

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERA DE SISTEMAS INFORMATICOS



**SOFTWARE PEDAGÓGICO PARA LA ENSEÑANZA DEL
IDIOMA INGLÉS A ESTUDIANTES NO VIDENTES DE LA
UNIVERSIDAD DE EL SALVADOR**

PRESENTADO POR:

**VÍCTOR HUGO BENÍTEZ ALEMÁN
JORGE ALEXÁNDER CORDERO MANCÍA
FARID ANTONIO PÉREZ ALDANA
CARLOS ENRIQUE VÁSQUEZ RODRIGUEZ**

PARA OPTAR AL TITULO DE:

INGENIERO DE SISTEMAS INFORMATICOS

CIUDAD UNIVERSITARIA, AGOSTO DE 2008

UNIVERSIDAD DE EL SALVADOR

RECTOR:

MSc. RUFINO ANTONIO QUEZADA SÁNCHEZ

SECRETARIO GENERAL:

LIC. DOUGLAS VLADIMIR ALFARO CHÁVEZ

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO:

ING. MARIO ROBERTO NIETO LOVO

SECRETARIO:

ING. OSCAR EDUARDO MARROQUÍN HERNÁNDEZ

ESCUELA DE INGENIERIA DE SISTEMAS INFORMATICOS

DIRECTOR:

ING. CARLOS ERNESTO GARCIA GARCIA

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA DE SISTEMAS INFORMATICOS

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO DE SISTEMAS INFORMATICOS

Título:

**SOFTWARE PEDAGÓGICO PARA LA ENSEÑANZA DEL
IDIOMA INGLÉS A ESTUDIANTES NO VIDENTES DE LA
UNIVERSIDAD DE EL SALVADOR**

Presentado por:

**VÍCTOR HUGO BENÍTEZ ALEMÁN
JORGE ALEXÁNDER CORDERO MANCÍA
FARID ANTONIO PÉREZ ALDANA
CARLOS ENRIQUE VÁSQUEZ RODRIGUEZ**

Trabajo de Graduación aprobado por:

Docente Director:

ING. JULIO ALBERTO PORTILLO

San Salvador, Agosto 2008

Trabajo de Graduación Aprobado por:

Docente Director:

ING. JULIO ALBERTO PORTILLO

Agradecimientos

Antes que nada agradezco a Dios todopoderoso y a nuestra madre la virgen María, por haberme permitido culminar este trabajo de graduación, por haber estado siempre a mi lado en la carrera iluminándome y ayudándome a vencer las barreras que se pusieron en mi camino, para poder completar con éxito este triunfo tan grande en mi vida.

Agradezco infinitamente a mis padres por quienes soy lo que soy y quienes nunca negaron una palabra de aliento y tuvieron una paciencia excepcional, a pesar de todas las situaciones que se pudieron suscitar ellos siempre estuvieron ahí, a mi lado apoyándome siempre y alentándome a superar todas las dificultades que se pusieran en frente, tanto en la carrera como en mi vida personal, gracias por amarme y apoyarme siempre y en todo momento, los amo mucho.

Agradezco especialmente a Carmen Beatriz Escalante Vásquez quien ha sido la pareja más incondicional que un hombre hubiera deseado, por saber darme la seguridad que necesitaba en el momento indicado, por su comprensión, por estar conmigo siempre a lo largo de la carrera, durante este proceso y en mi vida, hoy y para siempre, Gracias por tu Amor.

A mis amigos y compañeros de tesis, Víctor Hugo Benítez, Carlos Vásquez y Farid Aldana por la dedicación y entrega que se vio reflejada en este producto que entregamos, gracias por el esfuerzo, por aguantarnos y soportarnos mutuamente en las buenas y en las malas, gracias por las largas noches en vela y también por las noches de dormir “5 minutos”, estoy seguro que sin alguno de ustedes esto no hubiera sido posible, También un agradecimiento enorme a sus familias por habernos tenido paciencia y habernos apoyado tanto económicamente como moralmente y por habernos recibido en sus hogares, que Dios les bendiga.

A mi hermano quien siempre ha sido un pilar fundamental en mi vida y una fuente de apoyo incondicional, gracias por toda la ayuda que me brindó así como su familia, a mi abuela porque todos sus consejos fueron bien recibidos y aplicados, a mi muy querida Tía Carmen Barahona porque ha sido como un ángel enviado por Dios así como mis demás familiares tanto en el país como en el exterior, tías, tíos, primas, primos, etc. etc., gracias infinitas porque solo Dios sabe que sin su apoyo nada de esto hubiera podido llevarse a cabo.

A mis compañeros y amigos que siempre nos brindaron su apoyo y con quienes comparto este logro: Gerson Claros, Jonathan Hernández, Jacqueline de Montiel, Víctor Ostorga, Axa Araujo, Salvador Cabrera, Cristina Cuellar, Otoniel Valdés, y a los demás que se me escapan no por olvido sino por espacio, gracias infinitas por colaborar con este triunfo, ustedes saben que representan una parte muy importante en mi vida universitaria.

Jorge Alexander Cordero Mancía.

Agradecimientos

A Dios, por haber permitido terminar mi carrera, y estar en los momentos más difíciles conmigo, por darme vida para lograr este objetivo.

A mis padres, por ser un pilar importantísimo en mi vida y en todo lo que he hecho durante toda mi vida.

A mi familia, por haberme apoyado con lo necesario para poder lograr terminar este proyecto, por tener la paciencia requerida en toda la carrera y hasta ahora poder ver los frutos de esa inversión.

A mi esposa, Yendri por comprenderme durante el tiempo que dedicaba a este proyecto y por estar a mi lado.

A mis compañeros de tesis, Jorge, Carlos y Farid, por haber demostrado la capacidad adquirida en todos los años de estudio, por los valores de compañerismo lealtad y lucha con que afrontaron este reto, por ser mis amigos y colegas desde que los conozco.

A mis compañeros y amigos, Gerson Claros, Jonathan Hernández, Victor Ostorga, por ser parte de este esfuerzo y haberme apoyado durante la carrera, y a todos mis demás compañeros con los que hice grupo en las diferentes actividades realizadas.

Víctor Hugo Benítez Alemán.

Agradecimientos

Agradezco a Dios por permitirme terminar con bien mi tesis, por darme la tranquilidad y la fuerza para continuar aun cuando las dificultades nos embargaban.

Gracias Señor por estar siempre a mi lado, te agradezco todo lo que haces por mi y mis seres queridos.

También quiero agradecer a mi familia, a mis padres en especial por ser un ejemplo siempre para mi, porque yo no sería nada como ser humano y como persona sin su correcta guía, los amo y les agradezco su apoyo incondicional aun cuando mis decisiones les afectaron, son los padres que todo hijo desea tener, a mi abuelita por aconsejarme y ser fuente de inspiración, mi madrecita bella que Dios te bendiga siempre.

A mi novia Alexia por estar siempre conmigo en las buenas y en las malas, por demostrarme que a pesar de las dificultades el amor y la comprensión siempre nos brindarán una luz en la oscuridad, por todo su apoyo y su paciencia y, sobre todo, por su amor.

Eres lo mas bello que me ha pasado en la vida.

A mis compañeros de tesis, a Víctor Hugo, a Carlos, a Jorge de quienes aprendí muchas cosas en este proceso, a todos ellos gracias por estar unidos hasta el final como amigos a pesar de tantos problemas y desacuerdos se que puedo contar con ellos. Nunca podría haber llegado a este punto sin ustedes, mi grupo de tesis.

A mis compañeros de la universidad, a mis amigos, que durante todos estos años me permitieron aprender de ellos y enseñarles humildemente lo que pude; a Carlos Pacheco, Danilo, Roberto Stanley, que desde mis años de colegio me han apoyado, a Jonathan, Víctor Ostorga, Gerson, Ledwin, José, Vanessa, Ana... a todos los que me brindaron su amistad durante todos estos años.

A Kirlian, mi gran amigo, que Dios te tenga en su presencia, por ayudarme siempre y ser un ejemplo de lo que un hombre debe ser, no solo con las personas conocidas, sino con cualquiera que necesitará ayuda, por enseñarme tantas cosas de la vida.

A las familias de mis compañeros que nos acogieron como si fuéramos parte de ella. A la familia de mi novia, a todos GRACIAS.

Farid Antonio Pérez Aldana.

INDICE

INTRODUCCION.....	I
ALCANCES Y LIMITACIONES.....	II
IMPORTANCIA DEL PROYECTO.....	III
JUSTIFICACIÓN DEL PROYECTO	IV
Capítulo I: Estudio preliminar	1
1. Antecedentes	2
1.1. Definición de discapacidad en El Salvador.....	2
1.2. Definición de Ceguera.....	2
1.3. Definición de Rehabilitación en El Salvador	2
1.4. Órganos estatales y políticas públicas	3
1.5. Organizaciones públicas, privadas y no gubernamentales de asistencia a no videntes	4
1.6. Estadísticas de Población Discapacitada en El Salvador.	4
1.7. Situación General de las Personas No Videntes	6
1.8. La Evolución del Software Educativo	9
1.9. Software para No Videntes	11
1.10. Software para Aprendizaje del Idioma Ingles	14
1.11. Teclado convencional versus Teclado Braille.....	17
2. Planteamiento del problema	18
2.1. Definición del Problema	18
2.2. Análisis del problema	18
2.3. Formulación del problema.....	22
3. Factibilidad del proyecto.....	22
3.1. Factibilidad técnica.....	22
3.2. Factibilidad operativa	23
Capítulo II: Situación actual	25
1. El Enfoque de Sistemas	26
1.2. Definición	26
1.3. Metodología para la elaboración del enfoque de sistemas	26
1.3. Diagrama de enfoque de sistemas.....	28
1.4. Descripción del Enfoque de Sistemas	29
2. Diagramas de flujo de datos	31
2.1. Definición	31
2.2. Metodología para la elaboración del diagrama de flujo de datos	31
2.3. Modelo de Contexto	34
2.4 Diagrama de nivel 0	35
2.4. Diagramas de nivel 1	36
3. Modelado de Procesos	41
2.1 Definición	41
3.2 Metodología para la Especificación Estructurada de Procesos EEP	43
3.3 Diagramas de Modelos de Procesos.....	45
Capítulo III: Determinación de los requerimientos	52
Requerimientos.....	53
1. Definición	53
2. Metodología para la obtención y determinación de los requerimientos.....	53
3. Requerimientos Funcionales	56
3.1 Especificación de Usuario	56
3.2 Requerimientos clasificados por usuario	57
3.3 Especificación de Sistema	59
4. Requerimientos de Información	60

5. Requerimientos No Funcionales	61
5.1 Generales	61
5.2 Requerimientos Operativos.....	61
5.3 Requerimientos de Desarrollo.....	63
Capítulo IV: Análisis del Sistema.....	68
1. Casos de Uso	69
1.1 Definición	69
1.2 Metodología para la elaboración de los casos de uso	70
2. Diagramas de Casos de Uso.....	73
3. Actores del sistema	78
4. Subsistemas.....	79
4.1 Casos de Uso del Subsistema de Lecciones (A)	80
Capítulo V: Diseño.....	81
1. Arquitectura del software.....	82
1.1. Subsistema de lecciones.....	82
1.2. Subsistema de ejercicios.....	84
1.3. Subsistema de Evaluaciones	85
1.4. Subsistema de Control.....	86
2. Estándares	88
2.1. Estándares de Diseño.....	88
2.2. Estándares de Codificación.....	89
2.3. Estándar de documentación Interna.....	91
2.4. Estándares de Interfaz Gráfica.....	94
2.5. Estándares de Interacción.....	94
3. Diccionario de datos (DD):	100
3.1. Definición	100
3.2. Objetivo.....	100
3.3. Metodología	100
3.3. Diccionario de Datos para SPEI3	102
4. Diagrama Entidad Relación (DER).....	106
4.1. Definición	106
4.2. Metodología para la elaboración del DER	107
4.3. Diagrama Entidad/ Relación.....	109
5. Modelo físico de la base de datos	110
5.1. Definición	110
5.2. Metodología para la elaboración del modelo físico de la BD	110
5.3. Modelo físico de la base de datos	111
Matriz de relaciones entre las tablas de la Base de Datos	113
Documentación de las tablas del modelo físico.....	114
6. Casos de Uso Reales.....	119
6.1. Definición	119
6.2. Metodología para la Elaboración de Casos de Uso Reales.....	120
6.3. Casos de Uso Reales del Subsistema de Lecciones (A).....	122
7. Diagramas de colaboración	124
7.1. Definición	124
7.2. Metodología para la elaboración de Diagramas de Colaboración.....	127
7.3. Diagramas de colaboración.....	128
8. Diagrama de clases	129
8.1. Definición	129
8.2. Metodología para la elaboración del diagrama de clases	129
8.3. Diagrama de clases	133

8.4 Documentación de clases	134
Capítulo VI: Desarrollo	137
1. Ámbito de programación.....	138
2. Condiciones tecnológicas.....	138
3. Configuración del ambiente de desarrollo.....	138
4. Realización de pruebas	138
5. Técnicas de programación.....	139
Capítulo VII: Documentación externa del software	140
1. Manual del usuario	141
2. Manual técnico.....	142
3. Manual de Instalación.....	143
CONCLUSIONES.....	144
Glosario de términos	145
BIBLOGRAFIA.....	146
Anexo 1 – Entrevistas con Mtl. Edgar Nicolás Ayala	147
Anexo 2 – Entrevista con Armando Madrid, Estudiante no vidente de la UES	150
Anexo 3 – Experiencias Estudiantes no videntes de la UES utilizando el SPEI3.	151

INTRODUCCION

Actualmente el conocer diferentes idiomas se ha vuelto una parte muy importante en la vida de toda persona que tiene entre sus objetivos incorporarse en el mercado laboral y en el ámbito profesional. Aun así, en El Salvador no se practica una total equiparación de oportunidades tanto laborales como educativas entre las personas sin importar si tienen características especiales o no.

En El Salvador son muchas las barreras que se le presentan a una persona con discapacidad que pretende incorporarse al ámbito laboral. Esto se debe a que son catalogados como personas poco productiva. Generalmente las empresas de nuestra sociedad brindan empleo a una persona discapacitada como acto de proyección social o caridad, algunas veces se ven obligados por la ley de equiparación de oportunidades.

Con el afán de sentar un precedente y romper este esquema de prejuicio ante el discapacitado, nace una idea en la Universidad de El salvador a través de la Escuela de Ingeniería de Sistemas Informáticos, de desarrollar un trabajo de graduación en beneficio de este sector tan vulnerable. Se relacionó la idea de apoyar a un sector discapacitado con la importancia que tiene el idioma ingles en el ámbito laboral nacional y se pensó en la forma de facilitar la enseñanza del mismo a personas faltas de visión. Esta idea se ha fundamentado a tal grado de definir el trabajo de graduación como “Software pedagógico para la enseñanza del idioma Inglés a estudiantes no videntes de la Universidad de El Salvador” - “SPEI3”, el cual se presenta en este documento.

SPEI3 es un software orientado a apoyar el proceso enseñanza – aprendizaje del idioma ingles a personas invidentes. Este software le permite al docente: Crear contenidos, ejercicios y evaluaciones en base a su propia pedagogía. Y organizarlos en base al programa de estudio que se esté desarrollando. Al mismo tiempo organizar y controlar el avance de los alumnos a su cargo. SPEI3 le permite al estudiante recibir y repasar las clases que su docente le prepare, reforzar y practicar los conocimientos adquiridos por medio de ejercicios y evaluaciones. Todo esto a través de una interfaz amigable y fácilmente comprensible.

Este documento lleva consigo un CD interactivo donde encontrará un detalle específico de la información que se necesite ampliar, para acceder a las diferentes opciones del CD adjunto haga clic en el archivo menu.exe que se encuentra en el directorio raíz del CD.

ALCANCES Y LIMITACIONES

Alcance

- El software incluirá metodologías de estudio adaptadas para personas no videntes brindándoles la oportunidad de aprender el idioma inglés en el nivel básico intensivo en una forma interactiva.
- Las entradas al software de parte del no vidente se realizarán mediante un teclado convencional. Las salidas del software hacia el ciego serán mediante dispositivos de audio sin incluir salidas impresas.
- El software brindará apoyo pedagógico a los docentes para la aplicación de metodologías de enseñanza adaptadas a personas no videntes.
- Software estará orientado a reforzar y mejorar la comprensión auditiva y el habla del idioma inglés no así la escritura.

Limitaciones

Las personas no videntes no están familiarizadas con la utilización de un software pedagógico debido a la escasez de estos en su medio.

La metodología de enseñanza actual esta orientada a personas sin ninguna discapacidad, es decir no contempla su aplicación a personas no videntes. Al no existir dicha metodología, se tiene que adaptar la existente, de modo tal que la discapacidad no afecte el proceso de aprendizaje de las personas ciegas.

La tecnología necesaria para desarrollar este tipo de software enfocado a personas con alguna disminución en sus capacidades visuales no está desarrollada actualmente en nuestro país, por lo tanto no se dispone del soporte necesario que respalde dicha tecnología.

Hay que reconocer que el gobierno muestra poco interés por ayudar a personas con discapacidad, muestra de esto es el bajo porcentaje del presupuesto que es encaminado para este propósito, el cual se reduce aun mas cuando nos enfocamos en enseñanza a personas no videntes para que desarrollen otros idiomas como el inglés, a pesar de lo necesario que es en la actualidad de nuestro país.

IMPORTANCIA DEL PROYECTO

El hecho de que las personas no videntes viven en un mundo diferente al de los videntes, en el cual los sonidos y el tacto se convierten en sentidos importantísimos en su vida, la motivación de todo ser humano se crea a partir de incentivos, significaría darles un incentivo para desarrollarse como personas, enseñarles un nuevo lenguaje es vital ya que de esta forma se multiplican las oportunidades laborales para ellos y se pueden desenvolver en una mayor variedad de trabajos en los cuales pueden obtener una mejor remuneración.

La importancia de este proyecto radica en el beneficio que conlleva la realización de un software pedagógico que este directamente interrelacionado con las características de la población a ser beneficiada, que en este caso es la población de estudiantes no videntes en nivel universitario, quienes representan una porción tan sensible de la sociedad pero a su vez tan importante ya que en ella se puede considerar una mano de obra calificada que actualmente no se esta aprovechando en su totalidad.

JUSTIFICACIÓN DEL PROYECTO

En la actualidad predomina el prejuicio de que una persona invidente es incapaz de manipular una computadora debido a las limitantes que le conlleva su incapacidad para recibir información por medio de la vista, esta idea es errónea ya que actualmente existe la comunicación entre un computador y un invidente, esta comunicación se realiza a través del teclado y salidas de audio, ejemplo de ello son los software utilizados por El Centro de Rehabilitación para Ciegos Eugenia de Dueñas, en esta institución se les enseña a las personas no videntes a adaptarse a la computadora por medio del teclado y ellos reciben información a través de sonidos emitidos por las computadoras mediante un software especializado.

En dicho centro se imparten clases a personas no videntes a través de un programa de enseñanza de un año para adultos a través del computador¹. Este centro de rehabilitación es una de las principales instituciones que atienden a personas no videntes en El Salvador preparándolos para su inserción a la educación regular.

Los costos de enseñanza personalizada para no videntes son altos si se toma en cuenta que los ingresos para esta población son generalmente escasos y que se tendría que pagar un profesor particular cuyo costo mensual aproximado por su servicios oscila entre los \$400 y \$600 por mes², en el Centro de Rehabilitación para Ciegos Eugenia de Dueñas las personas no videntes reciben este servicio de forma gratuita por ser instituto nacional.

Además de contar con el interés de esta institución, se realizaron investigaciones en la Universidad de El Salvador (UES), específicamente en el Departamento de Idiomas de la Facultad de Ciencias y Humanidades, quienes tienen alumnos videntes y no videntes los cuales reciben la formación académica indistintamente. Es en este punto donde surge la problemática ya que cuando uno o varios alumnos no pueden ver, conlleva para el maestro una gran responsabilidad, y debe ingeniárselas para transmitir sus clases a todos sus estudiantes independientemente si son videntes o no.

Dicho departamento cuenta con un proyecto especial de atención a estudiantes que tienen como objetivo reforzar las clases ya recibidas a cualquier tipo de alumno, sin tomar en cuenta si tiene alguna discapacidad o no, es a estos programas donde asisten las personas no videntes, aprovechando las ventajas de dicho proyecto, por tal razón se denotó la necesidad de una enseñanza diferenciada debido a la afluencia de estas personas a las clases de refuerzo. Este es un problema al que se espera dar solución mediante un software que brinde apoyo pedagógico a los docentes en la enseñanza del idioma inglés a alumnos no videntes, el cual facilite la enseñanza y permita utilizar la metodología adecuada.

En el Departamento de Idiomas existen las condiciones adecuadas para la utilización del software ya que cuentan con un centro de cómputo para ello y están dispuestos a proporcionar información de los contenidos programáticos que son impartidos en el curso de inglés básico intensivo, con el fin de adecuarlos a través del software para que la discapacidad de estos alumnos no interfiera con su proceso de aprendizaje. Actualmente ninguna institución del país cuenta con un software para la enseñanza del idioma inglés a

1 Según entrevista realizada a la coordinadora de maestros de dicha institución

2 Según información recabada a través de un maestro que trabaja en forma personalizada y cobra \$10 o \$15 por hora durante 2 horas diarias 5 días a la semana

personas no videntes, aun cuando la asignatura de inglés es exigida por el MINED y es una habilidad esencial para desenvolverse en el ambiente laboral, se estima que el software propuesto favorecerá al 100% de la población no vidente estudiante del Departamento de Idiomas de la Facultad de Humanidades.

Para identificar la población potencial beneficiada por el proyecto se utilizaron estadísticas nacionales obtenidas por la DIGESTYC³ en las cuales se observa en el país hay 99,958 personas que tienen algún tipo de discapacidad, esta cifra representa el 1.51 % de la población nacional total, de esta cifra se desprende que 14,072 personas discapacitadas son no videntes y representan el 11.53% del total de discapacitados, los cuales representan la población potencial beneficiada por este software.

El software propuesto favorecerá a toda persona no vidente que tenga acceso a una computadora (con equipo multimedia) mediante la enseñanza de los fundamentos del idioma inglés para que así dicha persona tenga mayor oportunidad de incorporarse a la sociedad productiva de nuestro país al poder competir contra las demás personas que si pueden ver.

Por lo anterior cabe señalar que la utilización de este software contribuirá en mejorar la atención a alumnos no videntes, ya que representará una herramienta muy útil para dar apoyo pedagógico en la enseñanza del idioma inglés a personas no videntes.

3 Resultados del "Módulo Suplementario a la Encuesta de Hogares de Propósitos Múltiples sobre personas con discapacidad".

Fuente: Encuesta de Hogares de Propósitos Múltiples EHPM-ELS-DIGESTYC.

Capítulo I: Estudio preliminar

1. Antecedentes

1.1. Definición de discapacidad en El Salvador

Toda restricción o ausencia, debida a una deficiencia, de la capacidad de realizar una actividad en la forma o dentro del margen que se considere normal para un ser humano⁴.

Restricción permanente o transitoria de una función psicológica, fisiológica o anatómica como resultado de una deficiencia orgánica⁵.

La ausencia, restricción o pérdida de la habilidad para desarrollar una actividad en la forma o dentro del margen considerado como normal para un ser humano⁶.

1.2. Definición de Ceguera

Según la Organización Mundial de la Salud (OMS), ceguera es aquella visión menor de 20/400 ó 0.05, considerando siempre el mejor ojo y con la mejor corrección. La baja visión es una visión insuficiente, aun con los mejores lentes Correctivos, para realizar una tarea deseada. Desde el punto de vista funcional, pueden considerarse como personas con baja visión a aquellas que poseen un resto visual suficiente para ver la luz, orientarse por ella y emplearla con propósitos funcionales.

La ceguera puede ser total o parcial; existen varios tipos de ceguera parcial dependiendo del grado y tipo de pérdida de visión, como la visión reducida, el escotoma, la ceguera parcial (de un ojo) o el daltonismo.

Aunque la ceguera y la baja visión pueden ser provocadas por algún accidente, también existen numerosas enfermedades que las desencadenan: Catarata, glaucoma, leucoma corneales, retinopatía diabética, retinopatía del prematuro, catarata y glaucoma congénitas, atrofia óptica, distrofia retinal y retinosis pigmentaria, entre otras.

1.3. Definición de Rehabilitación en El Salvador

La legislación de El Salvador aborda el servicio de rehabilitación con enfoque multidisciplinario y comunitario, señalando que las instituciones del Estado deben conformar equipos de profesionales que aseguren una atención multidisciplinaria para cada persona según lo precise y que garanticen su integración socio-comunitaria⁷.

La rehabilitación se puede definir como:

Proceso continuo y coordinado, de duración limitada, que permite que una persona con deficiencia alcance un nivel físico, mental y/o social funcional óptimo, a fin de proporcionarle los medios para modificar su vida. Las medidas pueden estar encaminadas a compensar la pérdida de una función, o limitación funcional o a facilitarles ajustes o reajustes sociales.

La rehabilitación se puede entender como el conjunto de medidas encaminadas a mejorar la capacidad de una persona para realizar actividades necesarias para su desempeño físico, mental, social, ocupacional y económico, que le permitan integrarse a su ambiente sociocultural⁸.

4 Ley de Equiparación de Oportunidades para las personas con discapacidad

5 www.conaipd.gob.sv/glosario

6 Guías de Atención de los Servicios de Rehabilitación, ISRI

7 Ley de Equiparación de Oportunidades para las personas con discapacidad

8 ídem 2

La rehabilitación, a partir de las definiciones anteriores, incorpora elementos que aplicados podrían lograr incidir en una verdadera participación de personas con discapacidad en los diferentes ámbitos de la sociedad; no obstante, en la práctica predomina la visión médica, tanto es así que en los diferentes servicios se les llama “pacientes” a las personas con discapacidad usuarias de los servicios de rehabilitación.

En general la rehabilitación se limita a terapias, desarrollo de potencialidades físicas, dotación de ayudas técnicas, enseñanza de habilidades y destrezas en actividades convencionales y promoción laboral a partir de la sensibilización.

La rehabilitación en El Salvador está dividida en dos grandes aportes: el gubernamental y el de beneficencia. Este último funciona mediante la recolección de fondos a través de “Teletones”, administradas por la fundación FUNTER.

1.4. Órganos estatales y políticas públicas

Con respecto al papel del Estado en la rehabilitación de personas con discapacidad, es El Consejo Nacional de Atención Integral a la Persona con Discapacidad (CONAIPD) creado por el decreto Ejecutivo No. 111, quien es el ente rector de la Política Nacional de Atención Integral a las personas con discapacidad y coordinador de las acciones desarrolladas por los diversos sectores en beneficio de las personas con discapacidad⁹. Las instituciones de rehabilitación deben formular sus planes de conformidad a la Política Nacional de Atención Integral que establezca el Consejo.

El CONAIPD trabaja con base en las siguientes comisiones:

Comisión de Inserción Laboral

Su objetivo es “Velar por el cumplimiento del marco institucional, legal y estratégico en el sector trabajo a fin de incrementar en forma progresiva y sostenible las oportunidades de formación e inserción laboral de las personas con discapacidad”.

Comisión de Divulgación

Su objetivo es velar por la permanente promoción del cambio de conocimientos, actitudes y valores en la sociedad sobre la persona con discapacidad y los retos institucionales, técnicos, sociales y financieros de la equiparación de oportunidades.

Comisión de Salud

Su objetivo es sentar las bases para el cumplimiento de la ley de Equiparación de Oportunidades para las personas con discapacidad, en lo relacionado a la rehabilitación en salud de esta población, mediante el montaje de un sistema integrado de atención posesionando al CONAIPD como el ente rector de la política de discapacidad en El Salvador.

Comisión de Educación

Su objetivo es garantizar el cumplimiento del marco institucional, legal estratégico que conlleve al incremento progresivo y sostenible de la capacidad y cobertura del sistema educativo nacional para la atención de la diversidad y la integración al sistema educativo formal de las personas con discapacidad.

9 Art.. 43, 36 y 9 de la Constitución de la República.

Comisión de Integración Comunitaria y Vida Autónoma

Su objetivo es garantizar en forma progresiva y sostenible la generación de entornos accesibles, vida autónoma y comunitaria para las personas con discapacidad, en los sectores públicos privados y comunitarios.

Políticas públicas en discapacidad

Para el logro de la atención integral de la salud, el estado salvadoreño se obliga a impulsar acciones encaminadas a prevención, detección precoz, diagnóstico oportuno e intervención temprana de discapacidades¹⁰. Dispone, asimismo, que las instituciones de rehabilitación deben formular sus planes de conformidad con la Política Nacional de Atención Integral que establezca el Consejo. Toda institución que inicie un determinado proceso de rehabilitación, debe coordinarse con otras entidades afines que desarrollen programas de seguimiento en servicios de menor complejidad, cercanos al domicilio de los usuarios o en planes de hogar que complementen su esfuerzo¹¹. El Ministerio de trabajo de El Salvador (MINTRAB) se encuentra actualmente en el proceso de elaboración y aprobación de una política nacional de empleo.

1.5. Organizaciones públicas, privadas y no gubernamentales de asistencia a no videntes

ASCES (Asociación de Ciegos de El Salvador)

Brinda capacitaciones en informática a personas no videntes. Cuentan con una imprenta braille.

ANCIESAL (Asociación Nacional de Ciegos Salvadoreños)

Brinda capacitaciones, apoyo jurídico y promueve la inserción laboral, entre otros.

ACOPASANTAL (Asociación Cooperativa)

Está conformada por personas no videntes que trabajan el área de colchonería y desarrollan capacitaciones productivas y de incidencia.

Ministerio de Trabajo (MINTRAB)

Aunque dentro de su estructura orgánica no tiene establecido el trabajo hacia las personas con discapacidad, a través de la Dirección General de Empleo y Previsión Social, destinan gestores a la búsqueda y colocación de oportunidades laborales y de formación a personas con discapacidad.

Ministerio de Educación

Cuenta dentro de su estructura orgánica con una División de Necesidades Educativas Especiales, quienes son responsables a nivel nacional de la educación para personas con discapacidad sensorial e intelectual.

1.6. Estadísticas de Población Discapacitada en El Salvador.

En El Salvador, se cuenta con estudios serios sobre la estadística de personas con discapacidad. No obstante, oficialmente se continúa manejando el dato de la OMS, que sugiere que el 10% de población tiene algún tipo de discapacidad.

¹⁰ Art. 8 de la Constitución de la República.

¹¹ Art. 9 y 10 de la Constitución de la República.

Según el Censo de Población proporcionado por la Dirección General de Estadística y Censo (DIGESTYC), 81.721 personas tienen discapacidad (1.6 % de la población total), de las cuales el 53.3% son hombres y el 46.7% mujeres¹².

Otras investigaciones reflejan el 6.56 % de prevalencia¹³ de discapacidad.

Una investigación realizada por DIGESTYC refleja que hay 99,958 personas con algún tipo de discapacidad, lo que supone una proporción de 1.51% de habitantes con alguna discapacidad¹⁴, el detalle de los datos se pueden ver en la tabla 1-1.

Por su parte, el RNPN¹⁵ registra, 153.583 personas con discapacidad, lo que representaría el 4.1 % de prevalencia¹⁶. Cabe aclarar que el RNPN registra únicamente a personas mayores de 18 años.

La base de datos del CONAIPD registra, en junio de 2005, 17,725 personas con discapacidad. Vale mencionar que esta base de datos se alimenta a través de diferentes iniciativas propias y de otros organismos que hacen levantamiento de información y posteriormente la comparten al CONAIPD. También se registran personas con discapacidad que se acercan al Consejo por razones individuales. Por tanto, no puede hacerse una proyección de prevalencia, pues el sistema no garantiza un registro de la población total del país por estar en constante alimentación y no prevé la depuración por fallecimiento. Dada la brecha que existe entre cada una de las investigaciones y el vacío en cuanto a un dato o cifra oficial por parte del ente rector, para efecto de este apartado se utilizará el dato de la OMS del 10% de personas con discapacidad.

Grupo de discapacidad	Personas con discapacidad	Tasa por 1000 habitantes de 6 a 64 años de edad
Ver	14,072	2.65
Oír	11,282	2.12
Hablar	19,682	3.70
Movilidad	25,874	4.86
Destreza	16,451	3.09
Amputación	5,579	1.05
Intelectuales	19,127	3.60
Psicológicas	9,497	1.79
Otros	512	0.10

TABLA 1-1 Personas de 6 a 64 años según tipo con discapacidad

nota: una persona puede estar en más de una categoría

Fuente: DIGESTYC - Encuesta de Hogares de Propósitos Múltiples - EHPM - ELS.

12 Resultados Censales sobre la población Discapacitada en El Salvador. DIGESTYC

13 Encuesta de personas con discapacidad, OMS-GTZ-UDB

14 Resultados del Módulo suplementario a la Encuesta de Hogares de Propósitos Múltiples sobre personas con discapacidad. DIGESTYC

15 Datos del Registro Nacional de Personas Naturales, el cual considera solo a personas mayores de 18 años de edad.

16 Para efecto de este análisis, se calculó aproximadamente la población mayor de 18 años, por no existir el registro estadístico exacto.

1.7. Situación General de las Personas No Videntes

Acceso a la Educación

Aunque el Sistema Nacional de Educación no cuenta con una partida o monto presupuestario para servicios de apoyo, se sabe que a partir de enero 2005 el Ministerio de Educación de El Salvador (MINED) entrega a los estudiantes ciegos, integrados a escuelas regulares desde el segundo grado escolar, únicamente regletas Braille. Ningún centro público de estudios tiene bibliografía en Braille.

El sistema nacional de educación cuenta también con instituciones educativas privadas que desarrollan alguna modalidad de atención a la educación especial (como escuelas y colegios que disponen de especialistas en conducta y aprendizaje que dan oportunidades a personas con discapacidad). Su financiamiento es aportado por las instituciones mismas y casi siempre amortizado en los cánones de matrícula y el pago de colegiaturas de los alumnos.

El sistema nacional de educación cuenta con 567 maestros de educación especial, quienes en alguna medida son responsables de las adecuaciones curriculares.

“Las adecuaciones curriculares forman parte importante de lo que nosotros MINED le llamamos Proyecto Curricular de Centro, el cual tiene como objetivo definir desde cada centro educativo la oferta curricular a desarrollar. Esta tiene tres niveles de concreción. El primer nivel es el que se define desde el nivel central, porque está orientado a lo que es la política educativa; el segundo nivel es el que le corresponde al centro educativo y el tercer nivel, el que le corresponde al docente en el aula. Es aquí donde se define que estudiantes requieren una adecuación curricular, sin embargo como está considerado como una estrategia de desarrollo del currículo, realmente no se lleva un registro de este¹⁷”.

Las adecuaciones significativas se registran y se incentiva a los maestros que las desarrollan y las no significativas no se registran. Al final del año lectivo, el MINED y el CONAIPD premian a los centros educativos cuyas adecuaciones curriculares han sido exitosas.

Acceso Al Trabajo

El MINTRAB, ente rector en materia laboral, no cuenta con un sistema de registro que incluya al total de población con discapacidad desempleada económicamente activa, por lo que se infiere a partir de los resultados de la encuesta de Hogares de Propósitos Múltiples, ver tabla 1-2.

Se deduce que a falta del dato estadístico que refleje la población con discapacidad que se encuentra buscando un empleo, resulta difícil calificar la satisfacción de estos resultados.

$$\begin{aligned} \text{Tasa de desempleo: } & 6.8\% \\ & 1,675/27131 = 6.17 \end{aligned}$$

Según los datos proporcionados por el MINTRAB, se ha colocado un 6.17 % de la población con discapacidad en relación con la población en general registrada (27,131) en la bolsa nacional de empleo, durante este periodo.

17 Entrevista realizada a Licda. Ada Montano, División de Necesidades Educativas Especiales, MINED

Situación laboral de las Personas con Discapacidad (PCD) respecto a las Personas Económicamente Activas (PEA)

Población con discapacidad desempleada		
PEA	PEA PCD	PCD Colocadas por MINTRAB
2,710,237 ¹⁸	271,023	1,675

TABLA: 1-2

PCD insertadas por el MINTRAB $1,675 / 271,023 = 0.62 \%^{19}$

Se desconoce el número de personas con discapacidad de este porcentaje que aún conservan el empleo, han ascendido, son reincidentes o están ahora desempleadas.

De acuerdo con la información del MINTRAB, se conoce que a nivel nacional, las personas con discapacidad se colocan en los sectores de: servicios, industria y comercio.

El MINTRAB actúa como intermediario entre la oferta y la demanda de empleo. En caso de que el empresario lo solicite, se hacen sugerencias o recomendaciones para adaptar los puestos de trabajo, pero no se dispone de una asignación específica para la temática.

Las personas con discapacidad insertadas a la fecha en el ámbito laboral, a través del MINTRAB, han iniciado con el salario mínimo. Se desconocen datos de incrementos o nivelaciones, por ser competencia directa entre el empleado y el patrono.

Para efecto de la formación profesional se cuenta con el INSAFORP, que es dependencia del Ministerio de Trabajo. No obstante, no cuenta con una estrategia sistemática para personas con discapacidad, sino que actúa conforme a solicitudes y convenios específicos.

El Salvador no cuenta con incentivos fiscales relacionados a la equiparación de oportunidades de personas con discapacidad en ningún área del quehacer nacional. Por esto, ninguna instancia se ocupa de realizar adaptaciones al puesto de trabajo ni a facilitar recursos ergonómicos que faciliten el desempeño laboral de personas con discapacidad.

Los procesos de selección de personal de las diferentes instituciones públicas carecen de procesos accesibles a personas con discapacidad, en tanto que no cuentan con personal ni recursos adaptados.

Por otra parte la inspección de trabajo no contempla aspectos dirigidos a garantizar la equiparación de oportunidades en el empleo para las personas con discapacidad.

Con la finalidad de promover el empleo, el MINTRAB ha desarrollado veinte jornadas de sensibilización a empleadores, con lo que se les informa acerca de la demanda del

18 Población Económicamente Activa de El Salvador, según EHPM 2003, DIGESTYC

19 Dato acumulativo (desde 2000) brindado por la Dirección General de Previsión Social y Empleo del MINTRAB

mercado laboral de personas con discapacidad. No obstante, no dispone de mecanismos operativos que garanticen la vigencia de la ley, que manda contratar a una persona con discapacidad por cada veinticinco empleados.

Acceso a la Información

La Biblioteca Nacional de San Salvador resulta bastante accesible en su primer nivel del edificio, cuenta con una sala braille, en la que todo el material bibliográfico está en braille y en audio.

El Centro de Información y Documentación en Discapacidad (CIDDI – CONAIPD) cuenta con material en la temática de discapacidad en braille. Las asociaciones de personas ciegas cuentan también con pequeños centros de documentación en braille.

La Casa de la Cultura del Ciego, posee material en braille, brinda el servicio de internet para ciegos y permite imprimir en braille la documentación, es una iniciativa entre CONCULTURA y una asociación de ciegos.

1.7.4 Programas y proyectos del sector público y no gubernamental dirigido a las familias de las personas con discapacidad.

El Centro de Rehabilitación de Ciegos, funciona con la modalidad de internos, medios internos y externos. Tiene una capacidad máxima de atención para 350 alumnos en las diferentes modalidades. Cuenta con 20 profesores, 5 asignados por el MINED y los demás con fondos propios. Las adecuaciones curriculares las realiza el Centro mismo.

Los chicos ingresan desde cero años de edad, reciben terapia de estimulación y todo el proceso que les permita enfrentar y superar los retos y barreras de la sociedad. El área educativa formal la reciben dentro del centro hasta el nivel de segundo grado; luego deben incorporarse y asistir a las escuelas regulares del sistema. Simultáneamente, se les enseñan actividades de la vida diaria, técnicas espaciales y programas vocacionales-productivos.

El presupuesto para funcionamiento está incluido en el general del ISRI, de ahí obtiene otros fondos en concepto de cuotas voluntarias de los jóvenes ciegos. La cuota máxima equivale a nueve dólares mensuales.

Oportunidades de Trabajo

La discapacidad, en general, siempre se aborda con un enfoque médico y asistencialista. En la mayoría de los casos, las asociaciones de personas con discapacidad abordan la definición de discapacidad desde una óptica seria de pleno derecho, reconociendo la diversidad y fomentando la potencialidad.

Por otra parte, el mercado y su visión capitalista, valora a la persona por lo que produce. Este enfoque se encuentra divorciado del abordaje sobre derechos humanos y vida autónoma y fomenta el mito de que las personas con discapacidad son poco o nada productivas.

Además, las oportunidades que se abren en las diferentes esferas en los diversos escenarios son encaminados a actividades de beneficencia distantes de un enfoque de derecho y de participación real de la población con discapacidad en el desarrollo y crecimiento económico del país.

Abordar la situación laboral de personas con discapacidad, resulta complejo y alarmante, intentar enumerar los obstáculos para su colocación implica contemplar diversidad de variables.

Tanto el Ministerio de Trabajo, como algunas ONG's que trabajan en orientación vocacional, inserción laboral y rehabilitación profesional, hacen esfuerzos por colocar a PCD en puestos de trabajo. Las estrategias van desde la gestión directa con los empresarios hasta el desarrollo de Ferias de Trabajo.

Para la gestión directa se invierten recursos humanos y financieros, sin capacidad o herramientas efectivas de hacer cumplir la ley, obteniendo apenas unas pocas oportunidades laborales para auxiliares de limpieza, obreros de maquilas, elevadoristas, recepcionistas y similares, lo que representa uno de los obstáculos institucionalizados.

Debido a que trabajan en base a metas, no responde a los propósitos de la demanda laboral y genera una reiteración de los estereotipos de la persona con discapacidad, considerando que existe un área específica de trabajo a cada discapacidad.

Respecto de las Ferias de Trabajo, resulta más que una expectativa una lotería de sentimientos, en las que personas con discapacidad con grandes esfuerzos de toda índole, acuden con la esperanza de encontrar una fuente de ingresos que les permita sobrevivir dignamente, esperanza que se ve truncada al pasar las hojas del calendario y no recibir noticias de todos los currículos que debió pagar y dejar en el mostrador de los oferentes de empleo. Por otra parte, un buen porcentaje de personas con discapacidad en edad productiva tiene bajo perfil académico y falta de profesión u oficio, elementos que dificultan su colocación en puestos técnicos o de alto nivel.

No obstante, el mayor y principal obstáculo que niega el desarrollo profesional y laboral de personas con discapacidad, es la carencia de un ente rector en discapacidad capaz de hacer cumplir la ley y política que beneficia la participación del colectivo. En tanto, las personas con discapacidad viven excluidas, discriminadas sistemáticamente de los procesos educativos, de salud, de rehabilitación, de formación profesional, de transporte y otros que les permita competir en igualdad de condiciones por un puesto de trabajo.

Concluyendo, los obstáculos vale mencionar la actitud de muchos empresarios, que por desconocimiento del tema, de las potencialidades de la personas y no olvidando su deber ser de empresas en un sistema capitalista, niegan las oportunidades laborales y sobre todo el derecho a un empleo al colectivo.

1.8. La Evolución del Software Educativo

Los adelantos científicos y tecnológicos han afectado significativamente las tradicionales formas de enseñanza-aprendizaje, que se han venido utilizando desde los últimos 30 años en que apareció "El software educativo".

El profesor Peré Marquez de la Universidad de Barcelona, da una preliminar definición de Software educativo, "... como los programas para ordenador creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje."

En ese entonces, se requerían complejas tareas de programación para crear los micromundos y las condiciones específicas en que se desarrollaban los "Tutoriales",

“ejercitadores” y “sistemas expertos”, pues los discos flexibles (5.1/4”) no poseían gran capacidad de almacenamiento, los lenguajes por ser de primera y segunda generación, eran muy rígidos y apenas se estaba empezando a difundir el uso de los llamados PC (Personal Computer), pues su costo era todavía muy elevado pero en la medida en que fueron evolucionando los dispositivos de hardware, se aumentó la capacidad en los discos de almacenamiento y aparecieron los disquetes de 3. 1/2” y los CD, fueron evolucionando los sistemas multimedia en los años 80, que incorporaron imágenes, sonido, animaciones y videos, lo que hizo que se propagara gran cantidad de “juegos”, que aumentaban significativamente la interactividad de los usuarios.

La gran revolución se presenta en los años 90 con el crecimiento significativo del Internet, pues se aumentaron las posibilidades de comunicación entre los usuarios a través del “Chat” y los servicios de “Correo” (e-mail), a costos reducidos y el uso de las posibilidades del “hipertexto”, que prácticamente, permitía enlazar al mundo de forma inmediata.

El poder ampliar la capacidad de comunicación entre los usuarios permitió la llegada de nuevos servicios a través de la Internet, entre ellos la implementación de los primeros “cursos virtuales” que permitieron que un estudiante desde cualquier lugar del mundo con una simple conexión a Internet, de manera asíncrona pudiera capacitarse, con casi todos los beneficios que conlleva la educación tradicional y a unos costos bastante reducidos, fue lo que permitió el desarrollo en el presente siglo de lo que hoy conocemos como “E-learning”.

Entonces, es cuando aparecen los “Manejadores de Contenido” que le permiten al profesor de una manera fácil, editar la información de su curso, definir su configuración, crear listas de estudiantes, sacar calificaciones (en algunos casos de manera automática), llevar registros, crear grupos, sacar copias de seguridad y actualizar su material de manera apropiada cuando lo estime conveniente.

Estos “manejadores”, le permiten trabajar de una manera cómoda, con todo lo que se ha trabajado hasta el momento, como son: “tutoriales”, “Ejercitadores”, “Videos”, “Multimedia interactiva”, “Sistemas expertos”, “Simuladores”, manejo de “Hipertexto”, y “Juegos interactivos”, entre otros.

Los profesores pueden trabajar todo lo anterior, combinándolo con algunas cosas que trabajaba en su clase tradicional, como el manejo de bibliografías, crucigramas, mapas conceptuales, foros, lecturas obligatorias y complementarias y trabajo de grupos.

Súmele a lo anterior las bibliografías de “Websites” especializadas en el manejo de los diferentes temas, el uso apropiado de “buscadores de información” y el trabajo de “blogs”.

Es por eso que el Docente vuelve a retomar la dirección de su curso, implementando diversos tipos de herramientas, que facilitan su labor en el proceso de “Enseñanza-Aprendizaje”, potencializando todos los aspectos pedagógicos, coadyuvando de esta manera con los objetivos trazados para el desarrollo de sus respectivas materias.

La última tendencia que se evidencia, de manera particular desde el año 2005, es la denominada de “Software Social”, el cual acompañado de los conceptos de software LIBRE y contenidos ABIERTOS, están implicando una revolución sin antecedentes, pues ha incentivado los trabajos de grupo, de comunidad y por otro lado, el más

importante, quizá, ha dado el acceso a un mayor número de gente, sin costo alguno que aprovecha de manera ética el trabajo realizado.

Desde luego, lo anterior se ha facilitado por los avances en la tecnología, la reducción de costos, el incremento significativo de usuarios del Internet, las nuevas y fáciles formas de programación, las políticas de los gobiernos para que en las escuelas desde temprana edad, los alumnos se familiaricen con el uso de las computadoras y el patrocinio de entidades como la Comunidad Económica Europea que subvenciona con grandes cantidades de dinero, la creación de programas, cuyo software, sea libre.

El desarrollo tecnológico sigue avanzando a pasos agigantados y no sabemos a donde va a parar:

Hoy en día ya nos estamos familiarizando con nuevas categorías como “wikis”, “Blogs” y otras “herramientas de aprendizaje colaborativo”;

En los medios ya contamos con poderosos dispositivos de almacenamiento, como las memorias USB y que decir de los “cast pod”, que son “emisoras de radio virtuales” que funcionan con el Internet y que tanto docentes como estudiantes, pueden implementar sin mayor esfuerzo técnico.

El crecimiento casi incontrolado de redes tanto locales como mundiales, nos ha obligado a participar en ambientes de “trabajo colaborativo”, con todas las repercusiones que ello conlleva.

Entonces ahora, las relaciones profesor – estudiante, se mueven dentro de un modelo de aprendizaje con apoyo virtual, enmarcado en un modelo constructivista.

De lo anterior se puede concluir al menos de manera preliminar, que si quisiéramos caracterizar lo que es software educativo, se entiende como tal, todos los programas, ayudas y medios que se puedan utilizar en un manejador de contenido, para facilitar el proceso enseñanza - aprendizaje.

1.9. Software para No Videntes

En la Actualidad existe un gran número de programas que intentan romper la barrera de las limitaciones a las personas no videntes con respecto de las computadoras, los mas difundidos y populares son los denominados screen readers que, como su nombre lo indica, leen todo lo que ocurre en la pantalla del monitor, estos lectores se apoyan en otro tipo de software que les permite a sus usuarios escuchar lo que esta escrito en la pantalla, es decir, cuentas con voces para interpretar el texto en la pantalla, a este tipo de programa se le llama Sintetizadores de voz, sus motores son utilizados por los screen readers para comunicarse con los usuarios.

Como es natural en el mundo de la informática, algunos de estos programas (tanto screen readers como sintetizadores) son libres y de código abierto, mientras que otros son propietarios y están protegidos por leyes de copyright y su precio depende de la complejidad del sistema.

Con el avance de las plataformas Linux en sus diferentes versiones, el software libre ha aumentado su popularidad y las aplicaciones de accesibilidad para no videntes han recibido un nuevo impulso tanto por los desarrolladores que intentan "democratizar" el acceso a los medios tecnológicos como por la misma sociedad de no videntes; algunos de estos programas han sido concebidos para ser multiplataforma y poseen instalaciones que no resultan complicadas para los usuarios ciegos.

A continuación se presentan algunos de los más populares programas de accesibilidad para personas no videntes y sus principales características.

- **Programas Lectores de Pantalla**

BLINDUX

Facilita la comunicación, la investigación, el procesamiento y almacenamiento de datos de manera eficaz e independiente. Opera en equipos de bajos requerimientos técnicos, trabaja con aplicaciones en modo de consola (versión texto) y esta disponible para el sistema operativo Linux.

JAWS (Job Access With Speech)

En su versión para el sistema operativo Windows, producto que actualmente fabrica y distribuye la compañía estadounidense Freedom Scientific. JAWS for Windows (JFW) es un potente lector de pantalla que permite a una persona totalmente ciega acceder a los contenidos de la salida visual de un ordenador personal mediante voz y/o el alfabeto Braille.

ORCA

Orca es una herramienta técnica libre de código abierto, flexible, extensible y potente para las personas ciegas y deficientes visuales.

Usando varias combinaciones de voz, braille y magnificación Orca ayuda a proporcionar accesibilidad a las aplicaciones y conjunto de herramientas que soporten AT-SPI (Interfaz de Proveedores de Servicio de Tecnologías de Accesibilidad) como el popular entorno de escritorio para Linux GNOME.

NVDA

Es un lector de pantallas para ciegos o personas con baja visión, realizado en código abierto sin depender de los modos de video de Windows. Usa sus recursos de accesibilidad, como MSAA para comunicar al usuario, por medio de una síntesis compatible con SAPI 5, lo que ocurre en la pantalla.

- **Sintetizadores de voz**

IBM ViaVoice Versión 6.6

Características:

Sistemas operativos: Familia Windows 95/Windows NT 4

Idiomas soportados: 15 (alemán, chino simplificado, danés, español castellano, español mejicano, finés, francés canadiense, francés estándar, inglés americano, inglés británico, italiano, japonés, noruego, portugués brasileño y sueco)

Interfaz de trabajo: Microsoft SAPI 4 (la más usada, compatible con la mayoría de programas de accesibilidad)

Calidad de las voces: buena (se ha dado en considerar que este motor es el que ofrece una voz más clara y comprensible, de ahí que sea el preferido por muchos usuarios y el más usado en el desarrollo de aplicaciones de accesibilidad, aunque preciso es decir que las versiones anteriores a la 5.0 poseían una calidad de voz aún superior)

Entonación de las voces: buena (se realizan las pausas e inflexiones adecuadas según la puntuación o construcción de las oraciones, aunque también en este caso era mejor la entonación del motor perteneciente a versiones precedentes a la 5.0).

FreeTTS

Creado por Sun Microsystems, ofrece una buena calidad de voz, la documentación es buena y existen foros de discusión acerca del desarrollo basado en este sintetizador.

No existen voces en español para este sintetizador, aunque en la documentación se afirma que las voces creadas mediante el FestVox (una aplicación del sintetizador de voz Festival) pueden ser portadas a este sintetizador, no se explica cuál es el procedimiento para realizar esta interacción. Está escrito totalmente en Java.

Para su funcionamiento requiere que el equipo tenga instalado el software de Java: J2SDK, que se obtiene gratuitamente de Internet.

Para su óptimo funcionamiento requiere mínimo de una máquina con procesador Intel Pentium 296 con MMX y 128 Mb de memoria Ram. El licenciamiento de este programa se acoge a los términos de la GPL y su código es de fuente abierta.

FESTIVAL

Creado por la Universidad de Edimburgo, ofrece una buena calidad de voz y cuenta con herramientas como el FestVox para la creación de nuevas voces.

Tiene voces en varios idiomas, incluyendo el español; la documentación es abundante y se actualiza frecuentemente. Está escrito en lenguaje C y Scheme y tiene funcionalidades para interactuar con las API del sistema Linux; cuenta además con la aplicación Festival_Server, la cual permite utilizar el sintetizador en modo cliente servidor, a través del puerto 1314.

Para obtener un óptimo funcionamiento requiere de una máquina con procesador Intel Pentium 133 con MMX y 64 Mb de memoria Ram.

AT&T Natural Voices text-to-speech

Características:

Sistemas operativos: Familia Windows 95/Windows NT 4, Linux VI/VII

Idiomas soportados: 6 (alemán, español latinoamericano, francés, inglés americano, inglés británico e inglés indio)

Interfaz de trabajo: Microsoft SAPI 4/5 (la más usada, compatible con la mayoría de programas de accesibilidad)

Calidad de las voces: muy buena (es sin lugar a dudas la más lograda de todos los sintetizadores por software existentes en el mercado actual, hasta el punto de que puede tomarse por voz humana natural si no se escucha poniendo especial atención en detectar sus matices)

Entonación de las voces: muy buena (también en este aspecto supera ampliamente a los demás productos de su familia, realizando una locución casi natural y de todo punto sorprendente por tratarse de una voz sintetizada.

MICROSOFT TEXT-TO-SPEECH

Versión 4.0

Características:

Sistemas operativos: Familia Windows 95/Windows NT 4

Idiomas soportados: 1 (inglés americano)

Interfaz de trabajo: Microsoft SAPI 4/5 (la más usada, compatible con la mayoría de programas de accesibilidad)

Calidad de las voces: BUENA (poseen una pronunciación muy elaborada e, incluso las que se han diseñado buscando más el efecto curioso que para una utilidad concreta, se entienden con gran facilidad)

Entonación de las voces: BUENA (es uno de los motores de voz en lengua inglesa que marca mejor la pronunciación y las inflexiones del texto, incluso trabajando con las voces robóticas o con características especiales).

- **Sistemas Operativos**

ORALUX

Es una distribución GNU/Linux para personas ciegas y deficientes visuales basada en Knoppix.

Oralux trabaja en modo texto (no tiene entorno gráfico) y es multilingüe, brinda la posibilidad de trabajar en español, inglés, francés, alemán y algún otro idioma más, proporcionando una síntesis de voz para cada uno de los idiomas mencionados.

No es necesaria la instalación en el disco duro, ya que se trata de un Live CD que se ejecuta nada más insertar el cd y encender el PC. Es accesible desde el primer momento, y nos proporciona un menú muy "amigable" para trabajar con distintas aplicaciones que van desde un editor de textos hasta un navegador web.

Oralux cubre un gran número de "tipos de usuarios" desde aquél que sólo desea desempeñar las tareas básicas en un PC, hasta aquél que desea administrar sistemas o desarrollar. Al estar basada en Knoppix, y ésta en Debian, cuenta con muchas posibilidades y mucho software disponible para Oralux.

Oralux puede ser instalada en el disco duro de manera muy sencilla y accesible respetando la configuración que se tenga seleccionada en ese instante.

Oralux ofrece tres modos de trabajo:

Usando el entorno Emacs con Emacspeak.

Usando el lector de pantalla Yasr.

Usando el lector de pantalla Speakup.

Todos estos entornos trabajando con las voces de Mbrola. Actualmente se está estudiando la posibilidad de en un futuro añadir entorno gráfico y el lector de pantalla Orca.

1.10. Software para Aprendizaje del Idioma Ingles

Antes de presentar software para aprendizaje del idioma ingles, se debe aclarar que en el mercado no existe actualmente un software pedagógico que permita enseñar ingles a personas no videntes, por tanto todos los programas aquí se describen están enfocados para un mercado eminentemente vidente y solo algunos módulos pueden ser aprovechados por los invidentes.

Learning English Teacher's Toolkit

El paquete combina tanto material de evaluación como recursos para la enseñanza.

Para que pueda evaluar y enseñar a estudiantes con diferentes aspiraciones, esta colección incluye tres diferentes tipos de evaluaciones: ESL (inglés como segunda lengua), TOEFL®, y TOEIC®.

El paquete incluye:

Examen de Evaluación de Inglés como Segunda Lengua: Con este examen de evaluación, el profesor puede conocer el nivel de su alumno en tres diferentes áreas de habilidad: gramática, lectura y comprensión auditiva. Cada una de estas áreas se divide en tres niveles: principiante, intermedio y avanzado. El alumno recibe una calificación al

final de cada sección. Se puede usar para establecer un punto de partida y después para constatar el avance del estudiante.

1000 Key English Words & Idioms

En inglés existen muchas palabras y expresiones que nos pueden resultar confusas; lo que se deriva frecuentemente en problemas de comprensión y ortografía. El propósito de este disco compacto, es ayudar al usuario a identificar cuales son las palabras que llevan a cometer errores con más frecuencia.

ESL PRO

Este innovador programa esta basado en los métodos de enseñanza más efectivos. Le ayuda a hablar inglés con claridad y confianza más rápido. Fue diseñado por lingüistas y profesionales de la enseñanza de idiomas.

TOEIC® Advantage 1, 2, & 3

Estos exámenes de práctica, desarrollados por profesores de inglés certificados, incluyen todas las áreas del examen TOEIC®. Los exámenes son prácticamente iguales en contenido y forma al verdadero, y proporcionan valiosa retroalimentación que da una VENTAJA REAL el día del examen. También se pueden usar para evaluar el nivel de inglés de una persona en un contexto de comunicación de negocios a nivel internacional.

TOEFL® Advantage 1 & 2

En este CD vienen dos exámenes TOEFL de práctica, iguales en formato y tipo de preguntas al TOEFL® original. Esta herramienta se puede usar para evaluar el nivel de inglés de una persona y/o prepararse para tres de la áreas del examen TOEFL®.

Combo de Juegos English Practice

Este disco compacto incluye una selección de juegos para practicar gramática, vocabulario, comprensión auditiva y ortografía en inglés. Estos juegos interactivos presentan a una serie de personajes para que los estudiantes del idioma se diviertan mientras refuerzan sus habilidades en las áreas más importantes del idioma. Esta selección de juegos es una herramienta que puede ayudar a personas de cualquier edad a mejorar sus habilidades en inglés.

Learning English Series

Excelente para el estudiante de inglés con un nivel intermedio y avanzado. Esta colección de cuadernos de trabajo (workbooks) "Learning English" ayuda al alumno a dominar los principios esenciales de la gramática, la lectura, la escritura y la comprensión auditiva. Cada cuaderno de trabajo incluye la instrucción de los conceptos clave y ejercicios de práctica con sus respectivas respuestas. Cada cuaderno de trabajo (Workbook) viene en formato PDF, ofreciendo así al estudiante la flexibilidad de trabajar directamente en la computadora o imprimirlo para trabajar con papel y lápiz.

la colección Learning English incluye:

Learning English Grammar Workbook: este cuaderno de trabajo contiene valiosas horas de instrucción y práctica. Te ayuda a construir bases sólidas en esta área tradicionalmente difícil del idioma. Los ejercicios que se incluyen probarán y mejorarán tus habilidades en la comprensión y buen uso de la gramática inglesa.

Learning English Reading Workbook: este cuaderno de trabajo contiene textos de entre 100 y 800 palabras. Los pasajes son similares a los que se encuentran en textos de

universidades en Estados Unidos y Canadá o en la vida cotidiana. Al terminar cada lectura una serie de ejercicios probarán tu nivel de comprensión lectura.

Learning English Writing Workbook: este cuaderno de trabajo se centra en la buena redacción en inglés de las tres diferentes formas de ensayos: expositivo, narrativo y descriptivo. También encontrarás consejos y técnicas de redacción en inglés, así como plantillas para correspondencia diversa.

Learning English Listening Workbook: con este cuaderno de trabajo mejoras tus habilidades de comprensión auditiva en inglés. La comunicación oral se vuelve fácil y divertida. Escucha a personas nativas en el idioma y lee los guiones al mismo tiempo; esto te permite desarrollar tus habilidades y medir tu nivel de comprensión oral en inglés

Hablemos Ingles - Ver 7.0

Este Programa enseña a: hablar, escuchar, leer, escribir.

Características excelentes:

Reconocimiento del habla, grabación de voz, vídeo y audio de hablantes nativos, películas culturales, plan de estudios en línea.

El curso completo tiene más de 100 lecciones que incluyen:

Pronunciación, vocabulario, gramática, lectura, escritura, conversación, práctica de hablar, práctica de escuchar, referencia, reconocimiento del habla, conversaciones simuladas, juegos de práctica, y películas culturales.

Requerimientos de Sistema:

Windows 98, Me, 2000, XP

Pentium 75 MHz or better

16 MB RAM

130 MB free hard disk space

800x600 monitor, 256 colors

Windows-compatible 16-bit sound card & speakers

Microphone

4X CD-ROM drive

1.10.4 Collins Bridge to English for Spanish Speakers

Windows 95/98/Me or NT4/2000/XP

Talk Now!

Esta diseñado para principiantes que desean aprender un idioma nuevo lo mas pronto posible. Talk Now! Presenta lo necesario para que inicie el proceso de aprendizaje. Al escuchar y practicar el habla autentica todos los días, se desarrollará su afinidad natural para el idioma.

En Europa la inmensa mayoría de la gente es bilingüe o trilingüe. The Euro Method™ es una forma fácil y simple que muchos europeos utilizan para aprender a hablar una segunda lengua. Basado en las mismas técnicas utilizadas y probadas que permiten a un niño de cuatro años aprender un idioma antes de saber leer o escribir, el método Instant Immersion English esta garantizado ser el mejor método para aprender ingles. Durante años, los lingüistas han sabido que el entorno de inmersión es absolutamente la manera mas rápida para aprender un segundo idioma; el concepto esta basado en estar rodeados de hablantes nativos y de una nueva cultura.

Instalación: Windows (No requiere instalación). Los discos incluyen una aplicación de arranque automático para que comience a funcionar en cuanto introduzca el disco. Macintosh (No requiere instalación).

1.11. Teclado convencional versus Teclado Braille

El teclado para escribir en braille es único en el mundo y esta diseñado para las personas no videntes con problemas motores en las manos. La Universidad Autónoma de Barcelos y de la ONCE son los creadores del teclado para PC en braille, el cual permite a las personas no videntes con problemas motores en las manos escribir sin mayor dificultad.

Jordi Roig, invidente desde hace unos años es investigador de la UAB, y fue uno de los encargados de presentar el revolucionario teclado, junto a José Fernández Coya director del Centro de Investigación y Desarrollo de Aplicaciones Tiflotécnicas (CIDAT) de la ONCE.

Entre las características que destacan del teclado están: la conexión a la computadora a través de una llave USB, su función es similar a los teclados normales, además el teclado está diseñado para las personas con problemas visuales y motores en las manos.

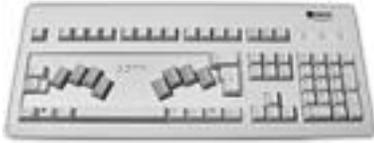
Tipo de Teclado	Teclado Convencional	Teclado Braille
		
Precio	Entre 5 y 30 dólares	Entre 600 y 800 euros
Capacidades para invidentes	No adaptado para no videntes. Deben adaptarse y aprender a utilizarlo	Totalmente adaptado para personas no videntes que conozcan Braille
Conocimiento y uso	Reciben un adiestramiento para que aprendan a usarlo	No reciben adiestramiento para su utilización
Accesibilidad a personas que no conocen Braille	Cuenta con características especiales con teclas específicas como la "F" o la "J" o el numero "5" en e teclado numérico para usarse como referencia conozcan o no Braille.	Solo puede ser utilizado por personas que conozcan el sistema Braille
Disponibilidad de adquisición	Se encuentra disponible de inmediato en cualquier establecimiento de venta de computadoras	No se cuenta con acceso a nivel local a menos que se haga la compra por medio de Internet
Se utilizara en este proyecto	Si, por los factores antes descritos	No, por las causas antes mencionadas

TABLA: 1-3 Comparativa entre el teclado Braille y el Teclado estándar para computadoras.

2. Planteamiento del problema

La educación superior para las personas ciegas en la Universidad de El Salvador (UES) esta limitada por no existir una metodología de enseñanza que contemple las características particulares y generales de la población no vidente que asiste a las clases, esta situación se repite en todas las universidades del país, las personas ciegas que optan por estudiar una carrera universitaria se ven afectadas por muchos factores que le impiden desenvolverse en las clases igual que el resto de estudiante, los estudiantes no videntes optan por acudir a programas de apoyo, que no están diseñados para atenderlos a ellos. Uno de estos programas es el Programa de Atención al Estudiante, este programa tiene como objetivo reforzar las clases ya recibidas a cualquier tipo de alumno, sin tomar en cuenta si tiene alguna discapacidad o no, el problema es que los estudiantes no videntes han hecho suyo este proyecto y utilizan todos los recursos que este les brinda dejando fuera a los otros estudiantes que también lo necesitan, aun con su asistencia a estos cursos el nivel de aprendizaje de las personas no videntes es deficiente.

Mas allá de que los ciegos ocupen o no los programas de apoyo, el principal problema radica en que no se ha considerado separar a estos estudiantes con discapacidad visual del resto de la población estudiantil a fin de que les sea impartida una educación acorde a sus limitaciones.

2.1. Definición del Problema

Dificultad en el proceso de enseñanza del idioma Ingles Básico Intensivo a estudiantes no videntes del Departamento de Idiomas de la Facultad de Humanidades de la Universidad de El Salvador.

2.2. Análisis del problema

Para esta sección se realizo un análisis causa y efecto, basándose en información obtenida en el departamento de idiomas donde se entrevisto al director del departamento de Idiomas (ver anexo 1), varios docentes y a estudiantes invidentes (ver anexo 2), además de observación directa.

- **Factores causales**

Los factores causales más importantes encontrados en la investigación se describen a continuación:

El docente: Es el encargado de practicar la pedagogía más adecuada al tipo de estudiante que tenga en su clase.

Material didáctico: Todos los recursos que sirven como apoyo al docente para elaborar e impartir sus clases.

Infraestructura: Comprende los salones de clase destinados para los estudiantes.

Metodología de enseñanza: Comprende todas las técnicas y habilidades que posee el docente para hacer que sus estudiantes comprendan lo que les imparte.

Infraestructura: Es la infraestructura física, los salones de clase y sus implementos.

Recursos educativos: Toda documentación, conocimientos, bibliografías y tecnologías que puedan ayudar al estudiante a comprender el contenido temático de una asignatura, en este caso, todo el material especializado para personas no videntes con propósito educativo.

- **Causas**

Las causas que abonan a la problemática en el departamento de idiomas de la facultad de humanidades de la UES son identificadas en el diagrama de causa y efecto (figura 1.2)

De este diagrama se infiere que las causas que provocan el problema son:

Por parte de los docentes:

Falta de capacitación: No existen programas de capacitación de docentes con respecto a la enseñanza a personas ciegas.

Inexperiencia: Ninguno de los docentes no están preparados pedagógicamente para atender a los no videntes.

En cuanto a la metodología de la enseñanza:

Poca atención a las características de los no videntes: Cuando se imparte una clase, no se toma en cuenta la ceguera en los métodos de enseñanza que se aplican, las personas no videntes necesitan un tipo de enseñanza más orientada a la audición y el tacto.

Uso excesivo del pizarrón: Se utiliza mucho el pizarrón en apuntes que el maestro considera que debe de tener anotado en su cuaderno el alumno, las personas ciegas no pueden leer del pizarrón.

No se aplica metodología especializada: En la actualidad los métodos utilizados para la enseñanza del idioma inglés, no toman en cuenta pedagogía para personas invidentes, es decir que los videntes y los invidentes utilizan los mismos métodos de aprendizaje.

Por parte de la infraestructura:

No hay señalización adecuada para identificar salones: Los no videntes tienen dificultades en localizar sus salones de clase ya que no existen rótulos con relieve o en Braille en puertas o corredores que les indiquen ruta a seguir o salón de clase.

Baja acústica: Los salones donde reciben clases los estudiantes no videntes necesitan tener mejor acústica ya que para ellos es fundamental el dictado en las lecciones diarias, la baja acústica dificulta seriamente la percepción de las lecciones para las personas ciegas.

Material Didáctico:

Métodos de evaluación no adecuados: Actualmente no se cuenta con exámenes en Braille, el docente tiene que realizar una evaluación personalizada por cada estudiante no vidente, y en caso de que los estudiantes tengan una ceguera parcial el docente opta por hacer un examen especial.

Costo elevado de material especializado: Los materiales que sirven a los docentes como guía o recurso didáctico son relativamente caros, la institución no ha asignado fondos especiales para el material que ayude a preparar lecciones de clase a personas ciegas.

Recursos Educativos:

Escasez de material en braille: En la escuela de Idiomas no se cuenta con material escrito en Braille, los estudiantes no videntes tienen que buscar material de estudio fuera de la universidad, la falta de material es un grave problema que causa dificultades en el aprendizaje.

No se cuenta con material audible: Actualmente no se utiliza material audible de ningún tipo que se le pueda brindar a cada estudiante no vidente como refuerzo a sus clases diarias, el único material audible que se utiliza es el que se emplea para escuchar ejercicios y resolverlos en el workbook; resulta evidente que dicho material no es suficiente para asimilar toda una lección.

Todas estas causas traen como consecuencia el problema de la “dificultad en el proceso de enseñanza del idioma inglés a personas ciegas”.

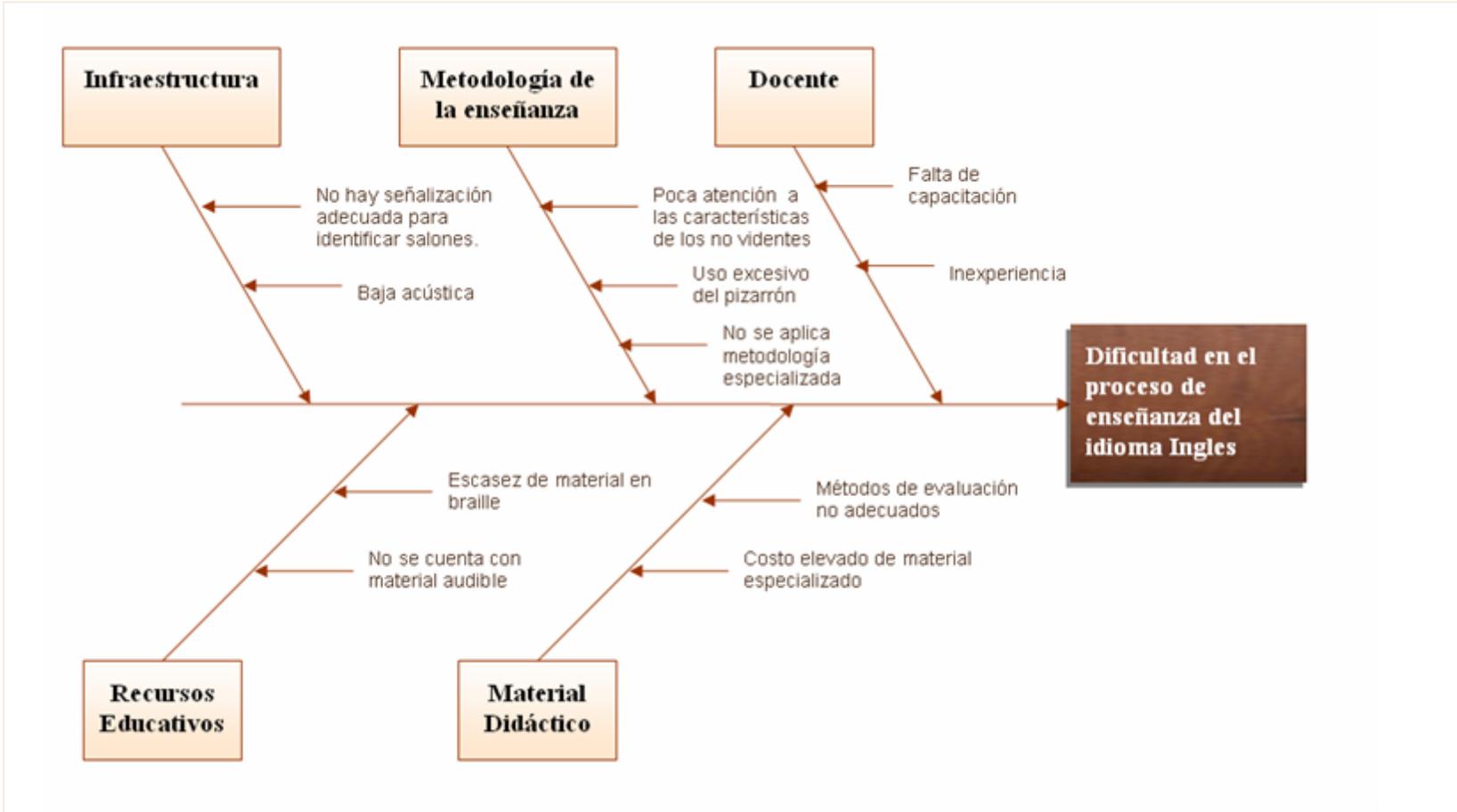


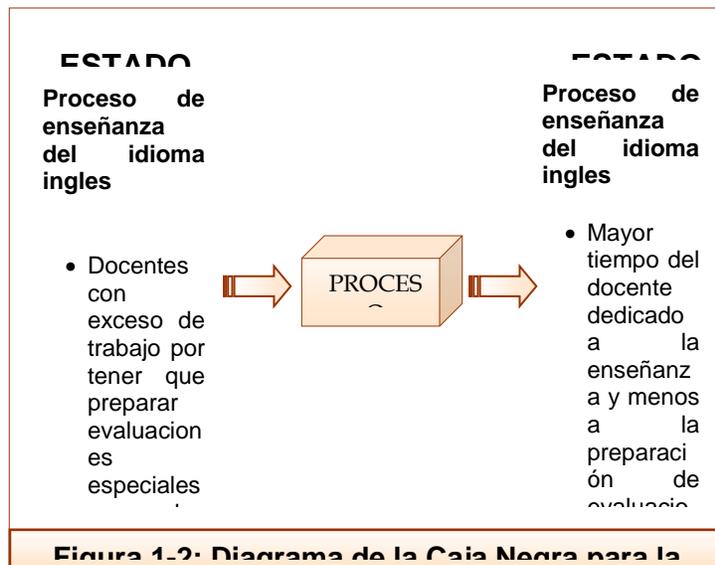
Figura 1-1 Diagrama causa y efecto situación actual

2.3. Formulación del problema

Sabiendo las causas y el problema utilizamos la caja negra para la formulación de nuestro problema estableciendo un estado actual y un estado esperado.

- **El Problema visto como un Sistema**

El siguiente diagrama muestra el proceso solucionador de problemas haciendo uso del método de la caja negra en el cual se considera al problema visto como un estado actual A que entra al proceso solucionador y del cual se espera obtener un estado favorable B.



En la figura 1-2 se observa que el proceso de enseñanza – aprendizaje del idioma inglés para personas invidentes en el estado actual A cuenta con problemas por parte de los docentes, material didáctico, infraestructura, metodología de enseñanza y recursos educativos; por otro lado está el estado deseado B en el cual el docente dedica más tiempo a la enseñanza es decir que cuenta con apoyo del proceso solucionador del problema, material didáctico donde se toman en cuenta la desventaja de la persona no vidente, clases personalizadas que

apoyan a los no videntes como un refuerzo de las clases presenciales, y una metodología adecuada para el no vidente pueda comprender mejor los temas.

3. Factibilidad del proyecto

En este apartado se describen la factibilidad de la creación de un software pedagógico a la enseñanza del idioma inglés para personas no videntes en la Universidad de El Salvador, dado que es un proyecto para beneficiar a una población discapacitada y sin fines de lucro no se evaluará económicamente el proyecto ya que no se cuenta con datos reales de implementaciones similares, por lo tanto no se pueden calcular datos reales de beneficios económicos que proporcionaría el software después de la implementación.

3.1. Factibilidad técnica

Para conocer si contamos con la tecnología necesaria para realizar el proyecto debemos preguntarnos que elementos tecnológicos se necesitan para llevar a cabo el mismo, para la evaluación técnica nos basaremos en tres elementos que son:

1. Herramientas de software
2. Hardware
3. Recurso humano

- **Herramientas de software**

El software necesario para el proyecto lo podemos clasificar en: sistemas operativos, ofimática, herramientas de desarrollo, herramientas de diseño y gestor de bases de datos, para cada uno de ellos existen varios productos, en la tabla 12-1 se muestran alternativas de productos terminados actualmente en el mercado:

Herramienta de software	Producto
Sistemas operativos	Windows XP, Linux, OS
Ofimática	Office, Open office
Herramienta de desarrollo	Visual Basic 6.0, Java, Visual Basic .NET
Herramientas de diseño	Microsoft Visio, IBM Rational
Gestor de base de datos	Oracle, SQL Server, PostgreSQL

TABLA 1-4

Para cada herramienta de software hay más de una alternativa la cual se seleccionara en el posterior análisis de requerimientos.

- **Hardware**

Se disponen de seis computadoras con características similares en cuanto a procesamiento 2 GHz y memoria principal 512 MB, disponen de tarjetas de sonido y de video con memoria solo de video de 128 MB; se dispone de acceso a Internet para todas las computadoras.

- **Recurso humano**

Se cuenta con cuatro egresados de Ingeniería de la Universidad de El Salvador, apto para desarrollo de sistemas informáticos y con capacidad de crear software, ya que cuentan con conocimientos sobre ingeniería de software, y desarrollo de aplicaciones multimedia.

- **Conclusión factibilidad técnica**

De lo anterior se concluye que el proyecto es factible técnicamente ya que se cuenta con la tecnología necesaria (hardware y software) para su desarrollo. Es ineludible identificar que no basta con disponer de la tecnología, sino que es necesario el conocimiento técnico sobre la misma. En el caso del hardware, que consiste en computadoras personales, los desarrolladores del proyecto tienen amplio conocimiento y experiencia en su manejo; y en el caso del software cuentan igualmente con el conocimiento necesario para su correcta aplicación.

3.2. Factibilidad operativa

Los elementos que garantizan el funcionamiento del software a realizar son:

Acceso a la información
Apoyo de la institución
Uso

- **Acceso a la información**

Para el desarrollo del software se necesitara retroalimentación por parte de los usuarios del mismo, debido a la total disponibilidad presentada por el director del departamento de idiomas de la Universidad de El Salvador se cuenta con apoyo por parte de los usuarios.

- **Apoyo de la institución**

La universidad de El Salvador destina fondos para proyectos de este tipo, el departamento de idiomas debe de gestionar recursos para poder implementar el software en el centro de cómputo con que cuentan.

- **Uso**

Se debe garantizar el uso del software terminado, en este caso se cuenta con la disponibilidad de los estudiantes no videntes que necesitan de apoyo para tener mejores condiciones para aprender ingles, el software debe contar con una interfaz de fácil uso y de acuerdo a las condiciones de los usuarios que lo ocupen.

- **Conclusión factibilidad operativa**

Debido a los tres factores mencionados anteriormente podemos concluir que se cuenta con condiciones favorables para que el software funcione, ya que se cuenta con acceso a la información, apoyo de la institución y se garantiza la usabilidad del software

Capítulo II: Situación actual

1. El Enfoque de Sistemas

Como parte de la situación actual en este apartado se presenta, el enfoque de sistema del proceso enseñanza-aprendizaje del idioma inglés a estudiantes no videntes, este enfoque nos ayuda a comprender como funcionan los elementos que intervienen así como también a conocer las entradas, los procesos y las salidas del sistema.

1.2. Definición²⁰

El enfoque de sistemas tiene sus orígenes en un proceso sistémico para la resolución de problemas planteado por Jhon Dewey, profesor de Filosofía de la Universidad de Columbia, quien identificó tres series de juicios que intervienen en la resolución satisfactoria de una controversia:

- Reconocer la controversia
- Sopesar aseveraciones alternativas
- Emitir un juicio

Dewey no utilizó el término enfoque de sistemas, pero reconoció la naturaleza secuencial de la resolución de problemas: Comenzar con el problema, considerar diferentes maneras de resolverlo, y finalmente seleccionar la solución que parezca mejor

1.3. Metodología para la elaboración del enfoque de sistemas

Se define al enfoque de sistemas como una serie de pasos para asegurar que se entienda el problema, que se consideren soluciones alternativas, y que la solución escogida funcione.

Los pasos del enfoque de sistemas ofrecen un criterio para clasificar las múltiples decisiones que es preciso tomar para resolver un solo problema. Aquí se usarán 10 pasos agrupados en tres fases, como se ilustra a continuación:

Fase I: Trabajo de Preparación

Paso 1: Ver el problema como un sistema

Paso 2: Reconocer el sistema ambiental

Paso 3: Identificar los subsistemas del sistema problema

Fase II: Trabajo de definición

Paso 4: Proceder del nivel sistema al nivel de subsistemas

Paso 5: Analizar las partes del sistema en cierto orden

Fase III: Trabajo de Resolución

Paso 6: Identificar soluciones alternativas

Paso 7: Evaluar las soluciones alternativas

Paso 8: Seleccionar la mejor solución

Paso 9: Implementar la solución

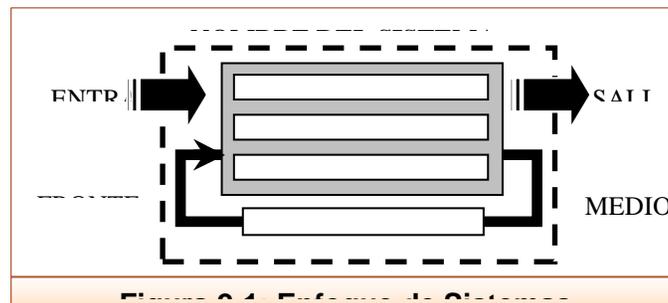
Paso 10: Dar seguimiento para asegurar que la solución sea eficaz

²⁰ McLeod, Raymond Jr., Sistemas de Información Gerencial, 7ª. Edición Prentice Hall.

Aunque las diversas descripciones del enfoque de sistemas siguen el mismo patrón básico, el número de pasos puede variar.

En resumen el enfoque de sistemas es una metodología de resolución de problemas en la cual se ve a la situación problemática como un sistema. En dicho sistema se debe observar sus características, comportamiento y situación medioambiental y definirlos en un nivel de abstracción que ayude a ver el problema de una manera simplificada que pueda ser solucionada mediante la realización de actividades o acciones para corregir dicha situación problemática.

Se realiza un diagrama que representa una visión más compleja del sistema en el cual se incluyen además de las entradas y salidas, los procesos internos así como el recurso necesario (humano y material) para que dichos procesos se lleven a cabo, el control, la frontera y el medio ambiente del sistema. Esta metodología permite describir detalladamente todos los aspectos importantes del problema visto como un sistema.



1.3. Diagrama de enfoque de sistemas

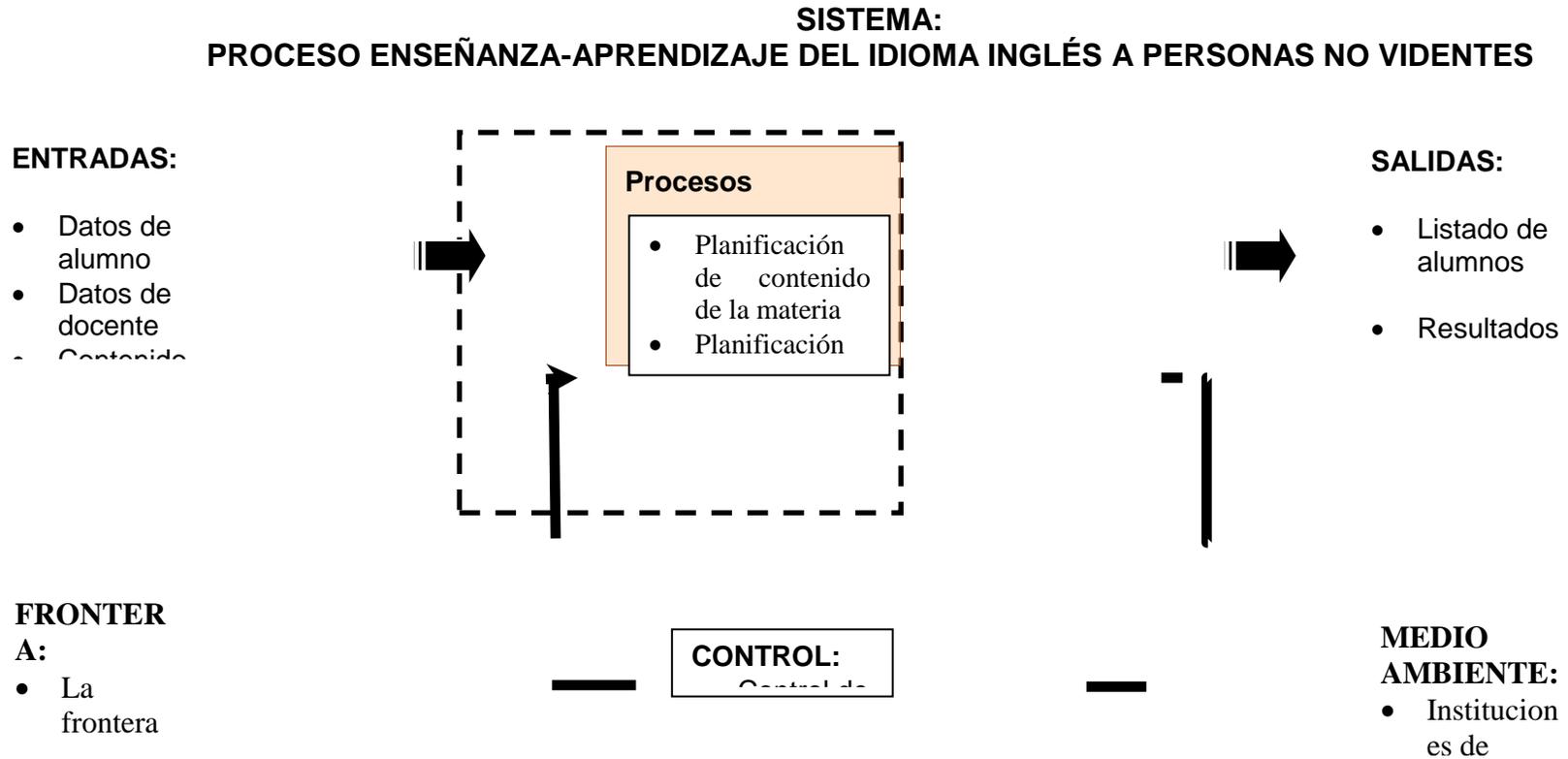


Figura 2-2
Enfoque de

1.4. Descripción del Enfoque de Sistemas

- **Objetivo del sistema**

Trasladar los conocimientos del idioma ingles hacia los alumnos no videntes que estudian en instituciones de educación superior los cuales.

- **Entradas**

Datos de alumno: Son datos personales que identifican al alumno como estudiante de la materia.

Datos de docente: Son datos personales que identifican al docente, el docente es la persona encargada de un grupo de alumnos.

Ejercicios y evaluaciones: Los problemas a resolver por parte de los alumnos son los ejercicios, las evaluaciones son ejercicios con una ponderación.

Libro guía: El libro guía sirve como referencia al estudiante para la lectura y comprensión de temas, además al maestro en la exposición de clase, en el caso de la persona invidente no puede utilizar el libro guía por si solo ya que este no esta diseñado para personas ciegas.

- **Procesos**

Planificación del contenido de la materia: En este proceso se realizan las actividades para organizar los contenidos de la materia por medio de unidades, lecciones y temas.

Planificación de clase: En este proceso se planifica como el maestro impartirá la clase.

Impartir clase: De acuerdo al proceso de planificación de la clase el proceso impartir clase es ejecutar lo planeado y desarrollarlo con los alumnos.

Realizar evaluación: En este proceso el estudiante de acuerdo a lo planificado realiza una evaluación que el docente le asigne.

Realizar evaluaciones diferidas: Este proceso se realiza cuando un alumno por motivo justificado se ausenta a una evaluación, entonces se realiza una prueba extraordinaria para reponer la evaluación original.

Realizar examen de suficiencia: Este proceso lo realiza un alumno que fue reprobado de la asignatura, se trata de una evaluación extra que permite al estudiante reprobado poder aprobar la materia.

Entrega de notas: Para conocer los resultados de la evaluación se realiza el proceso de entrega de notas, este es un informe que realiza el docente y lo muestra a los alumnos.

- **Salidas**

Listado de alumnos: Los alumnos que inscriben la materia son ordenados grupos, y para cada grupo se genera un listado.

Resultados de evaluaciones: El proceso de enseñanza incluye evaluaciones de desempeño, estas arrojan resultado de la escala 0.1 a 10.0 dependiendo del nivel de comprensión de los estudiantes invidentes.

Informe de aprobados y reprobados: Los resultados de las evaluaciones traen como consecuencia como medida de control del sistema que los estudiantes repitan el proceso de enseñanza hasta que este saque un promedio superior al 6.0

- **Control**

Evaluaciones regulares: Durante el periodo del proceso de enseñanza se realizan diferentes evaluaciones para verificar que los conocimientos están siendo asimilados por los alumnos, entre estos podemos mencionar exámenes parciales, laboratorios evaluados, tareas, etc.

Evaluaciones diferidas: Son evaluaciones complementarias que se realizan a los alumnos cuando no pueden asistir a una evaluación regular por algún motivo justificado siempre y cuando se presente el trámite correspondiente en los primeros 3 días hábiles posteriores a la evaluación regular.

Examen de suficiencia: Son evaluaciones remediales que se realizan con el fin de que un alumno logre aprobar la materia, cuando este ha reprobado dicha materia siempre y cuando la nota oscile entre 5.0 y 5.99, esta nota va directamente promediado con la nota de ciclo de la materia reprobada.

Gestión administrativa: Se refiere a todos los procesos administrativos que debe hacer un alumno no vidente que este estudiando en un ciclo normal, entre estos tenemos figuras como el talonario de pagos, la inscripción de materias, y todos los procesos administrativos que son realizados por los alumnos no videntes.

- **Frontera**

La frontera del sistema no puede limitarse a una frontera física sino que mas bien es una frontera virtual ya que el proceso esta delimitado a la enseñanza del idioma ingles el cual es concepto no tangible, si establecemos la frontera por los usuarios que hacen uso de él este incluye a la personas no videntes que estudian el idioma inglés, y si se especifica por contenido esta delimitado por los contenidos de la materia de inglés nivel básico intensivo para alumnos de educación superior.

- **Medio Ambiente**

En el medio ambiente del sistema se encuentran todas aquellas instituciones que atienden personas no videntes con el fin de que estas se eduquen, entre las que podemos mencionar las instituciones de enseñanza de educación superior, el MINED como ente regulador, así como también las ONG's que dedican sus esfuerzos para que personas no videntes puedan salir adelante en sus vidas.

2. Diagramas de flujo de datos

Para lograr comprender mejor la situación actual, en este apartado se hace uso de diagramas de flujo de datos, para describir los procesos que intervienen en la enseñanza del idioma y inglés y que actualmente realizan en el departamento de idiomas de la universidad de El Salvador, en estos diagramas se muestra el flujo de información que hay y las interacciones que se realizan de un proceso a otro.

Los procesos descritos en este apartado son:

“Proceso de enseñanza aprendizaje del idioma inglés”, este se describe mediante el Modelo de contexto figura 2-3.

“Planificación de contenido de la materia”, este se describe en la figura 2-5.

“Planificación de clase”, este se describe en la figura 2-6.

“Impartir Clase”, este se describe en la figura 2-7.

“Realizar evaluación”, este se describe en la Figura 2-8.

“Entrega notas”, este se describe en la Figura 2-9.

2.1. Definición²¹

El diagrama de flujo de datos (DFD) es una herramienta que permite representar un sistema como una red de procesos funcionales conectados entre sí por “flujos” de datos y “almacenes” de datos.

El diagrama de flujo de datos también se conoce como:

Carta/diagrama de burbujas.

Modelo de procesos/función

El DFD de más importancia a las funciones de un sistema de información que a los datos que maneja, por eso se dice que está orientado a las funciones. Los DFD no sólo se usan para los sistemas de información, sino también para la representación de la empresa entera.

Limitaciones del diagrama de flujo de datos:

No define el uso de dispositivos.

Los esquemas están representados sin secuencia del flujo.

Ventajas del diagrama de flujo de datos:

La notación permite un fácil entendimiento de los usuarios.

Permite aislar las áreas de interés.

Se logra un mejor nivel de detalle de las partes específicas de cada proceso.

El análisis va de lo general a lo particular.

2.2. Metodología para la elaboración del diagrama de flujo de datos

Los diagramas de flujo de datos se pueden dibujar con sólo cuatro notaciones sencillas. El uso de esta notación para cada elemento depende de qué enfoque se utilice, el de Yourdon o el de Gane y Sarson.

²¹ Análisis y Diseño de Sistemas, CCPM, Mc Graw Hill.

La siguiente tabla muestra los componentes del diagrama de flujo de datos y sus notaciones más comunes:

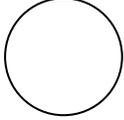
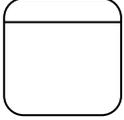
	YOURDON/DEMARCO	GANE/SARSON
Proceso		
Flujo de Datos		
Origen / Destino		
Almacén		

Tabla 2-1

Proceso

Muestra la parte del sistema que transforma las entradas de datos en salidas. Los sinónimos usados son: burbuja, función o transformación. El proceso se identifica con un número y un nombre que se compone de un verbo – Objeto para describir lo que hace. El número del proceso es para su identificación, no para definir una secuencia.

Flujo de datos

El flujo describe el movimiento de paquetes de datos de una parte del sistema a otra. Representa datos en movimiento. El flujo de datos se representa con una flecha porque los datos viajan en una dirección específica desde su origen hacia su destino. Todo flujo debe tener un nombre que define la información que transporta. La única excepción a esta regla es cuando el flujo se dirige a un almacén ya que se supone que transportará la información guardada en él.

Origen / Destino de los datos

Representan entidades externas al sistema que se comunican con él y que están fuera de su control. También se les conoce como terminador. Un terminador puede ser una persona o grupo de personas, por ejemplo, una organización o un departamento. El origen ó destino de los datos se identifica con un rectángulo y un nombre representativo de la entidad. Es importante tomar en cuenta que los terminadores son externos al sistema y no se pueden cambiar sus contenidos ni funciones. Las relaciones que existen entre los terminadores no se representan en el DFD, ya que no son parte del sistema bajo estudio.

Almacén

El almacén se utiliza para representar una colección de paquetes de datos en reposo. No se refiere exclusivamente a los medios de almacenamiento electrónico como bases de datos en discos duros o discos ópticos, sino también a archiveros metálicos o cualquier medio que permita guardar datos en tarjetas u hojas de papel. Los flujos que van hacia el almacén representan los datos que serán guardados en él. Los flujos que vienen desde el almacén representan los datos que serán extraídos de él. El almacén se identifica con un nombre (generalmente en plural) que representa su contenido.

Sinónimos

Cuando un elemento "entidad" se debe especificar más de una vez, sin que este represente otra entidad del mundo real, este se convierte en un sinónimo del elemento inicial agregando una pequeña línea inclinada en la esquina inferior derecha para hacer referencia a la misma entidad.

Descomposición

Cuando a un proceso se subdivide en un nivel más bajo en su especificación se coloca una pequeña línea inclinada y un signo " + " en la esquina inferior derecha para hacer denotar que el proceso esta descompuesto en un nivel de procesos menor.

Niveles de un DFD.

El DFD representa el sistema de información desde lo general hacia lo específico. Los niveles de un DFD señalan el grado de detalle de la representación del sistema, estos son:

El Diagrama de contexto.

El Diagrama de nivel 0.

Los Diagramas de nivel 1.

El Diagrama de contexto muestra el sistema como un solo proceso para hacer énfasis en los terminadores y los flujos de datos. El diagrama de nivel 0 muestra los procesos más importantes del sistema para resaltar la transformación de los datos.

Desde los diagramas de nivel 1 y subsecuentes se describirán los procesos de manera más detallada. Se deberá elaborar un diagrama de nivel 1 para cada proceso de nivel 0. De ser necesario, se elaborarán diagramas de nivel 2 para detallar cualquier proceso de nivel 1, y así sucesivamente. Los flujos de entrada y salida del diagrama de nivel 1 debe concordar con los flujos del proceso del diagrama de nivel 0 correspondiente.

2.3. Modelo de Contexto

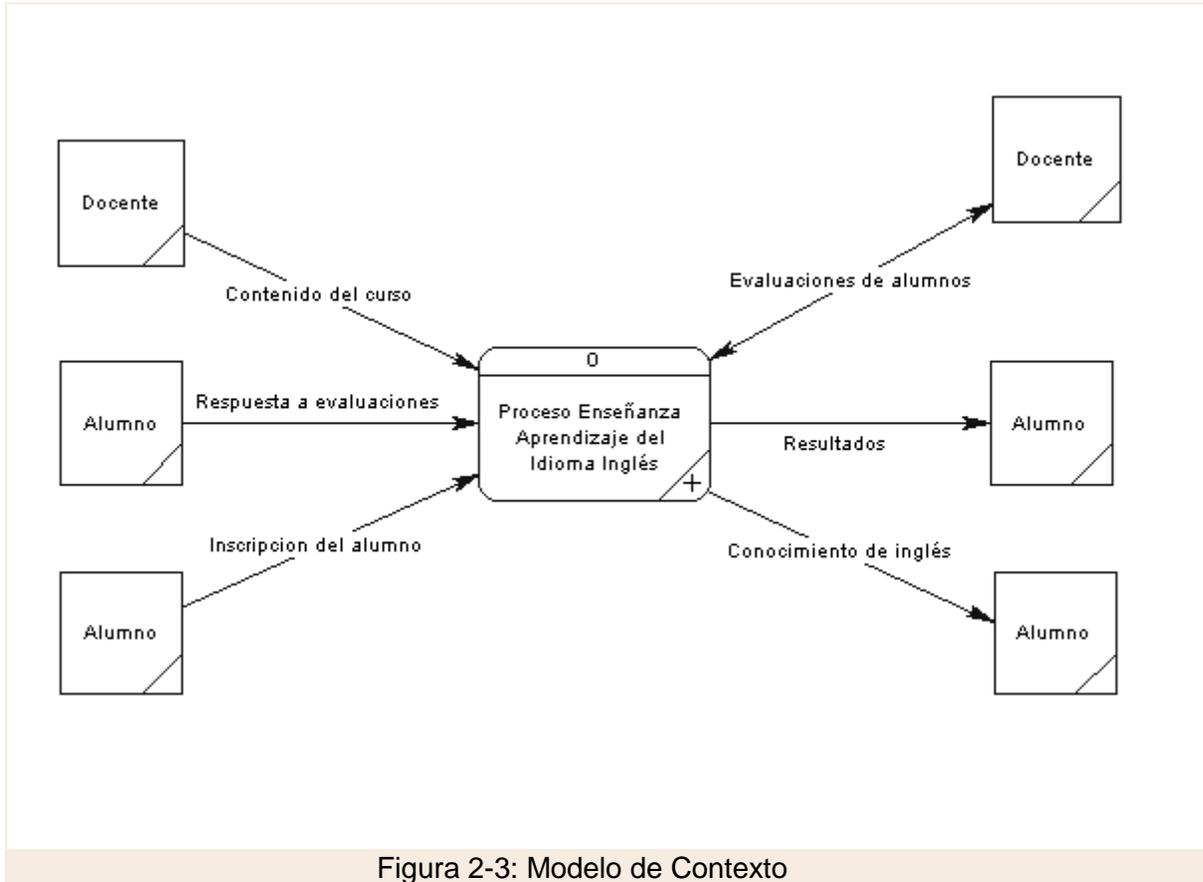


Figura 2-3: Modelo de Contexto

En el modelo de contexto se puede observar los flujos que entran y salen del proceso completo de enseñanza-aprendizaje, además de los actores que se ven involucrados en sus actividades.

2.4 Diagrama de nivel 0

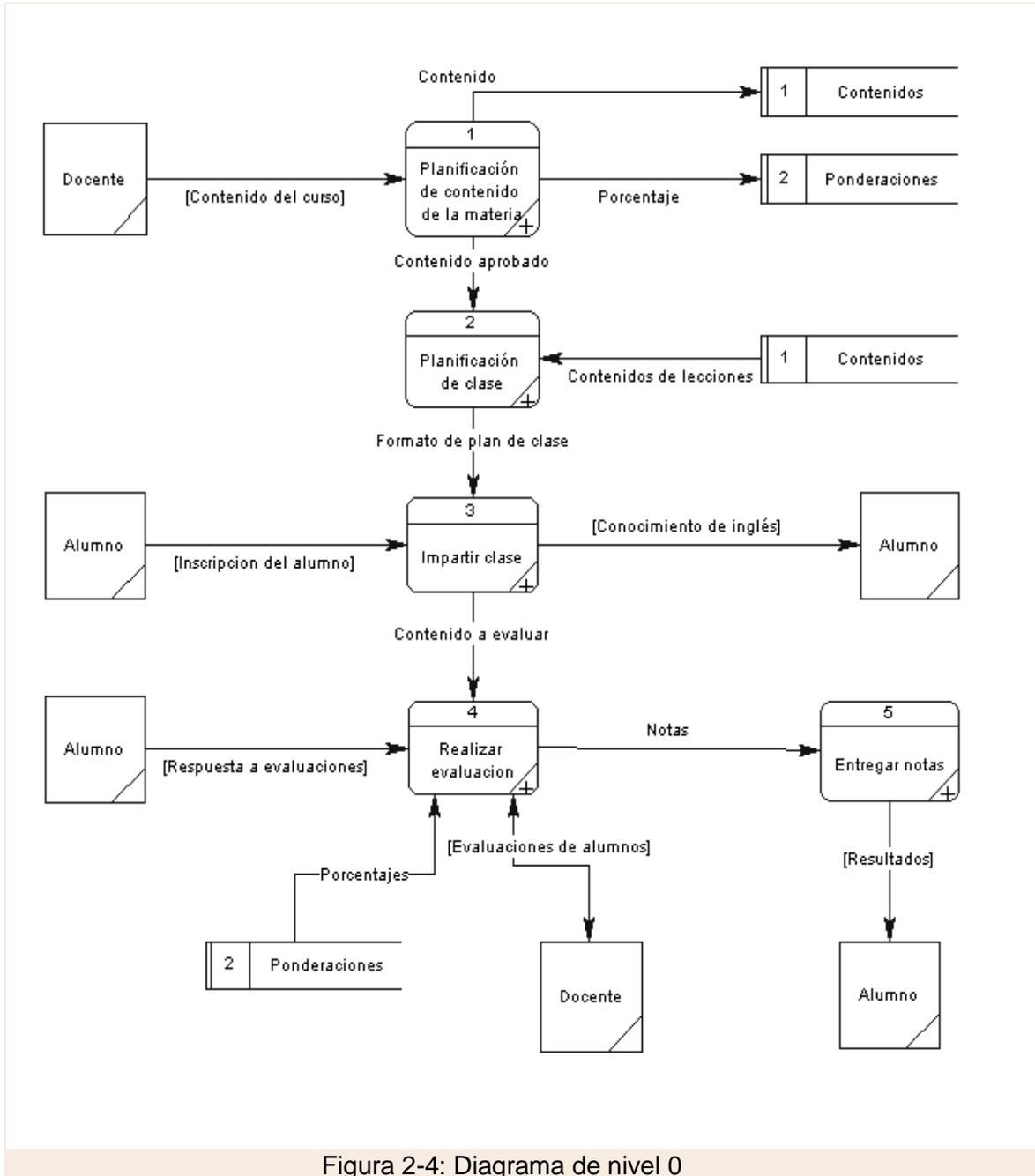


Figura 2-4: Diagrama de nivel 0

En el diagrama de nivel cero se observan los procesos, los flujos de datos que salen y entran a ellos, los almacenes que proveen y se alimentan de datos, según se observa en el diagrama el proceso incluye 5 procesos principales, 2 actores y 2 almacenes.

2.4. Diagramas de nivel 1

Proceso: Planificación de contenido de la materia.

Descripción:

Este proceso representa la acción de planificar el contenido al iniciar un periodo de clases en el cual se debe obtener un cronograma de actividades ha seguir durante todo el periodo.

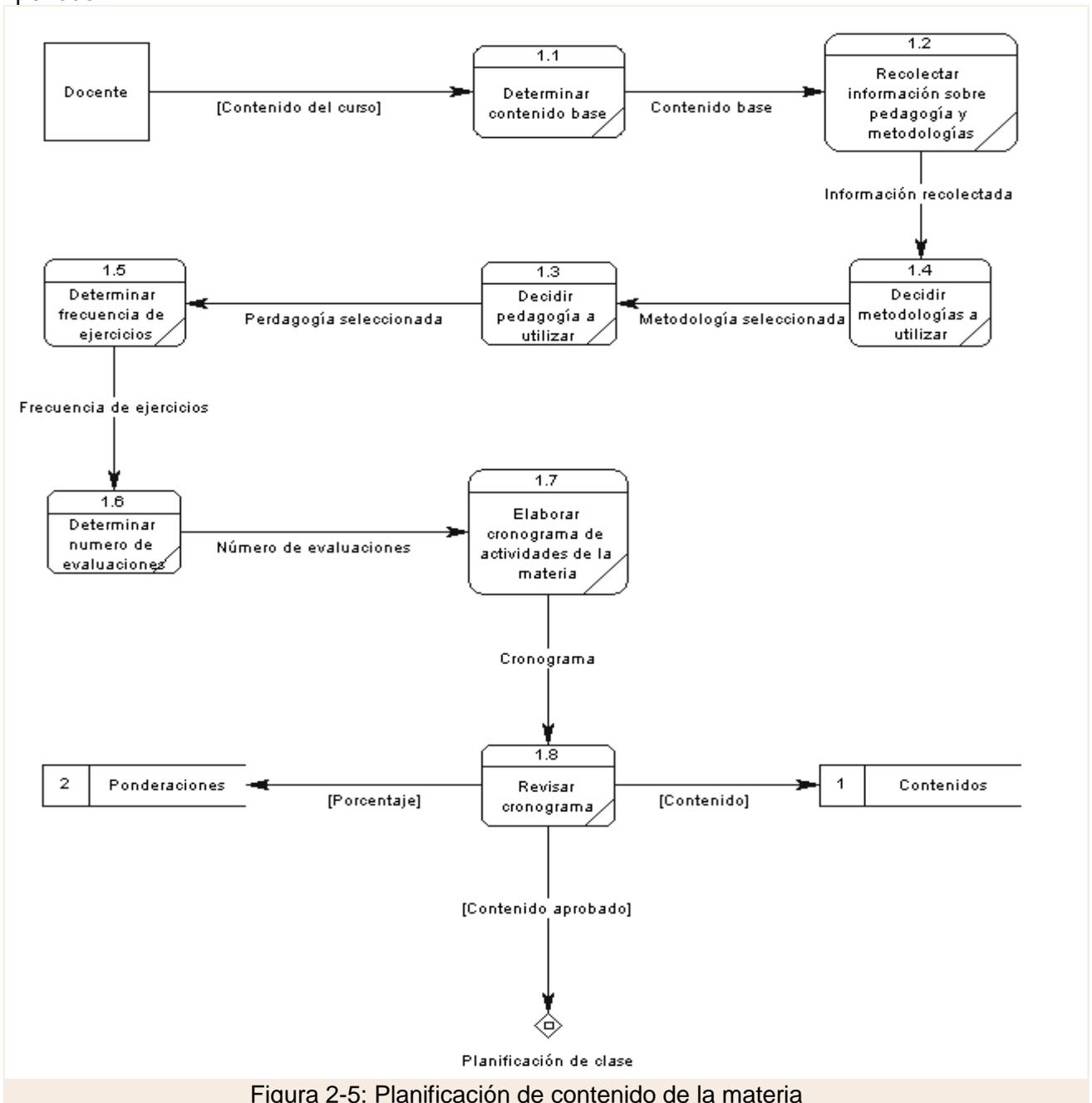


Figura 2-5: Planificación de contenido de la materia

Proceso: Planificación de clase

Descripción:

Este proceso tiene lugar cuando un maestro planifica una clase en particular basándose en el contenido según el cronograma de actividades de la materia, incluye la planificación de todos los aspectos previos para impartir una clase.

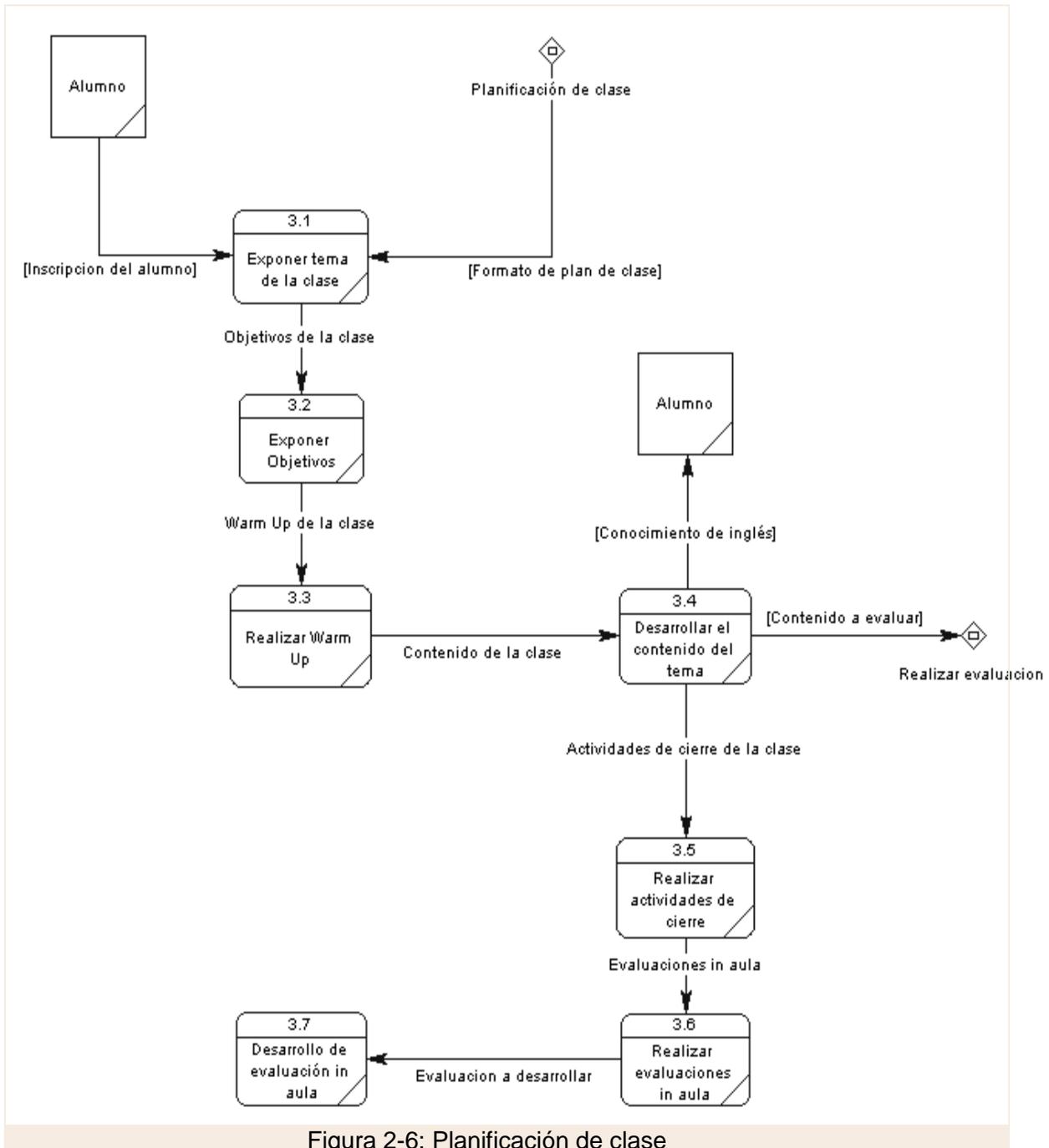


Figura 2-6: Planificación de clase

Proceso: Impartir clase

Descripción:

En este proceso se realiza el acercamiento del conocimiento hacia el alumno ya que es en la clase que se desarrollan los contenidos y actividades que fueron planificadas previamente, dicho proceso incluye también la realización de pequeñas evaluaciones in aula o ejercicios prácticos.

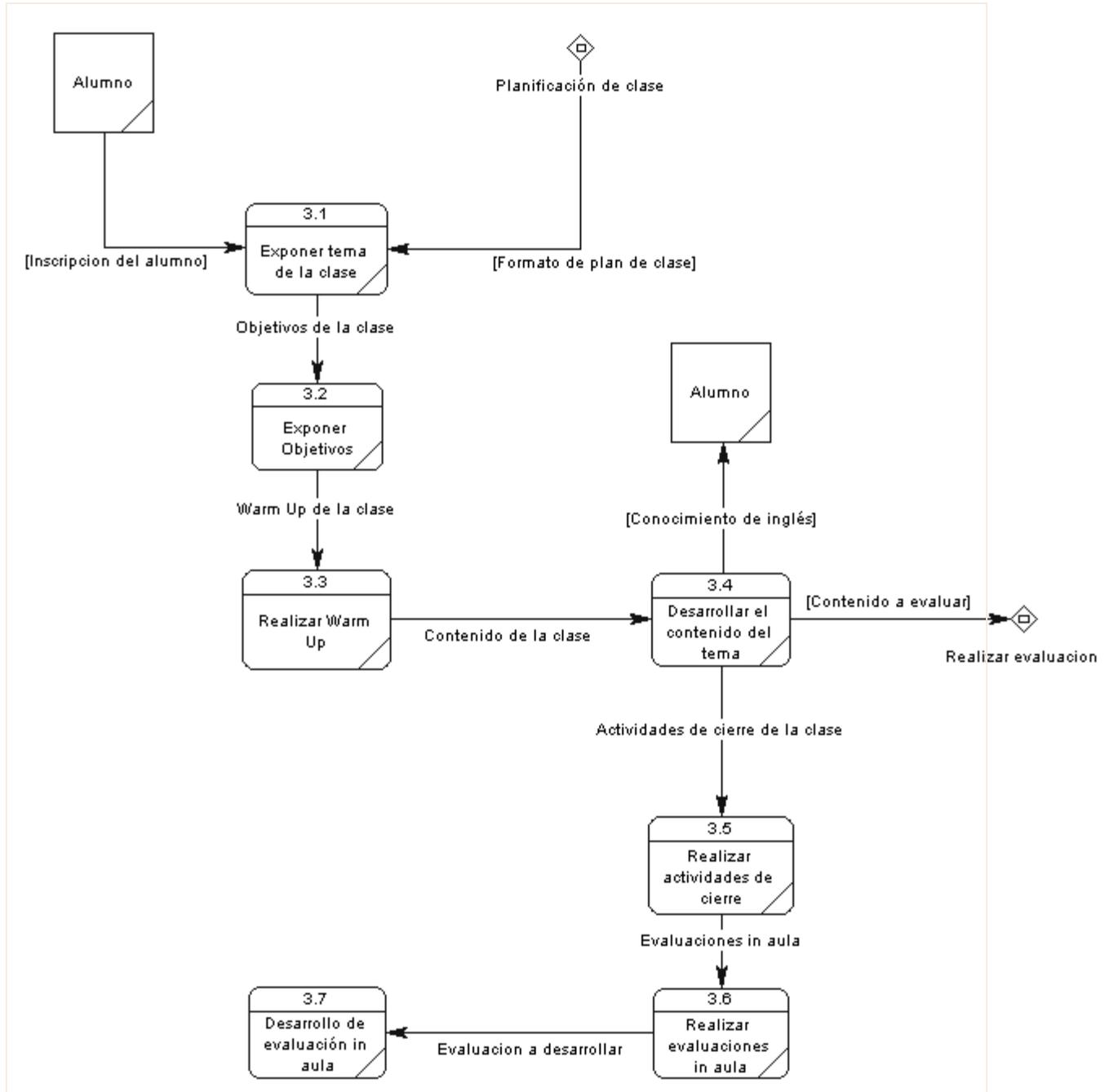


Figura 2-7: Impartir clase

Proceso: Realizar evaluación

Descripción:

Este proceso incluye los pasos necesarios para realizar una evaluación de un período hasta obtener la calificación obtenida en dicha evaluación la cual esta basada en ponderaciones previamente planificadas, también se incluyen aquí las situaciones en las que los alumnos realizan evaluaciones diferidas o un examen de suficiencia.

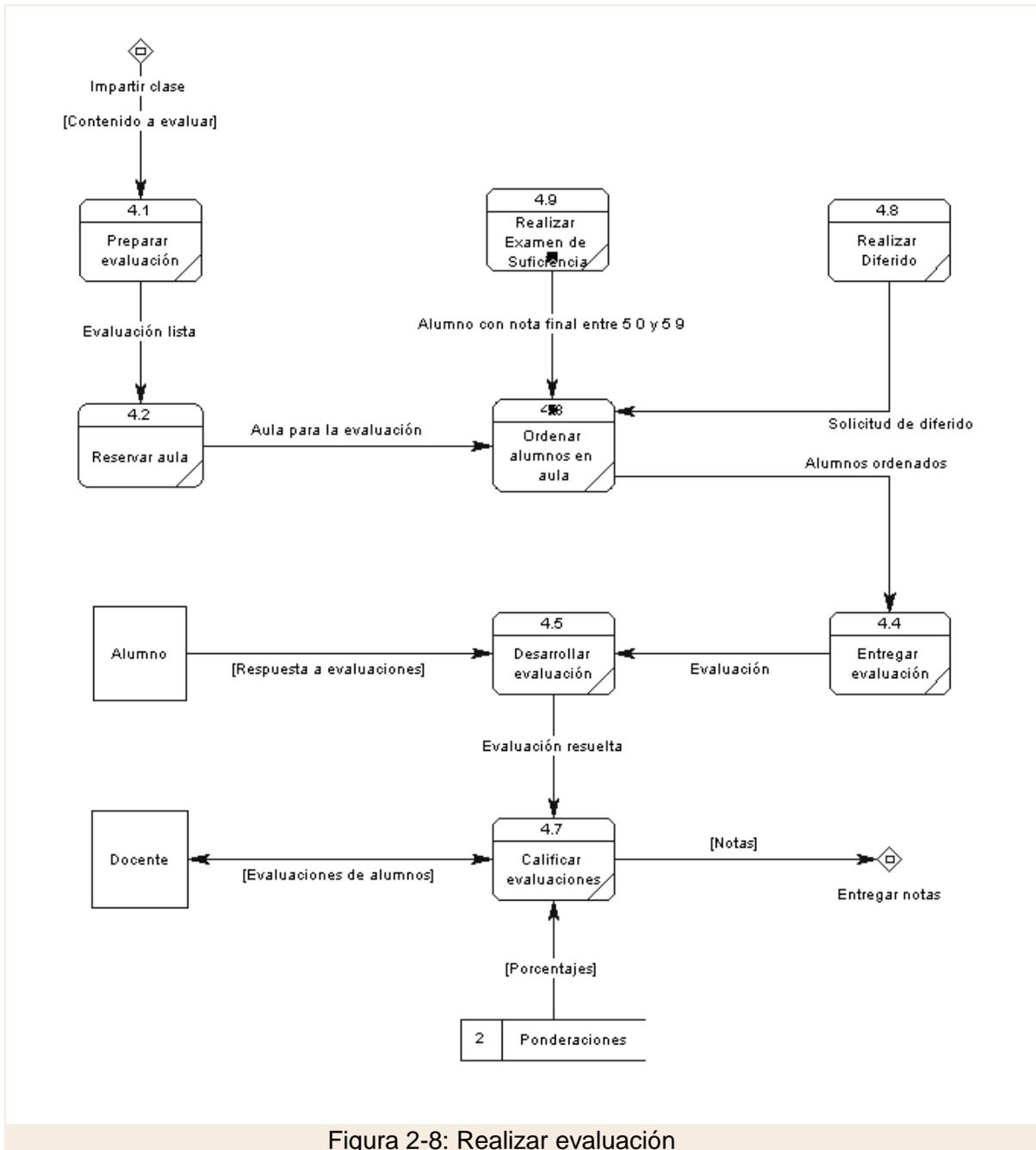
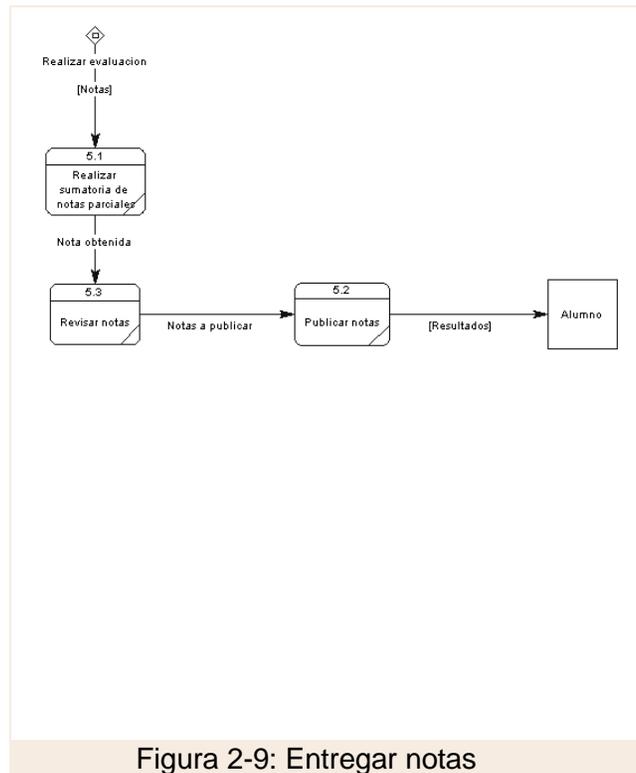


Figura 2-8: Realizar evaluación

Proceso: Entregar notas

Descripción:

Este proceso incluye las actividades involucradas en dar a conocer a los alumnos los resultados obtenidos en las evaluaciones, el cual debe ocurrir en no más de 20 días luego de realizada la evaluación²².



²² Reglamento de la Administración Académica de la Universidad de El Salvador Capítulo IV Evaluación Art. 17 – B

3. Modelado de Procesos

En el apartado 2.4. Se describieron los procesos principales que intervienen en el proceso enseñanza aprendizaje del idioma ingles, en este apartado se modelan los procesos mediante diagramas de flujos o flujogramas; esto se realiza con el objetivo de comprender las actividades realizadas por los diferentes actores que intervienen, tanto los Diagramas de Flujo de Datos como los flujogramas se utilizan para modelar procesos es decir desarrollar una descripción lo mas exacta por esa razon se utilizan estas dos herramientas para describir los procesos, ademas en este apartado se incluyen dos subprocesos mas que son: “realizar evaluaciones diferidas” y “realizar examen de suficiencia”, los cuales no se describieron en el apartado anterior debido a que solo se pusieron los mas importantes.

Los procesos descritos mediante flujogramas en este apartado son:

“Planificacion de contenido de la materia”.

“Planificacion de clase”.

“Impartir Clase”.

“Realizar evalluacion”.

“Realizar evaluaciones diferidas”

“Realizar examen de suficiencia”

“Entrega notas”.

2.1 Definición²³

Frecuentemente los sistemas son difíciles de comprender por ser amplios, complejos y confusos; con múltiples puntos de contacto entre sí y con un buen número de áreas funcionales, departamentos y puestos implicados. Un modelo puede dar la oportunidad de organizar y documentar la información sobre un sistema.

Un modelo es una representación de una realidad compleja. Modelar es desarrollar una descripción lo más exacta posible de un sistema y de las actividades llevadas a cabo en él.

Cuando un proceso es modelado, con ayuda de una representación gráfica (diagrama de proceso), pueden apreciarse con facilidad las interrelaciones existentes entre distintas actividades, analizar cada actividad, definir los puntos de contacto con otros procesos, así como identificar los subprocesos comprendidos. Al mismo tiempo, los problemas existentes pueden ponerse de manifiesto claramente dando la oportunidad al inicio de acciones de mejora.

Diagramar es establecer una representación visual de los procesos y subprocesos, lo que permite obtener una información preliminar sobre la amplitud de los mismos, sus tiempos y los de sus actividades.

²³ <http://www.aiteco.com/web/index.php/modelado-de-procesos.html>

La representación gráfica facilita el análisis, uno de cuyos objetivos es la descomposición de los procesos de trabajo en actividades discretas. También hace posible la distinción entre aquellas que aportan valor añadido de las que no lo hacen, es decir que no proveen directamente nada al cliente del proceso o al resultado deseado.

En este último sentido cabe hacer una precisión, ya que no todas las actividades que no proveen valor añadido han de ser innecesarias; éstas pueden ser actividades de apoyo y ser requeridas para hacer más eficaces las funciones de dirección y control, por razones de seguridad o por motivos normativos y de legislación.

Diagramar es una actividad íntimamente ligada al hecho de modelar un proceso, que es por sí mismo un componente esencial en la gestión de procesos

3.2 Metodología para la Especificación Estructurada de Procesos EEP²⁴

Cuando un proceso es modelado, mediante una representación gráfica, pueden apreciarse con facilidad las interrelaciones existentes entre distintas actividades, analizar cada actividad, definir los puntos de contacto con otros procesos, así como identificar los subprocesos comprendidos en él.

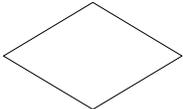
Al mismo tiempo al utilizar columnas por actores, los procesos se pueden poner de manifiesto claramente para mostrar la relación existente entre sí y los responsables de dicho proceso.

La representación gráfica facilita el análisis, uno de cuyos objetivos es la descomposición de los procesos de trabajo en actividades discretas.

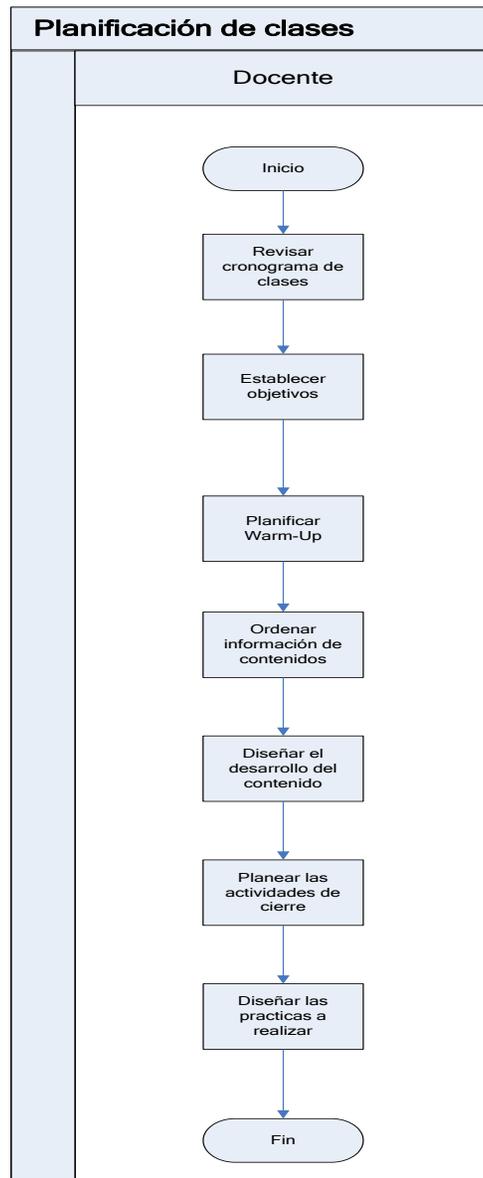
También hace posible la distinción entre aquellas actividades que aportan valor añadido de las que no lo hacen, es decir que no proveen directamente nada al usuario del proceso o al resultado deseado.

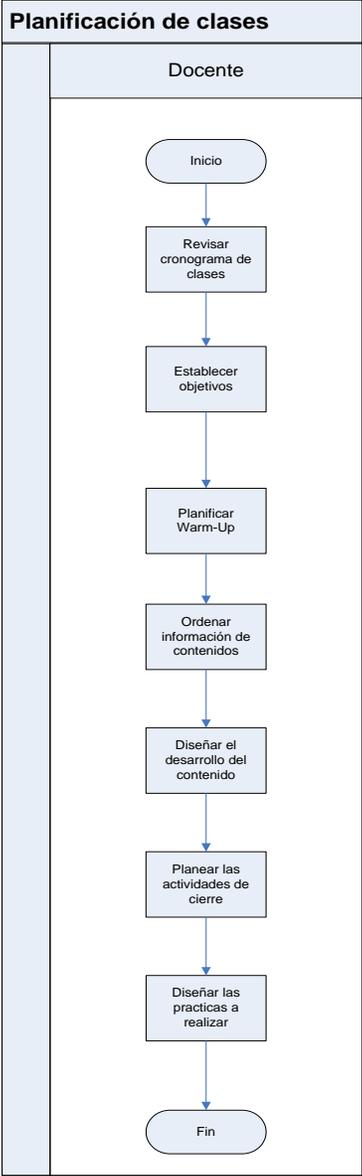
En este último sentido cabe hacer una precisión, ya que no todas las actividades que no proveen valor añadido han de ser innecesarias; éstas pueden ser actividades de apoyo y ser requeridas para hacer más eficaces las funciones de dirección y control, por razones de seguridad o por motivos normativos y de legislación.

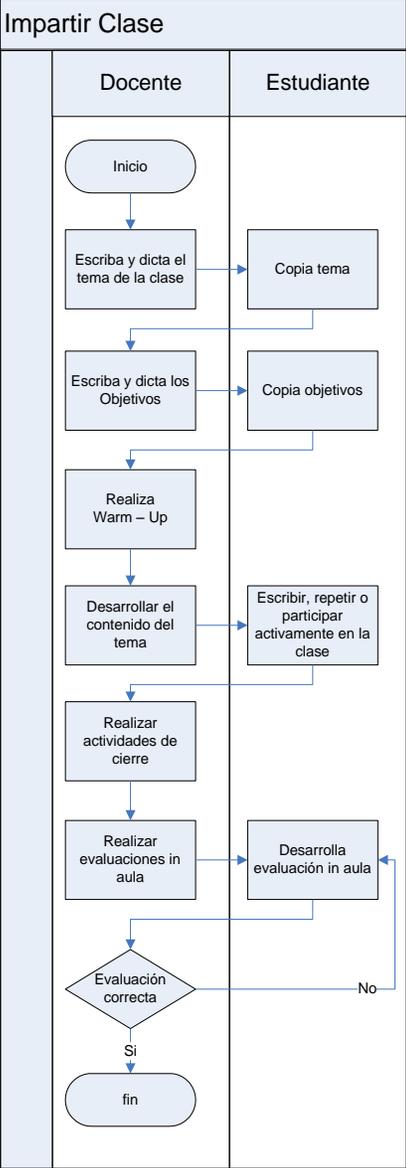
²⁴ http://es.wikipedia.org/wiki/Diagrama_de_flujo
<http://www.elticus.com/?contenido=19>

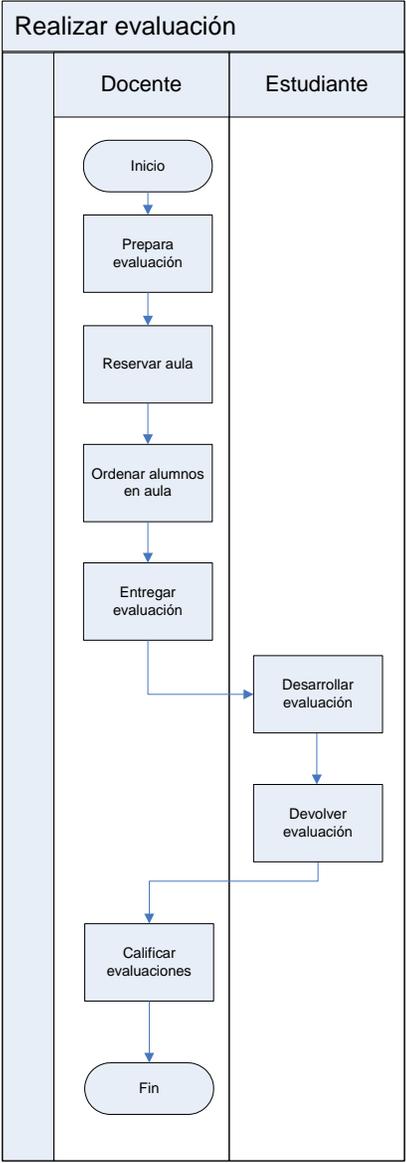
Nombre	Representación	Descripción
Inicio / Terminación		Este símbolo se utiliza para señalar el comienzo así como el final de un diagrama. Es el único símbolo que solamente tiene una conexión ya sea de salida o de entrada, dependiendo del uso.
Proceso		Este símbolo se utiliza para señalar operaciones matemáticas, aritméticas o procesos específicos. Este símbolo siempre deberá tener al menos una conexión de entrada y una de salida.
Bifurcación		Se utiliza para representar una condición. Normalmente el flujo de información entra por arriba y sale por un lado si la condición se cumple o sale por el lado opuesto si la condición no se cumple.
Desplegado de información		Este símbolo se utiliza para mostrar un resultado, el cual puede representar la solución al problema que se pretende resolver y que fue conseguida a través del resto del diagrama.
Flecha		Indica el sentido y trayectoria del proceso de información o tarea.
Conector		Este símbolo se utiliza para indicar un salto dentro del diagrama. Se utiliza con el propósito de facilitar la disposición plana de un diagrama y evitar el cruce excesivo de líneas a través del mismo.
Conector de página		Este conector indica que se debe buscar el "gemelo" en una página distinta de la actual. Este conector lleva asociado una especie de salto entre páginas.

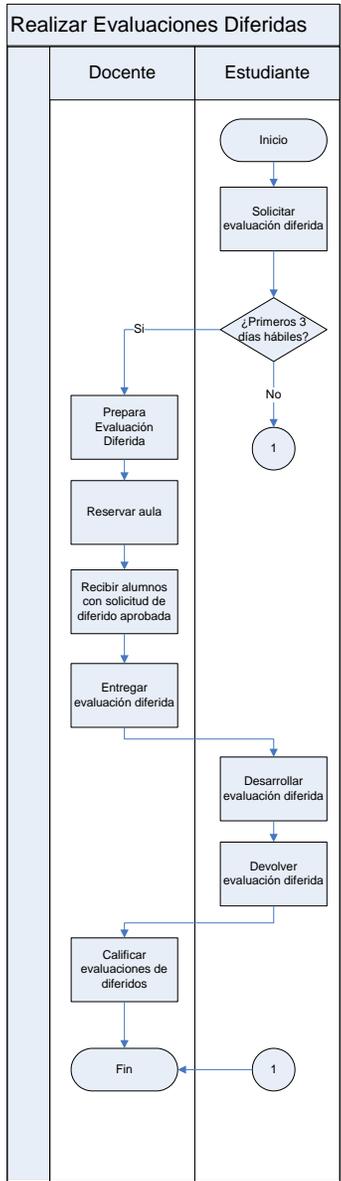
3.3 Diagramas de Modelos de Procesos

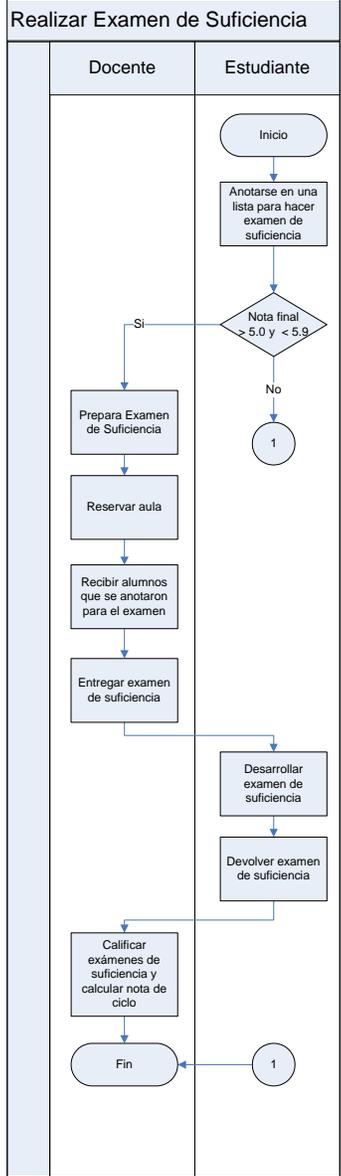


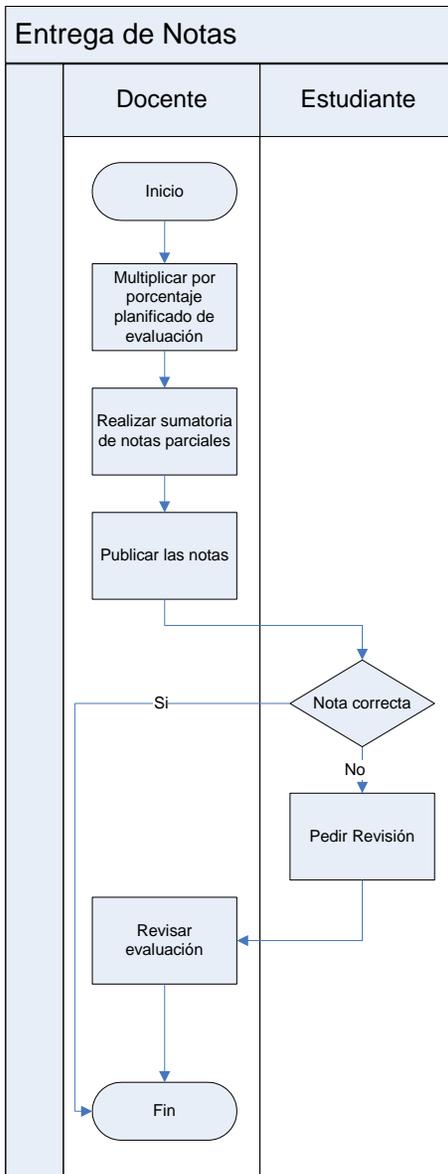












Capítulo III: Determinación de los requerimientos

Requerimientos

En este apartado, se describe como se produjo la obtención de requerimientos para el software SPEI3, se especifican los requerimientos funcionales y no funcionales. Las especificaciones se dividen en dos: de usuario y de sistema.

1. Definición

Es la redacción en lenguaje natural de los requerimientos dados por el usuario²⁵. Esta determinación de los requerimientos fue el producto de la investigación realizada para la cual se aplicaron técnicas de investigación y métodos de especificación de requerimientos que se detallan a continuación.

2. Metodología para la obtención y determinación de los requerimientos

Para la obtención de requerimientos se utilizaron las técnicas de investigación siguientes: entrevista, observación y lluvia de ideas. A continuación se describen cada una de estas técnicas y se detalla como se utilizaron en la investigación.

Entrevista

*Definición*²⁶

La entrevista es una conversación entre dos o más personas, en la cual uno es el que pregunta (entrevistador). Estas personas dialogan con arreglo a ciertos esquemas o pautas de un problema o cuestión determinada, teniendo un propósito profesional. Presupone la existencia de personas y la posibilidad de interacción verbal dentro de un proceso de acción recíproca. Como técnica de recolección va desde la interrogación estandarizada hasta la conversación libre, en ambos casos se recurre a una guía que puede ser un formulario o esquema de cuestiones que han de orientar la conversación.

Aplicación del método en el proyecto

Para el proyecto se realizaron entrevistas al jefe del departamento de idiomas, (ver Anexo 1) y al estudiante invidente Armando Madrid,(ver Anexo 2) quienes muy cordialmente prestaron su total disponibilidad y colaboración con el proyecto. También pueden ver los videos de la realización de ambas entrevistas en el CD anexo a este documento.

²⁵ Ingeniería de software Ian Sommerville 100p

²⁶ Rodríguez, Indiana. Guía sobre metodología y técnica de la investigación. San José: Colon La Paix. 1992. 198p.

Observación

*Definición*²⁷

La observación es la técnica más antigua y la más empleada en investigación, pero ella tendrá valor en la medida que: Sirva a un objetivo ya formulado en la investigación, sea planificada en forma sistemática, se busque relacionarla con proposiciones más generales, esté sujeta a comprobaciones y controles de validez y confiabilidad.

En el acto de observación se pueden distinguir:

- 1) El observador;
- 2) el objeto de observación;
- 3) los medios para observar;
- 4) las condiciones de la observación y,
- 5) el sistema de conocimientos relacionados con la finalidad de las observaciones y las interpretaciones que resulten de ella.

Aplicación del método en el proyecto

La observación se aplicó con el objetivo de obtener insumos para la lluvia de ideas, se realizó en el Centro de Rehabilitación para Ciegos “Eugenia de Dueñas” y en la Universidad de El Salvador. Esta fue hecha de forma sistemática y planificada. Se realizaron 5 visitas al Centro de Rehabilitación “Eugenia de Dueñas” para observar como las personas les enseñaban a los ciegos materias comunes ya que en este instituto no se cuenta con la enseñanza del idioma inglés además se realizaron 12 visitas al Departamento de Idiomas de la Universidad de El Salvador. Los resultados de la observación se anotaban en papel y se discutía en el equipo del proyecto.

Lluvia de ideas

*Definición*²⁸

La lluvia de ideas o Brainstorming, también denominada Tormenta de ideas es una herramienta de trabajo grupal que facilita el surgimiento de nuevas ideas sobre un tema o problema determinado. La lluvia de ideas es una técnica de grupo para generar ideas originales en un ambiente relajado.

Esta herramienta fue creada en el año 1941, por Alex F. Osborne, cuando su búsqueda de ideas creativas resultó en un proceso interactivo de grupo no estructurado que generaba más y mejores ideas que las que los individuos podían producir trabajando de forma independiente; dando oportunidad de sugerir sobre un determinado asunto y aprovechando la capacidad creativa de los participantes.

La regla primordial es aplazar el juicio en un principio toda idea es válida y ninguna debe ser rechazada. Habitualmente, en una reunión para resolución de problemas, muchas ideas, tal vez aprovechables, mueren precozmente ante una observación "juiciosa" sobre su inutilidad o carácter disparatado. De ese modo se impide que las ideas generen, por analogía, más ideas, y, además, se inhibe la creatividad de los participantes. En un brainstorming se busca tácticamente la cantidad sin pretensiones de calidad y se valora la originalidad. Cualquier persona del grupo, podrá aportar cualquier idea de cualquier

²⁷ Información obtenida del documento “La técnica de observación” Monografía de Prof. Alexis Labarca
http://www.umce.cl/publicaciones/mie/mie_modulo3.pdf

²⁸ http://es.wikipedia.org/wiki/Lluvia_de_ideas

índole, que crea conveniente para el caso tratado. Un análisis ulterior explotará estratégicamente la validez cualitativa de lo producido con esta técnica. Luego la técnica es complementada con otras como clasificación de ideas, selección de ideas y cuantificación de ideas.

Aplicación del método en el proyecto

La lluvia de ideas se realizó mediante reuniones grupales, los insumos de estas reuniones fueron de la aplicación de los métodos de la entrevista y observación, la reunión consistió en que los miembros del grupo explicaba el requerimiento y lo anotaba en un papel, luego se discutía los distintos puntos de vista expresados por los miembros del grupo y se emitía un juicio, esta determinación dio como resultado la obtención de cada requerimiento funcional y no funcional.

Clasificación de los requerimientos²⁹

A menudo, los requerimientos de sistemas de software se clasifican en funcionales y no funcionales o como clasificación del dominio.

1. **Requerimientos funcionales:** Son declaraciones de los servicios que proveerá el sistema, de la manera en que este reaccionará a entradas particulares y de cómo se comportará en situaciones particulares.
2. **Requerimientos no funcionales:** Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, etc.
3. **Requerimientos del dominio:** Son requerimientos que provienen del dominio de aplicación del sistema y que reflejan las características de ese dominio. Estos pueden ser funcionales o no funcionales.

En realidad, la distinción entre estos tipos diferentes de requerimientos no es tan clara como sugieren estas definiciones. Por ejemplo, un requerimiento del usuario sobre seguridad podría parecer uno no funcional. Sin embargo, cuando se desarrolla en detalle, conduce a otros que son claramente funcionales, como la necesidad de incluir en el sistema los recursos para la autorización del usuario. Por lo tanto aunque en la discusión sea de utilidad clasificar de esta forma los requerimientos, debe recordar, que en la realidad es una distinción artificial.

Especificación de los requerimientos³⁰

Para especificar los requerimientos se tomo en cuenta dos puntos de vista, el de usuario final y el de los desarrolladores del software, para ello se realizaron dos especificaciones conocidas como de usuario y de sistema.

❖ *Especificación de usuario.*

Las especificaciones de usuario son los requerimientos expresados de forma clara y entendible para usuarios no especializados en el desarrollo de software.

²⁹ Ingeniería de Software, Ian Sommerville, Cap 5. Req. Del Software, Pág. 100

³⁰ Ingeniería de Software, Ian Sommerville, Cap 5. Req. Del Software, Págs. 111 y 112

❖ *Especificación de sistema*

Después de realizar la clasificación de los requerimientos, se debe de tener una visión mas detallada de los requerimientos funcionales dados por el usuario, por cada definición dada por el usuario se utilizó un formato estándar que incluye la siguiente información:

1. Una descripción de la función o entidad a especificar.
2. Una descripción de sus entradas y de donde provienen.
3. Una descripción de sus salidas y hacia donde van.
4. Una indicación de que otras entidades se utilizan.
5. Si se utiliza un enfoque funcional, una precondition que indique lo que se debe cumplir antes de que la función sea invocada y una poscondición que especifique lo que será verdad después de que dicha función se haya invocado.
6. Una descripción de los efectos colaterales (si existen) de la operación.

Estos seis puntos son aplicados para la primera especificación de sistema de los requerimientos contenido en este apartado, para la segunda *especificación de sistema* se realizaron casos de uso que se explican con mas detalle en la siguiente sección, en ella se presenta la metodología que se utiliza para la elaboración de los casos de uso y la especificación de los mismos.

3. Requerimientos Funcionales

Son declaraciones de los servicios que proveerá el sistema, de la manera en que este reaccionará a entradas particulares y de cómo se comportará en situaciones particulares. Estos requerimientos también declaran explícitamente la funcionalidad y los servicios que se espera que éste provea.

Éstos dependen del tipo de software y del sistema que se desarrolle y de los posibles usuarios del software. Cuando se expresan como requerimientos del usuario, habitualmente se describen en forma *general*, mientras que los requerimientos funcionales del sistema se describen con detalle ya que indican la función de éste, sus entradas y salidas, excepciones, etcétera.³¹

3.1 Especificación de Usuario

1. Impartir clases

Se necesita una herramienta que sea capaz de impartir lecciones del idioma inglés de nivel básico-intensivo de una manera audible para que los alumnos no videntes puedan estudiar o repasar el contenido de las lecciones.

2. Registrar avances de alumno

Se necesita tener un registro de las lecciones cursadas por los alumnos así como la precisión con la cual las realiza, y el tiempo que ha dedicado a ellas para su correspondiente revisión por parte del docente.

³¹ Ingeniería de Software, Ian Sommerville, Cap 5. Req. Del Software, Pág.100

3. Realizar ejercicios prácticos

Se necesita una herramienta que permita a los alumnos no videntes la realización de ejercicios prácticos al estar cursando las lecciones, que los ejercicios sigan la técnica de practica controlada³² y que puedan ser de diferentes tipos como: selección múltiple, formar palabras con letras desordenadas, rellenar espacios en blanco.

4. Evaluar desempeño del alumno

Se necesita una herramienta que permita realizar evaluaciones de las lecciones cursadas que han sido cursadas por el alumno brindando luego de la evaluación un resultado cualitativo de forma que ayude a comprender el desempeño del alumno.

5. Planificar clases

Se necesita tener una planificación de las clases que los alumnos van a recibir por lo que el maestro establecerá el orden de los elementos de la clase y los tipos más adecuados de pedagogía a utilizar según su criterio.

6. Administrar usuarios

Se debe poder conocer que docentes y que alumnos están siendo involucrados en el proceso de enseñanza del idioma inglés a personas no videntes, así como también que actividades están desempeñando cada uno.

3.2 Requerimientos clasificados por usuario

Estos requerimientos se obtuvieron a partir de la información recopilada en las entrevistas realizadas. Dichos requerimientos se refinaron para obtener los siguientes poder clasificarlos por usuario solicitante.

✓ Requerimientos del Alumno

RF-1 Seleccionar Tema a Reproducir.

RF-1.1 Desplazarse entre elementos del tema.

RF-1.2 Pausar reproducción de contenido.

RF-1.2.1 Reanudar reproducción contenido.

RF-1.3 Detener tema.

RF-1.4 Controlar Velocidad de Reproducción de Voz

RF-2 Desarrollar ejercicios y mostrar resultado.

RF-3 Desarrollar evaluaciones y mostrar resultado

RF-3.1 Acceder directamente a evaluación de lección

RF-4 Modificar cuenta de usuario estudiante

✓ Requerimientos del Docente

RF-5 Diseñar ejercicios.

RF-5.1 Tipos de ejercicios

RF-5.1.1 Selección múltiple.

³² Ver Glosario de Términos.

- RF-5.1.2 Formar palabras con letras desordenas.
- RF-5.1.3 Rellenar espacios en blanco.
- RF-5.2 Estructurar ejercicio.
- RF-5.3 Vista preliminar del ejercicio.
- RF-5.4 Guardar ejercicio y asociarlo a un tema.
- RF-5.5 Modificar ejercicio y asociarlo a un tema.

- RF-6 Diseñar evaluaciones.
 - RF-6.1 Tipos de evaluaciones
 - RF-6.1.1 Selección múltiple
 - RF-6.1.2 Formar palabras con letras desordenas
 - RF-6.1.3 Rellenar espacios en blanco.
 - RF-6.2 Estructurar evaluación
 - RF-6.3 Vista preliminar de la evaluación
 - RF-6.4 Guardar evaluación y asociarla a un tema.
 - RF-6.5 Modificar evaluación y asociarla a un tema.

- RF-7 Diseñar contenidos.
 - RF-7.1 Estructurar contenido.
 - RF-7.2 Vista preliminar del contenido.
 - RF-7.3 Guardar contenido y asociarlo a un tema.
 - RF-7.4 Modificar contenido y asociarlo a un tema.

- RF-8 Seleccionar Tema a Organizar
 - RF-8.1 Organizar elementos de temas

- RF-9 Administración de alumno
 - RF-9.1 Eliminar usuario alumno
 - RF-9.2 Crear usuario alumno
 - RF-9.3 Modificar usuario alumno
 - RF-9.4 tiempo de práctica del usuario alumno
 - RF-9.5 tema trabajado por usuario alumno
 - RF-9.6 Nota cualitativa del usuario alumno
 - RF-9.7 Tiempo de sesión del usuario alumno
 - RF-9.8 Indicar temas a reforzar
 - RF-9.9 Ver listado de alumnos
 - RF-9.10 Consultar grado de avance por alumno

- RF-10 Modificar cuenta de usuario docente
 - RF-10.1 Iniciar sesión docente

- RF-11 Administración de grupos
 - RF-11.1 Crear grupo
 - RF-11.2 Cambiar alumno de un grupo
 - RF-11.3 Eliminar grupo
 - RF-11.4 El docente solo puede administrar los grupos creados por el mismo, mas no sobre los de otro docente.

✓ **Requerimientos para el Administrador de Docentes**

RF-12 Administración de docentes

- RF-12.1 Crear usuario Docente.
- RF-12.2 Eliminar usuario Docente.
- RF-12.3 Modificar usuario Docente.
- RF-12.4 Ver listado de Docentes.
- RF-12.5 Modificar su cuenta de usuario.
- RF-12.6 Iniciar sesión docente.

RF-13 Realizar cierre de ciclo

- RF-13.1 Respalidar datos de alumnos de ciclo a cerrar
- RF-13.2 Iniciar nuevo ciclo

3.3 Especificación de Sistema

Para que el software se pueda comprender mas claramente, se puede definir además de los requerimientos antes mencionados un nivel mas detallado de requerimientos en los cuales se detallan ciertos aspectos importantes como las entradas que necesitan los requerimientos para lograr producir las salidas requeridas por los usuarios del software SPEI3, además de quienes proporcionan dichas entradas, hacia donde va esa información, si tienen restricciones o si tienen condiciones previas o posteriores que se deben satisfacer.

Requerimientos del Alumno:

RF-1 Seleccionar Tema a Reproducir	
Descripción	El estudiante podrá seleccionar un tema a reproducir dentro del conjunto de temas que contiene determinada lección y el software iniciará su reproducción hasta que termine o el usuario detenga o pause el proceso de reproducción, cuando el tema haya terminado de reproducirse SPEI3 deberá anunciar que el tema a finalizado.
Entradas	Tema seleccionado.
Fuente	El tema es seleccionado por el estudiante.
Salidas	Tema reproducido.
Destino	Tema escuchado por el estudiante.
Requerimientos	El tema es seleccionado por el estudiante.
Precondición	El estudiante debe haber seleccionado una lección.
Post condición	N/A.
Efectos colaterales	Si el tema esta en reproducción y es detenido y el estudiante da la orden de reproducir, el tema se reproducirá desde el inicio, no así cuando el proceso es pausado, ya que el tema se reproducirá desde el punto donde fue puesta la pausa.

TABLA 3-1

Para consultar las descripciones de los requerimientos funcionales puede hacerlo desde el CD adjunto a este documento en la opción "Requerimientos".

4. Requerimientos de Información

Usuario Alumno

Requerimientos asociados:

- RF-1 Reproducción de temas
- RF-2 Mostrar resultado de ejercicios
- RF-3 Mostrar resultado de evaluaciones

Usuario Docente

Requerimientos asociados:

- RF-9.4 Mostrar el tiempo de práctica de los alumnos
- RF-9.5 Mostrar los temas trabajados por los alumnos
- RF-9.6 Mostrar las notas cualitativas obtenidas por los alumnos
- RF-9.7 Mostrar el tiempo de sesión de los alumnos
- RF-9.9 Ver listado de alumnos
- RF-9.10 Consultar grado de avance por alumno

Usuario Administrador

Requerimientos Asociados:

- RF-12.4 Ver listado de docentes

5. Requerimientos No Funcionales³³

Son requerimientos que no se refieren directamente a las funciones específicas que entrega el software, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento. También definen las restricciones del software como la capacidad de entrada/salida y la representación de datos que se utiliza en las interfaces del software.

5.1 Generales

✓ Almacenamiento, modificabilidad y actualización

RNF1. El SPEI3 debe almacenar un código único, nombre de usuario y contraseña para cada uno de sus usuarios y esta debe quedar registrada en la base de datos del software.

RNF2. El SPEI3 deberá guardar el avance de cada alumno automáticamente siempre y cuando este haya completado y aprobado cualquier evaluación que sea necesaria para acceder a una lección posterior.

RNF3. El SPEI3 se diseñará incluyendo características sencillas de administrar que sean comprensibles para los usuarios docentes, para que de esta manera sean estos los dueños de la información que en el software se utilice.

✓ Soporte

RNF4. El presente proyecto de desarrollo de software no incluye la fase de soporte técnico sin embargo se incluirá la respectiva documentación en forma completa que sea necesaria para su correcto funcionamiento

✓ Documentación

RNF5. Se deberá realizar al final de la construcción del SPEI3 los respectivos manuales entre los que se detallan: Manual de Usuario, Manual de Instalación y manual de Configuración.

5.2 Requerimientos Operativos

✓ Desempeño-

RNF6. Para asegurar la asimilación de los contenidos del idioma inglés se considerarán tiempos prudenciales necesarios para la buena comprensión del contenido.

³³ Ingeniería de Software, Ian Sommerville, Cap 5.

✓ **Manejo de errores**

RNF7. Todas las entradas deben contar con su respectiva validación para asegurarse que la información procesada es la correcta, en su defecto se debe informar del error obtenido al usuario

RNF8. El software deberá utilizar alertas personalizadas de acuerdo al tipo de usuario, así los mensajes de error se enviarán por medio de sonidos característicos y descriptivos al alumno y como mensajes visuales descriptivos cuando el usuario sea el docente.

✓ **Requerimientos de Interfaz**

RNF9. Usuario Alumno: Se requiere que la interfaz del usuario para la reproducción de los temas, ejercicios, evaluaciones e instrucciones y ayudas sea netamente auditiva y táctil (salidas y entradas respectivamente) haciendo uso de audífonos³⁴ y el teclado convencional para que se acople a sus capacidades sensoriales.

RNF10. Usuario Docente: Se debe contar con una interfaz gráfica que resulte sencilla de utilizar para el usuario docente y que cuente con las ayudas necesarias para la comprensión de los objetivos de las pantallas. Las pantallas que conformen dicha interfaz deberán estar diseñadas bajo un mismo estándar de apariencia.

RNF11. Usuario Administrador: Se requiere que la interfaz gráfica del usuario administrador denote de forma clara las actividades administrativas del mismo y que cuente con las ayudas necesarias para la comprensión de los objetivos de las pantallas. Las pantallas que conformen dicha interfaz deberán estar diseñadas bajo un mismo estándar de apariencia.

✓ **Portabilidad**

No se tomará en cuenta el grado de portabilidad para este software ya que según entrevistas con los docentes el equipo actual y uno que están por recibir tienen sistema operativo MS Windows por lo que es en esta plataforma sobre la cual se desarrollará, no hay ninguna consideración adicional que realizar al respecto.

✓ **Disponibilidad**

RNF12. El software SPEI3 utilizará una arquitectura de red basada en el modelo cliente-servidor con efecto de que cualquier modificación hecha en el contenido o en los temas sea automáticamente reflejada en los demás equipos.

RNF13. Un usuario registrado podrá iniciar sesión en cualquier equipo dentro de la red de cómputo del recinto en que esté instalado el software.

RNF14. El SPEI3 debe contar con una gama de contenidos de muestra para demostrar la funcionalidad del software

³⁴ Ver Glosario de Términos

✓ **Seguridad**

RNF15. El software SPEI3 debe contar con un apartado especial para la seguridad, debe incorporar niveles de seguridad por usuario ya que esto asegura que ciertas tareas o actividades dentro del software estén disponibles para un tipo de usuario pero oculta para otro tipo de usuarios que no deben ver dicha información.

5.3 Requerimientos de Desarrollo

✓ **Hardware**

Hardware de Desarrollo

El hardware utilizado para el desarrollo del software se puede ver detallado en la siguiente tabla de especificación técnica

Hardware	Descripción
Procesador	Pentium IV de 2.6 GHz
Memoria Principal	256 MB
Almacenamiento Secundario	40 GB
Impresor	De Inyección de Tinta
Quemador de CD	Quemador CD 52X con lector de DVD
Scanner	De calidad normal tamaño carta.
Multimedia	Tarjeta de sonido y bocinas
UPS	600VA, 15Min

TABLA 3-2

Hardware de Operación

El hardware a ser utilizado cuando el software esté en operación será los equipos ubicados dentro del centro de cómputo del departamento de idiomas y cuenta con las siguientes características:

Hardware	Descripción
Procesador	Pentium IV de 2.6 GHz
Memoria Principal	256 MB
Almacenamiento Secundario	40 GB
Multimedia	Tarjeta de sonido y bocinas o audífonos
UPS	600VA, 15Min

TABLA 3-3

✓ Software

En esta sección se detalla los diferentes requerimientos de desarrollo que se deben especificar para la posterior etapa de construcción

Lenguaje de desarrollo: MS Visual Basic .Net³⁵

Este es un lenguaje de programación orientado a objetos que se puede considerar una evolución de Visual Basic implementada sobre el framework .NET.

Al estar implementado en el framework de .Net permite utilizar toda la funcionalidad que brinda esta muy completa plataforma la cual además es muy usada actualmente ya que la mayoría de desarrolladores de esta tecnología utilizan este lenguaje de programación.

Entorno de desarrollo: Microsoft Visual Studio .Net³⁶

Es un entorno integrado de desarrollo (IDE, por sus siglas en inglés) para sistemas Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones de escritorio, sitios y aplicaciones Web, así como servicios Web en cualquier entorno que soporte la plataforma .NET. Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas Web y dispositivos móviles.

Entorno de Ejecución: .Net Framework³⁷

A continuación se resumen las ventajas más importantes que proporciona .Net Framework:

- **Código administrado:** El CLR realiza un control automático del código para que este sea seguro.
- **Interoperabilidad multilenguaje:** El código puede ser escrito en cualquier lenguaje compatible con .Net siempre se compila en código intermedio (MSIL).
- **Compilación just-in-time:** El compilador JIT incluido en el Framework compila el código intermedio (MSIL) generando el código máquina propio de la plataforma.
- **Garbage collector:** El CLR proporciona un sistema automático de administración de memoria denominado recolector de basura.
- **Seguridad de acceso al código:** Se puede especificar que una pieza de código tenga permisos de lectura de archivos pero no de escritura.
- **Despliegue:** Por medio de los ensamblados resulta mucho más fácil el desarrollo de aplicaciones distribuidas y el mantenimiento de las mismas.

³⁵ Microsoft Visual Basic .Net http://es.wikipedia.org/wiki/Visual_Basic.NET

³⁶ Microsoft Visual Studio .Net http://es.wikipedia.org/wiki/Microsoft_Visual_Studio

³⁷ Algunas de las ventajas e inconvenientes de la plataforma .Net. <http://www.desarrolloweb.com/articulos/1329.php>

Sintetizador de Voz: L&H TTS3000³⁸

Durante los últimos años ha surgido un nuevo tipo de interfaces humano-computadora, que combina la más reciente tecnología del lenguaje para procurar un mejor acceso y transferencia de información a través del habla.

Es notable que software de este tipo necesite hacer uso de un sintetizador de voz como apoyo, ya que finalmente lo que se busca es dar la información que sea necesaria al usuario en forma verbal. Por lo tanto los sintetizadores de voz también son un recurso importante para el desarrollo.

Lernout & Hauspie TTS3000: Este es el sintetizador seleccionado y fue creado por Lernout & Hauspie Speech Products el cual se utilizará con la mecánica de software de tercero el cual es de libre distribución y es proporcionado gratuitamente por Microsoft ©

Este recurso de software será utilizado mediante una de las ventajas que se mencionaron anteriormente del entorno de ejecución, como lo es la interoperabilidad multilinguaje, en el cual dicho recurso de software utiliza esta característica que brinda el .net framework para la comunicación con tecnologías ActiveX de modo que esta acción resulta ser transparente al desarrollador.

Cadenas de texto³⁹

Un sistema de texto a voz se compone de dos partes: un front-end y un back-end. A grandes rasgos, el front-end toma como entrada cadenas de texto y produce una representación lingüística fonética. El back-end toma como entrada la representación lingüística simbólica y produce una forma de onda sintetizada.

El front-end desempeña dos tareas principales. Primero, toma las cadenas de texto y convierte partes problemáticas como números y abreviaturas en palabras equivalentes. Este proceso se llama a menudo normalización de texto o preprocesado. Entonces asigna una transcripción fonética a cada palabra, y divide y marca el texto en varias unidades prosódicas, como frases y oraciones. El proceso de asignar transcripciones fonéticas a las palabras recibe el nombre de conversión texto a fonema (TTP en inglés) o grafema a fonema (GTP en inglés). La combinación de transcripciones fonéticas e información prosódica constituye la representación lingüística fonética.

Gestor de base de datos: PostgreSQL

PostgreSQL es un potente sistema de base de datos relacional open source. Tiene más de 15 años de activo desarrollo y arquitectura probada que se ha ganado una muy buena reputación por su confiabilidad e integridad de datos. Funciona en todos los sistemas operativos importantes: Linux, UNIX y Windows. Posee soporte total para foreign keys, joins, views, triggers, y stored procedures (en múltiples lenguajes). Incluye la mayoría de los tipos de datos SQL92 y SQL99; INTEGER,

³⁸ Información del sintetizador de voz L & H TTS3000, Pagina 6
http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/navarro_g_c/capitulo2.pdf

³⁹ información de sintetización de voz
http://es.wikipedia.org/wiki/Sintetizador_de_voz

NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También soporta almacenamiento de objetos grandes binarios, incluyendo gráficos, sonidos, o vídeos. Tiene interfaces de programación nativos para C/C++, Java, Perl, Python, Ruby, Tcl, ODBC , entre otros, y extensa documentación.

Las tecnologías .Net incluyen un modelo de acceso a datos denominado ADO .Net el cual toma muchos elementos de su antecesor ADO (ActiveX Data Objects), lamentablemente, ADO .Net solamente incluyo soporte nativo a su base de datos SQL Server, el acceso a otros gestores de base de datos debe de hacerse a través de ODBC, lo cual resulta muy ineficiente. Esta situación obligaba a los usuarios .Net a usar SQL como repositorio de datos si deseaban obtener un desempeño aceptable en sus aplicaciones. Con el tiempo, desarrollos ajenos a Microsoft implementaron librerías de clases para poder acceder de forma directa a otras bases de datos como Oracle, mySQL y postgresQL, algunas de estas librerías son propietarias y otras son de código abierto. Para realizar el acoplamiento de PostgreSQL con .Net se necesita una librería que implemente las clases necesarias para acceder de forma nativa a los datos almacenados en postgresQL.

Interfaces de conexión: Npgsql

Npgsql es un Proveedor de datos para la plataforma .Net permitiendo a cualquier programa desarrollado en el .Net Framework acceder a un servidor de base datos postgresQL. La funcionalidad provista por este componente al .Net Framework es equivalente a la que provee el driver JDBC para la plataforma Java. Npgsql implementa completamente el protocolo de comunicación de PostgreSQL y esta escrito completamente en C#. Npgsql soporta las versiones 7.x y 8.x.

En la figura 3.1, se muestra un esquema de la integración de las herramientas de software para el desarrollo del SPEI3.

INTEGRACION DE HERRAMIENTAS DE

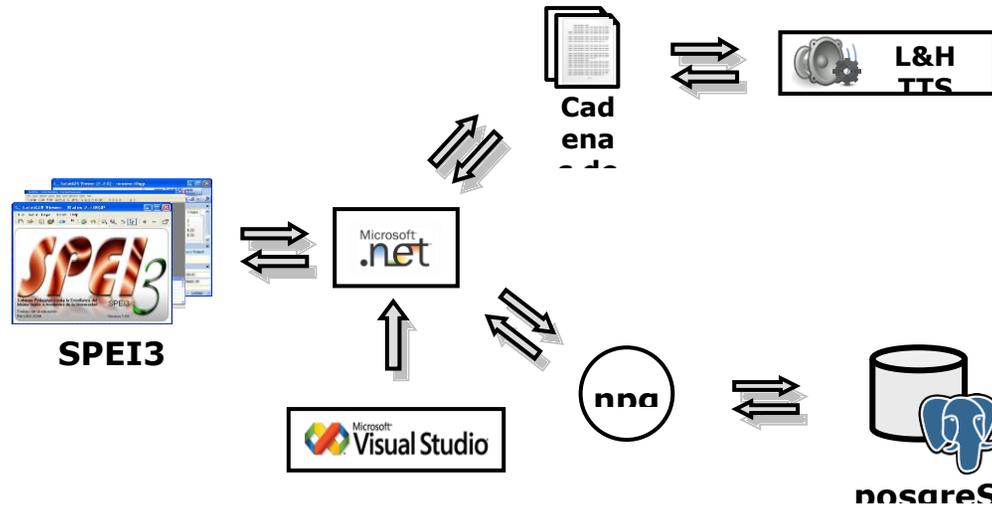


Figura 3.1 Integración de herramientas de desarrollo

Capítulo IV:

Análisis del Sistema

1. Casos de Uso

El Unified Modeling Language (UML) es un lenguaje gráfico, semiformal, que ha sido aceptado como estándar para describir sistemas de software orientados a objetos.

UML define varios tipos de diagramas que se utilizan para describir diferentes aspectos o vistas de un sistema. Los diagramas de Casos de Uso se utilizan para capturar los requerimientos de los sistemas y guiar su proceso de desarrollo. Los distintos Casos de Uso que se definen a lo largo de un proceso de desarrollo no son independientes sino que es posible establecer relaciones entre ellos. Las principales relaciones consideradas por UML son: Generalización (Generalization), Inclusión (Include) y Extensión (Extend)⁴⁰.

Los casos de uso como se explico en el apartado anterior, se utilizaron para especificar los requerimientos de sistema.

En esta sección se define la metodología utilizada para la elaboración de los casos de uso y se finaliza con la especificación de todos los casos de uso del proyecto.

1.1 Definición⁴¹

En ingeniería del software, un caso de uso es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización de software. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico. Normalmente, en los casos de usos se evita el empleo de jergas técnicas, prefiriendo en su lugar un lenguaje más cercano al usuario final. En ocasiones, se utiliza a usuarios sin experiencia junto a los analistas para el desarrollo de casos de uso.

En otras palabras, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo, por ejemplo la relación y la generalización son relaciones. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar como reacciona una respuesta a eventos que se producen en el mismo.

⁴⁰ Revista Colombiana de Computación, Volumen 1, número 1, Págs. 73

⁴¹ Información de Casos de Uso http://es.wikipedia.org/wiki/Caso_de_uso

1.2 Metodología para la elaboración de los casos de uso

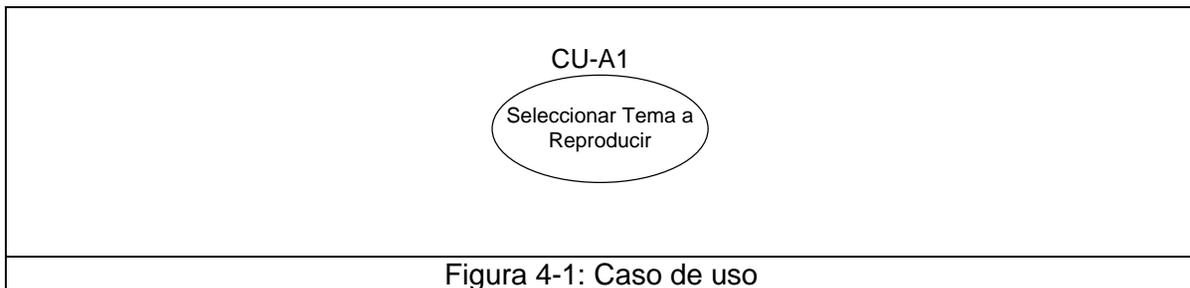
Para crear los casos de uso se siguieron los siguientes pasos:

1. Elaborar los diagramas de casos de uso

Partiendo de los requerimientos obtenidos de los usuarios, se dividió por subsistemas las funciones principales del sistema esta identificación de subsistemas fue realizada por el grupo de desarrollo en el cual se identificó los siguientes subsistemas:

- Subsistema de lecciones (A)
- Subsistema de ejercicios (B)
- Subsistema de evaluaciones (C)
- Subsistema de control (D)

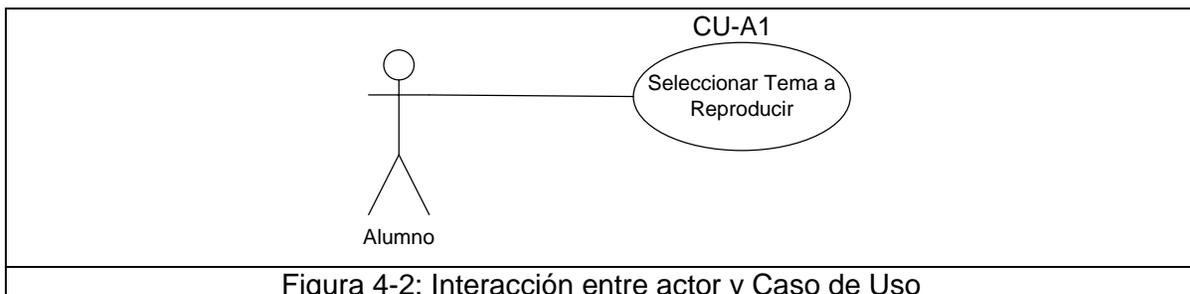
A estos subsistemas se les asignó una letra que sirve para identificar que casos de uso lo conforman.



La figura 4-1 muestra claramente lo antes mencionado: CU-A1 es el identificador del caso de uso y debe leerse de la forma “Caso de uso perteneciente al subsistema de lecciones numero uno”; el nombre esta encerrado dentro del elipse que representa al caso de uso, en este ejemplo el nombre es “Seleccionar tema a reproducir”.

Los casos de uso interactúan con actores, un actor es un ente que interactúa con el caso de uso. En el SPEI3 se identificaron tres actores estos son: alumno, docente y administrador de docentes.

El actor que interactúa con el caso de uso de ejemplo es el alumno, esto se especifica como en la figura 4-2, las líneas que unen al actor con el caso de uso representa la relación entre los dos.



2. Identificar los escenarios

En este paso se identifica el curso normal de eventos al realizar un caso de uso, es decir se enumeran los pasos necesarios que implica un caso de uso, además por cada curso normal se identifica posibles cursos alternativos o pasos a seguir si no se lleva a cabo un curso normal de eventos.

3. Especificar el caso de uso

Para ello se utilizo el formato que se detalla a continuación el cual se describe cada elemento del caso de uso.⁴²

CU-SS####	Nombre del caso de uso	
Actores	Nombre del actor o actores..	
Requerimientos asociados	Código de los requerimientos que se relacionan con el caso de uso.	
Descripción	Descripción breve y clara del caso de uso.	
Precondición	Estado necesario para iniciar con este caso de uso.	
Secuencia Normal	Paso	Acción
	1	Pasos de la secuencia normal.
	
	
	N	Ultimo paso.
Post-condición	Estado resultante después del casos de uso.	
Excepciones	Paso	Acción
	1.a	Paso de una secuencia alternativa por excepciones del curso normal.
Frecuencia esperada	Estimado de ocurrencias del caso de uso por un tiempo definido: Diaria, semanal, mensual, por ciclo, anual.	
Importancia Comentarios	Grado cualitativo de la importancia del caso de uso. Comentarios adicionales que ayuden a comprender el caso de uso.	

⁴² Documento de Requerimientos del Sistema
Apuntes de Documento de Requerimientos,
Dr. Narciso Cerpa.

Consideraciones previas a la especificación de casos de uso para el no vidente

- El software no imprimirá ni utilizará en ningún momento material en braille para interactuar con el invidente.
- El invidente interactuará con el software únicamente utilizando el teclado para enviar sus entradas y los audífonos para percibir las salidas del software. Debido a las limitaciones del invidente es abolido el uso del Mouse como dispositivo de entrada.
- El software guiará de forma audible al invidente a través de los diferentes menús e interfaces de reproducción indicándole las teclas precisas para seleccionar las opciones.
- Es importante aclarar que el software estará orientado a reforzar y mejorar la comprensión auditiva y el habla del idioma inglés no así la escritura. Esto significa que, en menor medida, el estudiante adquirirá conocimientos de escritura de palabras que se encuentren dentro de los ejercicios a desarrollar.

2. Diagramas de Casos de Uso

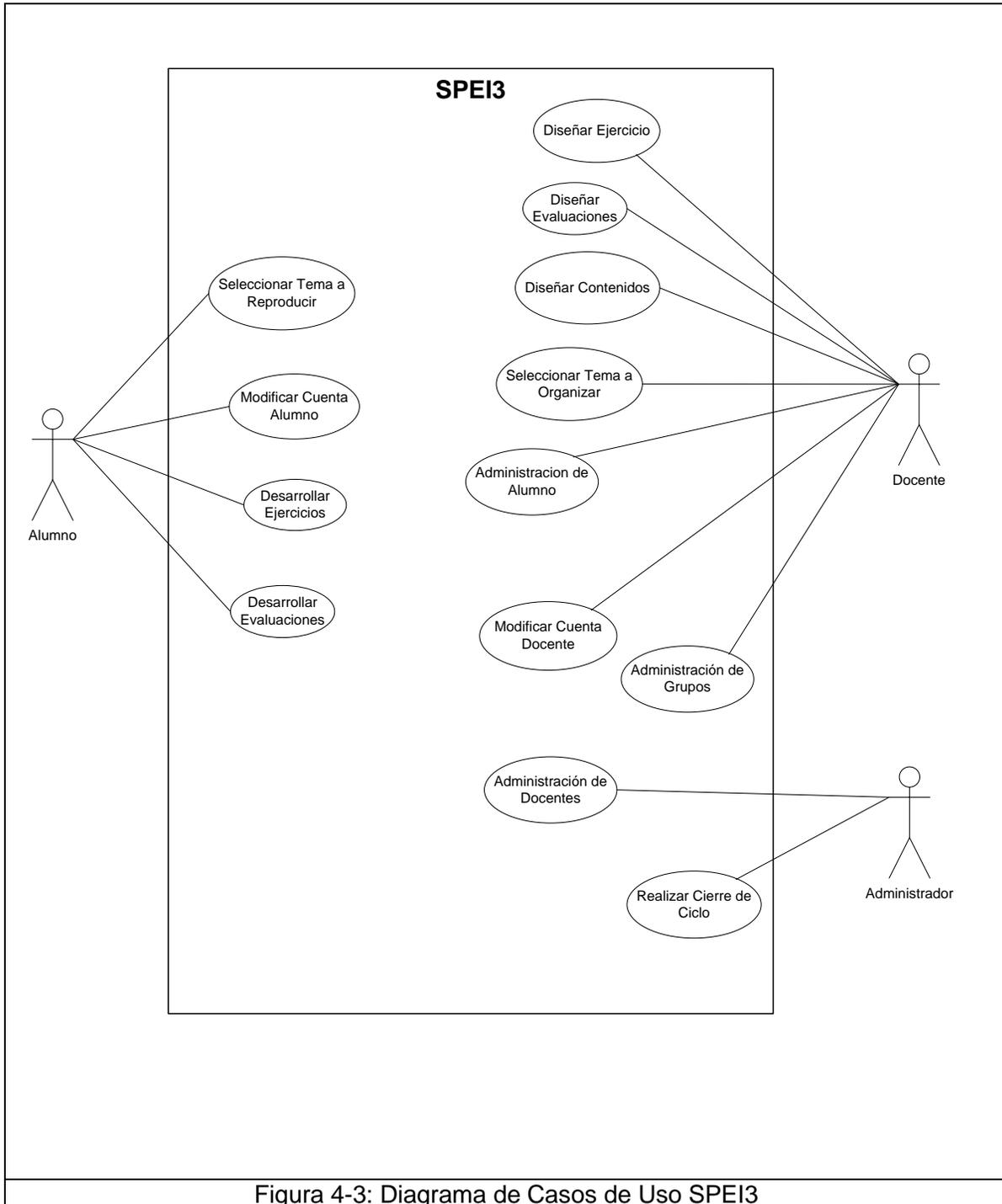


Figura 4-3: Diagrama de Casos de Uso SPEI3

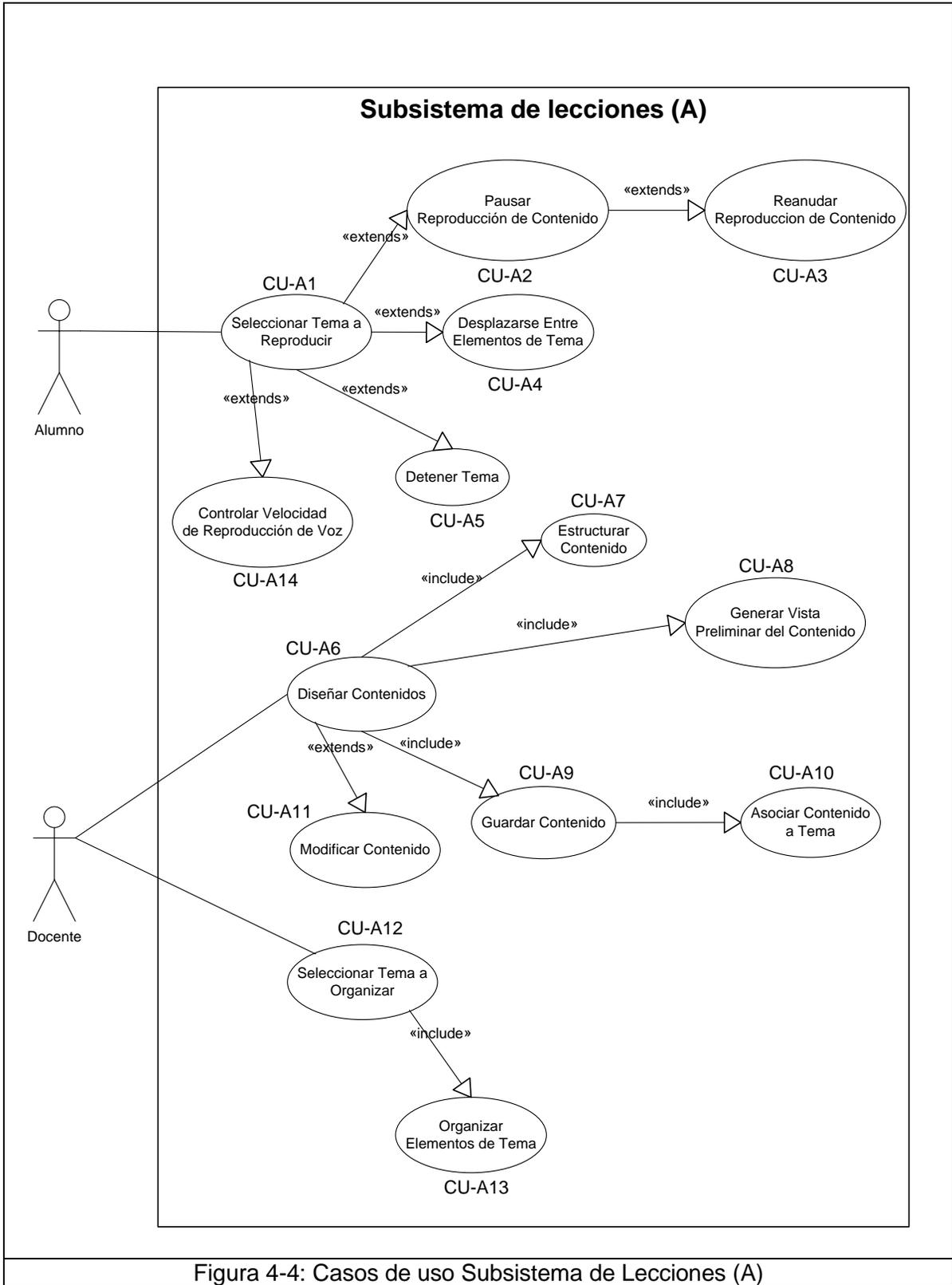


Figura 4-4: Casos de uso Subsistema de Lecciones (A)

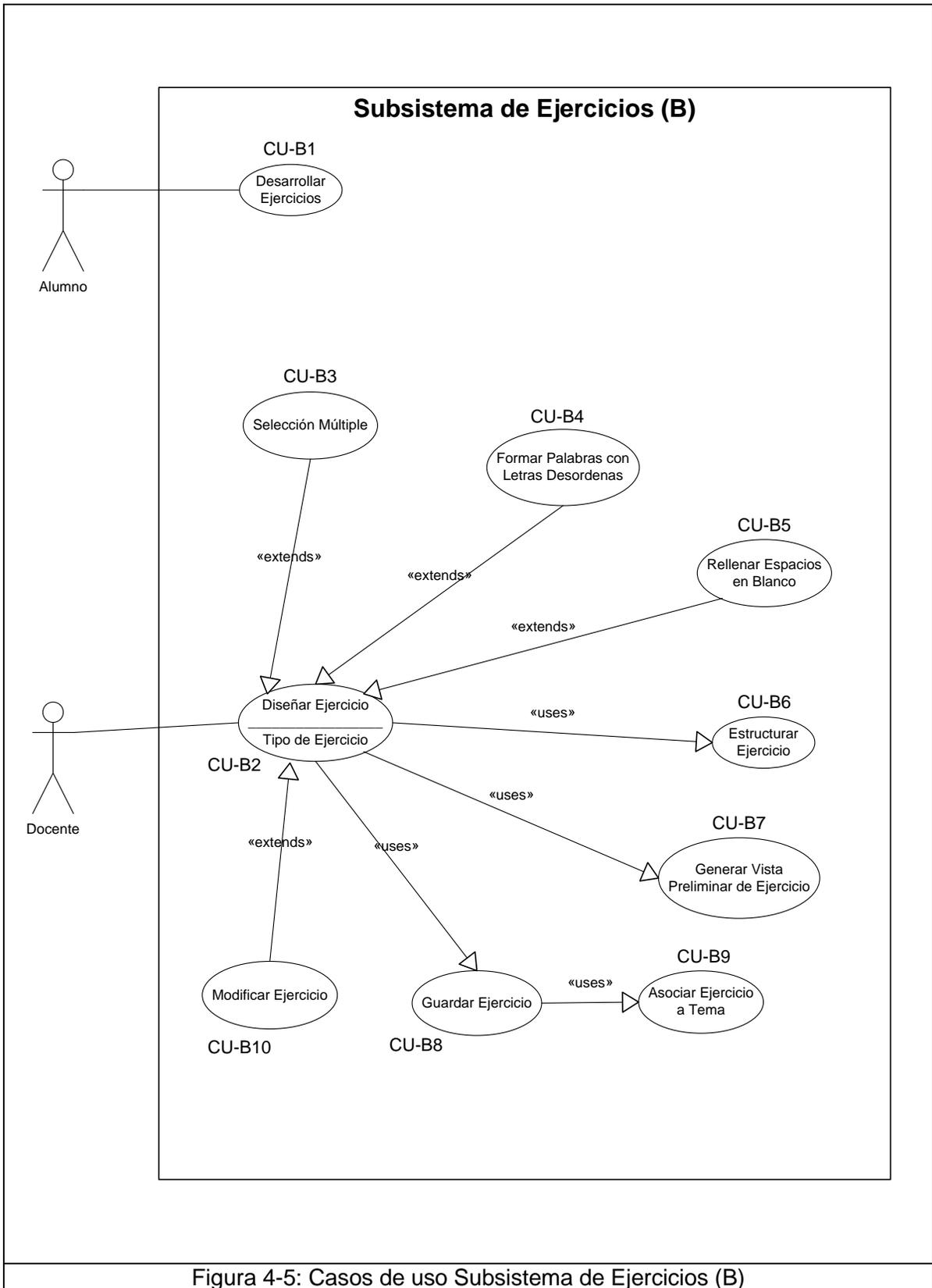


Figura 4-5: Casos de uso Subsistema de Ejercicios (B)

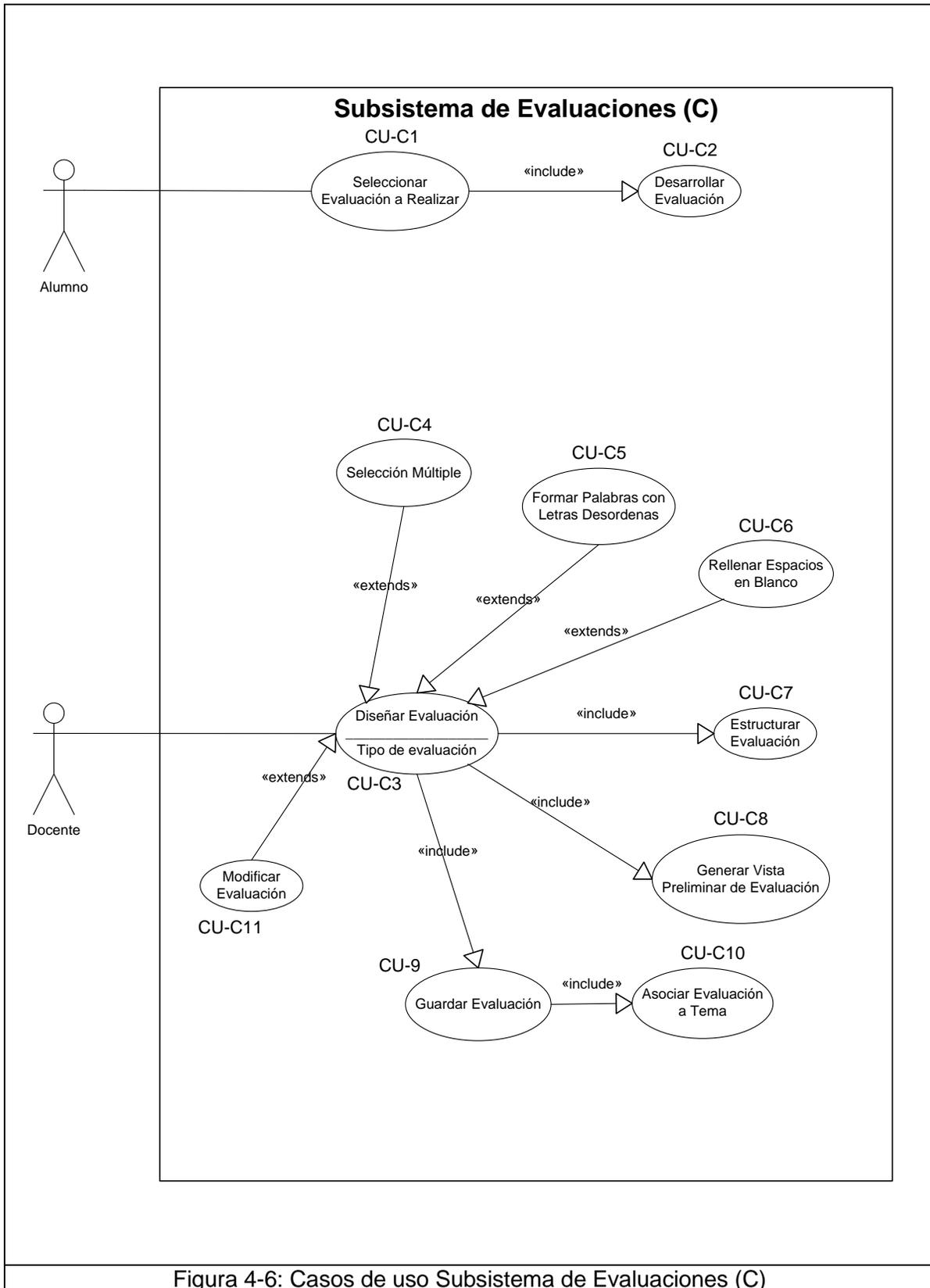


Figura 4-6: Casos de uso Subsistema de Evaluaciones (C)

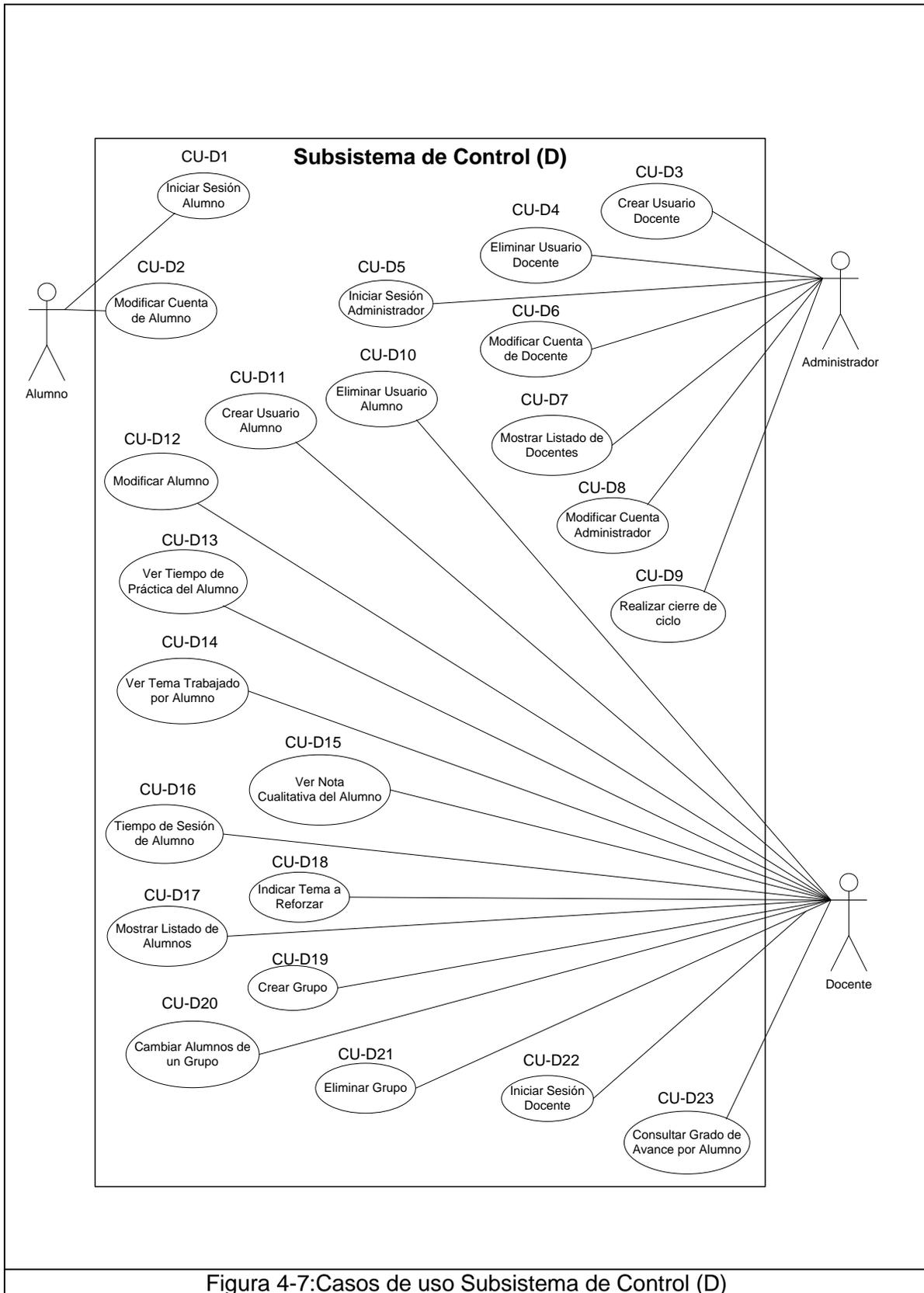


Figura 4-7: Casos de uso Subsistema de Control (D)

3. Actores del sistema

AC-01	Alumno
Descripción	Este actor representa a los alumnos invidentes del departamento de idiomas de la Universidad de El salvador que harán uso del SPEI3
Importancia Comentarios	Vital Existirá un usuario de este tipo por cada alumno no vidente.

AC-02	Docente
Descripción	Este actor representa a los docentes del departamento de idiomas de la Universidad de El salvador. Quienes impartirán clases y administrarán sus alumnos a través del SPEI3
Importancia Comentarios	Vital Existirá un usuario de este tipo por cada docente que esté involucrado con alumnos no videntes.

AC-03	Administrador
Descripción	Este actor representa a la persona que llevará el control general sobre algunos aspectos vitales del SPEI3 como por ejemplo los cierres de ciclo y la administración y creación de docentes.
Importancia Comentarios	Vital Existirá únicamente con el fin de administrar a los docentes, se sugiere que sea el coordinador de docentes.

4. Subsistemas

SS-A	Subsistema de Lecciones
Descripción	Este subsistema proveerá las funciones necesarias para la creación y reproducción de lecciones.
Importancia Comentarios	Vital Ninguno

SS-B	Subsistema de Ejercicios
Descripción	Este subsistema proveerá las funciones necesarias para la creación de ejercicios por parte del docente y permitirá el desarrollo de los mismos por parte del alumno.
Importancia Comentarios	Vital ninguno

SS-C	Subsistema de Evaluaciones
Descripción	Este subsistema proveerá las funciones necesarias para la creación de evaluaciones por parte del docente y permitirá el desarrollo de las mismas por parte del alumno.
Importancia Comentarios	Vital ninguno

SSD	Subsistema de Control
Descripción	Este subsistema proveerá las funciones administrativas del SPEI3 para los tres tipos de usuario (Alumno, docente y administrador).
Importancia Comentarios	Vital ninguno

4.1 Casos de Uso del Subsistema de Lecciones (A)

CU-A1	Seleccionar tema a reproducir	
Actores	Alumno.	
Requerimientos asociados	RF-1.	
Descripción	Este caso de uso le permite al usuario especificar el tema a ser reproducido.	
Precondición	El alumno ha iniciado sesión satisfactoriamente en el software SPEI3 y ha seleccionado reproducir tema.	
Secuencia Normal	Paso	Acción
	1	El SPEI3 comunica al usuario las unidades a las que puede tener acceso.
	2	El usuario selecciona la unidad.
	3	El SPEI3 informa al usuario las lecciones a las que puede tener acceso.
	4	El usuario selecciona la lección.
	5	El SPEI3 comunica al usuario los temas que conforman la lección seleccionada.
	6	El usuario selecciona el tema a reproducir.
	7	El SPEI3 reproduce el tema seleccionado.
Post-condición	El tema ha sido seleccionado y es reproducido.	
Excepciones	Ninguno.	
Frecuencia esperada	Diaria.	
Importancia	Vital.	
Comentarios	Ninguno.	

Para consultar todos los casos de uso (especificación de requerimientos) puede hacerlo desde el CD adjunto a este documento en la opción “Casos de uso”.

Capítulo V: Diseño

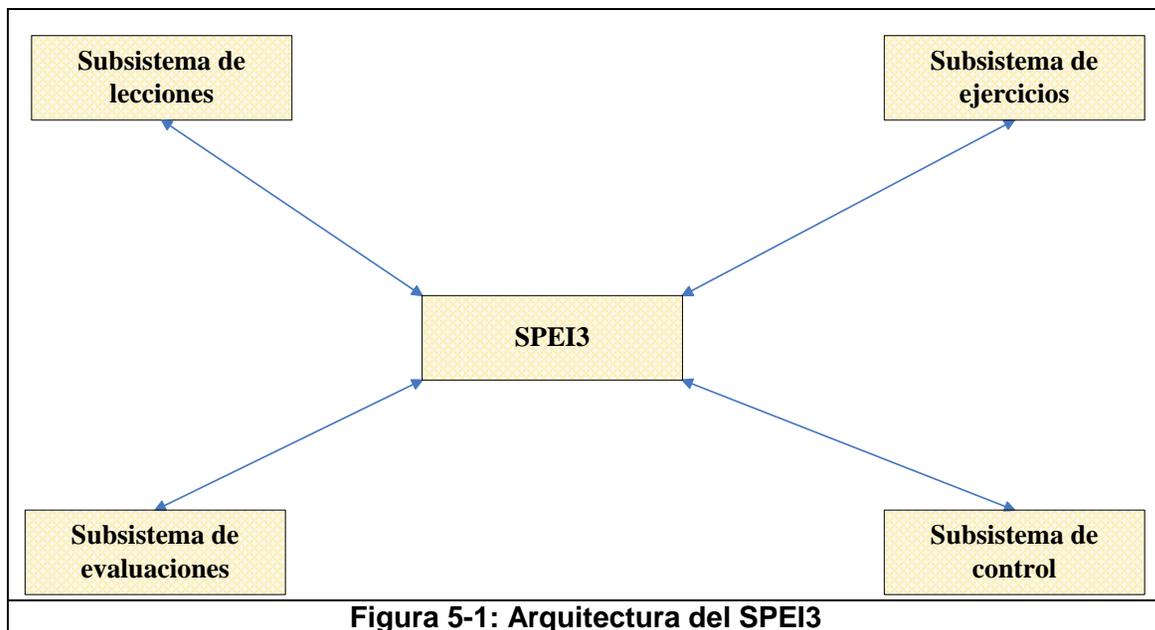
1. Arquitectura del software

El SPEI3 lo podemos estructurar en funciones básicas, en este apartado se muestra el diseño arquitectónico del software, este se realiza mediante la descripción de todos los componentes identificados en el análisis.

La funcionalidad que contendrá el software la podemos dividir en operaciones agrupadas en subsistemas, cada uno de los subsistema cuenta con una función básica que fueron identificadas en el análisis.

Los subsistemas para el SPEI3 son los siguientes:

1. Subsistema de lecciones
2. Subsistema de ejercicios
3. Subsistema de evaluaciones
4. Subsistema de control



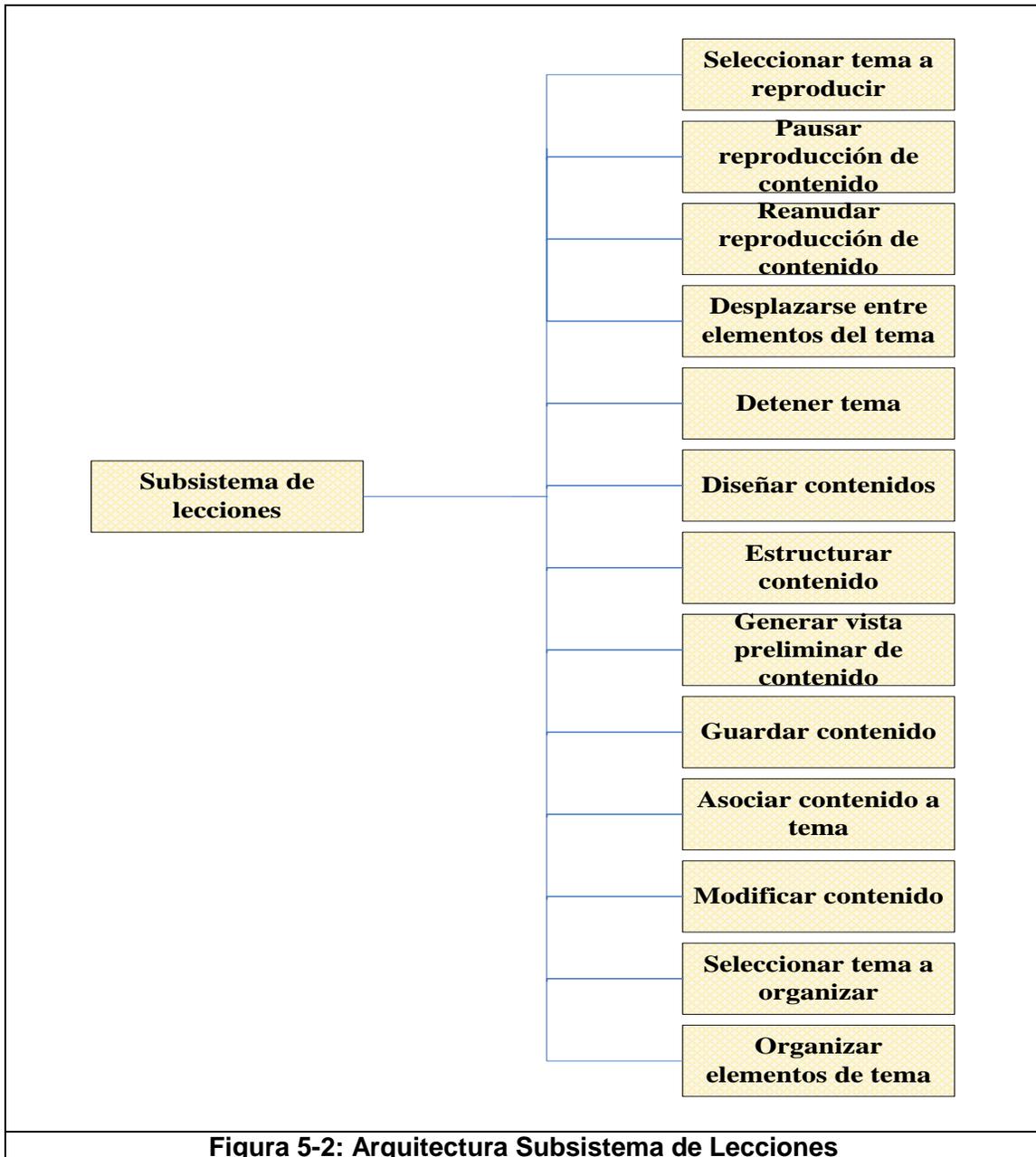
Descripción de subsistemas

1.1. Subsistema de lecciones

Este subsistema comprende la administración las lecciones, una lección esta formada por varios temas y estos temas están compuestos por varios contenidos, a partir de estos dos conceptos (tema y contenido) y de los requerimientos de los usuarios se crearon los siguientes componentes que estructuran el subsistema de lecciones.

- 1.1 Seleccionar tema a reproducir
- 1.2 Pausar reproducción de contenido
- 1.3 Reanudar reproducción de contenido
- 1.4 Desplazarse entre elementos del tema

- 1.5 Detener tema
- 1.6 Diseñar contenidos
- 1.7 Estructurar contenido
- 1.8 Generar vista preliminar de contenido
- 1.9 Guardar contenido
- 1.10 Asociar contenido a tema
- 1.11 Modificar contenido
- 1.12 Seleccionar tema a organizar
- 1.13 Organizar elementos de tema



1.2. Subsistema de ejercicios

Este subsistema maneja las operaciones a realizar por el software sobre los ejercicios que son partes de los temas, estos ejercicios son administrados por el usuario docente, los componentes de este subsistema son:

- 2.1 Desarrollar ejercicios
- 2.2 Diseñar ejercicio
- 2.3 Selección Múltiple
- 2.4 Formar palabras con letra desordenadas
- 2.5 Rellenar espacios en blanco
- 2.6 Estructurar ejercicio
- 2.7 Generar vista preliminar de ejercicio
- 2.8 Guardar ejercicio
- 2.9 Asociar ejercicio a tema
- 2.10 Modificar ejercicio

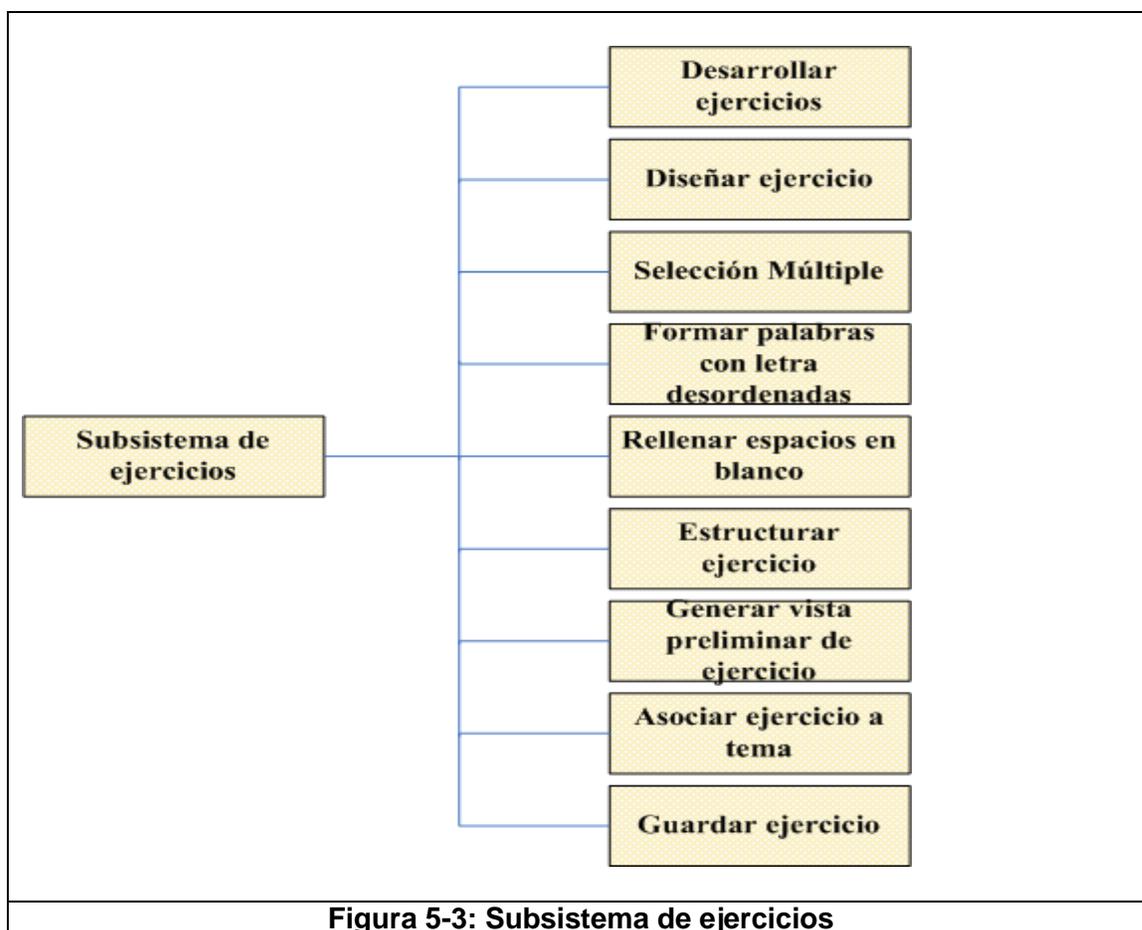
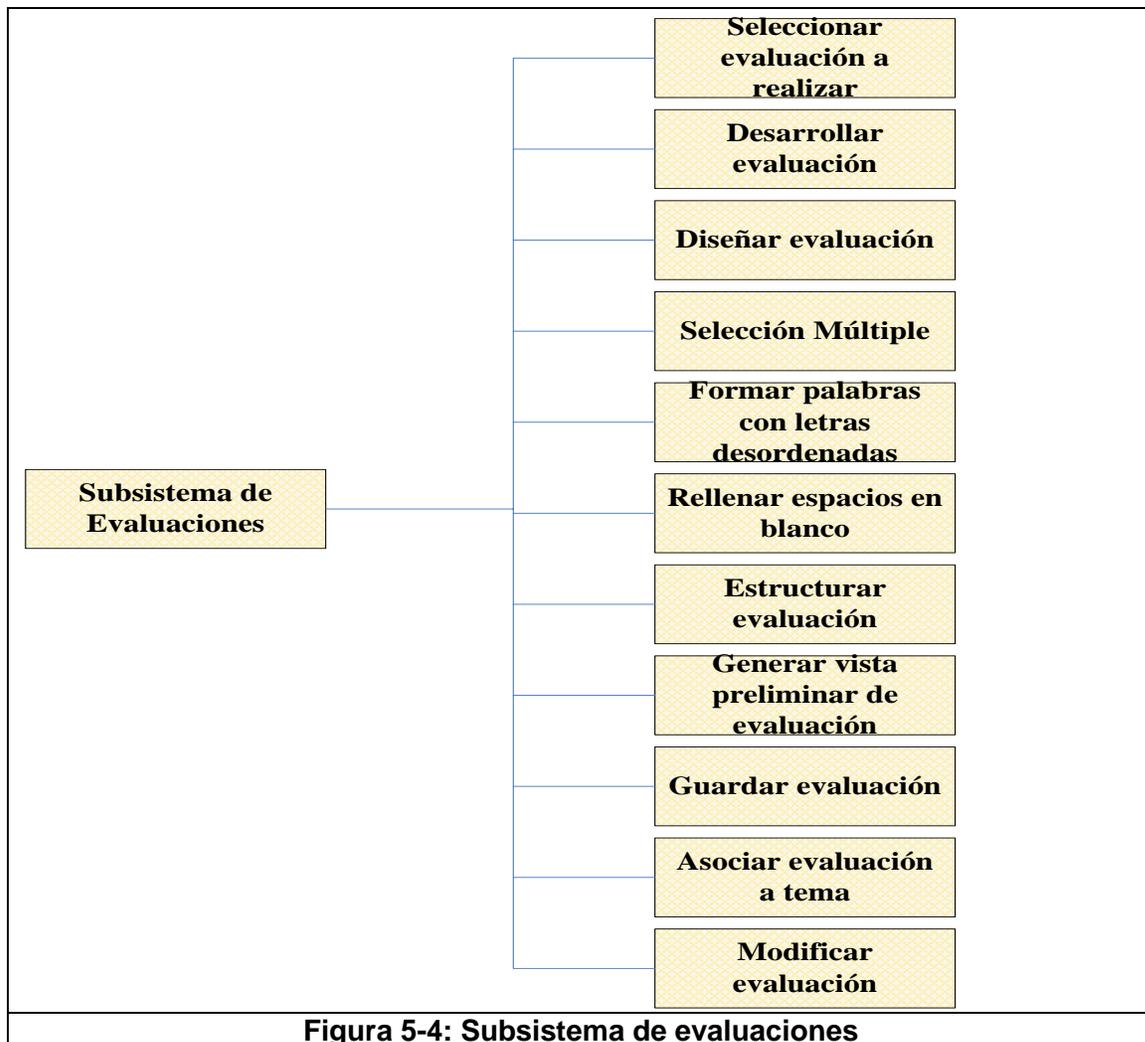


Figura 5-3: Subsistema de ejercicios

1.3. Subsistema de Evaluaciones

Este componente trata los aspectos de operación sobre las evaluaciones tomando en cuenta que una evaluación se crea a partir de consideraciones que corresponden al usuario docente, y además de un grupo de temas se crea una evaluación, los componentes que forman este subsistema son.

- 3.1 Seleccionar evaluación a realizar
- 3.2 Desarrollar evaluación
- 3.3 Diseñar evaluación
- 3.4 Selección Múltiple
- 3.5 Formar palabras con letras desordenadas
- 3.6 Rellenar espacios en blanco
- 3.7 Estructurar evaluación
- 3.8 Generar vista preliminar de evaluación
- 3.9 Guardar evaluación
- 3.10 Asociar evaluación a tema
- 3.11 Modificar evaluación



1.4. Subsistema de Control

Este subsistema regula el acceso a los subsistemas anteriores, definiéndose en el la estructuras para el control de los diferentes usuarios (alumno, docente y administrador) esta formado por:

- 4.1 Iniciar sesión alumno
- 4.2 Modificar cuenta alumno
- 4.3 Crear usuario docente
- 4.4 Eliminar usuario docente
- 4.5 Iniciar sesión administrador
- 4.6 Modificar cuenta de docente
- 4.7 Modificar listados de docente
- 4.8 Modificar cuenta administrador
- 4.9 Realizar cierre de ciclo
- 4.10 Eliminar usuario alumno
- 4.11 Crear usuario alumno
- 4.12 Modificar alumno
- 4.13 Ver tiempo de práctica del alumno
- 4.14 Ver tema trabajado por alumno
- 4.15 Ver nota cualitativa del alumno
- 4.16 Tiempo de sesión alumno
- 4.17 Mostrar listado alumnos
- 4.18 Indicar tema a reforzar
- 4.19 Crear grupo
- 4.20 Cambiar alumnos de un grupo
- 4.21 Eliminar grupo
- 4.22 Iniciar sesión docente
- 4.23 Consultar grado de avance alumno

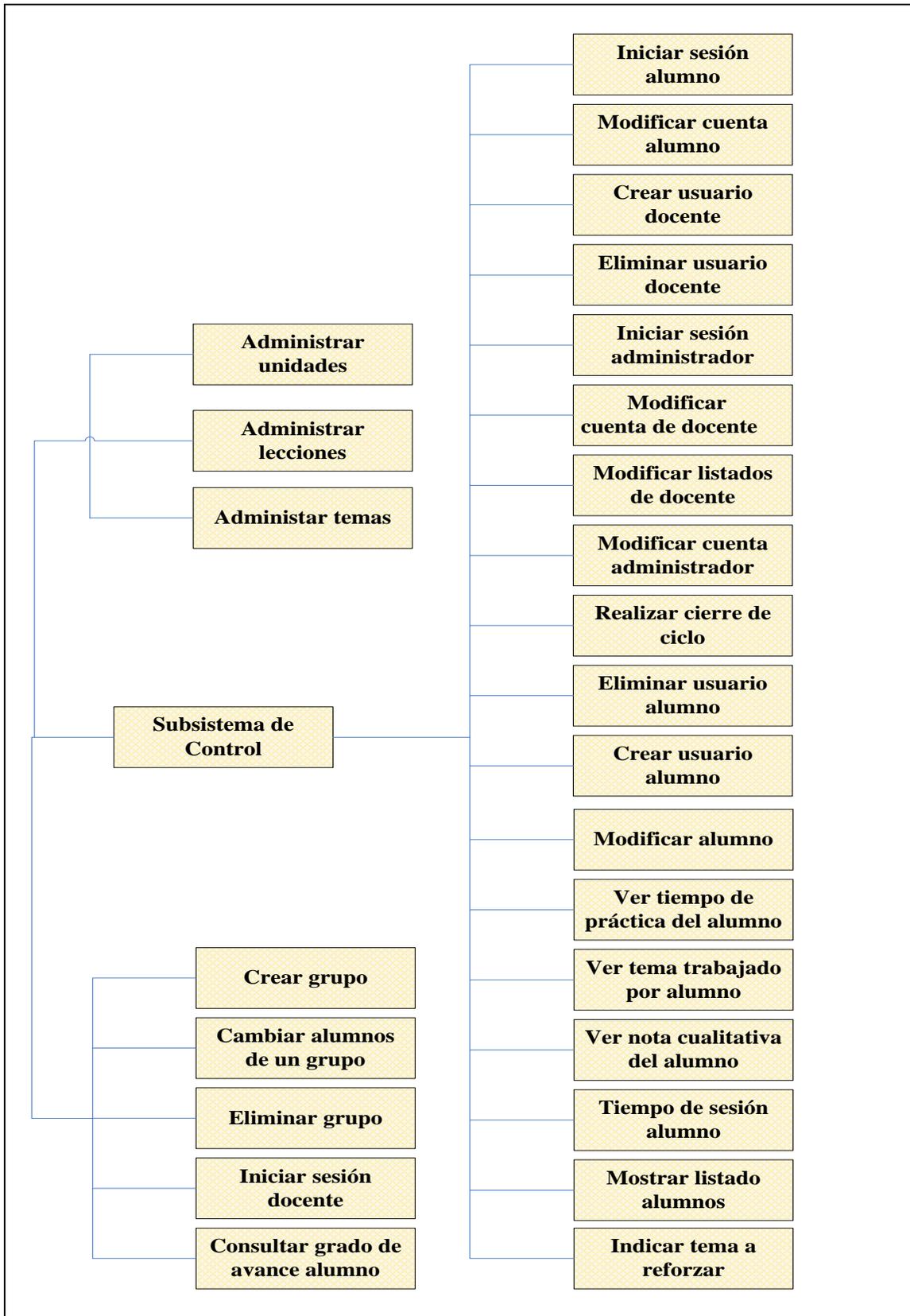


Figura 5-5: Subsistema de control

2. Estándares

La Estandarización o Normalización es el proceso de elaboración, aplicación y mejora de las normas que se aplican a distintas actividades con el fin de ordenarlas y mejorarlas. Se define la normalización como el proceso de formular y aplicar reglas para una aproximación ordenada a una actividad específica para el beneficio y con la cooperación de todos los involucrados.

Según la ISO (International Organization for Standardization) la Normalización es la actividad que tiene por objeto establecer, ante problemas reales o potenciales, disposiciones destinadas a usos comunes y repetidos, con el fin de obtener un nivel de ordenamiento óptimo en un contexto dado.

La documentación de cualquier proyecto de software es un punto clave que puede definir en gran parte la calidad del software. Esto es especialmente cierto para software que deba ser reutilizado a nivel de interfase (caja negra). La definición de estándares permite que las distintas etapas del proyecto se realicen bajo normas específicas que faciliten la comprensión y la visión en el diseño y programación del software. Bajo este propósito se describen los estándares de diseño, codificación, documentación e interfaz.

2.1. Estándares de Diseño

UML es el lenguaje de modelado de sistemas de software más conocido en la actualidad; es el estándar internacional aprobado por la OMG (Object Management Group), consorcio creado en 1989 responsable de la creación, desarrollo y revisión de especificaciones para la industrial del software.

UML es un grupo de especificaciones de notación orientadas a Objeto, las cuales están compuesta por distintos diagramas, que representan las diferentes etapas del desarrollo de un proyecto de software. En el presente trabajo se utilizarán diagramas UML, de modo que es necesaria la definición de un estándar para su presentación.

Formato de diagramas

Todos los diagramas de UML a elaborar deben cumplir con las siguientes normas de formato:

- Fuente: Arial
- Tamaño: 10 pixeles⁴³
- Color: Negro
- Línea: Continua, 1pixel
- Fondo: Blanco / sin fondo

El formato se aplicará respetando las notaciones estándares ya aprobadas para diagramas UML y solo será modificado para su conveniente visualización.

⁴³ <http://support.microsoft.com/kb/253680/es> “La parte más pequeña de la pantalla del monitor es un punto cuadrado o rectangular que recibe el nombre de píxel. Un píxel se describe de forma más correcta como una unidad lógica, y no física, ya que el tamaño físico de un píxel individual lo determina el fabricante del monitor.”

2.2. Estándares de Codificación

Los estándares de codificación tienen como propósito facilitar el entendimiento entre las personas que crearán y modificarán código de tal manera que se sigan ciertas normas que permitan agilizar la depuración y mantenimiento del código fuente.

Nombres

Estos estándares comprenden las reglas de nombrado de variables, procedimientos, funciones, procedimientos, comentarios, y otros identificadores.

- Todo lo que necesite ser identificado por un nombre dentro del código se debe de hacer de tal manera que identifique el “qué” y no el “cómo”. Por ejemplo es mas claro nombrar un procedimiento de acceso a datos SeleccionarRegistro() que RealizarConsultaSelect(), porque lo que importa (para que otra persona entienda el código) es que se supone que hace el método y no como lo hace.
- Otra directiva es la de utilizar nombres tan largos como para que se entiendan pero a la vez tan cortos como para que no den información irrelevante, por ejemplo es mejor emplear SeleccionarLeccion() que SeleccionarlaLeccionMenu().

Estructuras (namespaces, procedimientos, clases, interfaces y propiedades)

- Los nombres de todas las estructuras de código deben ser en español.
- Los namespaces deben empezar por el nombre de la institución seguido del nombre de proyecto y la funcionalidad:

SPEI3.Ciclo.Unidad.AccesoDatos
[institución].[proyecto].[funcionalidad]

- El nombre del ensamblado y el namespace root deben ser idénticos
- El nombre de la clase y el archivo fuente deben ser iguales.
- No se debe usar la notación húngara, la cual prefija una abreviatura referente al tipo de objeto: lblAceptar (label), btnOK(Botón), etc.
- Se deben evitar nombres imprecisos que permitan interpretaciones subjetivas, como por ejemplo Pendiente nombre(), o bien taru1 para una variable. Tales nombres contribuyen más a la ambigüedad que a la abstracción.
- En la Programación Orientada a Objetos es redundante incluir nombres de clases en el nombre de las propiedades de clases, como por ejemplo: Verbos.VerbosIrregulares, en su lugar, utilice Verbos.Irregulares, pues el nombre de la clase ya contiene dicha información.
- Se debe de utilizar la técnica verbo-sustantivo para nombrar procedimientos que ejecuten alguna operación en un determinado objeto, como por ejemplo CalcularPromedio().
- Es importante incluir en los nombres de funciones o clases que deban ser reutilizables, algo que las distinga de funciones o clases implementadas en otras librerías o en código que utilice éstas librerías. La forma clásica de hacer esto es poniendo como prefijo de toda clase o función global exportada (pública), el nombre de la librería o su sigla en minúsculas (SIN mayúsculas al principio). En este caso se colocan todas las clases y funciones definidas en la librería dentro de un namespace con el nombre de la librería.
- Los nombres de clases comienzan con mayúsculas, separadas por mayúsculas, de la misma forma que las variables miembro.
- Los nombres de las Interfaces deben de comenzar con el prefijo "I", seguido de un nombre o una frase nominal, como IComponente, o con un adjetivo que describa el comportamiento de la interfaz, como IPersistible. No se debe de utilizar el guión bajo “_” (con la excepción de las variables privadas), y se deben de minimizar al máximo el

uso de las abreviaturas.

- Los procedimientos y funciones deben de ser lo más atómico posibles, es decir, deben de realizar un propósito específico y no ser “multipropósito”.
- Los métodos miembros de clases siguen la notación idéntica que para las variables miembro de clases, es decir, comenzando con/y separando las palabras con mayúsculas, salvo que es en este caso preferible omitir la variación que comienza el símbolo con una "M".
- Las funciones que no son miembro de ninguna clase siguen la misma notación que las variables globales, es decir, con una "g" minúscula como prefijo y separando las palabras con mayúsculas. Ejemplo: void gUnaFuncionGlobal (int aParam1, int aParam2);.
- Se deben de utilizar sentencias Select-Case o Switch en lugar de utilizar sentencias if-then-else anidadas

Variables

- En la declaración de variables se debe de seleccionar el tipo de dato más pequeño que pueda contener el valor de dicha variable, por ejemplo para un entero pequeño escoger un tipo short en lugar de un int.
- Se debe de mantener el tiempo de vida de las variables tan corto como sea posible con el propósito de minimizar la utilización de los recursos, por ejemplo variables que representan conexiones a bases de datos o a recursos de red.
- Se debe de mantener la visibilidad de las variables tan corta como sea posible, con el propósito de evitar confusiones, facilitar el mantenimiento de código y minimizar la dependencia sobre dicha variable.
- Se deben de evitar el uso de variables públicas dentro de una clase, de esta manera es mejor utilizar métodos “getter” y “setter” y se evita acceder directamente a la variable desde otras clases.
- Es recomendado que las variable booleanas contengan una palabra que describa su estado: puedeEliminarse, esGrande, tieneEjercicios, etc. Y siempre se debe referir al estado verdadero: tieneTiempo en lugar de noTieneTiempo.
- En el caso de las variables de poco uso, aunque aparezcan en unas pocas líneas de código, se debe de utilizar un nombre descriptivo. Utilice nombres de variables de una sola letra, como i o j sólo para índices (ciclos for).
- No se deben de utilizar números o cadenas literales, como por ejemplo For i = 1 To 7; sino se deben de emplear constantes con nombre, del tipo For i = 1 To Enumeracion.length para que resulten fáciles de mantener y comprender.
- Se debe de evitar el uso de “casting” de variables a menos que sea en una instrucción simple.
- Las variables miembros de clase se escriben con minúsculas, las variables con nombres compuestos, comienzan con una letra "m" minúscula y separa las palabras iniciando con mayúscula.
- Los nombres de variables locales comienzan con el prefijo "l" y continúan siguiendo el esquema de las variables miembro de clases (palabras en minúscula separadas por mayúscula). Es importante saber a partir del nombre el alcance de la variable.
- Las variables estáticas de clase se escriben comenzando con "sm", el resto de la palabra se escribe como una variable miembro ordinaria (es decir, minúsculas separadas por mayúsculas).
- Los nombres comienzan con el prefijo "g" y continúan siguiendo el esquema de las variables miembro de clases (palabras en minúscula separadas por mayúscula).
- Para variables estáticas los nombres comienzan con el prefijo "s" y continúan

siguiendo el esquema de las variables miembro de clases (palabras en minúscula separadas por mayúscula).

- Se deben de liberar las referencias a objetos de las variables por medio de la asignación `variable=null` o `variable=nothing`

Bases de datos

Estos estándares especifican las normas a tomar en cuenta en el momento de realizar una conexión, nomenclatura de tablas, y procedimientos almacenados.

- Cuando se ocupen conexiones a base de datos se debe de minimizar al máximo la duración de las transacciones
- Cuando se defina el nombre de las tablas se debe de hacer en singular. Por ejemplo, se debe de utilizar `Empleado` en lugar de `Empleados`.
- Cuando se defina el nombre a las columnas de las tablas, no se debe de repetir el nombre de la tabla; por ejemplo, un campo llamado `EstudianteApellido` de una tabla llamada `Estudiante`.
- No se debe de agregar el tipo de datos en el nombre de una columna.
- No ponga prefijos `sp` a los procedimientos almacenados, ya que se trata de un prefijo reservado para la identificación de procedimientos almacenados de sistemas.

Nomenclatura

Dentro de este apartado se especifican normas adicionales sobre la finalidad del nombre de los identificadores que se utilicen en el código fuente.

- Cuando se nombre un procedimiento y éste retorne un valor, se debe de incluir además del proceso, lo que devuelve por ejemplo: `EscribirArchivo`, `CalcularImpuestos`.
- Los archivos y los nombres de carpetas, al igual que los nombres de procedimientos, deben describir claramente su finalidad.
- Se debe de evitar utilizar el uso de caracteres especiales tales como `“!”` ó `“&”`
- Se debe de ser específico cuando se declaren objetos que puedan generar colisión, por ejemplo si se tienen dos métodos con el mismo nombre en diferentes *namespaces* se deben de escribir con el nombre completo incluyendo el del paquete.

2.3. Estándar de documentación Interna

La documentación interna está formada por los comentarios que los programadores escriben dentro del código fuente durante la fase de desarrollo.

Uno de los problemas de la documentación de software interna es garantizar que se mantengan y actualicen los comentarios al mismo tiempo que el código fuente.

Aunque unos buenos comentarios en el código fuente no tienen ningún valor en el tiempo de ejecución, resultan valiosísimos para un programador que tenga que mantener una parte de software particularmente compleja.

La documentación interna debe de cumplir con las siguientes reglas:

- Los comentarios deben ser en español.
- Al principio de cada rutina, resulta útil hacer comentarios estándar, repetitivos, que indiquen el propósito de la rutina, las suposiciones y las limitaciones. Un comentario repetitivo podría consistir en una breve introducción que explicara por qué existe y qué puede hacer.
- Se deben evitar añadir comentarios al final de una línea de código, porque lo hacen

más difícil de leer. Sin embargo, los comentarios de final de línea sí son apropiados al anotar declaraciones de variables.

- Se deben evitar los comentarios recargados, como las líneas enteras de asteriscos, para lo cual es mejor utilizar líneas en blanco para separar comentarios y código.
- Al momento de realizar la implementación se debe de verificar que todos los comentarios temporales hayan sido eliminados, con el propósito de evitar confusiones en los futuros mantenimientos.
- Se deben de utilizar frases completas al momento de elaborar un comentario, ya que el propósito de los comentarios es aclarar el código, no confundir más al lector.
- Se deben de añadir los comentarios al mismo tiempo que se programa, ya que al posponer esta tarea se necesita un mayor tiempo para comprender lo que se hizo.
- Se deben de evitar los comentarios inapropiados, por ejemplo “Método que me toco hacer a las 3 am”
- La ortografía es un factor importante al momento de realizar un comentario, nunca debe de obviarse este factor.

Formato del código

El formato del código es un factor clave que ayuda a depurar y mantener rápidamente una aplicación, ya que evita que se realice un esfuerzo extraordinario al momento de mantenerlo.

Se deben de seguir las siguientes reglas con respecto al formato del código:

- Se debe de evitar albergar múltiples clases en un solo archivo.
- Se debe de ocupar un tamaño estándar de sangría a lo largo de todo el código
- Se debe de utilizar único tipo de letra cuando publique versiones impresas del código fuente.
- Se deben de agregar los namespaces en orden descendente empezando por los del sistema y terminado por los personalizados o de usuario.
- Una clase debe estar definida en orden descendente de la siguiente manera: Variables Miembro, Constructores, Enumeraciones, Estructuras o Clases anidadas, Propiedades y por ultimo los Métodos.
- La secuencia de declaración de acuerdo a los modificadores de acceso debe ser la siguiente:

public
protected
friend
private

- Se deben de alinear verticalmente las inicios y cierre de cualquier bloque de código:

Public void Main()

```
{  
...  
}
```

ó

Public Sub Main()

```
...  
End Sub
```

- Se debe de establecer una longitud de línea máxima para el código.
- Se debe de utilizar espacios antes y después de los operadores siempre que eso no

altere la sangría aplicada al código:

miVariable = 3; en vez de miVariable=3;

- Cuando tenga que dividir una línea en varias, aclare que el código sigue en la línea de más abajo mediante un operador de concatenación colocado al final de cada línea, y no al principio.
- Siempre que sea posible, no se debe de colocar más de una instrucción por línea, a excepción de los bucles.
- Cuando se escriban instrucciones SQL, se deben de utilizar mayúsculas para las palabras clave: SELECT, UPDATE, WHERE, FROM, etc.
- Cuando se utilicen las cláusulas SQL, estas deben de estar en líneas separadas, de modo que las instrucciones sean más fáciles de leer y editar:
SELECT Nombre, Apellido
FROM Estudiantes
WHERE Fecha= 'No8.Today';
- No se debe de comparar strings con "" o con string.empty, use la propiedad length del string: If(String.length == 0)
- Se debe de evitar utilizar recursividad, en su lugar se debe de utilizar ciclos for anidados.
- Se debe de utilizar el operador condicional ternario solo en casos triviales, cuando sean casos complejos se debe de evitar.

Documentación de clase (lenguaje orientado a objetos)

Se debe definir un bloque de comentario inmediatamente antes de la declaración de la clase. Deben utilizarse al máximo los tags definidos por la herramienta de documentación utilizada para potenciar la comprensión del código.

Documentación de funciones y métodos

Previo a la declaración de cada función o método de clase debe incluirse un bloque de comentario. Dicho bloque debe describir brevemente la funcionalidad del método, si éste es concreto, o el contrato general, si éste es virtual puro. Es imprescindible incluir comentarios sobre cada parámetro y valor de retorno de la función / método en cuestión.

Comentarios de directorios ("Leame.txt, Readme.txt")

En cada directorio del árbol de directorios que forme el proyecto debe incluirse un archivo que detalle el contenido de dicho directorio, el nombre del proyecto.

- El propósito de los archivos que se encuentran bajo este directorio
- Un comentario de una línea por cada archivo en el directorio (o directorio). Puede automatizarse esta parte a partir de la información en los encabezados.
- Dar referencias a la gente sobre documentación relacionada:
- Directorios de fuente
- Documentación en línea
- Documentación impresa
- Documentación de diseño

Adicionales

Normas de propósito general a tomar en cuenta en el desarrollo de la aplicación:

- Nunca se deberán de omitir los modificadores de acceso, se deben de declarar siempre explícitamente, ya que permiten acelerar la comprensión y el mantenimiento del código fuente.
- Para el control de Código Fuente se debe utilizar Visual Source Safe, con lo que se provee un medio de control del código cuando dos o más desarrolladores codifican al mismo tiempo una aplicación.

- Se deben de utilizar siempre el manejo de excepciones con el propósito de que al suceder un comportamiento anormal en la ejecución de la aplicación, se le de una respuesta apropiada al usuario manteniendo la aplicación en ejecución.

2.4. Estándares de Interfaz Gráfica

Para definir estándares para interfaz gráfica de usuario (GUI) de SPEI-3 es necesario recopilar información acerca de las normas estándares GUI. Si bien no hay entes reguladores como la W3C⁴⁴, existen ciertas reglas agrupadas que se generan del sentido común, uso y experiencias.

Desde cuestiones bastantes obvias hasta detalles apenas perceptible; estos estándares, en su medida, se esfuerzan por hacer el sistema más usable y accesible para todos sus usuarios.

Estos estándares GUI, están divididos en 2 grupos:

- **Interacción:** Se basa en la interacción con el usuario, intenta generar una comunicación armónica con los usuarios del software.
- **Presentación:** Intenta optimizar la forma en la que se le presentan los datos al usuario y, a la vez, crear un entorno de trabajo donde el usuario sepa siempre donde se encuentra en el sistema.

El fin de estandarizar la interfaz GUI, es hacer que las interfaces de los sistemas estén pensadas en el usuario, que en definitiva es él que interactúa con el sistema.

2.5. Estándares de Interacción

Estos estándares se remiten a la comunicación entre el usuario y la computadora, estos se comunican a través del uso de varios controles. Seleccionar dichos controles para la interacción, seleccionar el control correcto requiere que el diseñador investigue estándares industriales, corporativos y que incluyan las necesidades del usuario.

Esta sección se basa en la información de las siguientes fuentes:

- “The Windows Interface Guidelines for Software Design” publicado por Microsoft Press.
- “User Interface”⁴⁵

Estas fuentes se han considerado que la GUI más conocida es la del sistema operativo Windows⁴⁶ propiedad de Microsoft® y para el cual esta orientado SPEI-3.

⁴⁴ <http://www.w3c.es/Consortio/>, El Consorcio World Wide Web (W3C) es un consorcio internacional cuya misión es desarrollar protocolos y pautas que aseguren el crecimiento futuro de la Web.

⁴⁵ <http://msdn2.microsoft.com/en-us/library/bb892735.aspx>, Visitar subsecciones para mayor referencia.

⁴⁶ <http://www.xitimonitor.com/en-us/technical/operating-systems-february-2007/index-1-2-3-73.html?xtor=11>

Controles

Los controles se refieren a la amplia gama de objetos que permiten al usuario interactuar con el software en operaciones de inserción, selección y eliminación,

Use botones de comando cuando el usuario deba realizar acciones frecuentes o críticas, estos botones deben reflejar el tipo de acción que deberían realizar.

Las etiquetas de los controles deben ser claras y concisas, en el caso de botones de comando, por ejemplo "Guardar como". Use palabras múltiples solamente cuando sea necesario aclarar el objetivo del control, utilice las reglas de capitalización para palabras significativas.

Se debe procurar que, de un tipo específico de control (cheques, botones de radio, etc.), existan menos de seis controles en un cuadro de diálogo. Si las opciones sobrepasan este número se debe considerar cambiar el tipo de control por uno que pueda agrupar dichas opciones por ejemplo menú lista.

Utilice los estándares industriales para las etiquetas de los botones de comando

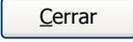
Etiqueta	Acción	Tecla de Acceso
	Realiza los cambios y cierra la ventana activa	
	No realiza los cambios y cierra la ventana activa	
	Cierra la ventana cuando los cambios no se realizan o se cancelan	
	Restaura a los valores predeterminados, deja la ventana activa abierta	
	Realiza los cambios y deja abierta la ventana activa	
	Abre la ayuda asociada	

Figura 5-6

El tamaño de los controles debe ser uniforme dependiendo del tipo (campo de texto, menú, botones de comando). Si se tratase de un campo de texto, por ejemplo, para el cual se conoce la longitud del valor que contendrá entonces si se debe ajustar el tamaño del control a la longitud del dato, no así cuando este valor varía.

Los botones de comando deben estar separados del resto de controles en los cuadros de diálogo.

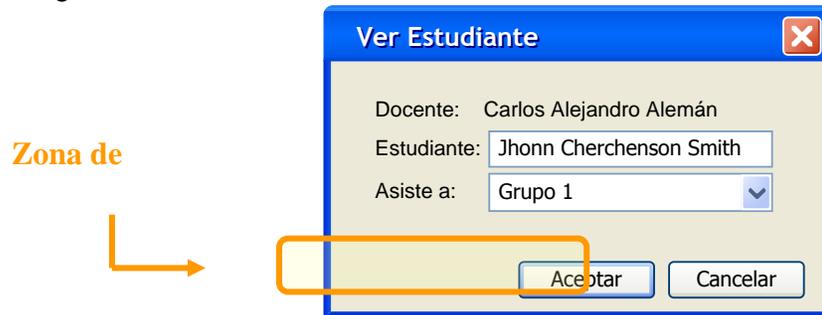
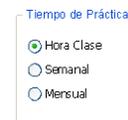


Figura 5-7

Debe existir un orden en el alineamiento de los controles, todos deben ser consistentes entre los distintos cuadros de diálogos del

Para opciones de Si/No utilice cuadro de cheque y no botones de

Las etiquetas de los controles deben alinearse todas a la izquierda cuando estas estén fuera del control.



ellos software.

radio.

Estándar de interacción usuario no vidente.

Estos estándares se remiten a la comunicación entre el usuario no vidente y la computadora, estos se comunicarán a través del uso del teclado estándar utilizando el siguiente modelo de teclas utilizables por ser fácilmente identificables para una persona no vidente.⁴⁷

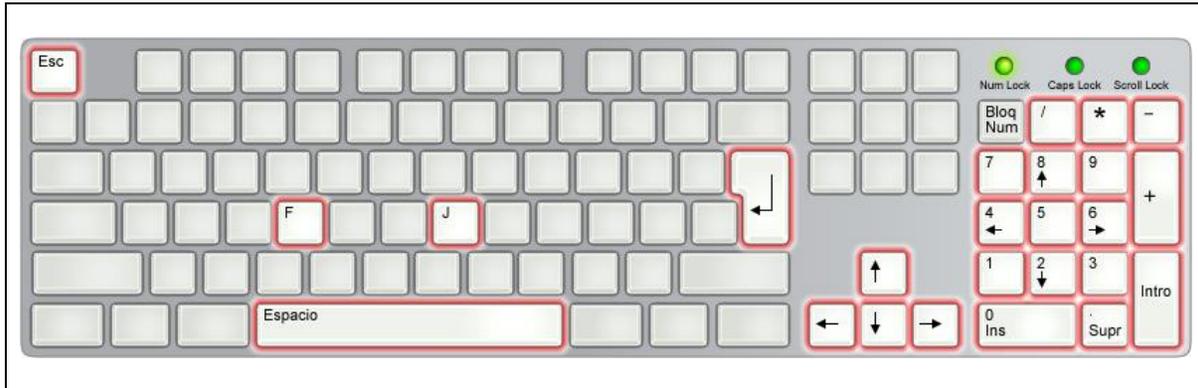


Figura 5-8 Modelo de teclado a utilizar en el SPEI3

Estas teclas se seleccionaron siguiendo los siguientes criterios:

- Las teclas “F”, “J” y “BloqNum 5” presentan relieve cuyo objetivo es situar al usuario en la posición de manos correcta, este relieve permite a los usuarios no videntes a encontrar fácilmente dichas teclas en el teclado.
- Las teclas “Esc” y direccionales, así como el resto de las teclas del texto numérico se han seleccionado por su posición estratégica en el teclado, estas teclas están situadas en los extremos y se encuentran aisladas del resto, individual o en grupo (en el caso del teclado numérico la tecla “BloqNum 5” sirve como referencia para las demás teclas)

Las teclas “BloqNum +”, “BloqNum -”, “BloqNum Intro”, “Espacio”, “BloqNum 0”, “Intro” se han seleccionado por el tamaño que poseen que las diferencia del resto.

Presentación

La presentación hace referencia a la forma en que los datos son presentados en pantalla, ventanas y en forma audible. La presentación debe considerar que es lo que los usuarios necesitan hacer con los datos. El despliegue apropiado de la información puede hacer la diferencia en que tan útil perciben los usuarios que es la interfase.

⁴⁷ Este modelo de teclado está basado en el teclado estándar en español de 109 teclas [http://es.wikipedia.org/wiki/Teclado_\(inform%C3%A1tica\)#Tipos_de_teclado](http://es.wikipedia.org/wiki/Teclado_(inform%C3%A1tica)#Tipos_de_teclado)

Distribución de Pantalla

Los principios de la distribución de pantalla no son estándares estáticos, estos deben actualizarse con el tiempo, pero estos cada vez son más importantes ya que mas y mas elementos visuales se van agregando a la pantalla.

Tamaño estándar de pantalla

Todas las pantallas deben ser desarrolladas teniendo en cuenta la capacidad del hardware. Actualmente esta debería ocupar un mínimo de 800x600 píxeles⁴⁸ y un máximo de 1024x760 píxeles, este último es el estándar actual pero el mínimo permitido aun es utilizado por muchos monitores.

Convención para nombres de formularios

Cada formulario debe tener un nombre único que debe mostrarse en la parte superior del formulario y debe ser descriptivo en su naturaleza. Otros títulos adicionales o nombres dentro del formulario son redundantes y no deben aparecer.

Use la cantidad apropiada de información

Cada ventana o cuadro de diálogo debe representar una tarea o subtareas, si la tarea es complicada se utilizarán mas de una ventana, una por cada tarea.

Ventana principal

La Ventana Principal es la pantalla o ventana de usuario situada al fondo de todas las tareas y es la base de donde dichas tareas son lanzadas. La ventana principal puede ser una ventana de datos, una lista, un formulario o simplemente no tener ningún dato.

La ventana principal no es, necesariamente, la primera ventana que el usuario debe ver cuando inicia la aplicación, pero debe llegar a ella de una u otra manera.

Esta no debe ser una ventana en blanco como en la figura 2.7, se utilizará una ventana con información importante para las tareas que se realizarán, por ejemplo un resumen de datos o un formulario listo para ser llenado (figura 2.8).

Tener una ventana principal útil ayuda a los usuarios a recordar que es lo que van a hacer, les da un punto de inicio concreto y les ayuda a ubicarse dentro de la interfaz.

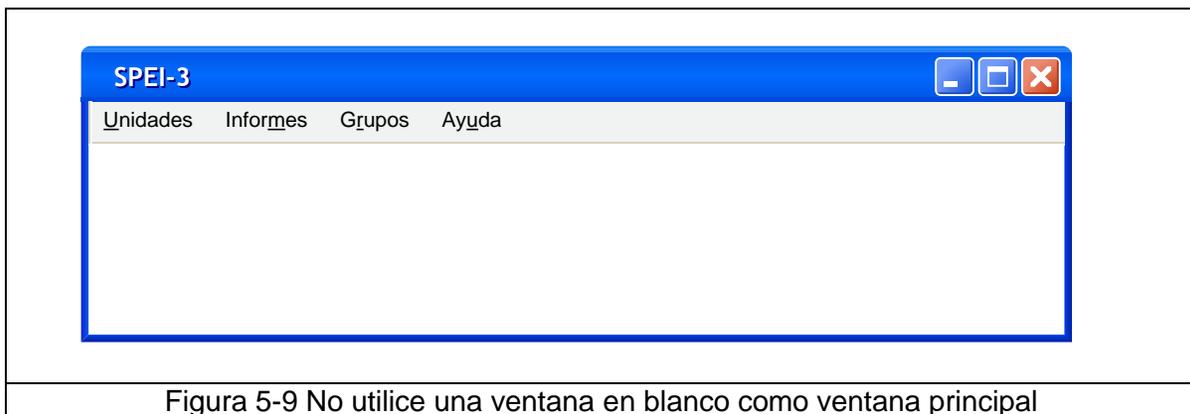


Figura 5-9 No utilice una ventana en blanco como ventana principal

⁴⁸ <http://es.wikipedia.org/wiki/P%C3%ADxel>. Un píxel (acrónimo del inglés *picture element*) es la menor unidad homogénea en color que forma parte de una imagen digital.

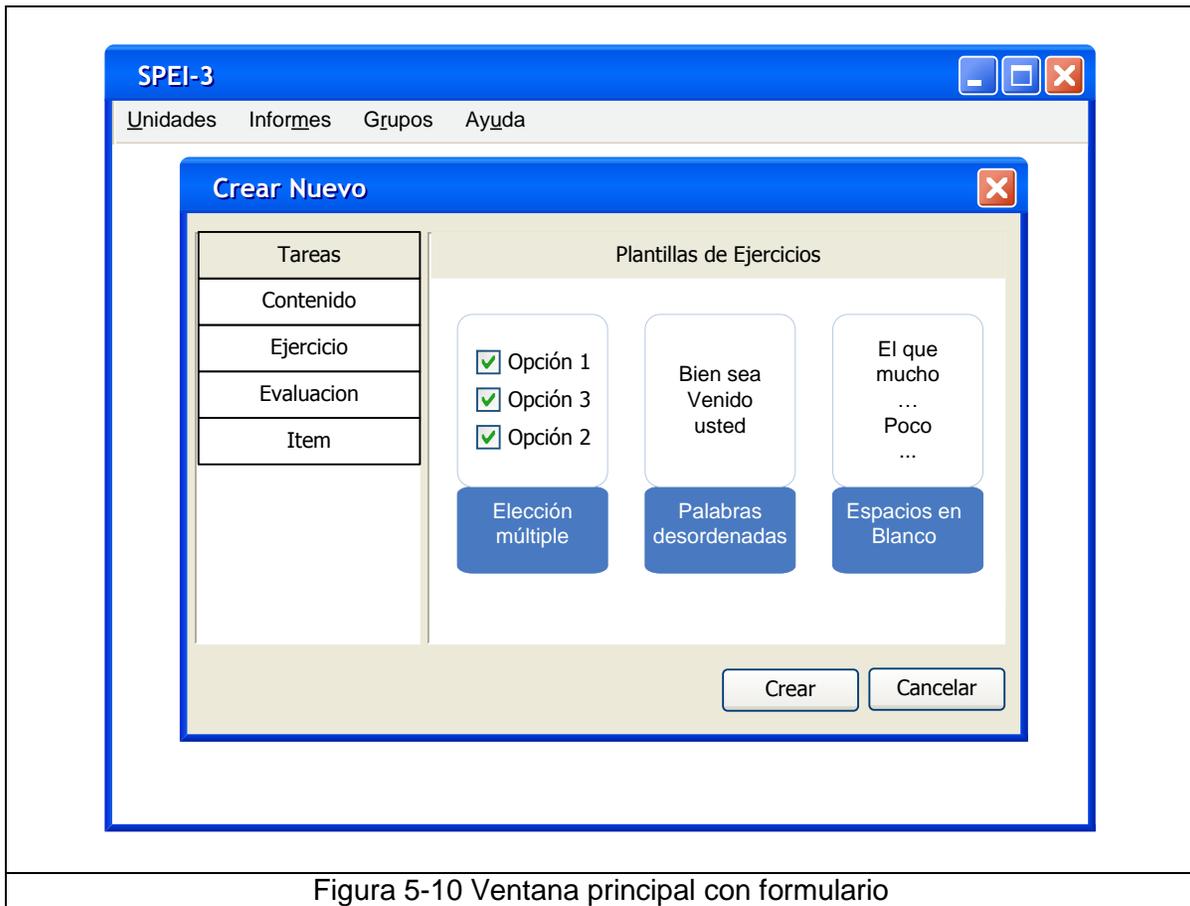


Figura 5-10 Ventana principal con formulario

Fuentes

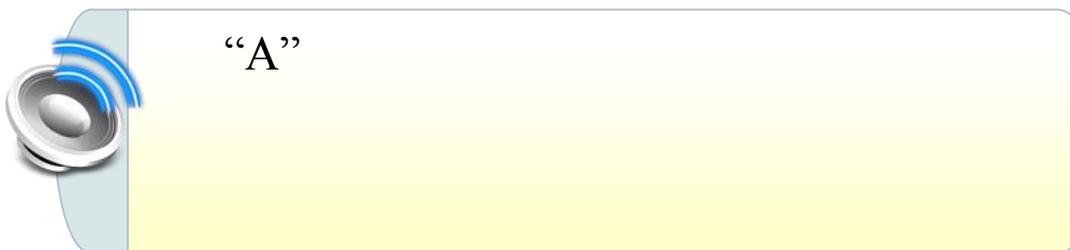
En lo que respecta a las interfaces gráficas, se tienen mas opciones sobre que tipo de letras se mostrarán.

Se utilizará la fuente Sans Serif para el texto y las viñetas ya que es fácil de leer en pantalla.

Deletreo

Cada vez que un usuario no vidente presione una tecla dentro de SPEI3 cuando le sea solicitado, SPEI3 inmediatamente debe reproducir el nombre de dicha tecla:

Si un usuario no vidente presionase la tecla 



3. Diccionario de datos (DD):

En este apartado se desarrolla el diccionario de datos para el SPEI3, para ello se describe la metodología utilizada para su elaboración.

3.1. Definición

Se ha propuesto el diccionario de datos como una gramática casi formal para describir el contenido de los elementos de información y ha sido definido de la siguiente forma:

El diccionario de datos contiene las definiciones de todos los datos, en una especificación del proceso y en el propio diccionario de datos. Los datos compuestos (datos que pueden ser además divididos) se definen en términos de sus componentes; los datos elementales (datos que no pueden ser divididos) se definen en términos del significado de cada uno de los valores que puede asumir. Por tanto, el diccionario de datos está compuesto de definiciones de flujo de datos, archivos [datos almacenados] y datos usados en los procesos [transformaciones]

3.2. Objetivo

Permite especificar el significado y la composición de los datos aplicable a flujos de datos y almacenes

3.3. Metodología

Para realizar el diccionario de datos es necesario antes conocer la simbología que se utilizará para su elaboración y su significado.

Símbolos BNF	Significado
@	Clave de acceso
-	Rango (no estándar)
“ “	Contenido textual (no estándar)
* *	Comentario
[]	Selección de alternativas
{ }	Iteración
()	Opcionalidad
+	y
=	Definición, Compuesto de

TABLA 5-1

Definiciones =

Una definición de un dato se introduce mediante el símbolo “=”; en este contexto el “=” se lee como “está definido por”, o “está compuesto de”, o “significa”. Para definir un dato completamente, la definición debe incluir:

Datos elementales

Son aquellos para los cuales no hay una descomposición significativa. Por ejemplo, puede ser que no se requiera descomponer el nombre de una persona en primer-nombre, apellido-materno y apellido-paterno; esto depende del contexto del sistema que se esté

modelando. Cuando se han identificado los datos elementales, deben ser introducidos en el DD y proveer una breve descripción que describa el significado del dato. En el caso de que el dato tenga un nombre significativo, se puede omitir la descripción, sin embargo; es importante especificar las unidades de medida que el dato puede tomar.

Ejemplo:

*Peso = * peso del paciente al ingresar al hospital **
** unidad: kilo, rango:2-150 **

Datos opcionales ()

Un dato opcional es aquel que puede o no estar presente como componente de un dato compuesto.

Ejemplo:

Dirección = calle + número + (ciudad) + (país) + (código-postal)

Selección [[]]

Indica que un elemento consiste de exactamente una opción de un conjunto de alternativas. Ejemplos:

Sexo = [Femenino | Masculino]

Iteración {}

Se usa para indicar ocurrencias repetidas de un componente en un elemento compuesto. Por ejemplo *Orden-de compra = nombre-cliente + dirección-de-envío + {artículo}* significa que una orden de compra siempre debe contener un nombre de cliente, una dirección de envío y cero o más ocurrencias de un artículo. Se pueden especificar límites superiores e inferiores a las iteraciones. *Orden-de compra = nombre-cliente + dirección-de-envío + 1{artículo}10* significa que una orden de compra siempre debe contener un nombre de cliente, una dirección de envío y de 1 a 10 artículos.

¿Qué debe tener una definición del diccionario de datos?

Tipo de elemento:	Debe tener:	
Primitivo y agregado	Significado	Descripción de elemento
Agregado	Composición	Los distintos elementos que lo conforman
Primitivo	Valores	Los valores que puede tomar el elemento primitivo

TABLA 5-2

3.3. Diccionario de Datos para SPEI3

Nombre	Descripción	Estructura
Ciclo	Periodo ordinario en el que se desarrollan las actividades educativas.	Ciclo = codigo_ciclo + nombre_ciclo
codigo_ciclo	Clave única que identifica cada ciclo	codigo_ciclo = 3{caracter}3 rango: ["A"-“Z” “a”-“z”]
nombre_ciclo	Nombre asignado al ciclo	nombre_ciclo = {caracter}20 rango: ["A"-“Z” “a”-“z”]
Unidad	En la Unidad se organizan todas las lecciones dentro de SPEI3	Unidad = código_unidad + num_unidad + nombre_unidad
codigo_unidad	Clave única que identifica la unidad	codigo_unidad = 1{caracter}4 rango: ["A"-“Z” “a”-“z”]
num_unidad	Numero de la unidad	num_unidad = entero2 rango: 1 – 99
nombre_unidad	Nombre de la unidad	nombre_unidad = {caracter}70 rango: ["A"-“Z” “a”-“z”]
Leccion	En las lecciones se organizan todos los temas dentro de SPEI3	Leccion = código_leccion + num_leccion + nombre_leccion
codigo_leccion	Clave única para cada lección	codigo_leccion= 1{caracter}4 rango: ["A"-“Z” “a”-“z”]
num_leccion	número de la lección	num_leccion = entero rango: 1 – 99
nombre_leccion	Nombre de la leccion	nombre_leccion = {caracter}70 rango: ["A"-“Z” “a”-“z”]
Tema	Tema posee el contenido que el docente desea impartir en sus clases	Tema = codigo_tema + num_tema + nombre_tema + descripción_tema
codigo_tema	Clave única que identifica al tema	codigo_tema = 1{caracter}8 rango: ["A"-“Z” “a”-“z”]
num_tema	Número del tema	num_tema = entero rango: 1 – 99
nombre_tema	Nombre del tema	nombre_tema = {caracter}70 rango: ["A"-“Z” “a”-“z”]
descripción_tema	Atributo que describe el tema	descripción_tema = {caracter}50 rango: ["A"-“Z” “a”-“z”]
Contenido	Contiene el desarrollo del contenido creado por un docente para un tema	Contenido = codigo_contenido + codigo_tipo_contenido +

	especifico	nombre_contenido + desarrollo_contenido + num_elemento_de_tema
codigo_contenido	Clave única que identifica cada contenido	codigo_contenido = 1{caracter}8 rango: ["A"-“Z” “a”-“z”]
codigo_tipo_contenido	Clave única que identifica cada tipo de contenido	codigo_tipo_contenido = 1{caracter}1 rango: ["A"-“Z” “a”-“z”]
nombre_contenido	Nombre del contenido	nombre_contenido = 1{caracter}70 rango: ["A"-“Z” “a”-“z”]
desarrollo_contenido	Este campo esta compuesto por todo el contenido agregado por el docente, dicho contenido puede estar en ingles y/o español	desarrollo_contenido = 1{caracter}255 rango: ["A"-“Z” “a”-“z”]
num_elemento_de_tema	Numero de elemento del tema al que pertenece	num_elemento_de_tema = entero rango: 1 – 99
Evaluacion	Contiene el nombre de la evaluación además del código.	Evaluacion = código_evaluacion + nombre_evaluacion + 1{Item}
codigo_evaluacion	Clave única que identifica cada evaluación	codigo_evaluacion = 1{caracter}8 rango: ["A"-“Z” “a”-“z”]
nombre_evaluacion	Contiene el detalle de la evaluación	nombre_evaluacion1 = 1{caracter}70 rango: ["A"-“Z” “a”-“z”]
num_elemento_de_tema	Numero de elemento del tema al que pertenece la evaluación	num_elemento_de_tema = entero rango: 1 – 99
Ejercicio	Evaluación por contenido de un tema específico	Ejercicio = código_ejercicio + nombre_ejercicio + num_elemento_de_tema
codigo_ejercicio	Clave única que identifica cada ejercicio	codigo_ejercicio = 1{caracter}8 rango: ["A"-“Z” “a”-“z” 1 – 9]
nombre_ejercicio	Contiene el detalle del ejercicio	nombre_evaluacion1 = 1{caracter}70 rango: ["A"-“Z” “a”-“z”]
num_elemento_de_tema	Numero de elemento del tema al que pertenece el ejercicio	num_elemento_de_tema = entero rango: 1 – 99
Item	Dificultades de las cuales estarán compuestos los ejercicios y las	Item = codigo_item + texto_item + solución_item + 1{Alternativas} +

	evaluaciones que forman parte de la estructura del tema.	1{tipo_items} + (ponderacion)
codigo_item	Clave única que identifica cada ítem	codigo_item = 1{caracter}10 rango : ["A"-“Z” “a”-“z”]
texto_item	Contiene el enunciado del ítem	texto_item = 1{caracter}300 rango : ["A"-“Z” “a”-“z”]
solucion_item	Contiene la alternativa que es la solución del ítem	solucion_item = entero rango: 1 – 99
Alternativa	Contienen las alternativas de solución para un ítem	Alternativa = código_item + numero_alternativa + texto_alternativa
codigo_item	Clave única que identifica cada alternativa	codigo_item = 1{caracter}10 rango: ["A"-“Z” “a”-“z”]
numero_alternativa	Numero de la alternativa	numero_alternativa = entero rango: 1 – 99
texto_alternativa	El contenido de la alternativa	texto_alternativa = 1{caracter}300 rango: ["A"-“Z” “a”-“z”]
Tipo_item	Contiene el tipo de ítem contenido en una evaluación o ejercicio.	Tipo_item
codigo_tipo_item	Clave única que identifica cada tipo de ítem	codigo_tipo_item = 1{caracter}1 rango: ["A"-“Z” “a”-“z”]
descripción_tipo_item	Detalle del tipo de ítem, contiene el nombre del tipo de ítem	descripción_tipo_item = 1{caracter}20 rango: ["A"-“Z” “a”-“z”]
ponderación	Ponderación del ítem, indica el valor del ítem	Ponderación = flotante Rango: 0.00 -10.00
alumno_nota	Contiene la nota por evaluación realizada por un alumno	alumno_nota = nota + nota_cualitativa
nota	Contiene la nota total de la evaluación realizada por el alumno	Nota = flotante Rango: 0.00 – 10.00
nota_cualitativa	Contiene la nota cualitativa de la evaluación realizada por el alumno	nota_cualitativa = 1{caracter}20 rango : ["A"-“Z” “a”-“z”]
Sesion	Contiene la información de los clientes de SPEI3 alumnos y docentes	Sesion = id_sesion + login_usuario + inicio_sesion + fin_sesion + tiempo_sesion
Id_sesion	Clave única que identifica cada sesión	Id_sesion = 1{caracter}20 rango : ["A"-“Z” “a”-“z” 1 – 9]
login_usuario	Nombre de usuario al que	login_usuario =

	pertenece la sesión	1{caracter}15 rango : ["A"-“Z” “a”-“z” 0 – 9]
inicio_sesion	Fecha de inicio de sesión	inicio_sesion = fecha
fin_sesion	Fecha de fin de sesión	fin_sesion = fecha
tiempo_sesion	Tiempo que dura en la sesión	tiempo_sesion = entero rango: 1 – 9999
Tema_visto	Contiene la información de los tiempos de practica de los temas	Tema_visto = tiempo_practica + necesita_reforzar
tiempo_practica	Tiempo de practica del alumno	tiempo_practica = entero rango: 0 – 9999
necesita_reforzar	Indica si el alumno necesita reforzar	necesita_reforzar = booleano rango: [0 1]
Grupo	Contiene la información del grupo al que pertenecen los alumnos y que administra el docente	Grupo = código_grupo + nombre_grupo
codigo_grupo	Clave única que identifica cada grupo	codigo_grupo = 1{caracter}2 rango : ["A"-“Z” “a”-“z” 1 – 9]
nombre_grupo	Nombres asignado al grupo	nombre_grupo = 1{caracter}20 rango : ["A"-“Z” “a”-“z”]
Docente	Contiene la información del docente en SPEI3	Docente = nombre_docente + apellido_docente + login_docente + clave_docente + es_administrador
codigo_docente	Clave única que identifica cada docente	codigo_docente = 1{caracter}4 rango : ["A"-“Z” “a”-“z”]
nombre_docente	Nombres propios del docente	nombre_docente = {caracter}50 rango : ["A"-“Z” “a”-“z”]
apellido_docente	Apellidos propios del docente	apellido_docente = {caracter}50 rango : ["A"-“Z” “a”-“z”]
login_docente	Nombre de usuario del docente	login_docente = 1{caracter}15 rango : ["A"-“Z” “a”-“z” 0 – 9]
clave_docente	Contraseña del docente	clave_docente = 1{caracter}15 rango : ["A"-“Z” “a”-“z” 0 – 9]
es_administrador	Indica si el usuario docente es administrador	es_administrador = booleano [0 1]

Alumno	Contiene la información del alumno en SPEI3	Alumno = codigo_alumno + nombre_alumno + apellido_alumno + carnet_alumno + login_alumno + clave_alumno
codigo_alumno	Clave única que identifica cada alumno	codigo_alumno = 1{caracter}8 rango : ["A"-“Z” “a”-“z”]
nombre_alumno	Nombres propios del alumno	nombre_alumno = {caracter}50 rango : ["A"-“Z” “a”-“z”]
apellido_alumno	Apellidos propios del alumno	apellido_alumno = {caracter}50 rango : ["A"-“Z” “a”-“z”]
carnet_alumno	Identificación universitaria única por alumno	carnet_alumno = 7{caracter}7 rango : ["A"-“Z” “a”-“z” 1 – 9]
Login_alumno	Nombre de usuario del alumno	Login_alumno = 1{caracter}15 rango : ["A"-“Z” “a”-“z” 0 – 9]
Clave_alumno	Contraseña del alumno	Clave_alumno = 1{caracter}15 rango : ["A"-“Z” “a”-“z” 0 – 9]

4. Diagrama Entidad Relación (DER)

Como parte del diseño se debe de conocer las entidades que intervienen en el software, junto a las relaciones que las unen para un mejor diseño de la base de datos del SPEI3, en este apartado se da una definición de DER una metodología para elaborar un DER y se presenta el DER para el SPEI3.

4.1. Definición

El diagrama entidad-relación (DER) es un modelo de red que describe la distribución de datos almacenados en un sistema.

La diferencia principal con el DFD, que está orientado a las funciones del sistema, es que el DER esta orientado a los datos.

El principal elemento del DER es la entidad, que se representa con un rectángulo.

Una entidad puede ser persona, lugar, cosa o evento cuya información necesaria para el sistema.

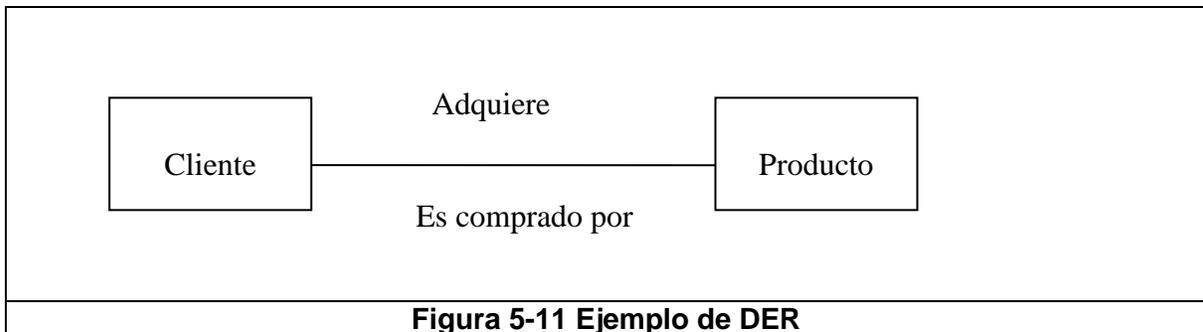
Una relación es la interacción entre las entidades y se representa con una línea que conecta las entidades asociadas.

4.2. Metodología para la elaboración del DER

Para crear el DER se comienza a partir del modelo conceptual creado en la etapa de análisis en este se identificaron las entidades y las relaciones existentes entre ellas, pero para el DER se necesita refinar todos los elementos incluidos en el modelo conceptual por ello se realizan los siguientes pasos:

1. Definir las entidades
2. Definir los atributos de las entidades
3. Definir los atributos llave de las entidades
4. Establecer las relaciones entre las entidades
5. Establecer la cardinalidad de las relaciones
6. Definir las relaciones recursivas
7. Definir los atributos en las relaciones

Siendo estos pasos un proceso repetitivo por lo que se puede acudir a un paso anterior en cualquier momento con el propósito de refinar el diagrama.



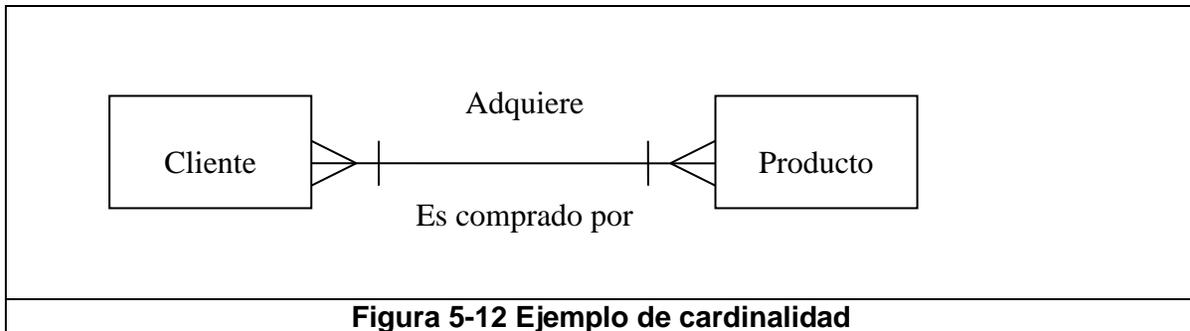
En este ejemplo, el diagrama se lee así:
 Cliente adquiere producto y producto es comprado por cliente.

Sin embargo, es necesario agregar a este diagrama la cardinalidad. La cardinalidad explica la forma exacta como se relacionan las entidades, la siguiente tabla muestra las posibilidades de la cardinalidad y su representación:

CARDINALIDAD	SE LEE	REPRESENTACIÓN
1:1	Uno a uno	
1:M	Uno a muchos	
1:0	Uno a ninguno	
M:1	Muchos a uno	
M:M	Muchos a muchos	
M:0	Muchos a ninguno	

TABLA 5-3

El ejemplo quedaría como sigue:



Y se lee:

Un cliente adquiere muchos productos.

Un producto es comprado por muchos clientes.

Con esta descripción se pudiera pensar que “el mismo producto” es comprado por muchos clientes, pero no se refiere a un elemento de producto sino al producto en sí.

4.3. Diagrama Entidad/ Relación

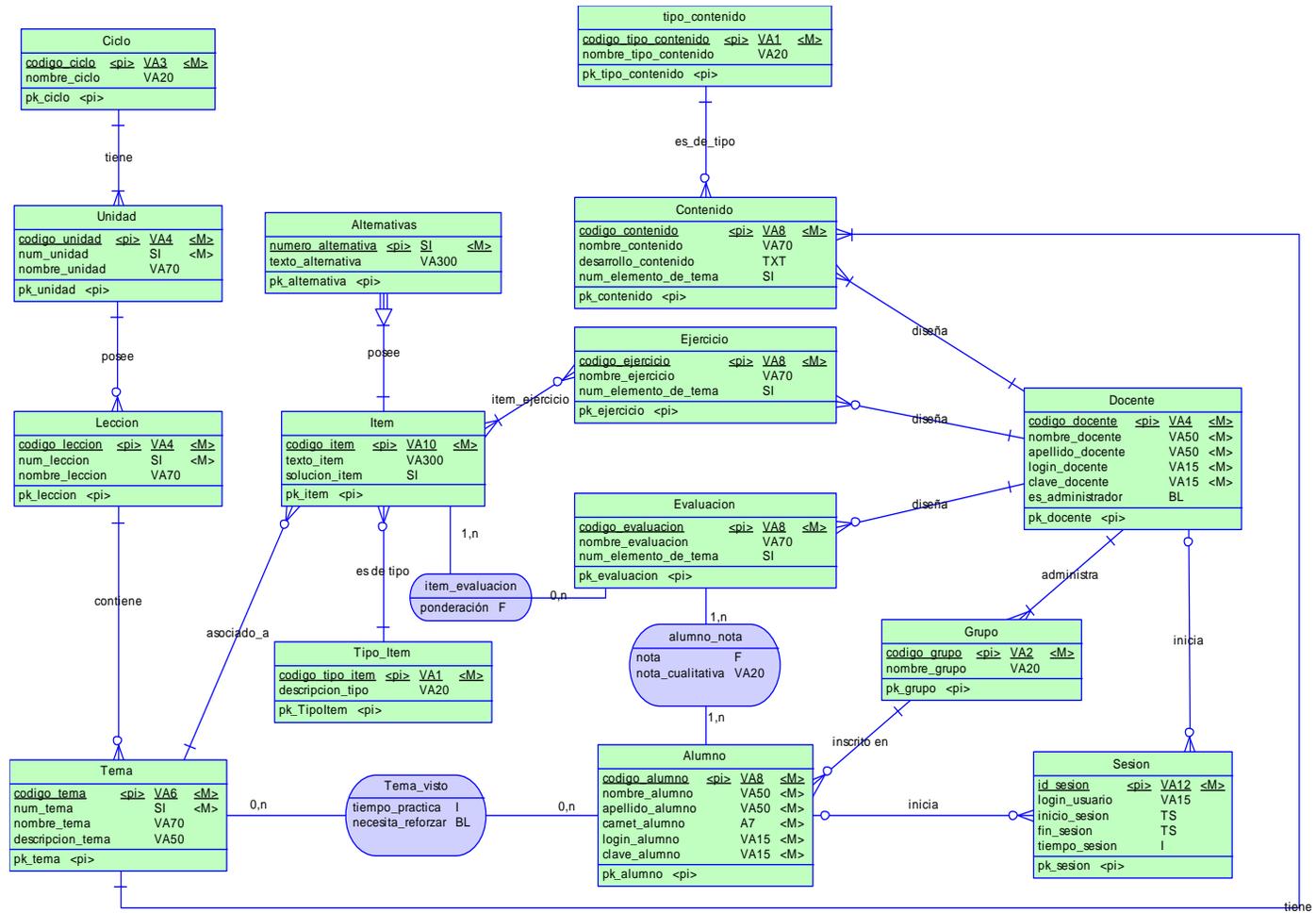


Figura 5-13 Diagrama Entidad-Relación SPEI3

5. Modelo físico de la base de datos

A partir del diagrama entidad relación se crea el modelo físico de la base de datos, en este apartado sobre el modelo físico de la base de datos se da una definición, una metodología y se presenta el del SPEI3.

5.1. Definición

El Modelo Físico o Esquema de Bases de Datos Relacional (BDR), es la representación gráfica de las diferentes tablas y relaciones existentes en una base de datos. Este esquema está basado en el Diagrama de Entidad Relación (DER), y al igual que en éste, es necesario representar las llaves primarias, las llaves foráneas, atributos y demás elementos que componen a cada entidad.

5.2. Metodología para la elaboración del modelo físico de la BD

Los pasos a seguir para la construcción del modelo físico son los siguientes:

1. Elaborar el Diagrama de Entidad Relación (DER), en donde se identifican entidades, atributos, atributos llaves y las relaciones existentes entre las diferentes entidades.
2. Cada entidad del DER pasará a ser una tabla en el modelo físico, y tendrá como llave primaria el atributo llave de la entidad.
3. Cada entidad débil pasará a ser una tabla, la cual tendrá una llave primaria compuesta, formada por el atributo llave de la entidad propietaria y el atributo llave parcial de la entidad débil.
4. Cuando en el DER exista una relación binaria de uno a uno (1:1), se debe incluir como atributo (llave foránea) en la tabla correspondiente a la entidad con participación total, el atributo llave de la entidad con participación parcial.
5. Para cada relación binaria de uno a muchos (1: n) existente en el DER, se debe incluir como atributo (llave foránea) en la tabla correspondiente a la entidad con cardinalidad de muchos (n), el atributo llave de la entidad con cardinalidad de uno (1).
6. Para cada relación binaria de muchos a muchos (n: m) existente en el DER, se debe crear una nueva tabla que contendrá como atributos (llaves foráneas), los atributos llaves de cada entidad participante en la relación; adicionalmente, estos atributos serán las llaves primarias de la tabla. Si la relación entre las entidades del DER contiene algún atributo, éste debe ser incluido también en la nueva tabla que se cree.
7. Una vez identificadas todas las tablas, sus llaves primarias, llaves foráneas y atributos, se deben trazar flechas entre aquellas tablas que guarden alguna relación entre sí. La flecha apuntará a la tabla a la cual haga referencia a la llave foránea incluida en la otra tabla.
8. Dentro de cada tabla existirán atributos los cuales son obligatorios y otros que no lo son. Para distinguirlos se debe escribir en negrita aquellos que son obligatorios.
9. Las llaves primarias deben ser escritas subrayadas, acompañados por las letras PK (Primary Key).

10. Las llaves foráneas deben ser acompañadas por las letras FK (Foreign Key).

5.3. Modelo físico de la base de datos

Los datos que se almacenarán en la Base de Datos del SPEI3 tienen la siguiente conformación:

Información a guardar	Descripción
Información de clases	Contienen las clases del idioma inglés de nivel básico-intensivo que serán estudiadas por los alumnos, están organizadas en unidades, lecciones y temas.
Registro de avances por alumno	Consiste en un registro de las lecciones cursadas por alumno.
Registro de desempeño por alumno	Contiene la precisión con la cual el alumno cursa las lecciones y el tiempo que ha dedicado a ellas.
Información de ejercicios prácticos	Información para la realización de ejercicios prácticos de diferentes tipos: selección múltiple, formar palabras con letras desordenadas, rellenar espacios en blanco.
Información de evaluaciones	Información de las evaluaciones a realizar por las lecciones cursadas para obtener un resultado cualitativo.
Información de planificación de clases	Consiste en una planificación del orden de los elementos de las clases que el maestro establecerá según su criterio.
Registro de docentes	Incluye información de los docentes que están involucrados en el proceso de enseñanza del idioma inglés a personas no videntes.
Registro de alumnos	Incluye la información de los alumnos no videntes involucrados en el proceso de enseñanza del idioma inglés.
Registros de auditoría	Información de las actividades que están desempeñando los involucrados en el SPEI3.

TABLA 5-4

A continuación se muestra el modelo físico de la Base de Datos el cual proviene del Diagrama entidad relación antes presentado y de la conformación de datos del cuadro anterior.

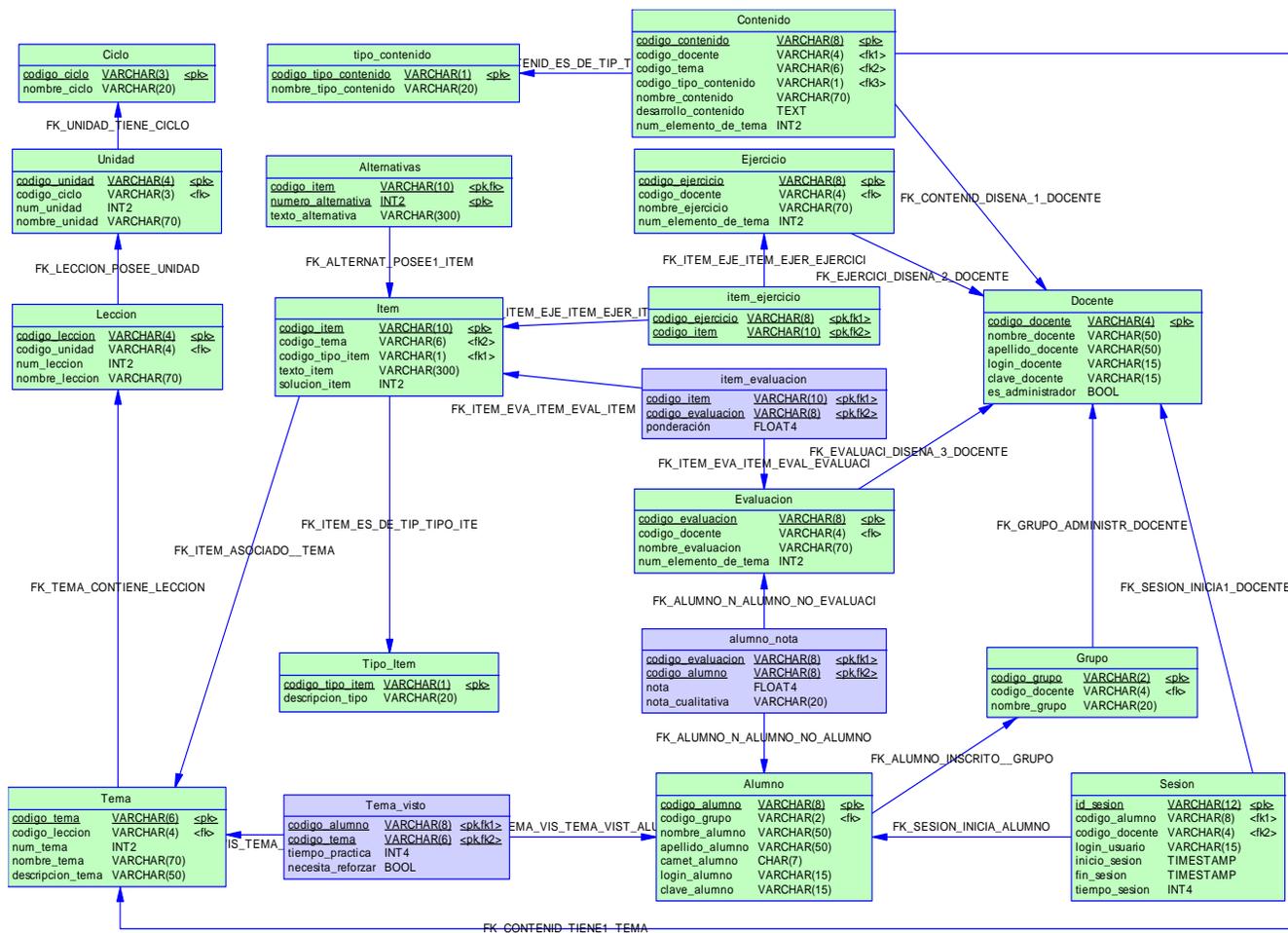


Figura 5-14 Modelo físico de la base de datos

Matriz de relaciones entre las tablas de la Base de Datos

	Ciclo	Unidad	Lección	Tema	Docente	Contenido	Ejercicio	Evaluación	Alumno_ nota	Alumno	Item	Alternativas	Tipo de Item	Grupo	Sesion	Item_ ejercicio	Item_ evaluacion	Tema_visto
Ciclo	X	X																
Unidad	X	X	X															
Lección		X	X	X														
Tema			X	X		X					X							X
Docente					X	X	X	X						X	X			
Contenido				X	X	X												
Ejercicio					X		X									X		
Evaluación					X			X									X	
Alumno_ nota								X	X	X								
Alumno									X	X				X	X			X
Item				X							X	X	X			X	X	
Alternativas											X	X						
Tipo de Item											X		X					
Grupo					X					X				X				
Sesion					X					X					X			
Item_ ejercicio							X				X					X		
Item_ evaluacion								X			X						X	
Tema_visto				X						X								X

TABLA 5-5

Documentación de las tablas del modelo físico

Las tablas involucradas en el modelo físico de la base de datos se documentarán usando el siguiente formato

Nombre de la tabla:	<Nombre>
<Imagen de la tabla>	Descripción: <Descripción de la tabla>
Llave primaria:	<Campo(s) que forman la llave primaria>
Llave foránea:	<Campo(s) que forman la(s) llave(s) foránea(s)>

Campos:

Nombre:	<Nombre del campo 1 >		
Descripción:	<Descripción del campo 1 >		
Tipo:	<Tipo de datos>	Longitud:	<máx. Longitud>
Requerido: Si / No	<Si o No>	Valor predeterminado:	
Valores permitidos:	<Rango de valores permitidos>		
Reglas de validación:	<Reglas que aseguran la validez del campo>		

...

...

...

Nombre:	<Nombre del campo N>		
Descripción:	<Descripción del campo N>		
Tipo:	<Tipo de datos>	Longitud:	<máx. Longitud>
Requerido: Si / No	<Si o No>	Valor predeterminado:	
Valores permitidos:	<Rango de valores permitidos>		
Reglas de validación:	<Reglas que aseguran la validez del campo>		

TABLA 5-6

Nombre de la tabla:	Ciclo										
	<table border="1"> <tr> <td colspan="3">Ciclo</td> </tr> <tr> <td>codigo_ciclo</td> <td>VARCHAR(3)</td> <td><pk></td> </tr> <tr> <td>nombre_ciclo</td> <td>VARCHAR(20)</td> <td></td> </tr> </table>	Ciclo			codigo_ciclo	VARCHAR(3)	<pk>	nombre_ciclo	VARCHAR(20)		Descripción: Cada fila de esta tabla representa un ciclo
Ciclo											
codigo_ciclo	VARCHAR(3)	<pk>									
nombre_ciclo	VARCHAR(20)										
Llave primaria:	codigo_ciclo										
Llave foránea:											

Campos:

Nombre:	codigo_ciclo		
Descripción:	Clave única que identifica cada ciclo		
Tipo:	varchar	Longitud:	3
Requerido: Si / No	Si	Valor predeterminado:	
Valores permitidos:			
Reglas de validación:	<>”		

Nombre:	nombre_ciclo		
Descripción:	Nombre asignado al ciclo		
Tipo:	Texto	Longitud:	20
Requerido: Si / No	Si	Valor predeterminado:	
Valores permitidos:			
Reglas de validación:	<>”		

TABLA 5-7

Para consultar toda la documentación de tablas del modelo físico de la base de datos puede hacerlo desde el CD adjunto a este documento en la opción “Documentación de tablas del modelo físico”.

Almacenes de datos históricos

Para el registro histórico de los datos de ciclos anteriores al ciclo actual se necesita tener ciertas tablas adicionales al modelo de Base de Datos del software con el objetivo de limpiar estas tablas para un mejor rendimiento en el software

- **Histórico mixto:** Las tablas cuyo volumen se considera elevado requiere la creación de una tabla adicional para almacenar sus datos históricos. Las tablas originales y las de bajo volumen de datos únicamente almacenan los datos del ciclo activo
- **Históricos con base de datos completa:** Todos los datos de las tablas se almacenan en una base de datos histórica con estructura exactamente igual a la base de datos original, esto sin depender del volumen de datos que manejan, esto permite que todas las tablas originales únicamente contengan los datos actuales del periodo y todos los datos de periodos anteriores se encuentra en las tablas de la base de datos histórica.
- **Históricos en la misma BD:** Todas las tablas contienen además del ciclo actual todos los datos de ciclos anteriores.

La opción de histórico a utilizar en este proyecto será el histórico mixto y las tablas adicionales se identificarán con un postfijo “_H” para diferenciarlas de las tablas operativas del ciclo actual.

Los siguientes diagramas nos ejemplifican el modelo histórico del SPEI3.

Modelo Entidad Relación Histórico

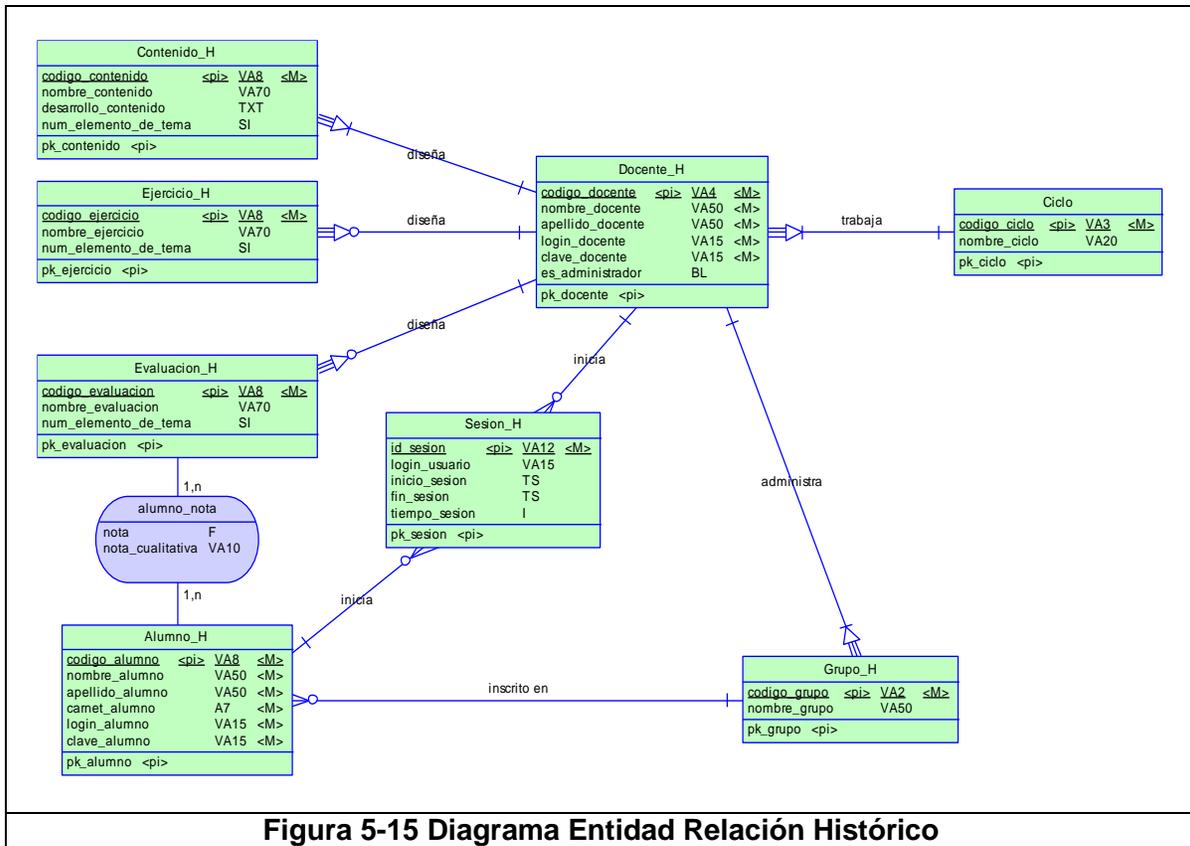


Figura 5-15 Diagrama Entidad Relación Histórico

Modelo físico Histórico

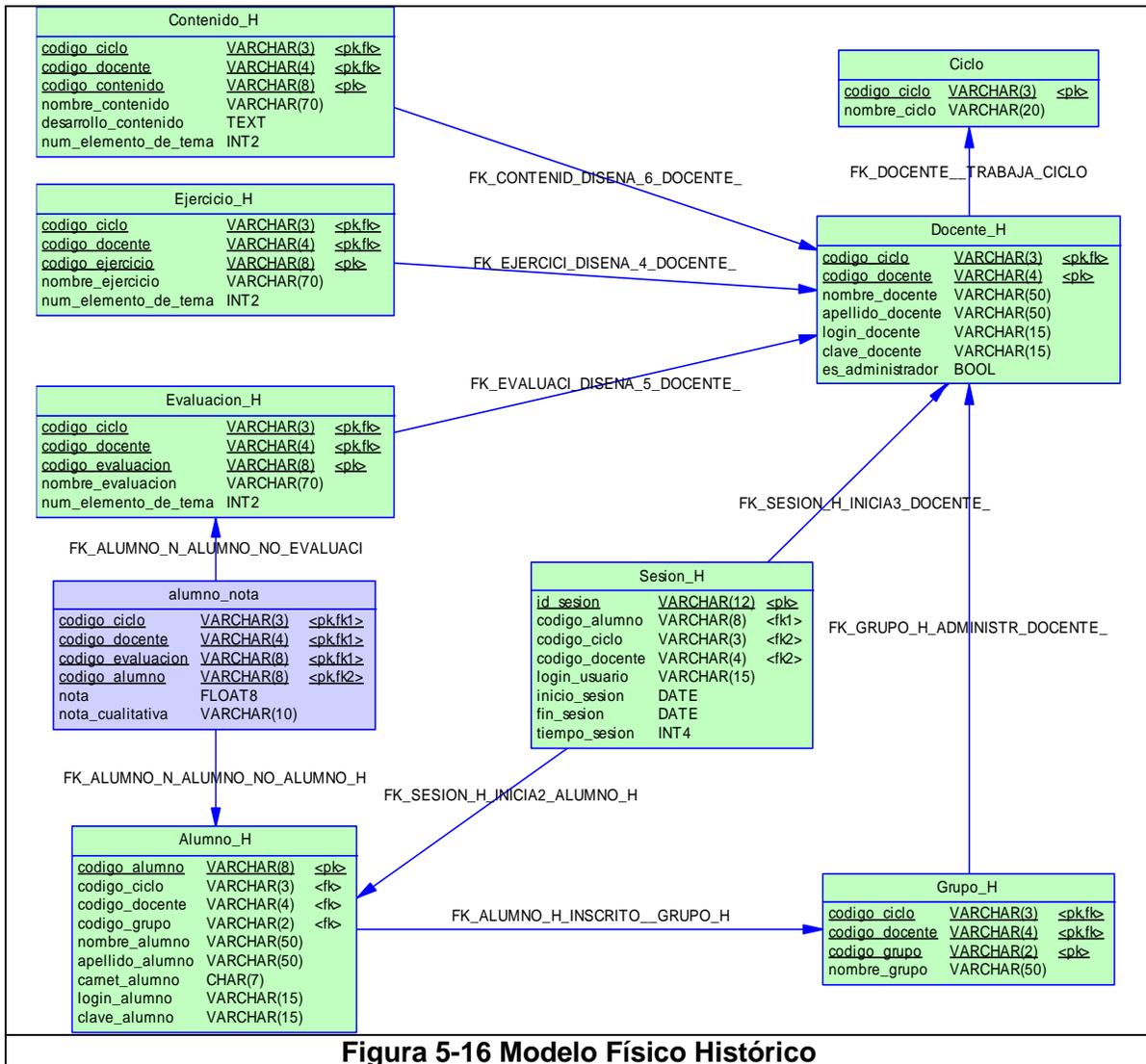


Figura 5-16 Modelo Físico Histórico

6. Casos de Uso Reales

Los procesos de software se pueden describir con casos de uso reales, estos se utilizan en el diseño de la aplicación, por ello para el SPEI3 en este apartado se presentan los casos de usos reales que contiene, clasificados por los subsistemas definidos en el apartado de la arquitectura del software 2.1.

6.1. Definición

Según el libro "UML y patrones" de Larman podemos ver que se hace una división de los casos de uso en tipos según conveniencia y la fase de proyecto en que uno se encuentre.

Casos de uso en formato de alto nivel.

"Se describe un proceso muy brevemente, en pocas líneas."

Conviene utilizar este tipo de formato durante el examen inicial de los requerimientos y del proyecto en sí, a fin de entender rápidamente el grado de complejidad y funcionalidad del sistema. Son casos muy vagos en lo que respecta a las decisiones de diseño.

Casos de uso en formato expandido.

"Se describe un proceso más a fondo"

La diferencia básica con el de alto nivel consiste en que cuenta con una sección destinada al detalle paso a paso de la evolución de los eventos.

Los casos de uso esenciales son casos expandidos expresados en forma teórica; esto supone que tienen pocos detalles sobre la futura implementación, las decisiones de diseño quedan pospuestas y se abstraen de la realidad (en especial aquellas concernientes a la interfaz de usuario, Por ej. Diremos "el usuario valida la introducción de los datos"; en vez de "el usuario pulsa el botón Aceptar", la manera en que se valida no queda detallada en el caso esencial).

Los casos de uso reales,

A diferencia de los otros casos de uso:

"Describen el proceso a partir de su diseño concreto".

Claro, estamos hablando de un proceso iterativo en el que inicialmente se definen los casos de uso de alto nivel y se pasaría después a crear un prototipo con más o menos detalle sobre el que posteriormente se basarían los casos de uso reales. Todo esto puede ser bueno o malo e incluso se puede cambiar el orden y omitir pasos (Por ej. nos saltamos los casos de alto nivel y pasamos directamente al prototipado) dependiendo de la envergadura de un proyecto y del tipo de proyecto.

Un Caso de Uso Real describe el diseño real del caso de uso según una tecnología concreta de entrada y de salida y su implementación. Si el caso de uso implica una interfaz de usuario, el caso de uso real incluirá bocetos de las ventanas y detalles de la

interacción a bajo nivel con los controles (botón, lista seleccionable, campo editable, etc.) de la ventana.

Como alternativa a la creación de los Casos de Uso Reales, el desarrollador puede crear bocetos de la interfaz en papel, y dejar los detalles para la fase de implementación.

6.2. Metodología para la Elaboración de Casos de Uso Reales

Para la elaboración de los casos de uso reales se deben realizar los siguientes pasos:

1. Elaborar un diseño previo de la Interfaz Gráfica de usuario que incluye ventanas y formularios que formarán parte del sistema
2. Partir de los casos de uso realizados en la etapa de análisis, y adaptar los pasos según convenga para mostrar la interacción que existiría entre los usuarios y la el software.
3. Se deben incluir en los pasos del caso de uso imágenes correspondientes al diseño de la interfaz gráfica de usuario involucrada en la ejecución de dicho paso.
4. Se debe incluir las excepciones de los casos de uso y adjuntarlas al caso de uso real como pasos alternos.

Especificación de los casos de uso reales

Para la especificación de los casos de uso reales se utilizará el siguiente formato:

CUR-##	Nombre del Caso de Uso
Actores	Nombre del actor o actores
Descripción	Descripción breve y clara del caso de uso.
Precondición	Estado necesario para iniciar con este caso de uso.
Post-condición	Estado resultante después del caso de uso.
Frecuencia esperada	Estimado de ocurrencias del caso de uso por un tiempo definido: Diaria, semanal, mensual, por ciclo, anual.
Pasos	Enumeración detallada de pasos ordenados lógicamente para llevar a cabo el objetivo el caso de uso.
Pasos Alternos	Enumeración detallada de pasos que se salen del curso normal de los eventos según diferentes situaciones.

TABLA 5-8

Actores: Incluye el nombre de los usuarios que se ven involucrados en la ejecución del caso de uso y que se ven.

Descripción: Se describe claramente la funcionalidad del caso de uso definiendo detalladamente los aspectos más relevantes que sirven para identificar el objetivo del mismo basado en un diseño específico y orientado a facilitar la etapa de construcción.

Precondición: Acción o estado que debe haberse completado previamente para que el caso de uso pueda iniciar, detalla las condiciones del entorno así como las acciones del usuario y respuestas del software que dan lugar a la ocurrencia del caso de uso.

Post-condición: Comprende las características de la situación al final de la realización del caso de uso.

Frecuencia esperada: Este valor corresponde con el número de veces durante un periodo de tiempo que se espera que el caso de uso se lleve a cabo, las escalas de tiempo pueden incluir la ejecución diaria, mensual, anual.

Respuestas del Software al usuario vidente.

La interacción por parte del software hacia el usuario que puede ver se mostrará en el caso de uso utilizando imágenes preliminares del diseño de la interfaz gráfica funcional (pantallas del software).

Respuestas del Software al usuario no vidente.

La interacción por parte del software hacia el usuario no vidente se mostrará en el caso de uso utilizando el siguiente formato de representación que indica que la salida del software al usuario es mediante la salida de audio estándar y que el software espera una entrada desde el teclado por parte del usuario no vidente.

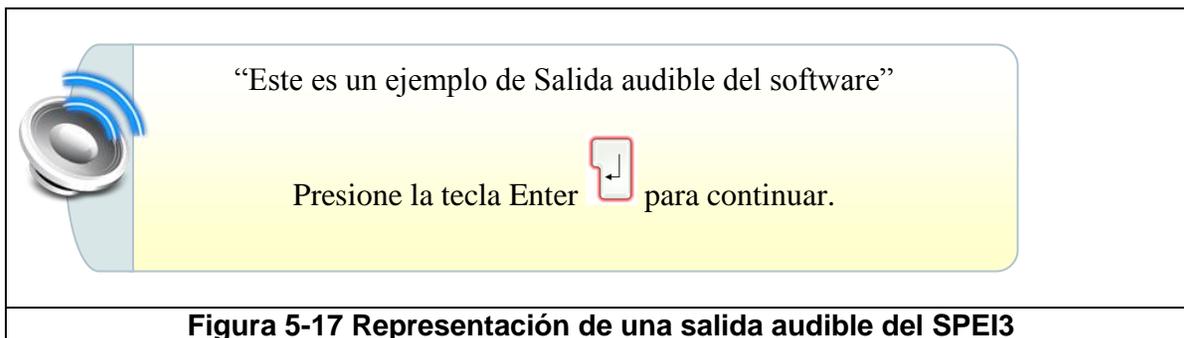


Figura 5-17 Representación de una salida audible del SPEI3

6.3. Casos de Uso Reales del Subsistema de Lecciones (A)

CUR-A1	Seleccionar tema a reproducir
Actores	Alumno.
Descripción	Este caso de uso le permite al usuario especificar el tema a ser reproducido.
Precondición	El alumno ha iniciado sesión satisfactoriamente en el software SPEI3 y ha seleccionado reproducir tema.
Post-condición	El tema ha sido seleccionado y es reproducido.
Frecuencia esperada	Diaria.

Pasos:

1. El SPEI3 comunica al usuario de las 12 unidades a cuales puede tener acceso una a una y se desplaza entre ellas con las teclas direccionales  , 



Seleccione la Unidad presionando la tecla Enter

- Unidad 1: Bienvenido
- Unidad 2: Conociendo Gente y Lugares
- Unidad 3: Piensa Verde
- Unidad 4: Impresiones
- Unidad 5: Casa y Hogar
- Unidad 6: Trabajos y Pasatiempos
- Unidad 7: De Viaje
- Unidad 8: Cerca de la Ciudad
- Unidad 9: Buen Provecho
- Unidad 10: Historias de Vida
- Unidad 11: Viendo Hacia Atrás
- Unidad 12: Metas

2. El usuario selecciona la unidad presionando la tecla Enter  al escuchar la unidad deseada.
3. El SPEI3 informa al usuario las lecciones a las que puede tener acceso y se desplaza entre ellas con las teclas direccionales  , 



Ha seleccionado la Unidad 1: Bienvenido
Puede escoger entre las lecciones:

- Lección 1: Conociendo gente
- Lección 2: Introducción
- Lección 3: En el club
- Lección 4: El recorrido para aprender inglés

4. El usuario selecciona la lección presionando la tecla Enter  al escuchar la lección deseada.
5. El SPEI3 comunica al usuario los temas que conforman la lección seleccionada y se desplaza entre ellos con las teclas direccionales , .



Ha seleccionado la Unidad 1, Lección 2 Introducción.
Puede escoger entre los temas:

- Tema 1: Gramática
- Tema 2: Funciones del lenguaje
- Tema 3: Vocabulario
- Tema 4: Pronunciación

6. El usuario selecciona el tema a reproducir presionando la tecla Enter  al escuchar el tema deseado.
7. El SPEI3 obtiene los elementos que conforman el tema seleccionado (contenidos, ejercicios, evaluaciones) y se encarga de reproducir las cadenas de texto en el idioma correspondiente.



Las teclas de control que puede usar son:

	→ Pausa la reproducción,
+  ó - 	→ Cambiar entre contenidos
Esc 	→ Detener Reproducción

{“ ... INICIA REPRODUCCIÓN ... ”}

Pasos Alternos

- 1, 3, 5 Si el usuario desea cancelar la operación presiona la tecla Esc  y vuelve al menú anterior.

Para consultar todos los casos de uso reales del diseño puede hacerlo desde el CD adjunto a este documento en la opción “Casos de uso reales”.

7. Diagramas de colaboración

En este apartado como parte del diseño del SPEI3 se realizan los diagramas de interacción para conocer la interacción entre los objetos del software, para ello se da una definición de diagramas de colaboración una metodología y se presentan los diagramas de colaboración del SPEI3.

7.1. Definición

Diagramas de Interacción

Este diagrama describe secuencias de intercambios de mensajes entre los roles que implementan el comportamiento de un sistema. Un rol clasificador, o simplemente "un rol", es la descripción de un objeto, que desempeña un determinado papel dentro de una interacción, distinto de los otros objetos de la misma clase. Esta visión proporciona una vista integral del comportamiento del sistema, es decir, muestra el flujo de control a través de muchos objetos. Los diagramas de interacción se exhiben en dos diagramas centrados en distintos aspectos pero complementarios: Centrados en los objetos individuales y centrados en objetos cooperantes.

Los objetos interactúan para realizar colectivamente los servicios ofrecidos por las aplicaciones. Los diagramas de interacción muestran cómo se comunican los objetos en una interacción. Existen dos tipos de diagramas de interacción: el *Diagrama de Colaboración* y el *Diagrama de Secuencia*.

El Diagrama de Colaboración ofrece una mejor visión espacial mostrando los enlaces de comunicación entre objetos, muestra las relaciones entre objetos y son mejores para comprender todos los efectos que tiene un objeto y para el diseño de procedimientos.

Los Diagramas de Interacción muestran el intercambio de mensajes entre instancias del modelo de clases para cumplir las post-condiciones establecidas en un contrato.

De entre ambos tipos se prefieren los Diagramas de Colaboración por su expresividad y por su economía espacial (una interacción compleja puede ser muy larga en un Diagrama de Secuencia).

La creación de los Diagramas de Colaboración de un sistema es una de las actividades más importantes en el desarrollo orientado a objetos, pues al construirlos se toman unas decisiones clave acerca del funcionamiento del futuro sistema.

Diagramas de Colaboración:

Es un Diagrama que muestra interacciones organizadas alrededor de los roles. A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran explícitamente las relaciones de los roles. Por otra parte, un diagrama de colaboración no muestra el tiempo como una dimensión aparte, por lo que resulta necesario etiquetar con números de secuencia tanto la secuencia de mensajes como los hilos concurrentes. Un diagrama de colaboración es también un diagrama de clases que contiene roles de clasificador y roles de asociación en lugar de sólo clasificadores y asociaciones. Los roles de clasificador y los de asociación describen la configuración de los objetos y de los enlaces que pueden ocurrir cuando se ejecuta una instancia de la colaboración. Cuando se instancia una colaboración, los objetos están ligados a los roles de clasificador y los enlaces a los roles de asociación. El rol de asociación puede ser desempeñado por varios tipos de enlaces temporales, tales como argumentos de procedimiento o variables locales del procedimiento. Los símbolos de enlace pueden llevar estereotipos para indicar enlaces temporales.

Un uso de un diagrama de colaboración es mostrar la implementación de una operación. La colaboración muestra los parámetros y las variables locales de la operación, así como asociaciones más permanentes. Cuando se implementa el comportamiento, la secuencia de los mensajes corresponde a la estructura de llamadas anidadas y el paso de señales del programa.

Un diagrama de colaboración muestra relaciones entre roles geoméricamente y relaciona los mensajes con las relaciones, pero las secuencias temporales están menos claras.

Es útil marcar los objetos en cuatro grupos: los que existen con la interacción entera; los creados durante la interacción (restricción {new}); los destruidos durante la interacción (restricción {destroyed}); y los que se crean y se destruyen durante la interacción (restricción {transient}). Aunque las colaboraciones muestran directamente la implementación de una operación, pueden también mostrar la realización de una clase entera. En este uso, muestran el contexto necesario para implementar todas las operaciones de una clase. Esto permite que el modelador vea los roles múltiples que los objetos pueden desempeñar en varias operaciones.

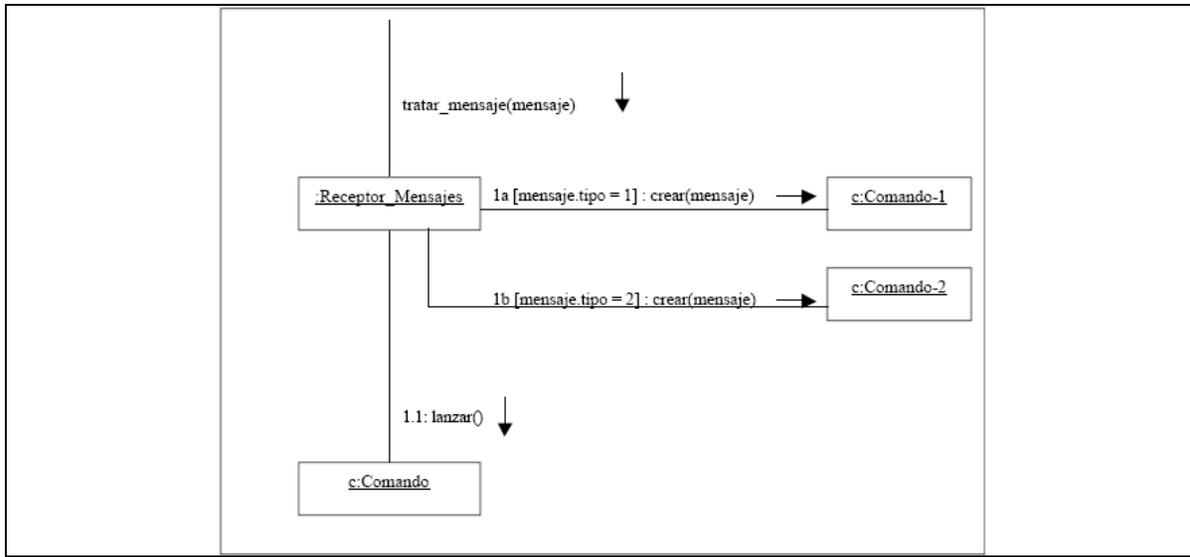


Figura 5-18 Diagrama de colaboración

Mensajes

Los mensajes se muestran como flechas etiquetadas unidas a los enlaces. Cada mensaje tiene un número de secuencia, una lista opcional de mensajes precedentes, una condición opcional de guarda, un nombre y una lista de argumentos y un nombre de valor de retorno opcional. El nombre de serie incluye el nombre (opcional) de un hilo. Todos los mensajes del mismo hilo se ordenan secuencialmente. Los mensajes de diversos hilos son concurrentes a menos que haya una dependencia secuencial explícita.

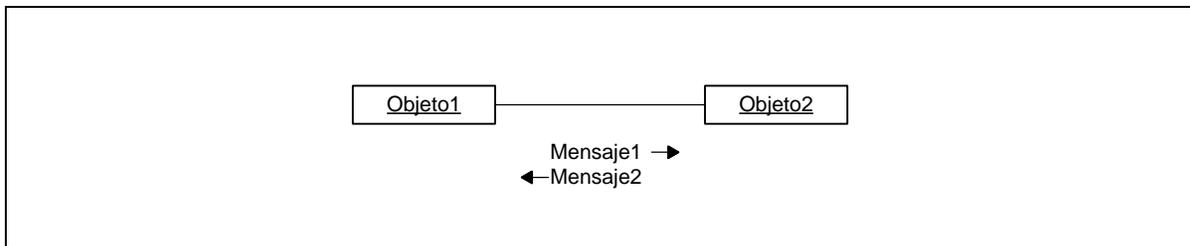


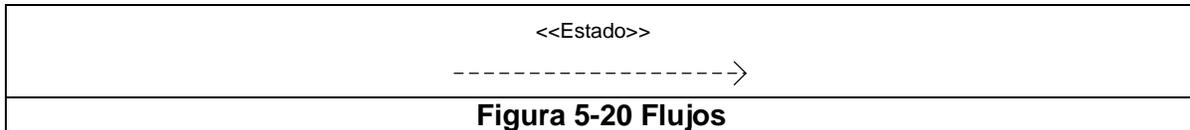
Figura 5-19 Mensajes entre objetos

Flujos

Generalmente, un diagrama de colaboración contiene un símbolo para un objeto durante una operación completa. Sin embargo, a veces, un objeto contiene diferentes estados que se deban hacer explícitos. Por ejemplo, un objeto pudo cambiar de localización o sus asociaciones pudieron diferenciarse.

Los diferentes símbolos de objeto que representan un objeto se pueden conectar usando flujos "become" o "conversion". Un flujo "become" es una transición, a partir de un estado de un objeto a otro. Se dibuja como una flecha de línea discontinua con el estereotipo

"become" o "conversion" y puede ser etiquetado con un número de serie para mostrar cuando ocurre. Un flujo de conversión también se utiliza para mostrar la migración de un objeto a partir de una localización a otra distinta.



7.2. Metodología para la elaboración de Diagramas de Colaboración

Los Diagramas de Colaboración destacan la organización de los objetos que participan en la interacción, muestran como se enlazan los objetos destacando las relaciones estructurales entre las instancias a través de las cuales pasan los mensajes.

Para crear los Diagramas de Colaboración se pueden seguir los siguientes consejos:

- Crear un diagrama separado para cada operación del sistema en desarrollo en el ciclo de desarrollo actual
- Para cada evento del sistema, hacer un diagrama con él como mensaje inicial
- Si el diagrama se complica, dividirlo en diagramas más pequeños.
- Usando los apartados de responsabilidades y de post-condiciones del contrato de operación, y la descripción del caso de uso como punto de partida, diseñar un sistema de objetos que interaccionan para llevar a cabo las tareas requeridas.

La capacidad de realizar una buena asignación de responsabilidades a los distintos objetos es una habilidad clave, y se va adquiriendo según aumenta la experiencia en el desarrollo orientado a objetos. Booch, Rumbaugh y Jacobson definen **responsabilidad** como “un contrato u obligación de una clase o tipo” [BJR97]. Las responsabilidades están ligadas a las obligaciones de un objeto en cuanto a su comportamiento. Básicamente, estas responsabilidades son de los dos siguientes tipos:

- Conocer:
 - Conocer datos privados encapsulados.
 - Conocer los objetos relacionados.
 - Conocer las cosas que puede calcular o derivar.
- Hacer:
 - Hacer algo él mismo.
 - Iniciar una acción en otros objetos.
 - Controlar y coordinar actividades en otros objetos.

Por ejemplo, puedo decir que “un *Recibo* es responsable de imprimirse” (tipo hacer), o que “una *Transacción* es responsable de saber su fecha” (tipo conocer).

Las responsabilidades de tipo “conocer” se pueden inferir normalmente del Modelo Conceptual.

Una responsabilidad no es lo mismo que un método, pero los métodos se implementan para satisfacer responsabilidades.

7.3. Diagramas de colaboración

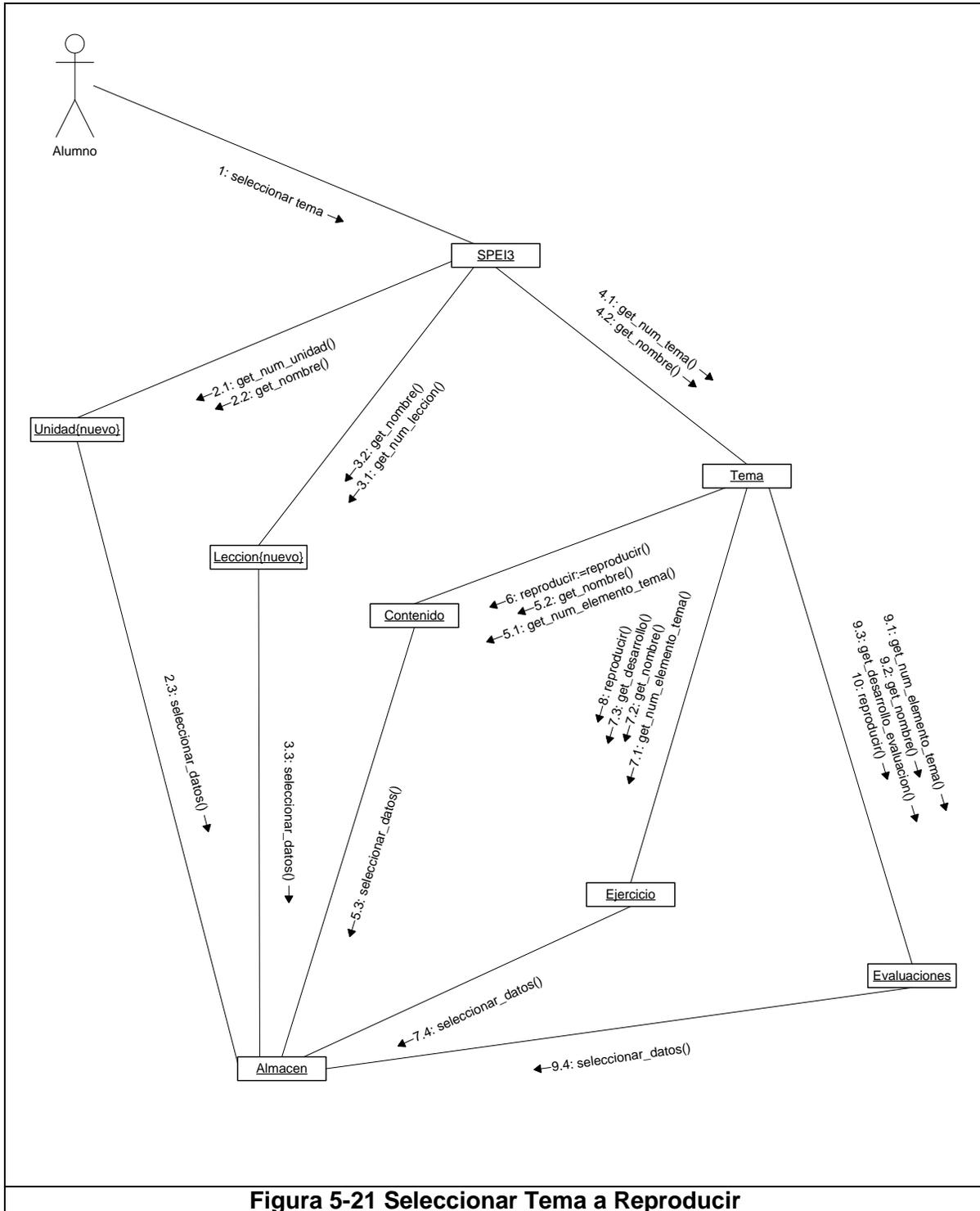


Figura 5-21 Seleccionar Tema a Reproducir

Para consultar todos los diagramas de colaboración del diseño puede hacerlo desde el CD adjunto a este documento en la opción "Diagramas de colaboración".

8. Diagrama de clases

En este apartado, se presenta las clases del SPEI3, para ello se da una definición, una metodología para elaboración, se presenta el diagrama de clases y su documentación.

8.1. Definición

El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. El modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones.

El mundo real puede ser visto desde abstracciones diferentes (subjetividad) o mecanismos de abstracción, dichos mecanismos se clasifican en:

- Clasificación / Instanciación
- Composición / Descomposición
- Agrupación / Individualización
- Especialización / Generalización

La **clasificación** es uno de los mecanismos de abstracción más utilizados. La clase define el ámbito de definición de un conjunto de objetos, y cada objeto pertenece a una clase, Los objetos se crean por **instanciación** de las clases.

8.2. Metodología para la elaboración del diagrama de clases

Marco Teórico

Para comprender lo que un diagrama de clases representa, es necesario tomar en cuenta algunas definiciones:

- Clase: Es la implementación de un tipo de datos abstracto (TDA). Define atributos y métodos que implementan la estructura de datos y operaciones del TDA, respectivamente. Las instancias de las clases son llamadas objetos. Consecuentemente, las clases definen las propiedades y el comportamiento de conjuntos de objetos.
- Atributos: Son los elementos comprendidos por una clase y que almacenan los valores necesarios para la utilización de la instancia de una clase (objeto).
- Métodos: Son aquellos elementos de la clase encargados de modificar sus atributos. Son rutinas enfocadas al manejo de los atributos para la realización de operaciones con éstos.
- Relaciones: Los enlaces entre objetos pueden representarse entre las respectivas clases y sus formas de relación son:
 - Asociación

- Agregación
- Generalización/Especialización.

Las relaciones de Agregación y Generalización forman jerarquías de clases.

- Asociación: La asociación expresa una conexión bidireccional entre objetos. Una asociación es una abstracción de la relación existente en los enlaces entre los objetos. Puede determinarse por la especificación de multiplicidad (mínima...máxima)
 - *Uno y sólo uno*
 - *0..1 Cero o uno*
 - *m..n Desde M hasta N (enteros naturales)*
 - *0..* Cero o muchos*
 - *1..* Uno o muchos (al menos uno)*

- Agregación: La agregación representa una relación *parte_de* entre objetos. En UML se proporciona una escasa caracterización de la agregación. Esta relación puede ser caracterizada con precisión determinando las relaciones de comportamiento y estructura que existen entre el objeto agregado y cada uno de sus objetos componentes.

Una agregación se podría caracterizar según:

1. *Puede el objeto parte comunicarse directamente con objetos externos al objeto agregado?*

No => inclusiva

Si => no inclusiva

2. *Puede cambiar La composición del objeto agregado?*

Si => dinámica

No => estática

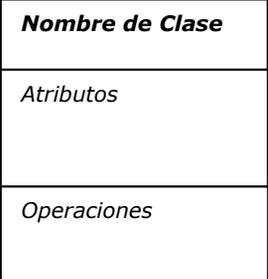
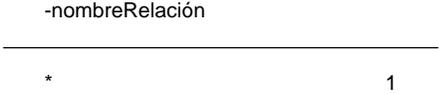
- Generalización: Permite gestionar la complejidad mediante un ordenamiento taxonómico de clases, se obtiene usando los mecanismos de abstracción de Generalización y/o Especialización. La Generalización consiste en factorizar las propiedades comunes de un conjunto de clases en una clase más general. La especialización es una técnica muy eficaz para la extensión y reutilización.

Pasos a seguir para la elaboración del diagrama de clases

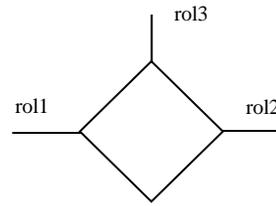
1. Elaborar una lista de las clases que debería incluir el sistema que se está diseñando. Estos conceptos son extraídos del mundo real y es necesario listar siempre los más importantes.
2. Definir los atributos y métodos pertenecientes a cada clase, tomando en cuenta el funcionamiento y finalidad de dichas clases.

3. Identificar las relaciones existentes entre las diferentes clases, tomando en cuenta si se trata de una asociación, una agregación o una generalización/especialización.
4. Una vez definidos todos los elementos mencionados con anterioridad, es necesaria la representación gráfica de éstos, siguiendo las notaciones de UML.

FORMATO DE PRESENTACIÓN

Figura	Presentación
<p>Clase: Se refiere al Tipo de Datos Abstracto formado por atributos y métodos, base de la Programación Orientada a Objetos.</p> <p>Cada clase se representa en un rectángulo con tres compartimientos:</p> <ul style="list-style-type: none"> • Nombre de la clase • Atributos de la clase • Operaciones de la clase 	
<p>Relación: Se refiere a los enlaces existentes entre las clases y que se dividen en:</p> <ul style="list-style-type: none"> • Asociación • Agregación • Generalización <p>Se representan con una línea recta que une a las dos clases relacionadas, con números en los extremos que indican la cardinalidad de cada clase.</p> <p>Cuando la cardinalidad es de “muchos”, se representa con un asterisco (*), cuando se refiere a un intervalo, se representa con los extremos del intervalo, separados por puntos suspensivos y encerrados en un paréntesis; por ej. si fuera un intervalo de uno a muchos se representaría así: (1..*).</p>	

Asociación N-aria: Consiste en una asociación en la que participan más de dos clases, las clases se unen con una línea a un diamante central. Si se muestra multiplicidad en un rol, representa el número potencial de tuplas de instancias en la asociación cuando el resto de los N-1 valores están fijos.



8.3. Diagrama de clases

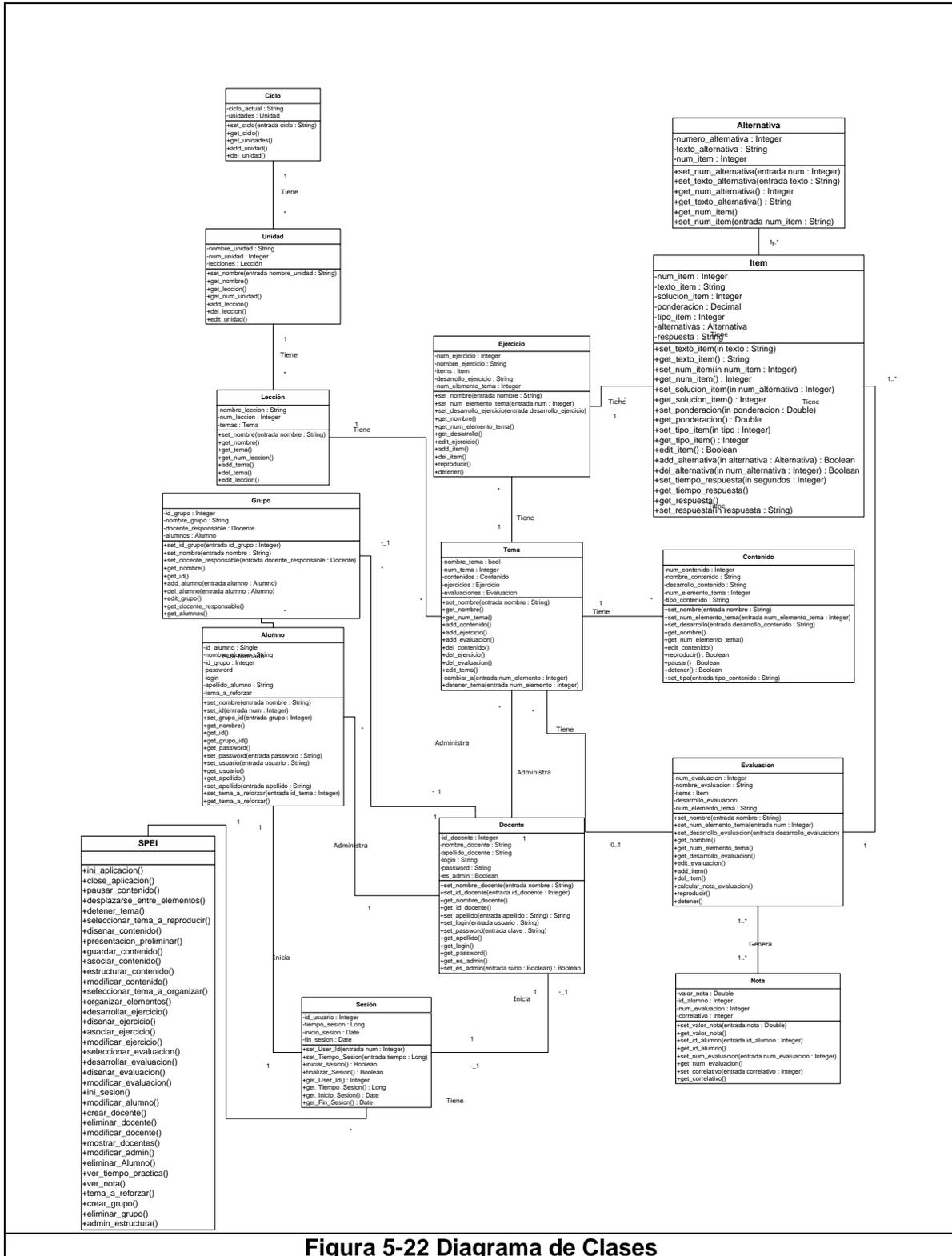


Figura 5-22 Diagrama de Clases

8.4 Documentación de clases

Nombre de Clase: 1	
Paquete al que pertenece: 2	
Súper Clase: 3	
	Descripción: 4

Atributos 5

Nombre 6	Descripción 7	Tipo 8	Long. 9	Val. Perm. 10
----------	---------------	--------	---------	---------------

Métodos 11

Nombre: 12			
Descripción: 13			
Valor que devuelve: 14			
Parámetros 15			
Nombre 16	Tipos 17	Calificador 18	Descripción 19

Leyenda

1. Nombre de la clase
2. Nombre del paquete al que pertenece
3. Clase de la que hereda
4. Descripción del funcionamiento o papel que desempeña la clase y figura de la clase mostrando atributos y operaciones
5. Sección de atributos
6. Nombre del atributo
7. Descripción del atributo
8. Tipo de dato o nombre de la clase de la

- que es instancia el objeto
9. Longitud permitida
10. Valores permitidos por el atributo. Si es objeto no se especifica
11. Sección de métodos
12. Nombre del método
13. Descripción del método
14. Tipo de dato devuelto o clase a la que hace instancia el objeto devuelto
15. Sección de parámetros
16. Nombre del parámetro
17. Tipo de dato o clase de la cual es instancia el objeto
18. Forma de transferir el parámetro (valor o referencia)
19. Descripción del contenido del parámetro

Nombre de Clase: Ciclo									
Paquete al que pertenece: SPEI3									
Súper Clase:									
<table border="1"> <thead> <tr> <th>Ciclo</th> </tr> </thead> <tbody> <tr> <td>-ciclo_actual : String</td> </tr> <tr> <td>-unidades : Unidad</td> </tr> <tr> <td>+set_ciclo(entrada ciclo : String)</td> </tr> <tr> <td>+get_ciclo()</td> </tr> <tr> <td>+get_unidades()</td> </tr> <tr> <td>+add_unidad()</td> </tr> <tr> <td>+del_unidad()</td> </tr> </tbody> </table>	Ciclo	-ciclo_actual : String	-unidades : Unidad	+set_ciclo(entrada ciclo : String)	+get_ciclo()	+get_unidades()	+add_unidad()	+del_unidad()	Descripción: La creación de esta permitirá el manejo de los ciclos ordinarios en los que se desarrollan las actividades educativas a las que dará soporte el SPEI3
Ciclo									
-ciclo_actual : String									
-unidades : Unidad									
+set_ciclo(entrada ciclo : String)									
+get_ciclo()									
+get_unidades()									
+add_unidad()									
+del_unidad()									

Atributos

Nombre	Descripción	Tipo	Long.	Val. Per.
ciclo_actual	Almacena el nombre del ciclo al que representa la instancia de esta clase	String	7	Únicamente nombres de ciclos
unidades	Almacenas las unidades de la asignatura comprendidas en el ciclo representado	Unidad	N/A	N/A

Métodos

Nombre:	set_ciclo		
Descripción:	Este método permite establecer el ciclo que representa la instancia de esta clase		
Valor que devuelve:	N/A		
Parámetros			
Nombre	Tipos	Calificador	Descripción
ciclo	String	valor	Almacena la cadena que se desea establecer como nombre del ciclo

Nombre:	get_ciclo
Descripción:	Este método permite obtener el ciclo que representa la instancia de esta clase
Valor que devuelve:	Ciclo
Parámetros: No posee	

Nombre:	get_unidades
Descripción:	Este método permite obtener las unidades que conforman el ciclo representado
Valor que devuelve:	Unidad
Parámetros: No posee	

Nombre:	add_unidad		
Descripción:	Este método permite agregar una unidad al ciclo representado por la instancia de esta clase		
Valor que devuelve:	Boolean		
Parámetros			
Nombre	Tipos	Calificador	Descripción
unidad	String	valor	Almacena la cadena que se desea establecer como unidad

Nombre:	del_unidad		
Descripción:	Este método permite eliminar una unidad al ciclo representado por la instancia de esta clase		
Valor que devuelve:	Bolean		
Parámetros			
Nombre	Tipos	Calificador	Descripción
unidad	String	Valor	Almacena el nombre de la unidad que se desea eliminar

Para consultar toda la documentación de clases puede hacerlo desde el CD adjunto a este documento en la opción “Documentación de clases”.

Capítulo VI: Desarrollo

1. Ámbito de programación

La programación de SPEI3 (Software pedagógico para la enseñanza del idioma inglés a estudiantes invidentes) ha sido desarrollada utilizando una red de área local creada para propiciar la consistencia de la información almacenada de forma centralizada.

2. Condiciones tecnológicas

Para el desarrollo de SPEI3 se utilizaron los siguientes recursos computacionales:

Hardware	Descripción	Cantidad
Computadora personal	Pentium VI, Procesador 3.0 Ghz, 384MB RAM, Disco duro de 40 GB	4
Impresor	Canon PIXMA IP1000	1
Scanner	Acer <u>ScanPrisa</u> 640p	1
Quemador de CD	LG Super Multi	1
UPS	600VA, 15Min	4

Tabla 6-1

3. Configuración del ambiente de desarrollo

Para desarrollar el proyecto se creó una red de topología estrella, para la cual fue necesario utilizar un SWITCH ethernet 10/100 de 8 puertos y cableado para instalar la red.

Se utilizaron las instalaciones de un local dedicado exclusivamente para el desarrollo del proyecto.

4. Realización de pruebas

Se realizaron diferentes tipos de pruebas para garantizar el funcionamiento esperado de SPEI3, de las cuales podemos mencionar:

Pruebas Unitarias

Este tipo de pruebas se realizó luego de la programación de los objetos individuales que se han utilizado para ensamblar objetos más grandes, este tipo de pruebas permitió mantener un control de calidad sobre los elementos que integran módulos más grandes, permitiendo que las partes que lo formaban se encontraran estables al momento de la utilización.

Pruebas de Integración

Fueron realizadas luego de la integración de todos los módulos que conforman SPEI3, tanto para el diseño como presentación para el estudiante, este proceso de pruebas implicó muchas complicaciones debido a que al crear los enlaces entre los objetos

surgieron nuevos cambios en la generación de mensajes para que todos los objetos respondieran al funcionamiento global esperado.

Pruebas finales del software integrado

Para realizar las pruebas finales de integración del software se utilizaron los casos de usos que fueron definidos en el análisis, permitiendo verificar que el funcionamiento que se esperaba haya sido cumplido, se agregaron algunas de las modificaciones que se realizaron en el funcionamiento de algunos módulos. Para cada módulo se plantea la forma en que fue probado y los errores que esas pruebas generaron. Se realizó una prueba por módulo basada en los requerimientos del software comparando el funcionamiento esperado con el real. Si desea mayor detalle puede consultar el manual de pruebas que se encuentra en el CD adjunto a este documento.

5. Técnicas de programación

Para el desarrollo de SPEI3 se utilizó exclusivamente la programación orientada a objetos. Incluyendo el software en un paquete (conjunto de clases y otros paquetes). Para realizar la división del software en clases se utilizó el tipo de arquitectura Modelo Vista Controlador que separa las definiciones lógicas, objetos de frontera o interfaz y objetos de control. Además se hizo uso de las características de herencia y polimorfismo de la POO (Programación orientada a objetos).

Capítulo VII: Documentación externa del software

1. Manual del usuario

En el manual del usuario usted encontrará la documentación y descripción para todos los procesos que puede llevar a cabo desde el SPEI3 de la siguiente manera:

- A) Manual del Usuario (documento): Este documento va orientado a los roles administrador y docente. En el encontrarán las diferentes funciones de SPEI3 y se le explicará la manera de utilizarlas. Este puede ser consultado desde el CD adjunto a este documento en la opción “Manuales” y luego en la opción “Manual del usuario”.
- B) Manual del Usuario Alumno (CD de audio): Este es un CD adicional orientado al alumno invidente, creado en formato de Audio para que sea fácilmente percibido por el estudiante. Este le describirá la manera en la que hará uso de la interfaz de usuario de SPEI3 y todas sus funciones y características.

2. Manual técnico

En este manual usted encontrará una guía que le permitirá extender la funcionalidad de SPEI3, en el se describe el lenguaje de programación, las diferentes clases que componen el SPEI3, los diferentes formularios, reportes y base da datos. Este puede ser consultado en el CD adjunto a este documento en la opción “Manuales” y luego en la opción “Manual técnico”.

3. Manual de Instalación

El manual de instalación es una guía clara y sencilla que le indicara los diferentes paso a seguir para llevar a cabo la instalación del SPEI3, mostrándole las pantallas e indicando las opciones que debe seleccionar. Inicia con el gestor de bases de datos, continua con software adicional necesario para el funcionamiento del SPEI3 (Net Frameworks 2.0, Windows Installer 3.11 y las librerías de las voces) y finaliza con la guía del instalador del SPEI3. Este puede ser consultado en el CD adjunto a este documento en la opción “Manuales” y luego en la opción “Manual de instalación”.

CONCLUSIONES

SPEI3 se concibe como un apoyo para el docente en su labor y como un insumo a la metodología de la enseñanza y no un sustituto de la misma.

SPEI3 puede ser fácilmente adoptado para apoyar el proceso de enseñanza – aprendizaje de otra asignatura a estudiantes invidentes.

SPEI3 no restringe al docente a contenidos y ejercicios estáticos, sino que le proporciona una gama de posibilidades para la creación y modificación de los mismos utilizando su creatividad y su propia pedagogía.

Además del apoyo que brinda SPEI3 en cuanto a contenidos, ejercicios y evaluaciones, el software proporciona los suficientes elementos para llevar a cabo un seguimiento del avance del alumno el cual se basa en los temas recibidos y el desempeño del alumno en las evaluaciones.

SPEI3 es una herramienta, de la cual manifiesta el departamento de idiomas de la Facultad de Ciencias y Humanidades de la Universidad de El Salvador, que aunque no soluciona totalmente la problemática a la que se enfrentan los docentes de alumnos invidentes, ayuda en gran medida a mejorar los procesos de enseñanza actuales que no contemplan las limitantes de los estudiantes invidentes.

Glosario de términos

- *Ciclo*: Periodo comprendido entre la inscripción de asignaturas y la entrega de notas finales, consta de 16 semanas de duración, dentro del cual se llevan a cabo todas las actividades académicas, como clases, evaluaciones, tareas, diferidos, etc.
- *Alumno o estudiante*: Persona que cursa sus estudios en una institución de enseñanza.
- *Docente*: Persona encargada de enseñar los contenidos de la asignatura que le ha sido asignada en un ciclo determinado.
- *Examen de suficiencia*: Evaluación especial que se realiza al final de cada ciclo con el objetivo de aprobar una asignatura que ha sido reprobada en el ciclo actual, siempre y cuando el alumno cumpla el requisito de haber reprobado la asignatura con una nota final entre 5.0 y 5.9, esta nota del examen de suficiencia va directamente promediada con la nota final de ciclo.
- *Evaluación diferida*: Evaluación de reposición a la un alumno puede optar al no haber hecho una evaluación normal, siempre y cuando haya sido por una causa justificada, el alumno debe tramitar la *Solicitud de Evaluación Diferida* dentro de los primeros tres días hábiles posteriores a la evaluación normal, y es opción del docente la fecha de realización, que puede ser cercana a la evaluación normal y puede tener el mismo contenido de la evaluación normal o ser de tipo acumulativo y hacerse al final del ciclo, las ponderaciones son las mismas entre la evaluación normal y la evaluación diferida.
- *Warm-up*: Término utilizado en la enseñanza del idioma inglés para indicar la etapa de la clase en la cual se exploran los conocimientos previos de los alumnos acerca del tema a tratar y se genera una idea global hacia los alumnos del contenido del tema en general.
- *Practica controlada*: Término utilizado en la enseñanza del idioma inglés para indicar que la actividad del estudiante sigue un curso de acción predefinido por el docente, este tipo de práctica permite evaluar al estudiante de una manera mas objetiva que en una practica no controlada o abierta.
- *Audífonos*: Aparato que tiene por objetivo permitir que quien los usa perciba mejor los sonidos.
- *Teclado convencional*: Periférico de entrada que permite la comunicación la del usuario con el computador, y mediante el cual se le pueden introducir órdenes e información, el teclado mas común tiene 102 teclas entre letras, números y teclas de función.

BIBLOGRAFIA

Libros:

- ✓ “El Lenguaje Unificado de Modelado” Autores: Rumbaugh James, Grady Booch, Jacobson Ivar. Editorial: Addison Wesley Iberoamericana, Madrid 1999.
- ✓ “Análisis y Diseño de Sistemas” Centro de Computación Profesional de México CCPM. Editorial “McGraw-Hill” Primera Edición, Junio de 2002, México.
- ✓ “Ingeniería de Software”, Autor: Ian Sommerville. Editorial “Pearson Educación” SEXTA EDICION 2002, México.

Tesis:

- ✓ “Software de apoyo a la enseñanza del lenguaje signado y la lecto-escritura del idioma español para personas sordas” Escuela de Ingeniería de sistemas informáticos, Facultad de Ingeniería y arquitectura, Universidad de El Salvador. Febrero de 2003.

Documentos:

- ✓ UML Total
 - Xavier Ferré Grau, María Isabel Sánchez Segura
 - Facultad de Informática – UPM

Sitios de Internet:

Los siguientes sitios sirvieron de fuente para estructurar la metodología para la elaboración del diagrama de clases.

- ✓ www.sel.unsl.edu.ar/licenciatura/ingsoft2/UML-DiagramaClaseObjeto.pdf
- ✓ <http://www.osmosislatina.com/lenguajes/uml/clasesob.htm>
- ✓ <http://www.creangel.com/uml/clases.php>
- ✓ http://es.wikipedia.org/wiki/Diagrama_de_clases

El siguiente sitio sirvió de fuente para estructurar la metodología para la elaboración de diagramas de colaboración.

- ✓ <http://www.creangel.com/uml/colaboracion.php>

Anexo 1 – Entrevistas con Mtl. Edgar Nicolás Ayala

Primera Entrevista:

El video de esta entrevista puede ser visualizado desde el CD adjunto a este documento en la opción “Videos” y luego en la opción “**Entrevista con Mtl. Edgar Nicolás Ayala**”.

Segunda Entrevista:

Entrevista realizada al Director del Departamento de Idiomas Mtl. Edgar Nicolás Ayala de la Facultad de Ciencias y Humanidades de la UES.

El objetivo de esta entrevista fue obtener una especificación mas profunda sobre los requerimientos del SPEI3.⁴⁹

1. ¿El software debería proveer formas para que el docente introdujera el contenido de una unidad, temas específicos?

- ❖ *Los contenidos de las unidades NO cambiarán, no es necesario, ni en los programas ni en los contenidos del programa de la materia ya que para eso se tiene una programación, los libros de ingles por ejemplo, siempre tienen el mismo contenido, lo único que cambia es la forma con que el docente da ese contenido.*

2. ¿Se podría planear las unidades que se verán en el software?

- ❖ *No, los programas ya están, los docentes solo se reúnen para planificar la duración y que libro guía se va a utilizar, Solo se modificara la forma en que se impartirá la clase, el docente puede introducir una técnica nueva, una nueva forma de dar el contenido de la clase pero no la clase en si.*

3. ¿Qué tipos de informe necesitaría que el software le brindara?

- ❖ *Necesitaría que si se manda a un estudiante a practicar un tema, tendría que saber, que fue lo que hizo, cuanto tiempo lo práctico, con que precisión lo hizo, la nota que sacó.*

4. El sistema no brindará notas cuantitativas...

- ❖ *Pero siempre tendría que haber una escala, porque al maestro lo que le interesaría es saber que tan bien realiza el estudiante una practica, siempre tiene que ver categorías de notas si no van a ser cuantitativas, entonces tendría que mostrarse la categoría de la nota*

– Todos los estudiantes deben tener una computadora, usted podría ir maquina por maquina viendo los resultados o recibir información directamente en su computadora de todo lo que se haga...

⁴⁹ Las intervenciones con viñeta y letra cursiva fueron por parte del Entrevistador y las de letra negrita fueron por parte del grupo.

Si algún cambio se hace en una maquina solo en esa maquina se guardará ese cambio y tendrá que cambiarse en las otras maquinas... pero si es cliente servidor el actualizar maquinas sería de forma rápida

5. ¿Cuál es la disponibilidad que tiene el departamento para montar la infraestructura cliente servidor?

- ❖ *Se tienen ocho maquinas en red, y un servidor desde donde se controla el uso de las demás computadoras.*

6. De esta manera usted podría obtener información para saber que es lo que están haciendo sus estudiantes...

- ❖ *Me gustaría que cada estudiante tenga un usuario y así monitorear lo que ha hecho por usuario, saber que es lo que ha hecho y cuantas horas y ver el record, ya sea que el maestro se lo haya asignado o que el por cuenta propia haya realizado.*

7. ¿El docente le puede programar actividades al alumno dependiendo de lo que observe en clase?

- ❖ *Sería bueno dosificar la enseñanza al estudiante. El profesor puede indicar al estudiante que temas tiene que reforzar, además debe monitorear el tiempo que ha pasado el estudiante reforzando dichos temas*

8. ¿Se manejará la relación de docente, estudiante y grupo?

- ❖ *Los no videntes llegan en distintos horarios y pueden llegar a la hora que ellos quieran sin necesidad que este el docente presente, pero el docente siempre podrá ver que es lo que ha hecho.*

9. ¿El docente solamente puede ver lo que ha hecho un estudiante bajo su tutela y nada más?

- ❖ *Si, aunque al final cada profesor podría sacar la información de sus grupos y compararlo, o que por razones de análisis del departamento podría comparar los datos de todos los docentes.*

10. ¿Es factible esta relación de alumno y grupo en una enseñanza casi personalizada como ahora?

- ❖ *La asignación de los estudiantes cae de manera natural, el estudiante asiste al grupo a donde se a suscrito, puede ser que un docente atienda a un estudiante que no es de su clase por los horarios flexibles pero siempre será el docente responsable de ese estudiante quien debe monitorear que es lo que esta haciendo y como.*

11. Puede ser que varios estudiantes reprueben en una unidad en el software, ¿se podría modificar el contenido de esta unidad debido a que el contenido no sea el adecuado?

- ❖ *No sería la culpa del paquete, ya que el paquete no es toda la materia, es solamente una herramienta de asistencia pero no lo es todo ya que depende del docente principalmente que sus estudiantes aprendan en su clase, yo veo el software como una herramienta pedagógica auxiliar y complementaria no como el todo. Es como un complemento a la clase presencial.*

12. ¿Cuáles son los principales problemas que se presentan con los no videntes?

- ❖ *Pues son más lentos en su aprendizaje, se necesita que la letra sea muy grande en las evaluaciones, o dictarle las preguntas para que pudieran dar las respuestas oralmente.*

13. ¿Serían los ejercicios de selección múltiple?

- ❖ *Deberían haber más tipos.*

14. Si, también para dar respuestas a los ejercicios se pueden asignar teclas específicas para responder las distintas opciones u otras teclas sugeridas

- ❖ *También debería poder dictar lo que hay que hacer en el software, ítem por ítem si es evaluación.*

15. ¿Estos ejercicios los diseñaran los docentes?

- ❖ *Los ejercicios deberían ser creados por los docentes y el resultado de dichos ejercicios debería ser parte de un informe de notas*

16. ¿Los estudiantes no videntes pueden utilizar el teclado completamente?

- ❖ *Si pueden, lo único que hay que tomar en cuenta que existen personas con discapacidad visual pero que tienen cierto grado de visión. Y que estos usuarios pueden escuchar bien.*

17. ¿El docente tiene que estar presente cuando se realice una evaluación?... puede ser que no este presente

- ❖ *No necesariamente, pero por eso mismo se debería guardar el resultado para que el docente después lo pueda revisar.*

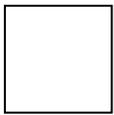
– El estudiante no puede pasar de una lección a otra si no ha aprobado la evaluación previa,

18. ¿Es obligatorio haber pasado una lección sin haber aprobado la evaluación correspondiente?

- ❖ *Si, así debe ser, aunque un estudiante podría realizar la evaluación de una lección directamente sin haber desarrollado los ejercicios de esa lección y si pasa el examen pasa de unidad.*

Anexo 2 – Entrevista con Armando Madrid, Estudiante no vidente de la UES

El video de esta entrevista puede ser visualizada desde el CD adjunto a este documento en la opción “Videos” y luego en la opción “Entrevista con Estudiante Invidente”.



Anexo 3 – Experiencias Estudiantes no videntes de la UES utilizando el SPEI3

Este video puede ser visualizado desde el CD adjunto a este documento en la opción “Videos” y luego en la opción “Experiencias”.