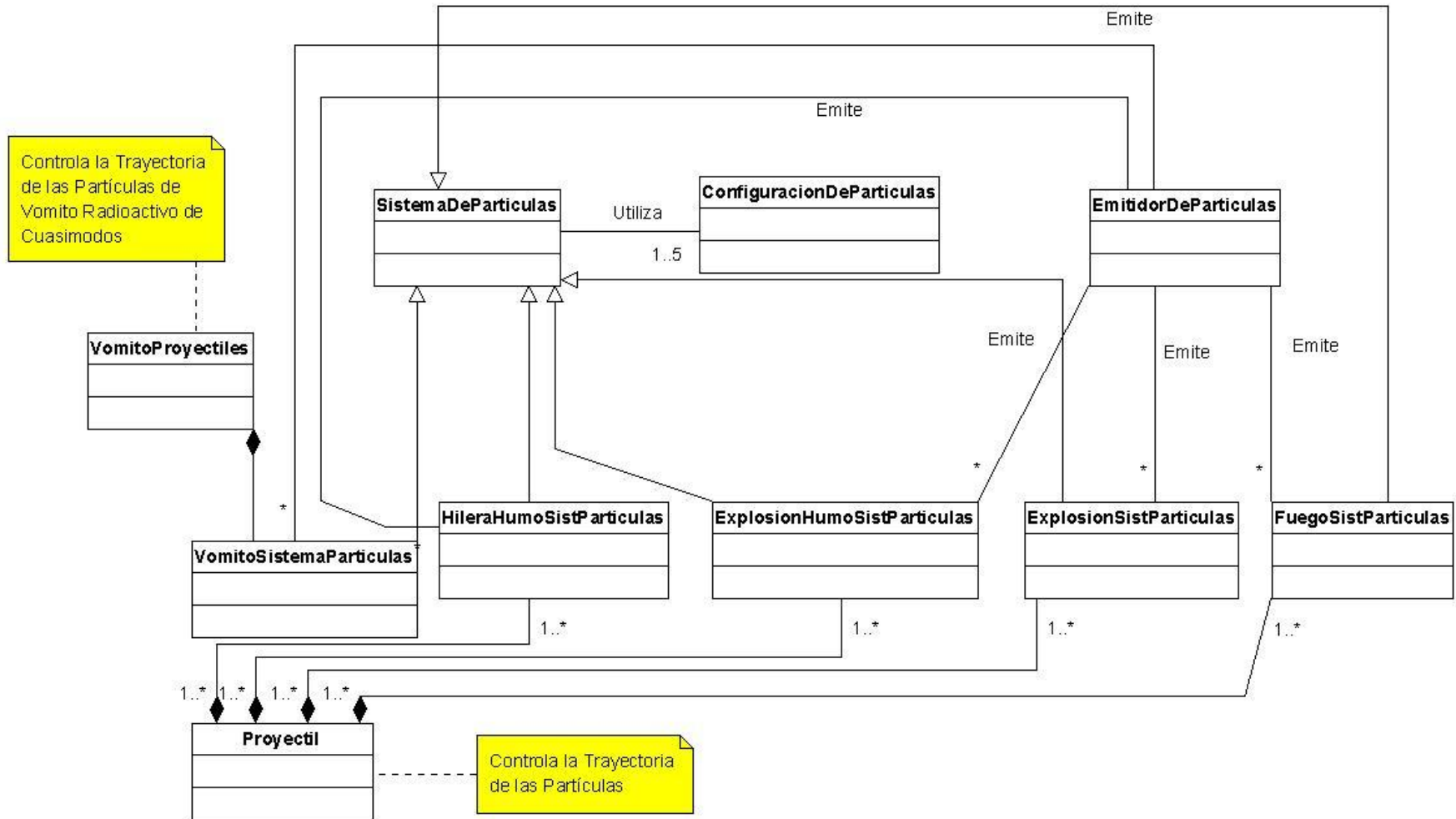


Figura 5.73 : Diagrama de clases de SIEMPOSAL



# **CAPITULO VI: PROGRAMACIÓN Y PRUEBAS DEL SOFTWARE**



## **6. DESARROLLO DE SIEMPOSAL**

En este apartado se describen las sub-etapas de desarrollo de SIEMPOSAL, se detallan los elementos indispensables para creación de texturas, modelos, animación, programación, así como también las condiciones tecnológicas necesarias, la ubicación física que tendrán los datos en el software y la metodología que se utilizara para el texturizado, modelado y animación y la integración de modelos en la programación y posteriormente las pruebas del mismo.

### **6.1. SUB-ETAPAS DE DESARROLLO DE SIEMPOSAL**

La etapa de desarrollo de un videojuego consta de varias sub-etapas las cuales se describen a continuación.

#### **6.1.1. Toma de fotografía.**

Cuando lo que se desea modelar son zonas reales y específicas de un país, las fotografías son la materia prima para la creación de texturas, pues basadas en estas, el modelador puede tener una visión clara de la situación real que se desea simular, para luego crear las texturas y modelos. Para el modelado de las escenas mostradas en el videojuego, fue necesario realizar la toma de fotografías en los escenarios siguientes:

- El centro capitalino (alrededor de catedral)
- EL Salvador del Mundo y sus alrededores
- Oficinas del Ministerio del Medio Ambiente
- Parte de galería (calle principal)
- Parque Montecristo
- Rio Lempa

Después de la toma de fotografías, se prosiguió a seleccionar las fotografías que más se apegaran a las necesidades del modelador.



### 6.1.2. Creación de Texturas.

Una vez seleccionadas las fotografías, se inició el proceso de texturizado. En este proceso el modelador crea una imagen mental del objeto 3D a construir, así como también la forma de texturizar, tomando en cuenta las siguientes situaciones.

- Textura repetitiva en el modelo
- Textura con relieves específicos o coloniales
- Variedad de colores que tenga el modelo a crear
- Geometría de lo que se va a texturizar

Luego se prosiguió a la creación de textura en el programa seleccionado (Adobe Photoshop), para ello se editaron las fotografías. En la edición de fotografías se estableció la rotación, escala, distorsión, deformación, perspectiva, brillo, contrastes, saturación, etc., necesarias para obtener al final texturas que cumplieron con las características siguientes:

- Las dimensiones de textura en potencias de 2
- La resolución de archivos \*.jpg reguladas de acuerdo a las necesidades del modelo a crear.
- En base al modelo que se desea crear las extensión pueden ser: .jpg, .png, .tga y .bmp.

Dependiendo del modelo a crear, esta sub-etapa al igual que el modelado y animación pueden consumir mucho tiempo, debido a las propiedades de lo que se desee crear.

### 6.1.3. Modelado y Animación.

Esta sub-etapa consiste de la creación, texturización y animación de modelos 3D.

Teniendo las texturas, se inició el modelado 3D en el programa establecido (fragMOTION). En esta sub-etapa, a partir de la creación de cubos, esferas, cilindros, conos, etc., se formaron los modelos estáticos y animados que se necesitaron.

Los modelos 3D creados fueron personas, animales o cosas, de los cuales los más complejos de crear son las personas.

Se realizó la texturización de los modelos, mediante la asignación de materiales a estos y el remapeo de las coordenadas.

En cuanto a la animación, se crearon las animaciones necesarias para los modelos, como por ejemplo, hablar, caminar, volar, etc., para ello, se crearon los esqueletos y se asignó grupo de vértices a cada hueso.



En esta sub-etapa se consideró algo muy importante que es la reducción de vértices innecesarios, caras duplicadas y vértices huérfanos que no se encuentren asignados, así como también la reducción de vértices que compartan la misma coordenadas en el espacio 3D. Todo esto se hizo tomando en consideración los equipos en los que el software ha de ejecutarse.

Con las aplicaciones anteriormente especificadas para el modelado y animación, se convirtieron algunos modelos prefabricados a formatos soportados por Microsoft XNA Game Studio 2.0, para poder incorporarlos al videojuego.

Esta sub-etapa, dependiendo de la complejidad de lo que se desea crear es la que más tiempo consumió.

Para finalizar esta sub-etapa, se han exportado todos los modelos al formatos de archivos \*.x los cuales son soportado por Microsoft XNA Game Studio 2.0.

#### **6.1.4. Codificación e Integración de Modelos y Sonidos.**

##### **6.1.4.1. Codificación.**

Para la codificación de SIEMPOSAL se utilizo exclusivamente la programación orientada a Objetos, dividiendo al software en paquetes (conjuntos de clases y otros paquetes) necesarias, con sus atributos y métodos, basadas en el diseño presentado, para cumplir con los requerimientos del videojuego desarrollado.

Para realizar la división del software en paquetes y clases se utilizó el tipo de arquitectura Modelo Vista Controlador que separa las definiciones lógicas, objetos de frontera o interfaz y objetos de control.

A continuación se describen algunas de las técnicas utilizadas en la sub-etapa de codificación.

##### Objetos entidad, frontera y control

En el diseño del software se ha tenido la necesidad (por el tipo de modelo seleccionado) de utilizar tres tipos de objetos que son:

**Objetos de Entidad:** Representan la información persistente rastreada por el software, contienen las definiciones lógicas de las entidades que conforman el software, en ningún momento poseen la capacidad de presentarse ante el usuario.

**Objetos de Control:** Representan la interacción entre los actores y el software. Son los encargados de administrar las operaciones que el usuario realiza, se ubican entre los objetos de control y los objetos de frontera



**Objetos de Frontera:** Representan las tareas realizadas por el usuario y soportadas por el software. Corresponden a la interfaz que el usuario u otros software pueden acceder y jamás conocen el objeto de entidad que posee los datos que ellos presentan.

La interacción entre estos tres tipos de objetos es básica para comprender como es que SIEMPOSAL esta desarrollado.

Una utilización más compleja de esta técnica es utilizar una sola frontera para editar múltiples entidades.

Otra ventaja de lograr separar la Definición, funcionalidad y forma de presentación es la reutilización de objetos.

#### Herencia.

La capacidad de hacer generalizaciones y luego especializaciones de ésta, es una técnica muy útil de la programación orientada a objetos, ya que permite una gran reutilización de código.

Esta técnica es ampliamente utilizada en SIEMPOSAL ya sea tanto para los objetos de Entidad, Control y Frontera.

#### Polimorfismo.

Es la técnica más poderosa de la programación orientada a objetos, puede ser implementada a través de herencia o interfaces y permite que objetos de clases específicas sean tratados como objetos de la clase genérica.

#### Encapsulamiento.

Esta técnica nos permite dar acceso a los métodos de otras clases solo si es requerido, con esto se logra proteger, englobar a los datos y la funcionalidad de dichos datos en una estructura que permite tenerlos asociados

La figura 6.1 presenta la distribución de paquetes de clases para cada componente del modelo MVC:



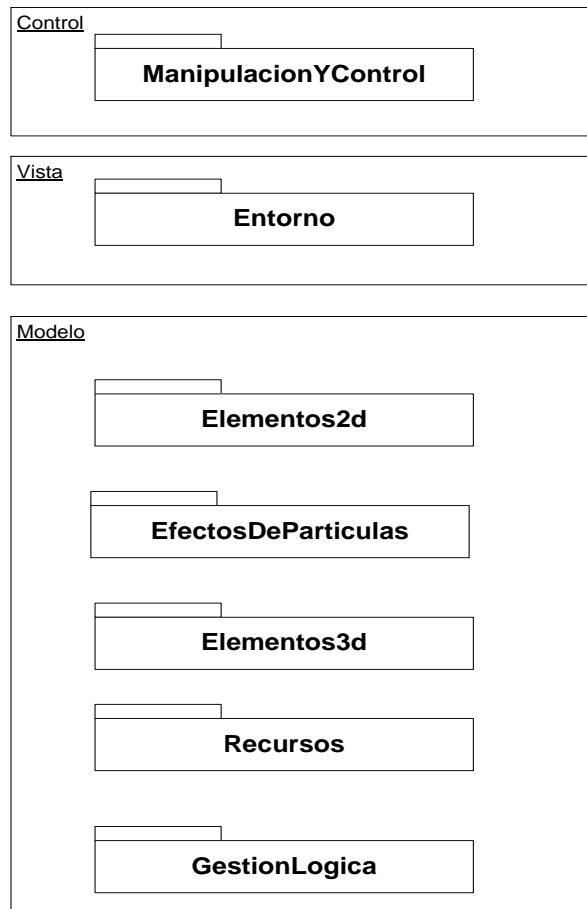


Figura 6.1: Distribución de paquetes en cada componente del modelo MVC

#### 6.1.4.2. Integración de modelos en el código

Para la integración de los modelos al código se realizó mediante la carga, dibujado, actualización y descarga de todo el contenido gráfico, asignándole los valores correspondientes desde las propiedades del procesador de contenido.

#### 6.1.4.3. Integración Sonidos en el código

Para la integración de los sonidos se ha utilizado la aplicación XACT que viene incluida en las bibliotecas de XNA y Directx. Con esta herramienta se centralizaron todos los recursos de sonidos en una biblioteca de sonidos que al compilarlo se convierte en una biblioteca binaria donde cada sonido es accedido, reproducido y manipulado a través de una clase tipo Sound. La biblioteca Sound de XNA funciona como un enlace entre la solución y la biblioteca mencionada para la edición de sonidos.



## 6.2. AMBITO DE DESARROLLO

En este apartado se detalla los elementos indispensables para la programación, tales como las condiciones tecnológicas necesarias, la ubicación física que tendrán los datos en el software y posteriormente las pruebas del mismo.

### 6.2.1. Condiciones tecnológicas

#### 6.2.1.1. Hardware Utilizado.

Para el desarrollo de SIEMPOSAL se utilizó el siguiente hardware con las características que a continuación se detallan:

| Recurso            | Cantidad | Características   |
|--------------------|----------|---|
| Cámara Fotográfica | 2        | Samsung de 7.2 Megapixeles  |
| Computador         | 1        | <b>Procesador:</b> AMD Athlon 64 X2 Dual Core Processor 4800+2.51 GHz<br><b>Memoria RAM:</b> 1GB<br><b>Disco duro:</b> 300GB<br><b>Monitor:</b> Dell E176FP Black 17" LCD,1280x1024 pixels<br><b>Video:</b> ATI Radeon 155x Series de 250 MB de video RAM, para shadders <sup>111</sup> de efectos 3.0<br><b>Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps |
| Computador         | 1        | <b>Procesador:</b> Intel Celeron 3.2 GHz<br><b>Memoria RAM:</b> 1GB<br><b>Disco duro:</b> 160 GB<br><b>Video:</b> GeForce 7800 GTX 256<br><b>Monitor:</b> Monitor CRT Compaq MV7540e de 17 pulgadas, 1280 x 1024 píxeles<br><b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps   |
| Computador         | 1        | <b>Procesador:</b> Intel Core Duo T2300 (1660 MHz)<br><b>Memoria RAM:</b> 1GB<br><b>Disco duro:</b> 80GB<br><b>Monitor:</b> Dell 14.1 inch 16:10 ,1440 x 900 pixels<br><b>Video:</b> Intel Graphics Media Accelerator 950 soporte pixel Shader 3.0<br><b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps                                     |

<sup>111</sup> Shader: Sombreador, termino entre desarrolladores de sistemas gráficos computaciones que hace referencia a un archivo de extensión .fx (de efectos) en donde se escribe código (hibrido entre Java-C# y C++). Este archivo es asociado con distintos modelos 3D que se implementan en el juego para aplicar efectos que dependerán de las capacidades de cada tarjeta de video. Si la tarjeta de video de la maquina cliente no soporta dicho efecto, simplemente no se muestra, permitiendo así la ejecución continua sin excepciones del software.



|                  |   |  |
|------------------|---|--|
| Computador       | 1 | <p><b>Procesador:</b> AMD Turion 64 ML-34 y 1MB de caché L2 (1.8 GHZ)</p> <p><b>Memoria RAM:</b> 512 MB de memoria de sistema DDR, a 333 MHZ</p> <p><b>Disco duro:</b> 80 GB (4.200 rpm)</p> <p><b>Monitor:</b> HP WXGA BrightView de alta definición de 15,4 pulgadas, 1280 x 800 píxeles</p> <p><b>Video:</b> ATI RADEON XPRESS 200M IGP con 128 MB de memoria DDR (dedicada)</p> <p><b>Interfaz de Red:</b> Tarjeta de red Ethernet con conector Rj-45, velocidad 10/100 Mbps</p> |
| Control Xbox 360 | 1 | <p><b>Marca:</b> MADCATZ</p> <p><b>Conector:</b> USB,</p> <p><b>Palancas:</b> 2</p> <p><b>Gatillos:</b> 2</p> <p><b>Botones:</b> 6,<br/>Botón dedicado de encendido y apagado</p>  |

Cuadro 6.1 Equipo de Hardware.

### 6.2.1.2. Software Utilizado.

Para el desarrollo de SIEMPOSAL se utilizó el software que se detalla a continuación:

| Software  | Descripción  |
|---|--|
| Windows XP SP 2   | Sistema Operativo del cual se obtiene librerías del FrameWork  |
| Microsoft Visual C# Express Edition                         | Lenguaje de programación orientado a objetos   |
| Microsoft Visual C# 2005 Express Edition ENU Service Pack 1 | Actualización del Lenguaje de programación   |
| Directx   | Set de bibliotecas que contiene los atributos y métodos para desarrollar videojuegos, inicializar modos gráficos, realizar copiado de datos de la memoria del computador a la tarjeta de video, etc. |
| Microsoft XNA Game Studio 2.0                               | Librerías basadas en Directx que permiten desarrollar con más facilidad videojuegos en C# para las plataformas Windows XP, Windows Vista y Xbox 360  |
| fragMOTION  | Herramienta utilizada para el modelado y animación de objetos 3D   |
| MilkShape   | Herramienta utilizada para conversión de formatos de modelos 3D y reducción de vértices.   |
| Autodesk 3ds Max  | Herramienta utilizada para conversión de formatos de modelos 3D  |
| Adobe Photoshop CS2   | Utilizado para la creación de texturas para los modelos 3D, mapas de relieve, mapa de terrenos en escalas de grises y filtros de mapa normal.  |
| NVIDIA Photoshop Plugins                                    | Permite utilizar dentro de Adobe Photoshop el filtro de relieve que provee la empresa NVIDIA.  |
| Creative Lab Media Studio                                   | Se utiliza para editar los sonidos y almacenar los archivos en formato .wav  |
| XACT  | Utilizado para centralizar todos los recursos de sonidos.  |
| Microsoft Office 2003                                       | Herramienta utilizada para la elaboración de la documentación de software  |
| Microsoft Speech SDK 4.0                                    | Motor de texto a voz.  |
| L&H TTS 3000 Español  | Utilizada para implementar las bibliotecas de texto a voz en español   |

Cuadro 6.2 Software utilizado en el desarrollo de SIEMPOSAL



### 6.2.2. Configuración del entorno de desarrollo

Para el desarrollo de SIEMPOSAL se configuró el entorno de trabajo de la siguiente manera:

Considerando como punto inicial, que ya se tenía el equipo con el Sistema Operativo Windows XP SP2, se instalaron los siguientes software de desarrollo que a continuación se detallan.

| Utilización  | Software   |
|--|--|
| Programación   | a) Microsoft Visual C# 2005 Express Edition<br>b) Microsoft Visual C# 2005 Express Edition-ENU Service Pack 1<br>c) Runtime Directx 9.X y opcionalmente el SDK de Directx.<br>d) Microsoft XNA Game Studio 2.0<br>e) Microsoft Speech SDK 4.0<br>f) L&H TTS 3000 Español<br><br>* Estas aplicaciones se encuentran en el CD que ha sido anexado al final de este documento.<br>* La instalación de las aplicaciones a), b), c) y d) debe hacerse en el orden descrito.<br>* La instalación de las aplicaciones e) y f) no tiene un orden específico. |
| Modelado y animación   | a) fragMOTION<br>b) Milkshape<br>c) Autodesk 3ds Max   |
| Texturizado de modelos 3D y mapas de relieve, mapas de terreno | a) Adobe Photoshop CS2<br>b) NVIDIA Photoshop Plug-ins   |

Cuadro 6.3 Software utilizado en el entorno de desarrollo de SEIMPOSAL

### 6.2.3. Metodología de trabajo del grupo de desarrollo

La creación de un software demanda un riguroso control del ambiente sobre el cual se desarrolla, para facilitar el trabajo en esta etapa es necesario que para la distribución, almacenamiento y manipulación de archivos se consideran los siguientes aspectos.

#### 6.2.3.1. Directorio de Trabajo

El almacenamiento de las distintas versiones de archivos de código fuente se realizaron por medio de una estructura jerárquica de directorios que contienen las diversas actualizaciones a clases y recursos<sup>112</sup> creados por cada desarrollador, así como los consolidados en cada versión del proyecto.

<sup>112</sup> Se refiere a archivos que se cargan en el Content de XNA como: videos, sonidos, texturas, fuentes y modelos 3D.



### 6.2.3.2. Control de versiones

Para el control de versiones del producto de software se creó una estructura de directorio que permitiera organizar las diferentes versiones del proyecto, de acuerdo a lo que se muestra en la figura 6.2.

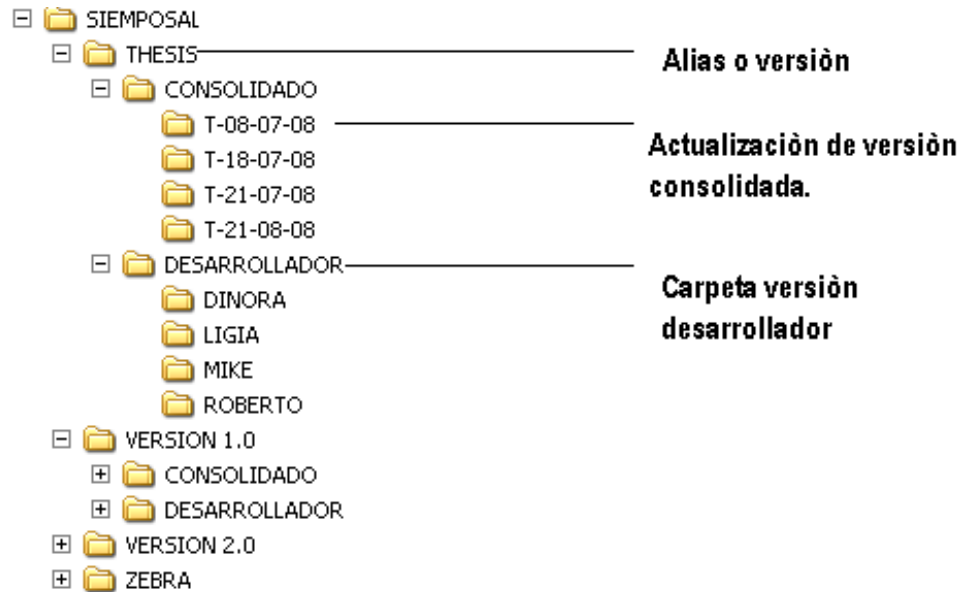


Figura 6.2: Estructura de directorios para el control de versiones del software.

Los lineamientos para mantener la estructura de directorios presentada son los siguientes:

El directorio raíz siempre tuvo como nombre las siglas del proyecto que se está desarrollando.

Se crearon bajo el directorio raíz carpetas de control de versiones, las cuales tenían un alias<sup>113</sup> o el número de versión.

Bajo el nivel del alias se ubicaron las carpetas consolidado y desarrollador que se utilizaron para controlar las actualizaciones de cada uno de los integrantes del grupo de desarrollo y de forma consolidada.

Las actualizaciones de los proyectos de forma consolidada se nombraron con la concatenación de la letra “T-“fecha de modificación.

Los recursos y código fuente creados por cada integrante del grupo se almacenaron en la carpeta con su respectivo nombre.

La metodología de control de versiones fue que una persona controlara de manera centralizada las versiones del producto. Es decir, que se creó en un único computador la

<sup>113</sup> Nombre código asignado al desarrollo de un software.



jerarquía de directorios planteada en la figura 6.2, esta se distribuyó por medio de dispositivos de almacenamiento y ftp.

En la carpeta DESARROLLADOR, cada integrante del grupo creó, almacenó y modificó los recursos y código fuente, para que posteriormente el encargado de integración de código, realizara la sincronización del mismo. Luego de terminado el proceso de integración éste procediera a publicar o distribuir la Versión consolidada a los diferentes desarrolladores.

Antes del proceso de integración se centralizó el código fuente y recursos de cada uno de los desarrolladores, en un entorno de red, mediante recursos compartidos, se puso a disposición de cada desarrollador una carpeta de trabajo personal, con la carpeta DESARROLLADOR, identificada con el nombre del desarrollador, por ejemplo: Dinora (Ver la siguiente figura 6.3) este recurso tenía permisos de lectura, escritura y modificación.

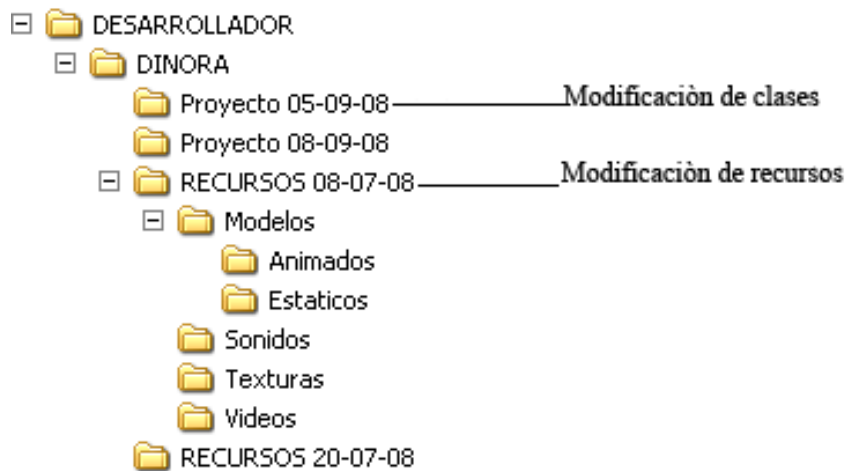


Figura 6.3: Estructura de directorios para versión desarrollador

En la fecha y hora establecida para la integración, se procedió a quitar el recurso compartido de la red, mientras se realizaba el proceso, las actualizaciones recibidas después de la fecha y hora fueron tomadas en cuenta para la siguiente versión.

Una vez el código fuente estaba integrado, el encargado del proceso debía enviar el proyecto integrado a cada desarrollador y crear dos copias de respaldo en el repositorio central.



### 6.2.4. Directorios y archivos

El cuadro 6.4 presenta la descripción de los archivos y carpetas de un proyecto base creado en XNA 2.0:

| Primer Nivel    | Segundo Nivel   | Tercer Nivel                                     | Descripción  |
|-----------------|-----------------|--|--|
| Content/        |                 |  | Esta carpeta contiene todos los recursos del videojuego como texturas, modelos .x, sonidos y videos.   |
|                 | Fonts/          | Archivos .font y XML de fuentes.                 | Carpeta predefinida para la inclusión de fuentes en un proyecto, esta es autogenerada por XNA.   |
|                 | Sound/          | Archivos .wav                                    | Dentro de esta carpeta se debe ubicar todo los archivos de audio a incluir en la solución.   |
|                 | Bin/            |  | Carpeta auto generada por XNA, contiene los archivos binarios de los recursos incluidos en el proyecto.  |
|                 | Obj/            |  | Directorio auto generada por XNA, almacena los archivos objetos de los recursos incluidos en el proyecto.  |
| Bin/            |                 |  | Contiene los archivos binarios de los recursos y clases incluidos en un proyecto XNA.  |
| DLL/            |                 | DLL's a incluir como referencias en el proyecto. | Carpeta que contiene las bibliotecas de procesos y manipulación de datos para ser agregadas en un proyecto mediante el "pipeline"                                |
| Obj/            |                 |  | Carpeta que almacena los archivos objetos de los recursos y clases incluidos en un proyecto XNA.   |
| Properties/     | Assemblyinfo.cs |  | Contiene información Assembly sobre el proyecto  |
| Proyecto.sln    |                 |  | Archivo que contiene la solución del proyecto la cual almacena la información de las librerías, clases, recursos y toda la información de un proyecto específico |
| VideoJuego3D.cs |                 |  | Archivo que contiene los métodos base para la creación de un videojuego y contiene el punto de entrada al videojuego.  |

Cuadro 6.4: Archivos y carpetas de un proyecto base creado en XNA 2.0

La figura 6.4 muestra el explorador de soluciones de Visual Studio Express Edition 2005 y la integración que este hace de los archivos y directorios expuestos en el cuadro 6.4

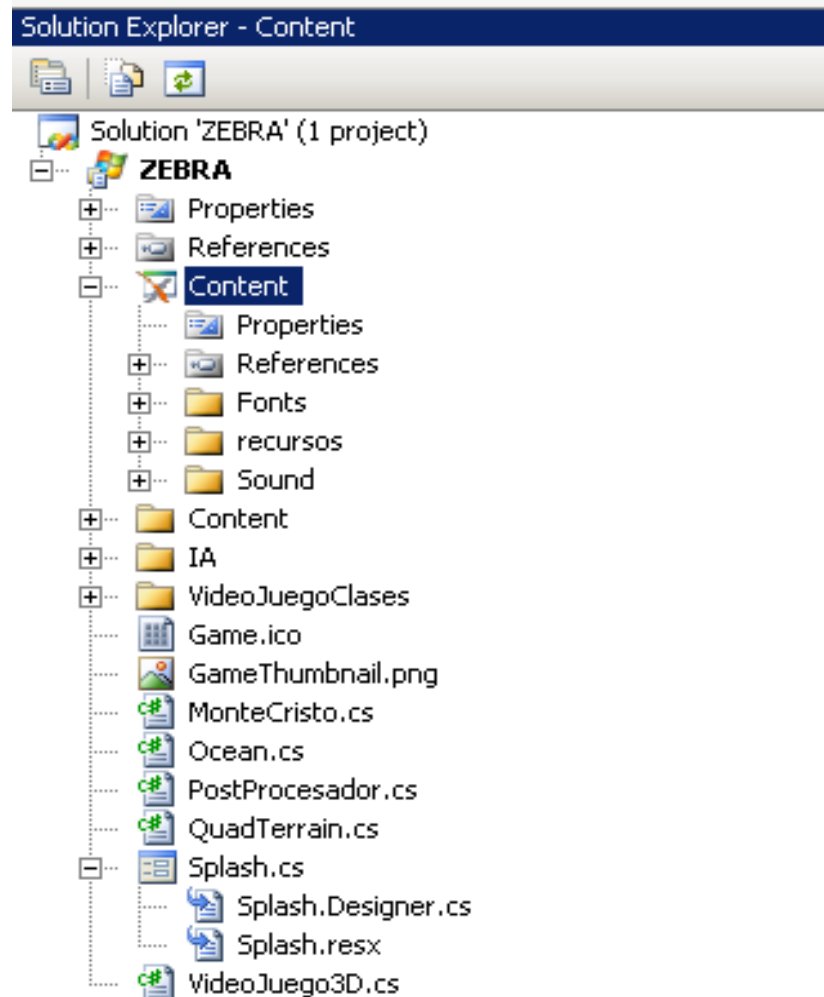


Figura 6.4: Estructura de la solución de un proyecto en XNA.

Debido a lo expuesto en el cuadro 6.4 y figura 6.4 la solución de un proyecto XNA limita a colocar todos los recursos del proyecto dentro de la carpeta Content, por lo que se ha creado una estructura de directorios como lo muestra la figura siguiente (figura 6.5).





Figura 6.5: Estructura de directorios para agregar recursos

La creación de clases se ha realizado dentro del directorio raíz del proyecto y en el mismo nivel de “Content”, para la creación de clases se han definido dos directorios “IA” que almacena la clase inteligencia artificial y VideojuegoClases; ambas estructuras se visualizan en la figura 6.6:

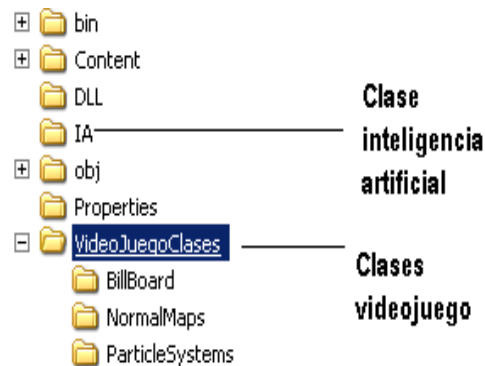


Figura 6.6: Estructura de directorios para la creación de clases



### 6.3. PRUEBAS DEL SOFTWARE.

En paralelo a la construcción de los componentes de la aplicación se realizó la revisión y validación de los mismos mediante pruebas, las cuales tiene una gran relevancia en el ciclo de vida del proyecto, siendo está la manera de llevar un control de la calidad de software.

A continuación se mencionan los métodos y estrategias de pruebas que se utilizaron para eliminar errores y garantizar una correcta funcionalidad de la aplicación.

#### 6.3.1. Generalidades de las pruebas realizadas

A continuación se describen los tipos de pruebas que se realizaron en cada uno de los módulos de SIEMPOSAL:

- **Estáticas**<sup>114</sup>: Son aquellas que no necesitan que el software que se está probando se esté ejecutando. Pueden hacerse manual o automáticamente, por ejemplo las revisiones de código pueden efectuarse manualmente, leyendo línea a línea o con ayuda de herramientas.
- **Dinámicas**<sup>115</sup>: Se requiere que el caso de prueba sea ejecutado sobre el ordenador para verificar los valores límites asociados a los datos de entrada y salida asociados al mismo.

Las técnicas aplicadas para realizar las pruebas fueron:

- **Pruebas de caja blanca**: Para la ejecución y verificación de las decisiones lógicas de los módulos y la navegabilidad de los mismos.
- **Pruebas de caja negra**: Para determinar la funcionalidad por módulos y de esa forma se identificaron errores de interfaz, de rendimiento y de iniciación.

La estrategia seguida a la hora de evaluar dinámicamente a SIEMPOSAL permitió comenzar por los componentes más simples y más pequeños e ir avanzando progresivamente hasta probar todo el software en su conjunto. Concretamente, los pasos a seguir son:

- **Pruebas Unitarias**: Comienzan con la prueba de cada módulo.
- **Pruebas de Integración**: A partir del esquema del diseño, los módulos probados se vuelven a probar combinados para probar sus interfaces.
- **Prueba del Sistema**: El software ensamblado totalmente con cualquier componente hardware que requiere se prueba para comprobar que se cumplen los requisitos funcionales.

<sup>114</sup> Sommerville Ian, Ingeniería del Software, Pearson Educación, S.A., Madrid, 2005, Pág. 220

<sup>115</sup> Ibid



- **Pruebas de Aceptación:** El cliente comprueba que el software funciona según sus expectativas.
- **Pruebas de regresión:** Las pruebas de regresión tienen por objetivo comprobar que los cambios sobre un componente del sistema, no generan errores adicionales en otros componentes no modificados.

Estas pruebas son las recomendadas para garantizar la calidad de un software desarrollado bajo el ciclo de vida en cascada, en el ámbito informático a la aplicación de estas se le conoce como Modelo cascada V<sup>116</sup>.

Para cada una de las pruebas se aplico la metodologías y estándar de formularios de documentación establecidas en la etapa de diseño de SIEMPOSAL.

El cuadro 6.5 y cuadro 6.6 presentan las generalidades de las pruebas realizadas para garantizar la calidad del software aquí descrito.

| Tipo     | Técnica o enfoque   | Estrategias | Descripción   |
|----------|---------------------|-------------|---|
| Estática | Recorridos          | Unitaria    | Al generarse errores de compilación o lógica se aislaba la clase y se revisaba cada segmento de código, hasta solventar la problemática.<br>Las unidades a las que se aplico este tipo de prueba son: <ul style="list-style-type: none"> <li>• Métodos</li> <li>• Sentencias.</li> <li>• Decisiones (ramas).</li> <li>• Condiciones múltiples.</li> </ul> |
|          | Revisiones formales | Unitaria    | Se realizo por medio de la metodología de checklist de código. Verificando el cumplimiento de: estándares de codificación y documentación en cada clase del software. Esta prueba se aplico estratégicamente a cada clase que conforma SIEMPOSAL.   |

Cuadro 6.5: Pruebas estáticas

<sup>116</sup>Modelo creado por el Departamento de Defensa Alemán, y sugerido por el Consejo Superior de Informática. Ministerio de Administraciones Públicas de España. <http://www.map.es/csi/metrica3>.



| Tipo     | Técnica o enfoque | Estrategias | Descripción   |
|----------|-------------------|-------------|---|
| Dinámica | Caja Negra        | Unitaria    | Se comprobó que cada nivel del videojuego presentara los escenarios e información deseada.  |
|          |                   | Integración | Esta estrategia se aplicó para validar que la integración de los diferentes niveles del videojuego fuera exitosa.   |
|          |                   | Sistema     | Al integrar todos los niveles se aplicó esta prueba para medir el rendimiento de la aplicación por medio de tiempo de carga de cada nivel, tiempo de respuesta, uso de RAM, CPU y GPU. En computadoras de diferentes características de hardware. |
|          |                   | Aceptación  | Por medio de las pruebas <b>Alfa</b> y <b>Beta</b> se comprobó que el software cumple con las expectativas de los usuarios.   |
|          |                   | Regresión   | Cada modificación al software es sujeta de generar errores o problemas con otros módulos. Esta prueba permite ir controlando cada modificación al sistema y su correcto funcionamiento.   |
|          | Caja Blanca       | Unitaria    | Se realizaron pruebas para cada una de la clase, en función de asegurar que las líneas de código que conforman cada método realizaran los procesos deseados.  |

Cuadro 6.6: Pruebas dinámicas

### 6.3.2. Recorridos

Todo programador debe revisar que cada línea de código de una clase realiza la lógica deseada de forma óptima y su objetivo principal es eliminar errores de compilación.

Es una técnica inherente a la programación de cualquier algoritmo, generalmente estas revisiones se plasman en la documentación interna de forma paralela a cada línea de código. Las líneas de código de SIEMPOSAL se verificaron por medio de la técnica de recorrido para evitar errores de sintaxis y lógica, apoyándose de la función de Microsoft Visual C# Express Edition que indica errores de sintaxis sin compilar el programa.

Todas las clases han sido documentadas usando de forma imperceptible esta técnica, las unidades o segmentos de código a las que se han especificado en el cuadro 6.6.



### **6.3.3. Revisiones Formales**

Es un proceso bien definido y disciplinado donde *un equipo de personas* analiza un producto software usando una *técnica de lectura* con el propósito de detectar defectos.

#### **6.3.3.1. Objetivo**

El objetivo principal de aplicar las revisiones formales es detectar faltas antes de que la fase de prueba comience. Cualquier desviación de una propiedad de calidad predefinida es considerada un defecto.

#### **6.3.3.2. Unidad Verificada**

Las revisiones formales se hicieron durante todo el proceso de codificación de las clases siguiendo la metodología especificada en el plan de pruebas, las unidades verificadas mediante esta técnica son las clases.

#### **6.3.3.3. Resultados de las Revisiones**

El cuadro 6.7 presenta los resultados de la revisión formal realizada a la clase Videojuego3D como resultado de esta revisión se estandarizaron 16 líneas de código (7 nombres de variables y 9 líneas documentadas).

Para cada una de las clases se completo la revisión formal, para visualizar el detalle de la prueba de cada clases del software ver el disco compacto adjunto a este documento en Pruebas/Revisiones formales/Detalles.doc

En base a los resultados obtenidos en el cuadro 6.8 se presenta la estadística de las correcciones al software que esta metodología permitió.



| <b>SOFTWARE INTERACTIVO PARA LA EDUCACIÓN MEDIOAMBIENTAL DE LA POBLACIÓN SALVADOREÑA.</b>  |   |                        |
|--|---|------------------------|
|  |   | Fecha: <u>08/07/08</u> |
| <b>Código Revisión:</b> R-0001   | <b>Alias:</b> Thesis                          |                        |
| <b>Encargado Revisión:</b>   |   |                        |
| <b>Clase:</b> VideoJuego3D   | <b>Nombre archivo físico:</b> VideoJuego3D.cs |                        |
| <b>Lógica del programa:</b>  | Si  | No                     |
| ¿Es correcta la lógica?  | ✓   |                        |
| <b>Estándares</b>  |   |                        |
| ¿Se han estandarizado los nombres de los métodos?  | ✓   |                        |
| ¿Todas las variables cumplen el estándar de nombre especificado en diseño?   | ✓   |                        |
| ¿Se ha cumplido con el estándar de documentación de métodos y clases?  | ✓   |                        |
| ¿Los recursos instanciadas en la clase se necesitan realmente?   | ✓   |                        |
| <b>Interfaces Internas:</b>  | ✓   |                        |
| ¿Son consistentes las definiciones de variables globales de la clase entre los métodos?  | ✓   |                        |
| ¿Se ha creado un constructor de clase correctamente?   | ✓   |                        |
| ¿Los modificadores de acceso de cada método son correctos?   | ✓   |                        |
| ¿Cada sobre carga de método es realmente necesaria?  | ✓   |                        |
| <b>Interfaces Externas:</b>  |   |                        |
| ¿Se crean los archivos .sav con todos sus atributos correctamente?   | ✓   |                        |
| ¿Se manejan correctamente las condiciones de excepción al realizar la apertura y cierre de los archivos?   | ✓   |                        |
| ¿Se les libera de memoria?   | ✓   |                        |
| ¿Se manejan correctamente los posibles errores de entrada/salida (try y catch)?  | ✓   |                        |
| <b>Datos</b>   |   |                        |
| ¿Todas las variables son usadas?   | ✓   |                        |
| ¿Las longitudes y tipos de las variables son correctas?  | ✓   |                        |
| ¿Los cálculos o divisiones por cero están validados?   | ✓   |                        |
| <b>Líneas de código estandarizadas:</b> 16   | <b>Instancia de recursos no usados:</b> 2     |                        |
| <b>Errores de Sintaxis:</b> 0  |   |                        |
| <b>Notas:</b> Se estandarizo la documentación de la clase en las líneas 5, 6, 82,174-176.  |   |                        |
| <b>Resultado:</b> Corrección de estándar de nombres de 7 variables, 9 líneas documentadas y 2 texturas estandarizadas, se corrigió el algoritmo de almacenamiento de archivos .sav ya que se almacenaban datos duplicados del nivel del jugador. |   |                        |

Cuadro 6.7: Revisión formal de la clase VideoJuego3D



| Correcciones   | Frecuencia |
|--|------------|
| Sintaxis   | 0          |
| Estándar de nombres de variables                                 | 12         |
| Estándar de nombre de métodos                                    | 6          |
| Estándar documentación interna.                                  | 55         |
| Constructor de la clase  | 4          |
| Modificador de acceso de los métodos(Private,Public o Protected) | 7          |
| Sobre carga de métodos   | 0          |
| Creación de archivo .sav   | 1          |
| Apertura de archivo  | 0          |
| Cierre de archivo  | 0          |
| Longitudes y tipos de variables                                  | 0          |
| Divisiones por cero  | 4          |
| Variables no usadas  | 110        |
| Recursos usados por la clase                                     | 16         |

Cuadro 6.8: Resultado de las revisiones formales

### 6.3.4. Caja negra

El objetivo de la aplicación de esta técnica fue encontrar una serie de datos de entrada cuya probabilidad de pertenecer al conjunto de entradas que causan un comportamiento erróneo sea lo más alto posible.

#### 6.3.4.1. Unitaria

Es el proceso de hacer pruebas sobre los componentes individuales (subprogramas o procedimientos) de un programa. El propósito es encontrar discrepancias entre la especificación de la interfaz del módulo y su comportamiento real.

##### Objetivo

El objetivo es comprobar que el módulo, entendido como una unidad funcional de un programa independiente, está correctamente codificado.

##### Unidad verificada

Esta prueba se realizó para cada uno de los módulos principales que conforman el videojuego durante toda la etapa de codificación.

##### Resultados de las pruebas

Para cada una de los niveles se crearon diferentes casos de prueba, el cuadro 6.9 presenta las pruebas realizadas al módulo de gestión de nivel de oxígeno.



| Software Interactivo para la Educación Medioambiental de la Población Salvadoreña  |         |  |  |       |        |                      |
|--|---------|--|--|-------|--------|----------------------|
|  |         |  |  |       |        | Fecha: 20/10/08      |
| Técnica :Caja negra  |         |  | Estrategia: Unitaria-modulo Nivel Futuro |       |        |                      |
| Código Prueba: P-0001  |         |  | Alias: Zebra                             |       |        |                      |
| Encargado de prueba:   |         |  |  |       |        |                      |
| Elemento evaluado: Gestión de nivel de oxigeno o vida en el nivel del futuro.  |         |  |  |       |        |                      |
| Objetivo de la prueba: Verificar que los niveles de oxigeno se actualicen correctamente de acuerdo a las acciones realizadas por el jugador. |         |  |  |       |        |                      |
| DATOS DE PRUEBA  |         |  |  |       |        |                      |
| No Prueba  | Entrada | Valor introducido  |  | Valor | Valido | Mensaje del software |
|  |         | Posicion Jugador (x,y,z)   | Resultado Esperado                       |       |        |                      |
| 1  | Tecla W | (10,50,70)<br>Recoge Bonus   | Oxigeno+10                               | -     | ✓      |                      |
|  | Tecla W | (10,50,220)<br>Sin interceptar Bonus   | Oxigeno                                  | -     | ✓      |                      |
| 2  | Tecla Y | (10,50,70)<br>Sin Bonus  | Oxigeno                                  | -     | ✓      |                      |
|  | Tecla Y | (10,50,70)<br>Sin bonus  | Oxigeno                                  | -     | ✓      |                      |
| 3  | Tecla W | (440,60,77)<br>Posición vomito<br>Quasimodo  | Oxigeno-10                               | -     | ✓      |                      |
|  | Tecla W | (440,60,79)<br>No intercepta Vomito  | Oxigeno                                  | -     | ✓      |                      |
| <b>Criterio aprobación de prueba</b>   |         | Si el Oxigeno se actualiza correctamente de acuerdo a las acciones del jugador en todo el nivel. |  |       |        |                      |
| <b>Resultado:</b>  |         | Los casos de prueba son validos, por lo tanto el oxigeno se actualiza correctamente              |  |       |        |                      |
| <b>OBSERVACION:</b>  |         | La prueba valido el nivel de oxigeno de 0 (todo rojo)a 256(verde total)                          |  |       |        |                      |

Cuadro 6.9: Resultado de la prueba unitaria para el modulo gestión de oxigeno

Para cada una de los módulos se completo la revisión formal, esto permitió verificar cada una de las funcionalidades del videojuego.

### 6.3.4.2. Integración

Las pruebas de integración consisten en una progresión ordenada de pruebas de los componentes como el armado de un rompecabezas, para lo cual los distintos módulos van siendo ensamblados y probados hasta haber integrado el sistema completo.





Objetivo

El objetivo de las pruebas de integración es verificar el correcto ensamblaje entre los distintos componentes una vez que han sido probados unitariamente con el fin de comprobar que interactúan correctamente a través de sus interfaces

Unidad verificada

Se aplico esta prueba a la integración de clases y niveles, así como a la integración de módulos como cargar avance, mostrar bitácora, configurar teclas y mostrar demo.

Resultados de las pruebas

La integración de los diferentes módulos del sistema fue exitosa, esta técnica permitió garantizar el correcto funcionamiento del videojuego, por medio de la depuración de errores generados por la integración de módulos.

El cuadro 6.11 presenta la prueba de integración entre los módulos guardar avance (implícito en cada nivel por medio de puntos de control) y cargar avance, por medio de los casos de prueba se certifico que los datos almacenados en los archivos .sav se cargan correctamente en el modulo de cargar avance.

De igual forma se procedió con todos los módulos hasta la unión de todo el videojuego, el detalle de estas pruebas se puede visualizar en Pruebas/Caja Negra /integracion.doc

Los principales problemas que se solventaron en esta prueba son los presentados en el siguiente cuadro:

| Correcciones   | Frecuencia |
|--|------------|
| Variables públicas compartidas en diferentes niveles se inicializaban globalmente. | 200        |
| Acumuladores y contadores globales no se reinicializaban al cargar los niveles.    | 30         |
| Recursos no incluidos en el proyecto   | 50         |

Cuadro 6.10: Principales errores detectados con esta prueba



| SOFTWARE INTERACTIVO PARA LA EDUCACIÓN MEDIOAMBIENTAL DE LA POBLACIÓN SALVADOREÑA   |                      |  |                    |                  |                |  |
|---|----------------------|--|--------------------|------------------|----------------|--|
|   |                      |  |                    |                  | Fecha: 2/10/08 |  |
| Técnica :Caja negra   |                      | Estrategia: Integrada  |                    |                  |                |  |
| Código Prueba: P-0002   |                      | Alias: Zebra   |                    |                  |                |  |
| Encargado de prueba:  |                      |  |                    |                  |                |  |
| Elementos a evaluar: Módulos guardar avance y cargar avance.  |                      |  |                    |                  |                |  |
| Objetivo de la prueba: Verificar que ante el progreso de un juego y sus almacenamientos mediante puntos de control; estos datos se encuentran disponibles para su utilización en el menú principal mediante el componente cargar juego. |                      |  |                    |                  |                |  |
| DATOS DE PRUEBA   |                      |  |                    |                  |                |  |
|   | Campo de entrada     | Valor introducido  |                    | Valor almacenado | Validación     | Respuesta del software   |
|   |                      | Valor  | Resultado Esperado |                  |                |  |
| 1   | Nivel                | True   | Válido             | True             | Válido         | Datos almacenados en los archivos .sav<br>Disponibles para ser cargados. |
|   | NivelOxigeno         | False  | Válido             | False            | Válido         |  |
|   | rapidezFlt           | False  | Válido             | False            | Válido         |  |
|   | seleccionInt         | False  | Válido             | False            | Válido         |  |
|   | EscenaInt            | False  | Válido             | False            | Válido         |  |
|   | EscenaInt            | 5  | Válido             | 5                | Válido         |  |
|   | VidasInt             | 2  | Válido             | 2                | Válido         |  |
|   | NiveldeVidaFlt       | 0.75   | Válido             | 0.75             | Válido         |  |
|   | dificultadInt        | 1  | Válido             | 1                | Válido         |  |
|   | PosicionV3D.x        | 10   | Válido             | 10               | Válido         |  |
|   | PosicionV3D.y        | 50   | Válido             | 50               | Válido         |  |
|   | PosicionV3D.Z        | 57   | Válido             | 57               | Válido         |  |
|   | EcoDolaresDbl        | 105.5  | Válido             | 105.5            | Válido         |  |
|   | EcoPuntosDbl         | 200  | Válido             | 200              | Válido         |  |
|   | objetosRecSTR.nombre | Jugo   | Válido             | Jugo             | Válido         |  |
| objetosRecSTR.cantidad  | 4                    | Válido   | 4                  | Válido           |                |  |
| <b>Criterio aprobación de prueba</b>  |                      | Si se cumplen en un 100% los resultados esperados y los valores almacenados son validos. |                    |                  |                |  |
| <b>Resultado:</b>   |                      | Integración de componentes confirmada.   |                    |                  |                |  |
| <b>OBSERVACION:</b>   |                      | Mientras no se validen los datos de entrada, las salidas serán erróneas.                 |                    |                  |                |  |

Cuadro 6.11: Resultado de la prueba de integración de los módulos guardar avance y cargar avance.



### 6.3.4.3. Sistema

Son pruebas de integración del sistema completo, y permiten probar el sistema en su conjunto y con otros sistemas con los que se relaciona para verificar que las especificaciones funcionales y técnicas se cumplen. Dan una visión muy similar a su comportamiento en el entorno de producción.

#### Objetivo de la prueba:

Las pruebas del sistema tienen como objetivo ejercitar profundamente el sistema, comprobando la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de sistemas con los que se comunica.

#### Unidad verificada

Se probó el rendimiento de cada nivel que compone el videojuego.

#### Resultados de la prueba:

Los resultados de esta prueba se presentan en base a los aspectos siguientes:

**Rendimiento:** Los equipos usados para las pruebas están ordenados de formas ascendentes donde el equipo 1 es el que proporciona las características de hardware con el rendimiento más bajo de entre los cuatro equipos y el 4 es el que posee las características de rendimiento más alto. Los dos primeros equipos tienen características bajas para aplicaciones 3D según los estándares de hardware actuales. Los equipos 3 y 4 se considera que tienen características medias altas según los estándares de hardware actuales. Por otro lado, se ha calificado rendimiento obtenido según el siguiente cuadro:

| Frecuencia de cuadros por segundo la mayor parte de tiempo | Calificación del rendimiento |
|--|------------------------------|
| <15 fps  | Pobre                        |
| 15 a 30 fps  | Bajo                         |
| 30 a 45 fps  | Aceptable                    |
| 45 a 60 fps  | Bueno                        |
| >60 fps  | Excelente                    |

Cuadro 6.12: Calificación de rendimiento



A continuación se muestran los resultados de las pruebas:

### Caso de prueba 1

Se realizaron las pruebas con el nivel con mayor número de vértices, para este caso el Centro de San Salvador posee 1500 vértices.

| Equipo | Características  | Fps mínimo | Fps Máximo |
|--------|--|------------|------------|
| 1      | Procesador Intel 600 MHz<br>RAM: 192 Mb<br>Tarjeta gráfica integrada Integrada con soporte DirectX 9.0 y Shader Model 1.1    | 5          | 10         |
| 2      | Procesador Pentium 1 GHz,<br>RAM: 256 Mb<br>Tarjeta gráfica integrada Integrada con soporte DirectX 9.0 y Shader Model 1.1   | 8          | 22         |
| 3      | Procesador Pentium 2.2 GHz,<br>RAM: 256 Mb<br>Tarjeta gráfica integrada Integrada con soporte DirectX 9.0 y Shader Model 1.1 | 12         | 42         |
| 4      | Procesador Pentium 2.8 GHz<br>RAM: 512 Mb<br>Soporte a DirectX 9.0 Shader Model 2.0 recomendado)                             | 35         | 60         |

Cuadro 6.13: Resultados prueba de rendimiento e

### Caso de prueba 2

Esta prueba se realizo con el nivel menor número de vértices 400, para este caso la Eco góndola.

| Equipo | Características  | Fps mínimo | Fps Máximo |
|--------|--|------------|------------|
| 1      | Procesador Intel 600 MHz,<br>RAM: 192 Mb<br>Tarjeta gráfica con soporte Shader Model 1.1             | 12         | 18         |
| 2      | Procesador Pentium 1 GHz,<br>RAM: 256 Mb<br>Tarjeta gráfica integrada con soporte Shader Model 1.1   | 20         | 28         |
| 3      | Procesador Pentium 2.2 GHz,<br>RAM: 256 Mb<br>Tarjeta gráfica integrada Integrada y Shader Model 1.1 | 35         | 45         |
| 4      | Procesador Pentium 2.8 GHz<br>RAM: 512 Mb<br>Soporte a DirectX 9.0 Shader Model 2.0 recomendado)     | 55         | 60         |

Cuadro 6.14: Resultados de prueba de rendimiento en nivel de bajos vértices



La prueba 1 se dio un rendimiento “Pobre” en equipos con prestaciones bajas, en equipos intermedios (2 y 3) el funcionamiento es “Aceptable”, en computadoras de buenas prestaciones fue “Excelente”. El cuadro 6.15 presenta datos promedios de la prueba de rendimiento en cuanto a uso de Memoria y CPU de la prueba 1:

| Característica                         | Unidades promedio |
|--|-------------------|
| Tiempo promedio de respuesta a eventos | 10 segundos       |
| Tiempo promedio de carga el videojuego | 20 segundos       |
| Promedio de uso de RAM                 | 40 MB             |
| Porcentaje de uso de CPU               | 55%               |

Cuadro 6.15: Rendimiento promedio del software en caso 1

En la prueba 2 del nivel menor de vértices se observó que el equipo 1 mostro un “Bajo rendimiento”. En equipos con prestaciones intermedias se obtuvo la mayor parte del tiempo un “Buen” rendimiento llegando en ocasiones a “Excelente”; en las computadoras con buenos rendimientos siempre se presentó un rendimiento “Excelente”.

| Característica                         | Unidades promedio |
|--|-------------------|
| Tiempo promedio de respuesta a eventos | 8 segundos        |
| Tiempo promedio de carga el videojuego | 12 segundos       |
| Promedio de uso de RAM                 | 45 MB             |
| Porcentaje de uso de CPU               | 44%               |

Cuadro 6.16: Rendimiento promedio del software en caso 2

#### 6.3.4.4. Aceptación

Las pruebas de aceptación tienen como finalidad validar que el sistema cumple los requisitos de funcionamiento establecidos y comprobar el grado de aceptación del sistema por parte de los usuarios finales.

##### Objetivo

El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento

##### Alpha

Son llevadas a cabo por un usuario en un entorno controlado por el equipo de desarrollo. Se requiere que el ambiente de trabajo dispuesto para estas pruebas presente o simule las mismas condiciones de las instalaciones reales de producción.



Características Mínimas y Máximas del equipo utilizado para la prueba son las siguientes:

| Prestaciones del equipo | Especificación del hardware   |
|-------------------------|---|
| Mínimo                  | <b>Procesador:</b> Intel Celeron 3.2 GHz<br><b>Memoria RAM:</b> 1GB<br><b>Disco duro:</b> 160 GB<br><b>Video:</b> GeForce 7800 GTX 256<br><b>Monitor:</b> Monitor CRT Compaq MV7540e de 17 pulgadas, 1280 x 1024 píxeles  |
| Máximo                  | <b>Procesador:</b> AMD Athlon 64 X2 Dual Core Processor 4800+2.51 GHz<br><b>Memoria RAM:</b> 1GB<br><b>Disco duro:</b> 300GB<br><b>Monitor:</b> Dell E176FP Black 17" LCD, 1280x1024 pixels<br><b>Video:</b> ATI Radeon 155x Series de 250 MB de video RAM, para shadders <sup>117</sup> de efectos 3.0 |

Cuadro 6.17: Hardware Mínimo y Máximo usado en pruebas

El videojuego fue probado por 99 estudiantes de diferentes instituciones, el tiempo promedio de duración de prueba es de 1 hora 30 minutos. El cuadro 6.18 presenta los tiempos mínimos y máximos de duración.

| Número de Equipo | Estudiantes | Tiempo de prueba Mínimo | Tiempo de prueba Máximo |
|------------------|-------------|-------------------------|-------------------------|
| 6                | 99          | 30 minutos              | 2 horas                 |

Cuadro 6.18: Pruebas de aceptación

A cada uno de los participantes en estas pruebas se le interrogo por medio de una encuesta, sobre los siguientes puntos:

En base a la siguiente escala seleccione como califica el videojuego:

Excelente

Bueno

Regular

Malo

<sup>117</sup> Ibid, Pág. 310 . Condiciones Tecnológicas - Hardware utilizado.



El 75% de los estudiantes califico el videojuego como excelente, mientras que el 17% como Bueno y solamente el 8% considero que era regular. A continuación se muestran los gráficos respectivos.

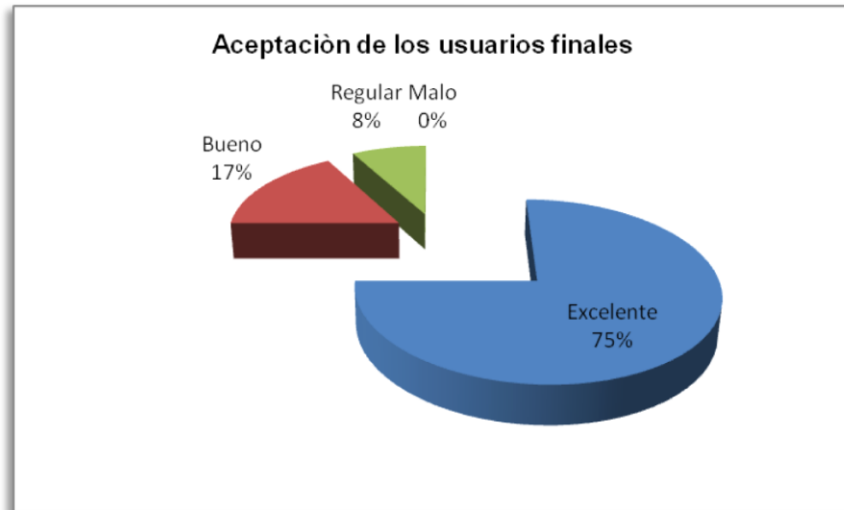


Gráfico 3.1: Aceptación del videojuego

En cuanto a su uso el videojuego le resultado:

Fácil

Complicado



Gráfico 3.2: Facilidad de Uso



Se llevo un control de los errores que se detectaron con esta prueba y que posteriormente permitieron su corrección:

| Correcciones   | Frecuencia |
|--|------------|
| Colisión nivel Salvador del Mundo                                  | 6          |
| Colisión de automóvil con calles                                   | 4          |
| Eliminación de Quasimodo nivel del futuro                          | 2          |
| Problema con terreno nivel del futuro debido a tarjetas de 16 bits | 1          |
| Parpadeo efecto mapa normal en nivel Salvador del Mundo            | 1          |
| Problema con animación de eco ambiente                             | 1          |

Cuadro 6.19: Correcciones de esta prueba

**Beta:**

Esta prueba se realizo en las instalaciones reales de los usuarios siguiendo la metodología establecida por el plan de pruebas, se instalo el software en 10 computadoras de 3 Aulas Informáticas ubicadas en el área metropolitana de San Salvador. Las características de las computadoras de las aulas informáticas se encuentran en los rangos presentados en el siguiente cuadro:

| Prestaciones del equipo | Especificación del hardware   |
|-------------------------|---|
| Mínimo                  | <b>Procesador:</b> Pentium IV<br><b>Memoria RAM:</b> 256 MB<br><b>Disco duro:</b> 60GB<br><b>Video Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps |
| Máximo                  | <b>Procesador:</b> Pentium D<br><b>Memoria RAM:</b> 1 GB<br><b>Disco duro:</b> 60GB<br><b>Video Tarjeta de red:</b> Ethernet con conector Rj-45, velocidad 10/100 Mbps    |

Cuadro 6.20: Rango de características del equipo de Aulas Informáticas MINED

Al evaluar el nivel que contiene más vértices y por tanto demanda mayores recursos de computador se obtuvieron los siguientes datos promedio:

| Equipo con prestaciones | Fps Mínimo | Fps Máximo |
|-------------------------|------------|------------|
| Mínimas                 | 12         | 17         |
| Promedio                | 33         | 44         |
| Máximo                  | 48         | 55         |

Cuadro 6.21: Frame por segundo en equipo de Aulas Informáticas

La prueba beta revelo que se solvento el problema de incompatibilidad con tarjetas de video de 16 bit.





En base al cuadro 6.21 se estableció que para equipos con las menores capacidades el rendimiento promedio del videojuego es “Bajo”, en las computadoras con hardware promedio el rendimiento es “Aceptable” y en equipo con buenas prestaciones es “Alto”.

#### Conclusiones de la prueba de aceptación

La prueba Alpha permitió corroborar que los usuarios finales en este caso los estudiantes interactuaran con el Videojuego, para proporcionarnos parámetros de aceptación de este software. En conclusión se ha establecido mediante la prueba Alpha que SIEMPOSAL tiene el 92% de aceptación y sus usuarios consideran que es una aplicación de fácil uso; con la prueba Beta se ha establecido que la aplicación corre aceptablemente en el hardware de las Aulas informáticas.

### **6.3.5. Caja blanca**

Las pruebas de caja blanca se detienen fundamentalmente en “como” está hecho el código. Para ello definimos la cobertura que es una medida porcentual del código que hemos cubierto con nuestras pruebas. Tenemos diferentes posibilidades para definir nuestra cobertura:

- Cobertura unitaria de métodos

#### **6.3.5.1. Objetivo**

El objetivo es comprobar que cada método, entendido como una unidad funcional de un programa independiente (clase), está correctamente codificado.

#### **6.3.5.2. Unidad verificada**

Son los métodos principales de cada clase que conforma a SIEMPOSAL. El cuadro 6.22 presenta la prueba realizada al método encargado de dibujar los árboles en los diferentes niveles.



| <b>Software Interactivo para la Educación Medioambiental de la Población Salvadoreña</b>  |   |
|---|---|
| Fecha: <u>24/10/08</u>  |   |
| Técnica :Caja Blanca  | Estrategia:Unitaria-Metodo  |
| <b>Código Prueba:</b> P-0003  | <b>Alias:</b> Zebra <b>Encargado de prueba :</b> Roberto Antonio Zelaya |
| Elementos a evaluar: Que los árboles se dibujen correctamente en el mundo ·3D   |   |
| Objetivo de la prueba: Verificar que se dibujan árboles con los efectos y técnicas especificadas.   |   |
| Salida esperada :Dibujado de un árbol con efectos   |   |
| <b>Clase:</b>   | Arboles   |
| <b>Ubicación:</b>   | SIEMPOSAL/Arboles.cs  |
| Código  |   |
| <pre>public void DibujarArboles(GraphicsDevice device, Matrix Proyeccion,Matrix Vista) {     bbEffect.CurrentTechnique = bbEffect.Techniques["CylBillboard"];     bbEffect.Parameters["scale"].SetValue(escalaX);     bbEffect.Parameters["xWorld"].SetValue(Matrix.Identity);     bbEffect.Parameters["xView"].SetValue(Vista);     bbEffect.Parameters["xProjection"].SetValue(Proyeccion)     bbEffect.Begin();         bbEffect.CurrentTechnique.Passes[0].End();         device.RenderState.DepthBufferWriteEnable = false;         device.RenderState.DestinationBlend = Blend.InverseSourceAlpha;         device.RenderState.AlphaTestEnable = true;         bbEffect.CurrentTechnique.Passes[0].Begin();         device.DrawPrimitives(PrimitiveType.TriangleList, 0, noTriangles);         bbEffect.CurrentTechnique.Passes[0].End();     device.RenderState.AlphaBlendEnable = false;     device.RenderState.DepthBufferWriteEnable = true;     device.RenderState.AlphaTestEnable = false;     bbEffect.End(); }</pre> |   |
| <b>Criterio aprobación de prueba:</b> Todas las líneas de código están 100% examinadas y se obtienen las salidas deseadas.  |   |
| <b>Líneas de Código Modificadas: 3</b>  |   |

Cuadro 6.22: Caja Blanca

### 6.3.5.3. Conclusiones del enfoque de caja blanca

Este enfoque permitió que durante todo el desarrollo del videojuego se evaluara los parámetros de entrada a cada método y se verificaran las correctas salidas en el mundo 3D. Se aplico este enfoque en el desarrollo de cada método que conforman las diversas clases del Videojuego y que se puede consultar en el diccionario de datos incluido en el disco compacto.



### **6.3.6. Conclusiones de las pruebas realizadas a SIEMPOSAL**

Por medio de las pruebas aplicadas a SIEMPOSAL se han descubierto errores de diferente índole permitiendo garantizar la calidad del software.

Las pruebas aplicadas consideran el rendimiento del software en el entorno de trabajo de las Aulas Informáticas y han corroborado que este tiene un funcionamiento óptimo en computadoras que cuentan con el hardware establecido en la etapa de requerimientos.

Se ha verificado que el software es fácil de usar para estudiantes entre el rango de edades: 7 años en adelante y la aceptación del videojuego es satisfactoria.



## **CAPITULO VII: DOCUMENTACIÓN Y PLAN DE IMPLANTACIÓN**



## 7. DOCUMENTACIÓN EXTERNA.

La documentación externa tiene por objeto presentar el conocimiento técnico y de usuario final para la implantación del SIEMPOSAL; los elementos que se agregan en esta fase son los siguientes:

### 7.1. DOCUMENTACIÓN DEL SOFTWARE.

**Manual del programador:** Este elemento permitirá conocer técnicamente la estructura de programación que el software del sistema contiene. Estos aspectos serán enfocados en los programas de aplicación y las validaciones así como las estructuras de programación.

**Manual de instalación:** Este proporciona información detallada de la instalación del software, se le muestran de manera detallada los pasos que debe realizar, así como también se le muestra las diversas pantallas que se visualizan en el proceso de instalación del videojuego. Tiene el objetivo de proporcionar una guía fácil y rápida para la instalación del software SIEMPOSAL.

**Manual del Usuario:** Este manual está dirigido a las personas que utilizaran el videojuego y presenta en forma detallada las acciones necesarias para el buen uso del software y su interacción en los escenarios.

Para consultar estos manuales acceda a los documentación proporcionada adjunta con este documento.

### 7.2. PLAN DE IMPLANTACIÓN DEL SOFTWARE.

Este manual contiene una pequeña descripción de la forma de distribución que tendrá el videojuego por las instituciones contraparte, ya sea por medio la distribución directa, indirecta y por el sitio web.

Para consultar este plan acceda a los documentación proporcionada adjunta con este documento.



## CONCLUSIONES

- En base al estudio de campo se detectó que existe en los estudiantes de básica, bachillerato y universitarios desconocimiento general de los métodos de preservación de los recursos naturales (83.3%) e instituciones encargadas de proteger el medio ambiente (76.1%) por lo que es necesario aplicar nuevas metodologías de educación ambiental.
- El software aquí descrito es una alternativa viable para concientizar a la población estudiantil y por medio del estudio de factibilidad se ha determinado que se cuenta con los recursos y condiciones necesarias a nivel técnico, económica, operativa y medioambiental para desarrollar e implementar este proyecto.
- Para un periodo de 5 años, el costo promedio de educar a un niño por medio del software es de \$0.95 el primer año, teniendo un ahorro de \$19.05 con respecto a la metodología actual, en los años restantes se tendría un costo promedio de \$0.60 por estudiante, teniendo un beneficio de \$19.40. En general se tendrá un ahorro total de \$ 1,930,237.66 al implementar el Videojuego como herramienta de concientización en 20,000 estudiantes.
- Al implementar el Videojuego, se tendrá un mayor alcance de estudiantes y un mayor alcance de temáticas medioambientales con respecto al sistema actual.
- Se cuenta con el respaldo necesario de parte de la Gerencia de Educación Ambiental (MARN) y el departamento de Infraestructura Tecnológica (MINED) para el desarrollo e implementación del proyecto.
- De acuerdo al estudio de campo realizado, se ha demostrado que el software será de gran aceptación en la mayoría de la población estudiantil (83%), niños y jóvenes del país (que es una población clave para la diseminación de información), por lo tanto se impactará significativamente en su forma de tratar al medio ambiente.
- Los beneficios que el software aportara a la sociedad salvadoreña no se limitan a los expuestos en el documento, ya que este impactará en la economía del país disminuyendo gastos debido a problemáticas ambientales, pérdida de productividad, gastos por administración de desechos sólidos, entre otros.



## RECOMENDACIONES

- Para disminuir el desinterés de la población en el involucramiento de actividades a favor del medioambiente es necesario aplicar metodologías innovadoras que impacten principalmente a los niños y jóvenes debido a que estos son los que presentan mayor desconocimiento sobre la problemática ambiental.
- Se debe mantener un estrecho contacto entre la Gerencia de Educación Ambiental y las Aulas Informáticas de los Centros Escolares involucrados en el Programa Nacional para el Reciclaje a fin de que se encuentren informados de los avances de la aplicación del software para brindar una retroalimentación oportuna que optimice el aprendizaje de los estudiantes.
- Es necesario que se distribuyan contenedores no solo en las escuelas donde se implemente el software, sino en todo nuestro país, en lugares estratégicos, con la finalidad de que se aplique en la realidad las acciones medioambientales que se promueven en el videojuego.
- La implementación del software por parte de la Gerencia de Educación Ambiental (MARN) no debe limitarse al sector educativo, también se puede alcanzar a la población no estudiantil que cuente con los recursos necesarios para ejecutar el software.
- Se deberán seguir los estándares de diseño y programación propuestos en este documento y los manuales correspondientes para facilitar el mantenimiento del software.



## BIBLIOGRAFÍA

### LIBROS

- Balbino Sebastián, Manual para Formulación Evaluación y Ejecución de Proyectos, 4ª. Edición, Grafic S.A. de C.V, 2004.
- García, Carlos Ernesto. Gerencia informática, 3ª Edición, San Salvador, Informatik S.A. de CV, 2005.
- Larman Craig. Uml y Patrones Introducción al análisis y diseño orientado a objetos, 1ª Edición, México D.F. Prentice Hall, 1999 -2003.
- Sommerville Ian, Ingeniería del Software, Pearson Educación, 1ª Edición, Madrid,SA, 2005.
- John W. Satzinger, JACKSON, BURD, Robert B. Jackson, Stephen D. Burd, SATZINGER, Systems Analysis and Design in a Changing World, 4ª Edición, Course Technology ,2006.
- "Analysis and Design" Considered Harmful (análisis y diseño que es considerado peligroso) Frequent pairing of different disciplines violates fundamental life-cycle principle. (El apareamiento frecuente de diferentes disciplinas viola principios fundamentales del ciclo de vida de proyectos), Conrad Weisert, Information Disciplines, Inc. February, 1999.
- Frank Buschmann , Regine Meunier , Hans Rohnert , Peter Sommerlad y Michael Stal, Arquitectura de Software orientado a Patrones, Editorial John Wiley and Sons LTD. Volumen 1, 2001.
- Scott W. Ambler, Agile Model Driven Development with UML 2. Ágil Desarrollo a través de Modelos con UML 2, The Object Primer 3ª Edición, Cambridge University, 2004.





## TRABAJOS DE GRADUACION

- **Carballo Ruiz, Elmer Arturo**, Análisis, diseño y desarrollo de un sistema de información mecanizado para el registro y manejo de los expedientes clínicos de los pacientes del Hospital Nacional Rosales, como una herramienta de apoyo para las investigaciones patológicas, Universidad de El Salvador, 2000.
- **Chávez Vásquez, Emely Maricruz**, Sistema informático para el apoyo a los procesos de compra, administración y despacho de medicamentos e insumos médicos del hospital nacional rosales de el salvador, Universidad de El Salvador, 2006.
- **Carlos Roberto Hernández Salomón**, Desarrollo de un Sistema de Información para el Sector Pesquero Artesanal de El Salvador, Universidad de El Salvador, 2003.

## SITIOS WEB

- <http://www.rree.gob.sv/sitio/sitio.nsf/0/9152999b18c314a706256eb600692813?OpenDocument>
- <http://www.marn.gob.sv/?fath=4&categoria=88>
- <http://www.microsoft.com/spain/empresas/productos/windowsxp/requisitos.mspx>
- <http://www.microsoft.com/spanish/msdn/vstudio/Express/Docs/Systemreqs/Systemreqs.mspx>
- [http://msdn.microsoft.com/en-us/library/cc627094\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/cc627094(VS.85).aspx)
- <http://www.microsoft.com/spanish/msdn/latam/estudiantes/articulos/nota3.aspx>
- <http://chumbalum.swissquake.ch/>
- <http://www.fragmosoft.com/fragMOTION/>
- <http://www.adobe.com/products/photoshop/photoshop/systemreqs/>
- <http://office.microsoft.com/en-us/products/HA101668651033.aspx>
- <http://www.gentleware.com/products.html>
- <http://support.microsoft.com/kb/837783>
- <http://www.microsoft.com/downloads/details.aspx?FamilyID=15fb9169-4a25-4dca-bf40-9c497568f102&displaylang=en>
- <http://www.microsoft.com/downloads/details.aspx?familyid=9226A611-62FE-4F61-ABA1-914185249413&displaylang=es>



- <http://web2development.blogspot.com/2007/05/patron-mvc.html>
- <http://jungla.dit.upm.es/~santiago/externos/docencia/doctorado/drci/apuntes/swing/sld016.htm>
- [http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ\\_3317/default.aspx#M16](http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_3317/default.aspx#M16)
- <http://www.osmosislatina.com/lenguajes/uml/compdeploy.htm>
- <http://www.liebertonline.com/doi/abs/10.1089/109493104322820057>
- <http://www.anderson.ucla.edu/x2555.xml>
- <http://www.liebertonline.com/doi/abs/10.1089/109493100316067?cookieSet=1&journalCode=cpb>
- <http://marketshare.hitslink.com/report.aspx?qprid=8>
- <http://www.enterate.unam.mx/Articulos/2004/agosto/vidjava.htm>
- <http://psicopedagogias.blogspot.com/2008/02/el-desarrollo-ldico.html>
- <http://www.merriam-webster.com/dictionary/>
- [http://www.ojp.usdoj.gov/BJA/evaluation/glossary/glossary\\_i.htm](http://www.ojp.usdoj.gov/BJA/evaluation/glossary/glossary_i.htm)



## GLOSARIO DE TERMINOS

### A

**Artefacto:** Objeto formado por un conjunto de piezas y fabricado para un fin determinado, en especial el que no constituye una máquina, aparato o dispositivo definidos.

**Atributos:** Los atributos de un objeto son aquellas variables que definen dicho objeto. Las funciones de la clase deben de permitir trabajar con ellos para realizar las modificaciones pertinentes o simplemente para ser consultados ya que son la parte fundamental de dicho objeto.

### C

**Casuística:** Conjunto de los diversos casos particulares que se pueden prever.

**Clase:** Abstracción o definición de un objeto.

**Coexistencia:** Sub-característica de portabilidad, que indica la capacidad del software de coexistir con otro independiente en un entorno común compartiendo recursos (ISO 9126).

### CH

**Check Point:** Punto de evaluación, en el contexto de videojuegos se refiere a una acción, nivel alcanzado o cualquier evento que al dispararse indique al sistema que se avanzará o guardará el nivel actual.

### D

**Desertificación:** El proceso de la degradación de las tierras de zonas áridas, semiáridas y sub-húmedas, secas resultantes de diversos factores, tales como las variaciones climáticas y las actividades humanas.

**DirectX:** Es una colección de APIs creadas y recreadas para facilitar las complejas tareas relacionadas con multimedia, especialmente programación de juegos y vídeo en la plataforma Microsoft Windows.

### E

**EA:** Abreviatura de Educación Ambiental.

### F

**Framework:** Presenta una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

### G

**Generalización:** Una clase puede abstraerse en una clase más general y viceversa. Esto nos permite tener varios tipos de clases hijos de una clase padre según un discriminante con un comportamiento común o individual.



**Grupos etarios:** Grupos de personas en un rango de edades.

**GPU (Graphics Processing Unit):** Unidad de Procesamiento Grafico de las computadoras.

**GRASP:** Patrones de Software Generales para la Asignación de Responsabilidades

## I

**Intellisense:** Propiedad de Algunos Lenguajes de alto nivel, en donde el entorno de desarrollo le propone al usuario los diferentes métodos, clases, propiedades, variables entre otros recursos disponibles para un objeto dado en tiempo real durante la codificación.

## M

**MARN:** Abreviatura de Ministerio de Medioambiente y Recursos Naturales

**MINED:** Abreviatura de Ministerio de Educación.

**Microflora:** Representada por hongos, algas unicelulares y vegetales microscópicos que se encuentran en un suelo.

**Modelo:** Término que se refiere a una figura en 3D, en general que esté formada por polígonos.

## N

**NET:** Es un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

## O

**ONG:** Abreviatura de organización no gubernamental.

**OpenGL:** Es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D.

## P

**Pixel:** El píxel (del inglés Picture element) es el elemento base del cual están formadas todas la imágenes digitales.

**Polimorfismo:** Permite funciones con el mismo nombre que se diferencien, por ejemplo, en el tipo de parámetros, que tengan comportamientos totalmente diferentes.

## R

**Render:** Término que se refiere a la generación de un FRAME o imagen individual. El render consume potencia de cómputo debido a los cálculos necesarios según el número de objetos, luces y efectos en la escena y según el punto de vista de la cámara

**S**

**Shader:** Es un conjunto de instrucciones gráficas destinadas para el acelerador gráfico, estas instrucciones dan el aspecto final de un objeto. Los Shaders determinan materiales, efectos, color, luz, sombra y etc.

**Sprite:** Un sprite es una imagen o figura plana, que es usado en los juegos, en XNA es una clase que permite crear texturas 2D, normalmente se utilizan para crear efectos especiales.

**SVGA:** Es un término que cubre una amplia gama de estándares de visualización gráfica de ordenadores, incluyendo tarjetas de video y monitores.

**T**

**Textura:** Es una imagen, generalmente un bitmap, que será usada como mapa para renderizar sobre los polígonos de un objeto con el fin de aplicarle una apariencia dada al mismo.

**U**

**Usabilidad:** La medida en la cual un producto puede ser usado por usuarios específicos para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso especificado.

**V**

**Vértice:** Es el punto común entre los lados consecutivos de una figura geométrica, o el punto común de los dos lados de un ángulo, punto en que concurren tres o más planos, punto de una curva en que la encuentra un eje suyo normal a ella.

**W**

**Wrapper” (del inglés Envoltorio”):** Se le llama así a bibliotecas SDK generadas a partir de otras bibliotecas existentes, creando nuevos métodos y clases a partir de estas.

**X**

**XAML:** Lenguaje Extensible de Formato para Aplicaciones

**XML:** Sigla en inglés de Extensible Markup Language (lenguaje de marcas extensible)

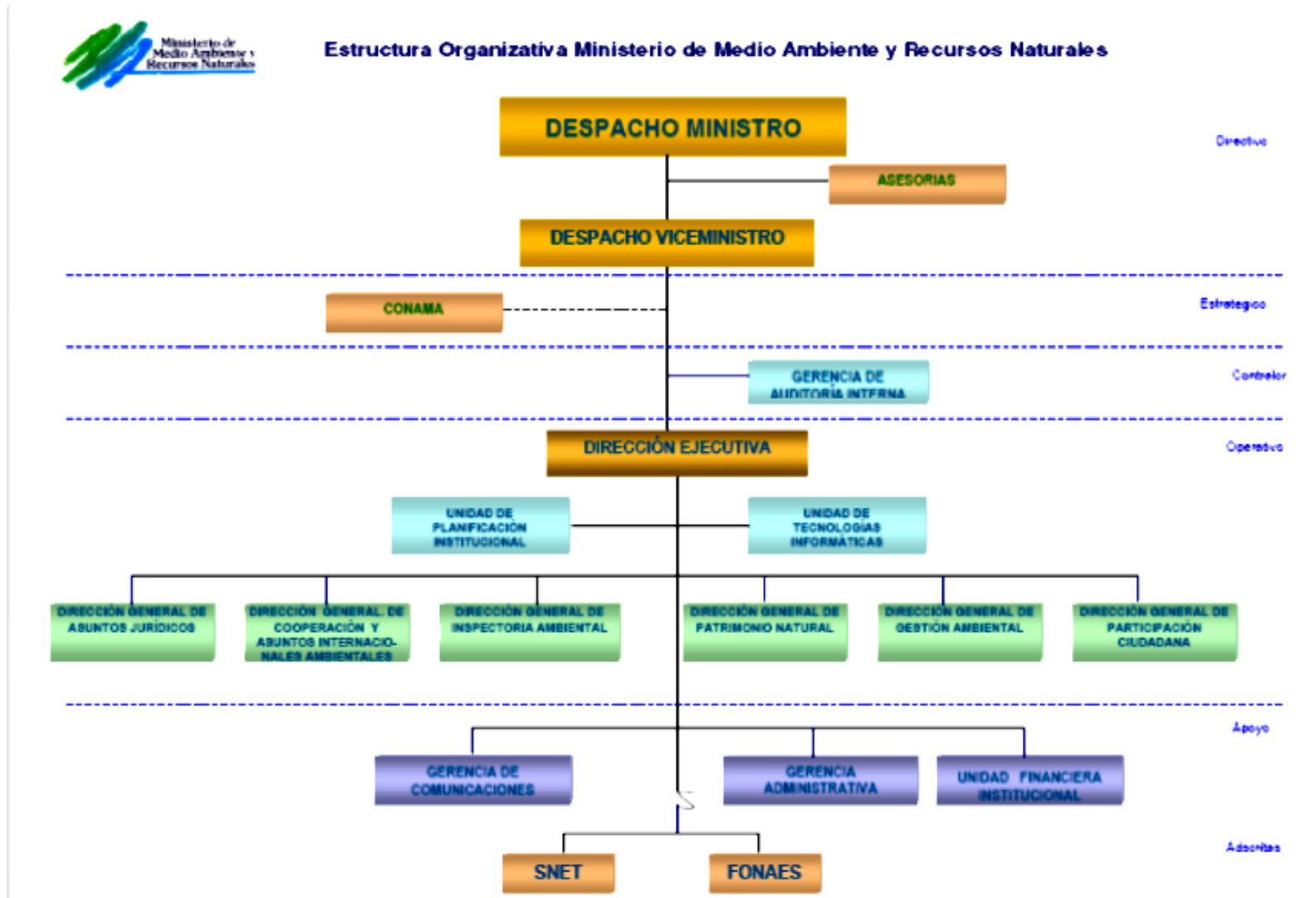
**XNA:** Conjunto de bibliotecas SDK para el desarrollo, pero no limitado a estos, de juegos para las plataformas Windows y XBOX 360.



# ANEXOS



**ANEXO No. 1: ORGANIGRAMA DEL MINISTERIO DE MEDIO AMBIENTE Y RECURSOS NATURALES**





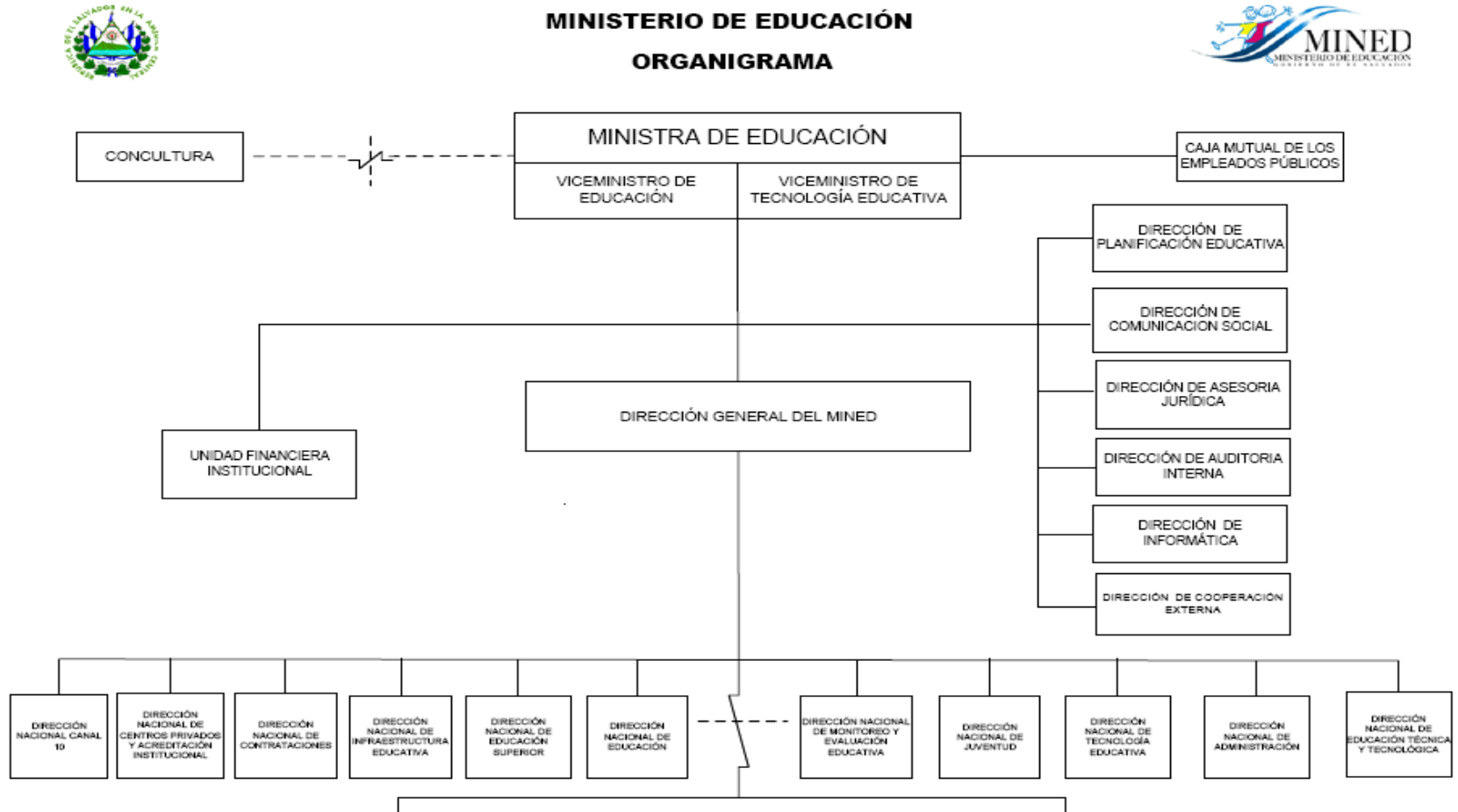
**ANEXO No. 2: ORGANIGRAMA DE LA DIRECCION GENERAL DE PARTICIPACION CIUDADANA (MARN)**







### ANEXO No. 3: ORGANIGRAMA DEL MINISTERIO DE EDUCACION





## ANEXO No. 4: CÁLCULO DE LOS VALORES ACTUALES NETOS (VAN) Y TASA INTERNA DE RETORNO (TIR)

Los valores mostrados en la Sección de Factibilidad Económica son valores ya resueltos y resumidos, obtenidos de las siguientes ecuaciones introducidas en una hoja de Excel 2007:

$$\text{VAN} = \sum_{n=0}^N \frac{I_n - E_n}{(1+i)^n}$$

En donde  $N$  es igual al número de años,  $I$  es ingreso,  $E$  el egreso e  $i$  es el impuesto (inflación para este caso)

Función Excel:

NPV (tasa, valor1, valor2,...)

### TIR

Igualando la VAN a cero

$$\text{VAN} = -I + \sum_{i=1}^N \frac{Q_i}{(1+TIR)^i} = 0$$

Y Donde  $Q_i$  es el Flujo de Caja en el periodo  $i$ .

Por el teorema del binomio:

$$(1+r)^{-n} \approx 1 - n * r$$

$$I = Q_1 * (1-r) + \dots + Q_n * (1 - n * r)$$

$$I - (Q_1 + \dots + Q_n) = -r * (Q_1 + \dots + n * Q_n)$$

De donde la TIR:

$$r = \frac{-I + \sum_{i=1}^n Q_i}{\sum_{i=1}^n i * Q_i}$$

Función Excel:

MIRR (valores, tasa ,0) en donde cero se implementa como tasa de reinversión, ya que no existe tal estudio se llama a la función Excel con esa variable establecida en cero.

Funciones consultadas desde [http://es.wikipedia.org/wiki/Tasa\\_interna\\_de\\_retorno](http://es.wikipedia.org/wiki/Tasa_interna_de_retorno) y [http://es.wikipedia.org/wiki/Valor\\_actual\\_neto](http://es.wikipedia.org/wiki/Valor_actual_neto).



## ANEXO No. 5: INSTRUMENTOS DE INVESTIGACION



**UNIVERSIDAD DE EL SALVADOR**  
**FACULTAD DE INGENIERIA Y ARQUITECTURA**  
**ESCUELA DE SISTEMAS INFORMATICOS**

Cuestionario para estudiantes de diferentes niveles educativos.

**Objetivo:** Recolectar información sobre el grado de concientización ambiental en los estudiantes de educación básica de El Salvador y el uso de videojuegos en la educación.

**Instrucciones:** Encierre en un círculo su respuesta(s) a las preguntas siguientes.

| No.  | PREGUNTAS  | CODIGO                     | ALTERNATIVAS   |
|--|--|----------------------------|--|
|  | Grado que cursa  | 1                          | Primero a tercer grado   |
|  |  | 2                          | Cuarto a sexto grado   |
|  |  | 3                          | Séptimo a noveno grado   |
|  | Sexo   | 1                          | Masculino  |
|  |  | 2                          | Femenino   |
|  | Edad   | 1                          | 5 a 9 años   |
|  |  | 2                          | 10 a 12 años   |
|  |  | 3                          | 13 a 16 años   |
|  |  | 4                          | 17 o más   |
| 1  | ¿Le gustan los videojuegos?  | 1<br>2                     | Si<br>No   |
| <b>Si respondió que no a la pregunta anterior por favor pase a la pregunta 7</b> |  |                            |  |
| 2  | ¿Con que frecuencia usa videojuegos?   | 1<br>2<br>3                | A diario<br>Frecuentemente<br>Escasamente  |
| 3  | ¿Cuáles de las siguientes, consolas o equipo de videojuegos posee o tiene acceso?        | 1<br>2<br>3<br>4<br>5<br>6 | Nintendo Wii<br>PlayStation 3<br>Nintendo DS<br>PlayStation Portable<br>Xbox 360<br>PC |
| 4  | ¿En cuál de las siguientes actividades preferiría pasar su tiempo libre?                 | 1<br>2<br>3<br>4           | Lectura<br>Televisión educativa.<br>Jugando videojuegos<br>Ninguno                     |
| 5  | ¿Considera que los videojuegos influyen en la conducta de las personas que los utilizan? | 1<br>2                     | Si<br>No   |
| 6  | ¿Crees que se aprende algo jugando videojuegos?  | 1<br>2                     | Si<br>No   |
| <b>Si , su respuesta fue No a la pregunta anterior pase a la número 11</b>       |  |                            |  |

Cuadro A.5.1: Cuestionario a estudiantes



| No.   | PREGUNTAS  | CODIGO  | ALTERNATIVAS  |
|---|--|---|---|
| 7   | ¿Cuáles de los siguientes habilidades y/o conocimientos, cree usted que son adquiridos por medio del uso de videojuegos? | 1<br>2<br>3<br>4<br>5<br>6<br>7<br>8<br>9<br>10<br>11 | Ingles/idiomas<br>Historia/geografía<br>Cultura general<br>Trabajo en equipo<br>Investigar/resolver problemas<br>Coordinar movimientos<br>Memoria<br>Concentración<br>Distinguir realidad<br>Creatividad<br>Otros.<br>Explique: _____ |
| 8   | ¿Le interesan a usted los problemas medio ambientales de nuestro país?   | 1<br>2  | Si<br>No  |
| 9   | ¿Cree que en nuestro país, se protege el medio ambiente?   | 1<br>2<br>3   | Si<br>No<br>No sabe   |
| 10  | ¿Cómo considera el estado del medio ambiente?  | 1<br>2<br>3<br>4<br>5                                 | Excelente<br>Bueno<br>Malo<br>Regular<br>No sabe  |
| 11  | ¿Ha recibido información sobre cómo proteger el medio ambiente?  | 1<br>2  | Si<br>No  |
| 12  | ¿Conoce métodos para proteger el medio ambiente?   | 1<br>2  | Si<br>No  |
| 13  | ¿Le gustaría aprender más sobre cómo proteger el medio ambiente?   | 1<br>2  | Si<br>No  |
| <b>Si, respondió que "No" a la pregunta anterior continúe respondiendo la pregunta número 19</b>  |  |   |   |
| 14  | ¿De cuáles de los siguientes problemas del medio ambiente, te gustaría saber más?  | 1<br>2<br>3<br>4<br>5<br>6<br>7                       | Contaminación de agua<br>Contaminación de aire<br>Extinción de especies<br>Reciclaje<br>Deforestación<br>Calentamiento global<br>Otros<br>Especifique: _____  |
| 15  | ¿Conoce usted instituciones que se encarguen de proteger el medio ambiente?  | 1<br>2  | Si<br>No  |
| <b>Si, respondió que "No" a la pregunta anterior, continúe respondiendo la pregunta número 21</b> |  |   |   |

Cuadro A.5.2: Cuestionario a estudiantes



**UNIVERSIDAD DE EL SALVADOR**  
**FACULTAD DE INGENIERIA Y ARQUITECTURA**  
**ESCUELA DE SISTEMAS INFORMATICOS**

**OBJETIVO:** Determinar la necesidad de ampliar y reforzar con nuevas técnicas la educación ambiental en la población estudiantil de educación media.

1. Grado al que imparte la materia Ciencia, salud y medio ambiente

|                        |                          |                        |                          |
|------------------------|--------------------------|------------------------|--------------------------|
| Primero a tercer grado | <input type="checkbox"/> | Séptimo a noveno grado | <input type="checkbox"/> |
| Cuarto a sexto grado   | <input type="checkbox"/> | Todos                  | <input type="checkbox"/> |

2. ¿Considera que los programas de estudio de la materia Ciencia, salud y medio ambiente, abarcan los contenidos necesarios para concientizar a los estudiantes de educación media sobre la problemática medioambiental de nuestro país?:

Si  No

3. ¿Le gustaría que se ampliaran los contenidos referentes a la educación ambiental en el programa de Ciencia, salud y medio ambiente?

Si  No

4. Si su respuesta en la pregunta anterior es "sí", ¿Cuales de los siguientes temas le gustaría que se abordaran o se ampliaran en las unidades del plan educativo de educación media?

|                       |                          |                       |                          |                      |                          |
|-----------------------|--------------------------|-----------------------|--------------------------|----------------------|--------------------------|
| Contaminación de agua | <input type="checkbox"/> | Extinción de especies | <input type="checkbox"/> | Deforestación        | <input type="checkbox"/> |
| Contaminación de aire | <input type="checkbox"/> | Reciclaje             | <input type="checkbox"/> | Calentamiento global | <input type="checkbox"/> |

Otros Especifique: \_\_\_\_\_

5. ¿Qué metodologías se utilizan en los programas de educación para concientizar a la población estudiantil sobre la problemática ambiental?

\_\_\_\_\_

6. ¿Considera que las metodologías señaladas en los programas de educación son suficientes para concientizar a la población estudiantil sobre la problemática ambiental?

Si  No

7. ¿Cree que el sistema educativo, necesita nuevas técnicas para apoyar la educación ambiental en nuestro país?

Si  No

8. ¿Existe en los centros educativos material de apoyo suficiente para que la población estudiantil de educación media, amplíe su conocimiento acerca de la problemática ambiental?

Si  No

9. ¿Cataloga usted la contaminación del medio ambiente como uno de los problemas más grandes de nuestro país?

Si  No

10. ¿Considera Ud. que por medio de un video juego educativo se podría fomentar la educación ambiental en los estudiantes sobre las problemáticas de nuestro medio ambiente?

Si  No



**UNIVERSIDAD DE EL SALVADOR**  
**FACULTAD DE INGENIERIA Y ARQUITECTURA**  
**ESCUELA DE SISTEMAS INFORMATICOS**

**Objetivo:** Conocer sobre los problemas que afronta nuestro medio ambiente por medio de la recopilación de información en los temas: Alto grado de deforestación, manejo adecuado de desechos sólidos, contaminación de aguas, contaminación del aire y extinción de especies y poder determinar la situación actual de nuestro medio ambiente y de nuestro país.

Aspectos Generales.

1. Cuál es el estado real en el que se encuentra el medio ambiente enmarcándonos en :
  - a. la flora
  - b. fauna,
  - c. las aguas,
  - d. el aire,
  - e. desechos sólidos
2. Mencione las causas que han llevado al deterioro de nuestro medio ambiente o nuestro planeta.
3. Cuáles han sido sus efectos?
4. Cuáles son los factores principales que contribuyen al deterioro de nuestro medio ambiente. Describir los factores.
5. Mencione las diferentes leyes por medioambiente están vigentes en El Salvador, en qué consisten.
6. Se ponen en prácticas las leyes que protegen el medio ambiente en nuestro país.
7. En qué medida se están poniendo en práctica las leyes que protegen a nuestro medio ambiente.
8. Cuáles son las estimaciones que se tienen del planeta en unos 10, 50 y 100 años si no cuidamos el medio ambiente.
9. Cuáles son los proyectos o programas que el MARN tiene para contrarrestar estos graves problemas.
10. Existen proyectos comunitarios permanentes u eventuales cuyos objetivos son de promover la concientización ciudadana en la correcta forma de despojarse de desechos sólidos.
11. Cree usted que la población no le da importancia a los problemas del medio ambiente.
12. Existen datos estadísticos que revelen el estado de nuestro medio ambiente?. Menciónelos.
13. De qué forma o que mecanismos utilizan para monitorear a lo largo de nuestro país que se respete y cuide nuestro medio ambiente.
14. Cuál cree que son los principales factores que intervienen en la falta de concientización sobre el cuidado del medio ambiente.
15. Hay algún programa de capacitación y destreza en inspección y apoyo técnico a los agricultores para que éstos alcancen y contribuyan a la sostenibilidad del medioambiente.
16. Cuál cree Ud. que es la mejor manera de transmitir la educación ambiental a la población.
17. Considera Ud. que los videojuegos, debido a su popularidad y adicción que crea en las personas, especialmente en los adolescentes, sea un medio idóneo por el cual se pueda concientizar al individuo sobre su medio ambiente y la forma en que este pueda contribuir para su sustentabilidad.



## ANEXO No. 6: ESTUDIANTES POR GRUPO ETARIO

| Grado              | Grupo etario      | Estudiantes |
|--------------------|-------------------|-------------|
| 4° a 6°            | De 10 a 12 años   | 114,359     |
| 7°a 9°             | De 13 a 15 años   | 98,732      |
| Bachillerato       | De 16 a 18 años   | 64,458      |
| Educación Superior | De 17 en adelante | 60,160      |
|                    |                   | 337,709     |

Cuadro A.6.1: Grupos etarios de la población estudiantil de educación básica<sup>118</sup>

## ANEXO No. 7: GRAFICO ACEPTACION DE VIDEOJUEGOS

El 83% de los estudiantes gustan de los videojuegos, esta cifra se observa en la siguiente, reflejando la gran aceptación del juego mediante consolas y ordenadores.

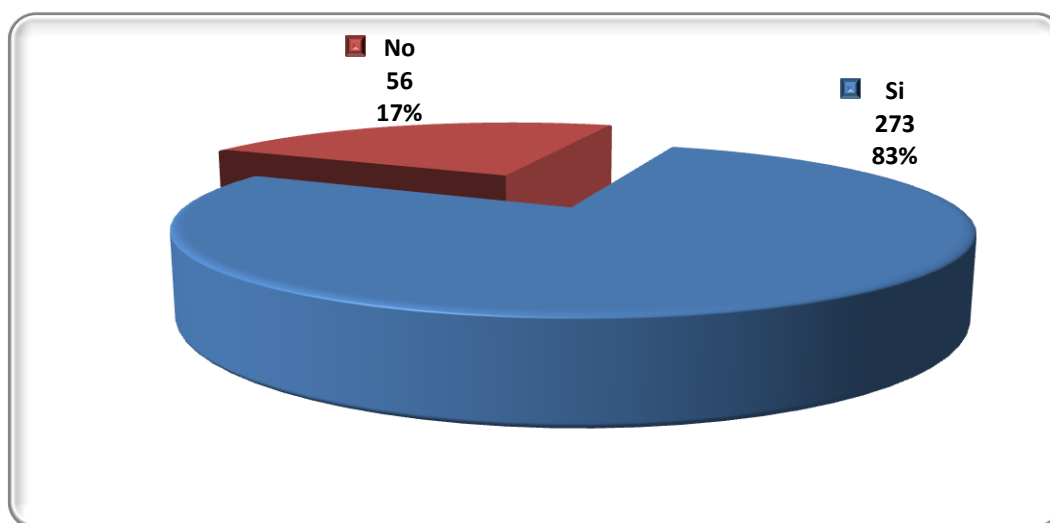


Gráfico A.7.1: Aceptación de los videojuegos en los estudiantes

<sup>118</sup> Esta información fue proporcionada por la Unidad de Estadísticas y Censos del MINED



## ANEXO No. 8: REQUERIMIENTOS DE HARDWARE PARA EL SOFTWARE DE DESARROLLO

Los requisitos de hardware que tienen las aplicaciones seleccionadas para el desarrollo del videojuego son las siguientes:

### **Sistema Operativo:**

Windows XP SP2<sup>119</sup>

| Hardware           | Mínimo                            | Recomendado                         |
|--------------------|-----------------------------------|-------------------------------------|
| Procesador         | Procesador Intel / Athlon 233 MHz | Procesador Pentium 300 MHz          |
| Memoria RAM        | 64 Mb                             | 128 Mb                              |
| Disco Duro         | 2 Gb                              | 5 Gb libres                         |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD                  |
| Pantalla           | SVGA 800 x 600 256 Colores        | SVGA 1024 x 768 32 Bits             |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible      |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles       |
| Conectividad       | Tarjeta de red 10 / 100 Mbps      | Tarjeta de red 10 / 100 / 1000 Mbps |

Cuadro A.8.1: Requerimientos de Windows XP SP2

### **Entorno Integrado de desarrollo:**

Visual C# Express<sup>120</sup>

| Hardware           | Mínimo                            | Recomendado                         |
|--------------------|-----------------------------------|-------------------------------------|
| Procesador         | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1 GHz            |
| Memoria RAM        | 128 Mb                            | 256 Mb                              |
| Disco Duro         | 500 Mb                            | 1.3 Gb libres                       |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD                  |
| Pantalla           | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits             |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible      |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles       |
| Conectividad       | Tarjeta de red 10 / 100 Mbps      | Tarjeta de red 10 / 100 / 1000 Mbps |

Cuadro A.8.2: Requerimientos de Visual C# Express Editions

<sup>119</sup> <http://www.microsoft.com/spain/empresas/productos/windowsxp/requisitos.msp>

<sup>120</sup> <http://www.microsoft.com/spanish/msdn/vstudio/Express/Docs/Systemreqs/Systemreqs.msp>





## Desarrollo de Videojuegos:

### DirectX 9c SDK

| Hardware        | Mínimo   | Recomendado   |
|-----------------|--|---|
| Procesador      | Procesador Intel / Athlon 233 MHz                    | Procesador Pentium 300 MHz                          |
| Memoria RAM     | 64 Mb  | 128 Mb  |
| Disco Duro      | 1 GB   | Mas de 1GB libres                                   |
| Tarjeta Gráfica | Integrada con soporte DirectX 9.0 y Shader Model 1.1 | Soporte a DirectX 9.0 Shader Model 2.0 recomendado) |
| Pantalla        | SVGA 640 x 480 256 Colores                           | SVGA 1024 x 768 32 Bits                             |
| Teclado         | Teclado Microsoft o compatible                       | Teclado Microsoft o compatible                      |
| Ratón           | Ratón Microsoft o compatibles                        | Ratón Microsoft o compatibles                       |

Cuadro A.8.3: Requerimientos de DirectX 9 SD

### XNA 2.0

| Hardware        | Mínimo   | Recomendado   |
|-----------------|--|---|
| Procesador      | Procesador Intel / Athlon 600 MHz                    | Procesador Pentium 1 GHz                            |
| Memoria RAM     | 192 Mb   | 256 Mb o más  |
| Disco Duro      | 20 Mb  | 60 MB libres  |
| Tarjeta Gráfica | Integrada con soporte DirectX 9.0 y Shader Model 1.1 | Soporte a DirectX 9.0 Shader Model 2.0 recomendado) |
| Pantalla        | SVGA 640 x 480 256 Colores                           | SVGA 1024 x 768 32 Bits                             |
| Teclado         | Teclado Microsoft o compatible                       | Teclado Microsoft o compatible                      |
| Ratón           | Ratón Microsoft o compatibles                        | Ratón Microsoft o compatibles                       |

Cuadro A.8.4: Requerimientos de XNA 2.0<sup>121</sup>

## Modelado 3D:

### Milkshape 3D 1.7 <sup>122</sup>

| Hardware    | Mínimo                            | Recomendado                    |
|-------------|-----------------------------------|--------------------------------|
| Procesador  | Procesador Intel / Athlon 500 MHz | Procesador Pentium 1 GHz       |
| Memoria RAM | 128 Mb                            | 256 Mb                         |
| Disco Duro  | 20 Mb                             | 60 MB libres                   |
| Pantalla    | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits        |
| Teclado     | Teclado Microsoft o compatible    | Teclado Microsoft o compatible |
| Ratón       | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |

Cuadro A.8.5: Requerimientos de Milkshape 3D 1.7

<sup>121</sup> <http://www.microsoft.com/spanish/msdn/latam/estudiantes/articulos/nota3.aspx>

<sup>122</sup> <http://chumbalum.swissquake.ch/>

fragMOTION 0.8.7<sup>123</sup>.

| Hardware    | Mínimo                            | Recomendado                    |
|-------------|-----------------------------------|--------------------------------|
| Procesador  | Procesador Intel / Athlon 500 MHz | Procesador Pentium 1 GHz       |
| Memoria RAM | 128 Mb                            | 256 Mb                         |
| Disco Duro  | 11 Mb                             | 20 MB libres                   |
| Pantalla    | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits        |
| Teclado     | Teclado Microsoft o compatible    | Teclado Microsoft o compatible |
| Ratón       | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |

Cuadro A.8.6: Requerimientos de Fragmotion 0.8

Photoshop Cs2<sup>124</sup>

| Hardware           | Mínimo                            | Recomendado                    |
|--------------------|-----------------------------------|--------------------------------|
| Procesador         | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1 GHz       |
| Memoria RAM        | 384 Mb                            | 512 Mb                         |
| Disco Duro         | 250 Mb                            | 650 MB libres                  |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD             |
| Pantalla           | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits        |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |

Cuadro A.8.7: Requerimientos de Photoshop CS2

**Utilitarios:**

## Word 2007

| Hardware           | Mínimo                            | Recomendado                                |
|--------------------|-----------------------------------|--|
| Procesador         | Procesador Intel / Athlon 500 MHz | Procesador Superior Intel / Athlon 500 MHz |
| Memoria RAM        | 256 Mb                            | Mayor de 512 Mb                            |
| Disco Duro         | 1.5 Mb                            | Mayor de 1.5 Mb                            |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD                         |
| Pantalla           | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits                    |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible             |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles              |

Cuadro A.8.8: Requerimientos: Microsoft Word 2007<sup>125</sup><sup>123</sup> <http://www.fragmosoft.com/fragMOTION/><sup>124</sup> <http://www.adobe.com/products/photoshop/photoshop/systemreqs/><sup>125</sup> Requerimientos solo de paquete Word 2007 de la suite office, <http://office.microsoft.com/en-us/products/HA101668651033.aspx>



## Poseidón 3.0 Community Edition para UML

| Hardware           | Mínimo                            | Recomendado                    |
|--------------------|-----------------------------------|--------------------------------|
| Procesador         | Procesador Intel / Athlon 600 MHz | Procesador Pentium 1 GHz       |
| Memoria RAM        | 256 Mb                            | 512 Mb                         |
| Disco Duro         | 51Mb                              | 650 MB libres                  |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD             |
| Pantalla           | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits        |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |

Cuadro A.8.9: Requerimientos Poseidón 3.0 CE para UML<sup>126</sup>

---

<sup>126</sup> Ayuda Poseidón 3.0, <http://www.gentleware.com/products.html>



## ANEXO No. 9: REQUERIMIENTOS DE HARDWARE PARA LA OPERACIÓN DEL VIDEOJUEGO

Estos se determinan en base a las aplicaciones siguientes:

### Service Pack 2<sup>127</sup>

| Hardware           | Mínimo                            | Recomendado                    |
|--------------------|-----------------------------------|--------------------------------|
| Procesador         | Procesador Intel / Athlon 233 MHz | Procesador Pentium 300 MHz     |
| memoria            | 64 Mb                             | 128 Mb                         |
| Disco Duro         | 395 MB                            | 770 MB                         |
| Unidad de CD o DVD | Unidad CD o DVD                   | Unidad CD-RW / DVD             |
| Pantalla           | SVGA 640 x 480 256 Colores        | SVGA 1024 x 768 32 Bits        |
| Teclado            | Teclado Microsoft o compatible    | Teclado Microsoft o compatible |
| Ratón              | Ratón Microsoft o compatibles     | Ratón Microsoft o compatibles  |

Cuadro A.9.1: Requerimientos SP2

### Microsoft XNA Framework Redistributable 2.0<sup>128</sup>

| Hardware        | Mínimo   | Recomendado   |
|-----------------|--|---|
| Disco Duro      | 26.1 Mb  | Mayor de 26.1 MB libres                             |
| Tarjeta gráfica | Integrada con soporte DirectX 9.0 y Shader Model 1.1 | Soporte a DirectX 9.0 Shader Model 2.0 recomendado) |

Cuadro A.9.2: Requerimientos XNA redistribuible

### DirectX End-User Runtimes<sup>129</sup>

| Hardware        | Mínimo   | Recomendado   |
|-----------------|--|---|
| Disco Duro      | 61.2 Mb  | Mayor de 61.2 MB libres                             |
| Tarjeta gráfica | Integrada con soporte DirectX 9.0 y Shader Model 1.1 | Soporte a DirectX 9.0 Shader Model 2.0 recomendado) |
| Pantalla        | SVGA 640 x 480 256 Colores                           | SVGA 1024 x 768 32 Bits                             |

Cuadro A.9.3: Requerimientos DirectX Runtime

<sup>127</sup> <http://support.microsoft.com/kb/837783>

<sup>128</sup> <http://www.microsoft.com/downloads/details.aspx?FamilyID=15fb9169-4a25-4dca-bf40-9c497568f102&displaylang=en>

<sup>129</sup> <http://www.microsoft.com/downloads/details.aspx?familyid=9226A611-62FE-4F61-ABA1-914185249413&displaylang=es>



## ANEXO No. 10: PLANEACIÓN DE RECURSOS

### COSTO DE DESARROLLO DE SIEMPOSAL

Los recursos a utilizar en el proyecto desarrollado denominado Software Interactivo para la Educación Medioambiental de la Población Salvadoreña SIEMPOSAL, se han clasificado en seis diferentes categorías: recurso humano, tecnológicos, operación, consumibles, otros gastos e imprevistos.

#### RECURSO HUMANO

Por la naturaleza del proyecto desarrollado y basados en datos reales, publicados en Game Developer magazine<sup>130</sup> se tiene que el sueldo promedio anual de un desarrollador de videojuegos es de \$73,600. El sueldo de un Ingeniero en Sistemas como Productor/Director de un proyecto de esta naturaleza es de \$78,716 Anual en promedio.

Por lo cual el costo de recurso humano para el proyecto desarrollado se muestra en el siguiente cuadro.

| Puesto                | Personas | Sueldo Anual (\$) | Costo por hora | Horas al mes | Meses | Total(\$)  |
|-----------------------|----------|-------------------|----------------|--------------|-------|------------|
| Director del Proyecto | 1        | 78,716.00         | 41.00          | 20           | 8     | 6,559.67   |
| Desarrolladores       | 4        | 73,600.00         | 38.33          | 160          | 8     | 196,266.67 |
| Total                 |          |                   |                |              |       | 202,826.34 |

Cuadro A.10.1: Salarios del recurso humano desarrollador de videojuegos.

#### RECURSO TECNOLÓGICO

Las computadoras utilizadas para el desarrollo del proyecto al momento de iniciar este contaban con un año de vida, por lo cual, para realizar su valoración se aplicó un ajuste del 80% a su precio de compra<sup>131</sup>, como se muestra en el siguiente cuadro.

<sup>130</sup> <http://www.gameinfowire.com/news.asp?nid=12010>

<sup>131</sup> Según artículo 30-A de la Ley de Impuesto Sobre la Renta.



| Computadoras | Valor de compra (\$) | Ajuste de valor (80%) | Años de uso | Valor Actual (\$) |
|--------------|----------------------|-----------------------|-------------|-------------------|
| Computador A | 1000.00              | 800                   | 1           | 800.00            |
| Computador B | 1400.00              | 1120                  | 1           | 1120.00           |
| Computador C | 700.00               | 560                   | 1           | 560.00            |
| Computador D | 1000.00              | 800                   | 1           | 800.00            |
| <b>Total</b> |                      |                       |             | 3,280.00          |

Cuadro A.10.2: Valor del equipo de computo al iniciar el proyecto.

El detalle del recursos tecnológico utilizado en este proyecto se presentan en el cuadro A.10.3 y cuadro A.10.4.

| Recurso Tecnológico                                       |          |                     |           |
|---|----------|---------------------|-----------|
| Equipo  | Cantidad | Costo Unitario (\$) | Total(\$) |
| <b>Hardware</b>   |          |                     |           |
| Computadoras A  | 1        | 800.00              | 800.00    |
| Computadoras B  | 1        | 1120.00             | 1120.00   |
| Computadoras C  | 1        | 560.00              | 560.00    |
| Computadoras D  | 1        | 800.00              | 800.00    |
| Reguladores   | 4        | 35.00               | 140.00    |
| Cámara Fotográfica  | 2        | 200.00              | 400.00    |
| Control Xbox 360 para PC                                  | 1        | 65.00               | 65.00     |
| Instalación de Red (cable UTP, 1 Switch)                  | 1        | 100.00              | 100.00    |
| Impresoras  | 2        | 42.4                | 84.80     |
| Depreciación del hardware por uso (33.33%) <sup>132</sup> | -        |                     | 1,356.46  |
| <b>Total</b>  |          |                     | 5,426.26  |

Cuadro A.10.3: Costo del recurso tecnológico, hardware

<sup>132</sup> El Porcentaje aplicado anualmente al equipo informático en concepto de depreciación es del 50% según Art. 30 de La Ley de Impuesto Sobre la Renta, en 8 meses la depreciación del hardware es de 33.36%



| Recurso Tecnológico                              |          |                       |                 |
|--|----------|-----------------------|-----------------|
| Software   | Cantidad | Costo Unitario (\$)   | Total(\$)       |
| <b>Software</b>                                  |          |                       |                 |
| Microsoft Office 2007                            | 1        | 109.59                | 109.59          |
| Microsoft. Project 2007                          | 1        | 163.91                | 163.91          |
| Microsoft Visio                                  | 1        | 139.99                | 139.99          |
| Herramientas de modelado 3d                      | 1        | 200.00                | 200.00          |
| Amortización del software (16.67) <sup>133</sup> |          |                       | 102.27          |
| <b>Internet</b>                                  |          |                       |                 |
| Contrato por un año 1024 Kbps                    | 1        | 528.84 <sup>134</sup> | 528.84          |
| <b>Total</b>                                     |          |                       | <b>1,244.60</b> |

Cuadro A.10.4: Costo del recurso tecnológico, software

## RECURSOS DE OPERACION

| Servicios Básicos             |                    |       |                 |
|-------------------------------|--------------------|-------|-----------------|
| Servicio                      | Costo \$ (mensual) | Meses | Total (\$)      |
| Agua                          | 10.00              | 8     | 80.00           |
| Energía eléctrica             | 30.00              | 8     | 240.00          |
| Teléfono                      | 35.00              | 8     | 280.00          |
| Alquiler del local de trabajo | 100.00             | 8     | 800.00          |
| <b>Total</b>                  |                    |       | <b>1,400.00</b> |

Cuadro A.10.5: Costo de servicios básicos

## RECURSOS CONSUMIBLES

El recurso consumible se refiere a todos aquellos insumos consumibles que se utilizaron para la preparación, elaboración y entrega de documentos a lo largo del proyecto.

<sup>133</sup> El artículo 30-A de la Ley de Impuesto Sobre la Renta, establece que la amortización del software es del 25% anual, equivalente a 2.1% mensual, en concepto de uso del Software, en 8 meses la amortización del software es de 16.67% .

<sup>134</sup> Contrato por un año con tarifa de \$44.07 mensual.



## Papelería

| Documentación                              | Cantidad | Cantidad de páginas | Total de páginas |
|--|----------|---------------------|------------------|
| <b>PERFIL DEL PROYECTO</b>                 |          |                     | 50               |
| Documento principal                        | 3        | 10                  |                  |
| Borrador                                   | 2        | 10                  |                  |
| <b>PLAN DE TRABAJO</b>                     | 6        | 8                   | 48               |
| <b>ANTEPROYECTO</b>                        |          |                     | 700              |
| Documento principal                        | 3        | 175                 |                  |
| Borrador de etapa                          | 1        | 175                 |                  |
| <b>ANÁLISIS DE REQUERIMIENTOS Y DISEÑO</b> |          |                     | 1,304            |
| Documento principal                        | 3        | 326                 |                  |
| Borrador de etapa                          | 1        | 326                 |                  |
| <b>CODIFICACIÓN Y DOCUMENTACIÓN</b>        |          |                     | 364              |
| Documento principal                        | 2        | 42                  |                  |
| Manual del Programador                     | 2        | 52                  |                  |
| Manual de Instalación                      | 2        | 8                   |                  |
| Manual del Usuario                         | 2        | 19                  |                  |
| Plan de Implementación                     | 2        | 9                   |                  |
| Borrador de etapa                          | 2        | 52                  |                  |
| <b>DOCUMENTO FINAL</b>                     |          |                     | 800              |
| Documento principal                        | 1        | 400                 |                  |
| Borrador de etapa                          | 1        | 400                 |                  |
| Totales                                    | 32       |                     | 3,266            |

Cuadro A.10.6: Detalle de papelería a utilizada para documentar el proyecto

De acuerdo al cuadro A.10.6, se imprimieron 32 documentos haciendo un total de 3,266 páginas. Cada resma de papel contiene 500 páginas y su costo actual en el mercado es \$ 5.00.

Cantidad de resmas = cantidad de página

500

Cantidad de resmas =  $\frac{3266}{500} = 6.532 \cong 7$  resmas

500

| Cantidad (Resmas) | Costo Unitario (\$) | Costo Total |
|-------------------|---------------------|-------------|
| 7                 | \$ 5.00             | \$ 35       |

Cuadro A.10.7: Costo de papelería.





## Fotocopias

A continuación se detallan la cantidad de fotocopias que se realizaron para cada una de las etapas del proyecto.

| Actividades                         | Número de Copias | Cantidad de páginas | Cantidad de copias | Costo por copia (\$) | Costo Total de copias (\$) |
|-------------------------------------|------------------|---------------------|--------------------|----------------------|----------------------------|
| Perfil del Proyecto                 | 6                | 10                  | 60                 | 0.02                 | 1.20                       |
| Plan de trabajo                     | 6                | 8                   | 48                 | 0.02                 | 0.96                       |
| Anteproyecto                        | 2                | 175                 | 350                | 0.02                 | 7.00                       |
| Análisis de Requerimientos y Diseño | 1                | 326                 | 326                | 0.02                 | 6.52                       |
| Codificación Y Documentación        | 1                | 130                 | 130                | 0.02                 | 2.60                       |
| Documento Final                     | 5                | 400                 | 2000               | 0.02                 | 40.00                      |
| Totales                             | 21               | 1049                | 2914               |                      | 58.28                      |

Cuadro A.10.8: Costo de fotocopias

## Costo de encuadernado de los entregables del proyecto

| Actividades                         | Cantidad | Tipo      | Precio Unitario (\$) | Costo Total de anillados (\$) |
|-------------------------------------|----------|-----------|----------------------|-------------------------------|
| Perfil del Proyecto                 | 12       | Folder    | 0.12                 | 1.44                          |
| Anteproyecto                        | 4        | Anillado  | 2.50                 | 10.00                         |
| Borrador de etapa                   | 1        | Anillado  | 2.50                 |                               |
| Análisis de Requerimientos y Diseño | 3        | Anillado  | 4.00                 | 8.00                          |
| Borrador de etapa                   | 1        | Anillado  | 4.00                 | 4.00                          |
| Codificación Y Documentación        | 8        | Anillado  | 1.30                 | 7.80                          |
| Borrador de etapa                   | 6        | Anillado  | 1.10                 | 6.60                          |
| Documento Final                     | 6        | Empastado | 10.00                | 60.00                         |
| Totales                             | 41       |           |                      | 97.84                         |

Cuadro A.10.9 : Costos de Anillados y empastado

## Impresión y almacenamiento

La cantidad de cartuchos de tinta a utilizar en el proyecto, se ha calculado en base al número de páginas impresas que son 3,266 (ver cuadro A.10.6) y la capacidad de imprimir 100 páginas por cartucho de tinta, en promedio, en la impresora IP 1800, tomando en consideración impresiones en calidad normal, y en un 80% full color.



$$\text{Número de cartuchos} = \frac{\text{No. de páginas impresas}}{\text{No. impresiones por cartucho}}$$

$$\text{Número de cartuchos} = \frac{3,266}{100} = 32.66 \cong 33 \text{ cartuchos de tinta}$$

Se utilizaron 16 cartuchos de tinta negra PG-30 y 17 cartuchos de tinta de color CL 31 para el impresor IP 1800, a continuación se presenta el detalle.

| Artículo                 | Cantidad | Precio Unitario(\$) | Costo Total (\$) |
|--------------------------|----------|---------------------|------------------|
| Cartuchos de tinta Negro | 16       | 20.00               | 320.00           |
| Cartuchos de tinta Color | 17       | 25.00               | 425.00           |
| CD en Blanco             | 6        | 0.50                | 3.00             |
| DVD en Blanco            | 15       | 0.80                | 12.00            |
| Total                    | 54       |                     | 760.00           |

Cuadro A.10.10: Costos de artículos consumibles

| Resumen costos de recursos consumibles |                  |
|--|------------------|
| Rubro                                  | Sub-Totales (\$) |
| Papelería (resmas)                     | 35.00            |
| Fotocopias                             | 58.28            |
| Anillados y empastados                 | 97.84            |
| Impresión y medios de almacenamiento   | 760.00           |
| Total                                  | 951.12           |

Cuadro A.10.11: Resumen costo de recursos consumibles

### OTROS GASTOS - Costo de Transporte

| Medio de Transporte | Costo por reunión (\$) | Reuniones al Mes | Meses Proyecto | Número de Personas | Costo Total (\$) |
|---------------------|------------------------|------------------|----------------|--------------------|------------------|
| Autobús             | 1.5                    | 16               | 8              | 4                  | 768.00           |
| Gasolina            | 1.0                    | 4                | 8              | 4                  | 128.00           |
| Presentaciones      | 18                     | -                | 8              | -                  | 54.00            |
| Total               |                        |                  |                |                    | 950.00           |

Cuadro A.10.12: Costo de transporte



## IMPREVISTOS

Los gastos de imprevistos se han determinado de manera directa y se ha asignado el 10% del valor total de los gastos del proyecto. Este monto es una previsión para cubrir gastos por actividades no estimadas o de emergencia en el proyecto.

## COSTOS TOTALES DE DESARROLLO.

| <b>Costo total del proyecto</b>   |                   |
|-----------------------------------|-------------------|
| <b>Rubro</b>                      | <b>Valor (\$)</b> |
| Recurso humano                    | 202,826.34        |
| Recurso tecnológico               | 6,670.86          |
| Recursos de Operación             | 1,400.00          |
| Recursos consumibles              | 951.12            |
| Otros Gastos                      | 950.00            |
| <b>Subtotal</b>                   | <b>212,798.32</b> |
| Imprevistos (10%)                 | 21,279.83         |
| <b>Costo total del desarrollo</b> | <b>234,078.15</b> |

Cuadro A.10.13: Costos totales de desarrollo del proyecto SIEMPOSAL



## **ANEXO No. 11: CAPACITACIÓN Y DISTRIBUCIÓN DIRECTA DEL VIDEOJUEGO A TRAVÉS DE CDS (NO INCLUYE MANUAL DE USUARIO IMPRESO)**

En la distribución directa, se está considerando que se hará la compra, grabación y distribución de un CD a cada Centro Escolar que cuente con aula informática, siendo estos un total de 2,141 Centros Escolares de nuestro país<sup>135</sup>, como también se está considerando que el Ministerio de Medio Ambiente y Recursos Naturales, realizará una convocatoria a los profesores de las aulas informáticas, para distribuir y capacitarlos en el uso del videojuego, en un periodo de 1 semana, siendo estos asistidos por 4 capacitadores con un costo de \$ 60.00 por día, cada uno.

Esta capacitación, se realizará por zonas o rutas: occidental, central y oriental. En el siguiente cuadro se presenta el detalle de dichos costos.

| <b>Costo de capacitación y distribución directa</b>   |                   |
|---|-------------------|
| <b>Rubro</b>  | <b>Valor (\$)</b> |
| Precio de adquisición de 2,141 CDs:   | 214.10            |
| Precio de Grabación de CDs ( a \$0.30 por unidad X 2,141 unidades)                                      | 642.30            |
| Precio de 2,141 cajas de distribución ( a \$0.20 por unidad X 2,141 unidades)                           | 428.20            |
| Capacitación y Distribución del software a los Ministerio del Medio Ambiente y Recursos Naturales MARN. | 1200.00           |
| Costo por combustible   | 1000,00           |
| <b>Total</b>  | <b>2,484.60</b>   |

Cuadro A.11.1: Costos de implantación del plan piloto de SIEMPOSAL

<sup>135</sup> En base a los datos publicados en la *Memoria de labores, 2006-2007 del Plan Nacional de Educación 2021*, del Ministerio de Educación.