

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



**SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA
ZOOLOGÍA DE EL SALVADOR PARA EL MUSEO DE
HISTORIA NATURAL SABURO HIRAO**

PRESENTADO POR:

**OSCAR ALCIDES CASTILLO VILLACORTA
NELSON GUILLERMO CHICAS GARCÍA,
RICARDO ANTONIO ESCOBAR ORELLANA
KAREN BEATRIZ MEDRANO CALDERÓN**

PARA OPTAR AL TITULO DE:
INGENIERO DE SISTEMAS INFORMÁTICOS

CIUDAD UNIVERSITARIA, AGOSTO DE 2013

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZA VALETA DE AMAYA

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

DIRECTOR :

ING. JOSÉ MARÍA SÁNCHEZ CORNEJO

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

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO DE SISTEMAS INFORMÁTICOS

Título

:

**SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA
ZOOLOGÍA DE EL SALVADOR PARA EL MUSEO DE
HISTORIA NATURAL SABURO HIRAO**

Presentado por

:

**OSCAR ALCIDES CASTILLO VILLACORTA
NELSON GUILLERMO CHICAS GARCÍA,
RICARDO ANTONIO ESCOBAR ORELLANA
KAREN BEATRIZ MEDRANO CALDERÓN**

Trabajo de Graduación Aprobado por:

Docente Director

:

ING. RUDY WILFREDO CHICAS VILLEGAS

San Salvador, Agosto de 2013

Trabajo de Graduación Aprobado por:

Docente Director :

ING. RUDY WILFREDO CHICAS VILLEGAS

DEDICATORIA

Primero le agradezco a Dios por todas las oportunidades que me ha dado en la vida, por la vida misma por mi familia, agradezco a mi padre Oscar Castillo Rivera por darme la oportunidad de seguir estudiando, por sus consejos y motivación para seguir adelante y por no abandonarme nunca, a mi madre Maira Cecilia Villacorta por apoyarme y darme tranquilidad en momentos difíciles, a mis abuelos por la paciencia que me han tenido en este tiempo y por su incondicional apoyo, a mis hermanas por estar pendientes de mí y ser un gran apoyo y alegría para mí.

Le agradezco a Querubina Barillas por todas las veces que me ha ayudado, muchas veces en el momento más oportuno, por el apoyo que me ha dado y por motivarme a no rendirme, por escuchar mis problemas y por estar siempre conmigo en todo lo que hago, igualmente a su familia que en algún momento nos ha apoyado en nuestra etapa de trabajo de graduación.

Quiero agradecer a todos mis amigos que he conocido, a los que aun veo, a los que ya no, a los que ya no están, cada quien me ha dado un consejo muy importante en la vida, muchos fueron un ejemplo para mí, alguna vez me ayudaron en una situación difícil sin que pidiera su ayuda, a todos ellos se los debo, de todos aprendí algo bueno y sino fue en algo académico entonces fue para ser una mejor persona, a cada uno lo recordare por siempre.

Quiero agradecer a todos los catedráticos, por compartimos su conocimiento, su experiencia e incluso su amistad. Por darnos consejos de la vida los cuales ya no son parte de la enseñanza universitaria sino que es algo que surge de ellos y se les agradece mucho.

A nuestro asesor Ing. Rudy Chicas por ser un balance de un asesor estricto pero comprensible, por todos sus consejos para el trabajo de graduación, para la vida laboral y para la vida misma, así también a nuestro asesor observador, Lic. Mauricio Coto por aconsejarnos en la realización de nuestro trabajo de graduación y con su crítica ayudarnos a mejorar la manera de trabajar, a ambos asesores se les agradece su paciencia para con nosotros. A ambos se les agradece, se les respeta y los tenemos como un ejemplo a seguir.

Agradezco a mi grupo, Karen Calderon, Ricardo Orellana y Nelson Chicas por permitirme ser parte de su grupo, por la paciencia que me han tenido y por su amistad, haciendo de esta etapa del trabajo de graduación algo mucho más agradable de lo que plantean las personas que ya realizaron su trabajo de graduación, no me arrepiento para nada de haber hecho grupo con ellos y estoy muy agradecido de haberlos conocido.

Oscar Alcides Castillo Villacorta

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



**SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA
ZOOLOGÍA DE EL SALVADOR PARA EL MUSEO DE
HISTORIA NATURAL SABURO HIRAO**

PRESENTADO POR:

**OSCAR ALCIDES CASTILLO VILLACORTA
NELSON GUILLERMO CHICAS GARCÍA,
RICARDO ANTONIO ESCOBAR ORELLANA
KAREN BEATRIZ MEDRANO CALDERÓN**

PARA OPTAR AL TITULO DE:
INGENIERO DE SISTEMAS INFORMÁTICOS

CIUDAD UNIVERSITARIA, AGOSTO DE 2013

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZA VALETA DE AMAYA

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

DIRECTOR :

ING. JOSÉ MARÍA SÁNCHEZ CORNEJO

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

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO DE SISTEMAS INFORMÁTICOS

Título

:

**SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA
ZOOLOGÍA DE EL SALVADOR PARA EL MUSEO DE
HISTORIA NATURAL SABURO HIRAO**

Presentado por

:

**OSCAR ALCIDES CASTILLO VILLACORTA
NELSON GUILLERMO CHICAS GARCÍA,
RICARDO ANTONIO ESCOBAR ORELLANA
KAREN BEATRIZ MEDRANO CALDERÓN**

Trabajo de Graduación Aprobado por:

Docente Director

:

ING. RUDY WILFREDO CHICAS VILLEGAS

San Salvador, Agosto de 2013

Trabajo de Graduación Aprobado por:

Docente Director :

ING. RUDY WILFREDO CHICAS VILLEGAS

DEDICATORIA

Al concluir un trabajo tan arduo y lleno de dificultades como lo es el desarrollo de un trabajo de graduación, es inevitable darse cuenta inmediatamente que la magnitud de ese aporte hubiese sido imposible sin la participación de personas e instituciones que han facilitado las cosas para que este trabajo llegue a un feliz término. Por ello, es para mí un verdadero placer utilizar este espacio para ser justo y consecuente con ellas, expresándoles mis más sinceros agradecimientos.

Agradezco primero y ante todo a mi familia, quienes más que el apoyo que me han brindado en el transcurso de la carrera y desarrollo de este trabajo, me han dado algo más valioso que no se pueden aprender de ninguna otra forma más que en el seno familiar, las lecciones de la vida. Muchas gracias Isaura García, Oscar Chicas, Luis Chicas y Jacinto Quintanilla por el infinito apoyo que me han dado todo es tiempo.

Estoy muy feliz de poder cumplir de este gran desafío junto a mi equipo, que más que un equipo de trabajo hemos llegado a ser casi hermanos, porque los buenos amigos son, familia que se elige. Estaré eternamente agradecido de poder haber trabajado y aprendido mucho de ellos, gracias, muchas gracias a Karen Calderón, Ricardo Escobar y Oscar Castillo.

Estoy enormemente agradecido con nuestros docentes, y especialmente con nuestro asesor, ingeniero Rudy Chicas, por todo el conocimiento que nos brindó y la paciencia que mantuvo para guiarnos siempre por el camino de un profesional apegado a la máxima calidad. Igualmente al licenciado Mauricio Coto por haber aportado todos los detalles que pulieron nuestro trabajo y nuestra experiencia como profesionales.

Siempre hay personas que apoyan de una u otra forma y siempre están para ofrecer ayuda cuando se necesita, por estas personas que apoyaron todo este proceso mostrando nada más que una gran vínculo conmigo, agradezco a Karla, Luis, Douglas, Máximo, Emilio, Raúl, Gabriel, Adriana, Jeni, Carlos, Tere, Daniel, Verónica, Mirna y muchas personas más que me gustaría mencionar.

Además debo agradecer de manera especial a la Universidad de El Salvador y por extensión a la sociedad, ya que tuve el privilegio de obtener una beca que facilitó y mejoró de gran manera mi proceso de aprendizaje, y que sin duda aportó en gran medida el éxito que estoy cosechando.

Nelson Guillermo Chicas García

DEDICATORIA

Lo primero y más importante, quiero agradecer a mi madre, Graciela Orellana, por apoyarme en este proyecto que me tomo más tiempo del pensado, pero que al final se logro. Igualmente a toda mi familia, que aunque no nos llevamos bien, nos amamos muy a nuestra forma muy particular y peculiar

Agradezco a la familia Márquez Escolero; niña Betty, don Alfredo, Elizabeth, Isaac, Ariana, Beto, y en especial Romy, gracias por su apoyo y por hacerme parte de su núcleo familiar.

Un sinfin de gracias a Regina y su familia, quienes me han brindado todo su amor y paciencia, y me dieron posada en esos momentos de ajeteo de la universidad.

Gracias mi amigo y hermano del alma, Cesar Castellanos, al cual sacrifique mucho para lograr este proyecto, pero que lo quiero con todo mi corazón y sé que él siente lo mismo por mí a pesar de nuestras diferencias abismales.

Agradezco mucho a todos aquellos compañeros de la Facultad (en especial a los de la carrera) con los cuales hice alguna tarea, no los puedo mencionar a todos pero ustedes me ayudaron a lograr esto. También a todos aquellos maestros que muy a su forma me brindaron un poco de su experiencia. Además, a todas las personas que conozco, ya que de una u otra forma, han colaborado en este proyecto directa o indirectamente.

También de forma muy especial agradezco a José Barrera y Andrés González quienes fueron las primeras personas que conocí en la Universidad y me brindaron de forma incondicional su amistad. De igual forma a Emilio Velásquez con el cual compartimos muchas cosas buenas y malas. También a todas las personas que conocí en esta aventura, que de forma directa o indirecta, me apoyaron.

Agradezco la oportunidad de dar clases en la Facultad de Economía y en el Ministerio de Hacienda. Realmente me encanto esa experiencia única en la vida. Espero se vuelva a repetir.

Quiero agradecer a mis exnovias porque hicieron que me concentrara más y más en estudiar para poder olvidarlas. También de forma especial agradezco a todos aquellos que conformamos el ChoryClub, que sin mentir levantaban el ánimo cuando las cosas no salían como uno espera.

Por último, y no menos importante, agradezco la paciencia y compañía de Karen Medrano (la Jefa), Nelson Chicas (el Chino) y Oscar Castillo (Limonero SuperSilver), el mejor grupo con el cual me toco trabajar (que vivan los Hokages siempre) y a nuestro asesor, el ing, Rudy Chicas, quien demostró una paciencia sin igual, y nos brindo su apoyo y experiencia hasta el último momento.

Una solo persona es fuerte, pero es débil comparado al trabajo en equipo.

"Recuerden que el eslabón más alto que pude alcanzar la especie humana es ser revolucionario." – Che Guevara.

Ricardo Antonio Escobar Orellana

DEDICATORIA

Primeramente quiero agradecer a Dios que me permitió llegar tan lejos ya que sin la guía que puso en mi camino no habría podido terminar esta etapa de mi vida; como su palabra lo dice **“Nadie te podrá hacer frente en todos los días de tu vida; como estuve con Moisés, estaré contigo; no te dejaré, ni te desampararé” (Josué 1:5).**

A mi familia que siempre están apoyándome y animándome a seguir adelante a pesar de las adversidades, especialmente a mi madre por todos esos consejos y la paciencia que tiene conmigo cuando siento que ya no puedo más, a mi tía por su apoyo incondicional a pesar de todo y por último a mi hermano que a siempre está haciéndome reír para que logre lo que quiero, ya que sin ellos no había podido llegar al proceso que estoy culminando.

Gracias a mi grupo de trabajo Nelson Chicas, Oscar Castillo, Ricardo Escobar ya que ellos me dieron la oportunidad de formar parte de un excelente grupo con el que aprendí muchas cosas y con las que convive un millón de experiencias que nunca olvidare y que sin la sabiduría, paciencia y ánimos que los caracterizan no habríamos podido llegar hasta donde estamos, nunca olvidare todo lo vivido y por todas esas cosas es que estoy infinitamente agradecida de que Dios me los haya puesto en el camino.

Gracias al ingeniero Rudy Chicas por ser nuestro asesor y brindarnos todos sus consejos y experiencias vividas, guiarnos durante todo el proceso y sobre todo ayudarnos a entender el correcto camino a seguir para convertirnos en unas mejores personas y lograr así nuestro objetivo; a nuestro observador Lic. Mauricio Coto por la orientación brindada durante el proyecto así como brindarnos pautas para realizar un mejor trabajo.

Gracias a mis amigos los cuales estuvieron a lo largo de toda mi carrera a los que estuvieron al principio y final de ella a Patricia por ser una consejera para mí brindándome ánimos cuando ya no podía mas, a Isabel por estar pendiente siempre, a Máximo por ayudarme a mantener la serenidad durante el proceso, a Ángel por su ánimos ya al final cuando más se necesitan y a todos los que estuvieron pendientes Denis, Cristian, Josué, Mely, Karen, Carlos, Cesar, Eder, Ricardo, Oscar M, Adriana, Henry, Clavel, Paulino, Erick, Alejandro y a todos que siempre están conmigo muchas gracias.

Estas cosas os he hablado para que en mí tengáis paz. En el mundo tendréis aflicción; pero confiad, yo he vencido al mundo (Juan 16:33)

Karen Beatriz Medrano Calderón

Contenido	Página
I. Introducción	i
II. Objetivos	ii
II.1. Objetivo General	ii
II.2. Objetivos Específicos	ii
III. Alcances Y Limitaciones.....	iii
III.1. Alcances	iii
III.2. Limitaciones	iii
IV. Justificación.....	iv
Capítulo 1:.....	1
Modelado Del Negocio	1
1.1. Situación Actual	2
1.1.1. Estructura Organizativa.....	2
1.1.2. Descripción De La Situación Actual	3
1.2. Modelado De Proceso Actual.....	5
1.2.1. Nomenclatura De BPMN	5
1.2.2. Diagramas De Procesos Actual	6
1.3. Diagnóstico Del Problema.....	9
1.3.1. Lluvia De Ideas	9
1.3.2. Diagrama Causa-Efecto.....	10
1.3.3. Modelo De La Problemática.....	11
1.4. Estudio De Factibilidades.....	14
1.4.1. Factibilidad Técnica	14
1.4.2. Factibilidad Económica.....	19
1.4.3. Factibilidad Operativa	21
1.4.4. Resumen de factibilidades.....	26
1.5. Modelado Del Proceso Propuesto	27
Capítulo 2:.....	34

Requerimientos.....	34
2.1. Metodología Para La Determinación De Requerimientos.....	35
2.2. Requerimientos Informáticos	36
2.2.1. Requerimientos Funcionales	37
2.2.2. Requerimientos No Funcionales.....	42
2.3. Requerimientos Operativos	46
2.3.1. Requerimientos Medioambientales	46
2.3.2. Requerimientos De Mantenimiento.....	47
2.3.3. Requerimientos Operativos De Recurso Humano.....	48
2.3.4. Requerimientos De Seguridad.....	49
2.4. Requerimientos De Desarrollo	51
2.4.1. Recurso De Hardware De Desarrollo	51
2.4.2. Recurso De Software De Desarrollo	53
2.4.3. Recurso Humano De Desarrollo.....	55
2.5. Requerimientos de Implementación	59
2.5.1. Recurso De Hardware De Implementación.....	59
2.5.2. Recurso De Software De Implementación	61
2.5.3. Recurso Humano De Implementación.....	62
Capítulo 3:.....	63
Análisis Y Diseño.....	63
3.1. Arquitectura Del Sistema Informático.....	64
3.2. Casos De Uso	69
3.2.1. Actores.....	69
3.2.2. Lista De Casos De Uso.....	72
3.2.3. Diagramas De Casos De Uso	76
3.2.4. Descripción De Casos De Uso	82
3.3. Diagramas De Interacción	147
3.3.1. Nomenclatura Del Diagrama De Interacción	147
3.3.2. Diagrama De Interacción Del Sistema Informático	149
3.4. Diagrama De Clases	151
3.4.1. Nomenclatura Del Diagrama De Clases.....	151
3.4.2. Diagrama De Clases Del Sistema Informático	152
3.5. Diagrama De Componentes.....	159

3.5.1.	Nomenclatura Del Diagrama De Componentes	159
3.5.2.	Diagramas De Componentes Del Sistema Informático	160
3.6.	Diagramas De Despliegue	172
3.6.1.	Nomenclatura Del Diagrama De Despliegue	172
3.6.2.	Diagrama De Despliegue Del Sistema Informático	173
3.7.	Base De Datos	174
3.7.1.	Modelo Entidad-Relación.....	174
3.7.2.	Modelo Conceptual	177
3.7.3.	Modelo Lógico	179
3.7.4.	Modelo Físico.....	181
Capítulo 4:	183
Construcción.....	183
4.1.	Estándares De Base De Datos	184
4.1.1.	Nombre De La Base De Datos	184
4.1.2.	Sobre Lo Concerniente A Todos Los Objetos.....	184
4.1.3.	Tablas Y Vistas	185
4.1.4.	Campos.....	185
4.1.5.	Índices	186
4.1.6.	Tablas De Referencia	186
4.1.7.	Procedimientos Almacenados	186
4.1.8.	Disparadores.....	187
4.1.9.	Documentación De La Base De Datos	187
4.2.	Estándares De Programación.....	190
4.2.1.	Organización De Archivos	190
4.2.2.	Sangría.....	192
4.2.3.	Comentarios.....	193
4.2.4.	Declaraciones	194
4.2.5.	Sentencias	196
4.2.6.	Espacios En Blanco	197
4.2.7.	Nomenclatura De Identificadores.....	197
4.2.8.	Prácticas De Programación.....	199
4.2.9.	Documentación Javadoc.....	202
4.3.	Estándares De Interfaces De Usuario	205

4.3.1.	Plantilla.....	205
4.3.2.	Controles Comunes	208
4.3.3.	Nombrado De Archivos.....	208
4.3.4.	Estructura De Directorios	209
4.4.	Estándares De Documentación.....	211
4.4.1.	Estándar De Documentos Impresos.....	211
4.4.2.	Estándar De Reportes	214
Capítulo 5:	218
Pruebas	218
5.1.	Pruebas Técnicas	219
5.1.1.	Pruebas Unitarias.....	219
5.1.2.	Pruebas De Integración	220
5.1.3.	Pruebas De Sistema	221
5.2.	Pruebas De Funcionamiento.....	224
Capítulo 6:	248
Despliegue	248
6.1.	Plan De Implementación	249
6.1.1.	Plan De Implementación Del Nuevo Sistema	249
6.1.2.	Planeación	250
6.1.3.	Organización	260
6.1.4.	Dirección	264
6.1.5.	Control.....	265
6.2.	Plan De Capacitaciones	268
6.2.1.	Propósito.....	268
6.2.2.	Alcance.....	268
6.2.3.	Funciones Que Requieren Capacitación.....	268
6.2.4.	Planificación De La Capacitación	269
6.2.5.	Metodología De Capacitación	269
6.2.6.	Recursos	270
6.2.7.	Ejecución De La Capacitación Del Personal.....	271
6.2.8.	Presupuesto De Capacitación	273
7.	Conclusiones	274
8.	Recomendaciones.....	275

9. Bibliografía.....	276
Libros	276
Páginas Web.....	276
10. Glosario	278
Anexos.....	I
A. Planificación De Recursos.....	II
B. Elección De Metodología De Trabajo.....	VI
B.1. Modelo Del Ciclo De Vida.....	VI
B.2. Enfoque De Desarrollo.....	IX
B.3. Etapas Del Desarrollo Del Proyecto.....	X
C. Pruebas	XVIII
C.1. Pruebas Unitarias.....	XVIII
C.2. Pruebas De Integración	XXI
C.3. Pruebas De Sistema.....	XXX
D. Elección De Método De Conversión.....	XXXV

I. INTRODUCCIÓN

El desarrollo de la sociedad se impulsa a través de los avances científicos y culturales. Estos avances se nutren de los conocimientos y experiencias previamente adquiridos. En este sentido, los museos juegan un papel importante como bancos de conocimiento que sirven como insumo para investigaciones que enriquezcan el conocimiento científico y su aplicación. El Museo de Historia Natural de El Salvador, adscrito a la Secretaría de Cultura de la Presidencia de la República, es uno de los museos más importantes del país y el único especializado en el área de zoología, posee una colección de alrededor de 7,600 especímenes de fauna que ofrecen una valiosa fuente de información para investigaciones universitarias, científicas y ambientales.

Actualmente y desde sus orígenes el área de zoología del museo mantiene los datos de los especímenes de forma manual, lo que genera dificultades para realizar los procesos del área de zoología de manera eficiente, sobre todo cuando se desea recuperar información, ya que los datos están escasamente organizados, lo que dificulta las consultas. Esto ha generado la necesidad de desarrollar un sistema que mecanice la clasificación de la zoología del país y que permita gestionar la información del museo fácil y eficientemente.

En consecuencia a esta situación, este documento describe el desarrollo del Sistema Informático para la Clasificación de la Fauna de El Salvador, dividido en seis capítulos, siguiendo un orden secuencial de su contenido.

Este documento comprende el modelado de la situación actual, la definición del problema, el estudio de factibilidades y modelado propuesto del sistema informático. Se muestran las necesidades que tiene que cubrir el sistema informático, y los recursos que fueron utilizados para su desarrollo. Además, se presenta el diseño del sistema de información, en donde se describe lo que se debe hacer para lograr lo establecido en los requerimientos informáticos. También incluye los estándares de programación, de diseño de base de datos, de interfaces de usuario y de documentación, así como las pruebas realizadas al sistema informático, las cuales garantizan su funcionamiento. Finalmente, se presenta el plan de implementación, donde se indica el proceso a seguir para poner en funcionamiento el sistema informático.

Este sistema informático es de gran importancia para el Museo de Historia Natural de El Salvador y por tanto, para el desarrollo científico y cultural del país, ya que proveerá las herramientas necesarias para mejorar el desempeño en las actividades del museo y mejorando enormemente los medios de consulta de información y a su vez expandiendo la cobertura de acceso a la información.

II. OBJETIVOS

II.1. OBJETIVO GENERAL

Desarrollar un sistema informático que permita mejorar el proceso de registro y clasificación de la colección de zoología del Museo de Historia Natural de El Salvador, que permita realizar consultas de la información con facilidad por los usuarios del área de zoología.

II.2. OBJETIVOS ESPECÍFICOS

- Analizar la situación actual con respecto a los procesos realizados en el área de zoología del Museo de Historia Natural de El Salvador.
- Identificar las necesidades existentes en el área de zoología como requerimientos.
- Definir los estándares de trabajo para el desarrollo del sistema informático.
- Diseñar las interfaces, modelo y base de datos del sistema informático en base a los requerimientos.
- Construir los subsistemas que conformarán al sistema informático en base al diseño.
- Realizar pruebas que garanticen el funcionamiento y calidad del sistema informático.
- Elaborar la documentación necesaria para el mantenimiento y usabilidad del sistema informático.
- Elaborar el plan de implementación del sistema informático indicando los recursos, procedimientos y condiciones que deben existir para su puesta en marcha.

III. ALCANCES Y LIMITACIONES

III.1. ALCANCES

Los alcances que se esperan cumplir en el desarrollo del proyecto son los siguientes:

- Un sistema informático funcional, validado y aprobado por el MUHNES que permita registrar, clasificar, modificar y borrar especímenes de la colección de zoología, generar reportes personalizados para uso interno y presentar la información de la fauna salvadoreña en ambiente Web a los interesados.
- La documentación necesaria para el funcionamiento y mantenimiento del sistema informático desarrollado, la cual comprende manual técnico y manual de usuario.
- El plan de implementación para poner en funcionamiento el sistema informático desarrollado.

III.2. LIMITACIONES

En vista de que las necesidades a cubrir son alcanzables y no existen impedimentos para satisfacer lo solicitado por la organización, se concluye que no se encuentran limitantes para el desarrollo del proyecto.

IV. JUSTIFICACIÓN

La accesibilidad y facilidad de consulta de información son aspectos que posibilitan y acrecientan la realización de estudios científicos según las autoridades encargadas MUHNES.

El MUHNES recibe visitas de 260 personas anuales, entre las cuales tenemos estudiantes universitarios, investigadores nacionales e internacionales y periodistas. Estas personas realizan investigaciones de gran valor científico y ambiental, que están relacionados primordialmente con estudios de impacto ambiental, conservación de especies y estudios de grado superior, de los cuales un grupo de estos hacen su trabajo en el área de zoología. Sin embargo, el museo no posee los recursos necesarios para poner a disposición la información de la colección de zoología al público en general y a los investigadores en específico de forma eficiente, lo que impide realizar estudios de manera práctica y efectiva.

Todo patrimonio cultural está protegido por las leyes del estado. Dentro del museo, en el área de zoología, el patrimonio cultural está conformado por los especímenes, los libros de registros escritos a mano y las fichas de información. Cada vez que un investigador llega a consultar información al museo, los libros de registro son expuestos a ser dañados por su excesivo uso, acelerando su deterioro ya que no existe otra fuente de información sobre los animales clasificados. Además, la búsqueda en los libros es lenta ya que esta información no está ordenada por ningún tipo de parámetro, así también se expone al deterioro al espécimen disecado ya que luego de encontrarlo en el libro, se saca del lugar de donde se almacena en bodega, para verlo e inspeccionarlo físicamente.

Actualmente la información está limitada solo para uso interno y para investigadores, por lo que gran parte de la información no es presentada al público debido a que al estar consultando los libros de registro para cada persona que quisiera buscar información, se deja de realizar el principal trabajo de los encargados del área el cual es el registro y clasificación de las especies, además de que estos libros se exponen a daños cada vez que se consultan, por lo que la única información que se le presenta al público es la de los animales que están en la sala de exposiciones del museo.

En respuesta ante las necesidades del museo, se hace indispensable contar con un sistema informático que permita registrar los datos y muestras de especies de fauna, facilitando la clasificación y el manejo seguro de los datos, y brindando la capacidad de proporcionar información inmediata de uso interno para el museo que agilice los procesos de registro de especies animales y generación de reportes. Además, el sistema informático proveerá una interfaz amigable e intuitiva de consulta de datos, que eliminará las restricciones geográficas y de nivel académico, ampliando y facilitando la capacidad de consulta, permitiendo realizar más investigaciones al público, estimando alrededor de más de 60,000 personas anualmente. Esto supondría una mayor disponibilidad de la información para los visitantes al museo y un importante ahorro de esfuerzo para investigadores, periodistas, estudiantes universitarios y al museo, dando oportunidad a una mejor y mayor investigación en el área zoológica. Por último, se resguardará la información de los especímenes animales para asegurar que esta nunca se pierda y se conserve este patrimonio invaluable.

CAPÍTULO 1:

MODELADO DEL NEGOCIO

El modelado de negocios es la interpretación de la realidad y flujo de trabajo de la organización en estudio, que tiene como resultado un modelo unificado que sirve como marco de referencia común para usuarios y desarrolladores, logrando un mejor entendimiento de la organización para fines del desarrollo del sistema informático, y un máximo nivel de acoplamiento fundamentado en la identidad de la organización.

1.1. SITUACIÓN ACTUAL

La situación actual presenta la estructura organizativa del MUHNES y describe los procesos actuales de clasificación, registro y consulta de la colección de zoología.

1.1.1. ESTRUCTURA ORGANIZATIVA

El Museo de Historia Natural de El Salvador (MUHNES) es una dependencia de la Secretaría de Cultura de la Presidencia de El Salvador. El museo está coordinado por la Dirección Nacional de Patrimonio Cultural. La Gerencia de Informática y Sistemas es la encargada de brindar los servicios informáticos a cada una de las dependencias de la Secretaria de Cultura, por lo que depende jerárquicamente de la Dirección de Administración.

Actualmente se están llevando cambios organizacionales y administrativos dentro de la Secretaría de Cultura. Sin embargo, es posible describir la posición jerárquica tanto del MUHNES como de la Gerencia Informática se mantiene, como se muestra en la figura 1.1.

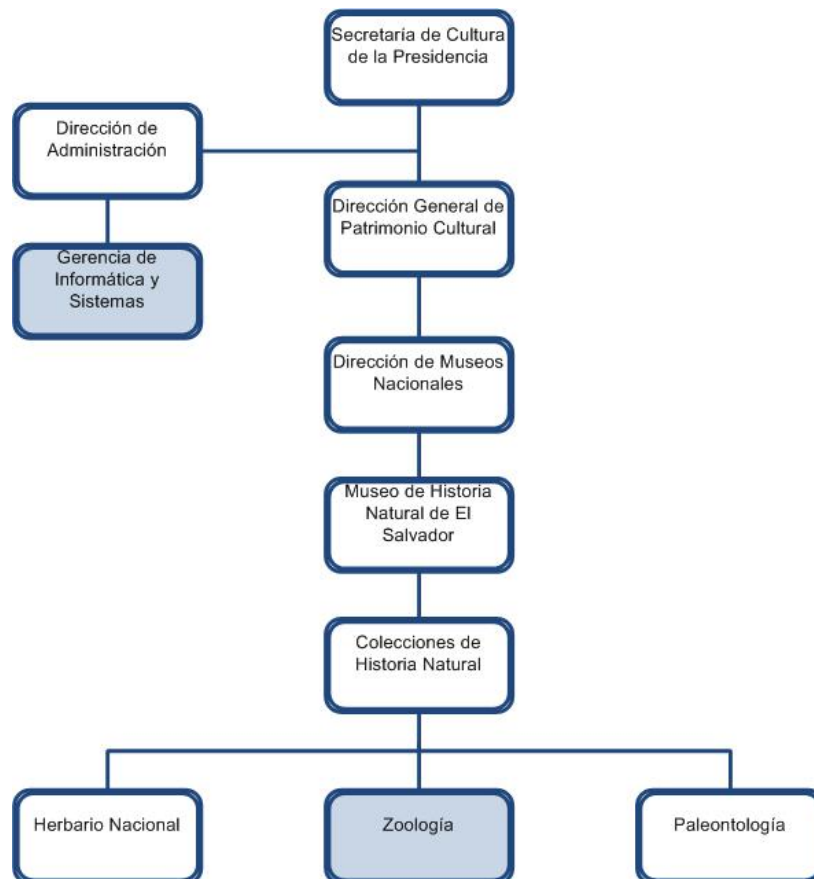


Figura 1.1. Estructura organizativa resumida.

La colección de especímenes del área de zoología (unidad de trabajo del presente proyecto), está dividida de la siguiente manera:

- **Vertebrados:** comprende los animales que poseen una espina dorsal o columna vertebral formada por vertebras. Esta clasificación se encuentra dividida en:
 - Herpetología (reptiles, serpientes y anfibios).
 - Mastozoología (mamíferos).
 - Ictiología (peces).
 - Ornitología (aves).
- **Invertebrados:** son todos aquellos animales que no son vertebrados. Esta clasificación incluye:
 - Entomología (insectos).
 - Malacología (moluscos).

1.1.2. DESCRIPCIÓN DE LA SITUACIÓN ACTUAL

Actualmente el MUHNES no cuenta con un sistema automatizado para apoyar la clasificación y registro de las especies que están siendo recolectadas y estudiadas, por lo que estos quedan clasificados y registrados en libros de registro que con el tiempo se han ido deteriorando con rapidez debido a que los libros están hechos de material orgánico, además del constante uso que tienen por parte de las personas que registran y clasifican las especies animales.

En los últimos años el número de personas encargadas del área de zoología ha ido disminuyendo de forma tal, que en la actualidad la persona encargada de estudiar moluscos también tiene que encargarse de estudiar los mamíferos, insectos, reptiles, arácnidos y demás, que no son su fuerte, por lo que tiene dificultades al momento del estudio de los demás especímenes.

Al momento de registrar un espécimen, esta se registra en el siguiente espacio disponible en el cuaderno de registro, lo que es un problema a la hora de buscar una especie solicitada por alguna institución o por personas que están haciendo investigación universitaria o científica, ya que esto representa una búsqueda secuencial, página por página, para encontrar una especie específica a pesar de que estas están clasificadas por un código que identifica el tipo de especie que se registra además de un correlativo para intentar mantener un orden.

Todos los especímenes se encuentran alojados en una bodega a una temperatura controlada, siendo su ubicación registrada dentro de un archivo de Microsoft Excel para facilitar la búsqueda del espécimen. Aun así, no todas las ubicaciones de los especímenes se encuentran registradas, y a veces se cometen pequeños errores o confusiones entre la ubicación y código del espécimen que no son correctos, por lo que es necesario que la persona encargada del área de zoología realice una búsqueda exhaustiva hasta encontrar el espécimen de interés.

La cantidad de tiempo para realizar las tareas cotidianas del área de zoología es demasiado cuando se le suma el tiempo de las complicaciones antes mencionadas y más aún cuando la cantidad de especies que están llegando a ser estudiadas supera en más de 20 veces al número de especies que son registradas y clasificadas por los encargados de zoología, más cuando es una sola persona encargada de estudiar todas las especies de tipo moluscos y las que no le competen.

La encargada del área de zoología tiene que elaborar dos reportes mensuales. El principal destino de estos reportes es auditoría interna del museo sobre restauraciones de especímenes y reportes de especímenes registrados en el mes. El principal problema para estos reportes es buscar los códigos de las especies a las que se les ha dado mantenimiento en los libros de registro ya que las especies solamente tienen una ficha de campo, la cual sirve para encontrar el código en el libro de registro. El tiempo para realizar un reporte es bastante debido a que no siempre se encuentra el código del espécimen animal rápidamente como consecuencia de la falta de orden.

La información de los especímenes está siendo compartida solamente con investigadores universitarios nacionales e internacionales, por lo que estudiantes de nivel inferior a la educación superior quedan excluidos debido a la cantidad de personas que conforman este nivel de estudio además del tiempo que se necesitaría para atender a todos estos estudiantes, incumpliendo de manera no intencional la misión del museo, siendo a su vez influenciado por las leyes que protegen el patrimonio cultural ya que estas especies son patrimonio cultural de la nación y al exponerlas al público están poniéndolas en riesgo. Así también las personas encargadas del museo están en riesgos en ser castigados económicamente o incluso con la cárcel por permitir daños al patrimonio.

Las computadoras que se utilizan en el museo provienen de una donación del gobierno de Costa Rica para facilitar el trabajo de las tareas del museo, siendo el principal propósito proporcionar un medio para compartir la información de toda de la zoología con el resto de museos de historia natural de Centroamérica. Esta tarea es muy difícil para el MUHNES ya que toda la información está escrita en libros de papel y las maquinas donadas solo han servido para poder tener cierta información reservada para uso interno.

Las personas encargadas del área de zoología atienden a los investigadores y estudiantes universitarios, presentando el inconveniente del tiempo para buscar información a estas personas ya que al hacer esto están usando su tiempo laboral y se atrasan con su trabajo.

1.2. MODELADO DE PROCESO ACTUAL

Para facilitar la comprensión de la forma en que se realizan los procesos actuales dentro del área de zoología, tanto para el equipo de trabajo como para los usuarios finales, se hará uso de la Notación para el Modelado de Procesos de Negocio (BPMN por sus siglas en inglés Business Process Modeling and Notation), que es una notación gráfica estandarizada que permite el modelado de procesos de negocio en un formato de flujo de trabajo.

1.2.1. NOMENCLATURA DE BPMN

La simbología utilizada por BPMN se muestra en la tabla 1.1.









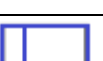


Símbolo	Nombre	Descripción
	Inicio	Representa el punto de inicio de un proceso.
	Fin	Indica cuando un proceso termina.
	Decisión exclusiva	Decisión basada en datos del sistema.
	Decisión inclusiva	Uno o más caminos pueden ser activados.
	Tarea	Actividad simple que se utiliza cuando el trabajo realizado dentro del proceso no está definido en un nivel más detallado.
	Sub-proceso	Es una actividad compuesta cuyo detalle se define como un flujo de otras actividades.
	Tarea de usuario	Tarea donde un humano realiza una tarea que tiene que ser completada en cierta cantidad de tiempo.
	Tarea de servicio	Tarea que usa algún tipo de servicio, que podría ser un servicio Web o una aplicación automática.
	Contenedor	Representa el contenedor de un proceso.
	Sub-contenedor	Es una subdivisión del contenedor y representa los diferentes participantes al interior de una organización.
	Fase	Representa segmentos de un proceso.

Tabla 1.1. Notación utilizada por BPMN.

1.2.2. DIAGRAMAS DE PROCESOS ACTUAL

Dentro del área de zoología se llevan a cabo 4 procesos realizados por la misma persona encargada de esta área:

1. **Clasificación de especímenes:** Este proceso trata de identificar el tipo de especie de aquellas muestras que llegan al área de zoología y que no pudieron ser clasificadas previamente. Los pasos a seguir son:
 - A. Observar las características taxonómicas del espécimen (recurriendo a la disección en algunos casos).
 - B. Consultar las características taxonómicas en libros, sitios web especializados e investigadores.
 - C. Comparar los resultados con la información de referencia.
 - D. Definir el tipo de espécimen.

Este proceso se muestra en la figura 1.2.

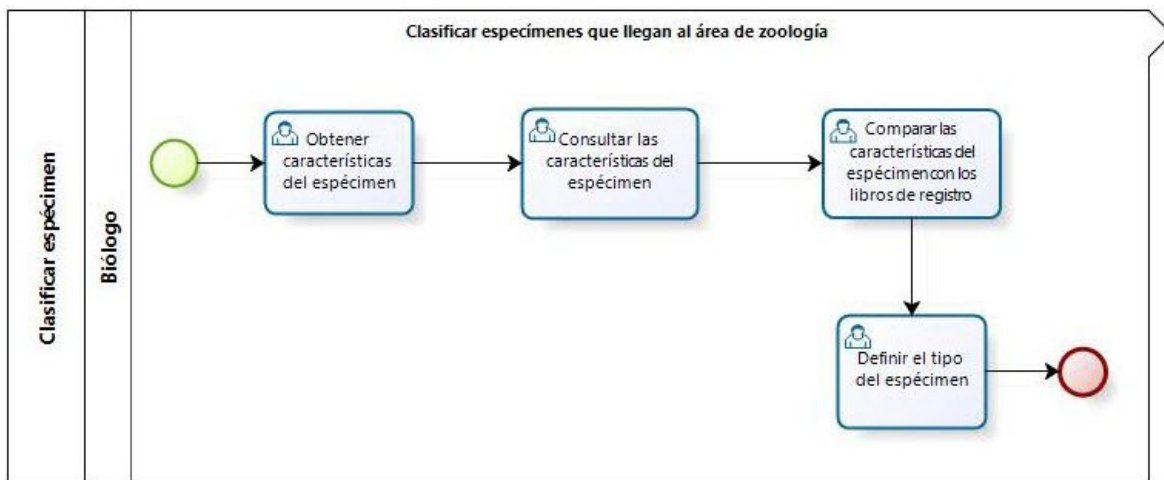


Figura 1.2. Proceso actual de clasificación de especímenes.

2. **Registro de especímenes:** Los especímenes pasan a ser registrados si estos se encuentran clasificados previamente. El proceso es descrito a continuación:
 - A. Recibir los especímenes previamente clasificados.
 - B. Consultar la ficha de campo de cada uno de los especímenes.
 - C. Registrar los especímenes en los libros de registro.
 - D. Almacenar cada espécimen en su lugar correspondiente.

La figura 1.3 muestra este proceso.

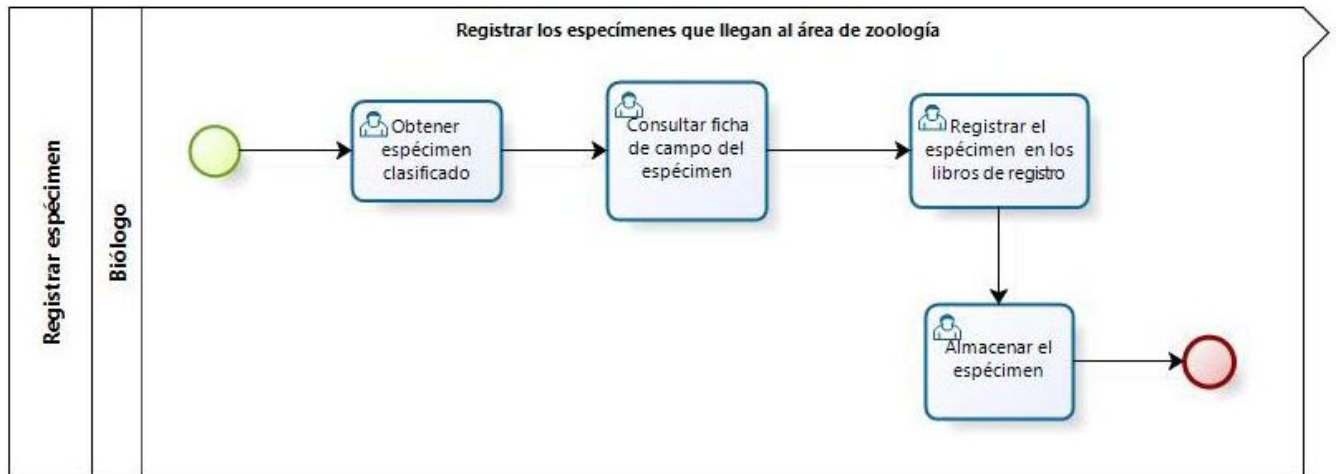


Figura 1.3. Proceso actual de registro de especímenes.

3. **Consulta de información:** La información es consultada desde los libros de registro de la colección de zoología, donde se buscan los campos de interés. A continuación se describe el proceso:

- A. Preparar los libros de registro a consultar.
- B. Consultar uno o más campos de interés.

Este proceso se muestra en la figura 1.4.

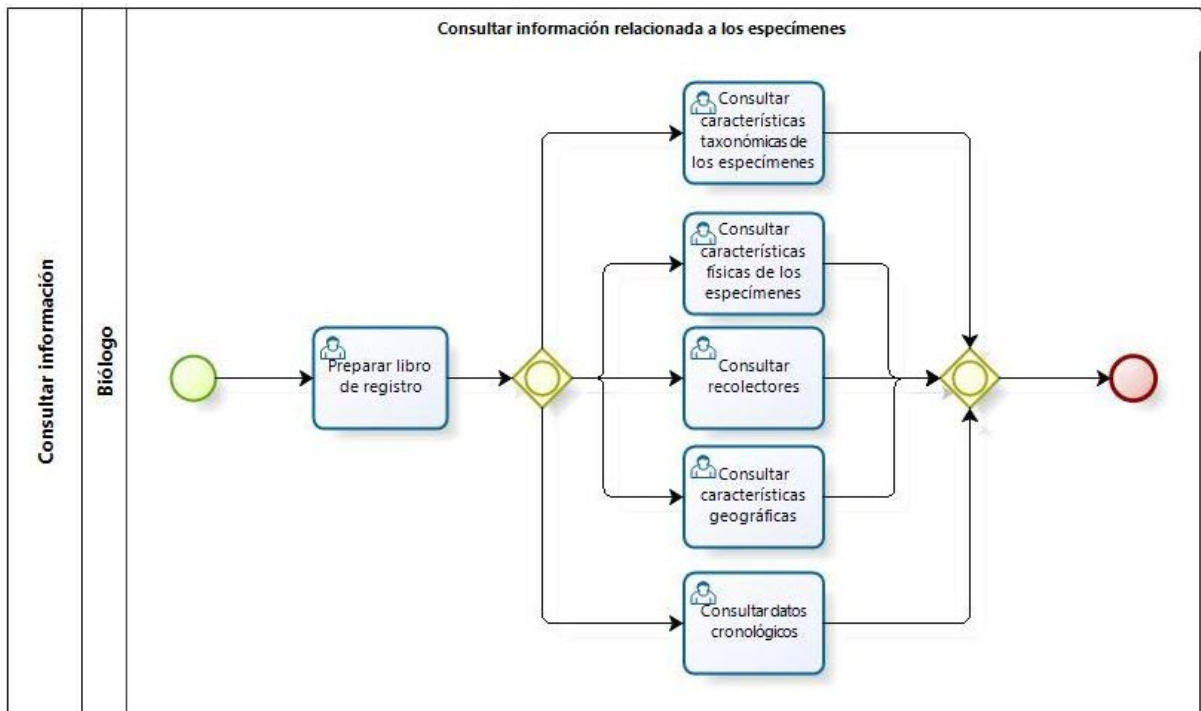


Figura 1.4. Proceso actual de consulta de información.

4. **Generación de reportes:** la información para la generación de reportes es extraída de los libros de registro de la colección de zoología. Los pasos a seguir son:
- A. Preparar los libros de registro.
 - B. Realizar el proceso de consulta de información.
 - C. Escribir o digitar la información de interés.
 - D. Enviar el reporte al solicitante.

Gráficamente se muestra este proceso en la figura 1.5.



Figura 1.5. Proceso actual de generación de reportes.

1.3. DIAGNÓSTICO DEL PROBLEMA

El área de zoología del MUHNES presenta actualmente dificultades en cuanto a la realización eficiente de los procesos que lleva a cabo. Para lograr identificarlos, se hará uso de herramientas de diagnóstico de problemas, las cuales incluye lluvia de ideas y diagrama de causa-efecto.

1.3.1. LLUVIA DE IDEAS

Es una herramienta de trabajo de grupo que facilita el surgimiento de nuevas ideas sobre un tema o problema determinado para presentarlas de forma ordenada. La principal regla es no juzgar; en un principio toda idea es válida ninguna debe ser rechazada. En una lluvia de ideas se busca la cantidad antes que la calidad y se valora la originalidad.

Partiendo de un bosquejo general, se plantea la situación de la problemática mediante un diagrama de lluvia de ideas representado en la figura 1.6.

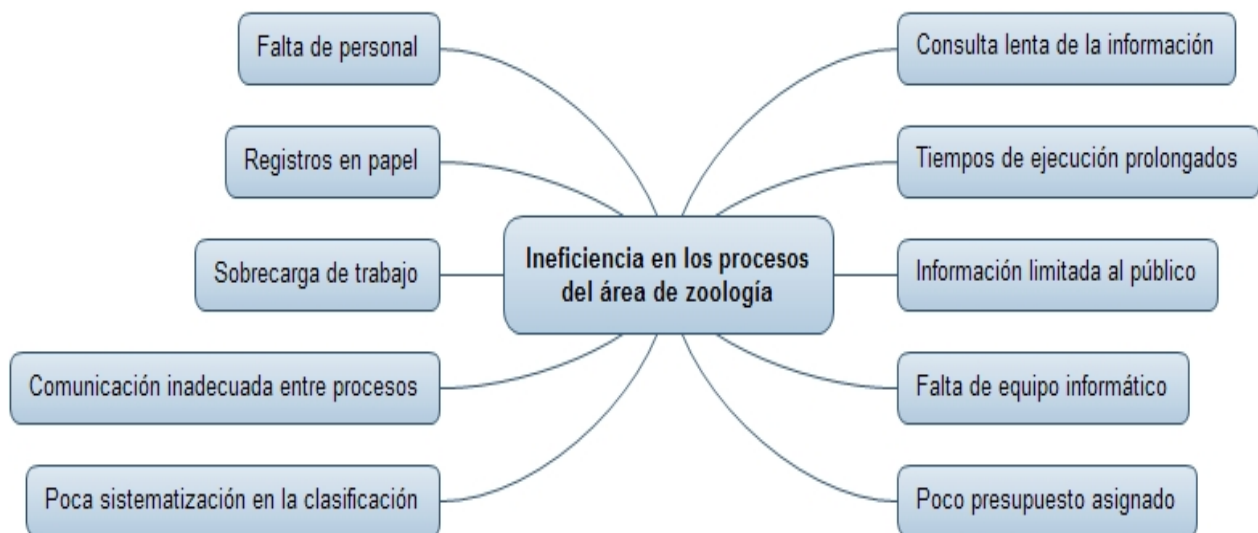


Figura 1.6. Lluvia de ideas.

Al analizar las ideas propuestas, se tiene que:

- El registro de los especímenes para la colección de zoología se hace de forma manual.
- La búsqueda de información de la colección de zoología toma demasiado tiempo.
- El personal con que cuenta el área de zoología es insuficiente para poder realizar las tareas diarias.

- El personal del área de zoología tiene sobrecarga de trabajo.
- Los reportes que genera el área de zoología se hacen de forma manuscrita.
- El museo no cuenta con equipo informático adecuado para ayudar el personal a desarrollar los procesos.
- La información de la colección del área de zoología no se difunde por los canales adecuados al público en general.
- Los recursos asignados al área de zoología son insuficientes para realizar los procesos.

Por último, se realiza una síntesis de las ideas anteriormente analizadas, obteniendo los siguientes resultados:

- El personal con que cuenta el área de zoología es insuficiente, lo que conlleva a una sobrecarga de trabajo y provocando retrasos en las actividades cuando se tiene que atender a los usuarios.
- El registro de especímenes y generación de reportes se lleva manualmente, por ende la búsqueda de la información requiere demasiado tiempo.
- La información de la colección de zoología solo está disponible para el personal autorizado, investigadores, reporteros y estudiantes universitarios de biología y carreras afines.
- La asignación insuficiente de recursos y el poco equipo informático con que se cuenta, imposibilita la realización de procesos más eficientes haciendo uso de este.

1.3.2. DIAGRAMA CAUSA-EFECTO

Este diagrama ofrece de una manera gráfica y fácil de comprender, un panorama completo de la problemática en estudio para facilitar la identificación de las posibles causas del problema detectado, y así aportar ideas para la recolección de datos y/o soluciones del mismo.

Partiendo de las ideas aportadas en la lluvia de ideas, se puede visualizar más fácil y organizadamente los elementos que tienen influencia en el problema a través del diagrama causa-efecto mostrado en la figura 1.7.

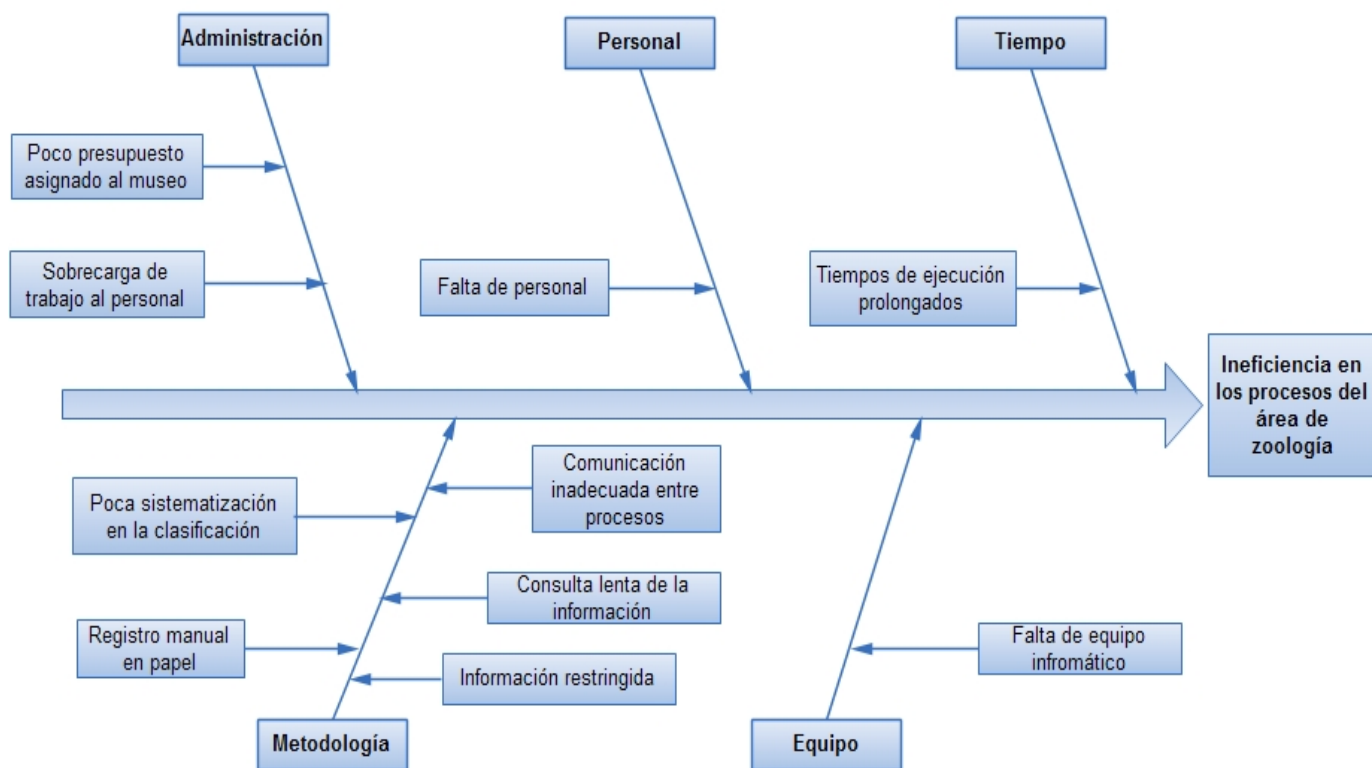


Figura 1.7. Diagrama causa-efecto.

1.3.3. MODELO DE LA PROBLEMÁTICA

Al analizar las ideas obtenidas y los elementos vinculados al problema se puede obtener de manera organizada la problemática actual del área de zoología del MUHNES, el cual es representado por el modelo mostrado en la figura 1.8.

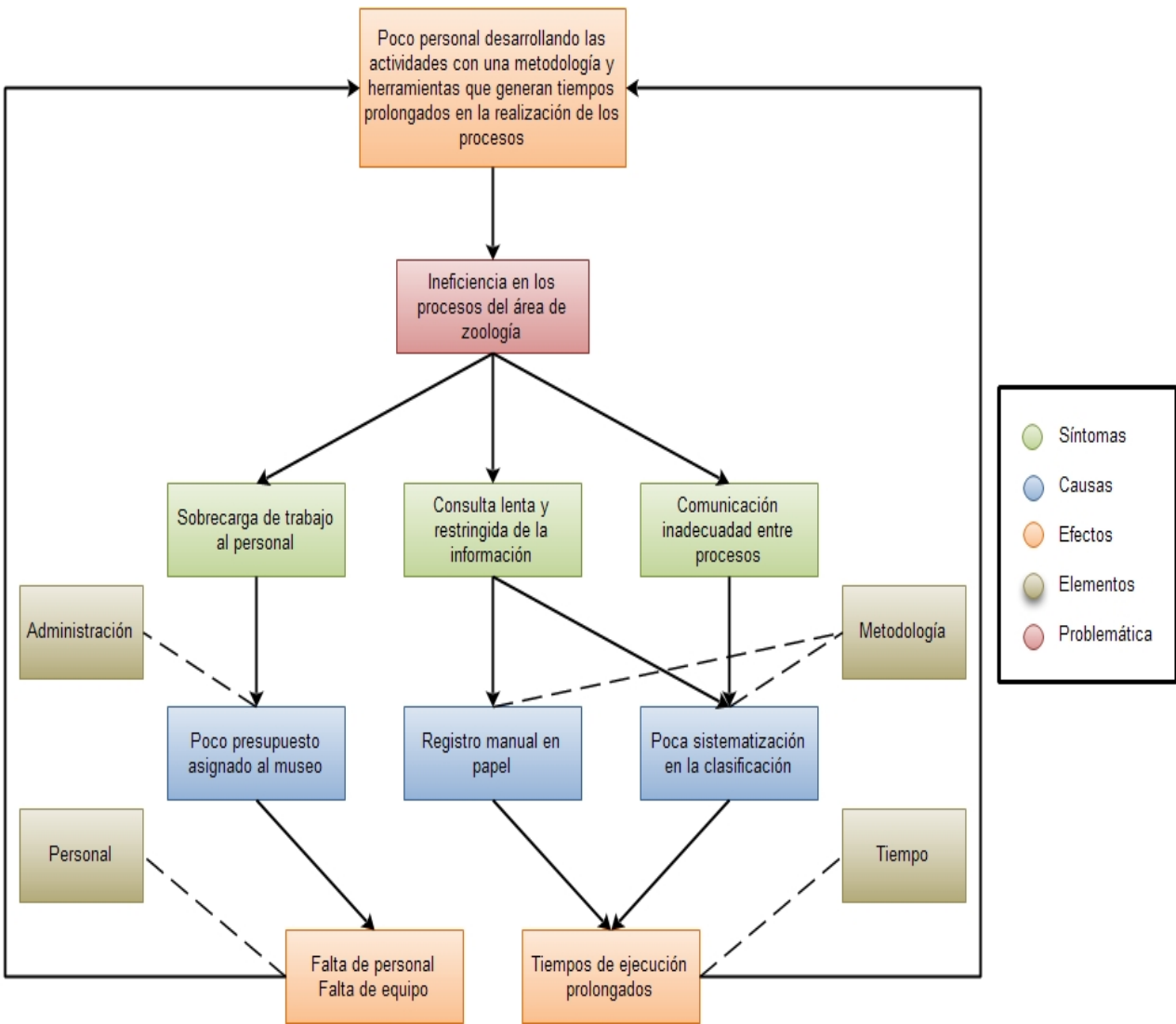


Figura 1.8. Modelo de la problemática.

Los elementos que conforman el modelo son los siguientes:

- **Síntomas:** Se puede visualizar con facilidad los eventos que ponen de manifiesto la existencia de un problema en el sistema actual, estos son, sobrecarga de trabajo al personal, consulta lenta de la información, información disponible solo para uso interno y comunicación inadecuada entre procesos.
- **Causas:** Estas son el origen de la situación que enfrenta actualmente la institución. Está es generada por el poco presupuesto asignado al museo y por consiguiente al área de zoología, el registro manual en papel de los datos de los especímenes de fauna y la poca sistematización de la clasificación de los especímenes.

- **Efectos:** Son los resultados que tiene que asumir el MUHNES debido a la problemática. Estos son: falta de equipo y personal, y tiempos prolongados en la ejecución de los procesos. El efecto global en el museo es una reducida cantidad de personas desarrollando las actividades con equipo y metodología inadecuados para desempeñar correctamente los procesos del área de zoología.
- **Elementos:** Estos son las entidades vinculadas a la situación de la institución. Se pueden dividir en entidades problemáticas (administración y metodología) y entidades afectadas (personal y tiempo).
- **Problemática:** En conclusión al análisis anterior, los tiempos prolongados en la realización de las actividades y el uso de metodologías y herramientas inadecuadas sugieren que la problemática del área de zoología está relacionada con la ineficiencia en la realización de los procesos.

1.4. ESTUDIO DE FACTIBILIDADES

Luego de haber definido el problema, es esencial realizar un estudio de factibilidad para estimar si el proyecto a desarrollar será o no factible desde los 3 puntos de vista: técnico, económico y operativo.

A continuación se presentan el estudio realizado para las factibilidades técnica, económica y operativa del desarrollo del sistema informático propuesto.

1.4.1. FACTIBILIDAD TÉCNICA

La factibilidad técnica consiste en una evaluación de la tecnología existente en la organización, con este estudio se obtendrá información de los recursos técnicos que posee la Secretaría de Cultura y el equipo de trabajo para el desarrollo del sistema informático. Estos son:

- **REQUERIMIENTOS DE SOFTWARE:**

- **Sistema operativo:** La unidad técnica especificó que el sistema operativo que utilizan en la producción de sus sistemas informáticos es Debian, por tanto el sistema operativo a ser usado para el desarrollo y producción será este mismo.
- **Lenguaje de programación:** La unidad técnica detalló que poseían sistemas informáticos basados en Java y PHP funcionando en su entorno de producción, y que por consiguiente, estas serían las opciones como lenguaje de programación que se tienen para el desarrollo.

Basándose en este dominio, la elección se realizará en base a criterios de evaluación que se calificarán: baja, media y alta, que tendrán un valor de 1, 2 y 3 respectivamente. A partir la calificación de ambos lenguajes se determinará cuál será es el lenguaje más adecuado para el desarrollo del proyecto. Esto se muestra en la tabla 1.2.

Criterio	Java	PHP
Documentación	Alta	Alta
Experiencias previas	Media	Media
Facilidad de aprendizaje	Media	Alta
Apego a la técnica de diseño	Alta	Media
Herramientas para desarrollo	Alta	Media
Frameworks y bibliotecas	Alta	Alta

Tabla 1.2. Comparación de lenguajes de programación.

El resultado de las calificaciones para los lenguajes es de 16 y 15 para Java y PHP respectivamente. A pesar que la diferencia es bastante baja, se considera que para el desarrollo de este proyecto Java es el lenguaje más adecuado, ya que por ser un lenguaje fuertemente tipado y con más herramientas para desarrollo facilitará el desarrollo del sistema informático.

- **Sistema gestor de base de datos:** De manera similar a los lenguajes de programación, la unidad técnica tiene sistemas informáticos funcionando haciendo uso de MySQL y PostgreSQL. Por tanto, estas son las opciones de elección de sistema gestor de base de datos. La comparación de ambos gestores se muestra en la tabla 1.3.

Criterio	MySQL	PostgreSQL
Documentación	Media	Media
Experiencias previas	Media	Media
Herramientas para desarrollo	Media	Media

Tabla 1.3. Comparación de sistemas gestores de bases de datos.

Dado que las ponderaciones para ambos gestores de base de datos son iguales, es decir, ambos tienen las mismas características para el desarrollo del sistema informático, se optará por utilizar PostgreSQL, ya que la unidad técnica afirmó que consideran este sistema gestor de base de datos como el más adecuado para sus proyectos.

- **Otro software:**
 - **Netbeans:** Entorno de desarrollo integrado, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo.
 - **Glassfish:** Servidor de aplicaciones que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación.
 - **PrimeFaces:** Componente para JavaServer Faces (JSF) que cuenta con un conjunto de componentes ricos que facilitan la creación de las aplicaciones web.

- **Hibernate:** Herramienta de Mapeo objeto-relacional (ORM) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.
- **Mozilla Firefox:** Navegador web

En resumen el software a utilizar para el desarrollo e implementación del sistema informático se presenta en la tabla 1.4.

Software	Uso	Categoría	Versión
Debian	Desarrollo/Producción	Sistema operativo	6 ¹
JRE	Desarrollo/Producción	Plataforma	7.4
JDK	Desarrollo	Plataforma	7.4
Java EE	Desarrollo	Plataforma	6.4
Netbeans	Desarrollo	Entorno de desarrollo	7.1.1
Glassfish	Desarrollo/Producción	Servidor	3.1.2 ¹
PostgreSQL	Desarrollo/Producción	Base de datos	9.4 ¹
PrimeFaces	Desarrollo/Producción	Herramienta	4.2.1
Hibernate	Desarrollo/Producción	Herramienta	4.1.2
Mozilla Firefox	Desarrollo	Navegador	12

Tabla 1.4. Software de desarrollo.

Los requerimientos para el software a utilizar se presentan en la tabla 1.5:

Software	Procesador	Memoria RAM	Espacio en disco
Debian	1 GHz	512 MB	5 GB
JRE	512 MHz	512 MB	128 MB
JDK	512 MHz	512 MB	256 MB
JavaEE	512 MHz	1 GB	512 MB
Netbeans	2.6 GHz	2 GB	1 GB
Glassfish	1 GHz	1 GB	512 MB
PostgreSQL	512 GHz	512 MB	128 MB
PrimeFaces	N/A	N/A	N/A
Hibernate	N/A	N/A	N/A
Mozilla Firefox	1 GHz	512 MB	256 MB

Tabla 1.5. Requerimientos de software².

La unidad técnica posee sistemas informáticos usando casi todas las tecnologías a usar en el desarrollo del sistema informático, a excepción de PrimeFaces e Hibernate que son componentes de fácil implementación, y por tanto no representa problemas su uso.

¹ Versión utilizada actualmente por la unidad técnica

² Los datos para procesador y memoria RAM son los valores recomendados ya incluyendo el uso de las plataformas subyacentes.

El equipo de trabajo puede tener acceso a estas herramientas informáticas ya que éstas están disponibles para su libre uso.

- **REQUERIMIENTOS DE HARDWARE:**

- **Equipo para desarrollo:** Los recursos técnicos para desarrollo presentes en la unidad técnica donde se realizará la implementación del sistema informático se especifican en la tabla 1.6.

Equipo	Disco duro	Memoria RAM	Procesador	Tarjetas de red
PC 1	320 GB	2 GB	Pentium Dual Core 2.7 GHz	Ethernet
PC 2	500 GB	2 GB	Core 2 Duo 2.93 GHz	Ethernet
PC 3	500 GB	2 GB	Core 2 Duo 2.93 GHz	Ethernet
PC 4	320 GB	2 GB	Pentium Dual Core 2.7 GHz	Ethernet

Tabla 1.6. Equipo de desarrollo de la unidad informática.

Para el desarrollo del sistema informático es necesario un equipo con 5.1 GHz de procesamiento, 4 GB de RAM y 7.75 GB de espacio en disco. Tomando en cuenta que todos los procesadores del equipo de desarrollo son de 2 núcleos (físico o virtual) con velocidad de entre 5.4 y 5.86 GHz, la memoria física más la memoria virtual de todos los equipos suman 4 GB y el amplio espacio en disco duro, el equipo para desarrollo es adecuado.

El hardware con el que cuenta el equipo de trabajo se muestra en la tabla 1.7.

Equipo	Disco duro	Memoria RAM	Procesador	Tarjetas de red
PC 1	160 GB	3 GB	Athlon X2 2.6 GHz	Ethernet
PC 2	500 GB	4 GB	Core 2 Duo 2.7 GHz	Ethernet
PC 3	250 GB	3 GB	Intel Core i3 2100 3.10 GHz	Ethernet
PC 4	500GB	2 GB	Core 2 Duo E4500 2.20 GHz	Ethernet

Tabla 1.7. Equipo de desarrollo del grupo de trabajo.

Para el desarrollo del sistema informático es necesario un equipo con 5.1 GHz de procesamiento, 4 GB de RAM y 7.75 GB de espacio en disco. Tomando en cuenta que todos los procesadores del equipo de desarrollo son de 2 núcleos (físico o virtual) con velocidad de entre 5.2 y 6.2 GHz, la memoria física más la memoria virtual de los equipos van desde 4 GB hasta 8 GB y el amplio espacio en disco duro, el equipo para desarrollo es adecuado.

- **Equipo para producción:** Los recursos técnicos para producción presentes en la unidad técnica donde se realizará la implementación del sistema informático se especifican en la tabla 1.8:

Equipo	Disco duro	Memoria RAM	Velocidad del procesador	Tarjetas de red
Servidor 1	2 x 600 GB	8 GB	2 x Xeon 2.4 GHz	Gigabit
Servidor 2	2 x 600 GB	8 GB	2 x Xeon 2.4 GHz	Gigabit

Tabla 1.8. Equipo para producción.

Para la implementación del sistema informático es necesario un equipo con 2 GHz de procesamiento, 2 GB de RAM y 5.75 GB de espacio en disco. Tomando en cuenta que todos los procesadores del equipo de desarrollo son de 2 núcleos con velocidad de 4.8 GHz, la memoria física más la memoria virtual de todos los equipos suman 16 GB y el amplio espacio en disco duro, el equipo para producción es adecuado.

- **Equipo de operación:** El equipo con que se cuenta para operar el sistema informático en las instalaciones del museo se muestra en la tabla 1.9.

Equipo	Disco duro	Memoria RAM	Velocidad del procesador	Tarjetas de red
Equipo 1	250 GB	4 GB	Core 2 Duo E750 2.93 GHz	Gigabit

Tabla 1.9. Equipo de operación.

Para la operación del sistema informático es necesario un equipo con 1 GHz de procesamiento, 1 GB de RAM y entre 3 GB y 16 GB de espacio en disco. Tomando en cuenta que el procesador del equipo de operación es de 2 núcleos con velocidad de 4.86 GHz, la memoria física más la memoria virtual suma 8 GB y el amplio espacio en disco duro, el equipo para operación es adecuado.

- **Otros recursos:** La Secretaría de Cultura cuenta también con una red local con topología de estrella con acceso a internet de banda ancha.

- **REQUERIMIENTOS DE PERSONAL:**

La unidad técnica posee 4 ingenieros de sistemas con conocimientos en análisis y diseño de sistemas, redes, soporte técnico, seguridad y desarrollo de software, quienes operan actualmente sistemas informáticos que usan las mismas tecnologías propuestas para el desarrollo e implementación del sistema informático a desarrollar.

Después de realizar el análisis previo, se concluye que:

- La unidad técnica posee sistemas informáticos en producción con similares características tecnológicas y requerimientos de recursos.
- La unidad técnica cuenta con el equipo y software necesarios para desarrollar, dar mantenimiento e implementar el sistema informático a desarrollar.
- Se cuenta con el personal con la capacidad técnica y conocimientos necesarios para administrar y operar el software a desarrollar.
- El equipo de trabajo cuenta con los recursos técnicos para el desarrollo del sistema informático.

Por lo tanto, el desarrollo del sistema informático es **factible técnicamente**.

1.4.2. FACTIBILIDAD ECONÓMICA

La factibilidad económica determina si los beneficios que se obtendrán a partir del sistema informático serán superiores a los costos de desarrollo, implementación y operación del mismo.

Cabe mencionar que en la actualidad, tanto en la Secretaría de Cultura como en el MUHNES no se lleva un registro detallado sobre el tiempo y dinero que se consume al momento de desarrollar, implementar y operar un sistema informático, así como el proceso de registro, consulta y generación de reportes del área de zoología. Sin embargo, el desarrollo del sistema informático y su implementación generará beneficios cualitativos que se mencionan en esta sección.

- **Evaluación sistema actual y del sistema informático propuesto:** El sistema actual de registro, clasificación, consulta y generación de reportes del área de zoología del MUHNES estima un costo de operación anual de \$150,000.00, que incluye además el mantenimiento de los especímenes, refrigeración, consumo eléctrico, limpieza, salarios, fumigaciones, instalaciones, entre otros gastos. Aun con la implementación del sistema informático propuesto, los costos de operación permanecerán invariables.

El objetivo principal del sistema informático es la sistematización de la información. El registro en libros siempre se mantendrá ya que son considerados patrimonio nacional del país, y por lo tanto su valor económico no puede ser calculado, tal como lo redactó Alfred Marshall, economista británico, en una publicación del año 1891: *“Es imposible evaluar objetos tales como los cuadros de los grandes maestros o las monedas extrañas, puesto que son únicos en su especie y no tienen ni equivalente ni competidor”*. Sin embargo, el sistema informático proveerá beneficios al área de zoología, entre los que se pueden mencionar la ampliación de la cobertura, mejorar la calidad del servicio, la optimización de recursos, entre otros.

- **Beneficios:** Ya que no se puede cuantificar con algún método científico o estadístico el beneficio que el sistema informático brindara al área de zoología del Museo de Historia Natural Saburo Hirao, los beneficios serán cualitativos, siendo descritos a continuación:
 1. **La preservación de la información de la colección de zoología para el museo.** El sistema informático pretende mantener en una base de datos la información de la colección de zoología que garantice la disponibilidad de la misma y reduzca la manipulación de los libros de registro, evitando su deterioro. La base de datos permite realizar copias de seguridad de forma más sencilla que al hacerlo con los libros de registro directamente. De esta forma, se preservará este patrimonio cultural.
 2. **Uso óptimo del recurso humano del área de zoología.** Actualmente los procesos de la colección de zoología los cuales consisten en registro de especímenes, generación de reportes, atención a visitantes e investigaciones de malacología son llevados por una sola persona. El sistema informático permitirá un uso óptimo del recurso humano, reduciendo la carga de trabajo del personal actual permitiendo que se realice el trabajo de manera eficiente.
 3. **Agilizar el registro de especies animales.** El registro de la colección de zoología se realiza actualmente de forma manuscrita en libros de registros. Sistematizando esta operación, se agilizará el registro de especies animales previamente identificadas, apoyando la clasificación y facilitando las búsquedas relacionadas al espécimen.
 4. **Garantizar información completa sobre la colección de zoología.** El registro de la colección de zoología en libros no permite detallar toda la información del espécimen debido al espacio físico de las hojas, lo cual afecta negativamente al momento de clasificar las muestras. El sistema informático garantizará que se registre toda la información necesaria sobre los especímenes.
 5. **Mejorar el tiempo de generación de reportes.** La complejidad de reportes depende de su fin. Estos van dirigidos a la coordinación del museo. Estos reportes son realizados por la encargada del área de zoología del museo, tardando una hora para la elaboración de cada uno de ellos. El sistema informático facilitará la generación de reportes debido a que el proceso estará automatizado, obteniendo los datos directamente de la base de datos evitando la lenta y secuencial búsqueda requerida actualmente para obtener la información necesaria para generarlos.
 6. **Mejorar la búsqueda de información.** La búsqueda de información de un espécimen en particular es un proceso complicado y lento para la carga de trabajo, ya que se tiene que hacer las búsquedas de los datos en libros de registro lo cual dificulta en gran medida la atención que se presta a las personas que se acercan al museo en busca de información para sus investigaciones. Con el sistema informático funcionando, la información se proporcionará de forma más ágil y segura.

7. **Mejorar la calidad de atención a visitantes del área de zoología.** La cantidad de personas que se acercan al Museo a realizar investigaciones varía según la época del año, en promedio el número de visitas es de 260. Entre estas personas tenemos investigadores nacionales e internacionales, estudiantes universitarios y reporteros. Debido a que la información que ellos solicitan se encuentra en libros de registros y habiendo solo una persona encargada en el área de zoología, el tiempo de atención a estas personas es bastante elevado. Con el sistema informático se mejorará la calidad de atención y facilitará la búsqueda de información para ser brindada a los interesados.
8. **Incremento de la cobertura de personas que consultan la información.** La consulta de la información de las especies animales de El Salvador no será exclusiva para investigadores sino que podrá ser consultada también por el público en general. Al ingresar al Museo, pueden observar una parte de la colección de zoología que se encuentra en exhibición, con una pequeña explicación de estos. Sin embargo, esta información no es suficiente y no se permite a estos estudiantes poder conocer información más detallada de especies de animales que existen en el país. Con el sistema informático se eliminarán las restricciones actuales que tienen los estudiantes para acceder a la información completa acerca de la fauna de El Salvador.

El sistema informático brindará beneficios cualitativos al Museo de Historia Natural Saburo Hirao, apoyando en gran medida a que pueda cumplir con su misión. Entre estos beneficios tenemos:

- Preservación de la información de la colección de zoología.
- Optimización del recurso humano del área de zoología.
- Agilizar el registro de los especímenes y contar con información completa sobre ellos.
- Mejorar la generación de reportes, búsqueda de información y la atención a los visitantes del área de zoología.
- Incrementar la cobertura de personas que buscan información sobre la fauna salvadoreña.

El costo del proyecto se estima en \$34,775.40¹; costo que puede considerarse poco comparado a la protección que tendrá la información de la colección de zoología, que es considerada como incalculable debido a que es patrimonio cultural de El Salvador. Desde ese punto de vista, **el proyecto es factible.**

1.4.3. FACTIBILIDAD OPERATIVA

La factibilidad operativa permite determinar si el sistema informático será utilizado por los usuarios finales una vez implementado, analizando aspectos como la resistencia al cambio, volúmenes de datos y tiempos de respuesta.

¹ Ver anexo A. Planificación de recursos.

La factibilidad operativa permite determinar si el sistema informático será utilizado por los usuarios finales una vez implementado, analizando aspectos como la resistencia al cambio, volúmenes de datos y tiempos de respuesta.

- **Volumen de datos:** El área de zoología del museo ha brindado la información que contienen los libros de registro, el cual puede ser observado en las tablas 1.10.a y 1.10.b.

Nombre del campo	Tipo	Longitud	Peso (bytes)
N° de registro	Numérica	6	12
Phylum	Texto	20	40
Orden	Texto	20	40
Suborden	Texto	20	40
Clase	Texto	20	40
Subclase	Texto	20	40
Familia	Texto	20	40
Subfamilia	Texto	20	40
Genero	Texto	20	40
Especie	Texto	20	40
Sinonimias	Texto	30	60
Nombre común	Texto	30	60
Número de muestras	Numérico	3	6
Estado de la muestra	Texto	30	60
Descripción de la muestra	Texto	50	100
• Textura	Texto	15	30
• Coloración	Texto	20	40
• Huevo----Larva---Renacuajo--- Juvenil---- Adulto	Texto	9	18
• Medidas			
o Longitud	Numérica	2	4
o Diámetro	Numérica	2	4
o Altura	Numérica	2	4
Características taxonómicas de la muestra	Texto	100	200
Sexo	Texto	6	12
Hábitat	Texto	10	20
Condición de marea	Texto	5	10
Tipo de sustrato	Texto	15	30
Profundidad (mts)	Numérica	4	8
Ph	Numérica	2	4
Salinidad	Numérica	4	8

Tabla 1.10.a. Campos del libro de registro

Nombre del campo	Tipo	Longitud	Peso (bytes)
Altura sobre nivel del mar(msnm)	Numérica	4	8
Lugar de recolecta	Texto	30	60
• Otras Localidades	Texto	30	60
Equipo de Recolecta	Texto	30	60
Forma de preservación de la muestra	Texto	20	40
Fecha de Recolecta	Numérica	8	16
Recolector	Texto	50	100
Determinó	Texto	50	100
Confirvió	Texto	50	100
Rango Distribución geográfico	Texto	40	80
Estatus Local	Texto	10	20
Fotografía de la muestra			2,457,600
Mapa de distribución de la muestra			
Observaciones	Texto	100	200
Referencias bibliográficas.	Texto	100	200
TOTAL			2,459,702

Tabla 1.10.b. Campos del libro de registro.

Con la información anterior, se pueden realizar los siguientes cálculos:

- Según la información brindada en el área de zoología, el formulario es completado en un promedio de 50 veces por mes:

Total de archivo maestro = $(2,459,702 \text{ Bytes} \times 50) = 122,985,100 \text{ Bytes}$

Total en KB = $122,985,100 \div 1024 \text{ KB} = 120,102.64 \text{ KB}$

Total en MB = 117.29 MB

Total Anual = $(117.29 \text{ MB mensual}) \times (12 \text{ meses})$

Total Anual = 1,407.48 MB \approx 1.38 GB \approx 1.4 GB

- Volumen de datos de índices anual:

Volumen de datos de índices anual = $(15\%) \times (\text{Volumen de datos de tablas anual})$

Volumen de datos de índices anual = $(0.15) \times (1,407.48 \text{ MB})$

Volumen de datos de índices anual = 211.13 MB \approx 212 MB \approx 0.2 GB

- Volumen de datos de tablas históricas anual:

Volumen de datos de tablas históricas anual = $(\text{Volumen de datos de tablas anual}) \times (3 \text{ Años})$

Volumen de datos de tablas históricas anual = $(1,407.48 \text{ MB}) \times (3 \text{ Años})$

Volumen de datos de tablas históricas anual = 4,222.44 MB \approx 4.2 GB

- Volumen de datos de tablas temporales anual:
 Volumen de datos de tablas temporales anual = $(10\%) \times (\text{Volumen de datos de tablas anual})$
 Volumen de datos de tablas temporales anual = $(0.10) \times (1,407.48 \text{ MB})$
Volumen de datos de tablas temporales anual = 140.75 MB \approx 0.14 GB \approx 0.2 GB
- Espacio de intercambio de un disco swap:
 Swap = $(50\%) \times (\text{Tamaño de Memoria RAM del Servidor de Producción})$
 Swap = $(0.5) \times (8 \text{ GB})$
Swap = 4 GB
- Sistema gestor de base de datos:
PostgreSQL (32-bit) = 100 MB \approx 0.1 GB
- Sistema operativo:
Debian Version 6 Squeeze (32-bit) = 5.0 GB
- Lenguaje de desarrollo:
Java = 100 MB = $100 \text{ MB} \div 1,024 \approx 0.1 \text{ GB}$
- Ofimática:
OpenOffice 3.3 = 220 MB = $200 \text{ MB} \div 1,024 \approx 0.2 \text{ GB}$
- Antivirus:
 El servidor con el sistema operativo debían no utiliza antivirus.

El total de los cálculos anteriores se pueden observar en la tabla 1.11.

Elementos	Tamaño en GB por Año
Tablas	1.4
Tablas históricas	4.2
Tablas temporales	0.2
Índices	0.2
Swap	4.0
Sistema manejador de base de datos	0.1
Sistema Operativo	5.0
Lenguaje de Desarrollo	0.1
Ofimática	0.2
Antivirus	0.0
Volumen de Datos Total Anuales	15.4

Tabla 1.11. Volumen de datos anuales

- **Tiempos de respuesta:** El tiempo de respuesta se puede representar como se muestra en la figura 1.9:

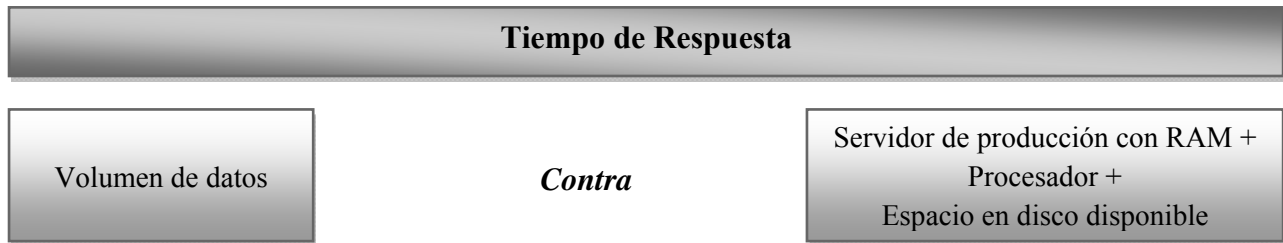


Figura 1.9 Representación del tiempo de respuesta

Si el tamaño del disco duro del servidor es de 600 GB y se sustrae el almacenamiento del volumen total de datos, se tiene:

$$\text{Espacio en disco disponible} = (\text{Tamaño de disco duro}) - (\text{Volumen de datos total})$$

$$\text{Espacio en disco disponible} = 600 \text{ GB} - 15.4 \text{ GB}$$

$$\text{Espacio en disco disponible} = \mathbf{584.6 \text{ GB}}$$

Aplicando una regla de 3, se dispone de:

$$600 \text{ GB} \text{ ----- } 100\%$$

$$584.6 \text{ GB} \text{ ----- } X$$

$$\text{Espacio en disco disponible en \%} = (584.6 \text{ GB} \times 100\%) \div 600 \text{ GB}$$

$$\text{Espacio en disco disponible en \%} = \mathbf{97.43 \%}$$

El espacio de almacenamiento es de 586.38 GB, que es un equivalente a 97.73% de espacio disponible en el servidor. Asumiendo un crecimiento lineal durante 5 años, tiempo en que se utiliza el equipo antes de ser reemplazado por otro, se tiene:

$$\text{Volumen de datos total} = 15.4 \text{ GB} \times 5 = \mathbf{77 \text{ GB}}$$

$$\text{Espacio en disco disponible} = 600 \text{ GB} - 77 \text{ GB} = \mathbf{523 \text{ GB}}$$

$$\text{Espacio en disco disponible en \%} = \mathbf{87.17 \%}$$

El tiempo de respuesta se puede observar en la tabla 1.12.

Tiempo de Respuesta		
Volumen de datos total		Servidor de producción
77 GB	Versus	Memoria RAM: DDR2 8 GB Procesador: Intel Xeon 2.4Ghz. Espacio en disco disponible: 600 GB

Tabla 1.12. Tiempo de respuesta.

- **Demanda de red:** Los datos referentes al ancho de banda con el que cuenta la Secretaría de Cultura, no fueron proporcionados. Sin embargo, tomando como parámetro base las 250 visitas anuales actuales que recibe el museo, se considera que la unidad técnica tiene el mínimo de capacidad necesario para que pueda operar el sistema sin problemas.
- **Resistencia al cambio:** En las diferentes reuniones con la persona que será la futura usuario directo del sistema, concluimos que no habrá resistencia al cambio ya que esta misma persona ha sido la que ha dado a expresar la urgencia de implementar un sistema informático para mejorar la manera en que se maneja la información.

Después de analizar la situación, se concluye que:

- No se espera ninguna resistencia al cambio con respecto al uso del sistema informático una vez implementado, ya que es una necesidad expresada por los usuarios.
- La capacidad del servidor en el que se implementaría el sistema tiene la capacidad suficiente para manejar toda la información que se genere y las peticiones que se le hagan a esta.
- Se cuenta con suficiente espacio disponible en disco duro para almacenar la información generada por el sistema.

Por lo tanto, se concluye que es **factible operativamente** la realización del sistema informático.

1.4.4. RESUMEN DE FACTIBILIDADES

Luego de haber realizado el estudio de factibilidades, se concluye que:

- La unidad de informática de la Secretaría de Cultura cuenta con sistemas informáticos en producción con similares características al software a desarrollar, además cuenta con el hardware, software y personal capacitado para operar y administrar el sistema informático.
- Los beneficios cualitativos que brindará el sistema informático al museo ayudarán a que se maneje de forma eficiente la información de la colección de zoología permitiendo la diseminación de la misma.
- El sistema informático será usado por el personal del museo sin ningún inconveniente, y se cuenta con los recursos suficientes para el almacenamiento de la información.

Luego de haber analizado que el proyecto cumple con las tres factibilidades, se puede concluir que el desarrollo del sistema informático es factible técnica, económica y operativamente, por lo **que el proyecto si puede llevarse a cabo.**

1.5. MODELADO DEL PROCESO PROPUESTO

A partir del análisis de la situación actual del área de zoología, se presenta una propuesta de solución para que los procesos se realicen de mejor manera utilizando la notación BPMN, siendo estos:

1. **Gestión de usuarios:** El sistema propuesto debe permitir agregar usuarios y definir las operaciones que estos podrán realizar dentro del sistema, así como su posterior modificación. Este proceso estará a cargo de un administrador del sistema. Los pasos sugeridos a seguir son:
 - A. Si es un nuevo usuario, proceder a registrarlo. En caso contrario, seguir al paso E.
 - B. Ingresar los datos del usuario, incluyendo nombre de usuario y contraseña.
 - C. Asignar un rol al usuario.
 - D. Guardar los cambios realizados dentro del sistema informático.
 - E. Si se modificara la información del usuario seguir al paso F, de lo contrario salir del sistema sin guardar ningún cambio.
 - F. Modificar la información del usuario.
 - G. Guardar los cambios realizados dentro del sistema informático.

En la figura 1.10 puede observarse este proceso con más detalle.

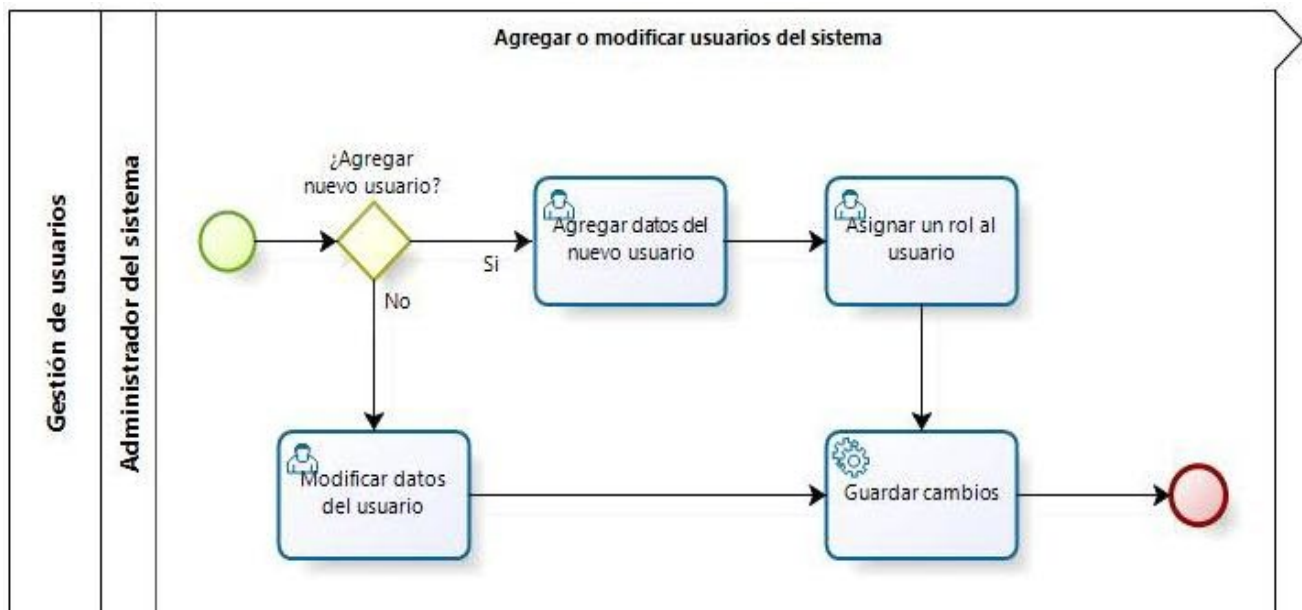


Figura 1.10. Proceso propuesto de gestión de usuarios.

2. **Clasificación de especímenes:** Este proceso se mantiene sin modificación, con la excepción que la información puede consultarse directamente desde la base de datos por el biólogo a cargo, facilitando la búsqueda de las características de la especie a identificar. Los pasos a seguir son:
 - A. Observar las características taxonómicas del espécimen (recurriendo a la disección en algunos casos).
 - B. Consultar las características taxonómicas en libros, sitios web especializados e investigadores.
 - C. Comparar los resultados con los libros de registros y la información contenida en el sistema.
 - D. Definir el tipo de espécimen.

En la figura 1.11 puede observarse este proceso.

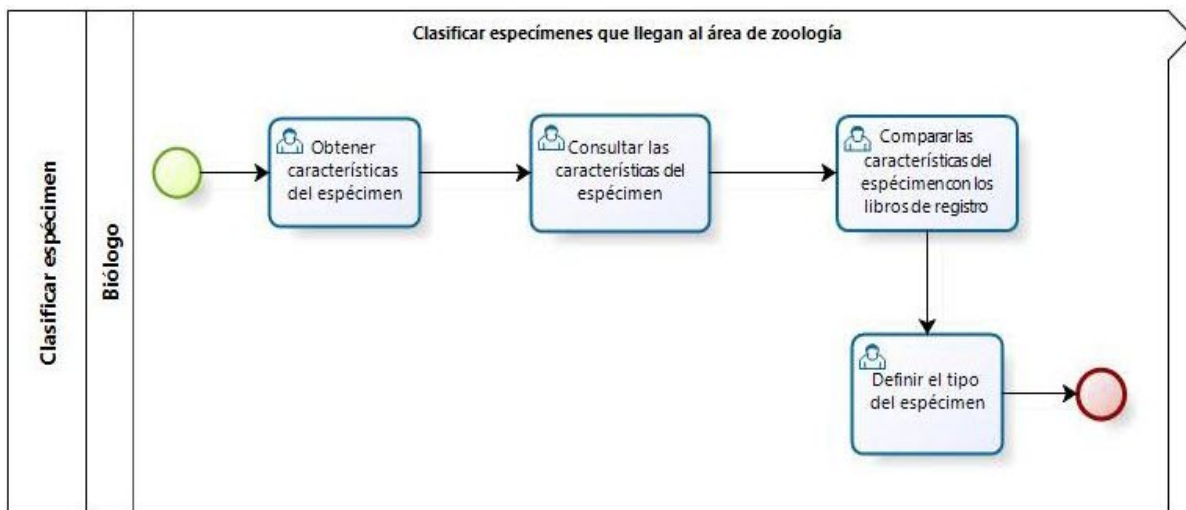


Figura 1.11. Proceso propuesto de clasificación de especímenes.

3. **Registro de especímenes:** Si los especímenes no están clasificados, se realizara el proceso de clasificación por el biólogo a cargo. En caso contrario, un digitador puede ingresar esta información al sistema. Los pasos que debe seguir este proceso son:
 - A. El biólogo a cargo recibe los especímenes previamente clasificados.
 - B. El digitador consulta la ficha de campo de cada uno de los especímenes.
 - C. El digitador registra los especímenes en los libros de registro y dentro del sistema informático.
 - D. El biólogo revisa la información de los especímenes en el sistema informático.
 - E. El biólogo a cargo almacena cada espécimen en su lugar correspondiente.

En la figura 1.12 se muestra este proceso.

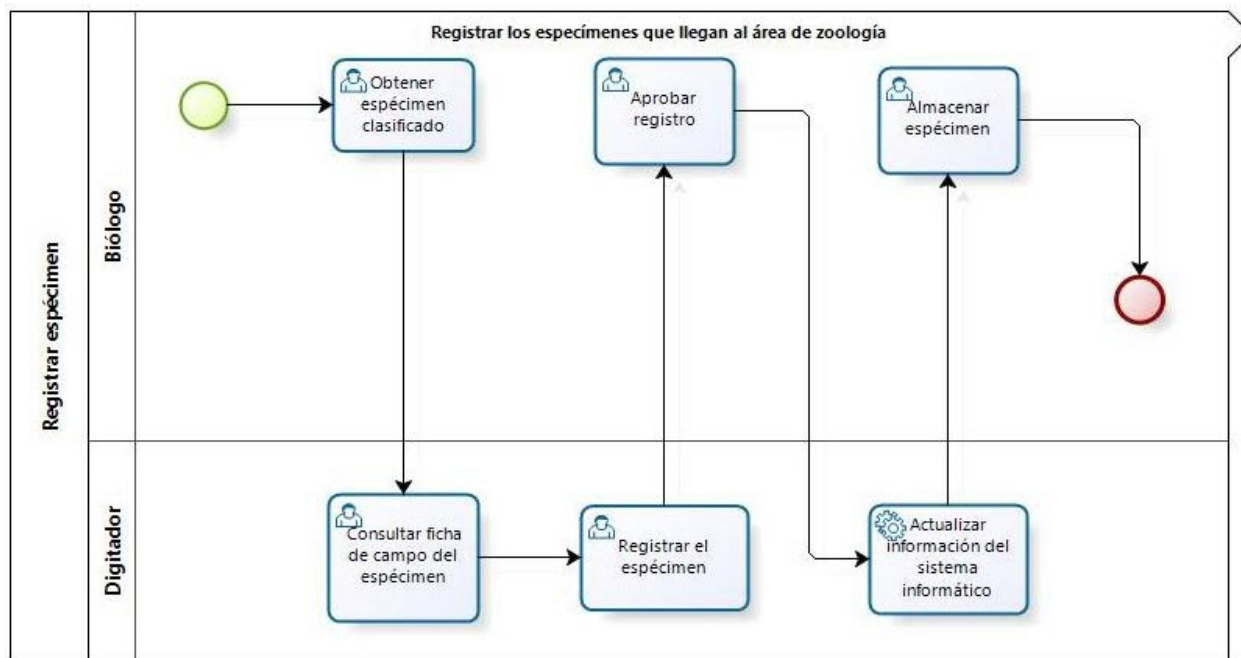


Figura 1.12. Proceso propuesto de registro de especímenes.

4. **Gestión de recolectores:** El sistema informático permitirá llevar una lista de los recolectores que participan en la colecta de especímenes para el área de zoología, guardando la información de contacto, código, y posterior modificación en caso de ser necesario. Este proceso será llevado a cabo por el biólogo a cargo. Los pasos sugeridos de este proceso se muestran a continuación:
- A. Si se ingresará un nuevo recolector, proceder a registrarlo. En caso contrario continuar con el paso D.
 - B. Ingresar la información personal del recolector y su código.
 - C. Guardar la información del recolector dentro del sistema.
 - D. Si se modificará la información del recolector, proceder a modificarla. De lo contrario continuar con el paso G.
 - E. Modificar la información del recolector.
 - F. Guardar los cambios realizados dentro del sistema.
 - G. Consultar la información de los recolectores.
 - H. Salir del sistema informático sin realizar ningún cambio.
- Se puede observar gráficamente este proceso en la figura 1.13.

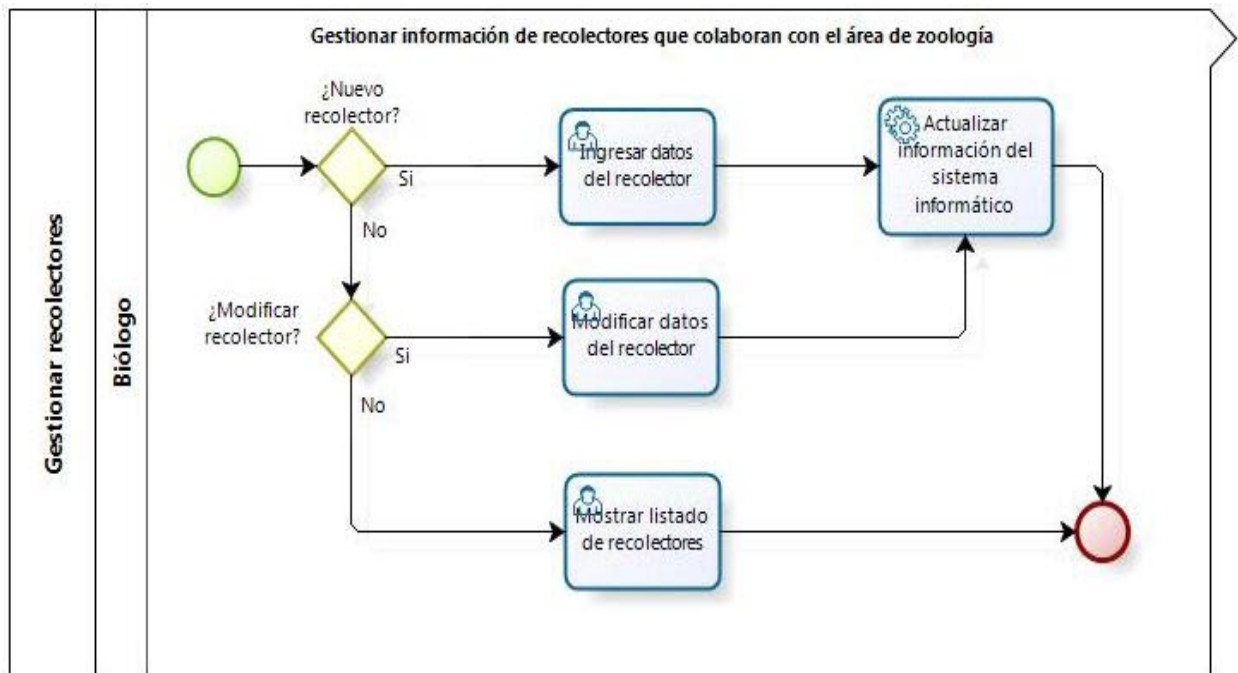


Figura 1.13. Proceso propuesto de gestión de recolectores.

5. **Consulta de información:** La información podrá ser consultada desde el sistema informático, permitiendo al usuario (incluye usuarios no registrados), la selección de los campos de interés, mostrándolos en pantalla, con la excepción de que un usuario no registrado no podrá consultar los recolectores. A continuación se describe el proceso:
- A. Definir qué información será consultada.
 - B. Consultar uno o más campos de interés.
 - C. Mostrar la información solicitada en pantalla.
 - D. Si se realizara otra consulta, regresar al paso A. De lo contrario salir del sistema.
- En la figura 1.14 se muestra con más detalle este proceso.

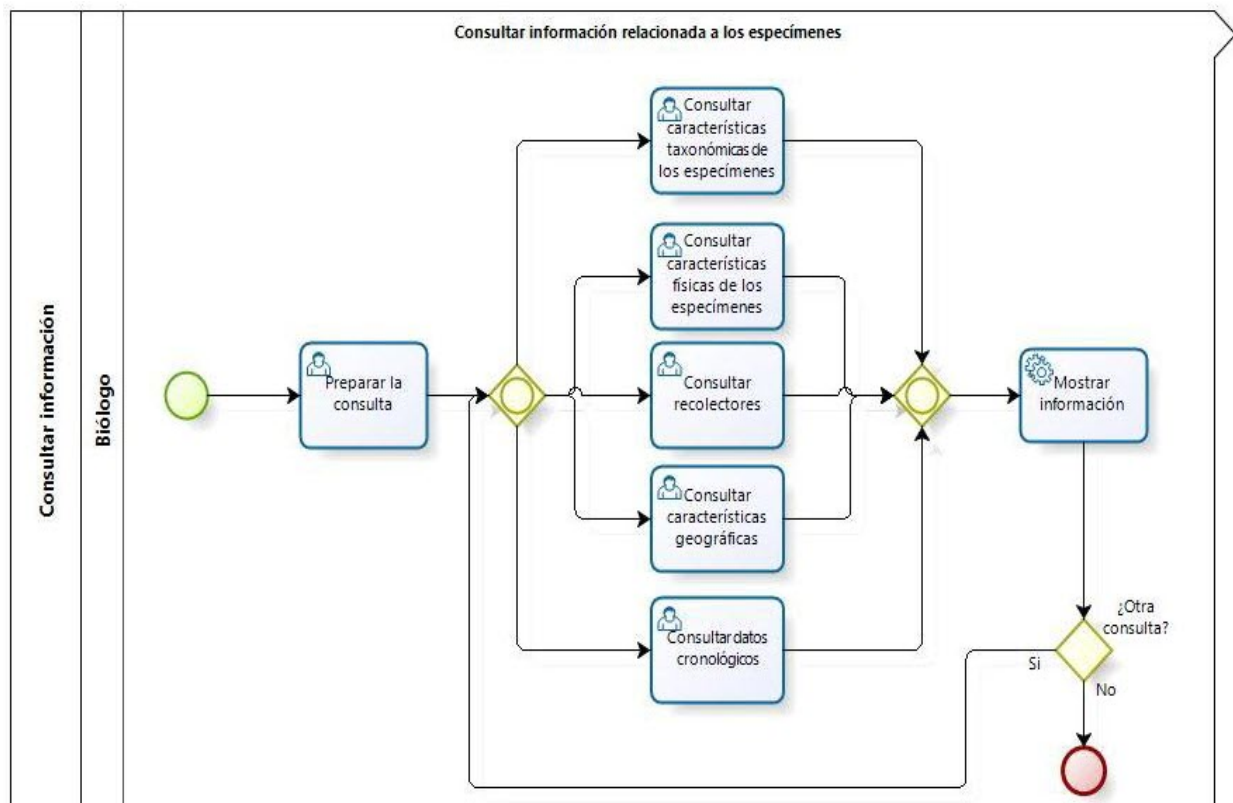


Figura 1.14. Proceso propuesto de consulta de información.

6. **Generación de reportes:** Los reportes serán generados por el sistema informático. Para ellos se realizará el subproceso de consulta de información para reportes, donde se seleccionará la información de interés. Luego se procederá imprimir dicho reporte o guardarlo dentro de una unidad de almacenamiento. El biólogo a cargo realizará este proceso. A continuación se presentan los pasos a seguir en este proceso:

- A. Identificar la información que se desea reportar.
- B. Realizar el proceso de consulta de información para reportes.
- C. Detallar la información que se desea reportar.
- D. Guardar el reporte en un medio de almacenamiento o proceder a la impresión del mismo.
- E. Enviar el reporte al solicitante

Este proceso se muestra en la figura 1.15.

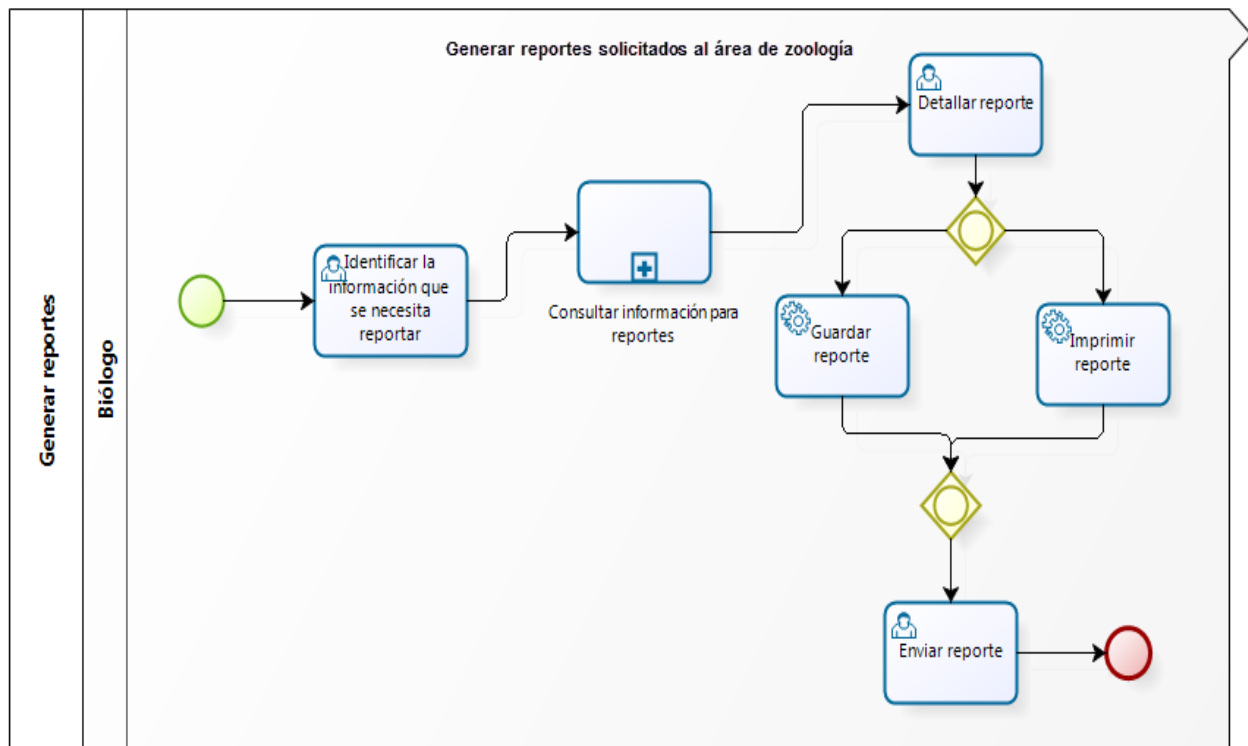


Figura 1.15. Proceso propuesto de generación de reportes.

7. **Gestionar restauraciones:** Las restauraciones son esenciales para la conservación de los especímenes. Para ello, se debe ingresar el código del espécimen, fecha de restauración, y las personas encargadas de éste. El biólogo a cargo realizará este proceso. A continuación se presentan los pasos a seguir en este proceso:
- A. Si se registrara una nueva restauración, seguir con el paso B. en caso contrario, seguir con el paso D.
 - B. Ingresar el código del espécimen, fecha de restauración, nombre de la persona que lo autoriza y responsable de la pieza.
 - C. Actualizar la información del sistema.
 - D. Si se registrara la finalización de un mantenimiento, seguir con el paso E. en caso contrario, seguir con el paso G.
 - E. Ingresar el código del espécimen, la fecha de finalización de la restauración y quien recibió la pieza.
 - F. Actualizar la información del sistema.
 - G. Consultar la información de las restauraciones realizadas y en proceso.
 - H. Salir del sistema.

Gráficamente este proceso se muestra en la figura 1.16.

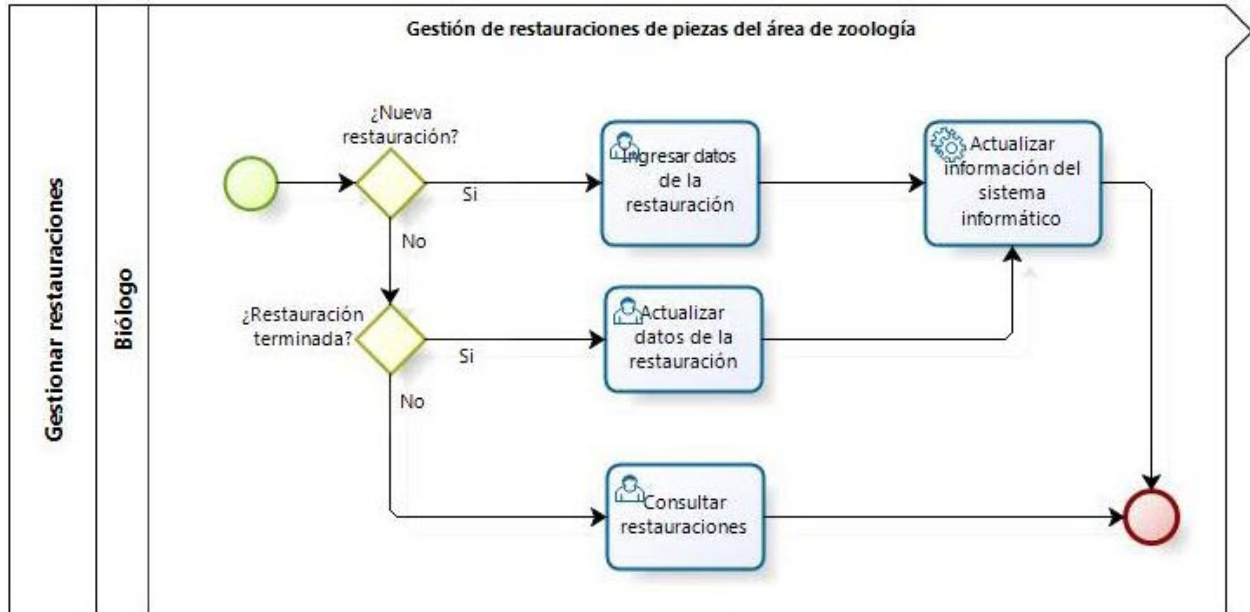


Figura 1.16. Gestión de restauraciones.

CAPÍTULO 2:

REQUERIMIENTOS

En la ingeniería de sistemas, un requerimiento es una necesidad documentada sobre el contenido, forma o funcionalidad de un sistema. Se usa en un sentido formal en la ingeniería de sistemas para definir qué es lo que el usuario requiere de un sistema. Los requisitos se utilizan como datos de entrada en la etapa de diseño del sistema. Establecen qué debe hacer el sistema, pero no cómo hacerlo.

2.1. METODOLOGÍA PARA LA DETERMINACIÓN DE REQUERIMIENTOS

Una metodología para la determinación de requerimientos permite obtener con certeza lo esperado por los usuarios al finalizar el desarrollo del proyecto. Los elementos utilizados para su determinación son:

- **Identificación de los requerimientos con los usuarios:** Auxiliados de los métodos de cuestionarios, entrevistas y observación directa, se realizaron reuniones con el personal del MUHNES y la Secretaría de Cultura. En dichas reuniones se expusieron las necesidades a satisfacer y características con las que deberá contar el sistema informático una vez puesto en producción. Como herramienta auxiliar en esta fase se procedió al uso de videograbadora con el consentimiento de los usuarios.
- **División de los requerimientos de acuerdo a su origen:** Los requerimientos identificados serán divididos de acuerdo a la forma en que estarán plasmados en el sistema informático, siendo estos:
 - Informáticos
 - Operativos
 - De desarrollo
 - De implementación
- **Elaboración del documento de especificación de requerimientos:** Se elaboró un documento de requerimientos que cumple con las características de un buen SRS (Especificaciones de Requisitos de Software, por sus siglas en español) definidas en la norma IEEE/ANSI 830- 1998¹, siendo estas:
 - Correcto
 - Inequívoco
 - Completo
 - Consistente
 - Comprobable
 - Modificable
 - Identificable
- **Validación de los requerimientos con los usuarios:** Las especificaciones de los requerimientos fueron presentados al personal del MUHNES y de la Secretaría de Cultura para su revisión, corrección, modificación y aprobación.

¹ http://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf

2.2. REQUERIMIENTOS INFORMÁTICOS

Los requerimientos informáticos son aquellas necesidades de información que debe satisfacer el nuevo sistema. Estas necesidades son expuestas por los usuarios de negocios, que para este proyecto están conformados por los usuarios del MUHNES y de la Secretaría de Cultura.

Los requerimientos informáticos están divididos en dos grupos:

- **Requerimientos funcionales:** Son los que definen la funcionalidad o las características que el sistema deberá contar, según las necesidades del usuario.
- **Requerimientos no funcionales:** Son los que describen aspectos del sistema que están relacionado con lo bien que cumple los requerimientos funcionales. Estos requerimientos están asociados con la calidad del sistema¹.

En la tabla 2.1 se muestra el listado de requerimientos informáticos funcionales.

Identificador	Requerimiento
RFC001	Autenticar usuario.
RFC002	Administrar privilegios.
RFC003	Administrar roles.
RFC004	Administrar usuarios.
RFC005	Gestionar información taxonómica.
RFC006	Gestionar animales.
RFC007	Gestionar reseñas de animales.
RFC008	Gestionar especímenes.
RFC009	Gestionar información geográfica.
RFC010	Gestionar datos de recolectores.
RFC011	Gestionar información de recolecciones.
RFC012	Gestionar restauraciones de piezas.
RFC013	Importar catálogo taxonómico.
RFC014	Exportar catálogo taxonómico.
RFC015	Generar reportes fijos.
RFC016	Generar reportes personalizados.
RFC017	Presentar información de los especímenes a usuarios no registrados.

Tabla 2.1. Requerimientos funcionales.

¹ Norma ISO/IEC 9126

En la tabla 2.2 se muestra el listado de los requerimientos no funcionales

Identificador	Característica	Requerimiento
RNF001	Usabilidad	Poseer una interfaz sencilla y usable.
RNF002	Funcionalidad	Perfiles de usuario.
RNF003	Funcionalidad	Expiración de sesión.
RNF004	Funcionalidad	Contraseñas cifradas.
RNF005	Funcionalidad	Cambio de contraseñas.
RNF006	Funcionalidad	Recuperación de contraseña
RNF007	Funcionalidad	Registrar en bitácora.
RNF008	Eficiencia	Uso adecuado de los recursos
RNF009	Fiabilidad	Tolerancia a fallos
RNF010	Mantenibilidad	Facilidad de cambios
RNF011	Portabilidad	Coexistencia con otros sistemas.

Tabla 2.2. Requerimientos no funcionales.

En la tabla 2.3 se muestra el formato de descripción de requerimientos informáticos, tanto funcionales como no funcionales.

Identificador	<< Identificador del requerimiento >>
Nombre	<< Nombre del requerimiento >>
Prioridad	<< Nivel de precedencia en el desarrollo del proyecto >>
Alcance	<< Indica todo lo que incluirá el requerimiento >>
Descripción	<< Descripción detallada del requerimiento >>
Comentarios	<< Comentarios u observaciones adicionales sobre el requerimiento >>

Tabla 2.3. Descripción de requerimientos informáticos.

2.2.1. REQUERIMIENTOS FUNCIONALES

Identificador	RFC001
Nombre:	Autenticar usuario.
Prioridad:	Alta.
Alcance:	Verificar las credenciales y perfil de acceso al sistema.
Descripción:	El sistema deberá verificar que el usuario pertenezca al sistema y que tenga los permisos necesarios para realizar los diferentes procesos del sistema.
Comentarios:	N/A.

Tabla 2.4. RFC001.

Identificador	RFC002
Nombre:	Administrar privilegios.
Prioridad:	Alta.
Alcance:	Deberá permitir el registro, modificación, consulta y eliminación de privilegios.
Descripción:	El sistema contara con diferentes tipos de privilegios, los cuales son necesarios para efectuar las operaciones dentro del sistema.
Comentarios:	Esta operación podrá ser realizada sólo por el administrador del sistema.

Tabla 2.5 RFC002.

Identificador	RFC003
Nombre:	Administrar roles.
Prioridad:	Alta.
Alcance:	Deberá permitir el registro, modificación, consulta y eliminación de roles.
Descripción:	El sistema deberá permitir asignar roles a usuarios y crear nuevos roles.
Comentarios:	Esta operación podrá ser realizada sólo por el administrador del sistema.

Tabla 2.6. RFC003.

Identificador	RFC004
Nombre:	Administrar usuarios.
Prioridad:	Alta.
Alcance:	Deberá permitir el registro, modificación, consulta y eliminación de usuarios.
Descripción:	El sistema deberá gestionar la información del usuario que tendrá permisos para realizar los procesos respectivos; los datos de usuarios deberán incluir: <ul style="list-style-type: none"> • Nombre • Correo electrónico • Contraseña
Comentarios:	La eliminación de usuarios será a nivel lógico para preservar los datos del usuario, ya que estarán vinculados a tablas de auditoría y podrá ser realizada sólo por el administrador del sistema.

Tabla 2.7. RFC004.

Identificador	RFC005
Nombre:	Gestionar información taxonómica.
Prioridad:	Alta.
Alcance:	Permitir las opciones de registrar, modificar, consultar y eliminar información taxonómica.
Descripción:	El sistema deberá permitir la gestión de la información taxonómica de los animales.
Comentarios:	La eliminación de esta información será a nivel lógico.

Tabla 2.8. RFC005.

Identificador	RFC006
Nombre:	Gestionar animales.
Prioridad:	Alta.
Alcance:	Permitir las opciones de registrar, modificar, consultar y eliminar animales.
Descripción:	El sistema deberá permitir la gestión de la información del animal que incluye: <ul style="list-style-type: none"> • Nombre común • Nombre científico • Referencias bibliográficas • Rango de distribución
Comentarios:	La eliminación del animal será a nivel lógico.

Tabla 2.9. RFC006.

Identificador	RFC007
Nombre:	Gestionar reseñas de animales.
Prioridad:	Media.
Alcance:	N/A
Descripción:	El sistema deberá permitir el ingreso de una reseña informativa para los animales en formato de texto enriquecido.
Comentarios:	N/A

Tabla 2.10. RFC007.

Identificador	RFC008
Nombre:	Gestionar especímenes.
Prioridad:	Alta.
Alcance:	Permitirá realizar las opciones de registro, consulta, modificación y eliminación del espécimen.
Descripción:	El sistema deberá permitir la gestión de especímenes en la colección de zoología, este registro incluirá: <ul style="list-style-type: none"> • Datos generales • Datos de verificación del registro
Comentarios:	La eliminación del animal será a nivel lógico.

Tabla 2.11. RFC008.

Identificador	RFC009
Nombre:	Gestionar información geográfica.
Prioridad:	Alta.
Alcance:	Permite el registro, consulta, modificación y eliminación de los departamentos, municipios y áreas protegidas de El Salvador.
Descripción:	El sistema deberá permitir el ingreso de la distribución geográfica de El Salvador.
Comentarios:	La eliminación es a nivel físico, ya que fácilmente se puede recuperar.

Tabla 2.12. RFC009.

Identificador	RFC010
Nombre:	Gestionar datos de recolectores.
Prioridad:	Media.
Alcance:	Se podrá registrar, modificar, consultar y eliminar recolector.
Descripción:	El sistema deberá permitir la gestión de la información del recolector como: <ul style="list-style-type: none"> • Nombre completo • Alias del recolector • Teléfono • Dirección • Correo electrónico
Comentarios:	La eliminación del recolector será a nivel lógico.

Tabla 2.13. RFC010.

Identificador	RFC011
Nombre:	Gestionar información de recolecciones.
Prioridad:	Alta.
Alcance:	Se podrá registrar, modificar, consultar y eliminar recolecciones.
Descripción:	El sistema deberá permitir la gestión de las recolecciones realizadas donde se recolectan las muestras para la colección de zoología.
Comentarios:	La eliminación del recolector será a nivel lógico.

Tabla 2.14. RFC011.

Identificador	RF0012
Nombre:	Gestionar restauraciones de piezas.
Prioridad:	Media
Alcance:	N/A
Descripción:	El sistema deberá permitir el manejo de quién y en qué fecha se da el mantenimiento a la pieza, para llevar un control del mismo.
Comentarios:	Las restauraciones de las piezas no podrán ser eliminadas.

Tabla 2.15. RFC012.

Identificador	RFC013
Nombre:	Importar catálogo taxonómico.
Prioridad:	Media.
Alcance:	Se podrá importar solo información taxonómica de los animales.
Descripción:	El sistema deberá permitir la importación de la información taxonómica desde fuentes externas, como lo son bibliotecas de datos de otros museos o de otras colecciones de zoología y añadirla a las tablas de taxonomía del sistema.
Comentarios:	La importación se realizará desde archivos CSV, XML y gestores de bases de datos.

Tabla 2.16. RFC013.

Identificador	RFC014
Nombre:	Exportar catálogo taxonómico.
Prioridad:	Media.
Alcance:	Se podrá exportar solo información taxonómica de los animales.
Descripción:	El sistema deberá permitir la exportación de la información taxonómica, las cuales podrán ser utilizadas por otras entidades para alimentar sus catálogos taxonómicos.
Comentarios:	La exportación se realizara a archivos CSV y XML.

Tabla 2.17. RFC014.

Identificador	RFC015
Nombre:	Generar reportes fijos.
Prioridad:	Alta.
Alcance:	N/A
Descripción:	El sistema deberá permitir la generación de reportes sobre los especímenes, recolecciones, restauraciones y usuarios del sistema.
Comentarios:	Los reportes deberán estar debidamente identificados cómo un documento del MUNHES.

Tabla 2.18. RFC015.

Identificador	RFC016
Nombre:	Generar reportes personalizados.
Prioridad:	Alta.
Alcance:	N/A
Descripción:	El sistema deberá permitir la generación de reportes personalizados sin restricciones de información.
Comentarios:	Los reportes deberán estar debidamente identificados cómo un documento del MUNHES.

Tabla 2.19. RFC016.

Identificador	RFC017
Nombre:	Presentar información de los especímenes a usuarios no registrados.
Prioridad:	Alta.
Alcance:	Los usuarios no registrados solo podrán realizar consultas de la información.
Descripción:	El sistema deberá permitir consultas personalizadas de cualquier tipo ya sea por datos del animal, espécimen, geográficos, de recolección o información taxonómica necesaria.
Comentarios:	Los datos geográficos de recolección deberán mostrarse como las coordenadas del municipio donde se realizó la recolección y no el punto exacto cuando la información sea presentada a usuarios no registrados.

Tabla 2.20. RFC017.

2.2.2. REQUERIMIENTOS NO FUNCIONALES

Identificador	RNF001
Nombre:	Poseer una interfaz sencilla y usable.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema tendrá una interfaz sencilla que permita que los usuarios que utilizan las aplicaciones actuales encuentren de manera fácil los elementos que actualmente utilizan y pueden incorporar a su práctica de trabajo las nuevas propuestas del sistema.
Comentarios:	N/A.

Tabla 2.21. RNF001.

Identificador	RNF002
Nombre:	Perfiles de usuario.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema deberá permitir el manejo de perfiles que indiquen las opciones a las que está autorizado cada uno de los usuarios, solo el administrador puede otorgar dichos perfiles.
Comentarios:	N/A.

Tabla 2.22. RNF002.

Identificador	RNF003
Nombre:	Expiración de sesión.
Prioridad:	Media.
Alcance:	N/A.
Descripción:	El sistema deberá cerrar la sesión luego de 15 minutos de inactividad en las operaciones y mostrar la pantalla para identificación de usuario.
Comentarios:	N/A.

Tabla 2.23. RNF003.

Identificador	RNF004
Nombre:	Contraseñas cifradas.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema cifrará todas las claves que el usuario ingrese a través del algoritmo de encriptación MD5 para mayor seguridad en la información, incluyendo las nuevas o las que se ingresen para identificación.
Comentarios:	N/A.

Tabla 2.24. RNF004.

Identificador	RNF005
Nombre:	Cambio de contraseña.
Prioridad:	Media.
Alcance:	N/A.
Descripción:	El sistema brindará la posibilidad de cambiar las contraseñas cuando el usuario lo solicite al administrador para mayor control del mismo.
Comentarios:	N/A.

Tabla 2.25. RNF005.

Identificador	RNF006
Nombre:	Recuperación de contraseña.
Prioridad:	Media.
Alcance:	N/A.
Descripción:	El sistema deberá recuperar las contraseñas en el caso de ser necesario, pero solo lo podrá hacer el administrador del sistema.
Comentarios:	N/A.

Tabla 2.26. RNF006.

Identificador	RNF007
Nombre:	Registrar en bitácora.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema debe registrar todo tipo de acciones que se realice en el mismo y se guardarán en la tabla bitácora para un control de las acciones realizadas.
Comentarios:	N/A.

Tabla 2.27. RNF007.

Identificador	RNF008
Nombre:	Uso adecuado de los recursos.
Prioridad:	Media.
Alcance:	N/A.
Descripción:	El sistema debe utilizar los recursos de hardware y software de manera de manera eficiente.
Comentarios:	N/A.

Tabla 2.28. RNF008.

Identificador	RNF009
Nombre:	Tolerancia a fallos.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema presentará opciones para disminuir los errores de la información ingresada como lo son listas desplegables y opciones de auto llenado.
Comentarios:	N/A.

Tabla 2.29. RNF009.

Identificador	RNF010
Nombre:	Facilidad de cambios.
Prioridad:	Alta.
Alcance:	N/A.
Descripción:	El sistema debe permitir modificaciones en su estructura de manera sencilla, contando con su respectiva documentación interna y externa.
Comentarios:	N/A.

Tabla 2.30. RNF010.

Identificador	RNF011
Nombre:	Coexistencia con otros sistemas.
Prioridad:	Alta.
Alcance:	N/A
Descripción:	El sistema no debe interferir con el funcionamiento de otros sistemas implementados.
Comentarios:	N/A

Tabla 2.31. RNF011.

2.3. REQUERIMIENTOS OPERATIVOS

Estos requerimientos indican lo que se necesita para que el sistema informático funcione de la manera más eficiente y segura posible. Esto incluye el medio ambiente que rodea al sistema y la ubicación del equipo operativo en un lugar seguro.

2.3.1. REQUERIMIENTOS MEDIOAMBIENTALES

Para que el sistema informático opere según lo previsto y de manera óptima se necesita condiciones medioambientales a cumplir, las cuales son:

- Las instalaciones eléctricas deben estar alambradas de forma ordenada y debidamente identificadas y probadas las conexiones de cableado de vivo, neutro y tierra. De esta forma se evitará el daño en el equipo por variaciones eléctricas de voltaje o corto circuitos.
- El equipo de operación debe estar en un ambiente fresco, preferible en un ambiente con aire acondicionado para ayudar a controlar la temperatura del equipo.
- El equipo de operación no debe estar ubicado bajo los aparatos de aire acondicionado. De esta forma se evitará percances cuando el aire acondicionado tenga problemas de goteo.
- Proteger los equipos conectándolos a baterías UPS para protegerlos en caso de fallo del servicio eléctrico. Estos UPS deben tener una capacidad mínima de 750 watts el cual da un tiempo de 15 minutos de respaldo para poder apagar el equipo de manera correcta. Respecto al servidor, este debe estar ubicado en unas instalaciones donde se cuente con un suministro de energía alterno, como lo es una planta de energía a combustión.
- El equipo debe encontrarse lejos de ventanas para evitar problemas cuando existan situaciones climatológicas de riesgo, por ejemplo lluvia torrencial o ráfagas de viento.
- Si el equipo se encuentra a una altura considerable del suelo, este debe estar a una distancia considerable de la orilla del objeto sobre el que se encuentra. Así se reducirá el riesgo de daños por movimientos sísmicos.
- El equipo no debe contener papeles, volantes ni cualquier otro objeto que obstruya la ventilación de este.
- Los cables de alimentación de corriente, de red y de otros aparatos conectados al equipo deben estar ordenados y en una ubicación donde no sean interceptados por los pies de personas. Es preferible el uso de canaletas para un mayor orden y también deben enrollarse los cables que luego de conectados tengan un sobrante de más de un metro.

2.3.2. REQUERIMIENTOS DE MANTENIMIENTO

El mantenimiento del equipo informático está dividido en 3 grupos:

- **Mantenimiento preventivo:** es el destinado a la conservación del equipo o instalaciones mediante realización de revisiones y reparaciones que garanticen su buen funcionamiento y fiabilidad. Entre las medidas a tomar tenemos:
 - Se recomienda un mantenimiento de limpieza del equipo por lo menos una vez al mes. Esto dependerá de las condiciones medioambientales que rodeen el equipo.
 - Deberá examinarse el estado del hardware y software del equipo solucionando o previniendo problemas que puedan surgir.
 - Deberá asegurarse que el ambiente que rodea al equipo informático se encuentre en condiciones limpias y sin humedad.
 - Se deberán hacer copias de seguridad de la información para que la institución no sea perjudicada por pérdidas de información en el dispositivo de almacenamiento principal o por daños que sufran estos dispositivos.
- **Mantenimiento correctivo:** este mantenimiento se realiza cuando se sustituye hardware, se reinstala o configura software que no está funcionando correctamente así también se realizan cambios en el ambiente que rodea al equipo y puedan ser un posible riesgo. Entre las recomendaciones cuando se realice este mantenimiento tenemos:
 - Revisar que las baterías de los UPS estén funcionando correctamente.
 - Verificar que la temperatura del equipo este en niveles aceptables.
 - Desinstalar cualquier programa que no debería estar instalado.
 - Desinstalar cualquier programa que no esté funcionando correctamente y volverlo a instalar.
 - Revisar que el cableado este ordenado.
 - Verificar que el equipo esté conectado a los UPS.
 - Si el equipo no está actuando como debería, analizar con un antivirus y antispyware.
 - Corregir problemas del registro del sistema operativo.
- **Mantenimiento predictivo:** este mantenimiento consiste en realizar acciones para estar un paso adelante a cualquier falla que ocurra en el equipo. Se recomienda:
 - Analizar periódicamente el equipo por virus y spyware.
 - Realizar copias de seguridad de la información periódicamente.
 - Realizar clonación del disco duro del equipo para tener una copia de seguridad de la configuración de las cuentas de usuario, de seguridad y de configuración de red del equipo y ahorrar tiempo de instalación y configuración del equipo por si necesita reinstalarse.

- Si se sospecha daños en el dispositivo de almacenamiento de la información, este se cambiará por otro para prevenir pérdidas de información y problemas inesperados.

2.3.3. REQUERIMIENTOS OPERATIVOS DE RECURSO HUMANO

Además de un buen equipo técnico informático, se necesita recurso humano con la experiencia necesaria para brindar soporte técnico ante los problemas que surjan mientras se trabaja.

En la tabla 2.32 se muestra el personal mínimo necesario para dar soporte al sistema SIZOO cuando se encuentre en funcionamiento.

Perfil	Descripción	Requisitos
Administrador de servidores	Encargado del óptimo funcionamiento del hardware y software del equipo servidor de la aplicación.	Conocimientos de Glassfish 3.1, redes y configuración de hardware y software de servidores. Conocimientos del framework Hibernate.
Administrador del sistema	Encargado de administrar roles y privilegios del sistema, así como atender los correos enviados por los usuarios.	Conocimiento de los procesos del área de zoología. Conocimientos del lenguaje de programación Java.
Administrador de base de datos	Encargado del acceso a la base de datos y de realizar respaldos de la información.	Conocimientos del gestor de base de datos PostgreSQL 9.
Analistas programadores	Encargados de dar mantenimiento a la aplicación.	Conocimientos del lenguaje de programación Java y los frameworks JSF e Hibernate.

Tabla 2.32. Perfiles de recurso humano operativo.

2.3.4. REQUERIMIENTOS DE SEGURIDAD

El control de la seguridad deberá basarse de la norma ISO/IEC 17799:2007¹ (Código de buenas prácticas para la gestión de la seguridad de la información). Estos requerimientos están divididos en dos grupos:

- **Seguridad Física y del entorno:** según la norma ISO/IEC 17799:2007 en el apartado seguridad se ha definido:
 - **Áreas seguras:** El equipo debe estar ubicado en un área con suficiente espacio para trabajar, en un lugar fresco, preferible con aire acondicionado y de no ser así, en un lugar con una distancia considerable de ventanas, encima de un mueble firme y alejado a una distancia considerable de las orillas, no debe estar debajo de aires acondicionados ni debe estar debajo de estantes que contengan recipientes líquidos u objetos pesados.
 - **Seguridad del equipo:** Evitar pérdidas, daños en los equipos, así como la interrupción de las actividades de la organización.
 - **Suministro eléctrico:** Los tomacorriente deben ser de 3 espigas, es de asegurarse que estos estén alambrados correctamente utilizando probadores portátiles en cada uno de los tomas incluso en los que no está conectado el equipo de trabajo. Los cables eléctricos deben estar ordenados evitando que se enrollen. Es preferible también el uso de regletas que contengan protección contra variaciones de alto voltaje para proteger el equipo.
 - **Mantenimiento de equipo:** Se deberá realizar mantenimiento preventivo, correctivo y predictivo para asegurar el bienestar del equipo informático.
- **Seguridad lógica:** Según la norma ISO/IEC 17799:2007 en el apartado control de accesos se toma en cuenta lo siguiente:
 - **Identificación y autenticación de usuario:** Todos los usuarios tendrán un usuario único y una contraseña para poder ingresar al sistema, de esta manera prevenir el ingreso de usuarios no deseados al sistema. La contraseña tiene la característica que es secreta y no se muestra en el momento de introducirla al sistema además de estar conformada de números, letras y símbolos.
 - **Control de acceso a las aplicaciones y a la información:** Asegura el acceso autorizado de los usuarios y prevenir accesos no autorizados a información que solo puede acceder personal privilegiado. Se definirán e implementarán diferentes niveles de acceso que serán proporcionados por el Administrador a cada usuario limitando la información que este puede ver según los permisos otorgados.
 - **Control de acceso a la red:** Se previene el acceso no autorizado a la red, controlando los servicios a las redes internas y externas, puertos y protocolos, esto en conjunto al evitar accesos no autorizados a los computadores y a la información contenida en el sistema.

¹ <http://www.pecert.gob.pe/media/uploads/isoiec17799.pdf>

- **Gestión de respaldo y recuperación:** Los datos son la parte más importante en un sistema informático, por lo tanto su seguridad es una de las prioridades del Administrador del sistema, éste deberá establecer un plan de copias de respaldo de la información.
- **Seguridad de las aplicaciones del sistema:** El sistema validará que los datos que se ingresen sean los correctos, este verificará que no se digite ninguna incongruencia protegiendo así la integridad de los datos almacenados.
- **Mantenimiento exclusivo del Administrador:** Solo el administrador del sistema o usuarios con derechos de administrador pueden dar mantenimiento al equipo, esto hace que la responsabilidad del buen funcionamiento del sistema caiga sobre el administrador.
- **Acceso restringido servidor de base de datos:** El servidor que almacenará la base de datos del sistema, deberá estar en un lugar de acceso restringido a personal no autorizado, de esta manera personas sin autorización no podrán tener acceso a los datos para modificarlos, copiarlos, robarlos o incluso dañarlos.
- **Bitácora del acceso al sistema:** La información que se maneja en el sistema es lo más importante para la institución por lo tanto al utilizar un sistema informático manejado por varios usuarios necesita que tenga una bitácora del trabajo realizado en el sistema por los usuarios para controlar las acciones realizadas en la información.

2.4. REQUERIMIENTOS DE DESARROLLO

Este apartado describe los recursos que se consideraron necesarios para el desarrollo del sistema informático para el MUHNES. Estos recursos están divididos en 3 grupos:

- Recurso de hardware
- Recurso de software
- Recurso humano

2.4.1. RECURSO DE HARDWARE DE DESARROLLO

En este apartado se describen las características del hardware disponible por el equipo de trabajo para llevar a cabo el desarrollo del sistema informático. Los recursos de hardware están divididos en:

1. **Servidor:** Las especificaciones de la computadora que fue usada como servidor durante el desarrollo del proyecto se muestran en la tabla 2.33.

Equipo	Procesador	Disco Duro	Memoria RAM	Tarjeta de red
Servidor 1	Athlon X2 2.6 GHz	160 GB	3 GB	Ethernet

Tabla 2.33. Servidor de desarrollo.

2. **Computadoras de desarrollo:** Los detalles técnicos de estos recursos pueden observarse en la tabla 1.7 del apartado 1.4.1. Factibilidad Técnica.
3. **Dispositivos de red:** Los dispositivos de la red de comunicaciones que se utilizaron durante el desarrollo del proyecto se muestran en la tabla 2.34.

Recurso	Cantidad	Especificaciones
Switch	1	Switch de 8 Puertos 10/100 MBPS (RJ-45); estándar IEEE 802.3 10base-t y estándar IEEE 802.3u 100base-tx, con soporte Full-dúplex y Half-dúplex para cada puerto.
Cables	10 m	Cable UTP categoría 5e con conectores RJ45 (cables prefabricados).

Tabla 2.34. Dispositivos de red.

4. **Otros recursos:** Las especificaciones del impresor y UPS que se utilizaron durante el desarrollo del proyecto se muestran en la tabla 2.35.

Recurso	Cantidad	Especificaciones
Impresor	1	Resolución 4,800 dpi, conexión USB, cartuchos de tinta negro y de color, promedio de impresión 7 páginas b/n por minuto y 5 páginas a color por minuto.
UPS	1	Voltaje de entrada/salida 120 V, frecuencia 60 HZ, potencia 750 w, 6 salidas.

Tabla 2.35. Otros recursos de desarrollo.

Todos los componentes anteriormente descritos fueron utilizados en la red de comunicación en la que se realizaron las pruebas del sistema. En la figura 2.1 se muestra el diagrama de red de desarrollo.

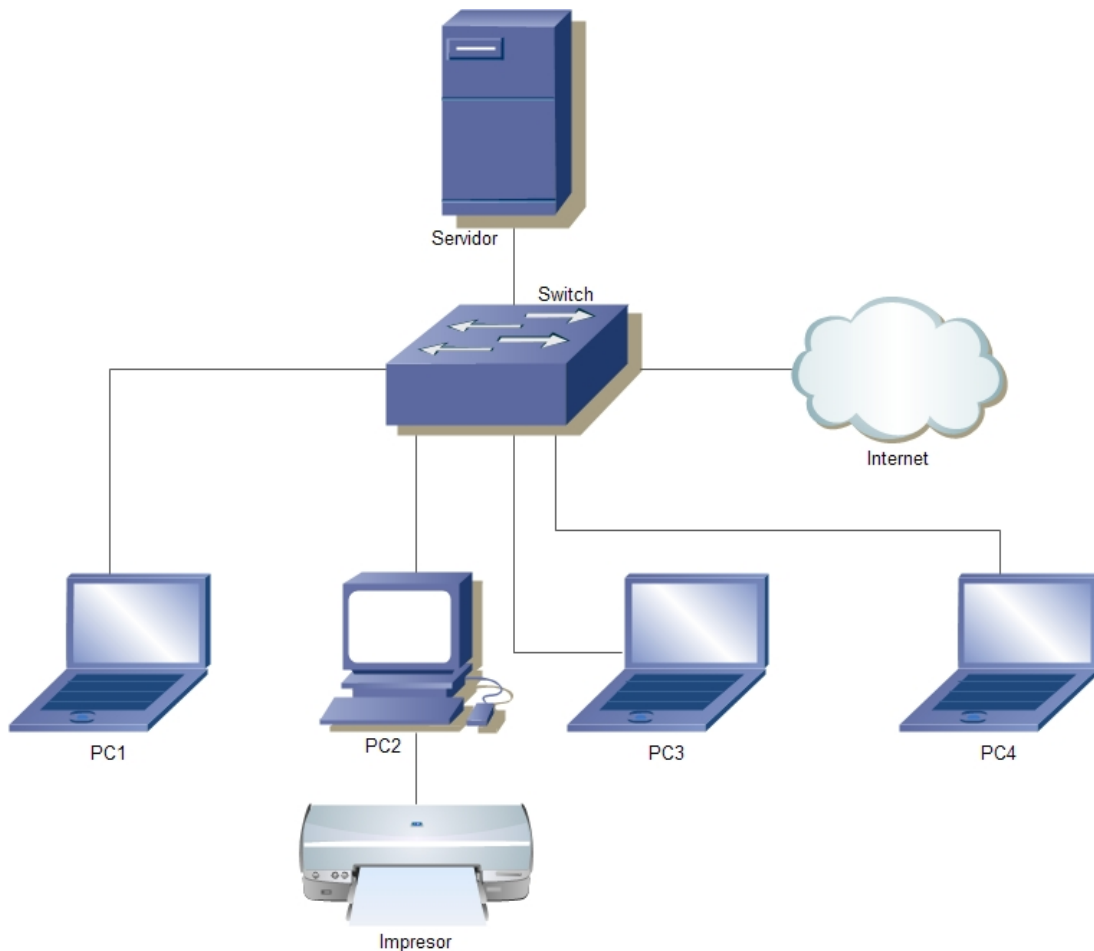


Figura 2.1. Diagrama de red de desarrollo.

2.4.2. RECURSO DE SOFTWARE DE DESARROLLO

Este apartado trata sobre la descripción de todo el software requerido para desarrollar el sistema informático. El software está dividido en las siguientes categorías:

1. **Sistemas operativos:** Los sistemas operativos que se utilizaron durante el desarrollo del proyecto se muestran en la tabla 2.36, junto a los requerimientos de hardware mínimos para su funcionamiento.

Nombre	Versión	Descripción	Procesador	Espacio en disco disponible	Memoria RAM	Memoria VRAM
Debian Squeeze	6.0.5 32-bits	Sistema operativo de código abierto orientado a servidores.	1 GHz	5 GB	128 MB	64 MB
Windows 7 Profesional	7.6.0.1 32-bits	Sistema operativo propietario de fácil manejo para los usuarios.	1 GHz	16 GB	1 GB	128 MB

Tabla 2.36. Sistemas operativos para el desarrollo del proyecto.

2. **Servidor:** Los detalles técnicos de este recurso puede observarse en las tablas 1.4 y 1.5 del apartado 1.4.1. Factibilidad Técnica.
3. **Herramientas de programación:** Los detalles técnicos de estos recursos pueden observarse en las tablas 1.4 y 1.5 del apartado 1.4.1. Factibilidad Técnica.

Además como parte de la plataforma de desarrollo se utilizaron los siguientes frameworks y librerías para facilitar el desarrollo:

- **JavaServer Faces:** También conocido como JSF es una tecnología y framework para aplicaciones Java basadas en Web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java.
 - **JasperReports:** Es una herramienta de creación de informes que tiene la habilidad de entregar contenido enriquecido al monitor, a la impresora o a archivos PDF, HTML, XLS, CSV y XML.
 - **PrimeFaces e Hibernate:** Estos recursos pueden verse las tablas 1.4 y 1.5 del apartado 1.4.1. Factibilidad Técnica.
4. **Gestor de base de datos:** Los detalles técnicos de este recurso puede observarse en las tablas 1.4 y 1.5 del apartado 1.4.1. Factibilidad Técnica.

5. **Herramientas de modelado:** Las herramientas que se utilizaron para la creación de modelos del sistema informático se muestran en la tabla 2.37.

Nombre	Versión	Descripción	Procesador	Espacio en disco disponible	Memoria RAM	Resolución de video
Sybase PowerDesigner	16.1	Herramienta de modelado de datos.	1.5 GHz	500 MB	1 GB	800 x 600
Edraw Max	6.3	Herramienta para diseño de diagramas.	750 MHz	20 MB	256 MB	1024 x 768
Bizagi Process Modeler	2.3	Herramienta para diagramar procesos usando la notación estándar BPMN	1 GHz	50 MB	1 GB	800 x 600

Tabla 2.37. Herramienta de modelado para el desarrollo del proyecto.

6. **Herramientas utilitarias:** En la tabla 2.38 se muestran las herramientas utilizadas en el desarrollo del proyecto para la creación de documentos y otros contenidos multimedia.

Nombre	Versión	Descripción	Procesador	Espacio en disco disponible	Memoria RAM	Resolución de video
Microsoft Office 2007	12.0	Grupo de herramientas para edición de documentos.	500 MHz	1.5 GB	256 MB	1024 x 768
Snagit	10.0	Software para capturas de pantalla.	2.4 GHz	125 MB	1 GB	800 x 600

Tabla 2.38. Herramientas ofimáticas para el desarrollo del proyecto.

7. **Herramientas colaborativas en línea:** Este proyecto se realizó utilizando herramientas colaborativas en línea mostradas en la tabla 2.39.

Nombre	Licencia	Sitio	Descripción
Cacoo	Freeware	https://cacoo.com/	Herramienta que permite la creación de diagramas con otros usuarios.
GoogleDocs	Freeware	https://docs.google.com/	Servicio que permite la creación de documentos en línea.
Dropbox	Freeware	https://www.dropbox.com/	Servicio que permite almacenar archivos en línea.
Skype	Freeware	http://www.skype.com/	Software que permite comunicaciones de texto, voz y vídeo sobre Internet

Tabla 2.39. Herramientas colaborativas para el desarrollo del proyecto.

8. **Herramientas de control de versiones:** durante la etapa de codificación del sistema, se utilizaron las herramientas mostradas en la tabla 2.40.

Nombre	Versión	Descripción
CollabNet Subversion Client	1.5	Sistema de control de versiones que permite ser usado por personas que se encuentran en distintas computadoras e integrar todo el trabajo en un solo repositorio.
Google Code	-	Sitio de Google para desarrolladores que sirve de repositorio de código.

Tabla 2.40. Herramientas de control de versiones.

El repositorio del sistema SIZOO puede consultarse al visitar la siguiente dirección URL:
<http://code.google.com/p/sizoo-ues/>

2.4.3. RECURSO HUMANO DE DESARROLLO

El recurso humano de desarrollo es el personal necesario con el que se cuenta para llevar a cabo el desarrollo del sistema informático.

El recurso humano del que se dispuso durante esta etapa fue el siguiente:

1. **Coordinador de proyecto:** En la tabla 2.41 se muestra el perfil del coordinador de proyecto.

Característica	Descripción
Nombre del perfil	Coordinador de proyecto.
Cantidad de personas	1
Descripción del perfil	Asumirá la coordinación del proyecto, lo cual incluye la asignación de actividades a los miembros del grupo de trabajo, la supervisión de los avances del proyecto, la representación del grupo de trabajo frente a los usuarios, el cumplimiento de metas del proyecto, la evaluación de riesgos y será el responsable del desarrollo del sistema informático, así como de las pruebas de la solución.
Responsabilidades	<p>Planificación y organización de actividades y recursos.</p> <p>Control y evaluación de alternativas para la gestión de riesgos.</p> <p>Planificar, coordinar y dirigir las actividades del desarrollo del proyecto.</p> <p>Coordinar reuniones de trabajo con los usuarios.</p> <p>Administrar correctamente los recursos del proyecto.</p> <p>Monitorear la validez de la información para la creación del sistema.</p> <p>Realizar las mismas funciones de los analistas programadores.</p>
Requisitos	<p>Egresado de la carrera de Ingeniería de Sistemas Informáticos.</p> <p>Conocimientos avanzados en lenguaje SQL.</p> <p>Capacidad para realizar planificación, gestión y control del proyecto.</p> <p>Capacidad para realizar presupuestos.</p> <p>Capacidad de organización de actividades y recursos.</p> <p>Capacidad de análisis y diseño de sistemas informáticos.</p> <p>Encargado del control de la calidad del desarrollo del proyecto.</p> <p>Habilidad para asimilar y transmitir conocimientos.</p> <p>Conocimientos de técnicas de programación.</p> <p>Conocimientos de técnicas de diseño de sistemas.</p> <p>Conocimientos en el proceso de evaluación y supervisión de las pruebas.</p> <p>Habilidad para trabajo en equipo.</p> <p>Capacidad para trabajar bajo presión.</p> <p>Dinámico.</p> <p>Liderazgo.</p>

Tabla 2.41. Perfil del coordinador de proyecto.

2. **Analista programador:** El perfil que debe tener el analista programador se muestra en la tabla 2.42.

Característica	Descripción
Nombre del perfil	Analista programador.
Cantidad de personas	3
Descripción del perfil	Asumirá la responsabilidad del análisis, diseño, codificación y construcción del sistema informático, también tendrá a su cargo la realización de pruebas de la solución.
Responsabilidades	Análisis y diseño del sistema. Codificación de la solución. Ejecución de pruebas del funcionamiento correcto del software. Realizar la documentación del sistema.
Requisitos	Egresado de la carrera de Ingeniería de Sistemas Informáticos. Capacidad de análisis y diseño de sistemas informáticos. Conocimientos de lenguaje SQL. Conocimientos de Java y PostgreSQL. Conocimientos de Bases de Datos Relacionales. Conocimientos de patrones de diseño. Conocimientos de técnicas de programación. Conocimientos de técnicas de diseño de sistemas. Conocimientos de programación orientada a objetos. Habilidad para resolver problemas. Habilidad para trabajo en equipo. Capacidad para trabajar bajo presión.

Tabla 2.42. Perfil de analistas programadores.

3. **Usuario de negocios:** El equipo de trabajo incluye a los usuarios quienes solicitaron el sistema informático. Estos incluyen a personal de la Gerencia de Informática y Sistemas de la Secretaría de Cultura de la Presidencia, y personal del área de zoología del MUHNES. En la tabla 2.43 se muestra las funciones que deben tener dichos usuarios.

Característica	Descripción
Nombre del perfil	Usuario de negocios.
Cantidad de personas	2 unidades administrativas.
Descripción del perfil	Asumirá la responsabilidad de brindar información oportuna para la continuidad de proyecto, así como el monitoreo de avances y brindar las recomendaciones oportunas en caso de ser requerido.
Responsabilidades	Reuniones periódicas con los demás miembros del grupo de trabajo. Aportación de ideas. Brindar recomendaciones. Monitoreo de avances.

Tabla 2.43. Perfil de usuarios de negocios.

En la tabla 2.44 se puede observar la distribución del recurso humano durante las etapas de desarrollo del proyecto, teniendo en cuenta que las horas a trabajar se han calculado al multiplicar los días utilizados para cada etapa por las 4 horas que cada uno de los miembros del grupo de trabajo (coordinador y analistas) dedicará a este proyecto.

Etapas	Horas a trabajar	No. de personas	Total horas-persona	Días utilizados	Porcentaje del tiempo
Fase de Iniciación	320	4	4	20	15%
Fase de Elaboración	864	4	4	54	40%
Fase de Construcción	752	4	4	47	35%
Fase de Transición	224	4	4	14	10%

Tabla 2.44. Distribución del recurso humano.

2.5. REQUERIMIENTOS DE IMPLEMENTACIÓN

Para la implementación del sistema se requiere de ciertos recursos, los cuales deben estar presentes para el correcto funcionamiento del sistema. Entre estos recursos tenemos:

- Recurso de hardware
- Recurso de software
- Recurso humano

2.5.1. RECURSO DE HARDWARE DE IMPLEMENTACIÓN

En éste apartado se describen las características del hardware en el que será puesto en operación el sistema informático. Estos recursos son:

1. **Servidor:** Los detalles técnicos de este recurso puede observarse en la tabla 1.8 del apartado 1.4.1. Factibilidad Técnica.
2. **Estaciones de trabajo:** Los detalles técnicos de estos recursos pueden observarse en la tabla 1.9 del apartado 1.4.1. Factibilidad Técnica.
3. **Otros dispositivos:** Entre los dispositivos adicionales que se necesitan para el pleno funcionamiento del sistema informático se muestran en la tabla 2.45.

Recurso	Descripción
Impresor	Necesario para la impresión de reportes y otra información que interese.
Sistema de energía alterno	Garantiza el funcionamiento del servidor en caso de que el suministro de energía falle.
UPS	Protege a las estaciones de trabajo en caso de que se interrumpa el suministro de energía.
Conexiones de red	Este incluye la infraestructura de red y las conexiones de las estaciones de trabajo.

Tabla 2.45. Otros dispositivos necesarios para la implementación del sistema.

Todo el hardware descrito anteriormente tendrá la configuración de red que se presenta en la figura 2.2.



Figura 2.2. Diagrama de red del sistema informático en funcionamiento.

El ancho de banda necesario para el óptimo funcionamiento del sistema se calcula mediante la siguiente fórmula:

$$\text{Ancho de banda necesario} = (\text{número de visitas} \div \text{tiempo en segundos}) \times (\text{tamaño de página} \times 10) \times 3^1$$

El cálculo del ancho de banda necesario hará uso de los siguientes supuestos:

- Se considerará para el primer año de funcionamiento del sistema un número de visitas diarias de 100.
- Un día está conformado por 24 horas, lo que equivale a 86,400 segundos.
- El tamaño de cada página web que conformará el sistema tiene un tamaño de 200 KB, incluyendo fotografías.
- El número 10 en la fórmula convierte los Bytes a bits.
- El número 3 en la fórmula es un factor que indica el máximo número de visitas que se puede tener en un día.

Luego de estos supuestos, se procede al cálculo del ancho de banda requerido sustituyendo los valores considerados:

$$\text{Ancho de banda necesario} = (100 \div 86,400) \times (200 \text{ KB} \times 10) \times 3$$

$$\text{Ancho de banda necesario} = 0.0012 \text{ segundos} \times 2000 \text{ Kb} \times 3$$

$$\text{Ancho de banda necesario} = 2.4 \text{ Kbps} \times 3$$

$$\text{Ancho de banda necesario} = 7.2 \text{ Kbps}$$

¹ <http://www.signt.net/desarrollo-web/como-calculiar-el-ancho-de-banda-necesario.html>

El ancho de banda calculado es de 7.2 Kbps. Ya que se trabajó con supuestos, es decir que pueden ser valores mayores, y teniendo en cuenta los otros sistemas que se encuentran brindando servicio en el servidor, lo recomendado al menos para el primer año de funcionamiento es de 1 Mbps.

Luego del primer año de funcionamiento, es recomendable hacer el análisis respectivo con valores estadísticos del número de visitas (por ejemplo con un contador de visitas diarias), para determinar si se necesita mayor ancho de banda o se mantiene el actual recomendado.

2.5.2. RECURSO DE SOFTWARE DE IMPLEMENTACIÓN

El software requerido para el pleno funcionamiento del sistema informático está dividido en 2 categorías, las cuales son:

1. **Servidor:** El software necesario en el servidor se presenta en la tabla 2.46.

Software	Nombre	Versión
Sistema operativo	Debian Squeeze	6.0.5; 64 bits
Sistema gestor de base de datos	PostgreSQL	9.1.3
Máquina virtual de aplicación	Java SE	6
Servidor de aplicación	GlassFish	3.1.2

Tabla 2.46. Software necesario en el servidor.

2. **Estaciones de trabajo:** El software necesario en las estaciones de trabajo se presenta en la tabla 2.47.

Software	Nombre	Versión/Edición
Sistemas operativos	Windows XP/Vista/Seven	32-64 bits
	Linux Debian/Ubuntu	32-64 bits
Navegadores	FireFox	3.6 o superior
	Google Chrome	19.0 o superior
	Internet Explorer	7.0 o superior
Máquina virtual de aplicación	Java SE	6 0 superior
Herramientas ofimáticas	Microsoft Office	2003 o superior
	Open Office	3.3 o superior

Tabla 2.47. Software necesario en las estaciones de trabajo.

2.5.3. RECURSO HUMANO DE IMPLEMENTACIÓN

El recurso humano que pondrá en marcha el sistema informático se muestra en la tabla 2.48.

Característica	Descripción
Nombre del perfil	Ingeniero de sistemas informáticos.
Cantidad de personas	4
Descripción del perfil	Asumirá la responsabilidad de implementar el sistema informático, así como garantizar el pleno funcionamiento del mismo.
Responsabilidades	Realizar copias de seguridad de la base de datos. Brindar mantenimiento preventivo y correctivo al hardware. Brindar asistencia técnica a los usuarios del sistema. Verificar el correcto funcionamiento del sistema.
Requisitos	Graduado de la carrera de Ingeniería de Sistemas o carreras afines. Capacidad de análisis y diseño de sistemas informáticos. Conocimientos avanzados de lenguaje SQL y Java. Conocimientos avanzados de Bases de Datos Relacionales. Conocimientos avanzados de técnicas de programación. Habilidad para asimilar y transmitir conocimientos. Facilidad de asimilación de conocimientos sobre nuevas tecnologías. Habilidad para resolver problemas. Habilidad para trabajo en equipo. Capacidad para trabajar bajo presión.

Tabla 2.48. Perfil de ingenieros de sistemas informáticos.

CAPÍTULO 3:

ANÁLISIS Y DISEÑO

El objetivo del análisis es traducir los requerimientos a una especificación que describe cómo construir el sistema. El análisis consiste en obtener una visión del sistema que se preocupa de ver que hace, de modo que sólo se interesa por los requerimientos funcionales. Este flujo de trabajo incluye el uso de elementos del análisis orientado a objetos que permita comprender lo que debe de hacer el sistema informático a desarrollar.

3.1. ARQUITECTURA DEL SISTEMA INFORMÁTICO

La arquitectura del sistema informático trabaja utilizando el patrón MVC (modelo, vista controlador), en el cual se separan los datos, la interfaz de usuario y la lógica interna del sistema.

La vista es la página web, el modelo es el sistema de gestión de la base de datos y la lógica interna del sistema y el controlador es el encargado de recibir los eventos del sistema y darles una solución. Una representación del patrón MVC se muestra en la figura 3.1.

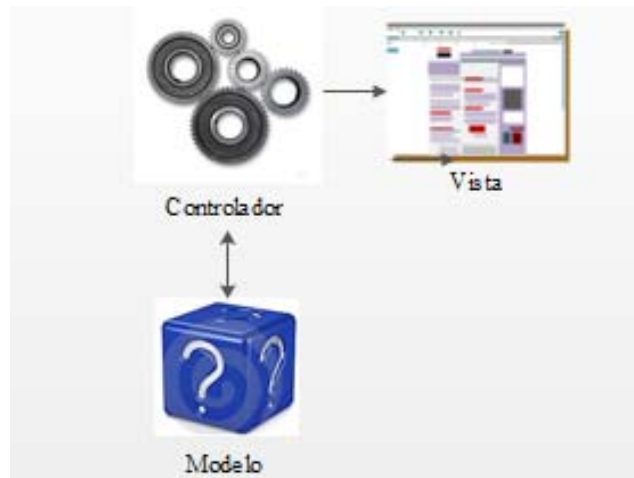


Figura 3.1. Modelo vista controlador.

Donde:

- **Modelo:** Es la representación de la información del sistema informático, el cual trabaja junto a la vista para mostrar la información al usuario y es accedido por el controlador para agregar, consultar, modificar y eliminar datos.
- **Vista:** Esta se presenta al usuario de una manera en que el usuario pueda interactuar con el de la mejor manera posible.
- **Controlador:** Recibe, trabaja y responde a los eventos enviados por el usuario o por la misma aplicación, este interactúa con el modelo y con la vista.

El usuario por medio de un navegador web interactúa con el sistema, haciendo solicitudes al sistema por medio de la vista activando un evento en la interfaz por medio de un botón, enlace, etc. Este evento es interceptado y trabajado por el controlador para producir un resultado utilizando la capa modelo lograr este objetivo, consultando la información que este contiene, como se muestra en la figura 3.2.

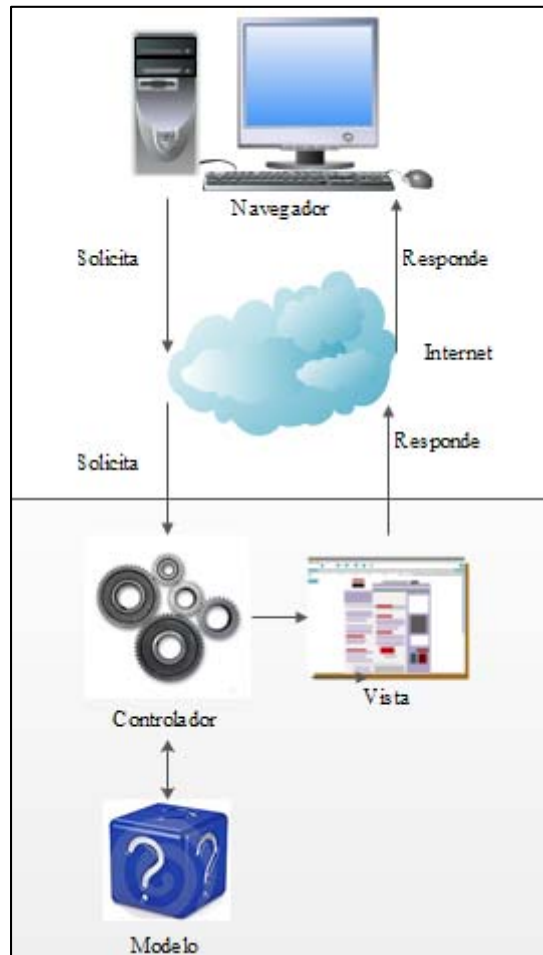


Figura 3.2. Interacción del patrón MVC.

Al igual que la generación del evento, el resultado también es presentado al usuario del sistema por medio de la vista. Luego de presentar el resultado, la vista queda en espera de un nuevo evento que se genere por parte del usuario.

Para llevar a cabo este patrón, es necesaria la ayuda de un servidor de aplicaciones. Un servidor de aplicaciones es un producto que se encuentra en la capa intermedia de la arquitectura, este se encarga de manejar la mayoría de las transacciones relacionadas con la lógica y el acceso a datos de la aplicación. El servidor de aplicaciones toma el evento gestionado por el controlador y se encarga del acceso a la base de datos para obtener o agregar la información para luego comunicar esto al controlador para convertirlo en una salida.

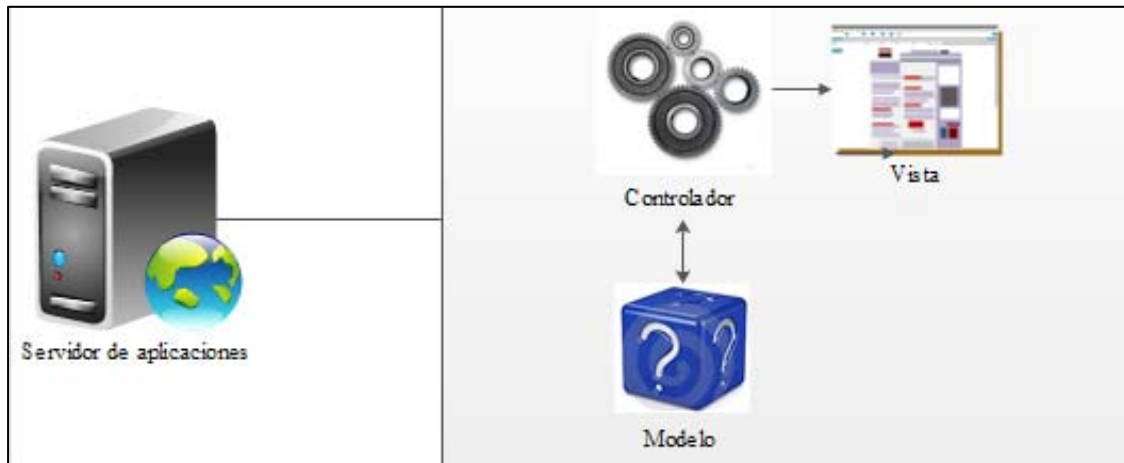


Figura 3.3. Servidor de aplicaciones.

El servidor de aplicaciones es el corazón del sistema ya que gracias a este es que el patrón MVC funciona. El servidor de aplicaciones más que un computador es un software que proporciona servicios a los clientes administrando el contenido utilizando procesos de autenticación de usuarios y utilizando reglas de autorización para decidir si un usuario previamente autenticado goza de privilegios suficientes para acceder a un recurso. Este servidor no necesita ser programado para establecer los permisos y los tipos de servicios que este ofrecerá al cliente.

Para acceder a la base de datos, se usa el patrón DAO (Objeto de Acceso a Datos por sus siglas en español) el cual utiliza componentes que permiten una comunicación persistente con la base de datos relacional, este patrón separa la capa lógica de negocios de la capa de persistencia, escondiendo la implementación del acceso a los datos de los clientes. El patrón DAO permite poder cambiar entre diferentes esquemas de acceso a datos sin la lógica de negocio o el cliente.

Por cada tabla de la base de datos, existe un objeto DAO que interactúa con esta base de datos. Para ingresar o extraer datos, se utilizan los objetos de transferencia de datos DTO (Data Transfer Object) en la cuales se almacenarán temporalmente los datos que se quieran ingresar para luego ser transportados utilizando a los objetos DAO como medio, de igual forma es para la tarea de obtener información de la base de datos, los objetos DTO le piden información a los objetos DAO, los objetos DAO acceden a la base de datos y luego de obtener la información se la transmiten a los objetos DTO para que el controlador produzca los resultados para luego enviarlos a la vista.

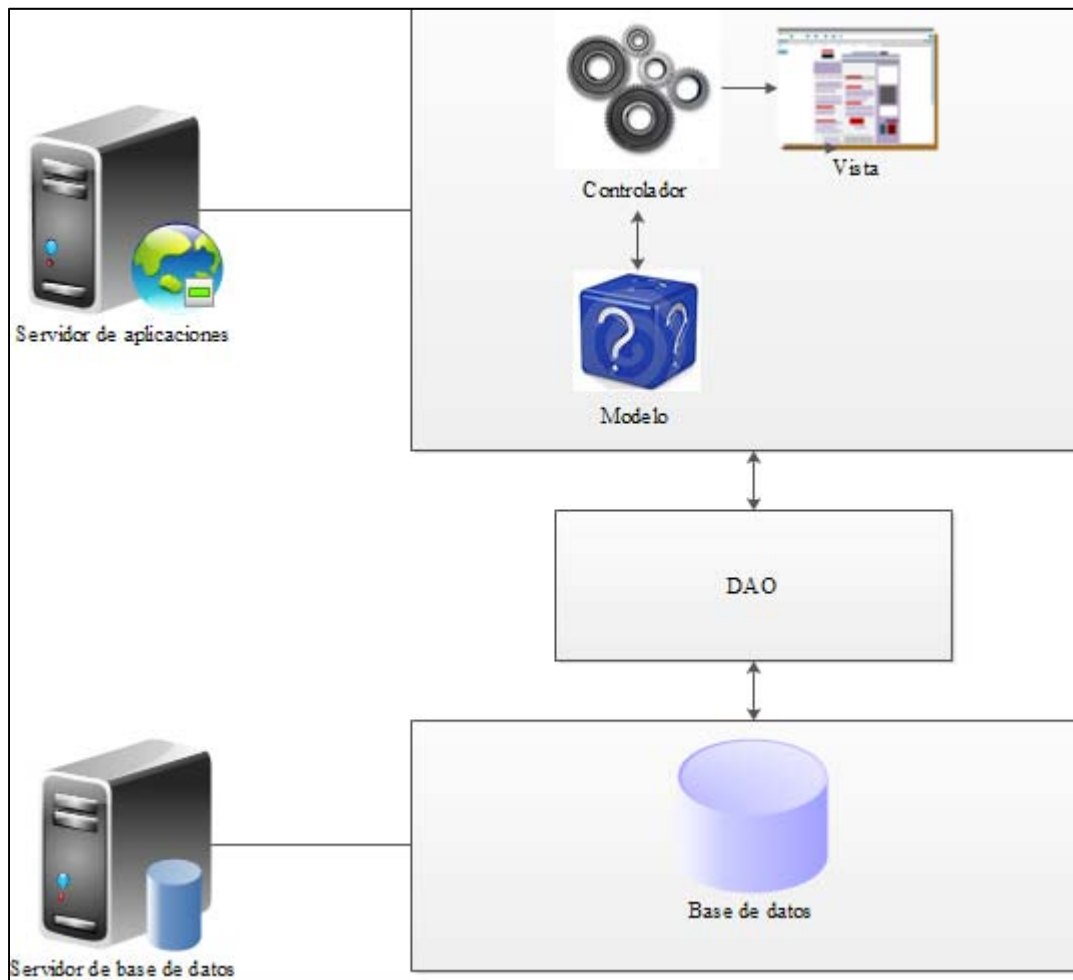


Figura 3.4. Patrón MVC y Patrón DAO

El sistema gestor de base de datos y la base de datos, se encuentran ubicados en un computador servidor cuya función es proveer acceso exclusivo a la base de datos además de brindar servicio de seguridad y protección de la información, manteniendo así un modelo cliente servidor.



Figura 3.5. Servidor de base de datos.

Cuando la capa DAO quiere tener acceso a la base de datos, el sistema gestor de base de datos entra en acción verificando que la información necesaria para los permisos de acceso sea correcta registrando, consultando, modificando o eliminando datos dependiendo de las instrucciones que provengan de la capa DAO.

Al unir todos estos elementos obtenemos la arquitectura que conforma el Sistema Informático para la Clasificación de Zoología de El Salvador, realizando las transacciones de una manera ordenada y con la posibilidad de poder hacer modificaciones a los elementos que conforman cada una de las capas sin necesidad de realizar complejas modificaciones a las demás capas de la arquitectura.

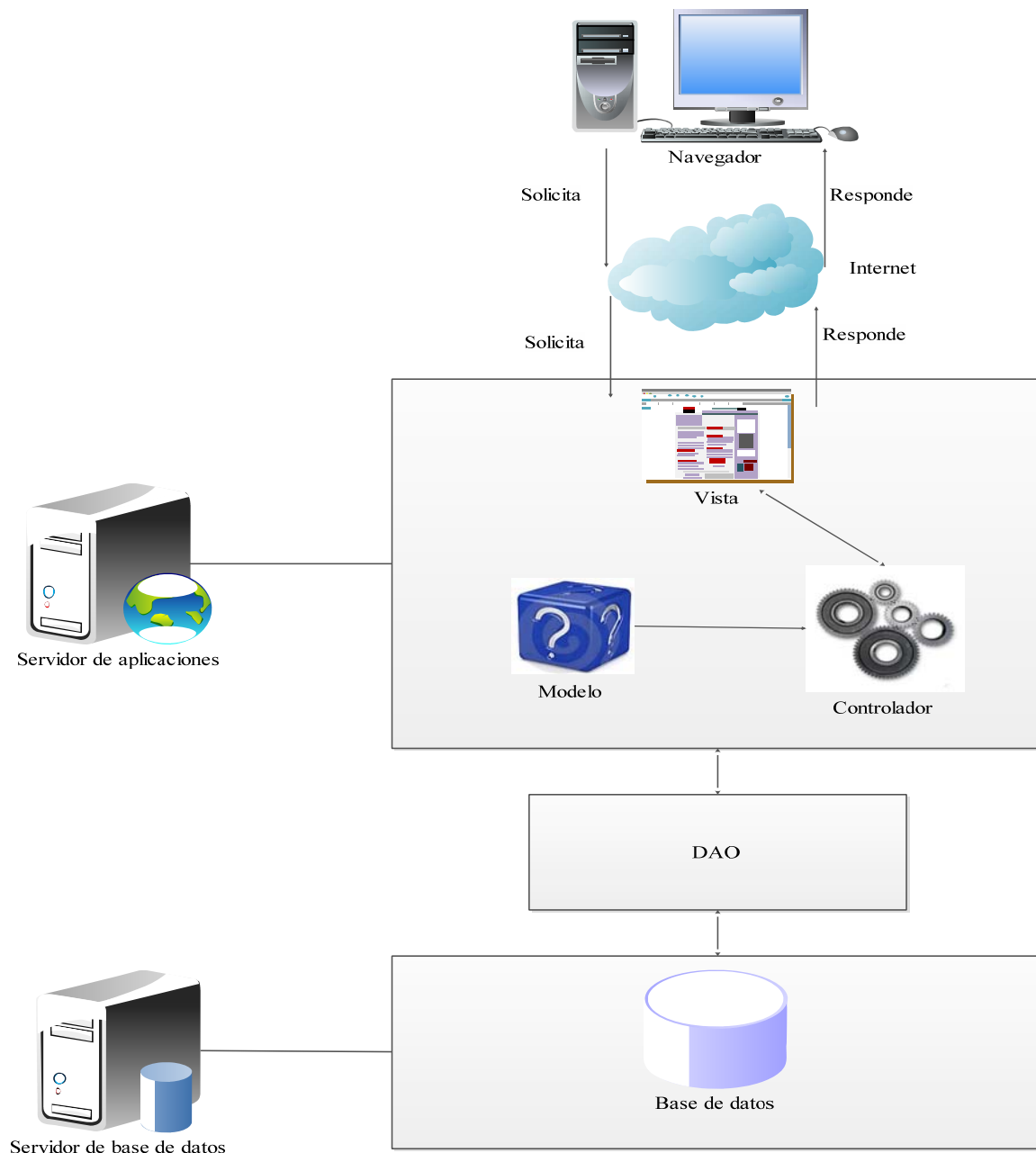


Figura 3.6. Arquitectura del sistema informático.

3.2. CASOS DE USO

En el contexto de ingeniería del software, 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/u otros sistemas.

En el diseño de los casos de uso se utiliza la simbología presentada en la tabla 3.1.

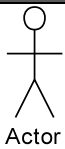
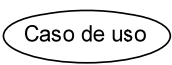
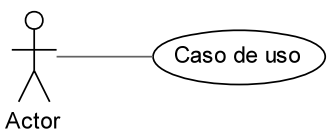
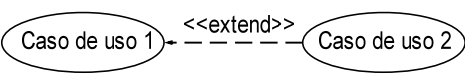
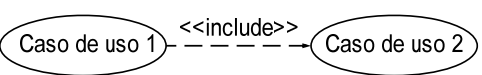
Símbolo	Descripción
 <p>Actor</p>	Representa un tipo de usuario del sistema, se representa por una figura humana. Un usuario es cualquier entidad externa, no necesariamente humana, que interactúa con el sistema.
 <p>Caso de uso</p>	Es una tarea que debe poder llevarse a cabo por medio del sistema que se está desarrollando. Se representa mediante un óvalo.
 <p>Actor</p> <p>Caso de uso</p>	Indica una asociación entre un actor y un caso de uso si el actor interactúa con el sistema para llevar a cabo el caso de uso.
 <p>Caso de uso 1</p> <p><<extend>></p> <p>Caso de uso 2</p>	Indica que al realizar el caso de uso 1 se puede o no ejecutar el caso de uso 2.
 <p>Caso de uso 1</p> <p><<include>></p> <p>Caso de uso 2</p>	Indica que para realizar el caso de uso 1 es necesario ejecutar el caso de uso 2.

Tabla 3.1. Simbología usada para casos de uso.

3.2.1. ACTORES

El formato que describe a los actores que interactúan con el sistema se presenta en la tabla 3.2.

Identificador	<< Identificador del actor >>
Actor	<< Nombre del actor >>
Descripción	<< Una breve descripción del actor >>
Características	<< Características que describen al actor >>
Relaciones	<< Relaciones que posee el actor con otros actores del sistema >>
Referencias	<< Elementos de desarrollo en los que interviene el actor >>
Comentarios	<< Comentarios u observaciones adicionales sobre el actor >>

Tabla 3.2. Formato de descripción de actores.

A continuación se presentan los actores identificados del sistema:

Identificador	U01-ADM
Actor	Administrador
Descripción	Es el que se encargara de manejar, operar, dar mantenimiento al sistema y administrar los privilegios, administrar a los usuarios y eliminar los datos que sean necesarios. Además puede realizar cualquier operación que el sistema necesite.
Características	Debe administrar la base de datos, dar seguridad al sistema, y realizar auditorías del sistema.
Relaciones	Biólogo
Referencias	CU084, CU085, CU086, CU087, CU088, CU089, CU090, CU091, CU092, CU093, CU094, CU095, CU096, CU097, CU098, CU099, CU100, CU101, CU102, CU103, CU104.
Comentarios	Debe de tener conocimientos previos de manejo de sistemas y bases de datos.

Tabla 3.3. Actor Administrador.

Identificador	U01-AUD
Actor	Auditor
Descripción	Es el que se encargará de observar las operaciones que se llevan a cabo dentro del sistema.
Características	Solo tendrá acceso a la bitácora del sistema.
Relaciones	Administrador
Referencias	CU080, CU081, CU082, CU083, CU084, CU100
Comentarios	Debe de tener conocimientos previos de manejo de sistemas y bases de datos.

Tabla 3.4. Actor Auditor.

Identificador	U02-BIO
Actor	Biólogo
Descripción	Es el que se encargará de revisar que la información que se encuentre en la base de datos este correcta así como también de generar los reportes; el biólogo puede ingresar tanto especies, especímenes o información para el portal público. Además, está encargado de autorizar el mantenimiento a las piezas así como la recepción de las mismas luego de finalizado el mantenimiento.
Características	Debe ser el que se encargue de toda la colección del museo, tener conocimientos de zoología y ser el responsable de la información del sistema.
Relaciones	Digitador
Referencias	CU001, CU002, CU003, CU004, CU005, CU006, CU007, CU008, CU009, CU010, CU011, CU012, CU013, CU014, CU015, CU016, CU017, CU018, CU019, CU020, CU022, CU023, CU024, CU025, CU026, CU027, CU028, CU029, CU030, CU031, CU032, CU033, CU034, CU035, CU036, CU037, CU038, CU039, CU040, CU041, CU044, CU043, CU044, CU045, CU046, CU047, CU048, CU049, CU050, CU051, CU052, CU053, CU054, CU055, CU056, CU057, CU058, CU059, CU060, CU061, CU062, CU063, CU064, CU065, CU066, CU067, CU068, CU069, CU070, CU071, CU072, CU073, CU074, CU075, CU076, CU077, CU078, CU079, CU080, CU081, CU082, CU083, CU084, CU103, CU104, CU105, CU106, CU107, CU108.
Comentarios	Las eliminaciones de cualquier tipo solo las puede realizar el administrador del sistema.

Tabla 3.5. Actor Biólogo.

Identificador	U03-DGT
Actor	Digitador
Descripción	Es la persona que se encargara del ingreso de la información es decir de agregar toda la información que este en los libros al sistema así como de los especímenes que ya estén clasificados.
Características	Responsabilidad y cuidado al momento del ingreso de información.
Relaciones	N/A
Referencias	CU037, CU038, CU039, CU042, CU043, CU044, CU045, CU046, CU084
Comentarios	Solo puede acceder a lo referente con el registro de especímenes.

Tabla 3.6. Actor Digitador.

Identificador	U04-INV
Actor	Investigador
Descripción	El investigador puede realizar cualquier tipo de consulta para obtener información de especies o especímenes en general; esta consulta la realizará a través del portal público que tendrá el sistema, donde se podrá visualizar descripciones acerca de los especímenes que se tiene así como realizar consultas avanzadas de interés para investigadores profesionales.
Características	Tener conocimientos mínimos de zoología para poder realizar consultas sobre taxonomía en específico.
Relaciones	N/A
Referencias	CU040, CU047
Comentarios	Las consultas pueden ser avanzadas que indican que ya son con respecto a taxonomía del espécimen o simples descripciones de animales, para realizar este tipo de consultas no es necesario ser usuario del sistema.

Tabla 3.7. Actor Administrador.

3.2.2. LISTA DE CASOS DE USO

El listado de los casos de uso propuestos se presenta en las tablas 3.3.a hasta la 3.3.d.

Identificador	Caso de uso
CU001	Gestionar Taxonomía
CU002	Gestionar phylum
CU003	Agregar phylum
CU004	Modificar phylum
CU005	Consultar phylum
CU006	Eliminar phylum
CU007	Gestionar clase
CU008	Agregar clase
CU009	Modificar clase
CU010	Consultar clase
CU011	Eliminar clase
CU012	Gestionar orden
CU013	Agregar orden
CU014	Modificar orden

3.3.a. Listado de casos de uso.

Identificador	Caso de uso
CU015	Consultar orden
CU016	Eliminar orden
CU017	Gestionar súper-familia
CU018	Agregar súper-familia
CU019	Modificar súper-familia
CU020	Consultar súper-familia
CU021	Eliminar súper-familia
CU022	Gestionar familia
CU023	Agregar familia
CU024	Modificar familia
CU025	Consultar familia
CU026	Eliminar familia
CU027	Gestionar genero
CU028	Agregar genero
CU029	Modificar genero
CU030	Consultar genero
CU031	Eliminar género
CU032	Gestionar especie
CU033	Agregar especie
CU034	Modificar especie
CU035	Consultar especie
CU036	Eliminar especie
CU037	Gestionar animal
CU038	Agregar animal
CU039	Modificar animal
CU040	Consultar animal
CU041	Eliminar animal
CU042	Agregar reseña
CU043	Modificar reseña
CU044	Gestionar espécimen
CU045	Agregar espécimen
CU046	Modificar espécimen
CU047	Consultar espécimen
CU048	Eliminar espécimen
CU049	Gestionar restauración
CU050	Agregar restauración
CU051	Modificar restauración
CU052	Consultar restauración

3.3.b. Listado de casos de uso.

Identificador	Caso de uso
CU053	Gestionar recolector
CU054	Agregar recolector
CU055	Modificar recolector
CU056	Consultar recolector
CU057	Eliminar recolector
CU058	Gestionar recolección
CU059	Agregar recolección
CU060	Modificar recolección
CU061	Consultar recolección
CU062	Eliminar recolección
CU063	Gestionar geografía
CU064	Gestionar departamento
CU065	Agregar departamento
CU066	Modificar departamento
CU067	Consultar departamento
CU068	Eliminar departamento
CU069	Gestionar municipio
CU070	Agregar municipio
CU071	Modificar municipio
CU072	Consultar municipio
CU073	Eliminar municipio
CU074	Gestionar área protegida
CU075	Agregar área protegida
CU076	Modificar área protegida
CU077	Consultar área protegida
CU078	Eliminar área protegida
CU079	Gestionar zona
CU080	Agregar zona
CU081	Modificar zona
CU082	Consultar zona
CU083	Eliminar zona
CU084	Autenticar usuario
CU085	Gestionar usuario
CU086	Agregar usuario
CU087	Modificar usuario
CU088	Consultar usuario
CU089	Eliminar usuario

3.3.c. Listado de casos de uso.

Identificador	Caso de uso
CU090	Gestionar privilegio
CU091	Agregar privilegio
CU092	Modificar privilegio
CU093	Consultar privilegio
CU094	Eliminar privilegio
CU095	Gestionar rol
CU096	Agregar rol
CU097	Modificar rol
CU098	Consultar rol
CU099	Eliminar rol
CU100	Presentar bitácora
CU101	Importar catalogo
CU102	Exportar catalogo
CU103	Gestionar reporte
CU104	Generar reporte dinámico
CU105	Generar ficha de especímenes
CU106	Generar reporte de restauraciones
CU107	Generar reporte de recolecciones
CU108	Generar reportes de usuarios

Tabla 3.3.d. Listado de los casos de uso.

Para describir cada uno de los casos de uso propuestos se usará el formato descrito en la tabla 3.4.

Identificador	<< Identificador del caso de uso >>		
Caso de uso	<< Nombre del caso de uso >>		
Actores	<< Actores que participan en el caso de uso >>		
Referencias	<< Casos de uso que tienen relación con este >>		
Precondiciones	<< Condiciones previas a satisfacer para la realización de este caso de uso >>		
Postcondiciones	<< Resultado esperado al realizar este caso de uso >>		
Propósito	<< Objetivo del caso de uso >>		
Descripción	<< Descripción detallada del caso de uso >>		
Flujo básico < Secuencia de pasos del proceso en una situación normal >			
Actor			Sistema
Corr.	<< Pasos del actor >>	Corr.	<< Pasos del sistema >>
Flujo alternativo < Secuencia de pasos del proceso en una situación anormal >			
<<Lista de casos excepcionales >>			
Observaciones	<< Comentarios u observaciones adicionales sobre el caso de uso >>		

Tabla 3.4. Formato de los casos de uso.

3.2.3. DIAGRAMAS DE CASOS DE USO

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario. Por lo tanto los casos de uso determinan los requerimientos funcionales del sistema, es decir, representan las funciones que un sistema puede ejecutar.

El diagrama general de casos de uso del sistema se muestra en la figura 3.7.

SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA ZOOLOGÍA DE EL SALVADOR PARA EL MUHNES

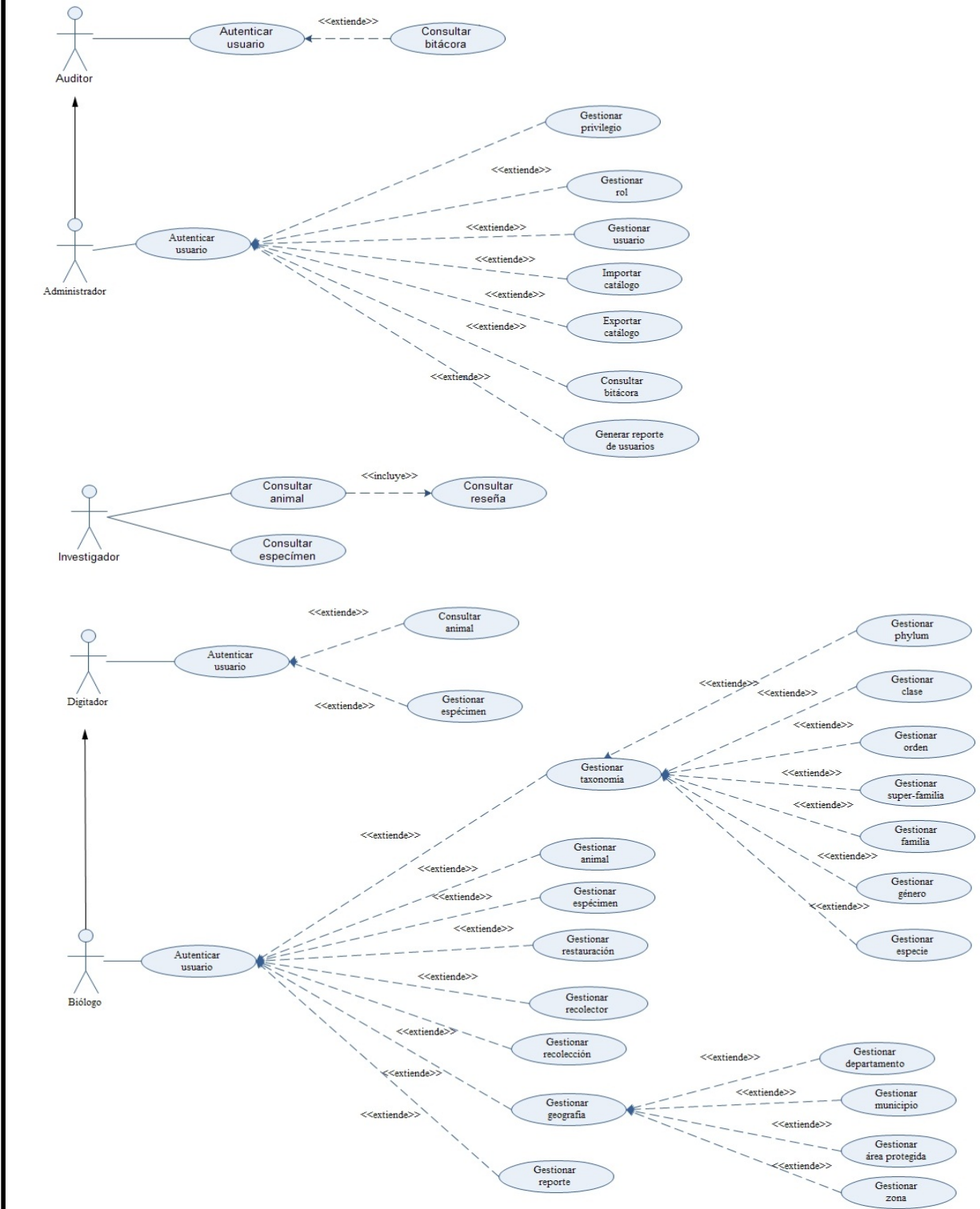


Figura 3.7. Diagrama general de casos de uso del sistema.

Los casos de uso por actor se muestran a continuación.

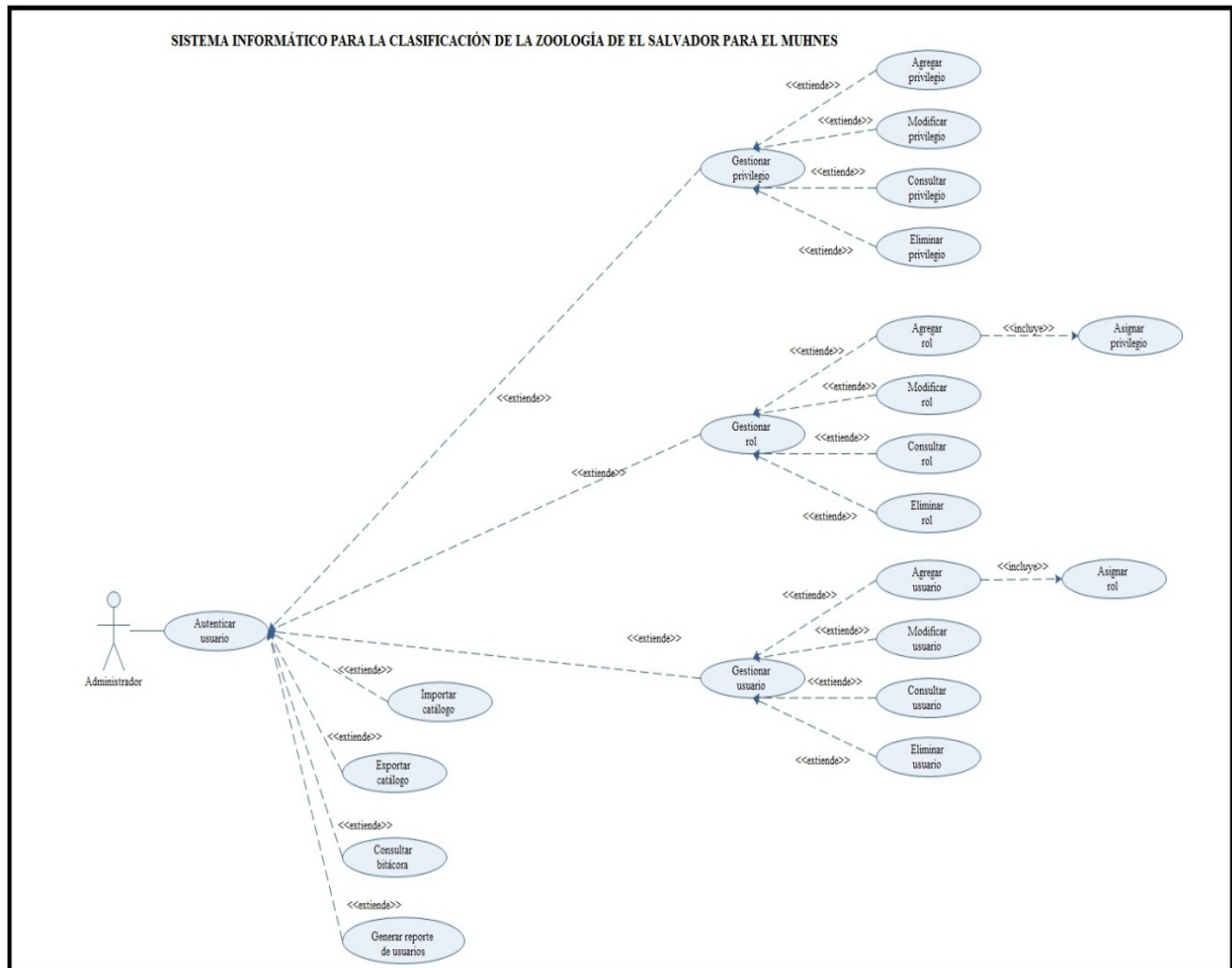


Figura 3.8. Diagrama de caso de uso para el actor Administrador.



Figura 3.9. Diagrama de caso de uso para el actor Auditor.

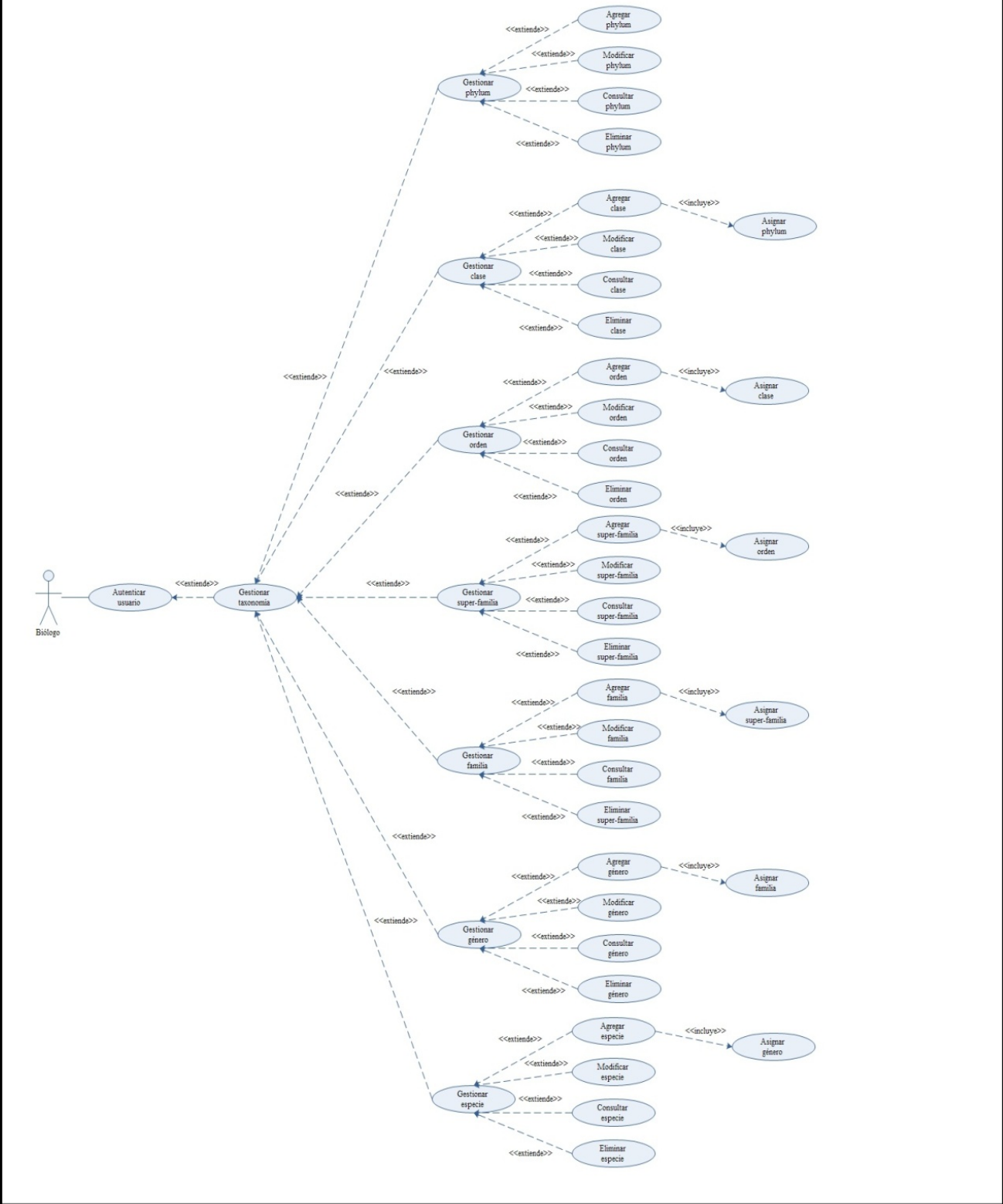


Figura 3.10. Diagrama de caso de uso para el actor Biólogo en Taxonomía.

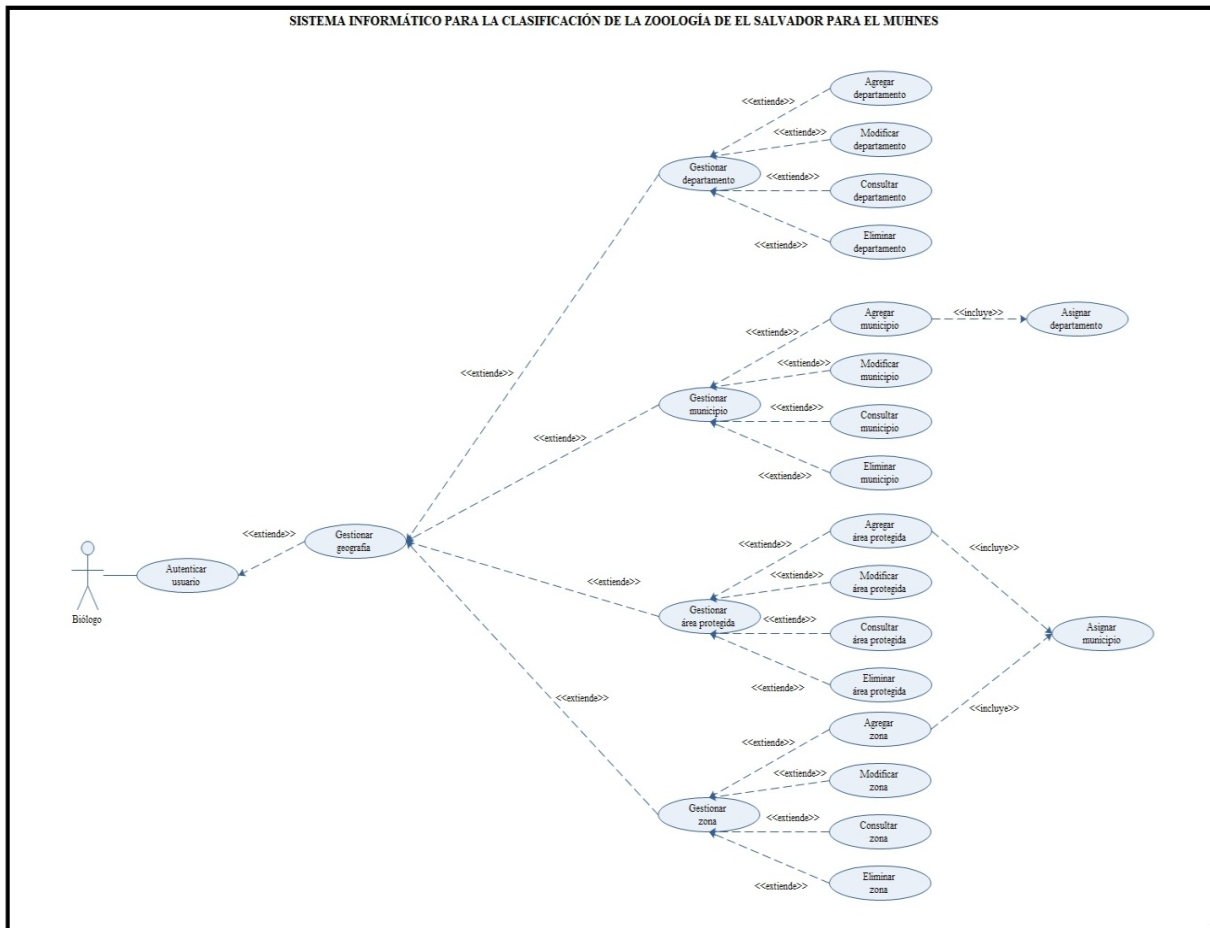


Figura 3.11. Diagrama de caso de uso para el actor Biólogo en Geografía.

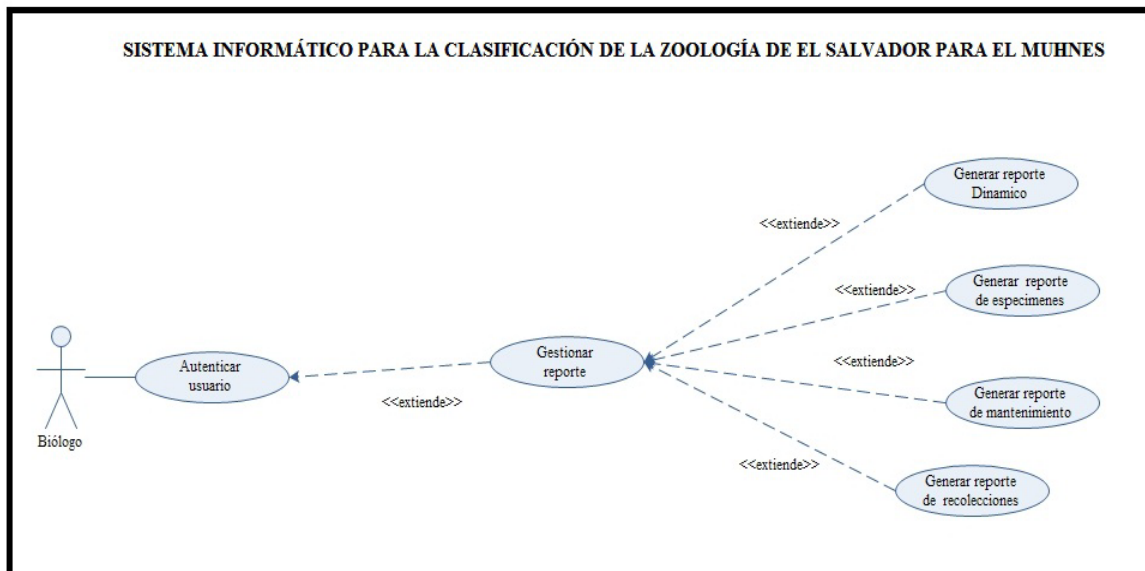


Figura 3.12. Diagrama de caso de uso para el actor Biólogo para la Generación de Reportes.

SISTEMA INFORMÁTICO PARA LA CLASIFICACIÓN DE LA ZOOLOGÍA DE EL SALVADOR PARA EL MUHNES

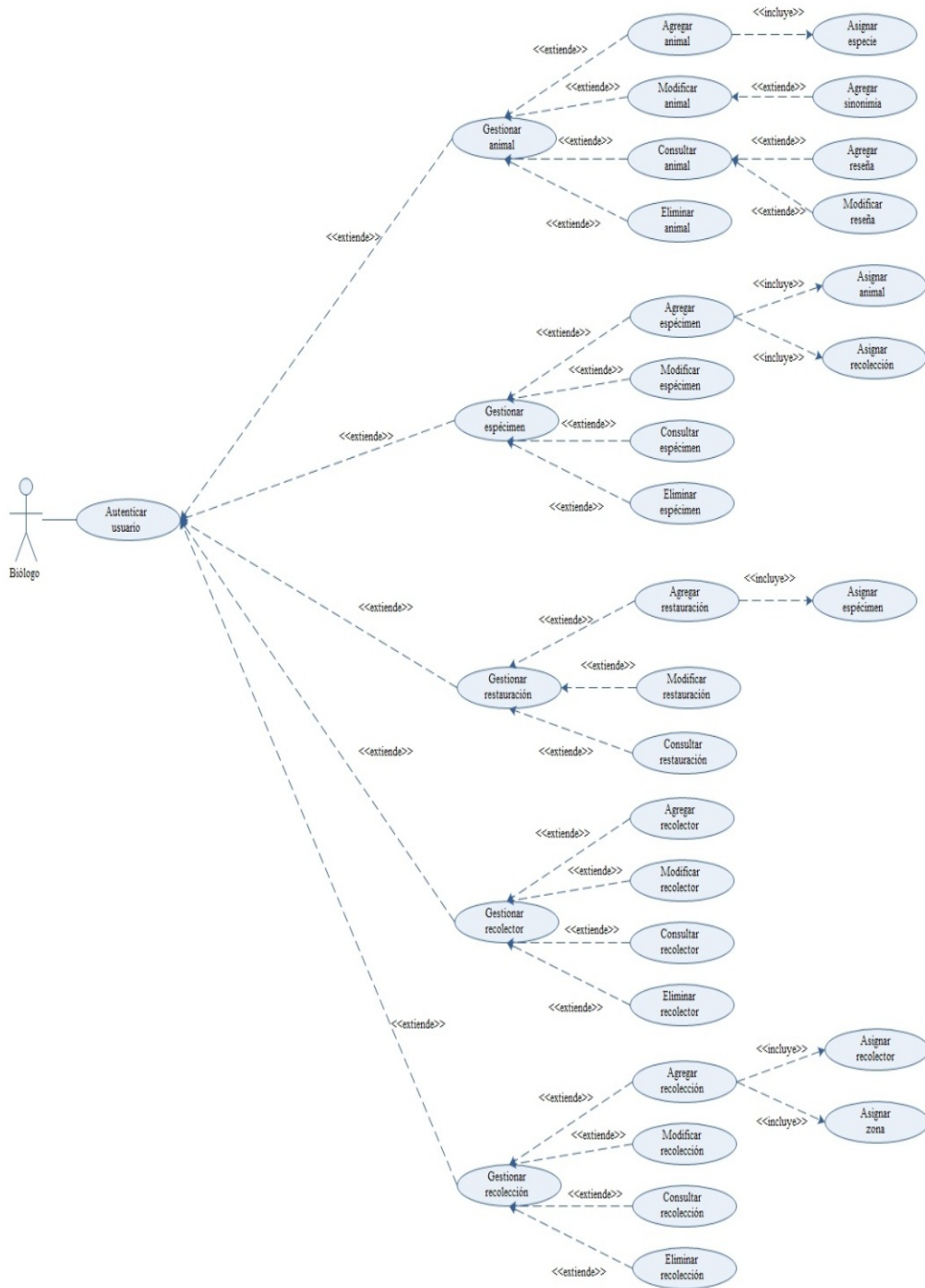


Figura 3.13. Diagrama de caso de uso para el actor Biólogo para especímenes.

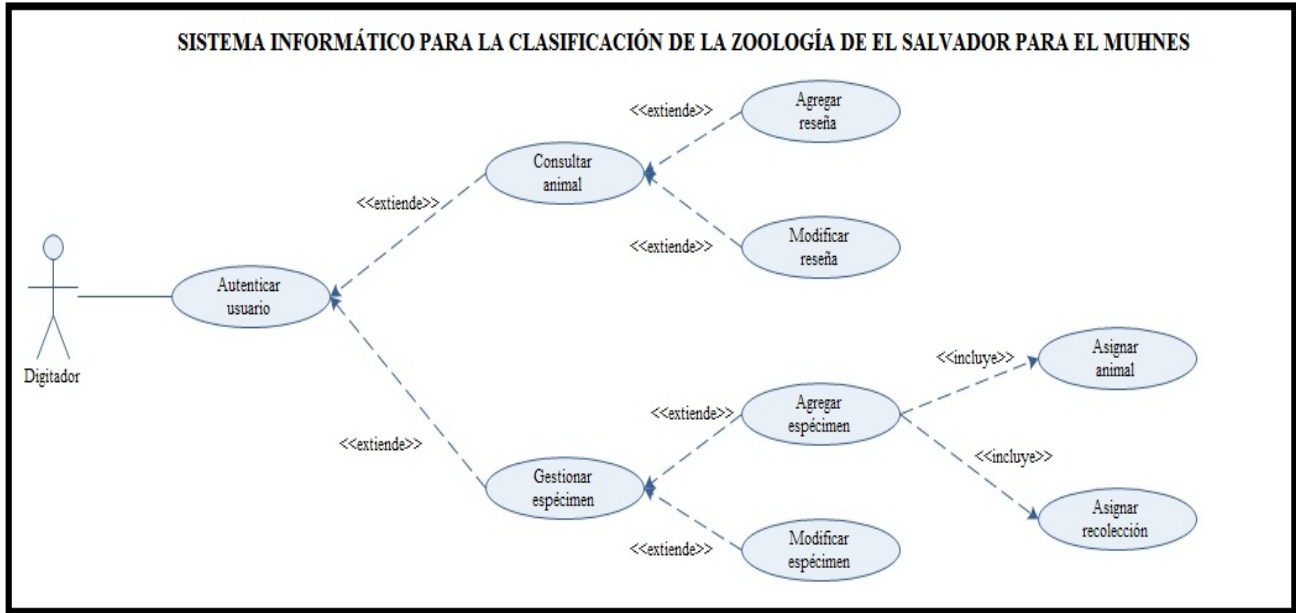


Figura 3.14. Diagrama de caso de uso para el actor Digitador.

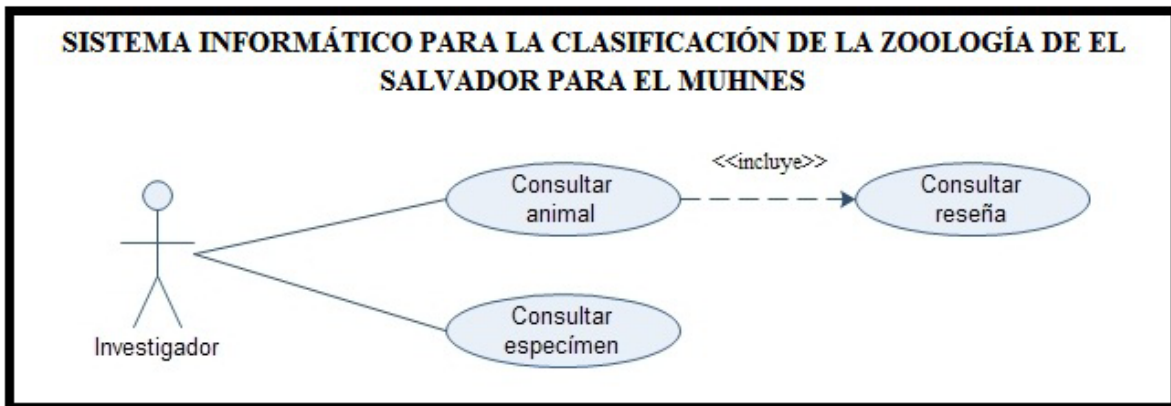


Figura 3.15. Diagrama de caso de uso para el actor Investigador.

3.2.4. DESCRIPCIÓN DE CASOS DE USO

A continuación se procede a la descripción de cada uno de los casos de uso con el formato establecido previamente.

Identificador	CU001		
Caso de uso	Gestionar taxonomía.		
Actor	Biólogo.		
Referencias	RFC001, RFC005 CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir entre gestionar phylum, clase, orden, súper-familia, familia, género o especie.		
Propósito	Dar mantenimiento al catalogo de taxonomía.		
Descripción	El biólogo a cargo elige la opción que quiere ejecutar las cuales son: gestionar phylum, clase, orden, súper-familia, familia, género y especie.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión taxonómica.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Gestionar phylum • Gestionar clase • Gestionar orden • Gestionar súper-familia • Gestionar familia • Gestionar genero • Gestionar especie.
3	Elige una opción.	4	Dirige a la interfaz respectiva.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.5. Caso de uso CU001.

Identificador		CU002	
Caso de uso		Gestionar phylum.	
Actor		Biólogo.	
Referencias		RFC001, RFC005 CU001, CU084	
Precondiciones		Autenticar usuario.	
Postcondiciones		El usuario podrá elegir entre agregar phylum, modificar, consultar y eliminar.	
Propósito		Mantener un catálogo de phylum.	
Descripción		El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar phylum.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión phylum.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar phylum • Modificar phylum • Consultar phylum • Eliminar phylum.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.6. Caso de uso CU002.

Identificador		CU003	
Caso de uso		Agregar phylum.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU002, CU084	
Precondiciones		Autenticar usuario, Gestionar phylum.	
Postcondiciones		Phylum agregado	
Propósito		Agregar un phylum para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos del phylum.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar phylum.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre.
3	Ingresar los datos solicitados.	4	Valida el formato del nombre
		5	Añade el nuevo phylum a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.7. Caso de uso CU003.

Identificador	CU004		
Caso de uso	Modificar phylum.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU002, CU084		
Precondiciones	Autenticar usuario, Gestionar phylum		
Postcondiciones	La Información estará actualizada con las correcciones respectivas.		
Propósito	Actualizar información del phylum.		
Descripción	El usuario elige la opción modificar phylum para actualizar el estado o el nombre.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar phylum.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del phylum en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.8. Caso de uso CU004.

Identificador	CU005		
Caso de uso	Consultar phylum.		
Actores	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU002, CU084		
Precondiciones	Autenticación de usuario, Gestión de phylum.		
Postcondiciones	N/A		
Propósito	Consultar datos de los phylum.		
Descripción	Se utiliza para consultar la información del phylum.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar phylum.	2	Presenta un listado de phylum.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.9. Caso de uso CU005.

Identificador	CU006		
Caso de uso	Eliminar phylum.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU002, CU084		
Precondiciones	Autenticar usuario, Gestionar phylum.		
Postcondiciones	Phylum eliminado.		
Propósito	Eliminar el phylum, dejándolo deshabilitado.		
Descripción	El usuario elige la opción Eliminar phylum, el cual deshabilita el phylum para que no se pueda asociar a clases nuevas pero se mantenga en las clases ya existentes.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar phylum.	2	Cambia el estado de phylum a deshabilitado.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.10. Caso de uso CU006.

Identificador	CU007		
Caso de uso	Gestionar clase.		
Actor	Biólogo.		
Referencias	RFC001, RFC005 CU001, CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir entre agregar, modificar, consultar y eliminar clase.		
Propósito	Mantener un catálogo de clases.		
Descripción	El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar clases.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión clases.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar clase • Modificar clase • Consultar clase • Eliminar clase.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.11. Caso de uso CU007.

Identificador		CU008	
Caso de uso		Agregar clase.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU007, CU084	
Precondiciones		Autenticar usuario, Gestionar clase.	
Postcondiciones		Clase agregada	
Propósito		Agregar una clase para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos de la clase.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar clase.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre. • Súper-clase • Infra-clase
3	Ingresar los datos solicitados.	4	Valida el formato los datos de la clase.
		5	Añade la nueva clase a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.12. Caso de uso CU008.

Identificador		CU009	
Caso de uso		Modificar clase.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU007, CU084	
Precondiciones		Autenticar usuario, Gestionar clase	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la clase.	
Descripción		El usuario elige la opción modificar clase para actualizar el estado, nombre, súper-clase, infra-clase.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar clase.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la clase en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.13. Caso de uso CU009.

Identificador	CU010	
Caso de uso	Consultar clase.	
Actores	Biólogo.	
Referencias	RFC001, RFC005, CU001, CU007, CU084	
Precondiciones	Autenticación de usuario, Gestión de clases.	
Postcondiciones	N/A	
Propósito	Consultar datos de las clases.	
Descripción	Se utiliza para consultar la información de las clases.	
Flujo básico		
Actor		Sistema
1	Ingresar a la interfaz para consultar clases.	2
		Presenta un listado de clases.
Flujo alternativo		
N/A		
Observaciones	N/A	

Tabla 3.14. Caso de uso CU010.

Identificador	CU011	
Caso de uso	Eliminar clase.	
Actor	Biólogo.	
Referencias	RFC001, RFC005, CU001, CU002, CU084	
Precondiciones	Autenticar usuario, Gestionar clase.	
Postcondiciones	Clase eliminada.	
Propósito	Eliminar la clase, dejándola deshabilitada.	
Descripción	El usuario elige la opción Eliminar clase, el cual deshabilita la clase para que no se pueda asociar a nuevas órdenes pero se mantenga en las órdenes ya existentes.	
Flujo básico		
Actor		Sistema
1	El usuario ingresa a la interfaz para eliminar clase.	2
		Cambia el estado de clase a deshabilitado.
Flujo alternativo		
N/A		
Observaciones	N/A	

Tabla 3.15. Caso de uso CU011.

Identificador		CU012	
Caso de uso		Gestionar orden.	
Actor		Biólogo.	
Referencias		RFC001, RFC005 CU001, CU084	
Precondiciones		Autenticar usuario.	
Postcondiciones		El usuario podrá elegir entre agregar, modificar, consultar y eliminar orden.	
Propósito		Mantener un catálogo de orden.	
Descripción		El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar orden.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión orden.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar orden • Modificar orden • Consultar orden • Eliminar orden.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.16. Caso de uso CU012.

Identificador		CU013	
Caso de uso		Agregar orden.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU012, CU084	
Precondiciones		Autenticar usuario, Gestionar orden.	
Postcondiciones		Orden agregada	
Propósito		Agregar una orden para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos de la orden.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar orden.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre. • Sub orden.
3	Ingresar los datos solicitados.	4	Valida el formato los datos de la orden.
		5	Añade la nueva clase a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.17. Caso de uso CU013.

Identificador		CU014	
Caso de uso		Modificar orden.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU012, CU084	
Precondiciones		Autenticar usuario, Gestionar orden	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la orden.	
Descripción		El usuario elige la opción modificar clase para actualizar el estado, nombre, sub-orden.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar orden.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la orden en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.18. Caso de uso CU014.

Identificador		CU015	
Caso de uso		Consultar orden.	
Actores		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU012, CU084	
Precondiciones		Autenticación de usuario, Gestión de órdenes.	
Postcondiciones		N/A	
Propósito		Consultar datos de las órdenes.	
Descripción		Se utiliza para consultar la información de las órdenes.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar órdenes.	2	Presenta un listado de órdenes.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.19. Caso de uso CU015.

Identificador		CU016	
Caso de uso		Eliminar orden.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU012, CU084	
Precondiciones		Autenticar usuario, Gestionar orden.	
Postcondiciones		Orden eliminada.	
Propósito		Eliminar la orden, dejándola deshabilitada.	
Descripción		El usuario elige la opción Eliminar la orden, el cual deshabilita la orden para que no se pueda asociar a nuevas súper-familias pero se mantenga en las súper-familias ya existentes.	
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar orden.	2	Cambia el estado de la orden a deshabilitado.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.20. Caso de uso CU016.

Identificador		CU017	
Caso de uso		Gestionar súper-familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005 CU001, CU084	
Precondiciones		Autenticar usuario.	
Postcondiciones		El usuario podrá elegir entre agregar, modificar, consultar y eliminar súper-familia.	
Propósito		Mantener un catálogo de súper-familia.	
Descripción		El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar súper-familia.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión súper-familia.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar súper-familia • Modificar súper-familia • Consultar súper-familia • Eliminar súper-familia.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.21. Caso de uso CU017.

Identificador		CU018	
Caso de uso		Agregar súper-familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU017, CU084	
Precondiciones		Autenticar usuario, Gestionar súper-familia.	
Postcondiciones		Clase agregada	
Propósito		Agregar una súper-familia para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos de la súper-familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar súper-familia.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> Nombre.
3	Ingresar los datos solicitados.	4	Valida el formato los datos de la súper-familia.
		5	Añade la nueva súper-familia a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.22. Caso de uso CU018.

Identificador		CU019	
Caso de uso		Modificar súper-familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU017, CU084	
Precondiciones		Autenticar usuario, Gestionar súper-familia	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la súper-familia.	
Descripción		El usuario elige la opción modificar súper-familia para actualizar el estado, nombre.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar súper-familia.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la súper-familia en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.23. Caso de uso CU019.

Identificador		CU020	
Caso de uso		Consultar súper-familia.	
Actores		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU017, CU084	
Precondiciones		Autenticación de usuario, Gestión de súper-familia.	
Postcondiciones		N/A	
Propósito		Consultar datos de las súper-familia.	
Descripción		Se utiliza para consultar la información de las súper-familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar súper-familia.	2	Presenta un listado de súper-familia.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.24. Caso de uso CU020.

Identificador		CU021	
Caso de uso		Eliminar súper-familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU017, CU084	
Precondiciones		Autenticar usuario, Gestionar súper-familia.	
Postcondiciones		Súper-familia eliminada.	
Propósito		Eliminar la súper-familia, dejándola deshabilitada.	
Descripción		Deshabilita la súper-familia para que no se pueda asociar a nuevas familias pero se mantenga en las súper-familia ya existentes.	
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar súper-familia.	2	Cambia el estado de súper-familia a deshabilitado.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.25. Caso de uso CU021.

Identificador		CU022	
Caso de uso		Gestionar familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005 CU001, CU084	
Precondiciones		Autenticar usuario.	
Postcondiciones		El usuario podrá elegir entre agregar, modificar, consultar y eliminar familia.	
Propósito		Mantener un catálogo de familia.	
Descripción		El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión familia.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar familia • Modificar familia • Consultar familia • Eliminar familia.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.26. Caso de uso CU022.

Identificador		CU023	
Caso de uso		Agregar familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU022, CU084	
Precondiciones		Autenticar usuario, Gestionar familia.	
Postcondiciones		Familia agregada	
Propósito		Agregar una familia para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos de la familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar familia.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre. • Sub-familia • Tribu
3	Ingresar los datos solicitados.	4	Valida el formato de los datos de la familia.
		5	Añade la nueva familia a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.27. Caso de uso CU023.

Identificador		CU024	
Caso de uso		Modificar familia.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU022, CU084	
Precondiciones		Autenticar usuario, Gestionar familia	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la familia.	
Descripción		El usuario elige la opción modificar familia para actualizar el estado, el nombre, sub-familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar familia.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la familia en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.28. Caso de uso CU024.

Identificador		CU025	
Caso de uso		Consultar familia.	
Actores		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU022, CU084	
Precondiciones		Autenticación de usuario, Gestión de familia.	
Postcondiciones		N/A	
Propósito		Consultar datos de la familia.	
Descripción		Se utiliza para consultar la información de la familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar familia.	2	Presenta un listado de familia.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.29. Caso de uso CU025.

Identificador	CU026		
Caso de uso	Eliminar familia.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU022, CU084		
Precondiciones	Autenticar usuario, Gestionar familia.		
Postcondiciones	Familia eliminada.		
Propósito	Eliminar la familia, dejándolo deshabilitada.		
Descripción	El usuario elige la opción Eliminar familia, el cual deshabilita la familia para que no se pueda asociar a genero nuevos pero se mantenga en los géneros ya existentes.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar familia.	2	Cambia el estado de familia a deshabilitado.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.30. Caso de uso CU026.

Identificador	CU027		
Caso de uso	Gestionar género.		
Actor	Biólogo.		
Referencias	RFC001, RFC005 CU001, CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir entre agregar, modificar, consultar y eliminar género.		
Propósito	Mantener un catálogo de género.		
Descripción	El biólogo a cargo elige la opción que quiere ejecutar las cuales son: agregar, modificar, consultar, eliminar género.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión género.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar género • Modificar género • Consultar género • Eliminar género.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.31. Caso de uso CU027.

Identificador		CU028	
Caso de uso		Agregar género.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU027, CU084	
Precondiciones		Autenticar usuario, Gestionar género.	
Postcondiciones		Género agregada	
Propósito		Agregar un género para dar mantenimiento al catálogo de taxonomía.	
Descripción		El usuario registra los datos del género.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar género.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> Nombre.
3	Ingresar los datos solicitados.	4	Valida el formato del nombre
		5	Añade la nueva familia a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.32. Caso de uso CU028.

Identificador		CU029	
Caso de uso		Modificar género.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU027, CU084	
Precondiciones		Autenticar usuario, Gestionar familia	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la familia.	
Descripción		El usuario elige la opción modificar familia para actualizar el estado, el nombre, sub-familia.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar familia.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la familia en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.33. Caso de uso CU029.

Identificador	CU030		
Caso de uso	Consultar género.		
Actores	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU027, CU084		
Precondiciones	Autenticación de usuario, Gestión de género.		
Postcondiciones	N/A		
Propósito	Consultar datos del género.		
Descripción	Se utiliza para consultar la información del género.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar género.	2	Presenta un listado de género.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.34. Caso de uso CU030.

Identificador	CU031		
Caso de uso	Eliminar género.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU027, CU084		
Precondiciones	Autenticar usuario, Gestionar género.		
Postcondiciones	Género eliminado.		
Propósito	Eliminar el género, dejándolo deshabilitada.		
Descripción	El usuario elige la opción Eliminar género, el cual deshabilita el género para que no se pueda asociar a especies nuevas pero se mantenga en las especies ya existentes.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar especies.	2	Cambia el estado del género a deshabilitado.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.35. Caso de uso CU031.

Identificador	CU032		
Caso de uso	Gestionar especie.		
Actor	Biólogo.		
Referencias	RFC001, RFC005 CU001, CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir entre agregar, modificar, consultar y eliminar especie.		
Propósito	Mantener un catálogo de las especies.		
Descripción	Se les da mantenimiento a las especies porque son parte del catálogo taxonómico que sirve para la clasificación de animales.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestionar especie.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Agregar especie • Modificar especie • Consultar especie • Eliminar especie.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.36. Caso de uso CU032.

Identificador	CU033		
Caso de uso	Agregar especie.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU032, CU084		
Precondiciones	Autenticar usuario, Gestionar especie.		
Postcondiciones	Especie agregada para ser utilizada en la clasificación de animales.		
Propósito	Agregar una especie para dar mantenimiento al catálogo de taxonomía.		
Descripción	El usuario registra los datos de la especie.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar especie.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre. • Sub-especie • Variedad
3	Ingresar los datos solicitados.	4	Valida el formato los datos de la especie.
		5	Añade la nueva especie a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.37. Caso de uso CU033.

Identificador		CU034	
Caso de uso		Modificar especie.	
Actor		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU032, CU084	
Precondiciones		Autenticar usuario, Gestionar especie	
Postcondiciones		La Información estará actualizada con las correcciones respectivas.	
Propósito		Actualizar información de la especie	
Descripción		El usuario puede modificar cualquier dato de la especie.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar especie.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la especie en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.38. Caso de uso CU034.

Identificador		CU035	
Caso de uso		Consultar especie.	
Actores		Biólogo.	
Referencias		RFC001, RFC005, CU001, CU032, CU084	
Precondiciones		Autenticación de usuario, Gestión de especie.	
Postcondiciones		N/A	
Propósito		Consultar datos de las especies.	
Descripción		Se utiliza para consultar la información de las especies.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar especie.	2	Presenta un listado de especies.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.39. Caso de uso CU035.

Identificador	CU036		
Caso de uso	Eliminar especie.		
Actor	Biólogo.		
Referencias	RFC001, RFC005, CU001, CU032, CU084		
Precondiciones	Autenticar usuario, Gestionar especie.		
Postcondiciones	Especie deshabilitada.		
Propósito	Eliminar la especie, dejándola deshabilitada.		
Descripción	El usuario elige la opción eliminar especie, el cual deshabilita la especie para que no se pueda asociar a nuevos animales pero se mantenga en las animales ya existentes.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar especie.	2	Cambia el estado de especie a deshabilitado.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.40. Caso de uso CU036.

Identificador	CU037		
Caso de uso	Gestionar animal.		
Actor	Biólogo.		
Referencias	RFC001, RFC006, CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir si registrar, consultar, modificar o eliminar animal.		
Propósito	Dar mantenimiento a las especies.		
Descripción	El biólogo a cargo elige la opción que quiere ejecutar las cuales son: Registrar animal, Modifica animal, Consultar animal y Eliminar animal.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión animal.	2	Presenta formulario para agregar a un animal.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.41. Caso de uso CU037.

Identificador	CU038		
Caso de uso	Agregar animal.		
Actor	Biólogo.		
Referencias	RFC001, RFC006, CU037, CU084		
Precondiciones	Autenticar usuario, Gestionar animal.		
Postcondiciones	Animal registrado.		
Propósito	Registrar el animal para facilitar el ingreso de especímenes.		
Descripción	El usuario registrará los datos generales, taxonómicos, geográficos y de recolección del animal.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar animal.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre científico • Nombre común • Rango de distribución • Referencia bibliográfica
3	Ingresar los datos solicitados.	4	Valida el formato de los datos.
		5	Añade el nuevo animal a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.42. Caso de uso CU038.

Identificador	CU039		
Caso de uso	Modificar animal.		
Actor	Biólogo.		
Referencias	RFC001, RFC006, CU037, CU084		
Precondiciones	Autenticar usuario, Gestionar animal.		
Postcondiciones	La Información estará actualizada con las correcciones respectivas.		
Propósito	Actualizar información del animal.		
Descripción	El usuario elige la opción Modificar animal para actualizar los datos que se deban cambiar por diferentes motivos, cuando un nombre científico se cambia el valor anterior pasara al dato sinonimia y el nuevo al dato correspondiente.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar animal.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del animal en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.43. Caso de uso CU0039.

Identificador	CU040		
Caso de uso	Consultar animal.		
Actores	Biólogo.		
Referencias	RFC001, RFC006, CU037, CU084		
Precondiciones	Autenticación de usuario, Gestión de animales.		
Postcondiciones	N/A		
Propósito	Consultar datos de los animales.		
Descripción	Se utiliza para consultar la información del animal.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar animal.	2	Presenta un listado de animal.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.44. Caso de uso CU040.

Identificador	CU041		
Caso de uso	Eliminar animal.		
Actor	Biólogo.		
Referencias	RFC001, RFC006, CU037, CU084		
Precondiciones	Autenticar usuario, Gestionar animal.		
Postcondiciones	Animal eliminado.		
Propósito	Mantener solo animales con información correcta.		
Descripción	El usuario elige la opción Eliminar animal en el caso de que la especie haya sido ingresada con características erróneas.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar animal.	2	Verifica que no existan registros vinculados al animal en la base de datos.
		3	Pregunta si en verdad se desea eliminar.
4	Acepta la verificación.	5	Se elimina el animal.
Flujo alternativo			
2a) Existen registros vinculados con el animal: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al animal. • El sistema redirige a la interfaz de gestión de animales. 			
4a) El usuario deniega la eliminación de animal: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de animales. 			
Observaciones	N/A		

Tabla 3.45. Caso de uso CU041.

Identificador	CU042		
Caso de uso	Agregar reseña.		
Actores	Digitador, Biólogo.		
Referencias	RFC001, RFC006, RFC007, CU037, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Información lista para que el público en general la pueda ver.		
Propósito	Agregar información descriptiva de las especies		
Descripción	Se utilizará para actualizar la información que se mostrará a los usuarios del público en general.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de edición de información descriptiva de animales.	2	Presenta formulario para editar la información de la descripción del animal.
3	Agrega el texto.	4	Guarda los cambios.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.46. Caso de uso CU042.

Identificador	CU043		
Caso de uso	Modificar reseña.		
Actores	Digitador, Biólogo.		
Referencias	RFC001, RFC006, RFC007, CU037, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Edición de la información descriptiva con texto enriquecido del animal.		
Propósito	Modificar información descriptiva de las especies		
Descripción	Se utilizará para actualizar la información que se mostrará a los usuarios del público en general.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de edición de información descriptiva de animales.	2	Presenta formulario de edición de texto enriquecido con la información ya ingresada.
3	Edita el texto	4	Guarda los cambios.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.47. Caso de uso CU043.

Identificador	CU044		
Caso de uso	Gestionar espécimen.		
Actor	Biólogo, Administrador.		
Referencias	RFC001, RFC008, CU084		
Precondiciones	Autenticar usuario.		
Postcondiciones	El usuario podrá elegir si agrega, consultar, modificar o eliminar espécimen.		
Propósito	Elegir una de las opciones de agregar, modificar, consultar y eliminar Espécimen.		
Descripción	El usuario elige la opción que quiere ejecutar las cuales son: Agregar Espécimen, Modifica espécimen, Consultar espécimen y Eliminar espécimen.		
Flujo básico			
Actor		Sistema	
1	Ingresas a la interfaz de gestión espécimen.	2	Presenta formulario para agregar a un espécimen y una lista ordenada de los especímenes existentes en el sistema con las opciones para modificarlos o eliminarlos.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.48. Caso de uso CU044.

Identificador	CU045		
Caso de uso	Agregar espécimen.		
Actor	Digitador, Biólogo.		
Referencias	RFC001, RFC008, CU044, CU084		
Precondiciones	Autenticar usuario, Gestionar espécimen.		
Postcondiciones	Especimen ya registrado.		
Propósito	Registrar un espécimen a la colección de zoología.		
Descripción	El usuario es quien ingresa los datos generales del espécimen.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar espécimen.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Código • Animal • Fecha de recolección • Muestra • Determino • Confirno • Ubicación almacén • Hábitat • Conservación • Fotografía • Donación • Rasgos físicos • Datos del hábitat.
3	Ingresa los datos solicitados.	4	Valida el formato de los datos.
		5	Añade el nuevo espécimen a la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.49. Caso de uso CU045.

Identificador		CU046	
Caso de uso		Modificar espécimen.	
Actor		Digitador, Biólogo.	
Referencias		RFC001, RFC008, CU044, CU084	
Precondiciones		Autenticar usuario, Gestionar espécimen.	
Postcondiciones		Especimen con información actualizada.	
Propósito		Actualizar la información del espécimen.	
Descripción		Se da en el caso de que haya cambiado alguno de los datos del espécimen o se necesite modificar información que antes no se tenía agregada para el mismo.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar espécimen.	2	Presenta un formulario para cambiar los campos del registro.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del espécimen en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.50. Caso de uso CU046.

Identificador		CU047	
Caso de uso		Consultar espécimen.	
Actores		Biólogo.	
Referencias		RFC001, RFC008, CU044, CU084	
Precondiciones		Autenticación de usuario, Gestión de especímenes.	
Postcondiciones		N/A	
Propósito		Consultar datos de los especímenes.	
Descripción		Se utiliza para consultar la información del espécimen.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar espécimen.	2	Presenta un listado de espécimen.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.51. Caso de uso CU047.

Identificador		CU048	
Caso de uso		Eliminar espécimen.	
Actor		Biólogo.	
Referencias		RFC001, RFC008, CU044, CU084	
Precondiciones		Autenticar usuario, Gestionar espécimen.	
Postcondiciones		Especimen eliminado.	
Propósito		Eliminar espécimen en caso de que este incorrecto o duplicado.	
Descripción		Solo el administrador del sistema tiene la opción de eliminar especímenes y solo se dará en el caso de que sea necesario.	
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar espécimen.	2	Verifica que no existan registros vinculados al espécimen en la base de datos.
		3	Pregunta si en verdad se desea eliminar el espécimen.
4	Acepta la verificación.	5	Se elimina el espécimen.
Flujo alternativo			
2a) Existen registros vinculados con el espécimen: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al espécimen. • El sistema redirige a la interfaz de gestión de especímenes. 			
4a) El usuario deniega la eliminación de espécimen: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de especímenes. 			
Observaciones	La eliminación será lógica no física.		

Tabla 3.52. Caso de uso CU048.

Identificador		CU049	
Caso de uso		Gestionar restauración.	
Actor		Biólogo, Administrador.	
Referencias		RFC001, RFC012, CU084	
Precondiciones		Autenticar usuario	
Postcondiciones		Se podrá elegir las opciones para gestionar restauración: agregar restauración, modificar restauración, consultar restauración.	
Propósito		Elegir una de las opciones de agrega, consulta o modificar restauración.	
Descripción		El biólogo a cargo elige la opción que quiere ejecutar las cuales son: Agregar, consultar o modificar restauración.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión de restauraciones.	2	Presenta formulario para agregar una restauración.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.53. Caso de uso CU049.

Identificador	CU050		
Caso de uso	Agregar restauración.		
Actor	Biólogo		
Referencias	RFC001, RFC012, CU049, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Control de restauraciones de las piezas.		
Propósito	Agregar restauración de la pieza.		
Descripción	El biólogo a cargo agrega la pieza o conjunto de piezas a las cuales se les brindará la restauración correspondiente.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar restauración.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Quien realiza • Quien autoriza • Fecha de inicio • Observaciones
3	Ingresar los datos solicitados.	4	Valida el formato de los datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.54. Caso de uso CU050.

Identificador		CU051	
Caso de uso		Modificar restauración.	
Actor		Biólogo	
Referencias		RFC001, RFC012, CU049, CU084	
Precondiciones		Autenticar usuario, Gestionar restauración.	
Postcondiciones		Datos de la restauración actualizados.	
Propósito		Actualizar la información de la restauración de las piezas.	
Descripción		Se modifican los datos de la restauración para cambiar las fechas en las que se finalizó la restauración y quien recibió la pieza ya tratada.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar restauración.	2	Presenta un formulario para cambiar los campos del registro. <ul style="list-style-type: none"> • Quien recibe. • Fecha de finalización • Observaciones.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la restauración en la base de datos.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.55. Caso de uso CU051.

Identificador		CU052	
Caso de uso		Consultar restauración.	
Actores		Biólogo.	
Referencias		RFC001, RFC012, CU049, CU084	
Precondiciones		Autenticación de usuario, Gestión de restauraciones.	
Postcondiciones		N/A	
Propósito		Consultar datos de las restauraciones.	
Descripción		Se utiliza para consultar la información de restauraciones.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar restauraciones.	2	Presenta un listado de restauraciones.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.56. Caso de uso CU052.

Identificador		CU053	
Caso de uso		Gestionar recolector.	
Actores		Biólogo.	
Referencias		RFC001, RFC010, CU084	
Precondiciones		Autenticación de usuario.	
Postcondiciones		Paso a las interfaces para agregar, actualizar, consulta o eliminar datos de recolector.	
Propósito		Agregar, modificar o eliminar datos de recolector.	
Descripción		Los datos del recolector son utilizados para reportes, auditorías entre otros.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión datos de recolector.	2	Presenta formulario para agregar a un recolector y una lista ordenada alfabéticamente de los recolectores existentes en el sistema con las opciones para modificarlos o eliminarlos.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.57. Caso de uso CU053.

Identificador		CU054	
Caso de uso		Agregar recolector.	
Actores		Biólogo.	
Referencias		RFC001, RFC010, CU053, CU084	
Precondiciones		Autenticación de usuario, Gestión de recolector.	
Postcondiciones		Recolector agregado a la base de datos.	
Propósito		Añadir un nuevo recolector a la base de datos.	
Descripción		Se debe de agregar la información básica para llevar un control del recolector que colecta el espécimen.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar recolector.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre • Alias • Teléfono • Dirección
3	Ingresa los datos solicitados.	4	Valida el formato de los datos.
		5	Añade el nuevo recolector a la base de datos.
		6	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.58. Caso de uso CU054.

Identificador	CU055		
Caso de uso	Modificar recolector.		
Actores	Biólogo.		
Referencias	RFC001, RFC010, CU053, CU084		
Precondiciones	Autenticación de usuario, Gestión de recolectores.		
Postcondiciones	Datos de recolector modificado.		
Propósito	Modificar recolector en la base de datos.		
Descripción	Se modifica los datos del recolector para mantener información actualizada del mismo.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar recolector.	2	Presenta un formulario para cambiar los campos: <ul style="list-style-type: none"> • Nombre • Alias • Teléfono • Dirección
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del recolector en la base de datos.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.59. Caso de uso CU055.

Identificador	CU056		
Caso de uso	Consultar recolector.		
Actores	Biólogo.		
Referencias	RFC001, RFC010, CU053, CU084		
Precondiciones	Autenticación de usuario, Gestión de recolectores.		
Postcondiciones	N/A		
Propósito	Consultar datos de los recolectores.		
Descripción	Se utiliza para consultar la información del recolector.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar recolector.	2	Presenta un listado de recolectores.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.60. Caso de uso CU056.

Identificador	CU057		
Caso de uso	Eliminar recolector.		
Actores	Biólogo.		
Referencias	RFC001, RFC010, CU053, CU084		
Precondiciones	Autenticación de usuario, Gestión de recolectores.		
Postcondiciones	Recolector eliminado de la base de datos.		
Propósito	Eliminar recolector de la base de datos.		
Descripción	En el caso de que se agregue una persona en los recolectores que no lo sea se eliminará.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar recolector.	2	Verifica que no existan registros vinculados al recolector en la base de datos.
		3	Pregunta si en verdad se desea eliminar al recolector.
4	Acepta la verificación.	5	Se elimina el recolector.
Flujo alternativo			
2a) Existen registros vinculados con el recolector: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al recolector. • El sistema redirige a la interfaz de gestión de datos de recolectores. 			
4a) El usuario deniega la eliminación de recolector: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de datos de recolectores. 			
Observaciones	N/A		

Tabla 3.61. Caso de uso CU057.

Identificador	CU058		
Caso de uso	Gestionar recolección.		
Actores	Biólogo.		
Referencias	RFC001, RFC011, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consulta o eliminar datos de recolección.		
Propósito	Agregar, modificar o eliminar recolección.		
Descripción	Los datos de la recolección son necesarios para saber donde se colecto el espécimen y quienes los recolectaron.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión datos de la recolección.	2	Presenta formulario que permita seleccionar entre agregar, modificar, consultar y eliminar recolección.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.62. Caso de uso CU058.

Identificador		CU059	
Caso de uso		Agregar recolección.	
Actores		Biólogo.	
Referencias		RFC001, RFC011, CU058, CU084.	
Precondiciones		Autenticación de usuario, Gestión de recolección.	
Postcondiciones		Recolección agregado a la base de datos.	
Propósito		Anadir una nueva recolección a la base de datos.	
Descripción		Se debe de agregar la información básica para llevar un control de la recolección.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar recolección.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Fecha de recolección • Latitud • Longitud • Observaciones Si es nacional <ul style="list-style-type: none"> • Nombre del recolector • Temperatura • Altura (mts.) • Recolectores auxiliares • Equipo de recolección • Localidad y zona Si es internacional <ul style="list-style-type: none"> • País • Distrito • Ciudad • Donante
3	Ingresa los datos solicitados.	4	Valida el formato de los datos.
		5	Añade la nueva recolección a la base de datos.
		6	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.63. Caso de uso CU059.

Identificador		CU060	
Caso de uso		Modificar recolección.	
Actores		Biólogo.	
Referencias		RFC001, RFC011, CU058, CU084.	
Precondiciones		Autenticación de usuario, Gestión de recolección.	
Postcondiciones		Datos de recolector modificado.	
Propósito		Modificar recolector en la base de datos.	
Descripción		Se modifica los datos del recolector para mantener información actualizada del mismo.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar recolección.	2	Presenta un formulario para cambiar los datos de la recolección.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos de la recolección en la base de datos.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.64. Caso de uso CU060.

Identificador		CU061	
Caso de uso		Consultar recolección.	
Actores		Biólogo.	
Referencias		RFC001, RFC011, CU058, CU084.	
Precondiciones		Autenticación de usuario, Gestión de recolección.	
Postcondiciones		N/A	
Propósito		Consultar datos de la recolección.	
Descripción		Se utiliza para consultar la información de la recolección.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar recolección.	2	Presenta un listado de recolección.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.65. Caso de uso CU061.

Identificador	CU062		
Caso de uso	Eliminar recolección.		
Actores	Biólogo.		
Referencias	RFC001, RFC011, CU058, CU084.		
Precondiciones	Autenticación de usuario, Gestión de recolecciones.		
Postcondiciones	Recolección eliminada de la base de datos.		
Propósito	Eliminar recolección de la base de datos.		
Descripción	La recolección se elimina en el caso de que sea necesario y no tenga datos relacionadas a especímenes.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar recolección.	2	Verifica que no existan registros vinculados a la recolección en la base de datos.
		3	Pregunta si en verdad se desea eliminar la recolección.
4	Acepta la verificación.	5	Se elimina la recolección.
Flujo alternativo			
2a) Existen registros vinculados con la recolección: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados la recolección. • El sistema redirige a la interfaz de gestión de datos de recolecciones. 			
4a) El usuario deniega la eliminación de recolección: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de datos de recolecciones. 			
Observaciones	N/A		

Tabla 3.66. Caso de uso CU062.

Identificador		CU063	
Caso de uso		Gestionar geografía.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU084	
Precondiciones		Autenticación de usuario.	
Postcondiciones		Se presentan las opciones de gestionar departamento, municipio, zona, área protegida.	
Propósito		Mantener un catálogo de la geografía del país.	
Descripción		Se da mantenimiento al catálogo de geografía que sirve de referencia para las recolecciones y ubicaciones de especímenes.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión de datos de geografía.	2	Presenta formulario que permita seleccionar: <ul style="list-style-type: none"> • Gestionar departamento • Gestionar municipio • Gestionar zona • Gestionar área protegida
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.67. Caso de uso CU063.

Identificador		CU064	
Caso de uso		Gestionar departamento.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU063, CU084	
Precondiciones		Autenticación de usuario.	
Postcondiciones		Paso a las interfaces para agregar, actualizar, consulta o eliminar datos del departamento.	
Propósito		Mantener un catálogo de los departamentos de El Salvador.	
Descripción		Los departamentos se deben de gestionar antes de gestionar cualquier elemento de la geografía como lo es municipios, zonas, áreas protegidas.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión de datos del departamento.	2	Presenta formulario que permita seleccionar entre agregar, modificar, consultar y eliminar departamento.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.68. Caso de uso CU064.

Identificador	CU065		
Caso de uso	Agregar departamento.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU064, CU084.		
Precondiciones	Autenticación de usuario, Gestión de departamento.		
Postcondiciones	Listado de departamentos.		
Propósito	Agregar departamento.		
Descripción	Se agregan los datos del departamento para poder agregar municipios.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar departamento.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre • Región geográfica
3	Ingresa los datos solicitados.	4	Valida el formato de los datos.
		5	Añade el nuevo departamento a la base de datos.
		6	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.69. Caso de uso CU065.

Identificador		CU066	
Caso de uso		Modificar departamento.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU064, CU084	
Precondiciones		Autenticación de usuario, Gestión de departamentos.	
Postcondiciones		Departamento actualizado.	
Propósito		Actualizar los datos del departamento.	
Descripción		Se pueden modificar todos los datos del departamento.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar departamento.	2	Presenta un formulario para cambiar los datos del departamento.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del departamento en la base de datos.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.70. Caso de uso CU066.

Identificador		CU067	
Caso de uso		Consultar departamento.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU064, CU084.	
Precondiciones		Autenticación de usuario, Gestión de departamento.	
Postcondiciones		N/A	
Propósito		Consultar datos del departamento.	
Descripción		Se utiliza para consultar la información del departamento.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar departamento.	2	Presenta un listado del departamento.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.71. Caso de uso CU067.

Identificador	CU068		
Caso de uso	Eliminar departamento.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU064, CU084.		
Precondiciones	Autenticación de usuario, Gestión de departamento.		
Postcondiciones	Departamento eliminado de la base de datos.		
Propósito	Eliminar departamento de la base de datos.		
Descripción	Los departamentos son eliminados si no tienen registros vinculados a ellos.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar departamento.	2	Verifica que no existan registros vinculados al departamento en la base de datos.
		3	Pregunta si en verdad se desea eliminar el departamento.
4	Acepta la verificación.	5	Se elimina el departamento.
Flujo alternativo			
2a) Existen registros vinculados con el departamento: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al departamento. • El sistema redirige a la interfaz de gestión del departamento. 			
4a) El usuario deniega la eliminación del departamento: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión del departamento. 			
Observaciones	N/A		

Tabla 3.72. Caso de uso CU068.

Identificador	CU069		
Caso de uso	Gestionar municipio.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU063, CU064, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consulta o eliminar datos de municipios.		
Propósito	Mantener un catálogo de los municipios de El Salvador.		
Descripción	Los municipios son agregados después de los departamentos para ser asignados a áreas protegidas, zonas y recolecciones que nos permitirán verificar ubicación de distintas especímenes.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión datos del municipio.	2	Presenta formulario que permita seleccionar entre agregar, modificar, consultar y eliminar municipio.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.73. Caso de uso CU069.

Identificador	CU070		
Caso de uso	Agregar municipio.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU069, CU084.		
Precondiciones	Autenticación de usuario, Gestión de municipio.		
Postcondiciones	Listado de municipios.		
Propósito	Agregar municipio.		
Descripción	Se maneja la información de las coordenadas del municipio así como el departamento al que pertenece y si es cabecera del mismo o no.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar municipio.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre • Latitud • Longitud • Cabecera departamental • Departamento
3	Ingresar los datos solicitados.	4	Valida el formato de los datos.
		5	Añade el nuevo municipio a la base de datos.
		6	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.74. Caso de uso CU070.

Identificador	CU071		
Caso de uso	Modificar municipio.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU069, CU084		
Precondiciones	Autenticación de usuario, Gestión de municipios.		
Postcondiciones	Municipio actualizado.		
Propósito	Actualizar los datos del municipio.		
Descripción	Se pueden modificar todos los datos del municipio a excepción del departamento al que está relacionado.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar municipio.	2	Presenta un formulario para cambiar los datos del municipio.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del municipio en la base de datos.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto:			
<ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.75. Caso de uso CU071.

Identificador	CU072		
Caso de uso	Consultar municipio.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU069, CU084.		
Precondiciones	Autenticación de usuario, Gestión de municipio.		
Postcondiciones	N/A		
Propósito	Consultar datos del municipio.		
Descripción	Se utiliza para consultar la información del municipio.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar municipio.	2	Presenta un listado del municipio.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.76. Caso de uso CU072.

Identificador	CU073		
Caso de uso	Eliminar municipio.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU069, CU084.		
Precondiciones	Autenticación de usuario, Gestión de municipio.		
Postcondiciones	Municipio eliminado de la base de datos.		
Propósito	Eliminar municipio de la base de datos.		
Descripción	Los municipios son datos que se si se pueden recuperar, por ende su eliminación si no hay datos relacionados al mismo será física.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar municipio.	2	Verifica que no existan registros vinculados al municipio en la base de datos.
		3	Pregunta si en verdad se desea eliminar el municipio.
4	Acepta la verificación.	5	Se elimina el municipio.
Flujo alternativo			
2a) Existen registros vinculados con el municipio: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al municipio. • El sistema redirige a la interfaz de gestión del municipio. 			
4a) El usuario deniega la eliminación del municipio: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión del municipio. 			
Observaciones	N/A		

Tabla 3.77. Caso de uso CU073.

Identificador	CU074		
Caso de uso	Gestionar área protegida.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU063, CU084.		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consulta o eliminar área protegida.		
Propósito	Mantener un catálogo de áreas protegidas del país.		
Descripción	Seleccionar una opción de las áreas protegidas.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión de áreas protegidas.	2	Presenta formulario que permita seleccionar entre agregar, modificar, consultar y eliminar áreas protegidas.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.78. Caso de uso CU074.

Identificador	CU075		
Caso de uso	Agregar área protegida.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU074, CU084.		
Precondiciones	Autenticación de usuario, Gestión de área protegida.		
Postcondiciones	Área protegida agregado a la base de datos.		
Propósito	Añadir una nueva área protegida a la base de datos.		
Descripción	Son reservas naturales en las que se pueden hacer recolecciones.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para agregar área protegida.	2	Presenta un formulario solicitando se elija el tipo de área protegida. <ul style="list-style-type: none"> • Nombre • Descripción • Municipio
3	Se ingresan los datos respectivos.	4	Validan los datos
		5	Guarda los datos
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones	N/A		

Tabla 3.79. Caso de uso CU075.

Identificador		CU076	
Caso de uso		Modificar área protegida.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU074, CU084.	
Precondiciones		Autenticación de usuario, Gestión de área protegida.	
Postcondiciones		Datos del área protegida modificada.	
Propósito		Modificar área protegida en la base de datos.	
Descripción		Se modifican todos los datos del área protegida.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar el área protegida.	2	Presenta un formulario para cambiar los datos del área protegida.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos del área protegida.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.80. Caso de uso CU076.

Identificador		CU077	
Caso de uso		Consultar área protegida.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU074, CU084.	
Precondiciones		Autenticación de usuario, Gestión de área protegida.	
Postcondiciones		N/A	
Propósito		Consultar datos del área protegida.	
Descripción		Se utiliza para consultar la información del área protegida.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar área protegida.	2	Presenta un listado de áreas protegidas.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.81. Caso de uso CU077.

Identificador	CU078		
Caso de uso	Eliminar área protegida.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU074, CU084.		
Precondiciones	Autenticación de usuario, Gestión de área protegida.		
Postcondiciones	Área protegida eliminada de la base de datos.		
Propósito	Eliminar área protegida.		
Descripción	Las áreas protegidas que no sean necesarias ni estén vinculadas podrán ser eliminadas.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar área protegida.	2	Verifica que no existan registros vinculados al área protegida.
		3	Pregunta si en verdad se desea eliminar al área protegida.
4	Acepta la verificación.	5	Se elimina al área protegida.
Flujo alternativo			
2a) Existen registros vinculados con al área protegida: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados al área protegida. • El sistema redirige a la interfaz de gestión de áreas protegidas. 			
4a) El usuario deniega la eliminación de áreas protegidas: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de áreas protegidas. 			
Observaciones	N/A		

Tabla 3.82. Caso de uso CU078.

Identificador	CU079		
Caso de uso	Gestionar zona.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU063, CU084.		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consulta o eliminar zona.		
Propósito	Dar mantenimiento a las distintas zonas que se encuentran en el país.		
Descripción	Los datos de la zona son necesarios para mantener un catálogo de los diferentes tipos que existen.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de gestión de zona.	2	Presenta formulario que permita seleccionar entre agregar, modificar, consultar y eliminar zona.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.83. Caso de uso CU079.

Identificador		CU080	
Caso de uso		Agregar recolección.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU079, CU084.	
Precondiciones		Autenticación de usuario, Gestión de zona.	
Postcondiciones		Zona agregado a la base de datos.	
Propósito		Anadir una nueva zona a la base de datos.	
Descripción		Las zonas se dividen entre urbanas, rurales y áreas protegidas.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar zona.	2	Presenta un formulario solicitando se elija el tipo de zona. <ul style="list-style-type: none"> • Área protegida • Urbano • Rural
3	El usuario elige el tipo de zona.	4	Se solicita la información necesaria dependiendo de la zona. <p>Urbana</p> <ul style="list-style-type: none"> • Municipio • Barrio • Colonia • Sector <p>Rural</p> <ul style="list-style-type: none"> • Municipio • Catón • Caserío <p>Área protegida</p> <ul style="list-style-type: none"> • Área protegida • Lugar
		5	Añade la nueva zona a la base de datos.
		6	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		N/A	

Tabla 3.84. Caso de uso CU080.

Identificador		CU081	
Caso de uso		Modificar zona.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU079, CU084.	
Precondiciones		Autenticación de usuario, Gestión de zona.	
Postcondiciones		Datos de la zona modificada.	
Propósito		Modificar zona en la base de datos.	
Descripción		Los datos de la zona se modifican dependiendo del tipo de zona que sea.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para modificar.	2	Presenta un formulario para cambiar los datos.
3	Actualiza los datos.	4	Valida el formato de los datos.
		5	Actualiza los datos en la base de datos.
		6	Muestra una notificación de modificación exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
Observaciones		El formulario presentado para modificación dependerá del tipo de zona.	

Tabla 3.85. Caso de uso CU081.

Identificador		CU082	
Caso de uso		Consultar zona.	
Actores		Biólogo.	
Referencias		RFC001, RFC009, CU079, CU084.	
Precondiciones		Autenticación de usuario, Gestión de zona.	
Postcondiciones		N/A	
Propósito		Consultar datos de la zona.	
Descripción		Se utiliza para consultar la información de la zona.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar zona.	2	Presenta un listado de zonas.
Flujo alternativo			
N/A			
Observaciones		N/A	

Tabla 3.86. Caso de uso CU082.

Identificador	CU083		
Caso de uso	Eliminar zona.		
Actores	Biólogo.		
Referencias	RFC001, RFC009, CU079, CU084.		
Precondiciones	Autenticación de usuario, Gestión de zona.		
Postcondiciones	Zona eliminada de la base de datos.		
Propósito	Eliminar zona.		
Descripción	Las zonas que no sean necesarias ni estén vinculadas podrán ser eliminadas.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar zona.	2	Verifica que no existan registros vinculados a la zona.
		3	Pregunta si en verdad se desea eliminar la zona.
4	Acepta la verificación.	5	Se elimina la zona.
Flujo alternativo			
2a) Existen registros vinculados con la zona: <ul style="list-style-type: none"> • Se notifica que existen registros vinculados la zona. • El sistema redirige a la interfaz de gestión de zonas. 			
4a) El usuario deniega la eliminación de zona: <ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de datos de zona. 			
Observaciones	N/A		

Tabla 3.87. Caso de uso CU083.

Identificador	CU084		
Caso de uso	Autenticar usuario.		
Actor	Administrador, Biólogo, Digitador.		
Referencias	RFC001		
Precondiciones	N/A		
Postcondiciones	Autenticar el usuario de forma exitosa.		
Propósito	Controlar que solo usuarios ingresen al sistema.		
Descripción	Se autentica al usuario para que solo los usuarios que estén relacionados con el sistema puedan acceder a este.		
Flujo básico			
Actor		Sistema	
		1	Solicita al usuario ingrese los datos de ingreso.
2	Se ingresa su nombre y contraseña.	3	Verifica que los datos sean correctos.
		4	Permite el ingreso al usuario.
Flujo alternativo			
3a) Datos incorrectos			
En el caso de que el nombre y usuario no cumplan con lo solicitado:			
A) El sistema mostrara un mensaje de datos incorrectos.			
B) El sistema solicitara ingrese los datos nuevamente.			
C) El usuario ingresa los datos.			
D) El sistema los verifica.			
E) El sistema permite el ingreso al usuario.			
Observaciones	En el caso de que no se posea usuario o el usuario no sea válido se debe de referenciar al administrador del sistema.		

Tabla 3.88. Caso de uso CU084.

Identificador	CU085		
Caso de uso	Administrar usuarios.		
Actores	Administrador.		
Referencias	RFC001, RFC004, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar o eliminar usuario.		
Propósito	Agregar, modificar o eliminar usuario.		
Descripción	El administrador puede realizar cualquier operación que sirva para dar mantenimiento al usuario.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de administración de usuarios.	2	Presenta formulario para agregar usuario y una lista ordenada alfabéticamente de los usuarios modificables existentes en el sistema con las opciones para modificarlos o eliminarlos.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.89. Caso de uso CU085.

Identificador		CU086	
Caso de uso		Agregar usuario.	
Actores		Administrador.	
Referencias		RFC001, RFC004, CU084, CU085	
Precondiciones		Autenticación de usuario, Administración de usuarios.	
Postcondiciones		Usuario agregado al sistema.	
Propósito		Aadir un nuevo usuario al sistema.	
Descripción		El administrador es el que se encarga de dar los permisos así como de agregar a los usuarios para que puedan realizar procesos en el sistema.	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para agregar usuario.	2	Presenta un formulario solicitando los datos: <ul style="list-style-type: none"> • Nombre • Correo
3	Ingresa los datos solicitados.	4	Valida el formato de los datos, que no exista otro usuario asociado al correo y que la contraseña tenga los requisitos de seguridad mínimos.
		5	Añade el nuevo usuario a la base de datos a la espera de verificación.
		6	Envía un correo a la dirección de correo proporcionado con un enlace de verificación de la cuenta.
		7	Muestra una notificación de adición exitosa.
Flujo alternativo			
4a) El usuario ingresa los datos con formato incorrecto: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que los errores sean corregidos. 			
4b) El usuario ingresa un correo que ya está asociado a un usuario del sistema: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que se cambie el correo del usuario. 			
4c) El usuario ingresa la contraseña sin cumplir los requerimientos de seguridad mínimos o la verificación no coincide: <ul style="list-style-type: none"> • El sistema presenta una notificación de error y deshabilita el botón de guardar hasta que se corrija la contraseña. 			
Observaciones	N/A		

Tabla 3.90. Caso de uso CU086.

Identificador	CU087		
Caso de uso	Modificar usuario.		
Actores	Administrador.		
Referencias	RFC001, RFC004, CU084, CU085		
Precondiciones	Autenticación de usuario, Administración de usuarios.		
Postcondiciones	Datos de usuario modificado.		
Propósito	Modificar usuario del sistema.		
Descripción	Se utiliza para actualizar la información de los usuarios en el caso de que sea necesario.		
Flujo básico			
Actor		Sistema	
1	Ingres a la interfaz para modificar usuario.	2	Presenta un formulario para forzar el cambio de nombre o contraseña.
3	Modifica el nombre del usuario		
4	Selecciona la opción de renovación de contraseña.	5	Actualiza la base de datos con los nuevos datos.
		6	Envía un correo al usuario que se está administrando con un enlace para que renueve su contraseña.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.91. Caso de uso CU087.

Identificador	CU088		
Caso de uso	Consultar usuario.		
Actores	Administrador.		
Referencias	RFC001, RFC004, CU084, CU085		
Precondiciones	Autenticación de usuario, Administración de usuarios.		
Postcondiciones	N/A		
Propósito	Consultar usuario del sistema.		
Descripción	Se utiliza para consultar la información de los usuarios.		
Flujo básico			
Actor		Sistema	
1	Ingres a la interfaz para consultar usuario.	2	Presenta un listado de usuarios.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.92. Caso de uso CU088.

Identificador	CU089		
Caso de uso	Eliminar usuario.		
Actores	Administrador.		
Referencias	RFC001, RFC004, CU084, CU085		
Precondiciones	Autenticación de usuario, Administración de usuarios.		
Postcondiciones	Usuario eliminado del sistema.		
Propósito	Eliminar usuario del sistema.		
Descripción	Invalida las credenciales del usuario.		
Flujo básico			
Actor		Sistema	
1	Se ingresa a la interfaz para eliminar.	2	Pregunta si en verdad se desea eliminar.
3	Acepta la verificación.	4	Actualiza los datos de usuario para que sus credenciales sean inválidas.
Flujo alternativo			
3a) El usuario deniega la eliminación del usuario que se está administrando.			
<ul style="list-style-type: none"> El sistema redirige a la interfaz de administración de usuarios. 			
Observaciones	Por motivos de auditorías y bitácoras, los usuarios no serán eliminados propiamente, sino se revocarán todos sus privilegios y su estado cambiará a inactivo.		

Tabla 3.93. Caso de uso CU089.

Identificador	CU090		
Caso de uso	Gestionar privilegios		
Actores	Administrador.		
Referencias	RFC001, RFC002, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consultar o eliminar privilegios.		
Propósito	Agregar, modificar, consultar o eliminar privilegios.		
Descripción	El administrador puede realizar cualquier operación que sirva para dar mantenimiento a los privilegios.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de administración de privilegios.	2	Presenta formulario para agregar, modificar o eliminar privilegios.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.94. Caso de uso CU090.

Identificador	CU091		
Caso de uso	Agregar privilegios.		
Actor	Administrador del sistema.		
Referencias	RFC001, RFC002, CU084, CU090		
Precondiciones	Autenticar Usuario.		
Postcondiciones	Usuario con Privilegios.		
Propósito	Asignar permisos al usuario para que pueda realizar procesos en el sistema.		
Descripción	El administrador es el que se encarga de asignar los privilegios de cada uno de los usuarios para que pueda realizar los procesos, este se da luego de registrar un usuario al sistema.		
Flujo básico			
Actor		Sistema	
1	El usuario entra a la interfaz de administración privilegios.	2	El sistema presenta la lista de los usuario administrables
3	El usuario selecciona al usuario a modificar privilegios	4	El sistema solicita asignar los privilegios.
5	El administrador del sistema asigna/modifica los privilegios.	6	Actualiza los privilegios.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.95. Caso de uso CU091.

Identificador	CU092		
Caso de uso	Modificar privilegio.		
Actores	Administrador.		
Referencias	RFC001, RFC002, CU084, CU090		
Precondiciones	Autenticación de usuario, Gestionar privilegio.		
Postcondiciones	Datos de privilegio modificado.		
Propósito	Modificar privilegios.		
Descripción	Se utiliza para actualizar la información de los privilegios en el caso de que sea necesario.		
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz para modificar privilegio.	2	Presenta un formulario con los datos del privilegio
3	Selecciona los datos a modificar.	4	Actualiza la base de datos con los nuevos datos.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.96. Caso de uso CU092.

Identificador	CU093		
Caso de uso	Consultar privilegio.		
Actores	Administrador.		
Referencias	RFC001, RFC002, CU084, CU090		
Precondiciones	Autenticación de usuario, gestionar privilegios		
Postcondiciones	N/A		
Propósito	Consultar privilegio.		
Descripción	Se utiliza para consultar los privilegios.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz para consultar privilegios.	2	Presenta un listado de privilegios.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.97. Caso de uso CU093.

Identificador	CU094		
Caso de uso	Eliminar privilegio.		
Actores	Administrador.		
Referencias	RFC001, RFC002, CU084, CU090		
Precondiciones	Autenticación de usuario, gestionar privilegio.		
Postcondiciones	Privilegio eliminado.		
Propósito	Eliminar privilegio.		
Descripción	Si el privilegio ya no es necesario se elimina.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar privilegio.	2	Pregunta si en verdad se desea eliminar al privilegio.
3	Acepta la verificación.		
Flujo alternativo			
3a) El usuario deniega la eliminación del privilegio que se está administrando.			
<ul style="list-style-type: none"> • El sistema redirige a la interfaz de gestión de privilegios. 			
Observaciones	N/A		

Tabla 3.98. Caso de uso CU094.

Identificador	CU095		
Caso de uso	Gestionar rol		
Actores	Administrador.		
Referencias	RFC001, RFC003, CU084		
Precondiciones	Autenticación de usuario.		
Postcondiciones	Paso a las interfaces para agregar, actualizar, consultar o eliminar roles.		
Propósito	Agregar, modificar, consultar o eliminar roles.		
Descripción	Los roles son los perfiles que se le agregan a los usuarios para que puedan realizar operaciones en los objetos.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de administración de roles.	2	Presenta las opciones como los son agregar rol, actualizar rol, consultar rol y eliminar rol.
3	Selecciona una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.99. Caso de uso CU095.

Identificador	CU096		
Caso de uso	Agregar rol.		
Actor	Administrador del sistema.		
Referencias	RFC001, RFC003, CU084, CU095		
Precondiciones	Autenticar Usuario.		
Postcondiciones	Listado de roles.		
Propósito	Usuario con roles para realizar procesos en el sistema.		
Descripción	Son los diferentes opciones que podrán realizarse están relacionados con los privilegios.		
Flujo básico			
Actor		Sistema	
1	El usuario entra a la interfaz de administración roles.	2	Se presenta un formulario solicitando los datos. <ul style="list-style-type: none"> • Nombre • Descripción.
3	El usuario ingresa la información.	4	El sistema verifica los datos.
		5	Guarda la información.
Flujo alternativo			
3a) Datos erróneos El sistema deshabilita el botón guardar y muestra mensajes de error.			
Observaciones	N/A		

Tabla 3.100. Caso de uso CU096.

Identificador	CU097			
Caso de uso	Modificar rol.			
Actores	Administrador.			
Referencias	RFC001, RFC003, CU084, CU095			
Precondiciones	Autenticación de usuario, Gestionar rol.			
Postcondiciones	Datos del rol modificado.			
Propósito	Modificar rol.			
Descripción	Se utiliza para actualizar la información de los roles en el caso de que sea necesario.			
Flujo básico				
Actor			Sistema	
1	Ingresar a la interfaz para modificar rol.	2	Presenta un formulario con los datos del rol.	
3	Selecciona los datos a modificar.	4	Actualiza la base de datos con los nuevos datos.	
Flujo alternativo				
N/A				
Observaciones	N/A			

Tabla 3.101. Caso de uso CU097.

Identificador	CU098			
Caso de uso	Consultar rol.			
Actores	Administrador.			
Referencias	RFC001, RFC003, CU084, CU095			
Precondiciones	Autenticación de usuario, gestionar rol.			
Postcondiciones	N/A			
Propósito	Consultar rol.			
Descripción	Se utiliza para consultar los roles.			
Flujo básico				
Actor			Sistema	
1	Ingresar a la interfaz para consultar roles.	2	Presenta un listado de roles.	
Flujo alternativo				
N/A				
Observaciones	N/A			

Tabla 3.102. Caso de uso CU098.

Identificador	CU099		
Caso de uso	Eliminar roles.		
Actores	Administrador.		
Referencias	RFC001, RFC003, CU084, CU095		
Precondiciones	Autenticación de usuario, gestionar rol.		
Postcondiciones	Rol eliminado.		
Propósito	Eliminar rol.		
Descripción	Si el rol ya no es necesario se elimina.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz para eliminar rol.	2	Pregunta si en verdad se desea eliminar al rol.
3	Acepta la verificación.		
Flujo alternativo			
3a) El usuario deniega la eliminación del rol que se está administrando.			
<ul style="list-style-type: none"> El sistema redirige a la interfaz de gestión de roles. 			
Observaciones	N/A		

Tabla 3.103. Caso de uso CU099.

Identificador	CU0100		
Caso de uso	Presentar bitácora		
Actores	Administrador.		
Referencias	RFC001, RFC013, CU084		
Precondiciones	Autenticar Usuario		
Postcondiciones	N/A		
Propósito	Controlar todas las acciones que se realicen en el sistema		
Descripción	Se utilizará para llevar un registro de que operaciones se realizan en el sistema y quien las realiza.		
Flujo básico			
Actor		Sistema	
1	Realiza una acción como registrar, actualizar o eliminar información.	2	Guardara la información de que usuario realizo la acción, que operación ejecuto, que información fue modificada y en la fecha y hora respectiva.
		3	Presenta un conjunto de todos los datos que se guardaron por todos los usuarios.
Flujo alternativo			
N/A			
Observaciones	Se lleva un registro de todas las acciones que modifique la información para auditorias y reportes del mismo.		

Tabla 3.104. Caso de uso CU100.

Identificador	CU101		
Caso de uso	Importar catalogo taxonómico		
Actores	Administrador		
Referencias	RFC001, RFC013, CU084		
Precondiciones	Autenticar Usuario		
Postcondiciones	N/A		
Propósito	Importar catalogo taxonómico de fuentes externas		
Descripción	Se importara de archivos externos las distintas características taxonómicas de las especies animales.		
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz de importación de catálogo.		
2	El usuario define la fuente de datos	3	Importa los datos
		4	Valida los datos
		5	Formatea los datos
		6	Guarda los datos
Flujo alternativo			
<p>3a) No se tiene autorización para acceder a los datos:</p> <ul style="list-style-type: none"> • Se presentará una notificación del error. • Se regresará a la interfaz de importación de catálogo. <p>3b) Falló la conexión con la fuente de datos:</p> <ul style="list-style-type: none"> • Se presentará una notificación del error. • Se revertirá cualquier cambio realizado en la base de datos. • Se regresará a la interfaz de importación de catálogo. <p>3c) Se canceló la operación:</p> <ul style="list-style-type: none"> • Se revertirá cualquier cambio realizado en la base de datos. • Se regresará a la interfaz de importación de catálogo. <p>4a) Falló la validación de los datos:</p> <ul style="list-style-type: none"> • Se ignorarán los registros involucrados • Se presentará una notificación del error para los registros específicos. <p>5a) Falló el formateo de los datos:</p> <ul style="list-style-type: none"> • Se ignorarán los registros involucrados • Se presentará una notificación del error para los registros específicos. 			
Observaciones	Se lleva un registro de todas las acciones que modifique la información para auditorias.		

Tabla 3.105. Caso de uso CU101.

Identificador		CU102	
Caso de uso		Exportar catalogo taxonómico	
Actores		Administrador	
Referencias		RFC001, RFC014, CU084	
Precondiciones		Autenticar Usuario	
Postcondiciones		N/A	
Propósito		Exportar catalogo taxonómico.	
Descripción		Se exporta el archivo a XML, CSV.	
Flujo básico			
Actor		Sistema	
1	El usuario ingresa a la interfaz de exportación de catálogo.	2	Se muestra un listado de toda la taxonomía.
2	El usuario elige el tipo de archivo a cual se desea exportar		
3	El usuario elige la ubicación y el nombre si desea cambiar esto.	4	Se guarda los datos.
Flujo alternativo			
N/A			
Observaciones	Se lleva un registro de todas las acciones que modifique la información para auditorias.		

Tabla 3.106. Caso de uso CU102.

Identificador		CU103	
Caso de uso		Gestionar reporte	
Actor		Biólogo, administrador.	
Referencias		RFC001, RFC015, RFC016, CU084	
Precondiciones		Autenticar usuario.	
Postcondiciones		El usuario podrá elegir entre generar reportes dinámicos, especímenes, restauración, recolecciones, usuarios.	
Propósito		Generar los diferentes reportes que brinden ayuda al usuario.	
Descripción		Los reportes se dividen en dinámicos que son los que no están definidos previamente pero son necesarios para el manejo de información y los ya definidos como los son reportes de especímenes, de restauración, de recolección y de usuarios	
Flujo básico			
Actor		Sistema	
1	Ingresa a la interfaz de gestión de reportes.	2	Se presentan las opciones que se quieren ejecutar las cuales son: <ul style="list-style-type: none"> • Generar reporte dinámico • Generar reporte de especímenes • Generar reporte de restauraciones • Generar reporte de recolecciones • Generar reporte de usuarios.
3	Elige una opción.	4	Dirige a la interfaz respectiva según la opción seleccionada.
Flujo alternativo			
N/A			
Observaciones	N/A		

Tabla 3.107. Caso de uso CU103.

Identificador	CU104		
Caso de uso	Generar reportes dinámicos		
Actor	Biólogo, administrador.		
Referencias	RFC001, RFC016, CU084, CU103		
Precondiciones	Autenticar usuario.		
Postcondiciones	Tener un reporte que no está definido como básico pero es necesario en algún momento y que el usuario personalizo.		
Propósito	Facilitar el elaborar reportes que no son de uso mensual pero pueden ser utilizados algunas veces.		
Descripción	Se presenta una interfaz que permita elegir las opciones que se deben de incluir en el reporte así como las diferentes clasificaciones de la información para facilitar la elaboración del mismo.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de generar reporte dinámico.	2	Se presentan un formulario donde debe seleccionar los campos a presentar en el reporte según la categoría.
3	El usuario elige los campos de cada categoría	4	Se le presenta otro formulario con los campos elegidos en listas desplegables para que el usuario pueda hacer filtros en cada uno de los campos.
5	El usuario ingresa los filtros en los campos deseados si los hay.	6	Se presenta un formulario con los campos para que el usuario elija como los desea ordenar.
7	El usuario elige la forma de ordenación si en caso solo da siguiente será de manera ascendente y por el orden de campos de selección.	8	El sistema presenta la consulta que será generada de acuerdo a las características y selecciones que el usuario eligió.
9	El usuario corrobora la información	10	Se presenta el reporte solicitado.
Flujo alternativo			
N/A			
Observaciones	Si el usuario no tiene el plugin de adobe Reader que es necesario para visualizar el reporte se descargara el reporte a su computadora, si tiene instalado el plugin será presentado en pantalla.		

Tabla 3.108. Caso de uso CU104.

Identificador		CU105	
Caso de uso		Generar Ficha del espécimen	
Actor		Biólogo.	
Referencias		RFC001, RFC015, CU084, CU103	
Precondiciones		Autenticar usuario.	
Postcondiciones		Ficha del espécimen que incluye la información completa del espécimen.	
Propósito		Elaborar una ficha que incluya la información del espécimen en forma ordenada y clasificada.	
Descripción		Sirve para imprimir la ficha del espécimen y mantener la información tanto digital como en papel así mismo para ver la información del espécimen de manera detallada.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de generar ficha del espécimen.	2	Se presentan un listado de todos los especímenes.
3	El usuario elige el espécimen.	4	Muestra la ficha del espécimen con la información del mismo.
Flujo alternativo			
N/A			
Observaciones	Si el usuario no tiene el plugin de adobe Reader que es necesario para visualizar el reporte se descargara el reporte a su computadora, si tiene instalado el plugin será presentado en pantalla.		

Tabla 3.109. Caso de uso CU105.

Identificador		CU106	
Caso de uso		Generar reporte de restauración	
Actor		Biólogo.	
Referencias		RFC001, RFC015, CU084, CU103	
Precondiciones		Autenticar usuario.	
Postcondiciones		Listado de las restauraciones realizadas.	
Propósito		Llevar un control de todas las restauraciones realizadas así como de los especímenes a las cuales pertenecen.	
Descripción		Muestra las restauraciones realizadas en el mes y la información de quien la realizo, observaciones que podría tener entre otros.	
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de generar reporte de restauración.	2	Se presenta un formulario donde se piden las fechas iniciales y de finalización del reporte
3	El usuario introduce las fechas.	4	Valida que las fechas estén correctas.
		5	Presenta el reporte con la información de las restauraciones.
Flujo alternativo			
N/A			
Observaciones	Si el usuario no tiene el plugin de adobe Reader que es necesario para visualizarlo se descargara el reporte a su computadora, si está instalado será presentado en pantalla.		

Tabla 3.110. Caso de uso CU106.

Identificador	CU107		
Caso de uso	Generar reporte de recolecciones		
Actor	Biólogo.		
Referencias	RFC001, RFC015, CU084, CU103		
Precondiciones	Autenticar usuario.		
Postcondiciones	Listado de recolecciones realizadas.		
Propósito	Llevar un control de todas las recolecciones y de las áreas en las que se recolectaron.		
Descripción	Sirve para tener un control de las recolecciones realizadas así como de los encargados de estas.		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de generar reporte de recolecciones.	2	Se presenta un formulario donde se piden las fechas iniciales y de finalización del reporte
3	El usuario introduce las fechas.	4	Valida que las fechas estén correctas.
		5	Presenta el reporte con la información de las recolecciones.
Flujo alternativo			
N/A			
Observaciones	Si el usuario no tiene el plugin de adobe Reader que es necesario para visualizarlo se descargara el reporte a su computadora, si está instalado será presentado en pantalla.		

Tabla 3.111. Caso de uso CU107.

Identificador	CU108		
Caso de uso	Generar reporte de usuarios del sistema		
Actor	Administrador.		
Referencias	RFC001, RFC015, CU084, CU103		
Precondiciones	Autenticar usuario.		
Postcondiciones	Listado de usuarios del sistema.		
Propósito	Tener un control de los usuarios del sistema.		
Descripción	Muestra los usuarios que pueden acceder al sistema así como los permisos que tienen para realizar las operaciones		
Flujo básico			
Actor		Sistema	
1	Ingresar a la interfaz de generar reporte de usuarios del sistema.	2	Presenta el reporte con la información de los usuarios del sistema.
Flujo alternativo			
N/A			
Observaciones	Si el usuario no tiene adobe Reader que es necesario para visualizarlo se descargara el reporte a su computadora, si está instalado será presentado en pantalla.		

Tabla 3.112. Caso de uso CU108.

3.3. DIAGRAMAS DE INTERACCIÓN

Los diagramas de interacción son diagramas que describen cómo grupos de objetos colaboran para conseguir algún fin. Estos diagramas muestran objetos, así como los mensajes que se pasan entre ellos dentro de los casos de uso. Es decir, capturan el comportamiento de los casos de uso.

Se describen dos tipos de diagramas de interacción:

- **Diagramas de secuencia:** Muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista de negocios del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos.
- **Diagramas de colaboración:** Muestra la implementación de una operación. La comunicación 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.

3.3.1. NOMENCLATURA DEL DIAGRAMA DE INTERACCIÓN

En la tabla 3.113 se muestra la nomenclatura del diagrama de colaboración.



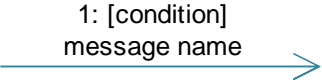
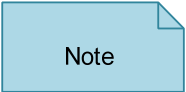
Símbolo	Nombre	Descripción
	Objeto de Clases	Representan las clases en el diagrama de colaboración.
	Conexión	Se utiliza para unir dos objetos de clases.
	Mensaje	Representa la acción que se hará de un objeto a otro, la flecha indica el objeto que resultara afectado. El número que va en el mensaje representa el orden que tiene el diagrama, es decir, cuál de los elementos se realizara primero.
	Notas	Se utilizan para comentar los mensajes u objetos en específico.

Tabla 3.113. Nomenclatura del diagrama de colaboración.

En la tabla 3.114 se muestra la nomenclatura del diagrama de secuencia.

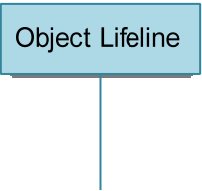
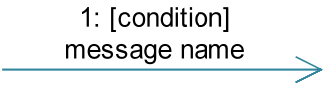
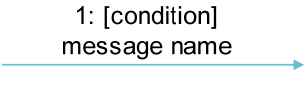
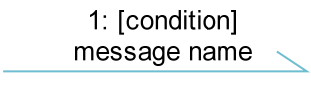
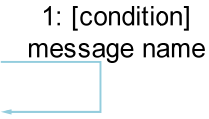
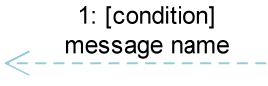
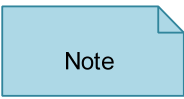
Símbolo	Nombre	Descripción
	Objeto de Clases	Los objetos de los diagramas de secuencia son representados por un rectángulo y una línea vertical donde se conectan cada mensaje del objeto a otro.
	Mensaje	Un mensaje que va de la línea de vida de un objeto al de otro. Puede ser simple, síncrono y asíncrono y se sabe dependiendo de la punta de la flecha. Mensaje simple: es la transferencia del control de un objeto a otro.
	Mensaje Síncrono	Es cuando el objeto espera la respuesta a ese mensaje antes de continuar con su trabajo.
	Mensaje Asíncrono	Es cuando el objeto no espera la respuesta a ese mensaje antes de continuar.
	Asociación	Un objeto puede enviarse un mensaje así mismo es decir de su línea de vida a su propia línea de vida. También es conocido como recursividad y se utiliza dependiendo del lenguaje de programación.
	Agregación	Mensajes de retorno de condición surgen dependiendo de la acción a realizar en el primer mensaje.
	Notas	Se utilizan para comentar los mensajes u objetos en específico.

Tabla 3.114. Nomenclatura del diagrama de secuencia.

3.3.2. DIAGRAMA DE INTERACCIÓN DEL SISTEMA INFORMÁTICO

JavaServer Faces (JSF) fue diseñado integrando el patrón de diseño MVC. Esto con el fin de que las aplicaciones fueran bien diseñadas dando facilidad en su mantenimiento a través de la separación de los componentes en capas con propósitos específicos.

Generalizando el comportamiento de una aplicación que utiliza el patrón MVC su estructura se puede definir mediante los siguientes diagramas presentados desde la figura 3.16 hasta la figura 3.18:

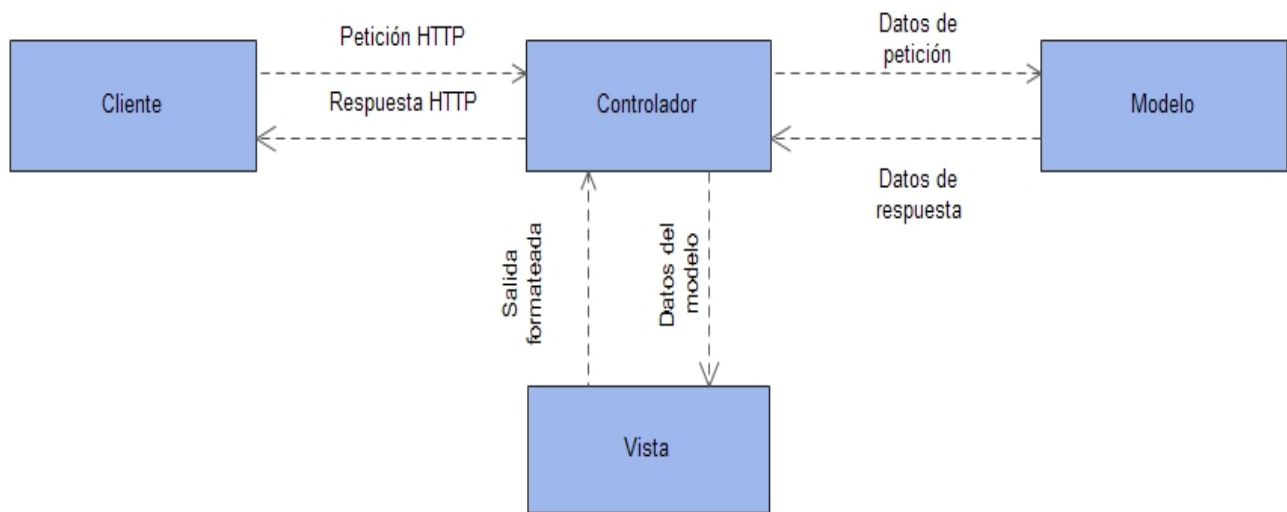


Figura 3.16. Diagrama de colaboración del patrón MVC.

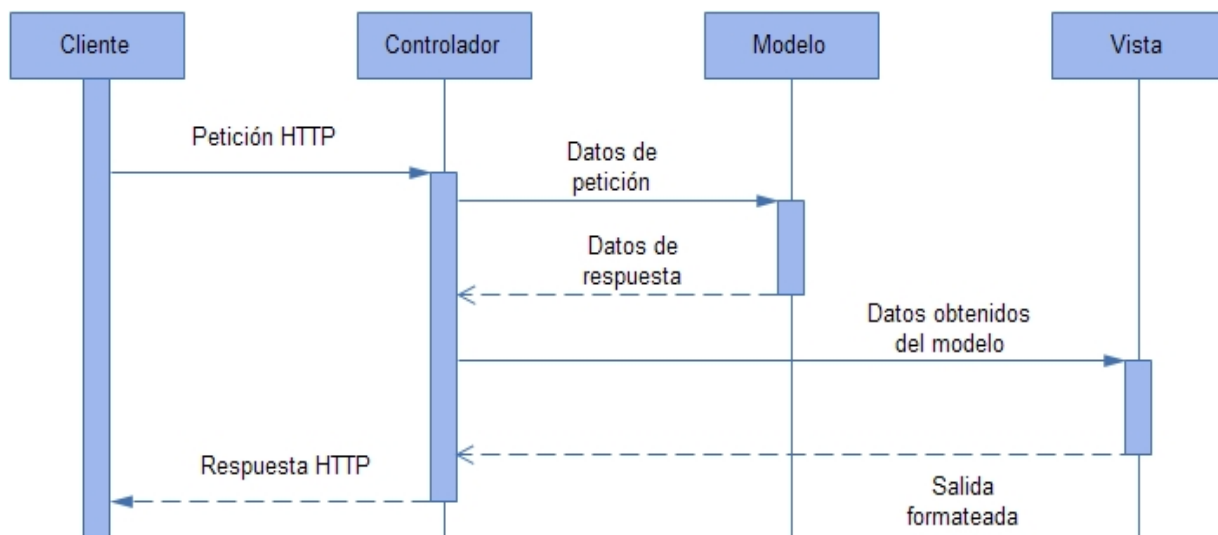


Figura 3.17. Diagrama de secuencia del patrón MVC.

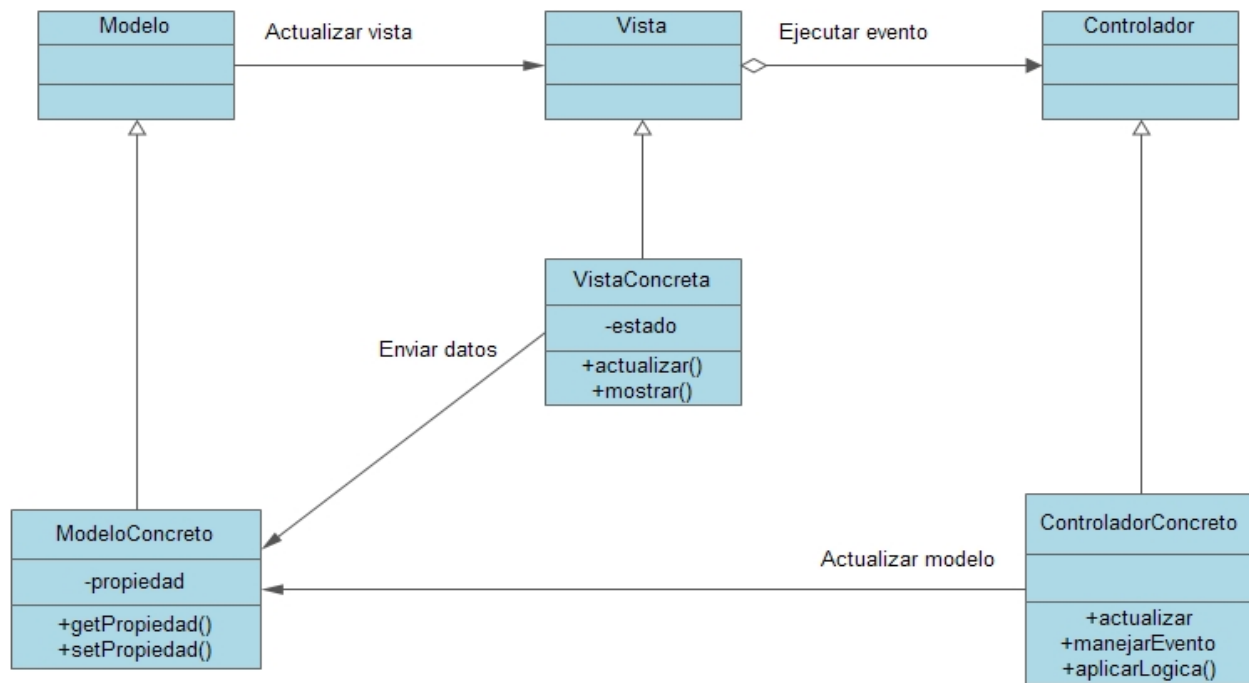


Figura 3.18. Diagrama de clases del patrón MVC.

Dado que el patrón MVC estandariza la forma en que las capas se comunican y por consiguiente los componentes dentro de ellas, el diseño de los diagramas de interacción (secuencia y colaboración) sería redundante, ya que este patrón de diseño predefine la interacción de los componentes en la aplicación y por tanto la secuencia de mensajes entre los componentes será similar. Por esta razón no se presenta el resto de diagramas de interacción.

3.4. DIAGRAMA DE CLASES

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

3.4.1. NOMENCLATURA DEL DIAGRAMA DE CLASES

La nomenclatura usada por el diagrama de clases se muestra en la tabla 3.115.a y 3.115.b.

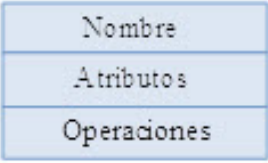


Símbolo	Nombre	Descripción
	Clase	Es el símbolo principal del diagrama de clases que encapsulan toda la información de un objeto, el cual es una instancia de una clase. Se representa de un rectángulo dividido en 3 partes. La parte superior contiene el nombre de la clase, el intermedio contiene los atributos y la inferior las operaciones que se realizan en cada clase.
	Multiplicidad	La multiplicidad indica el número de veces que una clase se asocia a otra. Puede ser de una a cero, de una a muchos como se mira en el ejemplo, de uno a uno ó se puede definir de una a varios.
	Asociación	La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre sí. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro. Puede ser unidireccional o bidireccional.

Tabla 3.115.a. Nomenclatura del diagrama de clases.

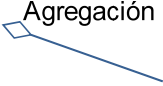
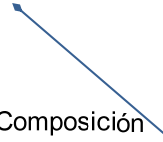
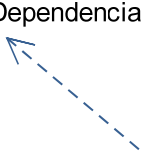
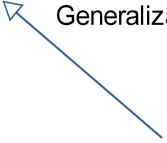
Símbolo	Nombre	Descripción
 <p>Agregación</p>	Agregación	La agregación es el termino de UML para la relación parte-todo, el diamante se coloca en el extremo que incluirá todas las clases.
 <p>Composición</p>	Composición	La composición se utiliza cuando necesitamos representar un todo y sus partes, en la composición cuando la clase que representa al todo se elimina todas las partes se eliminara.
 <p>Dependencia</p>	Dependencia	El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra.
 <p>Generalización</p>	Generalización	Indica que una subclase hereda los métodos y atributos especificados por una Súper Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Súper Clase

Tabla 3.115.b. Nomenclatura del diagrama de clases.

3.4.2. DIAGRAMA DE CLASES DEL SISTEMA INFORMÁTICO

La estructura de clases de la aplicación, por las tecnologías utilizadas en el desarrollo, se puede generalizar en 6 categorías, estas son:

- **Clases de estructura:** Sirven para definir la arquitectura del sistema y la estructura de otras clases, entre éstas están incluidas las interfaces y clases abstractas.
- **Clase de fuente de datos:** Representa la implementación de la fuente de datos. Una fuente de datos podría ser una base de datos como un RDBMS, un OODBMS, un repositorio XML, un archivo plano, fuentes de datos externas, etc.

- **Clase de persistencia:** Representa una clase utilizada para el transporte de datos. Una clase de acceso a datos podría utilizar una clase de persistencia para devolver los datos al cliente. La clase de acceso a datos también podría recibir datos desde el cliente en una clase de transferencia para actualizar los datos en la fuente de datos.
- **Clase de acceso a datos:** Abstrae la implementación del acceso a datos subyacente a la clase de negocios para permitirle un acceso transparente a la fuente de datos. La clase de negocios también delega las operaciones de carga y almacenamiento en la clase de acceso a datos.
- **Clase de negocios:** Representa los datos del cliente y la lógica de negocios para procesar estos datos. Es el objeto que requiere el acceso a la fuente de datos para obtener y almacenar datos.
- **Clases de formato:** Estas clases sirven para la conversión y validación de los datos ingresados por el usuario, están encargadas de proporcionar el formato adecuado de los datos para poder ser procesados y almacenados.

El diagrama de clases parcial correspondiente al modelo de los datos del sistema se presenta en la figura 3.19 y 3.20.

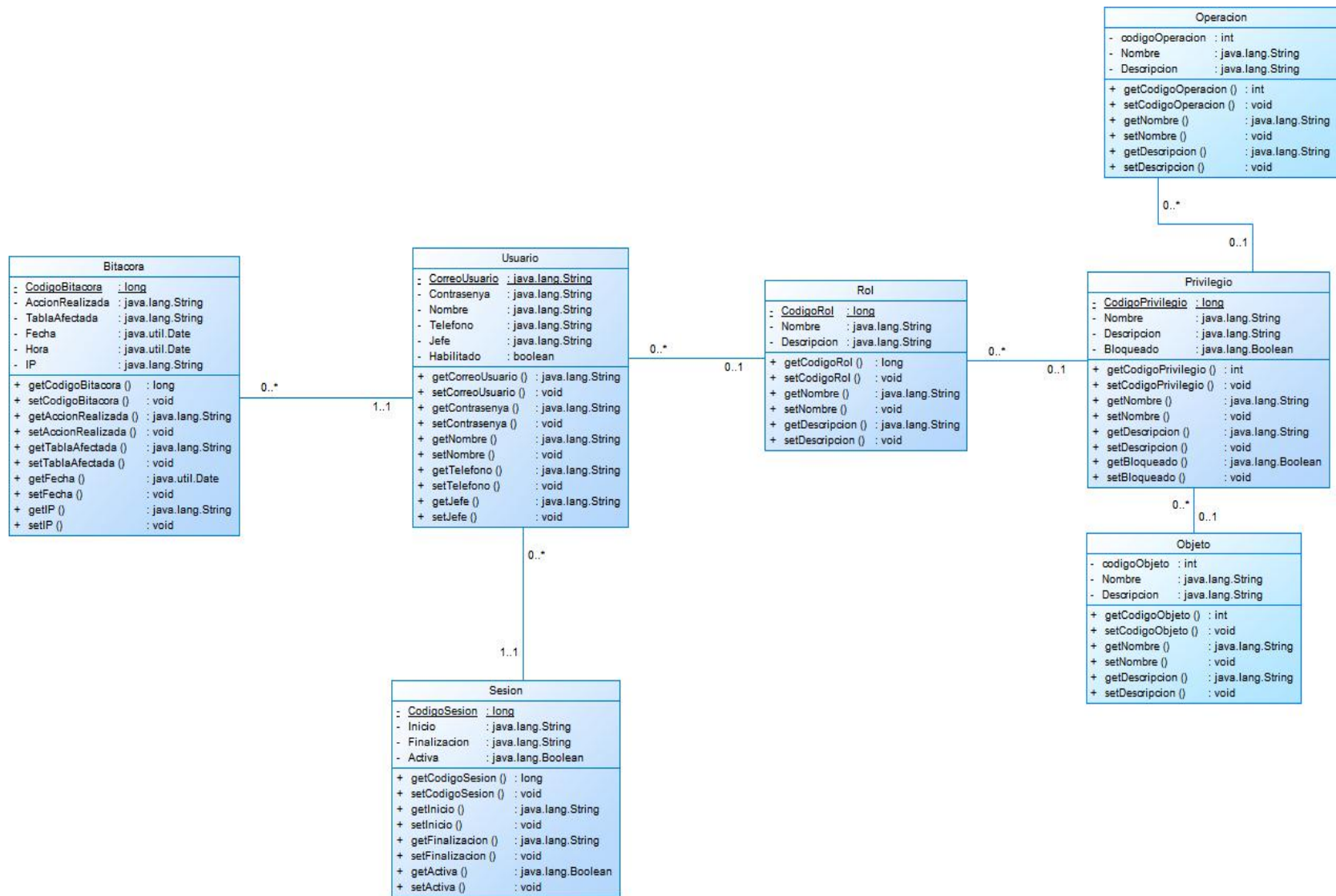


Figura 3.19. Diagrama de clases parcial del esquema de seguridad.

A partir del diagrama de clases parcial anterior y las clasificaciones de las clases, el diagrama de clases generalizado del sistema se presenta en la figura 3.21.

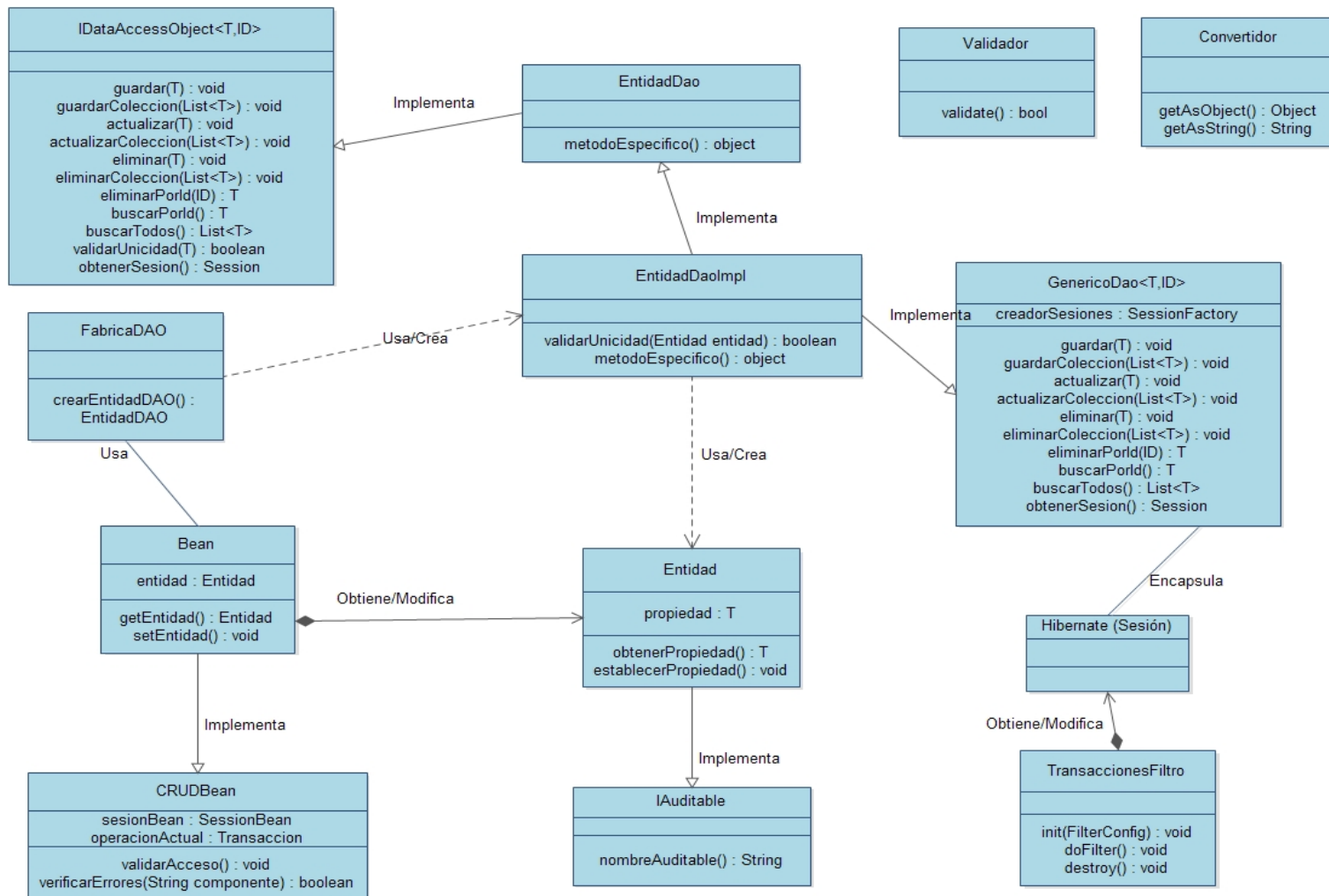


Figura 3.21. Diagrama de clases generalizado.

Las clases de estructura están representadas únicamente por la clase *IDataAccessObject*, la cual proveerá la estructura que deberán tener las clases *EntidadDAO* con el objetivo de implementar el patrón DAO para el acceso a los datos.

Las clases de fuente de datos serán provistas por el componente ORM *Hibernate*. Este componente será el encargado de proveer el acceso a la base de datos del sistema.

Las clases de persistencia están englobadas en el diagrama como la clase *Entidad*. Estas clases serán los mapeos de las tablas de la base de datos como clases de transferencia. Estas clases tendrán por la tanto una equivalencia de 1 a 1 con las tablas de la base de datos del sistema.

Las clases de acceso de datos se representan con la clase *EntidadDAO*. Esta categoría de clases brindará acceso estandarizado a las fuentes de datos y contendrán las operaciones para poder manipular los datos contenidos en las clases de persistencia. Habrá una clase DAO para operar sobre cada clase *Entidad*, por tanto, habrá una equivalencia de 1 clase DAO por cada clase *Entidad* que proveerá los métodos específicos para la entidad definidos por la interfaz *IDataAccessObject*.

Las clases de negocios están presentadas como la clase *Bean*, que es una generalización de los *ManagedBeans* de JSF. Habrá una equivalencia de al menos un *ManagedBean* por cada página a desarrollar. Las clases *Bean* a desarrollar con sus respectivas clases de entidad asociadas se muestran en la tabla 3.116.a y 3.116.b:

Clases Bean	Clases Asociadas
<i>SesionBean</i>	• <i>Usuario</i>
<i>UsuarioBean</i>	• <i>Usuario</i>
<i>RolBean</i>	• <i>Rol</i> • <i>Privilegio</i>
<i>ObjetoBean</i>	• <i>Objeto</i>
<i>OperacionBean</i>	• <i>Operación</i>
<i>BitacoraBean</i>	• <i>Bitacora</i>
<i>PhylumBean</i>	• <i>Phylum</i>
<i>ClaseBean</i>	• <i>Clase</i>
<i>OrdenBean</i>	• <i>Orden</i>
<i>Súper-familiaBean</i>	• <i>Súper-familia</i>
<i>FamiliaBean</i>	• <i>Familia</i>
<i>GeneroBean</i>	• <i>Genero</i>
<i>EspecieBean</i>	• <i>Especie</i>

Tabla 3.116.a. Clases Bean y sus clases asociadas.

Clases Bean	Clases Asociadas
<i>AnimalBean</i>	<ul style="list-style-type: none"> • <i>Animal</i> • <i>Sinonimia</i> • <i>Resenya</i>
<i>RecolectorBean</i>	<ul style="list-style-type: none"> • <i>Recolector</i>
<i>RecoleccionBean</i>	<ul style="list-style-type: none"> • <i>Recoleccion</i> • <i>Nacional</i> • <i>Internacional</i>
<i>EspecimenBean</i>	<ul style="list-style-type: none"> • <i>Especimen</i> • <i>Terrestre</i> • <i>Acuatico</i>
<i>RestauracionBean</i>	<ul style="list-style-type: none"> • <i>Restauracion</i>
<i>DepartamentoBean</i>	<i>Departamento</i>
<i>MunicipioBean</i>	<i>Municipio</i>
<i>ZonaBean</i>	<ul style="list-style-type: none"> • <i>Zona</i>
<i>AreaprotegidaBean</i>	<ul style="list-style-type: none"> • <i>AreaProtegida</i>
<i>ImportarBean</i>	-
<i>ExportarBean</i>	
<i>ConsultaBean</i>	-
<i>ReporteBean</i>	-

Tabla 3.116.b. Clases Bean y sus clases asociadas.

Las clases de formato están representadas a través de las clases *Convertidor* y *Validador*, éstas serán implementadas de la siguiente forma:

- Convertidores:
 - *CoordenadasConverter*
- Validadores:
 - *CoordenadasValidator*

3.5. DIAGRAMA DE COMPONENTES

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes.

Debido a que los diagramas de componentes son más parecidos a los diagramas de casos de usos, éstos son utilizados para modelar la vista estática y dinámica de un sistema. Muestra la organización y las dependencias entre un conjunto de componentes. No es necesario que un diagrama incluya todos los componentes del sistema, normalmente se realizan por partes. Cada diagrama describe un apartado del sistema. En él se situarán librerías, tablas, archivos, ejecutables y documentos que formen parte del sistema.

3.5.1. NOMENCLATURA DEL DIAGRAMA DE COMPONENTES

El diagrama de componentes usa la nomenclatura mostrada en la tabla 3.117.





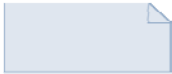
Símbolo	Nombre	Descripción
	Componente	Es el símbolo principal del diagrama de componentes representa cada elemento del diagrama.
	Interfaz	Representa las interfaces que se incluirán en los componentes.
	Flecha discontinua	La punta donde se encuentra la flecha indica que elemento es dependiente de otro.
	Librerías	Se utiliza para las librerías que incluirá el diagrama.
	Restricción	Representa comentarios o restricciones que se tengan en los componentes.

Tabla 3.117. Nomenclatura del diagrama de componentes.

3.5.2. DIAGRAMAS DE COMPONENTES DEL SISTEMA INFORMÁTICO

- Diagrama de componentes del sistema informático:

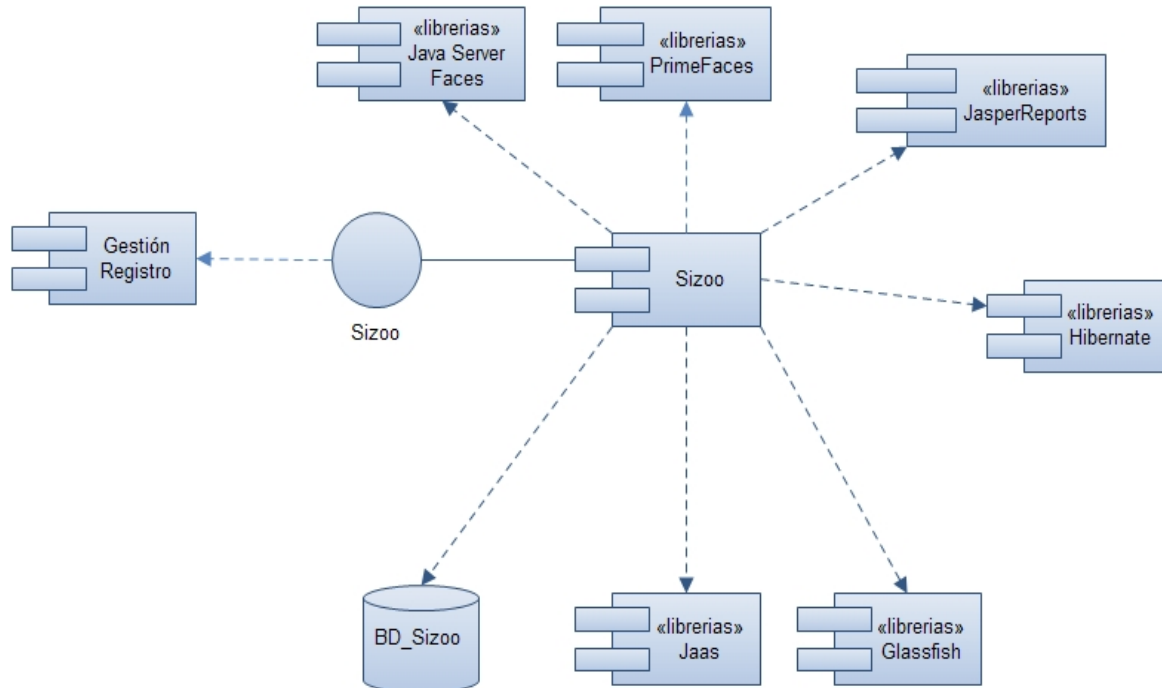


Figura 3.22. Diagrama de componentes general del sistema informático.

- Diagrama de componentes de la sesión de usuario:

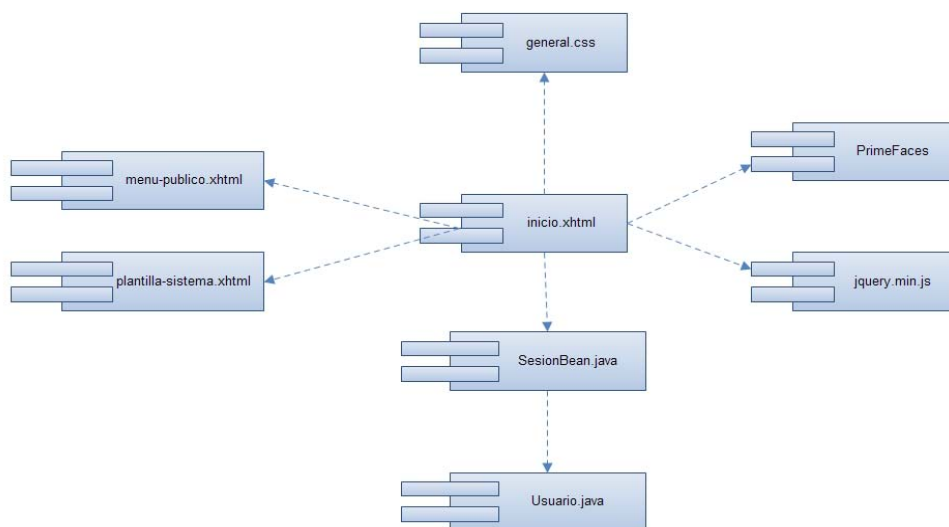


Figura 3.23. Diagrama de componentes de la sesión de usuario.

- **Diagrama de componentes del usuario:**

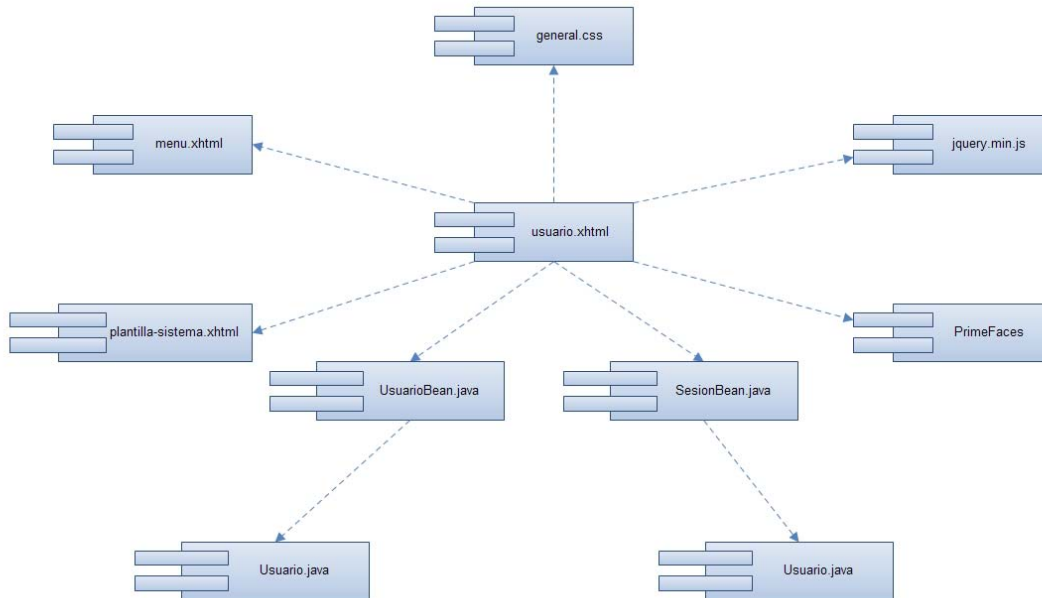


Figura 3.24. Diagrama de componentes del usuario.

- **Diagrama de componentes de los roles:**

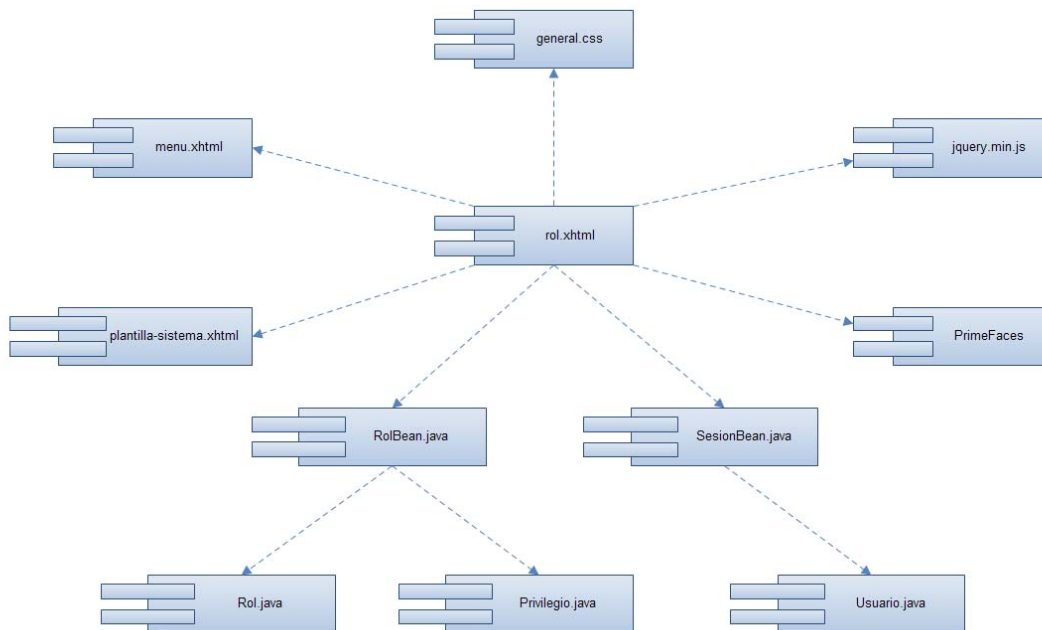


Figura 3.25. Diagrama de componentes de los roles.

- **Diagrama de componentes de bitácora:**

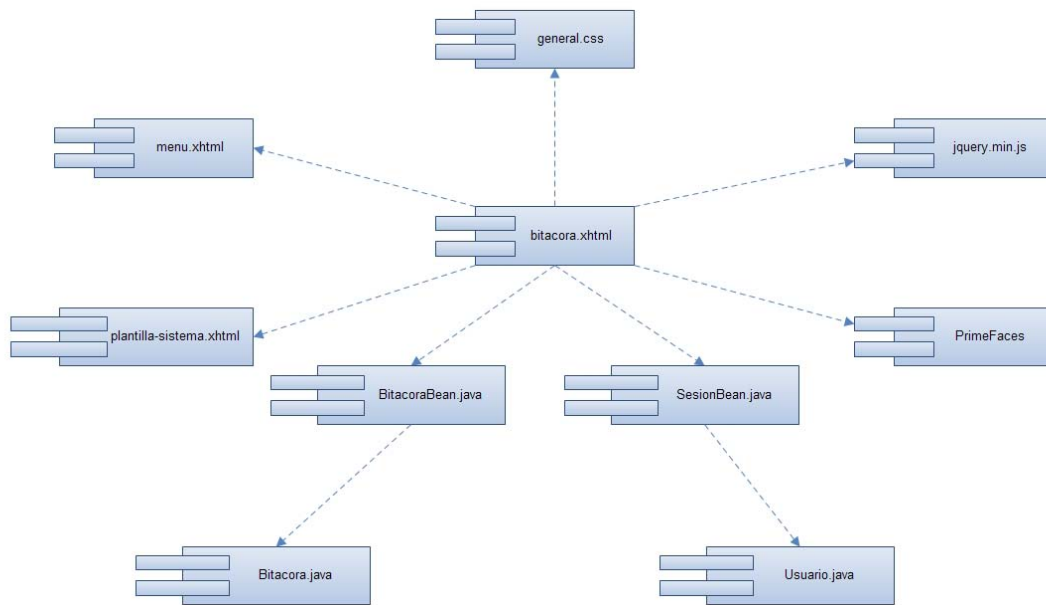


Figura 3.26. Diagrama de componentes de bitácora.

- **Diagrama de componentes del phylum:**

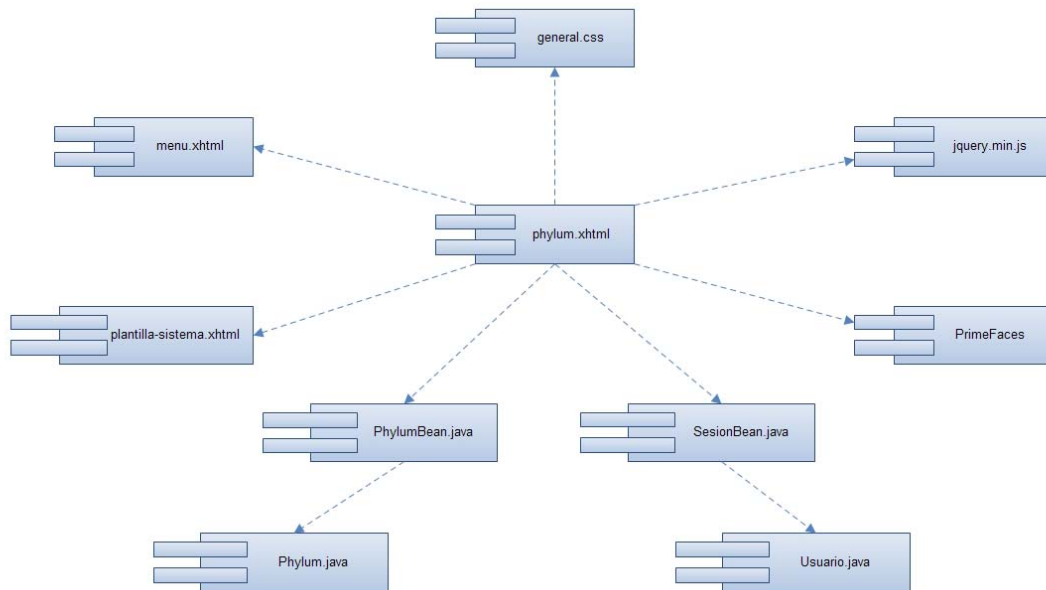


Figura 3.27. Diagrama de componentes del phylum.

- **Diagrama de componentes de la clase:**

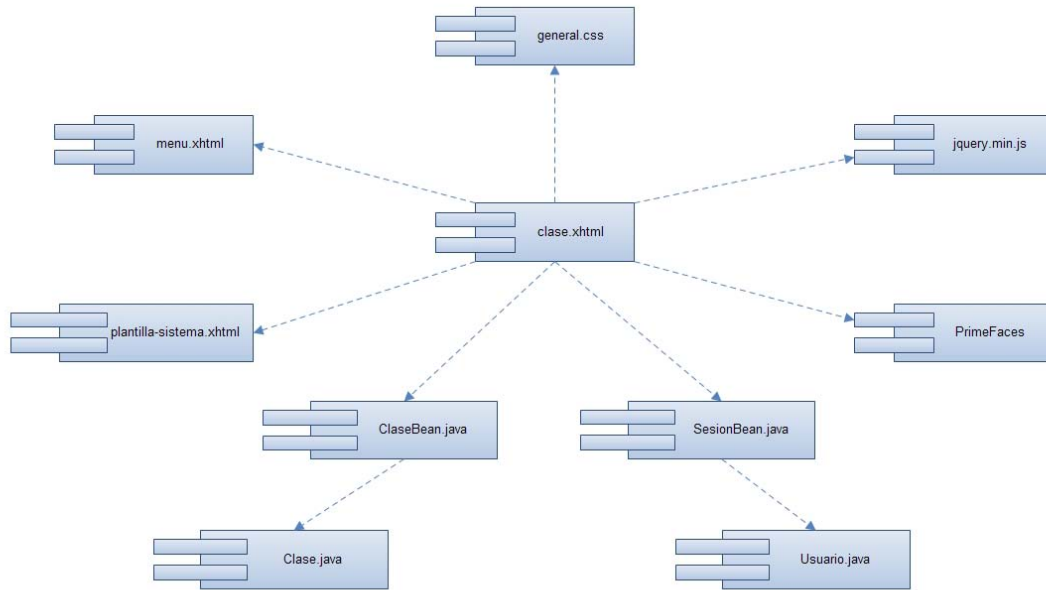


Figura 3.28. Diagrama de componentes de la clase.

- **Diagrama de componentes del orden:**

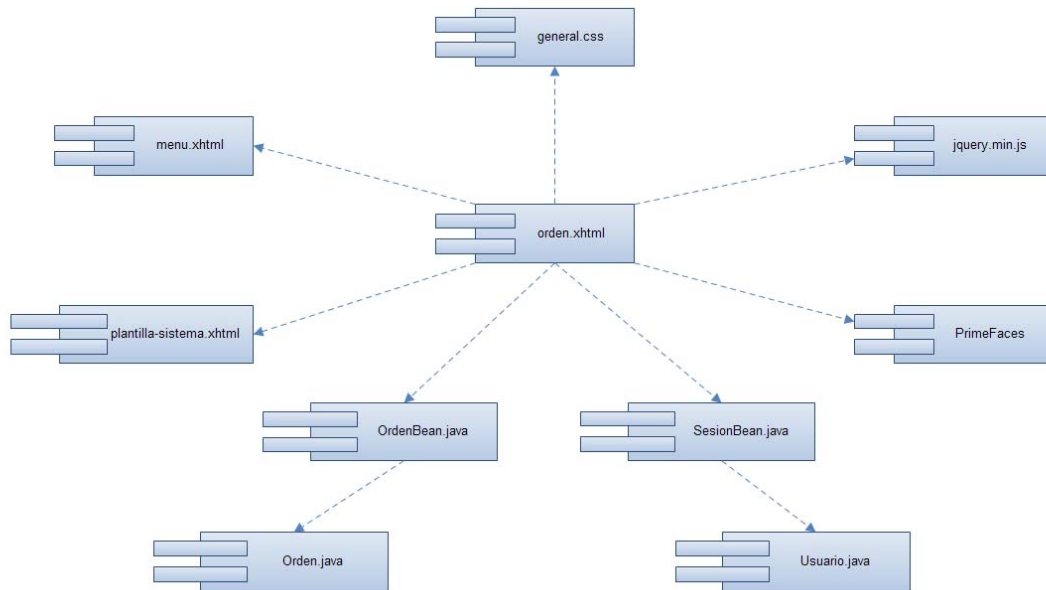


Figura 3.29. Diagrama de componentes del orden.

- **Diagrama de componentes de la súper-familia:**

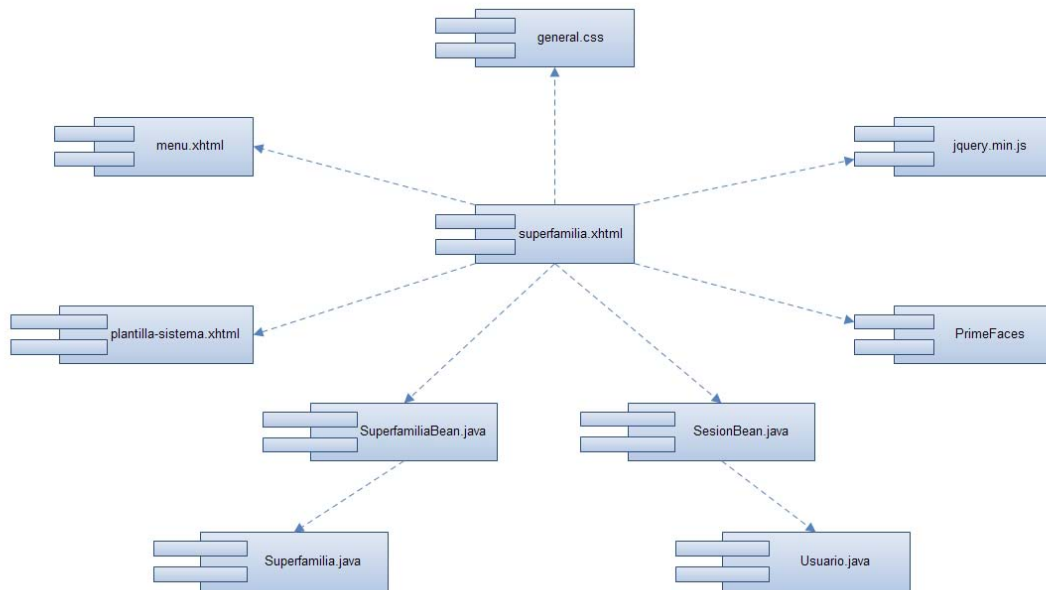


Figura 3.30. Diagrama de componentes de la súper-familia.

- **Diagrama de componentes de la familia:**

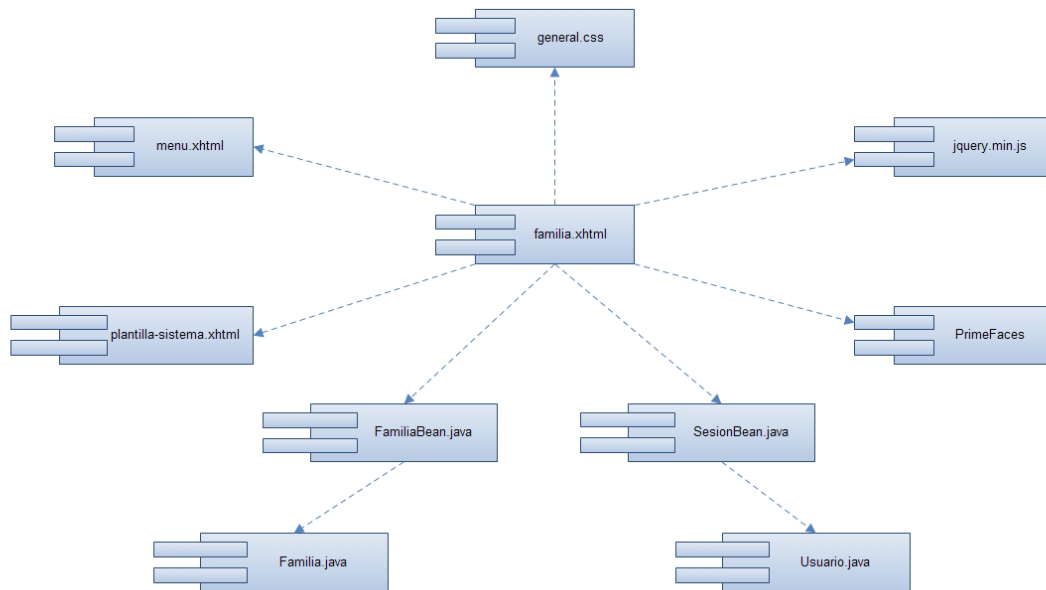


Figura 3.31. Diagrama de componentes de la familia.

- **Diagrama de componentes del género:**

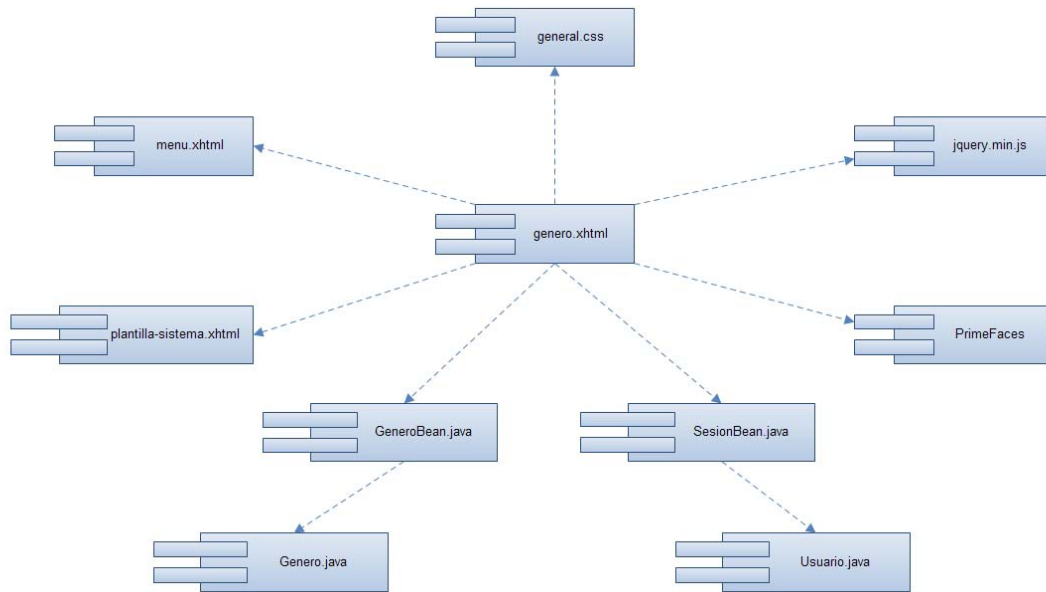


Figura 3.32. Diagrama de componentes del género.

- **Diagrama de componentes de la especie:**

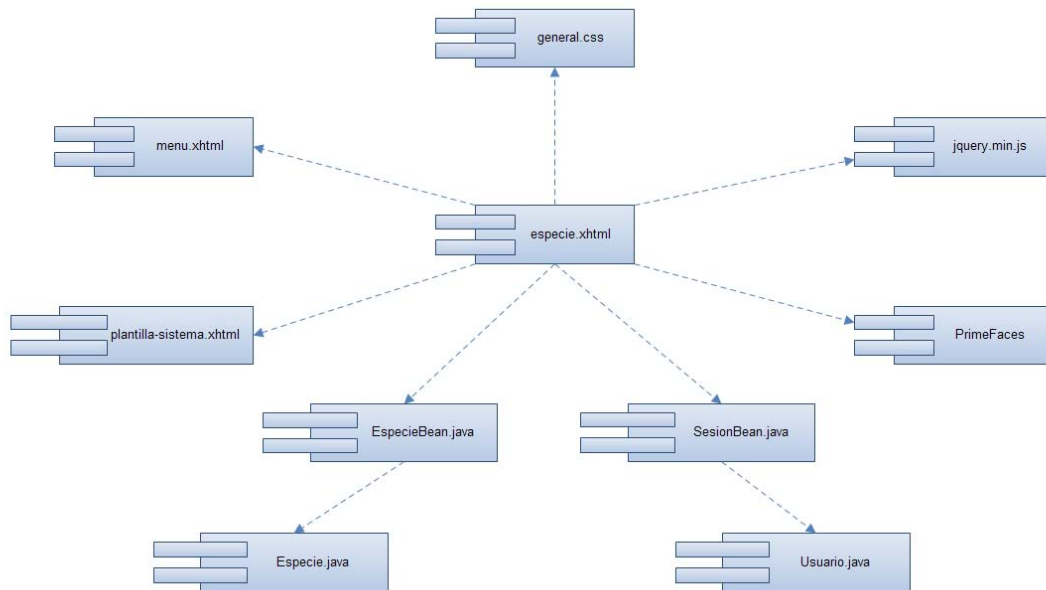


Figura 3.33. Diagrama de componentes de la especie.

- **Diagrama de componentes de animales:**

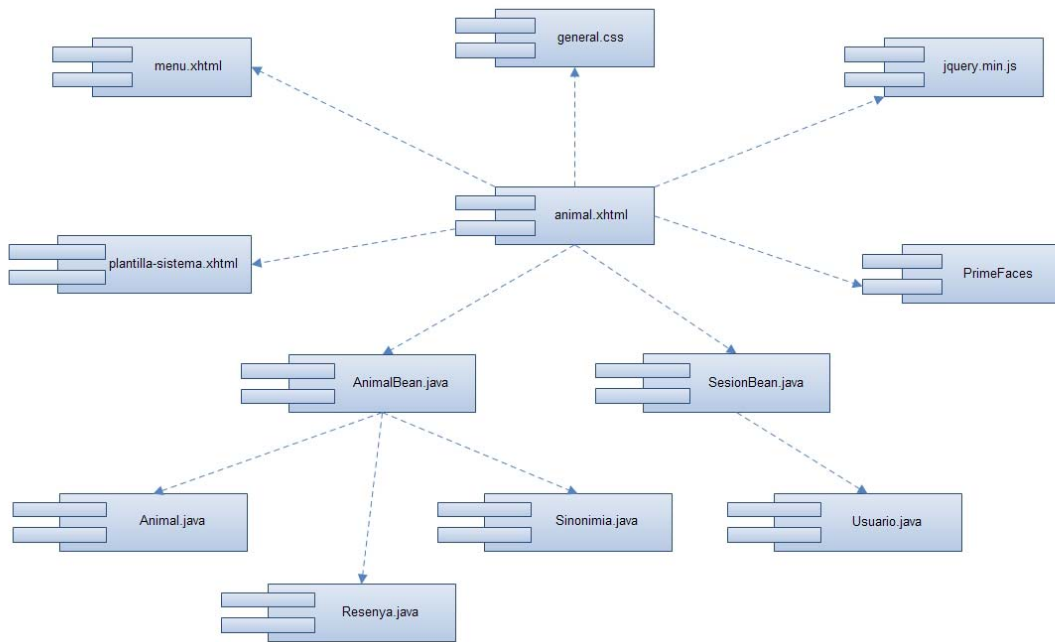


Figura 3.34. Diagrama de componentes de animales.

- **Diagrama de componentes de los especímenes:**

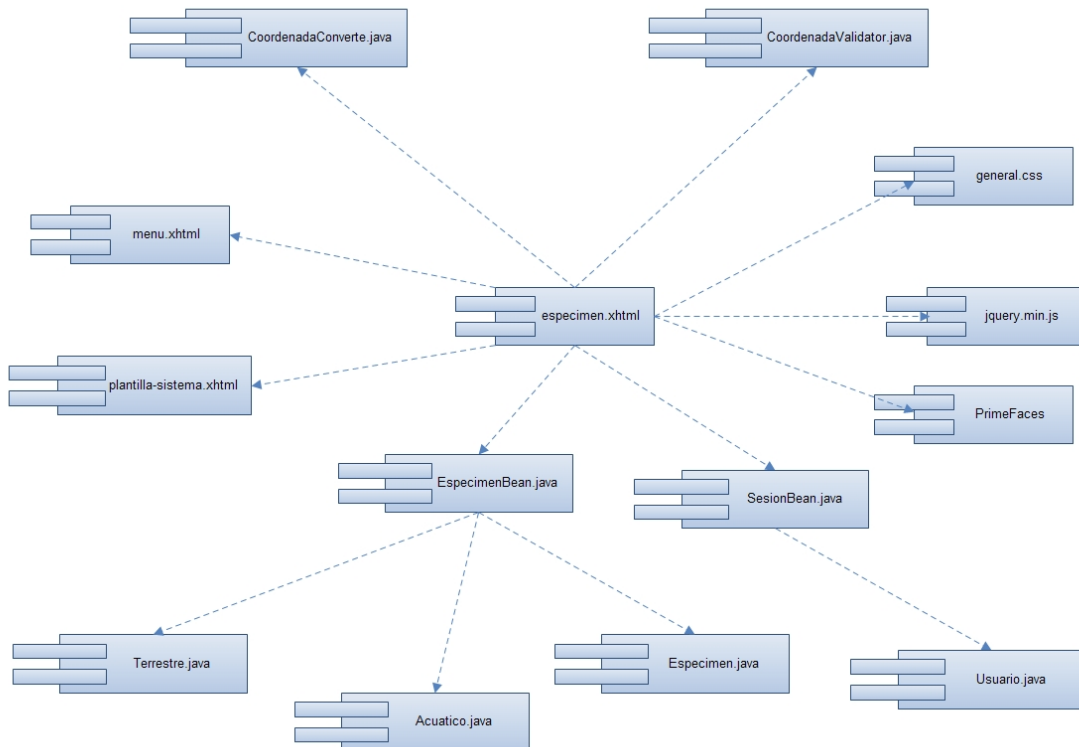


Figura 3.35. Diagrama de componentes de los especímenes.

- **Diagrama de componentes de los recolectores:**

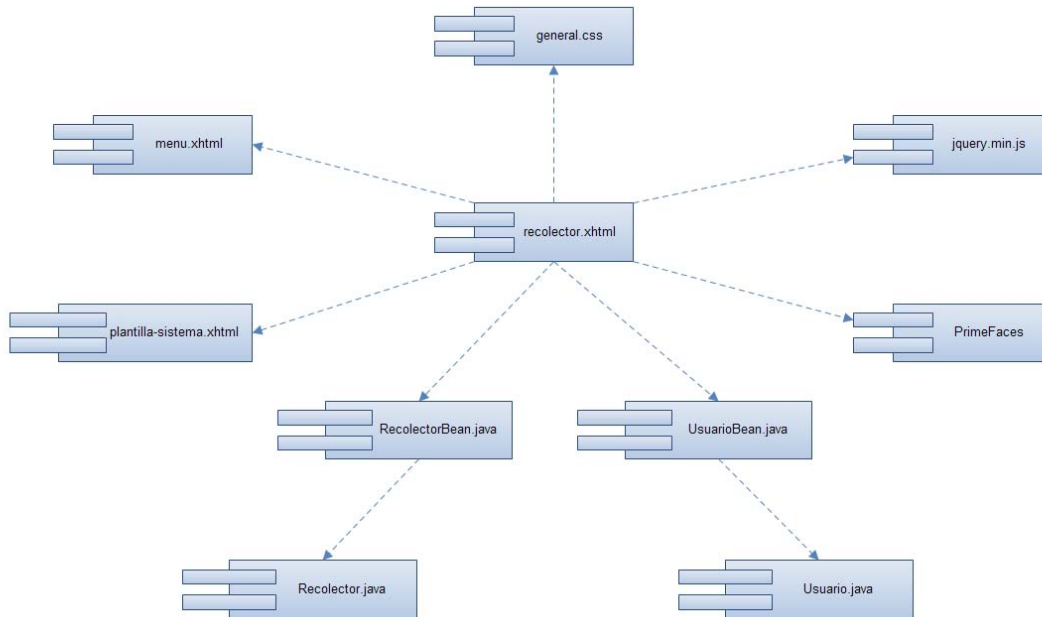


Figura 3.36. Diagrama de componentes de los recolectores.

- **Diagrama de componentes de las restauraciones de los especímenes:**

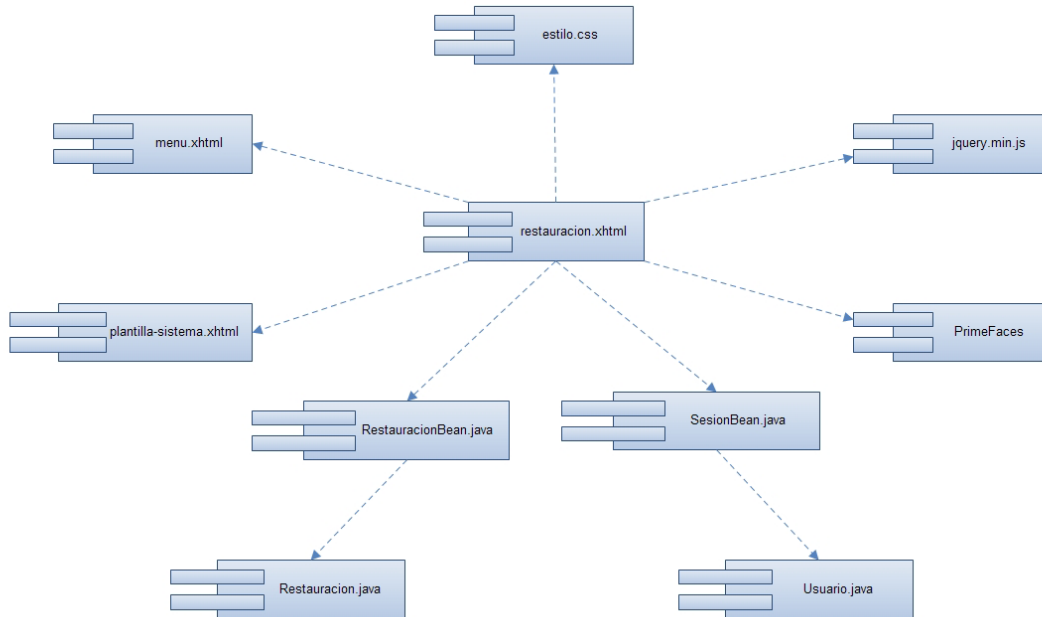


Figura 3.37. Diagrama de componentes de las restauraciones de los especímenes.

- **Diagrama de componentes de las recolecciones:**

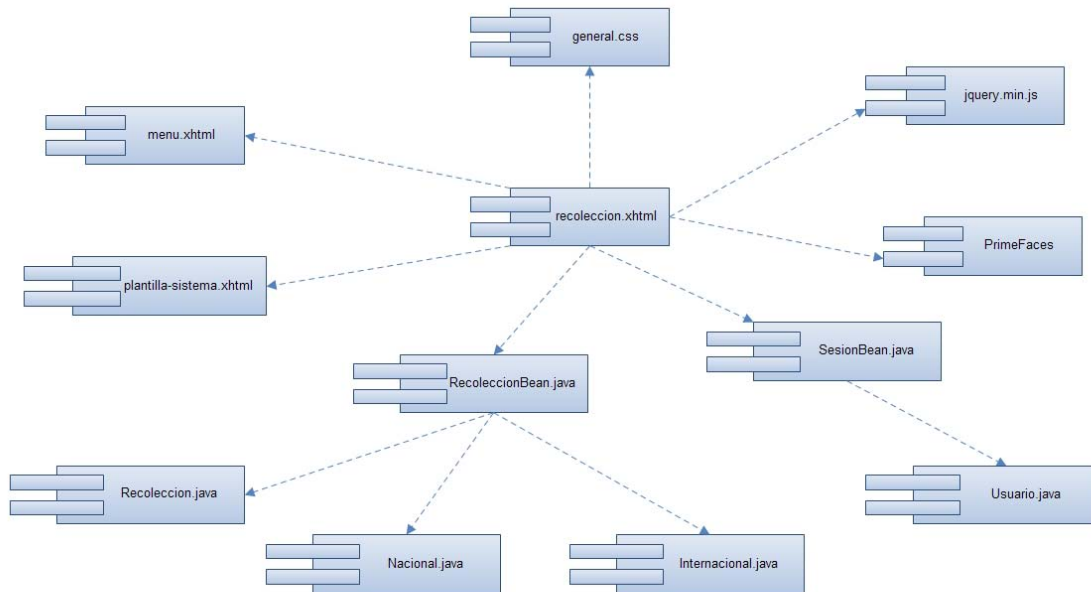


Figura 3.38. Diagrama de componentes de las recolecciones.

- **Diagrama de componentes del departamento:**

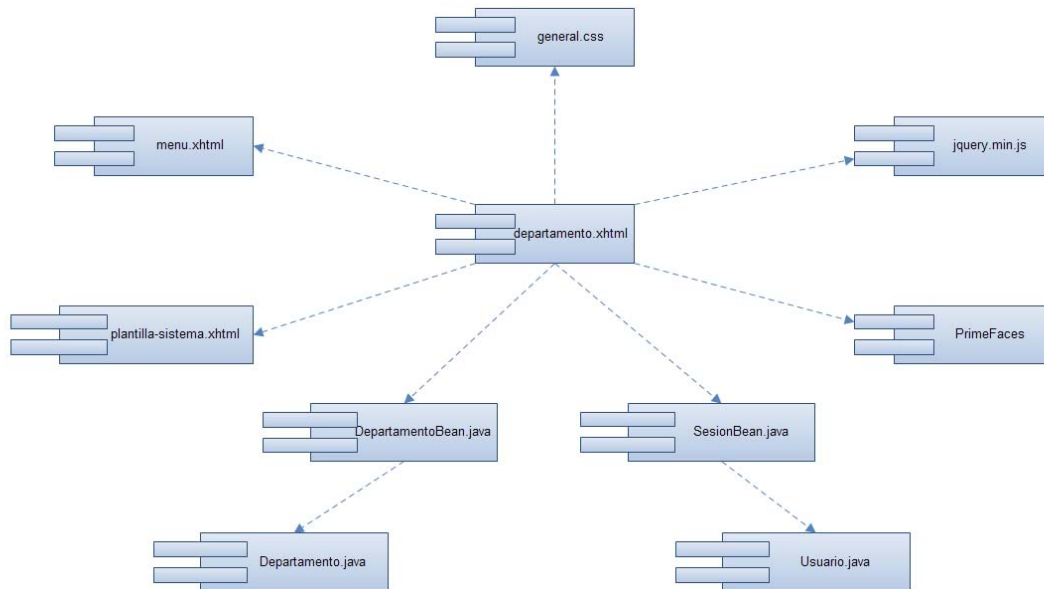


Figura 3.39. Diagrama de componentes del departamento.

- **Diagrama de componentes del municipio:**

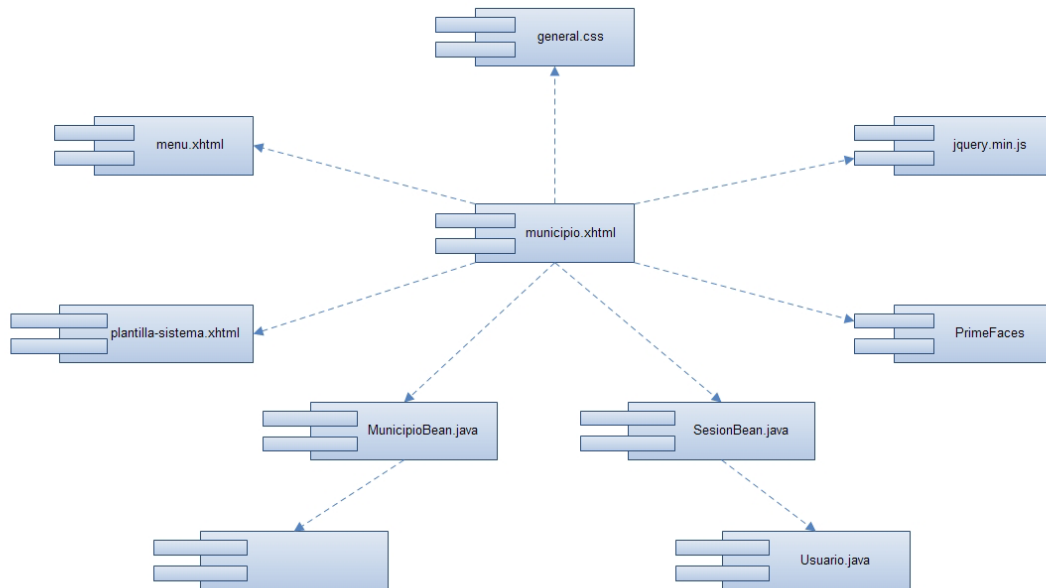


Figura 3.40. Diagrama de componentes del municipio.

- **Diagrama de componentes de la zona:**

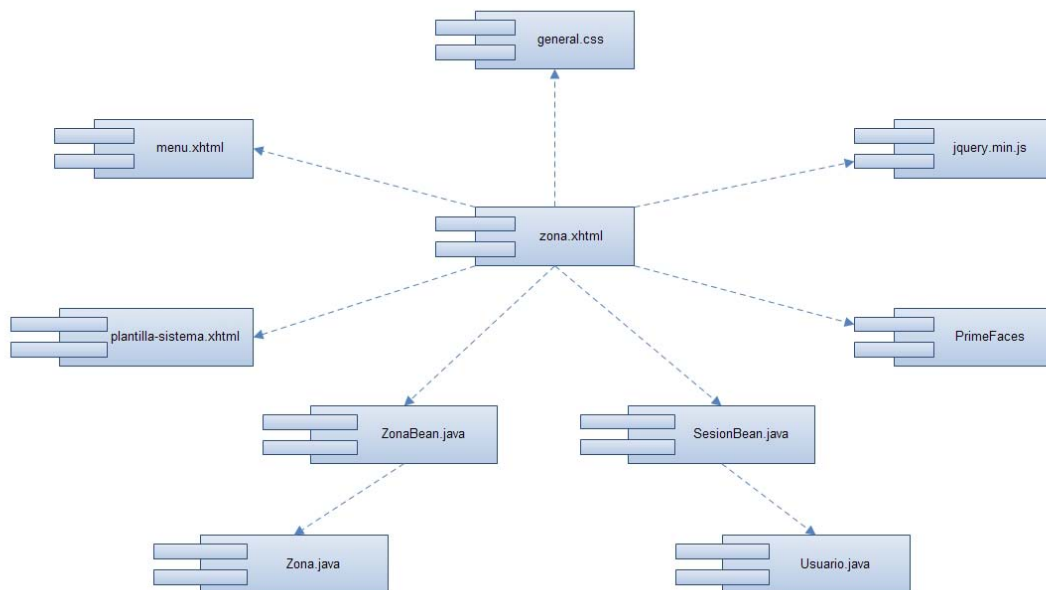


Figura 3.41. Diagrama de componentes de la zona.

- **Diagrama de componentes del área protegida:**

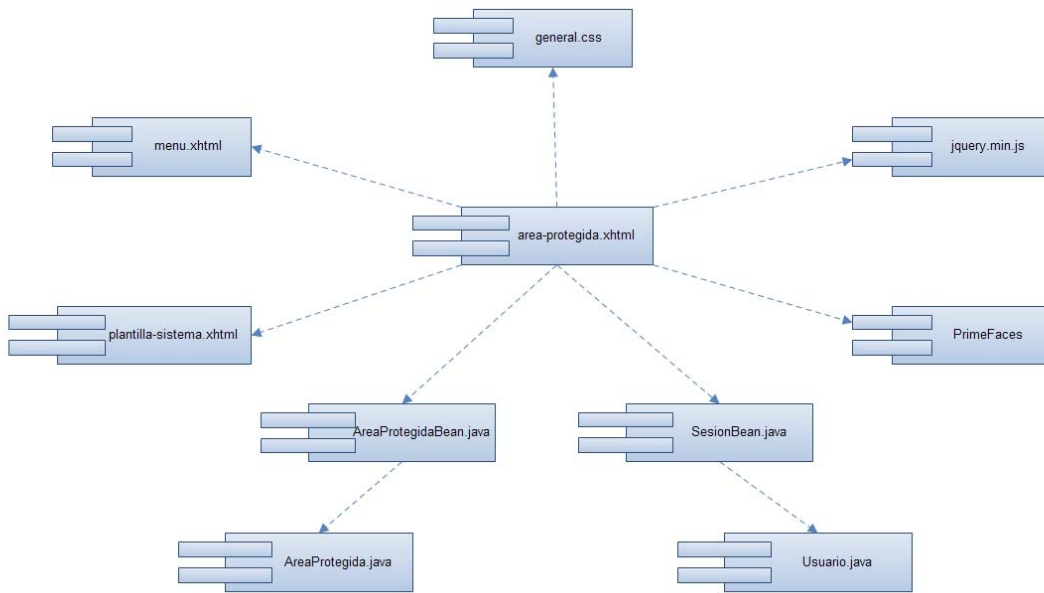


Figura 3.42. Diagrama de componentes del área protegida.

- **Diagrama de componentes de importación de catálogos:**

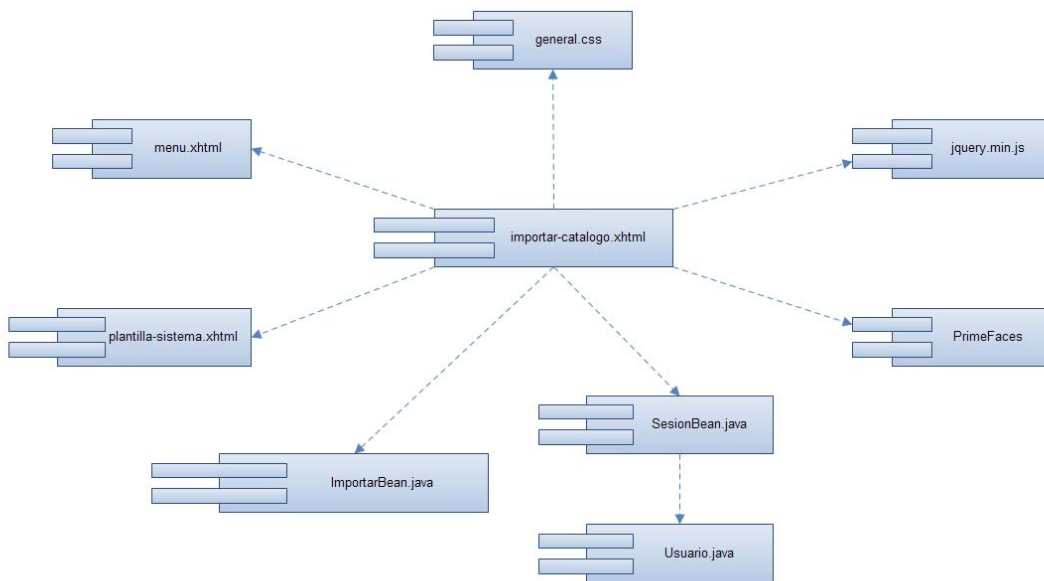


Figura 3.43. Diagrama de componentes de importación de catálogos.

- **Diagrama de componentes de exportación de catálogos:**

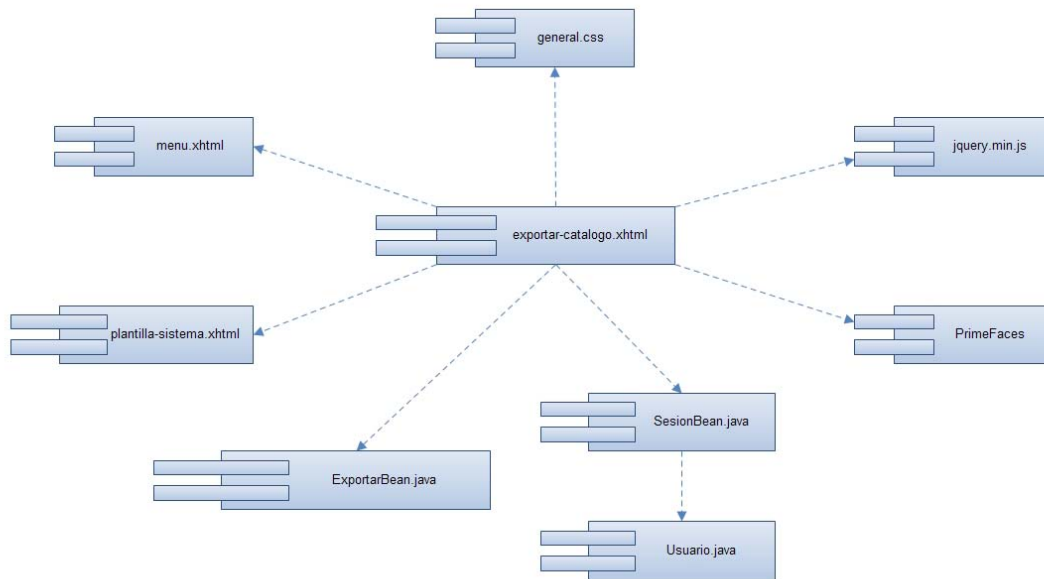


Figura 3.44. Diagrama de componentes de exportación de catálogos.

- **Diagrama de componentes de reportes:**

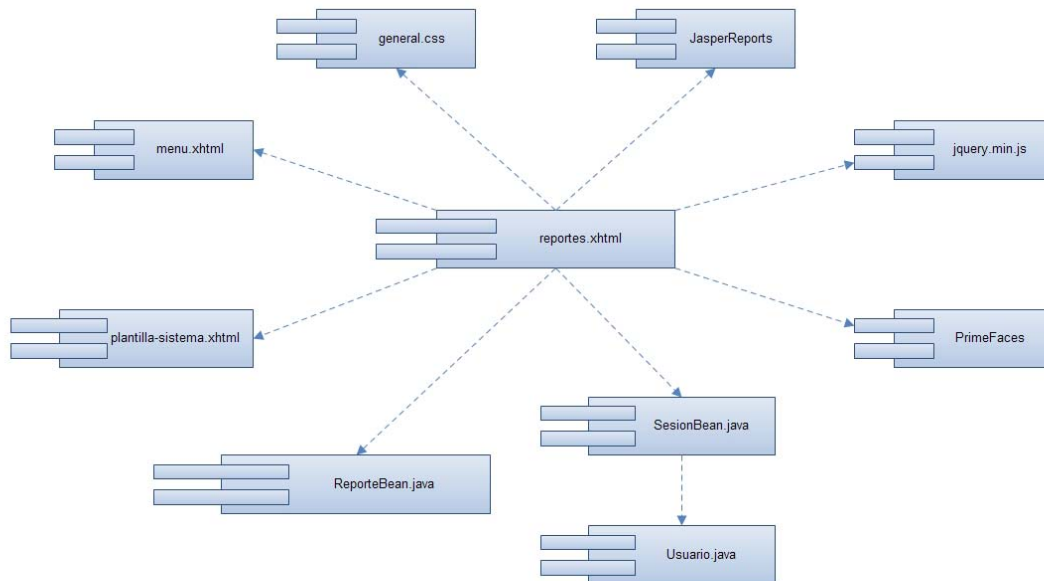


Figura 3.45. Diagrama de componentes de reportes.

3.6. DIAGRAMAS DE DESPLIEGUE

El diagrama de despliegue es un tipo de diagrama de UML que se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes. Es decir, esquematiza los elementos físicos y lógicos de un sistema.

3.6.1. NOMENCLATURA DEL DIAGRAMA DE DESPLIEGUE

El diagrama de despliegue usa la nomenclatura mostrada en la tabla 3.118.

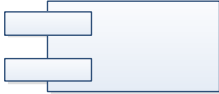
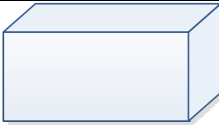


Símbolo	Nombre	Descripción
	Componente	Describe recursos de software del sistema.
	Nodo	Describe un recurso computacional.
	Comunicación	Indica una comunicación entre dos o más nodos.
	Dependencia en dos vías	Indica la dependencia en ambos sentidos entre varios recursos de software y/o hardware.

Tabla 3.118. Nomenclatura del diagrama de despliegue.

3.6.2. DIAGRAMA DE DESPLIEGUE DEL SISTEMA INFORMÁTICO

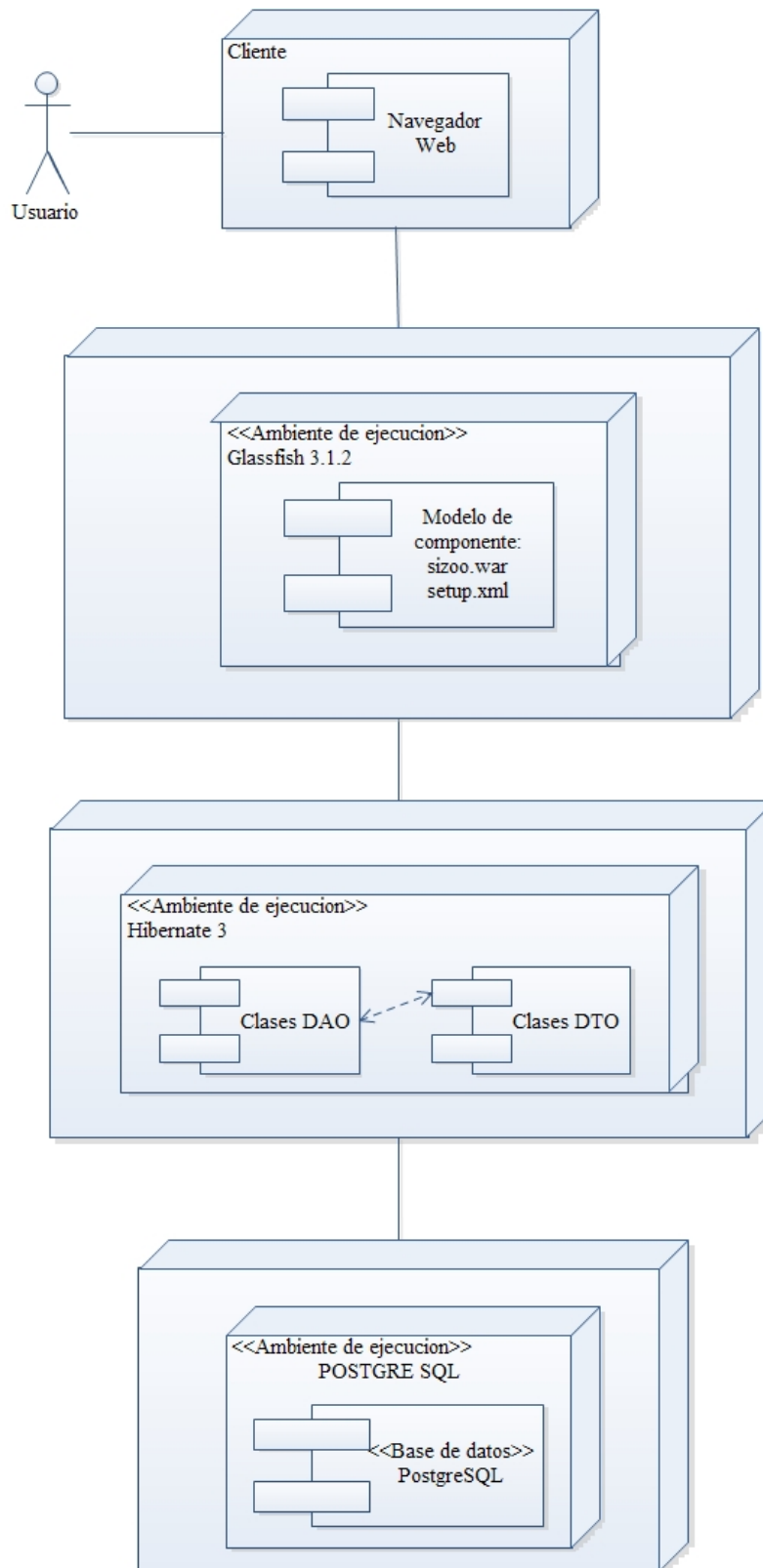


Figura 3.46. Diagrama de despliegue del sistema informático.

3.7. BASE DE DATOS

La base de datos es un sistema que almacena datos que están relacionados entre sí, donde se puede almacenar y recuperar información.

El diseño de la base de datos está conformado por los siguientes modelos:

- Modelo entidad relación.
- Modelo conceptual.
- Modelo lógico.
- Modelo físico.

3.7.1. MODELO ENTIDAD-RELACIÓN

El diagrama entidad-relación (abreviado como DER) es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

La simbología a utilizar para representar el DER se presenta en la tabla 3.119.



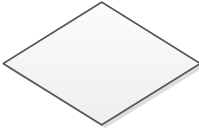

Símbolo	Nombre	Descripción
	Entidad	Representa una cosa u objeto del mundo real con existencia independiente, es decir, se diferencia unívocamente de otro objeto o cosa, incluso siendo del mismo tipo, o una misma entidad.
	Atributo	Los atributos son las características que definen o identifican a una entidad. Si esta subrayado indica que es un atributo único que diferencia a la entidad de las demás.
	Relación	Describe cierta dependencia entre entidades o permite la asociación de las mismas. La relación puede ser uno a uno (1:1), uno a muchos (1:N), muchos a uno (N:1) y muchos a muchos (M:N).
	Disjunción	Indica que las entidades hijas tienen dependencia directa de una entidad padre.

Tabla 3.119. Simbología del diagrama entidad-relación.

En las figuras 3.47 y 3.48 se muestran los diagramas entidad-relación del sistema SIZOO.

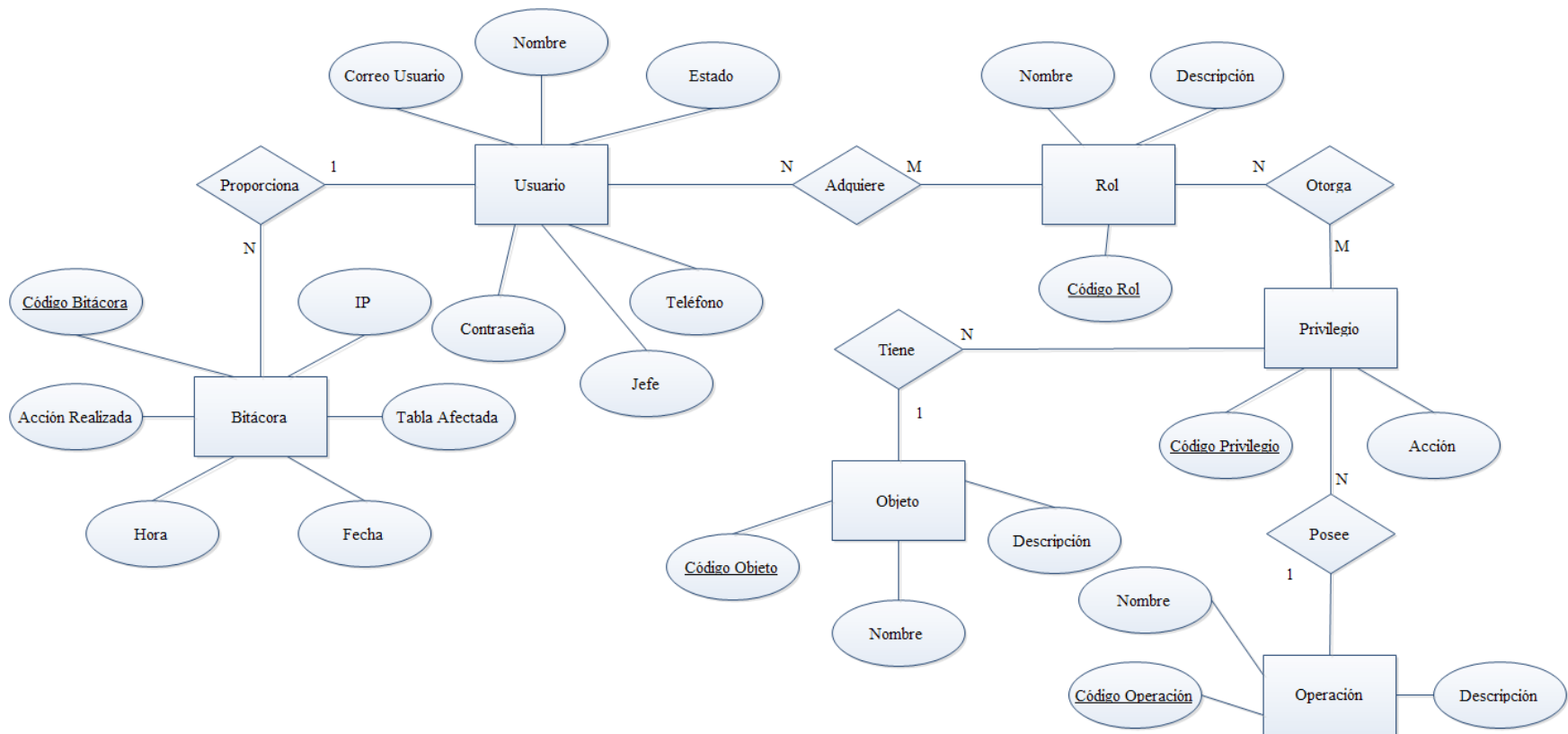


Figura 3.47. Modelo entidad-relación del esquema de seguridad.

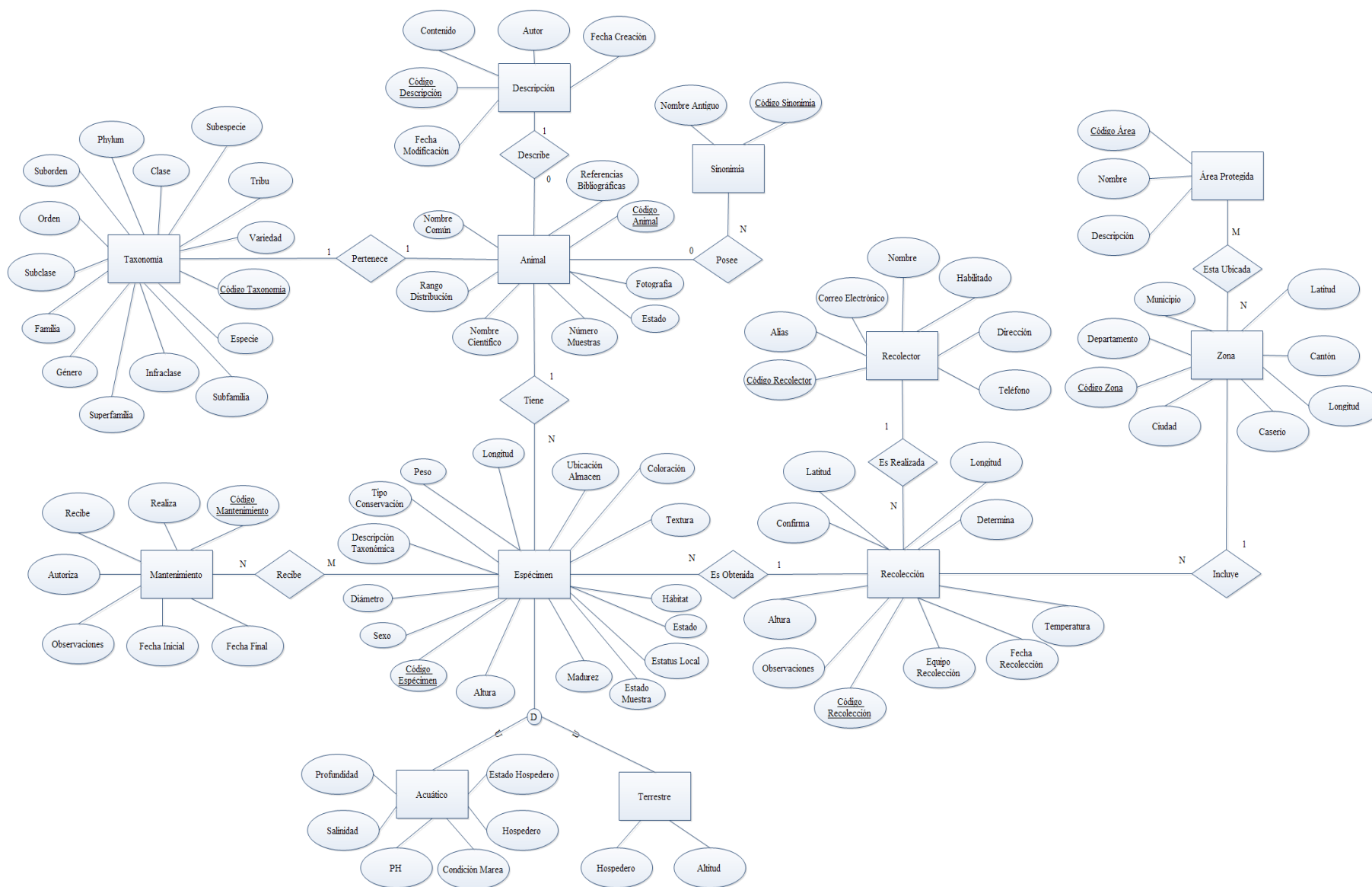


Figura 3.48. Modelo entidad-relación del esquema de zoología.

3.7.2. MODELO CONCEPTUAL

El modelo conceptual explica cuales son y cómo se relacionan los conceptos relevantes en la descripción del problema. En las figuras 3.49 y 3.50 se muestran los modelos conceptuales del sistema SIZOO.

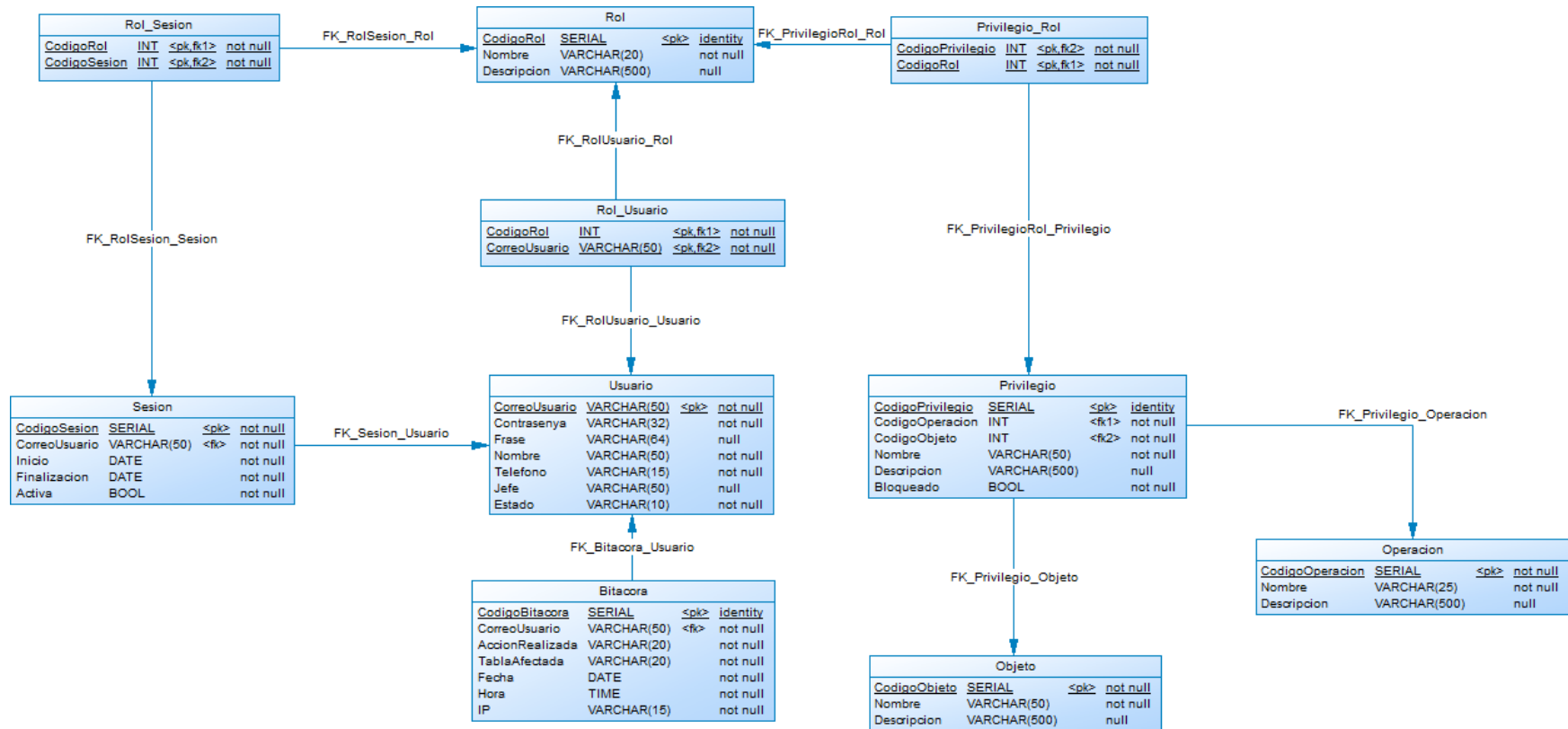


Figura 3.49. Modelo conceptual del esquema de seguridad.

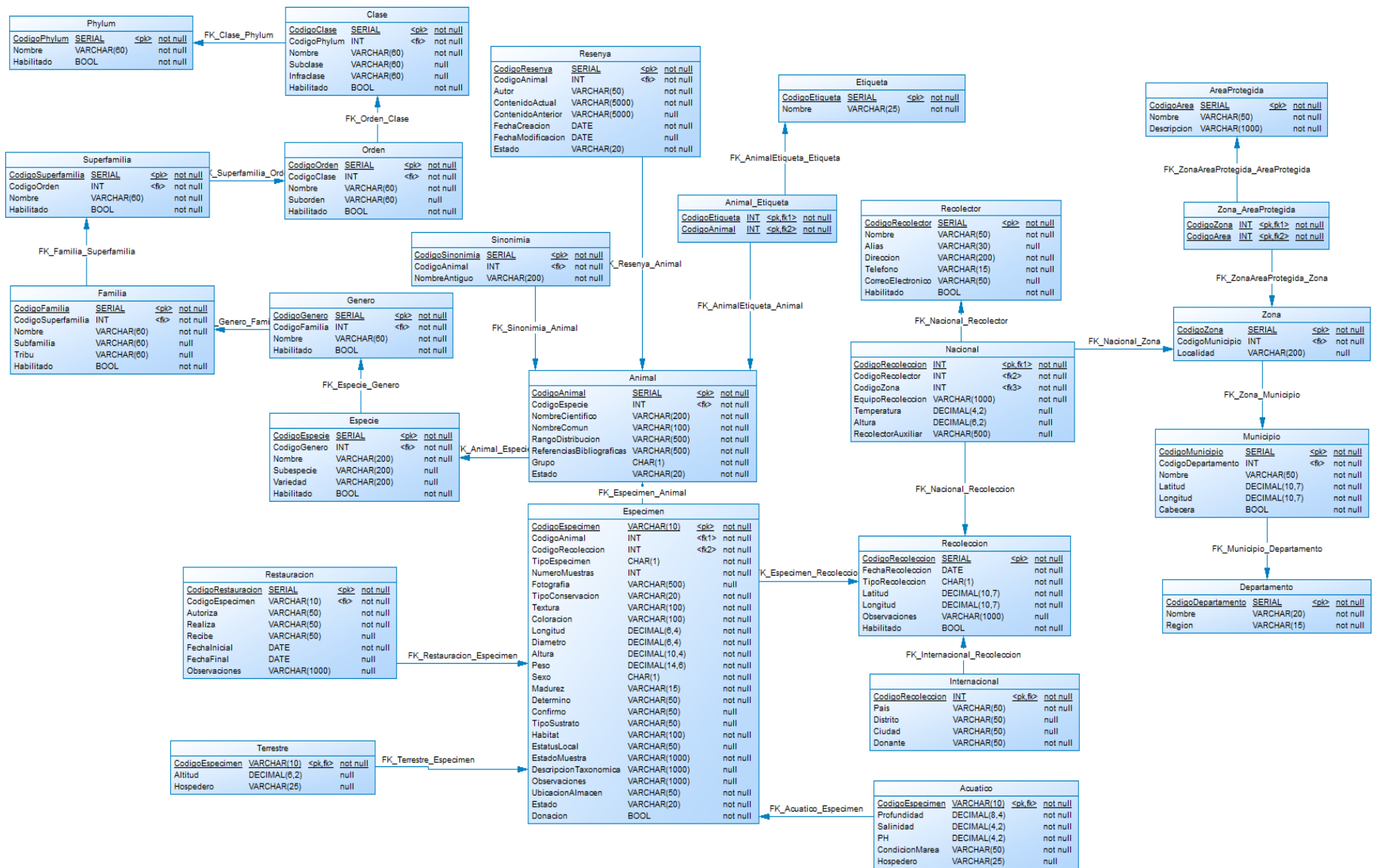


Figura 3.50. Modelo conceptual del esquema de zoología

3.7.3. MODELO LÓGICO

El objetivo del diseño lógico es convertir los esquemas conceptuales locales en un esquema lógico global que se ajuste al modelo de la base de datos sobre el que se vaya a implementar el sistema. En las figuras 3.51 y 3.52 se muestran los modelos lógicos del sistema SIZOO.

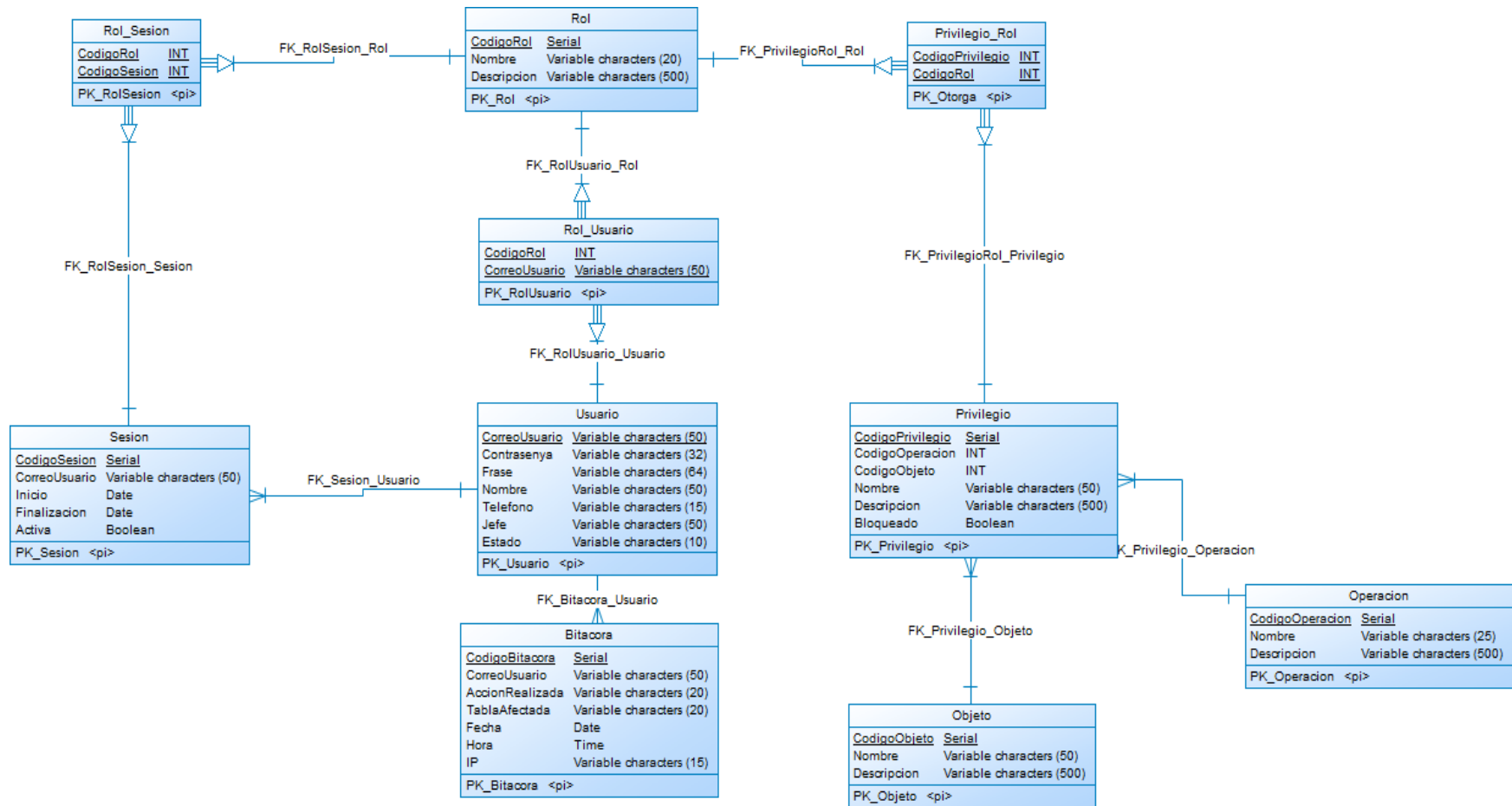


Figura 3.51. Modelo lógico del esquema de seguridad.

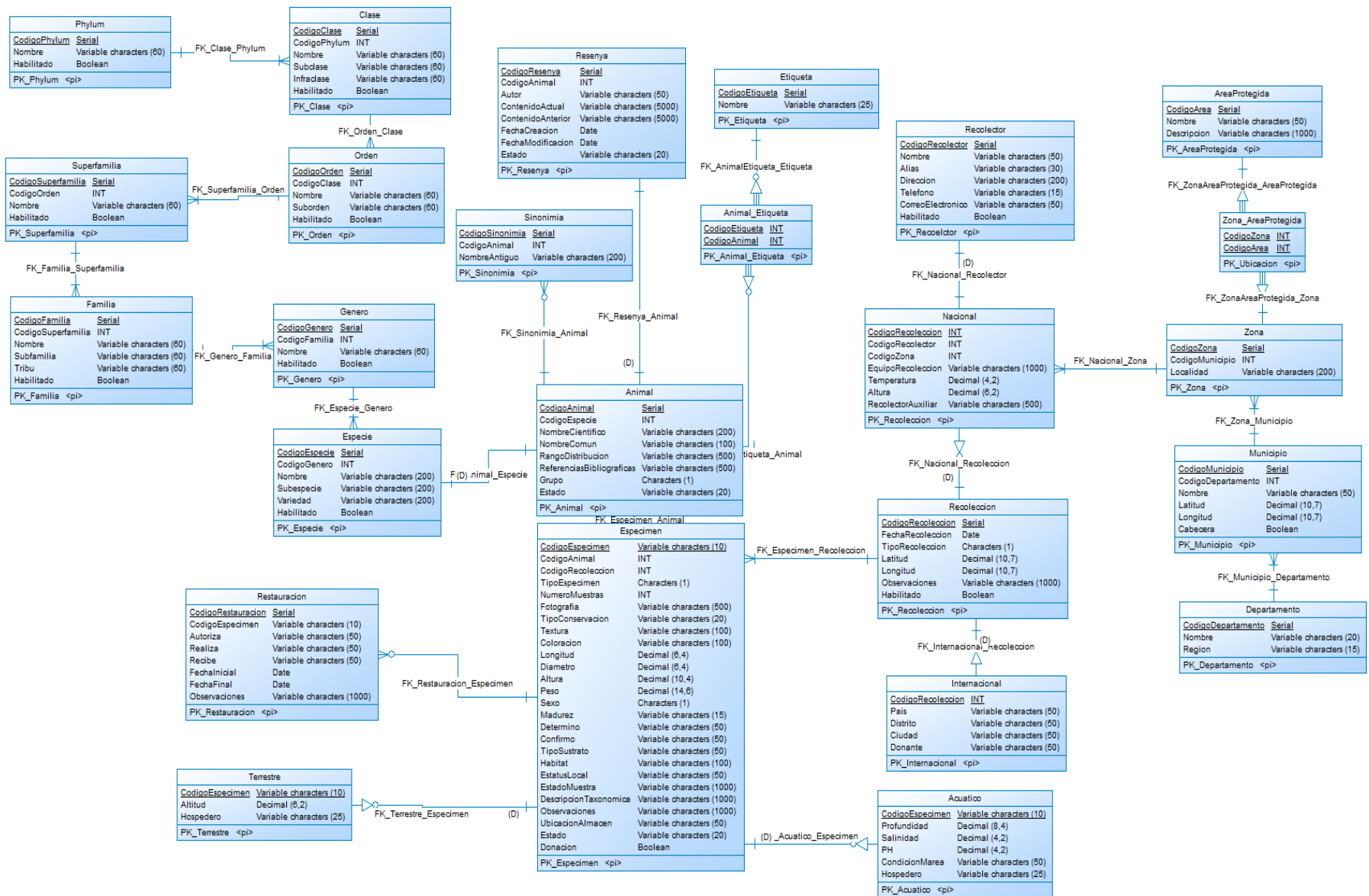


Figura 3.52. Modelo lógico del esquema de zoología.

3.7.4. MODELO FÍSICO

El modelo físico es una descripción de la implementación de la base de datos en memoria secundaria: las estructuras de almacenamiento y los métodos utilizados para tener un acceso eficiente a los datos. En las figuras 3.53 y 3.54 se muestra el modelo físico de la base de datos del sistema SIZOO.

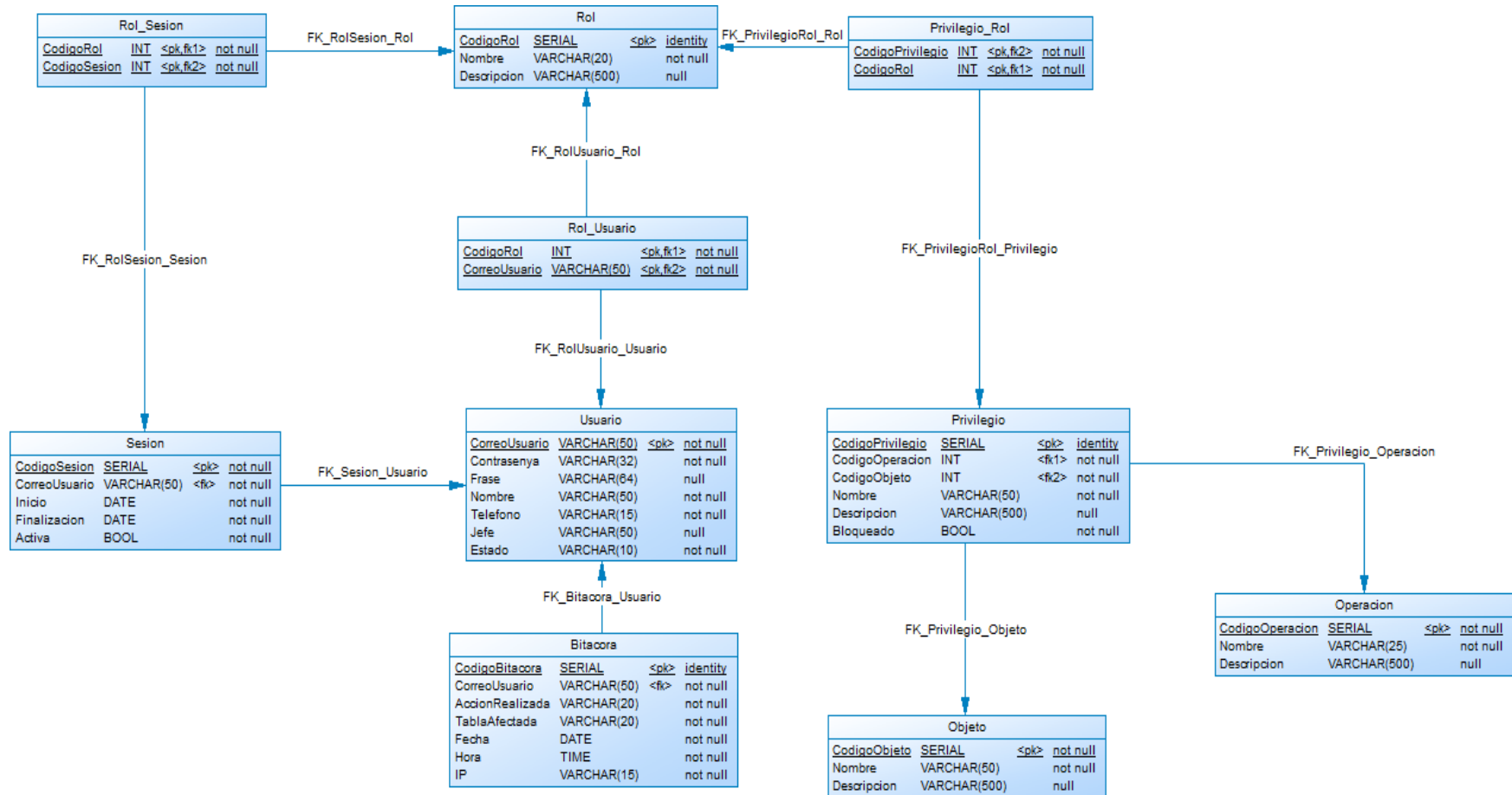


Figura 3.53. Modelo físico del esquema de seguridad.

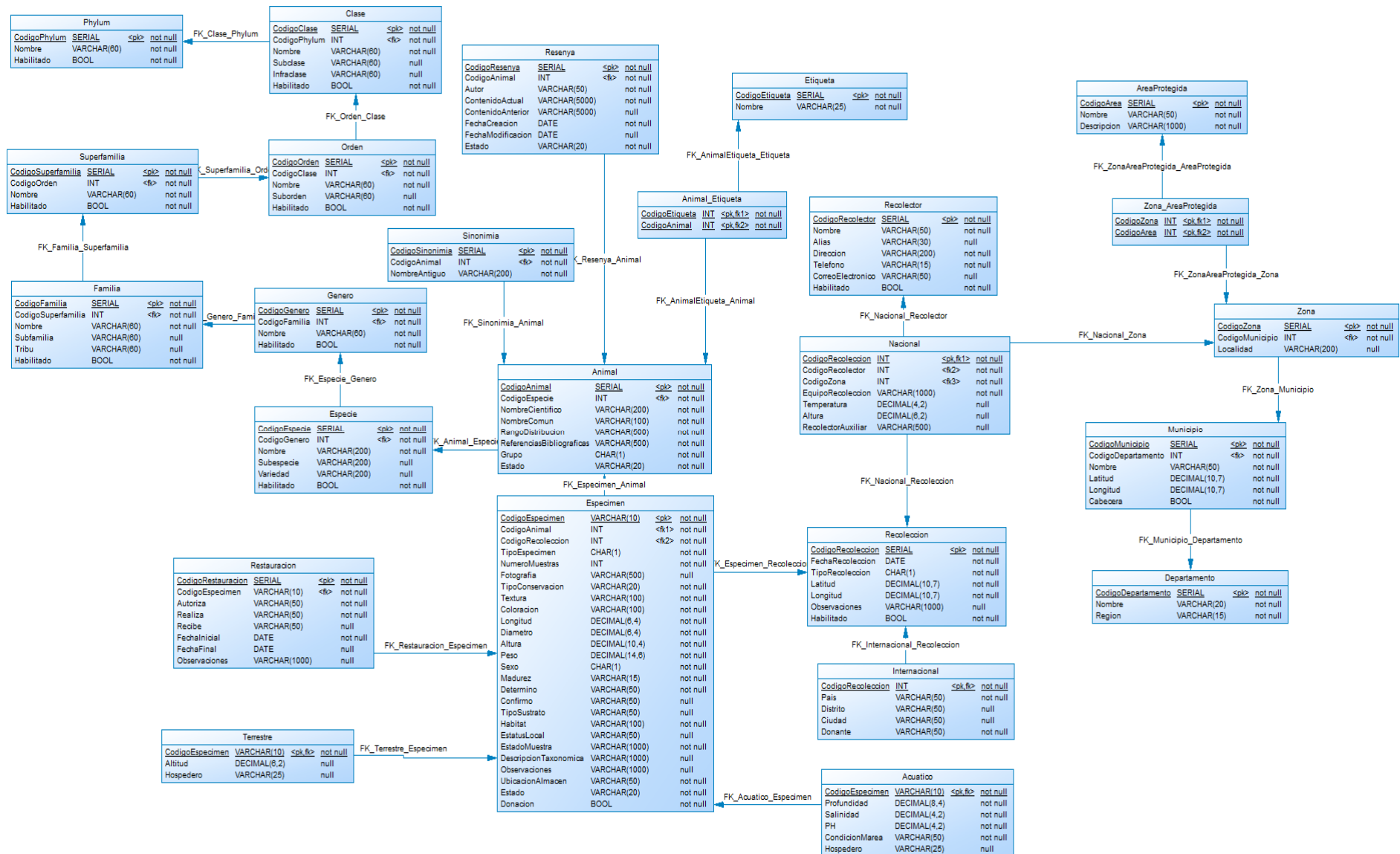


Figura 3.54. Modelo físico del esquema de zoología.

CAPÍTULO 4:

CONSTRUCCIÓN

Un estándar de diseño son convenciones de diseño (generalmente obligatorias) ajustadas para predeterminar de manera consistente, características de diseño resolutivo en apoyo a los objetivos de la organización y optimizados para entornos específicos de una organización. Es a través del uso de estándares interno de diseño que las organizaciones pueden entregar soluciones ajustadas a sus entornos de manera consistente, recursos, objetivos y prioridades.

4.1. ESTÁNDARES DE BASE DE DATOS

El objetivo de los estándares de base de datos es asegurar el entendimiento del diseño de está en su totalidad, incluso para personas que no estén involucradas en el proyecto. Facilitando el trabajo para los desarrolladores.

Los siguientes estándares están basados en el documento “Convenciones de nombres para el nombrado de la base de datos¹”.

4.1.1. NOMBRE DE LA BASE DE DATOS

- El nombre de la base de datos reflejará de manera precisa y exacta el contenido de la base y su función.
- Se escribirá en minúscula los nombres de las tablas y los campos. Por ejemplo Código Área se escribirá *codigoarea*. Las palabras propias del lenguaje SQL se escribirán en minúscula también, por ejemplo para crear una tabla en la base de datos se escribirá *create table nombretabla*.
- El nombre del archivo script será igual al nombre de la base de datos desde donde se genera el archivo.

4.1.2. SOBRE LO CONCERNIENTE A TODOS LOS OBJETOS

Para la nomenclatura de todos los objetos deben tomarse en cuenta las siguientes consideraciones:

- Se limitará el nombre a 30 caracteres (entre menos mejor).
- Se usará únicamente letras (no se admiten números).
- Se usará el menor número posible de guiones bajos (_).
- Se usarán letras como el primer carácter del nombre (no comenzar con guiones bajos).
- Se evitara abreviaciones (pueden llevar a una mala interpretación de los nombres).
- Se evitará usar acrónimos (ya que estos pueden tener más de un significado).
- Se buscara hacer el nombre legible (debe poderse pronunciar fácilmente en voz alta).
- Se evitará usar espacios en blanco en los nombres aunque el sistema lo permita.

¹ <http://ensenada.gob.mx/Micrositios/Informativa/wp-content/uploads/2011/08/DBNC.pdf>

4.1.3. TABLAS Y VISTAS

- Los nombres de las tablas serán con una palabra en singular.
- Únicamente se utilizarán caracteres alfabéticos, salvo que por la naturaleza del nombre se necesiten dígitos numéricos. No se hará uso de caracteres de puntuación o símbolos.
- Las relaciones entre tablas se implementarán mediante restricciones o constraints (claves foráneas) con integridad referencial.
- Las letras acentuadas se reemplazarán con las equivalentes no acentuadas, y en lugar de la letra ñ se utilizará ny. Por ejemplo Año Fundación se escribirá *anyofundacion*.
- La integridad referencial deberá actualizarse en cascada en todos los casos, y restringir el borrado salvo para las entidades débiles.
- Los nombres de las vistas seguirán las mismas convenciones que los nombres de las tablas teniendo como prefijo *vw* siguiéndole un guión bajo (*_*). Por ejemplo *vw_vistaventa*.
- Las tablas de históricos deben tener el prefijo o sufijo *hist*. Por ejemplo Histórico Ventas será *histventas*.

4.1.4. CAMPOS

Las siguientes recomendaciones aplican a todos los campos:

- Cada nombre de campo debe ser único dentro de su tabla correspondiente.
- Los campos clave se ubicarán al inicio de la definición de la tabla (deben ser los primeros).
- Las claves primarias utilizan el texto *pk* como sufijo o prefijo, seguido de un guión bajo, según se considere conveniente. Por ejemplo *pk_codigocliente*.
- Las claves foráneas utilizan el texto *fk* como sufijo o prefijo, seguido de un guión bajo, según se considere conveniente. Además, estas claves deben nombrarse mencionando su tabla de origen y tabla de destino. Por ejemplo *fk_origendestino*.
- Cada nombre de campo debe derivarse del identificador utilizado durante el proceso de análisis de negocio. Si el campo no fue identificado durante el proceso de análisis del sistema, se le debe asignar un nombre abreviado que lo identifique fácilmente.

- No se utilizarán palabras reservadas como nombres de campos. Por ejemplo `public`, `class`, `int`, etc.
- Las claves compuestas se utilizaran en casos específicos, por ejemplo, tablas de relación o entidades débiles.

4.1.5. ÍNDICES

- Los índices se nombrarán considerando la tabla a la que están relacionados y el propósito del índice. Los índices utilizan el sufijo o prefijo `ix`, seguido de un guion bajo, según se considere conveniente. Por ejemplo `ix_localidadmostrar`.

4.1.6. TABLAS DE REFERENCIA

- Los nombres de las tablas que son usadas para guardar referencias (*Muchos a Muchos*) deben de definirse en lo posible usando el nombre de ambas tablas.

4.1.7. PROCEDIMIENTOS ALMACENADOS

- **Nombre de los procedimientos almacenados:** por lo general estos procedimientos pueden realizar una o más acciones con los objetos de la base de datos, entre las acciones se encuentran las operaciones básicas de inserción, borrado, pero además pueden invocar eventos u otros procedimientos almacenados. Por esta razón se recomienda que los procedimientos almacenados sean nombrados de acuerdo con la actividad o proceso que realizan.
- **Prefijos:** los procedimientos almacenados deben comenzar con el prefijo `proc_` y de la misma manera que en el nombrado de las tablas, pueden ser utilizados otros los prefijos para agrupar procedimientos almacenados. Por ejemplo `proc_insertar`.

4.1.8. DISPARADORES

- **Prefijos:** el nombrado de los disparadores de eventos o triggers deben de seguir las mismas recomendaciones generales usadas para nombrar las tablas. Se hará uso del prefijo *trg_* con el propósito de hacer saber que el objeto en la base de datos es un disparador.
- **Nomenclatura:** se incluirá tanto el nombre de la tabla a la que el disparador está vinculado, así como la acción que ocasiona que dicho disparador sea ejecutado. Un ejemplo de disparador es *trg_validareliminacionusuario*

4.1.9. DOCUMENTACIÓN DE LA BASE DE DATOS

- **Libretos:** Todas las bases de datos deben contar con un script (libreto) de creación en lenguaje *SQL* (o aquel que corresponda a la base de datos) que incluya la definición precisa de todos los objetos involucrados así como los valores predeterminados de los registros y los catálogos.
- **Encabezado del libreto:** El libreto deberá incluir una sección inicial que indique la información necesaria para la identificación rápida del sistema al que pertenece, sus dependencias funcionales, requerimientos e información de autoría y de contacto. Dicho encabezado será construido de la siguiente manera:

```
/*=====
[  SCRIPT DE CONFIGURACION DE LA BASE DE DATOS  ]
|=====
| NOMBRE DEL SISTEMA:
| MODULO:
| VERSION:
| DEPENDENCIAS DE MODULO:
| REQUERIMIENTOS:
| AUTOR:
| E-MAIL:
| COPYRIGHT:
|=====*/
```

- NOMBRE DEL SISTEMA: Tal como lo dice será el nombre clave del sistema a desarrollar.
- MODULO: Si aplica, se debe incorporar el nombre del módulo del sistema a que corresponda el libreto.
- VERSION: Versión del libreto.
- DEPENDENCIAS: Libreto hacia otros módulos o sistemas.
- REQUERIMIENTOS: Si aplica, deben especificarse requerimientos para el uso del libreto, como por ejemplo el tipo y la versión de la base de datos.
- AUTOR: Debe especificarse el o los autores del libreto.
- E-MAIL: Debe especificarse el correo electrónico de soporte para el sistema a que corresponde el libreto.
- COPYRIGHT: Debe especificarse la nota de derecho de autor como sigue:

UES 2012, all rights reserved.

- **Encabezados de las tablas:** todas las tablas deben incluir en su libreto un encabezado con la información necesaria para la rápida identificación de su función y relevancia del contenido en el sistema al que pertenecen. El encabezado se compondrá de la siguiente información:

```

/*=====
| TABLA :
| DESCRIPCION :
| OBSERVACIONES :
=====*/

```

- TABLA: Nombre de la tabla.
- DESCRIPCION: Descripción del uso o la información contenida en la tabla.
- OBSERVACIONES: Comentarios y observaciones sobre la estructura y/o contenido de las tablas.

- **Encabezados de modificaciones a las tablas:** La alteración en la estructura de las tablas es bastante común, ya sea adicionar o eliminar campos, cambiar tipos de datos, agregar llaves o índices, etc. El encabezado mostrará la siguiente información

```

/*=====
| TIPO DE MODIFICACION EN LAS TABLAS RELACIONADAS
=====*/

```

- TIPO DE MODIFICACION EN LAS TABLAS: Se tiene que especificar si la modificación consiste en LLAVES PRIMARIAS, LLAVES FORANEAS o INDICES, mencionando las tablas afectadas.

- **Encabezados de procedimientos y disparadores:** Los procedimientos almacenados y disparadores tendrán en su encabezado la siguiente información:

```

/*=====
| NOMBRE :
| DESCRIPCION :
| TABLAS RELACIONADAS :
| OBSERVACIONES :
===== */

```

- NOMBRE: Nombre del disparador o procedimiento.
- DESCRIPCION: Descripción de lo que realizara el disparador o procedimiento.
- TABLAS RELACIONADAS: Especificación de las tablas que tienen relación con el disparador.
- OBSERVACIONES: Comentarios y observaciones.

- **Comentarios adicionales:** Los comentarios adicionales son necesarios cuando cierta sentencia puede no dejar muy claro cuál es su función, por ejemplo una llave foránea. Estos comentarios tienen que ser colocados una línea arriba de la sentencia a la cual se hace referencia. El formato a seguir es:

```

/*          COMENTARIO          */

```

4.2. ESTÁNDARES DE PROGRAMACIÓN

Un estándar de programación es una forma de normalizar la programación de forma tal que al trabajar en un proyecto, cualquier persona involucrada en el mismo tenga acceso y comprenda el código. Esto permite:

- Definir la escritura y organización del código fuente de un programa.
- Facilita a un programador la modificación del código fuente aunque no haya participado en la codificación inicial.
- Definir la forma en que deben ser declaradas las variables, las clases, los comentarios, etc.
- Especificar qué datos deben incluirse acerca del programador y de los cambios realizados al código fuente, etc.

El estándar de programación a usar en el desarrollo del proyecto es la convención de código de Java.¹

4.2.1. ORGANIZACIÓN DE ARCHIVOS

Las clases en Java se agrupan en paquetes. Estos paquetes se deben organizar de manera jerárquica, de forma que todo código desarrollado para el MUNHES tendrá que estar incluido dentro del paquete munhes.sizoo.

Un archivo consta de secciones que deben estar separadas por líneas en blanco y comentarios opcionales que identifiquen cada sección.

Deben evitarse los archivos de gran tamaño que contengan más de 1,000 líneas. En ocasiones, este tamaño excesivo provoca que la clase no encapsule un comportamiento claramente definido, albergando una gran cantidad de métodos que realizan tareas funcional o conceptualmente heterogéneas.

Cada archivo fuente Java (.java) debe contener una única clase o interfaz pública. El nombre del archivo tiene que coincidir con el nombre de la clase. Cuando existan varias clases privadas asociadas funcionalmente a una clase pública, podrán colocarse en el mismo archivo fuente que la clase pública. La clase pública debe estar situada en primer lugar dentro del archivo fuente.

En todo archivo fuente Java distinguimos las siguientes secciones:

- Comentarios de inicio.
- Sentencia de paquete.
- Sentencias de importación.
- Declaraciones de clases e interfaces.

¹ JavaCode Conventions; <http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

- **Comentarios de inicio:** todo archivo fuente debe comenzar con un comentario que incluya el nombre de la clase, descripción de la clase (2 líneas máximo), información sobre la versión del código, la fecha y el autor.

Dentro de estos comentarios iniciales podrían incluirse adicionalmente comentarios sobre los cambios efectuados sobre dicho archivo (mejora, incidencia, error, etc.). Estos comentarios son opcionales si los archivos están bajo un sistema de control de versiones bien documentado, en caso contrario se recomienda su uso. Estos comentarios constituyen el historial de cambios del archivo. Este historial es único para cada archivo y permitirá conocer rápidamente el estado y la evolución que ha tenido el archivo desde su origen.

A continuación se muestra un comentario de inicio para la clase Usuario.java:

```
/*
 * @(#)Usuario.java
 *
 * Esta clase es una entidad de modelo.
 *
 * @version 1.5
 * @date 10/10/10
 * @author Juan Perez
 *
 */
```

- **Sentencias de paquete:** la primera línea no comentada de un archivo fuente debe ser la sentencia de paquete, que indica el paquete al que pertenece(n) la(s) clase(s) incluida(s) en el archivo fuente. Por ejemplo:

```
package com.dominio.modelo;
```

- **Sentencias de importación:** tras la declaración del paquete se incluirán las sentencias de importación de los paquetes necesarios. Esta importación de paquetes obligatorios seguirá el siguiente orden:
 - Paquetes del JDK de java.
 - Paquetes de utilidades no pertenecientes al JDK de Java, de *frameworks* de desarrollo o de proyectos *opensource* tales como *apache*, *hibernate*, *springframework*, etc.
 - Paquetes de la aplicación.

Se recomienda minimizar en la medida de lo posible el uso de importaciones del tipo "*package.**", pues dificultan la comprensión de las dependencias existentes entre las clases utilizadas por la aplicación. En caso contrario, se recomienda utilizar comentarios de línea tras la importación. Un ejemplo se muestra a continuación:


```

import java.io.*; // BufferedReader, FileInputStream, File
import java.util.ArrayList;
import org.apache.log4j.Logger;
import org.apache.lucene.analysis.Analyzer;
import com.compania.depuracion;

```

- **Declaraciones de clases e interfaces:** la tabla 4.1 describe los elementos que componen la declaración de una clase o interfaz, así como el orden en el que deben estar situados.

Elementos de declaración de una clase / interfaz	Descripción
Comentario de documentación de la clase/interfaz	Permite describir la clase/interfaz desarrollada. Necesario para generar la documentación de la api mediante javadoc.
Sentencia class / interface	N/A
Comentario de implementación de la clase/interfaz, si es necesario	Este comentario incluye cualquier información que no pueda incluirse en el comentario de documentación de la clase/interfaz.
Variables de clase (estáticas)	En primer lugar las variables de clase públicas (public), después las protegidas (protected), posteriormente las de nivel de paquete (sin modificador), y por último las privadas (private).
Variables de instancia	Primero las públicas (public), después las protegidas (protected), luego las de nivel de paquete (sin modificador), y finalmente las privadas (private).
Constructores	N/A
Métodos	Deben agruparse por funcionalidad en lugar de agruparse por ámbito o accesibilidad. Por ejemplo, un método privado puede estar situado entre dos métodos públicos. El objetivo es desarrollar código fácil de leer y comprender.

Tabla 4.1. Declaraciones de Clases e interfaces.

4.2.2. SANGRÍA

Como norma general se establecen 4 caracteres como unidad de sangría. Los entornos de desarrollo integrado (IDE) más populares, tales como Eclipse o NetBeans, incluyen facilidades para formatear código Java.

- **Longitud de línea:** la longitud de línea no debe superar los 80 caracteres por motivos de visualización e impresión.
- **División de líneas:** cuando una expresión ocupe más de una línea, esta se podrá romper o dividir en función de los siguientes criterios:
 - Tras una coma.
 - Antes de un operador.
 - Se recomienda las rupturas de nivel superior a las de nivel inferior.
 - Alinear la nueva línea con el inicio de la expresión al mismo nivel que la línea anterior.
 - Si las reglas anteriores generan código poco comprensible, entonces estableceremos tabulaciones de 8 espacios.

Por ejemplo:

```
unMetodo(expresionLarga1, expresionLarga 2, expresionLarga 3,
          expresionLarga 4, expresionLarga 5);

if ((condicion1 && condicion2)
    || (condicion3 && condicion4)
    || !(condicion5 && condicion6)) {
    unMetodo();
}
```

4.2.3. COMENTARIOS

Distinguiamos dos tipos de comentarios: los comentarios de implementación y los de documentación.

- **Comentarios de implementación:** estos comentarios se utilizan para describir el código (el cómo), y en ellos se incluye información relacionada con la implementación, tales como descripción de la función de variables locales, fases lógicas de ejecución de un método, captura de excepciones, etc.

Distinguiamos tres tipos de comentarios de implementación:

- Comentarios de bloque: permiten la descripción de archivos, clases, bloques, estructuras de datos y algoritmos. Por ejemplo:

```
/*
 * Esto es un comentario
 * de bloque
 */
```

- **Comentarios de línea:** son comentarios cortos localizados en una sola línea y tabulados al mismo nivel que el código que describen. Si ocupa más de una línea se utilizará un comentario de bloque. Deben estar precedidos por una línea en blanco. Por ejemplo:

```
/* Esto es un comentario de línea */
// Esto es otro comentario de línea
```

- **Comentario a final de línea:** comentario situado al final de una sentencia de código y en la misma línea. Por ejemplo:

```
int contador = 4 + 10; // Inicialización del contador
contador++; /* Incrementamos el contador */
```

4.2.4. DECLARACIONES

- **Una declaración por línea:** se recomienda el uso de una declaración por línea, promoviendo así el uso de comentarios. Ejemplo:

```
int idUnidad; // Identificador de la unidad organizativa
String[] funciones; // Funciones de la unidad
```

- **Inicialización:** toda variable local tendrá que ser inicializada en el momento de su declaración, salvo que su valor inicial dependa de algún valor que tenga que ser calculado previamente. Por ejemplo:

```
int idUnidad = 1;
String[] funciones = { "Administración", "Intervención", "Gestión" };
```

- **Localización:** las declaraciones deben situarse al principio de cada bloque principal en el que se utilicen, y nunca en el momento de su uso. Por ejemplo:

```
public void unMetodo() {
    int contador = 0; // inicio del método
    ...
}
```

La única excepción a esta regla son los índices de los bucles *for* (hacer desde-hasta), ya que, en Java, pueden incluirse dentro de la propia sentencia *for*.

```
for (int i=0; contador<10; i++) {  
    ...  
}
```

Se debe evitar el uso de declaraciones que oculten a otras declaraciones de ámbito superior.

```
int contador = 0; // Inicio del método  
public void unMetodo() {  
    if (condicion) {  
        int contador = 2; // ¡¡ EVITAR !!  
        ...  
    }  
    ...  
}
```

- **Declaración de clases / interfaces:** durante el desarrollo de clases / interfaces se deben seguir las siguientes reglas de formateo:
 - No incluir ningún espacio entre el nombre del método y el paréntesis inicial del listado de parámetros.
 - El carácter inicio de bloque ({) debe aparecer al final de la línea que contiene la sentencia de declaración.
 - El carácter fin de bloque (}) se sitúa en una nueva línea tabulada al mismo nivel que su correspondiente sentencia de inicio de bloque, excepto cuando la sentencia sea nula, en tal caso se situará detrás de ({).
 - Los métodos se separarán entre sí mediante una línea en blanco.

Por ejemplo:

```
public classe ClaseEjemplo extends Object {  
  
    int variable1;  
    int variable2;  
  
    public ClaseEjemplo() {  
        variable1 = 0;  
        variable2 = 1;  
    }  
    ...  
}
```

4.2.5. SENTENCIAS

Cada línea debe contener como máximo una sentencia. Ejemplo:

```
int contador++;  
int variable--;
```

Las sentencias pertenecientes a un bloque de código estarán tabuladas un nivel más a la derecha con respecto a la sentencia que las contiene.

El carácter inicio de bloque ({) debe situarse al final de la línea que inicia el bloque. El carácter final de bloque (}) debe situarse en una nueva línea tras la última línea del bloque y alineada con respecto al primer carácter de dicho bloque.

Todas la sentencias de un bloque deben encerrarse entre llaves ({ ... }), aunque el bloque conste de una única sentencia. Esta práctica permite añadir código sin cometer errores accidentalmente al olvidar añadir las llaves. Ejemplo:

```
if (condicion) {  
    variable++;  
}
```

La sentencia try/catch siempre debe tener el formato siguiente:

```
try {  
    sentencias;  
} catch (ClaseException e) {  
    sentencias;  
}
```

En el bloque catch siempre se imprimirá una traza de error indicando el tipo de excepción generada y posteriormente se elevará dicha excepción al código invocante, salvo que la lógica de ejecución de la aplicación no lo requiera.

Siempre se utilizará el bloque finally para liberar recursos y para imprimir trazas de monitorización de fin de ejecución, como se muestra en el siguiente ejemplo:

```
try {  
    sentencias;  
} catch (ClaseException e) {  
    sentencias;  
} finally {  
    sentencias;  
}
```

4.2.6. ESPACIOS EN BLANCO

Las líneas y espacios en blanco mejoran la legibilidad del código permitiendo identificar las secciones de código relacionadas lógicamente. Se utilizarán espacios en blanco en los siguientes casos:

- Entre una palabra clave y un paréntesis. Esto permite que se distingan las llamadas a métodos de las palabras clave. Por ejemplo:

```
while (true) {  
    ...  
}
```

- Tras cada coma en un listado de argumentos. Por ejemplo:

```
objeto.unMetodo(a, b, c);
```

- Para separar un operador binario de sus operandos, excepto en el caso del operador punto (.). Nunca se utilizarán espacios entre los operadores unarios (por ejemplo, ++ o --) y sus operandos. Por ejemplo:

```
a += b + c;  
a = (a + b) / (c + d);  
contador++;
```

Para separar las expresiones incluidas en la sentencia for. Por ejemplo:

```
for (expresion1; expresion2; expresion3)
```

Al realizar el moldeo o "casting" de clases. Ejemplo:

```
Unidad unidad = (Unidad) objeto;
```

4.2.7. NOMENCLATURA DE IDENTIFICADORES

Las convenciones de nombres de identificadores permiten que los programas sean más fáciles de leer y por tanto más comprensibles. También proporcionan información sobre la función que desempeña el identificador dentro del código, es decir, si es una constante, una variable, una clase o un paquete, entre otros.

- **Paquetes:** se escribirán siempre en letras minúsculas para evitar que entren en conflicto con los nombres de clases o interfaces. El resto de componentes del paquete se nombrarán de acuerdo a las normas internas de organización: proyecto, sección, organismo, área, categorización, etc.

Generalmente se suele utilizar el nombre de dominio de Internet en orden inverso. Cuando dicho nombre contenga un carácter (-), este se sustituirá por el carácter (_). Ejemplos:

```
com.dominio.reporteria
com.dominio.atributos
com.dominio.controlador
```

- **Clases e interfaces:** los nombres de clases deben ser sustantivos y deben tener la primera letra en mayúsculas. Si el nombre es compuesto, cada palabra componente deberá comenzar con mayúsculas.

Los nombres serán simples y descriptivos. Debe evitarse el uso de acrónimos o abreviaturas, salvo en aquellos casos en los que dicha abreviatura sea más utilizada que la palabra que representa (URL, HTTP, etc.).

Las interfaces se nombrarán siguiendo los mismos criterios que los indicados para las clases. Como norma general toda interfaz se nombrará con el prefijo I para diferenciarla de la clase que la implementa (que tendrá el mismo nombre sin el prefijo I).

```
class Especimen
class AnimalDAO
class ReporteService
class ITaxon
```

- **Métodos:** los métodos deben ser verbos escritos en minúsculas. Cuando el método esté compuesto por varias palabras cada una de ellas tendrá la primera letra en mayúsculas. Por ejemplo:

```
public void insertarUnidad(Unidad unidad);
public void eliminarAgenda(Agenda agenda);
public void actualizarTramite(Tramite tramite)
```

- **Variables:** las variables se escribirán siempre en minúsculas. Las variables compuestas tendrán la primera letra de cada palabra componente en mayúsculas.

Las variables nunca podrán comenzar con el carácter (_) o (\$). Los nombres de variables deben ser cortos y sus significados tienen que expresar con suficiente claridad la función que desempeñan en el código. Debe evitarse el uso de nombres de variables con un sólo carácter, excepto para variables temporales. Ejemplos:

```
Unidad unidad;  
Agenda agenda;  
Tramite tramite;
```

- **Constantes:** todos los nombres de constantes tendrán que escribirse en mayúsculas. Cuando los nombres de constantes sean compuestos las palabras se separarán entre sí mediante el carácter de subrayado (`_`). Ejemplo:

```
int LONGITUD_MAXIMA;  
int LONGITUD_MINIMA;
```

4.2.8. PRÁCTICAS DE PROGRAMACIÓN

- **Visibilidad de atributos de instancia y de clase:** los atributos de instancia y de clase serán siempre privados, excepto cuando tengan que ser visibles en subclases herederas, en tales casos serán declarados como protegidos.

El acceso a los atributos de una clase se realizará por medio de los métodos `get` y `set` correspondientes, incluso cuando el acceso a dichos atributos se realice en los métodos miembros de la clase. Por ejemplo:

```
public class Unidad {  
  
    private int id;  
    private String nombre;  
    ...  
  
    public void actualizaUnidad(Unidad unidad) {  
        this.setId(unidad.getId());  
        this.setNombre(unidad.getNombre());  
    }  
  
    ...  
}
```

- **Referencias a miembros de una clase:** evitar el uso de objetos para acceder a los miembros de una clase (atributos y métodos estáticos). Utilizaremos en su lugar el nombre de la clase. Por ejemplo:

```
metodoUtilidad(); // Acceso desde la propia clase estática  
ClaseUtilidad.metodoUtilidad(); // Acceso común desde cualquier  
clase
```


- **Constantes:** los valores constantes (literales) nunca aparecerán directamente en el código. Para designar dichos valores se utilizarán constantes escritas en mayúsculas y se declararán, según su ámbito de uso, o bien en una Clase de constantes creada para tal efecto, o bien en la clase donde sean utilizadas.

A continuación se muestran ejemplos del uso incorrecto y correcto respectivamente:

```
// Uso incorrecto
codigoErrorUsuarioNoEncontrado = 1;
...
switch (error) {
    case codigoErrorUsuarioNoEncontrado:
        ...
}

// Uso correcto
public final int CODIGOERROR_USUARIONOENCONTRADO = 1;
...
switch (error) {
    case CODIGOERROR_USUARIONOENCONTRADO:
        ...
}
```

- **Asignación sobre variables:** de deben evitar las asignaciones de un mismo valor sobre múltiples variables en una misma sentencia, ya que dichas sentencias suelen ser difíciles de leer. Por ejemplo:

```
int a = b = c = 2; // ¡¡ EVITAR !!
```

No utilizar el operador de asignación en aquellos lugares donde sea susceptible de confusión con el operador de igualdad. Por ejemplo:

```
// INCORRECTO
if ((c = d++) == 0) { }

// CORRECTO
c = d++;
if (c == 0) { }
```

No utilizar asignaciones embebidas o anidadas. Ejemplo:

```
c = (c = 3) + 4 + d; // ¡¡ EVITAR !!
```

Debería escribirse:

```
c = 3;
c = c + 4 + d;
```

- **Paréntesis:** Es una buena práctica el uso de paréntesis en expresiones que incluyan distintos tipos de operadores para evitar problemas de precedencia de operadores. Aunque la precedencia de operadores nos pueda parecer clara, debemos asumir que otros programadores no tengan un conocimiento exhaustivo sobre las reglas de precedencia. Por ejemplo:

```
if (w == x && y == z)          // INCORRECTO
if ((w == x) && (y == z)) // CORRECTO
```

- **Valores de retorno:** Los valores de retorno tendrán que ser simples y comprensibles, de acuerdo al propósito y comportamiento del objeto en el que se utilicen. Por ejemplo:

```
// INCORRECTO
public boolean esProgramador(Empleado emp) {

    if (emp.getRol().equals(ROL_PROGRAMADOR)) {
        return true;
    } else {
        return false;
    }
}

// CORRECTO
public boolean esProgramador(Empleado emp) {

    boolean esUnProgramador = false;

    if (emp.getRol().equals(ROL_PROGRAMADOR)) {
        esUnProgramador = true;
    }

    return esUnProgramador;
}
```

- **Expresiones en el operador condicional ternario:** toda expresión compuesta, por uno o más operadores binarios, situada en la parte condicional del operador ternario deberá ir entre paréntesis. Ejemplo:

```
(x >= y) ? x : y;
```

- **Comentarios especiales (TODO, FIXME, XXX):** Utilizaremos XXX para comentar aquella porción de código que, aunque no tenga mal funcionamiento, requiera modificaciones. Usaremos FIXME para señalar un bloque de código erróneo que no funciona. Emplearemos TODO para comentar posibles mejoras de código, como puedan ser las debidas a optimizaciones, actualizaciones o refactorizaciones.

4.2.9. DOCUMENTACIÓN JAVADOC

Se aconseja, como buena práctica de programación, incluir en la entrega de la aplicación la documentación de los archivos fuente de todas las clases. Dicha documentación será generada por la herramienta javadoc.

La herramienta javadoc construirá la documentación a partir de los comentarios (incluidos en las clases) encerrados entre los caracteres (`/**`) y (`*/`). Distinguimos tres tipos de comentarios javadoc, en función del elemento al que preceden: de clase, de variable y de método.

Dentro de los comentarios javadoc podremos incluir código html y etiquetas especiales de documentación. Estas etiquetas de documentación comienzan con el símbolo arroba (`@`), se sitúan al inicio de línea del comentario y nos permiten incluir información específica de nuestra aplicación de una forma estándar.

Como norma general utilizaremos las siguientes etiquetas:

- **@author Nombre:** añade información sobre el autor o autores del código.
- **@version InformacionVersion:** permite incluir información sobre la versión y fecha del código.
- **@param NombreParametro Descripción:** inserta el parámetro especificado y su descripción en la sección "Parameters:" de la documentación del método en el que se incluya. Estas etiquetas deben aparecer en el mismo orden en el que aparezcan los parámetros especificados del método. Esta etiqueta no puede utilizarse en comentarios de clase, interfaz o campo. Las descripciones deben ser breves.
- **@return Descripción:** inserta la descripción indicada en la sección "Returns:" de la documentación del método. Este tag debe aparecer en los comentarios de documentación de todos los métodos, salvo en los constructores y en aquellos que no devuelvan ningún valor (void).
- **@throws NombreClase Descripción:** añade el bloque de comentario "Throws:" incluyendo el nombre y la descripción de la excepción especificada. Todo comentario de documentación de un método debe contener una etiqueta "@throws" por cada una de las excepciones que pueda elevar. La descripción de la excepción puede ser tan corta o larga como sea necesario y debe explicar el motivo o motivos que la originan.
- **@see Referencia:** permite incluir en la documentación la sección de comentario "See also:", conteniendo la referencia indicada. Puede aparecer en cualquier tipo de comentario "javadoc". Nos permite hacer referencias a la documentación de otras clases o métodos.
- **@deprecated Explicación:** esta etiqueta indica que la clase, interfaz, método o campo está obsoleto y que no debe utilizarse, y que dicho elemento posiblemente desaparecerá en futuras versiones. "javadoc" añade el comentario "Deprecated" en la documentación e incluye el texto explicativo indicado tras la etiqueta. Dicho texto debería incluir una sugerencia o referencia sobre la clase o método sustituto del elemento "deprecado".
- **@since Version:** se utiliza para especificar cuándo se ha añadido a la API la clase, interfaz, método o campo. Debería incluirse el número de versión u otro tipo de información.

El siguiente ejemplo muestra los tres tipos de comentarios *javadoc*:

```
/**
 * UnidadOrganizativa.java:
 *
 * Clase que muestra ejemplos de comentarios de documentación de código.
 *
 * @author jlflorido
 * @version 1.0, 05/08/2008
 * @see documento "Normas de programación v1.0"
 * @since jdk 5.0
 */
public class UnidadOrganizativa extends PoolDAO {

    /** Trazas de la aplicación */
    private Logger log = Logger.getLogger(UnidadOrganizativa.class);

    /** Identificador de la unidad organizativa */
    private int id;

    /** Nombre de la unidad organizativa */
    private String nombre;

    /** Obtiene el identificador de esta unidad organizativa */
    public int getId() {
        return id;
    }

    /** Establece el identificador de esta unidad organizativa */
    public void setId(int id) {
        this.id = id;
    }

    /** Obtiene el nombre de esta unidad organizativa */
    public String getNombre() {
        return nombre;
    }

    /** Establece el nombre de esta unidad organizativa */
    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    /**
     * Inserta la unidad organizativa en el sistema.
     *
     * @param unidad Unidad organizativa a insertar
     * @throws Exception Excepción elevada durante el proceso de inserción
     */
    public void insertarUnidad(UnidadOrganizativa unidad) throws Exception{

        log.debug("-> insertarUnidad(UnidadOrganizativa unidad)");

        Connection conn = null;
        PreparedStatement pstmt = null;
    }
}
```

```

StringBuffer sqlSb = null;

try {
    conn = this.dameConexion();

    sqlSb = new StringBuffer("")
        .append("INSERT INTO ORG.UNIDAD_ORGANIZATIVA ")
        .append("(ID, NOMBRE) VALUES (?, ?)");

    pstmt = conn.prepareStatement(sqlSb.toString());
    pstmt.setInt(1, unidad.getId());
    pstmt.setString(2, unidad.getNombre());
    pstmt.executeUpdate();

} catch (Exception e) {

    log.error("Error: error al insertar la unidad. " +
        "Descripción:" + e.getMessage(), e);

    throw e;

} finally {

    log.debug("<- insertarUnidad(UnidadOrganizativa unidad)");

}
}
}

```

4.3. ESTÁNDARES DE INTERFACES DE USUARIO

La interfaz de usuario es el medio con que el usuario puede comunicarse con una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

La interfaz gráfica de usuario es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o una aplicación.

A continuación se presentan los estándares que han sido definidos para la interfaz gráfica de usuario para el sistema informático de clasificación de zoología.

4.3.1. PLANTILLA

El entorno principal del sistema se encuentra dividido en 4 áreas:

- **Encabezado:** Mostrará el logotipo del MUHNES y Secretaría de Cultura, además del título del sistema y sitio público.
- **Menú y contenido:** Mostrará las opciones con las que cuenta el usuario de acuerdo a los privilegios otorgados, así también la información a la cual se tendrá acceso.
- **Pie:** Mostrará la información del MUHNES y Secretaría de Cultura.

Las 4 áreas se presentan en la figura 4.1.

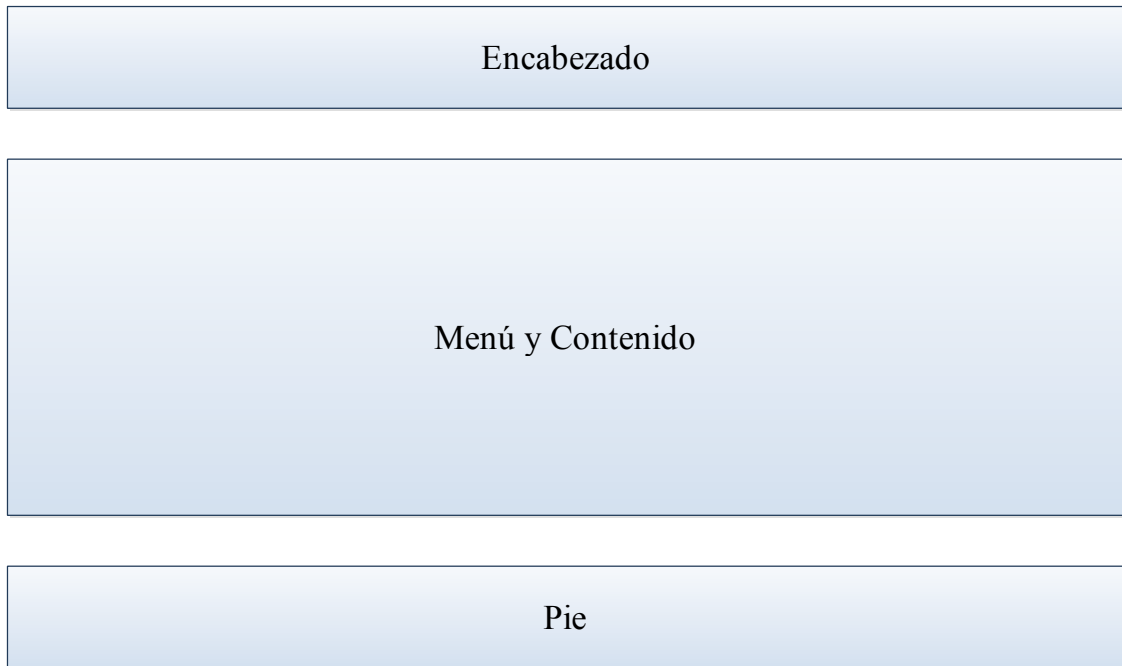


Figura 4.1. Plantilla.

El diseño del entorno principal del sistema informático se presenta a continuación en la figura 4.2.

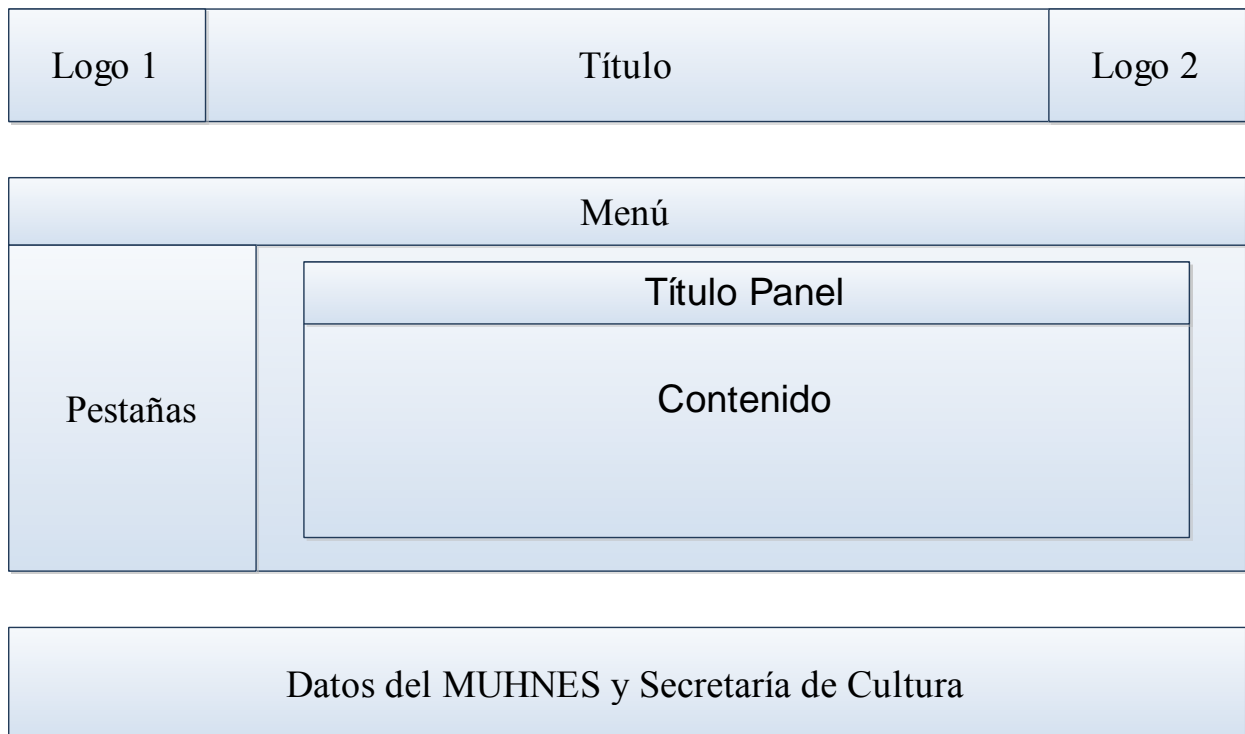


Figura 4.2. Diseño del entorno principal.

En la figura 4.3 se presenta un ejemplo completo de la distribución de los elementos del sistema en pantalla.

Sistema de Registro de Especímenes

Museo de Historia Natural de El Salvador

SECRETARÍA DE CULTURA DE LA PRESIDENCIA

Inicio Gestión de Clasificaciones Gestión del Sistema Gestión de Recursos Gestión del Área Informes Soporte

ID del Administrador
Terminar sesión

Animal	Conservación	Sexo	Madurez	Habitat	Estados Local.	Estado	Descripción Taxonomica	Estado de la muestra	Utilización	Físico	Tipo	Recolección
C20	C20	Masculino	C15	C25	C100	C20	C100	C200	C50	Ver	Ver	Ver

Fecha de recolección	Equipo de recolección	Confirma	Determina	Temperatura	Altura	Latitud	Longitud	Observaciones
01/01/2012	C200	C50	C50	N	N	N	N	C200
01/01/2012	C200	C50	C50	N	N	N	N	C200
01/01/2012	C200	C50	C50	N	N	N	N	C200

Museo de Historia Natural de El Salvador
Barro, San Jacinto, Valle de las Virgenes, Colonia Niangua, San Salvador
República de El Salvador, C.A.
(503) 2270-1387

Secretaría de Cultura de la Presidencia
Urbanización Barro Alto II, Pasaje Mar de Plata, No. 1, San Salvador
República de El Salvador, C.A.
(503) 224-2300

Figura 4.3. Ejemplo de pantalla.

El contenido del área central de la pantalla será mostrado en pestañas que estarán ubicadas en la parte izquierda de la pantalla, separando la información. Todo el contenido será mostrado dentro de un panel con su respectivo título.

Si la información a presentar necesita un orden específico, será mostrada dentro de una tabla, como se muestra en la figura 4.4.

Clasificación científica	
Phylum	Nombre del filo
Clase	Nombre de la clase
Subclase	Nombre de la subclase
Infraclase	Nombre de la infraclase
Orden	Nombre del orden
Suborden	Nombre de la suborden
Familia	Nombre de la familia
Superfamilia	Nombre de la superfamilia

Figura 4.4. Tabla de datos.

4.3.2. CONTROLES COMUNES

Toda interfaz gráfica utiliza controles que son comunes para mantener una mejor interacción con el usuario permitiéndole así adaptarse de manera eficiente al sistema que se le está presentando.

Los controles utilizados son proporcionados por los estándares html, JSF 2.0 y PrimeFaces. Estos controles se nombrarán cuando el formato de presentación tiene que ser distinto a sus valores por defecto, permitiendo una mejor presentación en pantalla. Los nombres de los componentes de la interfaz deberán seguir el formato:

nombreNemónicoAbreviaturaControl

En la tabla 4.2 se muestran varios ejemplos de nombres de objetos:

Etiqueta	Descripción
<code><h:inputText id="duiTxt"/></code>	El control almacena el DUI de una persona.
<code><p:tab id="recolectorTab"></code>	Pestaña que muestra los recolectores
<code><h:selectBooleanCheckBox id="activoChk"/></code>	Este checkBox almacena el estado activo.
<code><h:form id="usuarioFrm"></code>	Formulario de mantenimiento de usuarios.
<code><p:dataTable id="clasificacionCientificaTbl"></code>	Tabla que muestra la clasificación científica.

Tabla 4.2. Nombre de los objetos.

La forma, color, tipo de letra y orientación de los controles es proporcionada por PrimeFaces 3.3.1, que es una extensión al estándar JSF 2.0, aplicando el tema visual bluesky-1.0.6. Sus valores por defecto pueden ser ajustados haciendo uso de hojas de estilo en cascada (css).

4.3.3. NOMBRADO DE ARCHIVOS

El sistema informático funcionará bajo un entorno web, el cual hace uso de las extensiones mostradas en la tabla 4.3.

Tipo de archivo	Extensión
Páginas web	xhtml
Hojas de estilo en cascada	css
Imágenes	jpg, png
Iconos	png, ico
Archivos de configuración	xml

Tabla 4.3. Nombrado de archivos.

Los nombres de los archivos deben ser sustantivos. Cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúsculas y que describan la función específica que realizarán. Evitar uso de símbolos como la tilde.

4.3.4. ESTRUCTURA DE DIRECTORIOS

La estructura de directorios que se usarán para organizar la interfaz de usuario se presenta en la figura 4.5.

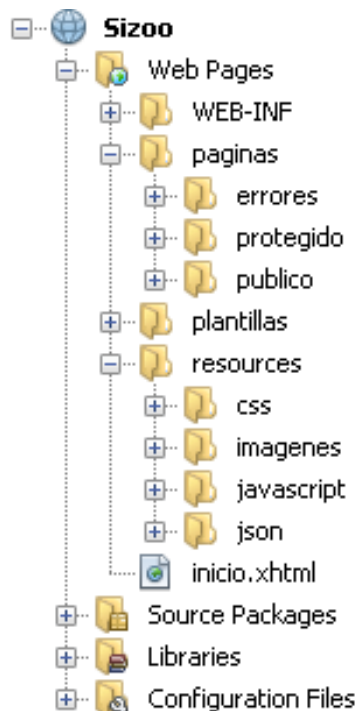


Figura 4.5. Estructura de directorios de la interfaz de usuario.

- **WEB-INF:** Dentro de esta carpeta se encontrarán los archivos de configuración de la interfaz.
- **paginas:** Contendrá todas las interfaces con extensión xhtml, a excepción de inicio.xhtml que estará en la raíz.
 - **errores:** Almacenará las páginas referentes a errores del protocolo http y excepciones de la aplicación.
 - **protegido:** Contendrá las páginas para las que se requiera autenticación de usuario.
 - **publico:** En esta carpeta se ubicarán las páginas que pueden ser accedidas por cualquier persona sin requerir credenciales de acceso. La única excepción es inicio.xhtml.

- **plantillas:** Esta carpeta tendrá las plantillas utilizadas para la interfaz, en formato *xhtml*.
- **resources:** Carpeta que contendrá directorios de librería (funcionalidad de JSF), para poder acceder a cualquier archivo contenido en ésta con la misma dirección desde cualquier página.
 - **css:** Contendrá todas las hojas de estilo con extensión *css*.
 - **imagenes:** Contendrá todas las imágenes de la interfaz en formato *jpg*.
 - **javascript:** Contendrá todos los script a utilizar en el sitio con formato *js*.
 - **json:** Contendrá archivos de configuración en formato *json*.

4.4. ESTÁNDARES DE DOCUMENTACIÓN

Una buena documentación es esencial para lograr un diseño correcto y un mantenimiento eficiente de los sistemas de información. Además de ser precisa y completa, la documentación debe ser instructiva, de modo que un ingeniero de pruebas, técnico de mantenimiento, o inclusive el usuario final, pueda averiguar cómo funciona el sistema con solo leer la documentación.

La documentación ha sido dividida en:

- Documentos impresos
- Reportes

4.4.1. ESTÁNDAR DE DOCUMENTOS IMPRESOS

Los documentos impresos tendrán el formato presentado a continuación:

- **Formato general de los documentos:** el formato general de los documentos impresos se presenta en la tabla 4.4.

Elemento	Formato
Color de hojas	Blanco
Tamaño de hojas	Carta
Orientación	Vertical
Margen superior	2.5 cm
Margen inferior	2.5 cm
Margen izquierdo	2.5 cm
Margen derecho	2.5 cm
Espaciado entre párrafos	6 pts
Interlineado	1.15 pts
Tipo de letra	Times New Roman
Color de letra	Negro
Tamaño de letra para títulos nivel 1	14
Tamaño de letra para títulos nivel 2	12
Tamaño de letra para títulos nivel 3	11
Tamaño de letra cuerpo del documento	11
Estilo de letra títulos nivel 1, 2 y 3	Negrita, versales

Tabla 4.4. Formato general del documento.

- **Contenido de los documentos impresos:** los documentos impresos tendrán la siguiente estructura:
 - Portada
 - Índice
 - Introducción
 - Objetivos
 - Cuerpo del documento
- **Portada:** la portada de los documentos impresos tendrán el formato presentado en la figura 4.6.

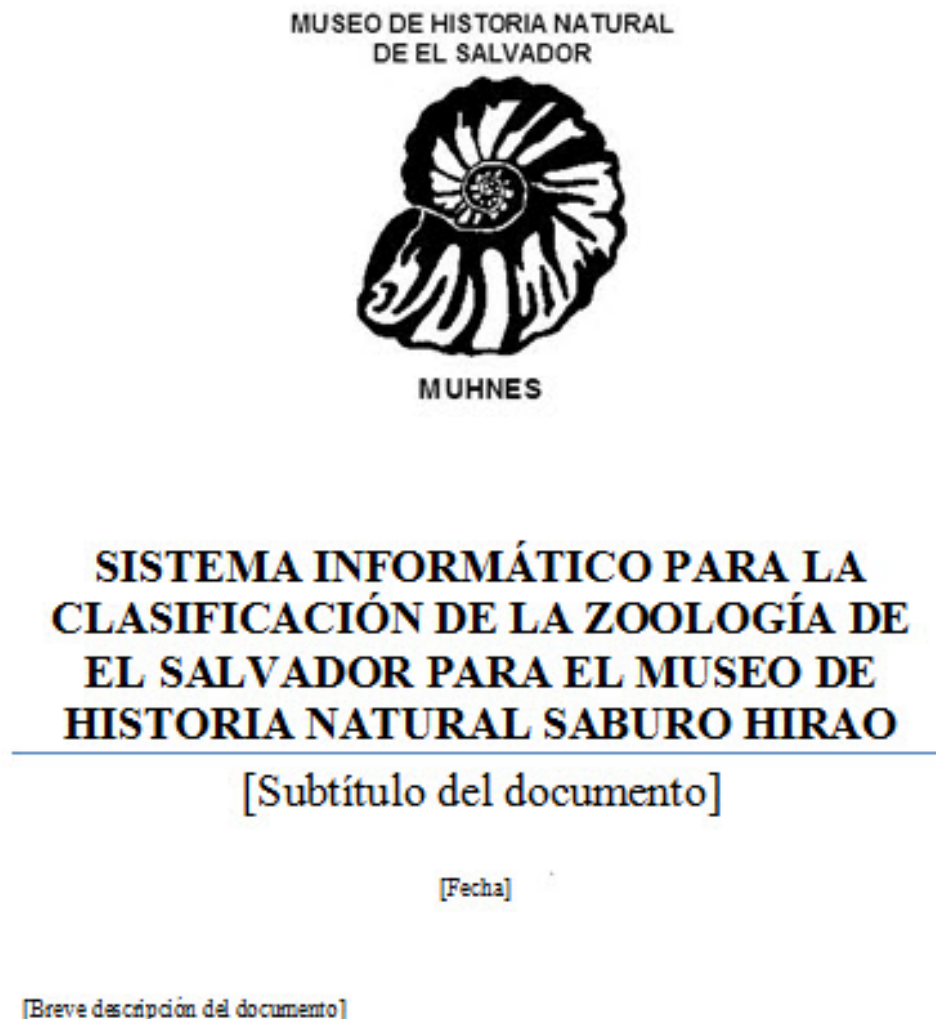


Figura 4.6. Formato de portada.

- **Encabezados y pie de página:** los formatos a usar para los encabezados y pie de páginas de los documentos impresos se presentan en las figura 4.7 y 4.8 respectivamente.

[Subtítulo del documento]

 [Fecha]

Figura 4.7. Formato de encabezado de página.

_____ **Página#** _____

Figura 4.8. Formato de pie de página.

- **Numeración y viñetas:** la numeración y viñetas son muy importantes al momento de presentar la información de forma ordenada. El formato a seguir para la numeración y viñetas se muestra en la tabla 4.5.

Numeración	Viñetas
1. Título nivel 1 1.1. Título nivel 2 1.1.1. Título nivel 3 2. Título nivel 1 2.1. Título nivel 2 2.1.2. Título nivel 3 2.2. Título nivel 2	<ul style="list-style-type: none"> • Elemento 1 <ul style="list-style-type: none"> ○ Elemento 2 <ul style="list-style-type: none"> ▪ Elemento 3 • Elemento 1 <ul style="list-style-type: none"> ○ Elemento 2 ○ Elemento 2 <ul style="list-style-type: none"> ▪ Elemento 3

Tabla 4.5. Ejemplo de numeración y viñetas.

- **Tablas:** las tablas serán nombradas con el número de tema y su correlativo, por ejemplo *Tabla 2.4*. En la figura 4.9 se muestra un ejemplo del estilo de tablas.

Columna 1	Columna 2	Columna 3

Tabla#.#. Título de tabla.

Figura 4.9. Ejemplo de tablas.

- **Imágenes:** las imágenes serán nombradas con el número de tema y su correlativo, por ejemplo *Figura 2.4.* en la figura 4.10 se muestra un ejemplo imágenes.

MUSEO DE HISTORIA NATURAL
DE EL SALVADOR



MUHNES

Figura#.#. **Titulo de la figura.**

Figura 4.10. Ejemplo de imágenes.

- **Líneas de código:** las líneas de código (tanto el lenguaje de programación como sentencias SQL) dentro del documento se presentaran con el estilo de letra Courier New, tamaño 10, interlineado sencillo y espaciado superior e inferior de 0 pts. Por ejemplo:

```
CREATE TABLE Privilegio(
    CodigoPrivilegio      INT          NOT NULL,
    Accion                VARCHAR(50) NOT NULL,
    CONSTRAINT PK_Privilegio PRIMARY KEY (CodigoPrivilegio)
);
```

4.4.2. ESTÁNDAR DE REPORTES

El objetivo de los reportes es la de presentar al usuario la información impresa en un determinado formato únicamente visual o en papel.

Los reportes se visualizarán en pantalla y luego podrán ser impresos si así lo desea el usuario. El estándar a seguir para los reportes se muestra en la figura 4.11.



MUSEO DE HISTORIA NATURAL
DE EL SALVADOR

MUHNES

<<Encabezado>>

<<Titulo del reporte>>

<<Criterios de búsqueda>>

<<Fecha de impresión>>

dd/mm/aaaa

<<Datos del reporte>>

Columna 1	Columna 2	Columna 3

<<Pie de página>>

Página <a> de <n>

Figura 4.11. Formato de reportes.

Los elementos del reporte son:

- **Encabezado:** el encabezado de reporte contendrá:
 - **Logotipo:** presenta el logo correspondiente al MUHNES. Las dimensiones del logo son de 5cm en largo y ancho.
 - **Titulo del reporte:** se identifica el reporte generado y a continuación se muestran los parámetros del reporte que son los criterios utilizados para filtrar los datos que forman parte del reporte. Estos parámetros pueden visualizarse de una de las siguientes maneras:
 - Para reportes con un solo criterio de búsqueda, se podrá incorporar como parte del título del reporte
 - Para reportes con varios criterios de consulta se tendrán que definir como parte de esta sección identificada como Datos del reporte.

- **Fecha de impresión:** representa la fecha de impresión del informe, donde dd es el día, mm es el mes y aaaa es en año.
- **Datos del reporte:** contendrá la información mostrada para cada una de las pantallas de salida. El detalle de la información tienen formatos muy diferentes dependiendo de la naturaleza y la complejidad de la información del mismo, lo que dificulta establecer un estándar para su presentación. El reporte podrá tener en su detalle:
 - El encabezado de columnas por las cuales la información será agrupada.
 - El detalle de cada registro que es resultado de los elementos que cumplieron con los criterios de búsqueda. Serán desplegados de forma vertical y si realmente se requiere, el uso de informes en páginas con orientación horizontal.
- **Pie de página:** presenta el número de página actual a con respecto al número total de páginas n .

El formato de los elementos que conforman el reporte se presenta en la tabla 4.6.

Elemento	Descripción
Tipo de letra	Times New Roman
Tamaño de letra títulos	12
Tamaño de letra información	11
Tamaño del papel	Carta
Márgenes	2.5 cm
Orientación	Vertical y horizontal
Colores	Blanco y negro

Tabla 4.6. Estándar de los reportes.

Las fichas técnicas contienen toda la información relacionada a los especímenes. El formato de los elementos que la conforman se muestra en la tabla 4.7.

Elemento	Descripción
Tipo de letra	Times New Roman
Tamaño de letra de títulos	12
Tamaño de letra del contenido	11
Tamaño del papel	Oficio
Márgenes	2.0 cm
Orientación	Vertical
Colores	Blanco y negro texto, color fotografía
Fotografía	227px X 165px

Tabla 4.7. Estándar de ficha técnica.

El diseño de la ficha técnica se muestra en la figura 4.12.

MUSEO DE HISTORIA NATURAL DEL EL SALVADOR
Sección de <<nombre de sección>>

Número de museo: _____
Ubicación en inventario: _____
*Número de muestras: _____

FOTOGRAFIA

Nombre científico: _____
Nombre común: _____
Sinonimia: _____
Phylum: _____
Clase: _____
Orden: _____
Familia: _____
Género: _____
Especie: _____
Estado de la muestra: _____
Muestra en: _____
Textura: _____
Coloración: _____
Longitud: _____ Diámetro: _____ Altura: _____ Peso: _____
Sexo: _____ Madurez: _____
Descripción taxonómica: _____
Hábitat: _____
Estatus local: _____
Hospedero: _____
Estado del hospedero: _____
**Profundidad: _____ Salinidad: _____ PH: _____ Condición de marea: _____
***Tipo de sustrato: _____ Altitud: _____
Fecha de recolección: _____
Lugar de recolección: _____
Equipo de recolección: _____
Rcolector: _____
Confirno: _____
Determino: _____
Observaciones: _____

* Solo malacología

** Solo acuáticos

*** Solo terrestres y aves

Figura 4.12. Formato de ficha técnica.

Además, el sistema informático permitirá la exportación de los reportes a otros formatos para ser guardados. Estos formatos son:

- Documentos de texto doc.
- Documentos pdf.
- Hojas electrónicas xls.

CAPÍTULO 5:

PRUEBAS

Las pruebas tienen gran importancia para demostrar que el software cumpla con los requerimientos especificados por los usuarios y que esté acorde a la lógica del negocio de sus procesos.

5.1. PRUEBAS TÉCNICAS

5.1.1. PRUEBAS UNITARIAS

Las pruebas unitarias consisten en probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

Las ventajas que se tienen al realizar este tipo de pruebas son:

- Permiten la modificación del código para mejorar su estructura, verificando si los resultados obtenidos con estos cambios son los mismos que resultaron al inicio.
- Permiten llegar a la fase de integración con un grado alto de seguridad de que el código está funcionando correctamente.
- Se pueden realizar las pruebas al código de funcionamiento y a la interfaz de forma independiente.
- Las propias pruebas son documentación del código puesto que ahí se puede ver cómo utilizarlo.
- Los errores son más fáciles de localizar ya que se trabaja por separado cada módulo.

Las pruebas unitarias realizadas a cada uno de los módulos que conforman el sistema es la técnica de la caja negra, asistido por las librerías proporcionadas por JUnit.

Se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. La representación gráfica de este método se puede apreciar en la figura 5.1.



Figura 5.1. Representación gráfica de la caja negra

JUnit es un conjunto de clases que permite realizar la ejecución de clases Java de manera controlada, para poder evaluar si el funcionamiento de cada uno de los métodos de la clase se comporta como se espera. Es decir, en función de algún valor de entrada se evalúa el valor de retorno esperado; si la clase cumple con la especificación, entonces JUnit devolverá que el método de la clase pasó exitosamente la prueba; en caso de que el valor esperado sea diferente al que regresó el método durante la ejecución, JUnit devolverá un fallo en el método correspondiente.

JUnit se encuentra incluido en NetBeans, lo que facilita la implementación de pruebas unitarias.

Las pruebas unitarias se realizaron con el formato presentado en la tabla 5.1.

Pruebas unitarias del sistema				
Tipo de prueba	Unitaria	Método de prueba	Caja negra	
Objetivo	Realizar las pruebas unitarias de caja negra a cada uno de los módulos que conforman el sistema.			
Correlativo	Clase	Método	Resultado esperado	Resultado obtenido
<<identificador de la prueba>>	<<nombre de la clase>>	<<nombre del método>>	<<descripción del resultado que se espera>>	<<exitoso o fallido>>

Tabla 5.1. Formato de pruebas unitarias

El resultado de las pruebas unitarias se muestra en el anexo C.1.

5.1.2. PRUEBAS DE INTEGRACIÓN

Las pruebas de integración son aquellas en las que módulos individuales del sistema son combinados y probados como un grupo. Son las pruebas posteriores a las pruebas unitarias y preceden a las pruebas del sistema.

En las pruebas de integración se examinan las interfaces entre grupos de componentes o subsistemas para asegurar que son llamados cuando es necesario y que los datos que se transmiten son los requeridos.

El tipo de pruebas de integración realizadas en el proyecto fueron pruebas de integración descendente.

El método de integración descendente consiste en comenzar por el módulo de control principal e ir incorporando módulos subordinados progresivamente. También se le conoce como técnica como Top-Down.

No hay un orden adecuado de integración, pero se recomienda:

- Integrar lo antes posible secciones críticas o complejas.
- El orden de integración debe incorporar cuanto antes los módulos de entrada/salida para facilitar la ejecución de pruebas.

Una representación gráfica de esta metodología se presenta en la figura 5.2.

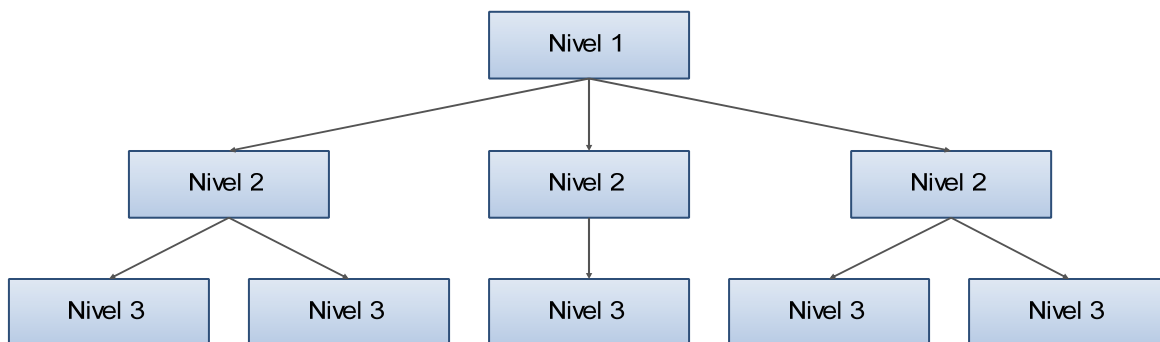


Figura 5.2. Representación gráfica de pruebas de integración descendente

Las pruebas se realizaron en profundidad, es decir, se probó primero la rama más a la izquierda hasta llegar a la rama final de la derecha. Cada rama representa una operación es específico, por ejemplo la rama más a la izquierda representa la captura de datos, y la rama más a la derecha la consulta de estos datos.

En la tabla 5.2 se muestra el formato utilizado para realizar las pruebas de integración.

Pruebas de integración del sistema			
Tipo de prueba	Integración	Método de prueba	Descendente
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.		
Correlativo	Operación a realizar	Resultado esperado	Resultado obtenido
<<identificador de la prueba>>	<<nombre de la operación a realizar >>	<<descripción del resultado que se espera>>	<<exitoso o fallido>>

Tabla 5.2. Formato de pruebas de integración.

El resultado de las pruebas unitarias se muestra en el anexo C.2.

5.1.3. PRUEBAS DE SISTEMA

Es el proceso de prueba de un sistema integrado de hardware y software. Las pruebas de sistema buscan:

- El cumplimiento de todos los requerimientos funcionales.
- El funcionamiento y rendimiento en las interfaces hardware, software, de usuario y de operador
- La adecuación de la documentación de usuario.
- La ejecución y rendimiento en condiciones límite y de sobrecarga.

Las pruebas de sistema que se realizaron fueron:

- **Pruebas de recuperación:** Este tipo de pruebas fuerzan un fallo del software y verifica que la recuperación se lleva a cabo apropiadamente.

El objetivo de estas pruebas es verificar que el proceso de recuperación del sistema restaura apropiadamente la base de datos, la aplicación y el sistema a un estado conocido o deseado.

- **Pruebas de seguridad:** Las pruebas de seguridad consisten en verificar los mecanismos de control de acceso al sistema para evitar alteraciones indebidas en los datos.

Durante este tipo de prueba el responsable desempeña el papel de intruso, tratando de violar la seguridad del sistema e intentando obtener las claves de acceso por cualquier medio externo.

- **Pruebas de resistencia:** Las pruebas de resistencia están diseñadas para confrontar el sistema integrado con situaciones anormales.

Las pruebas de resistencia ejecutan un sistema de tal manera que requiera una frecuencia o un volumen anormal de recursos, por ejemplo la sobrecarga de datos.

- **Pruebas de desempeño:** El objetivo de las pruebas de desempeño o rendimiento es asegurar que el sistema responde satisfactoriamente en los márgenes establecidos en cuanto al tiempo de respuesta, tiempo de ejecución y tiempo de utilización de recursos, así como los volúmenes de espacio en disco y capacidad.

En la tabla 5.3 se muestra el formato utilizado para realizar las pruebas de sistema.

Pruebas de sistema			
Fecha	<<día/mes/año>>	No. de caso de prueba	<<identificador de la prueba>>
Tipo de prueba	Sistema	Subtipo de prueba	<<seguridad, recuperación, desempeño o resistencia>>
Objetivo	<< objetivo que persigue la prueba>>		
Descripción	<<descripción de la prueba>>		
Procedimiento	<<listar el procedimiento a seguir>>		
Condiciones	<<listar las condiciones de la prueba>>		
Resultados			
Resultados esperados	<<descripción de los resultados que se esperan>>		
Resultados obtenidos	<<descripción de los resultados obtenidos>>		
Evaluación	<<exitoso o fallido>>		
Observaciones	<<observación sobre la prueba realizada>>		

Tabla 5.3. Formato de pruebas de sistema.

Luego de haber realizado las pruebas correspondientes, se procedió a completar el formato presentado en la tabla 5.4.

Verificación de las pruebas de sistema		
Tipo de prueba	Resultado de la prueba	Observaciones
Seguridad	<<exitoso o fallido>>	<<observaciones sobre los tipos de pruebas>>
Recuperación	<<exitoso o fallido>>	<<observaciones sobre los tipos de pruebas>>
Desempeño	<<exitoso o fallido>>	<<observaciones sobre los tipos de pruebas>>
Resistencia	<<exitoso o fallido>>	<<observaciones sobre los tipos de pruebas>>

Tabla 5.4. Formato de verificación de pruebas de sistema.

Los resultados de las pruebas de sistema se muestran en el anexo C.3.

5.2. PRUEBAS DE FUNCIONAMIENTO


Los casos de prueba que el equipo de trabajo realizó para comprobar el funcionamiento del sistema informático SIZOO se presenta a continuación:

- **Inicio de sesión de Administrador:**

Caso de prueba:	Iniciar sesión de Administrador.
Propósito:	Ingresar al sistema como un usuario registrado con el correo electrónico del usuario y contraseña establecida.
Prerrequisito:	El usuario debe estar registrado en el sistema y contar con rol de administrador.
Datos de prueba correctos:	<ul style="list-style-type: none">• Correo: administrador@gob.sv• Contraseña: 12345
Datos de prueba erróneos:	<ul style="list-style-type: none">• Correo: administrador@gob.svx• Contraseña: 12345
Pasos:	<ol style="list-style-type: none">1. Ingresar a la página principal del sistema.2. Ingresar el correo electrónico de usuario y contraseña.3. Hacer clic en el botón Iniciar sesión.
Salida esperada:	Visualizar la interfaz de sesión del administrador del sistema.
Salida Obtenida:	Visualizar la interfaz de sesión del administrador del sistema.
Observaciones:	Respetar las letras mayúsculas y minúsculas.

- Inicio de sesión correcto:

Inicio de Sesión

 Para poder ingresar debe haber recibido sus credenciales previamente del administrador del sistema

Correo:
Ejemplo: usuario@secultura.gob.sv

Contraseña:

[¿Olvidó su contraseña?](#)

- Inicio de sesión fallido:

Inicio de Sesión



Para poder ingresar debe haber recibido sus credenciales previamente del administrador del sistema

 El usuario o la contraseña son incorrectos


Correo:

Ejemplo: usuario@secultura.gob.sv

Contraseña:

[¿Olvidó su contraseña?](#)

- **Gestión de Operaciones**

Caso de prueba:	Gestión de operaciones.
Propósito:	Ingresar al sistema los distintos operaciones que se pueden efectuar dentro del sistema
Prerrequisito:	El usuario debe tener privilegios de crear operaciones.
Datos de prueba correctos:	<ul style="list-style-type: none"> Nombre de operación: Consultar Descripción de operación: Consultar información del sistema
Datos de prueba erróneos:	<ul style="list-style-type: none"> Nombre de operación: Consultar 2 Descripción de operación:
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Sistema/Operaciones. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de operaciones el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de operaciones el nuevo creado.
Observaciones:	Evite el uso de números y los campos son obligatorios

- Ingreso de operación correcto:

Datos del Operacion

Nombre de la operacion:

Descripcion de la operacion:

- Ingreso de operación erróneo:

Datos del Operacion

Nombre de la operacion:

Descripcion de la operacion:

- **Gestión de objetos:**

Caso de prueba:	Gestión de objetos.
Propósito:	Ingresar al sistema los distintos objetos que conforman el sistema.
Prerrequisito:	El usuario debe tener privilegios de crear objetos.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre de objeto: espécimen • Descripción del objeto: tabla de especímenes
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre de objeto: • Descripción del objeto:
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Sistema/Objetos. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de objetos el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de objetos el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda en el campo Nombre de objeto, escribirlo en minúscula sin signos de puntuación ni acentos. • los campos son obligatorios

- Ingreso de operación correcto:

Datos del Objeto

Nombre del objeto:

Descripcion del objeto:

- Ingreso de operación erróneo:

Datos del Objeto

Nombre del objeto:

Descripcion del objeto:

- **Gestión de privilegios:**

Caso de prueba:	Gestión de privilegios.
Propósito:	Ingresar al sistema los distintos privilegios que se tendrán dentro del sistema.
Prerrequisito:	El usuario debe tener privilegios de crear usuario.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre: consultar usuario • Descripción: privilegio de prueba • Operación: consultar • Objeto: usuario
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre: Guardar Usuario 2 • Descripción: Ejemplo erróneo de datos • Operación: generar • Objeto: phylum
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Sistema/Privilegios. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de privilegios el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de privilegios el nuevo creado.
Observaciones:	Respetar las letras mayúsculas y minúsculas.

- Ingreso de privilegio correcto:

The screenshot shows a dialog box titled "Gestión de privilegio" with a close button (X). It contains the following fields:

- Nombre:** A text input field containing "consultar usuario".
- Descripción:** A text area containing "Privilegio de prueba". Below it, it says "480 Caracteres Restantes".
- Operación:** A dropdown menu with "consultar" selected.
- Objeto:** A dropdown menu with "usuario" selected.

At the bottom, there are two buttons: "Cancelar" and "Guardar".

- Ingreso de privilegio erróneo:

The screenshot shows a dialog box titled "Gestión de privilegio" with a close button (X). It contains the following fields:

- Nombre:** A text input field containing "Guardar Usuario 2", which is highlighted with a red border.
- Descripción:** A text area containing "Ejemplo erroneo de datos". Below it, it says "475 Caracteres Restantes".
- Operación:** A dropdown menu with "generar" selected.
- Objeto:** A dropdown menu with "phylum" selected.

At the bottom, there are two buttons: "Cancelar" and "Guardar".

- **Gestión de roles:**

Caso de prueba:	Gestión de roles.
Propósito:	Ingresar al sistema los distintos roles que se tendrán asociados a los privilegios.
Prerrequisito:	El usuario debe tener privilegios de crear roles.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre: Prueba • Descripción: Rol de prueba • Privilegios: agregar sesión, modificar sesión, eliminar sesión, agregar rol
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre: Prueba • Descripción:
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Sistema/Roles. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar. 4. Conceder los privilegios deseados. 5. Cerrar la ventana de privilegios.
Salida esperada:	Visualizar en la tabla de roles el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de roles el nuevo creado.
Observaciones:	El campo Descripción es obligatorio.

- Ingreso de rol correcto:

The screenshot shows a web form titled "Gestión de rol". It contains two input fields: "Nombre" with the value "Prueba" and "Descripción" with the value "Rol de prueba". Below the description field, it indicates "487 Caracteres Restantes". At the bottom of the form, there are two buttons: "Cancelar" and "Guardar".

Nombre	Operación Seleccione...	Objeto Seleccione...	Permiso Habilitado	Opciones
Agregar sesion	agregar	sesion	Deshabilitado	Revocar
Modificar sesion	modificar	sesion	Deshabilitado	Revocar
Eliminar sesion	eliminar	sesion	Deshabilitado	Revocar
Agregar rol	agregar	rol	Deshabilitado	Revocar
Modificar rol	modificar	rol	Habilitado	Conceder
Eliminar rol	eliminar	rol	Habilitado	Conceder
Agregar privilegio	agregar	privilegio	Habilitado	Conceder
Modificar privilegio	modificar	privilegio	Habilitado	Conceder
Eliminar privilegio	eliminar	privilegio	Habilitado	Conceder
Agregar operación	agregar	operacion	Habilitado	Conceder

- Ingreso de rol erróneo:

Gestión de rol

Nombre:

Descripción:
500 Caracteres Restantes.

- **Gestión de usuarios:**

Caso de prueba:	Gestión de usuarios.
Propósito:	Ingresar al sistema los usuarios, quienes tendrán uno o varios roles asociados.
Prerrequisito:	El usuario debe tener privilegios de crear usuarios.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre: Roxana Cardoza • Correo: rcardoza@muhnes.gob.sv • Contraseña: 12345 • Confirmación: 12345 • Teléfono: (503)2222-1212 • Jefe: Jaime Alas • Rol: conceder Biólogo
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre: Roxana Cardo5a • Correo: www.muhnes.com.sv • Contraseña: 12345 • Confirmación: 67890 • Teléfono: 2222-1212 • Jefe: Jaime Alas
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Sistema/Usuarios. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar. 4. Conceder los roles deseados. 5. Cerrar la ventana de roles.
Salida esperada:	Visualizar en la tabla de usuarios el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de usuarios el nuevo creado.
Observaciones:	Respetar los formatos de correo electrónico y número telefónico.

- Ingreso de usuario correcto:

Gestión de usuario

Nombre:

Correo:

Contraseña:

Confirmación:

Telefono:

Jefe: ▼

Roles		
Nombre	Concedido	Opciones
Biologo	Si	Revocar
Administrador	No	Conceder
Digitador	No	Conceder

- Ingreso de usuario erróneo:

Gestión de usuario

Nombre:

Correo:

Contraseña:

Confirmación:

Telefono:

Jefe:

- **Gestión de departamentos:**

Caso de prueba:	Gestión de departamentos.
Propósito:	Ingresar los departamentos de El Salvador al sistema.
Prerrequisito:	El usuario debe tener privilegios de crear departamentos.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre del Departamento: Nuevo Departamento • Región Geográfica: central
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre del Departamento: Departamento 1 • Región Geográfica: central
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Geografía/Departamentos. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de departamentos el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de departamentos el nuevo creado.
Observaciones:	Se recomienda respetar mayúsculas y minúsculas.

○ Ingreso de departamento correcto:

Departamento

Nombre del Departamento:

Región Geográfica:

○ Ingreso de departamento erróneo:

Departamento

Nombre del Departamento:

Región Geográfica:

● **Gestión de municipios:**

Caso de prueba:	Gestión de municipios.
Propósito:	Ingresar los municipios de El Salvador al sistema.
Prerrequisito:	El usuario debe tener privilegios de crear municipios.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Departamento: Nuevo Departamento • Nombre: Nuevo Municipio • Latitud: 14.02 • Longitud: -88.23 • Cabecera: sin seleccionar
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Departamento: Nuevo Departamento • Nombre: Municipio 2 • Latitud: 17.19 • Longitud: -80.12 • Cabecera: seleccionar
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Geografía/Municipios. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de municipios el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de municipios el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Los valores de latitud son entre 12 y 16 y la longitud entre -91 y -84.

- Ingreso de municipio correcto:

Gestión de municipio

Departamento:

Nombre:

Latitud:

Longitud:

Cabecera: *

- Ingreso de municipio erróneo:

Gestión de municipio

Departamento:

Nombre:

Latitud:

Longitud:

Cabecera: *

- **Gestión de áreas protegidas:**

Caso de prueba:	Gestión de áreas protegidas.
Propósito:	Ingresar las áreas protegidas de El Salvador al sistema.
Prerrequisito:	El usuario debe tener privilegios de creación de áreas protegidas.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre del área protegida: Nueva área protegida • Descripción: Área protegida de prueba • Municipios: Seleccione Nuevo Departamento, Nuevo Municipio
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Geografía/Áreas protegidas. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de áreas protegidas el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de áreas protegidas el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Se pueden seleccionar varios municipios.

- Ingreso de departamento correcto:

Datos del Área Protegida

Nombre del área protegida:

Descripción:
Restan 978 caracteres.

Municipio(s):

- **Gestión de zonas:**

Caso de prueba:	Gestión de zonas.
Propósito:	Ingresar las áreas rurales, urbanas y reservas de El Salvador al sistema.
Prerrequisito:	El usuario debe tener privilegios de creación de zonas.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Tipo de zona: Urbano • Municipio: Nuevo Departamento, Nuevo municipio • Barrio: Nuevo barrio • Colonia: Nueva Colonia • Sector: Nuevo Sector
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Geografía/Departamentos. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de zonas el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de zonas el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • La zona indica un lugar más específico dentro de un área protegida o municipio.

- Ingreso de zonas correcto:

The image shows two sequential screenshots of a web form titled "Datos de Zona".

First Screenshot: The form has three radio buttons for "Tipo de zona": "Rural", "Urbano" (which is selected), and "Reserva". Below this is a dropdown menu for "Municipio" with the selected value "Nuevo Departamento, Nuevo Municipio". At the bottom right, there are three buttons: "Cancelar", "Guardar", and "→ Siguiente".

Second Screenshot: This screenshot shows the next step in the form. It has three text input fields: "Barrio" with the value "Nuevo barrio", "Colonia" with the value "Nueva Colonia", and "Sector" with the value "Nuevo Sector". At the bottom left, there is a button "← Anterior". At the bottom right, there are two buttons: "Cancelar" and "Guardar".

- **Gestión de recolectores:**

Caso de prueba:	Gestión de recolectores.
Propósito:	Ingresar los datos de los recolectores del MUHNES.
Prerrequisito:	El usuario debe el privilegio de agregar recolector.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre completo: Sandra Celina González • Alias: S.G. • Dirección: San Jacinto • Teléfono: (503)2212-1212 • Correo electrónico: s.gonzalez@muhnes.gob.sv
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre completo: Sofía 4las • Alias: • Dirección: San Jacinto • Teléfono: • Correo electrónico: www.google.com
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Recolecciones/Recolector. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de recolectores el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de recolectores el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • El campo Alias es opcional.

- Ingreso de recolector correcto:

Datos del Recolector

Nombre completo:

Alias:

Dirección:

Teléfono:

Correo electrónico:

- Ingreso de recolector erróneo:

Datos del Recolector

Nombre completo:

Alias:

Dirección:

Teléfono:

Correo electrónico:

- **Gestión de phylum:**

Caso de prueba:	Gestión de phylum.
Propósito:	Ingresar los phylum de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar phylum.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre del Phylum: Chordata
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre del Phylum: Prueba1
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Phylum. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de phylum el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de phylum el nuevo creado.
Observaciones:	Se recomienda respetar mayúsculas y minúsculas.

○ Ingreso de phylum correcto:

Datos del Phylum

Nombre del Phylum:

○ Ingreso de phylum erróneo:

Datos del Phylum

Nombre del Phylum:

- **Gestión de clases:**

Caso de prueba:	Gestión de clases.
Propósito:	Ingresar las clases de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar clase.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Nombre del Phylum: Chordata • Nombre de la clase: Mammalia • Nombre de la sub-clase: Theria • Nombre de la infra-clase: Euteria
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Nombre del Phylum: Chordata • Nombre de la clase: Mamma-lia
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Clase. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de clases el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de clases el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Sub-clase e Infra-clase son campos opcionales.

- Ingreso de clase correcto:

Datos de la Clase

Nombre del phylum:

Nombre de la clase:

Nombre de la sub-clase:

Nombre de la infra-clase:

- Ingreso de clase erróneo:

Datos de la Clase

Nombre del phylum:

Nombre de la clase:

Nombre de la sub-clase:

Nombre de la infra-clase:

- **Gestión de órdenes:**

Caso de prueba:	Gestión de órdenes.
Propósito:	Ingresar las órdenes de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar orden.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Clase Sub-clase Infra-clase: Mammalia Theria Eutheria • Nombre del orden: Carnivora • Nombre del sub-orden: Caniformia
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Clase Sub-clase Infra-clase: Mammalia Theria Eutheria • Nombre del orden: Carnivora 2
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Orden. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de órdenes el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de órdenes el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Sub-orden es un campo opcional.

- Ingreso de orden correcto:

Datos de la Orden

Clase Sub-clase Infra-clase:

Nombre del orden:

Nombre del sub-orden:

- Ingreso de orden erróneo:

Datos de la Orden

Clase Sub-clase Infra-clase:

Nombre del orden:

Nombre del sub-orden:

- **Gestión de súper-familias:**

Caso de prueba:	Gestión de súper-familias.
Propósito:	Ingresar las clases de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar súper-familia.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Orden Sub-orden: Carnivora Caniformia • Nombre de la súper-familia: Canoidea
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Orden Sub-orden: Carnivora Caniformia • Nombre de la súper-familia: Canoidea 1
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Superfamilia. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de súper-familias el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de súper-familias el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas.

- Ingreso de súper-familia correcto:

Datos de Súper-Familia

Orden Sub-orden:

Nombre de la Súper-familia:

- Ingreso de súper-familia erróneo:

Datos de Súper-Familia

Orden Sub-orden:

Nombre de la Súper-familia:

- **Gestión de familias:**

Caso de prueba:	Gestión de familias.
Propósito:	Ingresar las familias de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar familia.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Súper-familia: Canoidea • Nombre de la familia: Canidae • Nombre de la sub-familia: • Nombre de la tribu:
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Súper-familia: Canoidea • Nombre de la familia: Canidae • Nombre de la sub-familia: • Nombre de la tribu:
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Familia. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de familias el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de familias el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Sub-familia y Tribu son campos opcionales.

- Ingreso de familia correcto:

Datos de Familia

Súper-familia:

Nombre de la familia:

Nombre de la sub-familia:

Nombre de la tribu:

- Ingreso de familia erróneo:

Datos de Familia

Súper-familia:

Nombre de la familia:

Nombre de la sub-familia:

Nombre de la tribu:

- **Gestión de géneros:**

Caso de prueba:	Gestión de géneros.
Propósito:	Ingresar los géneros de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar género.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Familia Sub-familia Tribu: Canidae • Nombre del género: Canis
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Familia Sub-familia Tribu: Canidae • Nombre del género: Canis 2
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Género. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de géneros el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de géneros el nuevo creado.
Observaciones:	Se recomienda respetar mayúsculas y minúsculas.

- Ingreso de género correcto:

Datos de Género

Familia Sub-familia tribu:

Nombre del género:

- Ingreso de género erróneo:

Datos de Género

Familia Sub-familia tribu:

Nombre del género:

- **Gestión de especies:**

Caso de prueba:	Gestión de especies.
Propósito:	Ingresar las especies de los animales.
Prerrequisito:	El usuario debe el privilegio de agregar especie.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Género: Canis • Nombre de la especie: Canis Lupus • Nombre de la sub-especie: Canis Lupus Familiaris
Datos de prueba erróneos:	<ul style="list-style-type: none"> • Género: Canis • Nombre de la especie: Canis Lupu5 • Nombre de la sub-especie: C. Lupus Familiaris
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Taxonomía/Especie. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de especies el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de especies el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Se recomienda no escribir contracciones o abreviaciones. • Sub-especie y variedad son campos opcionales.

○ Ingreso de especie correcto:

Datos de Especie

Género:

Nombre de la especie:

Nombre de la sub-especie:

Nombre de la variedad:

○ Ingreso de especie erróneo:

Datos de Especie

Género:

Nombre de la especie:

Nombre de la sub-especie:

Nombre de la variedad:

● **Gestión de recolecciones:**

Caso de prueba:	Gestión de recolecciones.
Propósito:	Ingresar las recolecciones de los especímenes de animales
Prerrequisito:	El usuario debe el privilegio de agregar recolecciones.
Datos de prueba de recolección nacional:	<ul style="list-style-type: none"> • Tipo de recolección: Nacional • Fecha: 02/11/2012 • Latitud: 13.803240 • Longitud: -89.50678 • Observaciones: Recolección nacional de prueba • Recolector: Sandra Celina González • Temperatura: 30 • Altura: 1500 • Equipo de recolección: Redes y jaulas • Localidad: Cerca de un arroyo • Zona: La Libertad, Nuevo Cuscatlán
Datos de prueba de recolección internacional:	<ul style="list-style-type: none"> • Tipo de recolección: Internacional • Fecha: 03/11/2012 • Latitud: 15 • Longitud: -89 • Observaciones: Ejemplo recolección internacional • País: Honduras • Distrito: Tegucigalpa • Ciudad. Tegucigalpa • Donante: Museo nacional de Honduras
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Recolecciones. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de recolecciones el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de recolecciones el nuevo creado.
Observaciones:	Se recomienda respetar mayúsculas y minúsculas.

- Ingreso de recolección nacional:

Datos de Recolección

Tipo de recolección: Nacional Internacional

Fecha:

Latitud:

Longitud:

Observaciones:
Restan 970 caracteres.

→ **Siguiente**

Datos de Recolección

Recolector:

Temperatura(°C):

Altura (°mt.):

Recolectores auxiliares:
Restan 500 caracteres.

Equipo de Recolección:
Restan 988 caracteres.

Localidad:
Restan 482 caracteres.

Zona:

← **Anterior**

- Ingreso de recolección internacional:

Datos de Recolección

Tipo de recolección: Nacional Internacional

Fecha:

Latitud:

Longitud:

Observaciones:
Restan 987 caracteres.

→ **Siguiente**

Datos de Recolección

País:

Distrito:

Ciudad:

Donante:

← **Anterior**

- **Gestión de animales:**

Caso de prueba:	Gestión de animales.
Propósito:	Ingresar los animales que poseen una clasificación taxonómica.
Prerrequisito:	El usuario debe el privilegio de agregar animales.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Especie: Canis Lupus • Nombre científico: Canis lupus familiaris • Nombre común: Perro • Rango de distribución: Todo el globo • Grupo: Mamíferos • Referencias bibliográficas: http://www.google.com.sv
Datos de prueba incorrectos:	<ul style="list-style-type: none"> • Especie: Canis Lupus • Nombre científico: Canis lupus familiaris • Nombre común: Perr0 • Rango de distribución: Todo el globo
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Recolecciones. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de animales el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de animales el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • Datos con números o símbolos en los nombres del animal no serán ingresados

- Ingreso de animal correcto:

Datos de Animal

Especie:

Nombre Científico:

Nombre Común:

Rango Distribución:
Restan 987 caracteres.

Datos de Animal

Grupo:

Referencias Bibliograficas:
979 Caracteres Restantes.

- **Gestión de especímenes**

Caso de prueba:	Gestión de especímenes.
Propósito:	Ingresar los especímenes con los que cuenta la sección de zoología del MUHNES.
Prerrequisito:	El usuario debe el privilegio de agregar especímenes.
Datos de prueba correctos:	<ul style="list-style-type: none"> • Código del espécimen: 50-100.1 • Nombre científico: Canis lupus familiaris • Lugar y fecha de recolección: Canadá, Museo de Canadá, 01/01/2011 • Tipo de espécimen: Terrestre • Numero de muestras: 2 • Tipo de conservación: seco • Determino: Joseph Smith • Confirno: Joseph Smith • Ubicación en almacén: G-209 • Peso (gr.): 50 • Sexo: masculino • Madurez: adulto • Longitud (mt.): 1 • Diámetro (mt.): 0.30 • Altura (mt.) 0.30 • Textura: suave • Coloración: cafésosa • Descripción: Animal de prueba • Observación. Animal donado por el gobierno de Canadá • Estatus Local: Abundante • Donación: si • Estado de muestra: en buen estado • Hábitat: montaña • Tipo de sustrato: Arcilloso • Altitud (mt.): 2500 • Hospedero: no
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Recolecciones. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de especímenes el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de especímenes el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda evitar caracteres como punto, coma, etc. en campos que solicitan nombres específicos. • La fotografía puede ser cualquier formato, recomendable jpg.

- Ingreso de datos de especímenes:

Datos de Especimen

Código Especimen:	<input type="text" value="50-100.1"/>
Nombre Científico:	<input type="text" value="Canis Lupus Familiaris"/>
Fecha Recolección:	<input type="text" value="Canada, Museo de Canada, 01/01/2011"/>
Tipo de Especimen:	<input checked="" type="radio"/> Terrestre <input type="radio"/> Acuatico
Número de Muestras:	<input type="text" value="2"/>
Tipo Conservación:	<input type="text" value="Seco"/>
Determinó:	<input type="text" value="Joseph Smith"/>
Confirmó:	<input type="text" value="Joseph Smith"/>
Ubicación Almacen:	<input type="text" value="G-209"/>

Datos de Especimen

Peso(gr.):	<input type="text" value="50"/>
Sexo:	<input type="text" value="Masculino"/>
Madurez:	<input type="text" value="Adulto"/>
Longitud(mt.):	<input type="text" value="1"/>
Diametro(mt.):	<input type="text" value="0.30"/>
Altura(mt.):	<input type="text" value="0.30"/>
Textura:	<input type="text" value="Suave"/>
Coloración:	<input type="text" value="Cafesosa"/>

Datos de Espécimen

Fotografía:	<input type="button" value="+ Fotografía:"/> <input type="text"/>
Descripción: Restan 984 caracteres.	<input type="text" value="Animal de prueba"/>
Observación: Restan 961 caracteres.	<input type="text" value="Animal donado por el gobierno de Canadá"/>
Estatus Local:	<input type="text" value="Abundante"/>
Donación:	<input checked="" type="radio"/> Sí <input type="radio"/> No
Estado Muestra: Restan 986 caracteres.	<input type="text" value="En buen estado"/>
Habitát:	<input type="text" value="Montaña"/>
Tipo sustrato:	<input type="text" value="Arcilloso"/>

Datos de Espécimen

Altitud(mt.):	<input type="text" value="2500"/>
Hospedero:	<input type="text" value="No"/>

- **Gestión de restauraciones**

Caso de prueba:	Gestión de restauraciones.
Propósito:	Ingresar las restauraciones de las piezas de la colección de zoología.
Prerrequisito:	El usuario debe el privilegio de agregar restauración.
Datos de prueba de inicio:	<ul style="list-style-type: none"> • Espécimen: 50-100.1 • Autoriza: Ana Rivera • Realiza: Karen Castillo • Fecha de realización: 01/11/2012
Datos de prueba de modificación:	<ul style="list-style-type: none"> • Recibe: Ana Rivera • Fecha de finalización: 09/11/2012
Pasos:	<ol style="list-style-type: none"> 1. Ingresar a Restauraciones. 2. Ingresar los datos. 3. Hacer clic en el botón Guardar. 4. Seleccionar la fila agregada y dar clic en el botón modificar 5. Ingresar los datos. 6. Hacer clic en el botón Guardar.
Salida esperada:	Visualizar en la tabla de restauración el nuevo creado.
Salida Obtenida:	Visualizar en la tabla de restauración el nuevo creado.
Observaciones:	<ul style="list-style-type: none"> • Se recomienda respetar mayúsculas y minúsculas. • El campo Observaciones es opcional. • La fecha de finalización no puede ser menor que el día actual y la fecha de inicio.

- Ingreso de especie correcto:

Datos de Restauración

Espécimen:

Autoriza:

Realiza:

Fecha de Realización:

Observaciones:
Restan 1000 caracteres.

- Ingreso de especie erróneo:

Datos de Restauración

Espécimen:

Recibe:

Fecha de Finalización:

Observaciones:
Restan 1000 caracteres.

CAPÍTULO 6:

DESPLIEGUE

El flujo de trabajo de despliegue de la metodología RUP tiene como objetivo asegurar que el sistema informático funcione en un ambiente similar al final, es decir, antes de ser entregado al usuario. Debido a los alcances del proyecto, el sistema informático no será puesto en producción, pero se desarrollara el plan de implementación para que sea puesto en marcha.

6.1. PLAN DE IMPLEMENTACIÓN

El plan de implementación refleja cual será la estrategia que se debe seguir para la puesta en marcha del proyecto, cuyo objetivo es poner en operación o producción el sistema desarrollado, así como el uso del mismo de parte de los usuarios finales.

6.1.1. PLAN DE IMPLEMENTACIÓN DEL NUEVO SISTEMA

El plan de implementación del Sistema Informático para la Clasificación de la Zoología de El Salvador (SIZOO) tiene sus raíces en el Proceso Administrativo¹, el cual gráficamente es representado en la figura 6.1.



Figura 6.1. Proceso administrativo.

¹ <http://futuroadministrador.galeon.com/apo.htm>

Las etapas del proceso administrativo siguen un ciclo, en el que una etapa se desarrolla inmediatamente después de la anterior, tomando como punto de partida la planeación.

Una implementación exitosa no garantiza (por sí misma) el mejoramiento de la organización que use el nuevo sistema, pero una instalación inadecuada lo impedirá.

6.1.2. PLANEACIÓN

La planeación o planificación permite establecer actividades específicas y lineamientos a seguir por cada uno de los actores involucrados en el proyecto que se esté desarrollando, para este caso en particular, la implementación de un nuevo sistema.

Uno de los resultados más significativos del proceso de planificación es una estrategia para la organización.

En esta etapa se establece la dirección que ha de seguirse para implementar el nuevo sistema informático, presentando el detalle de las actividades que comprenden el Plan de Implementación del sistema a poner en marcha, siguiendo para ello una serie de fases, iniciando con la actividad de preparación del entorno, siguiendo con la organización del recurso requerido de la implementación, luego la ejecución del plan donde se incluirá el plan de capacitación; posteriormente se realizan actividades de control y la puesta en marcha del sistema.

Estas fases están distribuidas en un esquema en cascada, lo cual implica que cada fase debe completarse antes de pasar a la siguiente, permitiendo minimizar problemas por fases incompletas

En la figura 6.2 se presenta gráficamente la planeación de la implementación del sistema SIZOO:

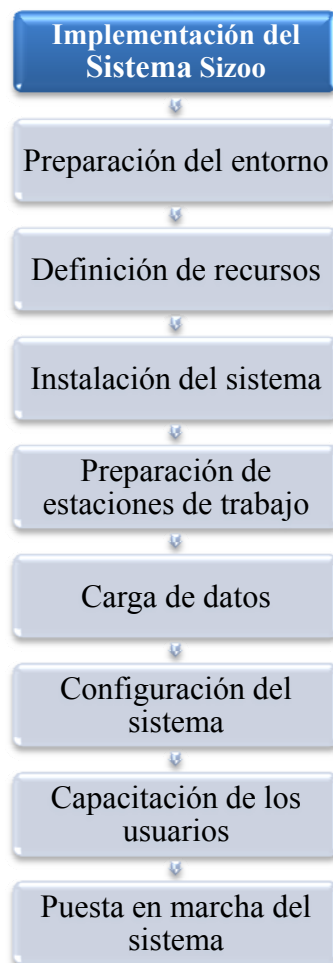


Figura 6.2. Etapas de la planeación de la implementación del sistema SIZOO.

- **Preparación del entorno:** La preparación del entorno implica la preparación de todos los elementos necesarios para iniciar el proceso de implementación del sistema SIZOO, para lo cual se deberán realizar una serie de actividades preparatorias e introductorias para que, al poner a funcionar el nuevo sistema, éste opere de manera óptima, cumpliendo los objetivos bajo el cual fue desarrollado.

En esta etapa se definen los responsables de ejecutar la implementación del sistema, así como las actividades correspondientes a cada uno de ellos.

La implementación y mantenimiento del sistema estará a cargo de la Gerencia de Informática y Sistemas de la Secretaría de Cultura de la Presidencia de la República.

La preparación para la implementación implica la presentación del Sistema Informático en la institución, con el fin de mostrar las capacidades y funcionalidad del mismo en respuesta a los requerimientos solicitados.

- **Definición de recursos:** Cuando se encuentran definidas las actividades necesarias para la implementación del nuevo sistema es importante definir los recursos destinados a cumplir con dichas actividades. Estos recursos se dividen en:

1. **Recurso humano:** Durante esta actividad debe gestionarse el recurso humano que participará en el proceso de implementación del sistema, entre los cuales están personal técnico con conocimientos de informática y los usuarios finales, los cuales conformaran el grupo de trabajo.

La estructura organizativa recomendada para la implementación del sistema SIZOO se muestra en la figura 6.3.

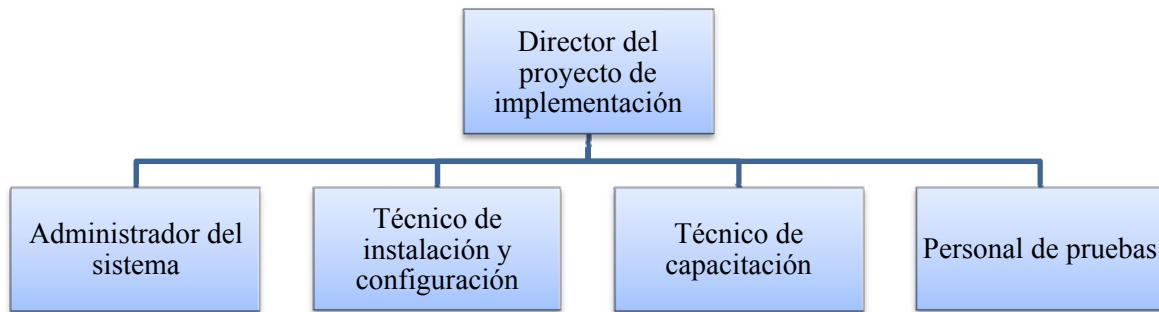


Figura 6.3. Estructura organizativa del proyecto de implementación.

El equipo de implementación posee diversas funciones, las cuales deben cumplirse adecuadamente para obtener los resultados deseados durante la ejecución del plan de implementación. Los perfiles que deben tener los miembros del equipo son:

- **Director del proyecto de implementación:**

Característica	Descripción
Nombre del perfil	Director del proyecto de implementación
Descripción del perfil	Responsable de supervisar el proceso de implementación y organizar los recursos que se necesitarán para el buen funcionamiento del nuevo sistema.
Responsabilidades	<ul style="list-style-type: none"> • Coordinar y controlar la ejecución de las diversas actividades del plan de implementación. • Garantizar que la instalación y funcionamiento de las aplicaciones del sistema se encuentren en forma óptima y correcta. • Supervisar la instalación y configuración del sistema SIZOO. • Recepción y supervisión de todo el equipo informático recibido para la implementación del sistema. • Realizar actividades de control que permitan evaluar los avances en la implementación. • Coordinar junto al técnico las capacitaciones de los usuarios finales. • Velar por que se realicen las respectivas pruebas y evaluaciones al sistema. • Definir medidas de contingencia que permitan solventar problemas que se presenten durante la ejecución del proyecto.
Requisitos	<ul style="list-style-type: none"> • Profesional graduado de cualquier carrera afín a la informática. • Experiencia comprobada en implementación de proyectos informáticos. • Capacidad deductiva y de análisis. • Experiencia en manejo de personal.

Tabla 6.1. Perfil del director del proyecto de implementación.

- **Administrador del sistema:**

Característica	Descripción
Nombre del perfil	Administrador del sistema.
Descripción del perfil	Encargado de la instalación y configuración del sistema SIZOO, así como de administrar la aplicación, la base de datos del sistema, y los recursos de hardware y software con los que cuenta la institución. Además, debe mantener en óptimo funcionamiento el sistema y brindar asistencia técnica cuando se solicite.
Responsabilidades	<ul style="list-style-type: none"> • Administrar la aplicación y la base de datos del sistema. • Realizar copias de respaldo de los datos de acuerdo a lo establecido por la institución. • Instalación y configuración del software necesario para el funcionamiento del sistema SIZOO. • Instalación del sistema SIZOO. • Configuración de la red de comunicación. • Encargado de verificar el funcionamiento de los equipos informáticos y redes de comunicación. • Encargado de verificar que los equipos informáticos cuenten con las herramientas y aplicaciones necesarias para que accedan al sistema. • Solventar los problemas de comunicación que existan entre el sistema con la base de datos. • Mantener en buen estado los equipos con el que funciona el sistema. • Brindar soporte a los usuarios en el uso del sistema.
Requisitos	<ul style="list-style-type: none"> • Conocimiento en sistemas operativos Windows y Linux en sus distribuciones de escritorio y servidores. • Conocimientos sólidos en sistemas gestores de base de datos PostgreSQL 9.1. • Conocimientos de programación y mantenimiento orientado al desarrollo Web bajo la plataforma Java. • Sólidos conocimientos en los frameworks PrimeFaces e Hibernate. • Conocimientos en configuración de servidor de aplicaciones Glassfish. • Conocimientos en instalación y configuración de redes. • Conocimientos para el mantenimiento preventivo, correctivo y predictivo de hardware, software y arquitectura de red.

Tabla 6.2. Perfil del administrador del sistema.

- **Técnico de instalación y configuración:**

Característica	Descripción
Nombre del perfil	Técnico de instalación y configuración.
Descripción del perfil	Encargado de la instalación y configuración del nuevo sistema informático y todos sus componentes en base a las especificaciones brindadas por el administrador del proyecto de implementación.
Responsabilidades	<ul style="list-style-type: none"> • Instalación y configuración del software en las estaciones de trabajo necesarios para el funcionamiento del sistema SIZOO. • Configuración del hardware y software que componen el nuevo sistema informático. • Verificar el funcionamiento de los equipos informáticos y redes de comunicación. • Realizar las correcciones pertinentes en caso de fallos en la implementación del nuevo sistema. • Participar en las pruebas de funcionamiento del nuevo sistema. • Evaluar el funcionamiento del nuevo sistema. • Brindar soporte a los usuarios en el uso del sistema.
Requisitos	<ul style="list-style-type: none"> • Egresado de carrera afin a la informática. • Experiencia en instalación de hardware y software. • Experiencia en instalación y configuración de redes. • Capacidad para la resolución de problemas. • Buenas relaciones interpersonales. • Proactivo.

Tabla 6.3. Perfil del técnico de instalación y configuración del sistema.

- **Técnico de capacitación:**

Característica	Descripción
Nombre del perfil	Técnico de capacitación.
Descripción del perfil	Responsable de asistir al director del proyecto en las diversas actividades establecidas para el desarrollo del plan de implementación, siendo la persona encargada de capacitar a los usuarios que harán uso del sistema tanto para pruebas como para el momento de implementación.
Responsabilidades	<ul style="list-style-type: none"> • Contribuir al cumplimiento exitoso de la implementación del nuevo sistema. • Coordinar junto al director del proyecto de implementación, las capacitaciones de los usuarios finales. • Impartir las capacitaciones definidas en el plan de implementación a los usuarios finales del sistema. • Realizar las pruebas y validaciones necesarias al sistema SIZOO.
Requisitos	<ul style="list-style-type: none"> • Egresado de carrera afín a la informática. • Experiencia en instalación de gestores de bases de datos y aplicaciones web. • Experiencia en el desarrollo de sistemas informáticos. • Conocimiento de técnicas de programación orientada a objetos. • Conocimientos en instalación y configuración de equipo informático. • Conocimientos de seguridad informática. • Trabajo en equipo. • Buenas relaciones interpersonales. • Facilidad de expresión. • Capacidad de enseñanza.

Tabla 6.4. Perfil del técnico de capacitación del sistema.

- **Personal de pruebas:**

Característica	Descripción
Nombre del perfil	Personal de pruebas.
Descripción del perfil	Usuarios finales del nuevo sistema informático quienes han expresado éste como necesidad a satisfacer.
Responsabilidades	<ul style="list-style-type: none"> • Expresar los resultados a los técnicos y administrador del proyecto de implementación. • Participar en las capacitaciones impartidas por el técnico de capacitación. • Participar en las actividades de pruebas realizadas al sistema informático.
Requisitos	<ul style="list-style-type: none"> • Reuniones con los miembros del grupo de trabajo. • Aportación de ideas que mejoren el funcionamiento del sistema. • Brindar recomendaciones para mejorar la implementación del sistema.

Tabla 6.5. Perfil del personal de pruebas del sistema.

Tanto la Secretaría de Cultura como el MUHNES cuentan con el personal anteriormente descrito.

2. Recursos de hardware y software: Las características mínimas de hardware y software con las que deben de contar el servidor y los equipos de los usuarios, son descritas en el manual técnico del sistema informático SIZOO.

- **Instalación del sistema:** Durante esta fase se hace efectiva la implementación del sistema SIZOO. Las actividades de instalación serán ejecutadas por el equipo de implementación y están definidas de la siguiente manera¹:

- 1. Instalación y configuración del software:** Durante esta se hace efectiva la instalación y configuración del equipo hardware, software y equipo de comunicación, que es requerido para la implementación del sistema SIZOO. Durante esta actividad están implícitas las siguientes tareas:
 - a. Configuración del servidor que alojará la aplicación y la base de datos.
 - b. Configuración de las máquinas de los usuarios.
 - c. Configuración de la red de comunicación.
 - d. Instalación y configuración del software necesario para que opere el sistema SIZOO tanto en las maquinas cliente como en el servidor.

¹ Para más detalles, consulte el Manual Técnico del Sistema SIZOO.

2. **Instalación y configuración del sistema:** Durante esta actividad se procede a instalar y configurar el sistema SIZOO, el cual involucra las siguientes tareas:
 - a. Creación de la base de datos.
 - b. Instalación de la aplicación.
 - c. Configuración de la aplicación.
 - d. Carga inicial de datos.

- **Preparación de estaciones de trabajo:** Los equipos desde los cuales se hará uso del sistema deben poseer un navegador web y tener acceso a la red de la institución, siendo estos verificados por el Técnico de instalación y configuración. Los recursos mínimos de hardware y software que deben poseer las estaciones de trabajos se muestran en el manual técnico del sistema SIZOO.

- **Carga de datos:** En esta actividad se hace el ingreso de los datos iniciales al sistema, los cuales son necesarios para su adecuado funcionamiento.

Estos datos serán ingresados por el administrador del sistema y abarcan aspectos de configuración inicial, con la información básica que, podrán ser cargados a través de un *script* de llenado con datos¹ para el funcionamiento inicial del sistema. Se recomienda introducir la información inicial en el siguiente orden:

A. Esquema de seguridad:

1. Catálogo de operaciones.
2. Catálogo de objetos.
3. Catálogo de privilegios.
4. Catálogo de roles.
5. Catálogo de usuarios.
6. Catálogo de sesiones.

¹ Para más detalles, consulte el Manual Técnico del Sistema SIZOO.

B. Esquema de zoología:

1. Catálogo de departamentos.
 2. Catálogo de municipios.
 3. Catálogo de áreas protegidas.
 4. Catálogo de phylum.
 5. Catálogo de clases.
 6. Catálogo de órdenes.
 7. Catálogo de superfamilias.
 8. Catálogo de familias.
 9. Catálogo de géneros.
 10. Catálogo de especies.
 11. Catálogo de animales.
 12. Catálogo de sinonimias.
 13. Catálogo de reseñas.
 14. Catálogo de recolectores.
- **Configuración del sistema:** Los pasos a seguir para la configuración inicial del sistema¹ son:
 1. **Configuración de la seguridad y parámetros del sistema:** La información que debe ingresarse al sistema corresponde a la configuración de la seguridad. Esta consiste en la creación de los perfiles y asignación de roles y privilegios a dichos perfiles.
 2. **Creación de cuentas de usuario en el sistema:** Al momento de realizar la configuración de funciones y perfiles del sistema de información, el administrador debe asignar permisos según el rol que posean los usuarios, para que pueda tener los privilegios de acceso al sistema según tipo de usuario. Esta cuenta deberá asociarse a un perfil de acuerdo a los permisos que el usuario tendrá dentro del sistema.
 3. **Ingreso de datos de los catálogos del sistema:** El administrador del sistema debe ingresar los datos requeridos para el llenado de catálogos del sistema.
 - **Capacitación de los usuarios:** La capacitación contempla la inducción que ha de realizarse, con el objetivo de dar a conocer a los usuarios, la manera más apropiada o correcta de utilizar el sistema informático.

¹ Para más detalles, consulte el Manual Técnico del Sistema SIZOO.

Para poner a funcionar el sistema es necesario programar un periodo de capacitación, dirigido a los usuarios finales, en el cual se pretende dar a conocer las diferentes opciones y funcionalidades del sistema.

El plan de capacitación¹ servirá para que los usuarios puedan familiarizarse con el sistema y hagan un uso más eficiente de éste.

- **Puesta en marcha del sistema:** Durante esta actividad se pone en funcionamiento el sistema una vez instalados y previamente configurados todos los componentes necesarios, teniendo capacitados a todos los usuarios finales y habiendo realizado las pruebas necesarias.

Se recomienda en esta etapa hacer una limpieza de datos de las tablas del sistema informático, los cuales fueron ingresados en la etapa de capacitaciones a los usuarios.

La puesta en marcha del sistema SIZOO, implica realizar la conversión del sistema actual al nuevo sistema, utilizando para ellos la conversión en paralelo².

Posteriormente a poner el sistema en funcionamiento, se lleva a cabo la actividad de control para detectar posibles fallos que se pueda presentar. En dicha actividad, se le da mantenimiento al sistema para que éste opere con normalidad y que los procesos se lleven a cabo de forma congruente. Esta actividad es llevada a cabo por el Administrador del Sistema.

6.1.3. ORGANIZACIÓN

Una vez definido y descrito las fases de la implementación, y dentro de ellas la estructura organizativa del equipo encargado de realizarla, deben asignarse estos para que desempeñen las funciones de cada puesto.

En esta sección se definen las responsabilidades de los miembros del equipo de trabajo y se calendarizan las activas a realizar para poder cumplir con el proyecto de implementación.

- **Matriz de responsabilidades:** De acuerdo a los perfiles definidos, cada persona tiene cierto grado de responsabilidad, en las diferentes actividades involucradas en el proyecto de implementación del nuevo sistema.

Para determinar este grado de responsabilidad se utilizara una matriz de responsabilidades en la cual se presentan las fases de implementación del proyecto con sus actividades más generales, donde para cada una de ellas, se asignan las funciones del proceso administrativo que competen a cada miembro del equipo de implementación. En la tabla 6.6 se muestra la matriz de responsabilidades.

¹ Ver sección 6.2. Plan de capacitaciones.

² Ver anexo D. Elección de método de conversión.

Nomenclatura P: planificar O: organizar D: dirigir C: controlar		Director del proyecto	Administrador del sistema	Técnico de instalación y configuración	Técnico de capacitación	Personal de pruebas
Lista de tareas						
Definición del equipo	Revisión y configuración del hardware y software	P, O, C	P, O, D, C			
Instalación	Instalación del nuevo sistema	P, O, C	P, O, D, C			
Carga de datos	Carga de catálogos	P, O, C	P, O, D, C			
Configuración	Configuración del sistema	P, O, C	P, O, D, C	D, C		
Capacitación	Inducción al nuevo sistema	P, O, C	C		D, C	O, D
Puesta en marcha	El sistema SIZOO en funcionamiento	P, O, C	C	D, C		

Tabla 6.6. Matriz de responsabilidades.

- **Programación y calendarización:** La programación de las actividades a realizar durante la implementación del sistema SIZOO, permitirá tener un mejor control de tiempos de cada actividad y el personal que será responsable de ejecutarlas.

Para ello se han establecido una serie de macro-actividades¹ que comprende la implementación, y sus respectivos tiempos de duración, los cuales se detallan en la tabla 6.7.

Actividad	Duración (días)
Preparación del entorno	4
Organización del personal	4
Ejecución de la implementación	4
Control de la implementación	4
Capacitación	9
Puesta en marcha	2
TOTAL	27

Tabla 6.7. Duración de macro-actividades.

¹ Macro-actividades definidas en el literal 6.1.2. Planeación.

- **Diagrama de Gantt:** Mediante la planificación realizada, se han definido diversas actividades que deben realizarse para llevar a cabo la implementación de la aplicación. Estas actividades deben tener una duración establecida, de manera que pueda darse un seguimiento y realizar un control sobre ellas. Se recomienda que estas actividades se realicen en el horario normal laboral, en días laborales.

En la tabla 6.8 se muestra los tiempos propuestos de las actividades del plan de implementación a realizar

Nombre de la tarea	Duración (días)
Plan de implementación del sistema SIZOO	27
Preparación del entorno	4
Revisión del hardware	1
Revisión del software	1
Revisión del equipo de comunicaciones	1
Preparación de las instalaciones físicas	1
Acondicionamiento de las instalaciones	2
Organización del personal	4
Gestionar el recurso humano	2
Organizar el personal	1
Revisión de la estructura organizativa	1
Verificación de los perfiles de puestos	1
Asignación de responsabilidades al personal	1
Ejecución de la implementación	4
Configuración de la red de comunicación	1
Instalación y configuración de software en el servidor	1
Instalación y configuración del software en los clientes	1
Instalación y configuración del sistema SIZOO	1
Carga inicial de datos	1
Control de la implementación	4
Determinación de tiempos de actividades	1
Definir actividades de control	1
Cumplimiento de las actividades	1
Elaboración de informes de avance de control	1
Capacitación	9
Planificación de la capacitación	1
Ejecución de la capacitación	8
Puesta en marcha	2
Limpieza de las tablas del sistema	1
Arranque del sistema SIZOO	1

Tabla 6.8. Duración de actividades.

En la figura 6.4 se presenta el diagrama de Gantt correspondiente a las actividades del proyecto de implementación del sistema SIZOO.

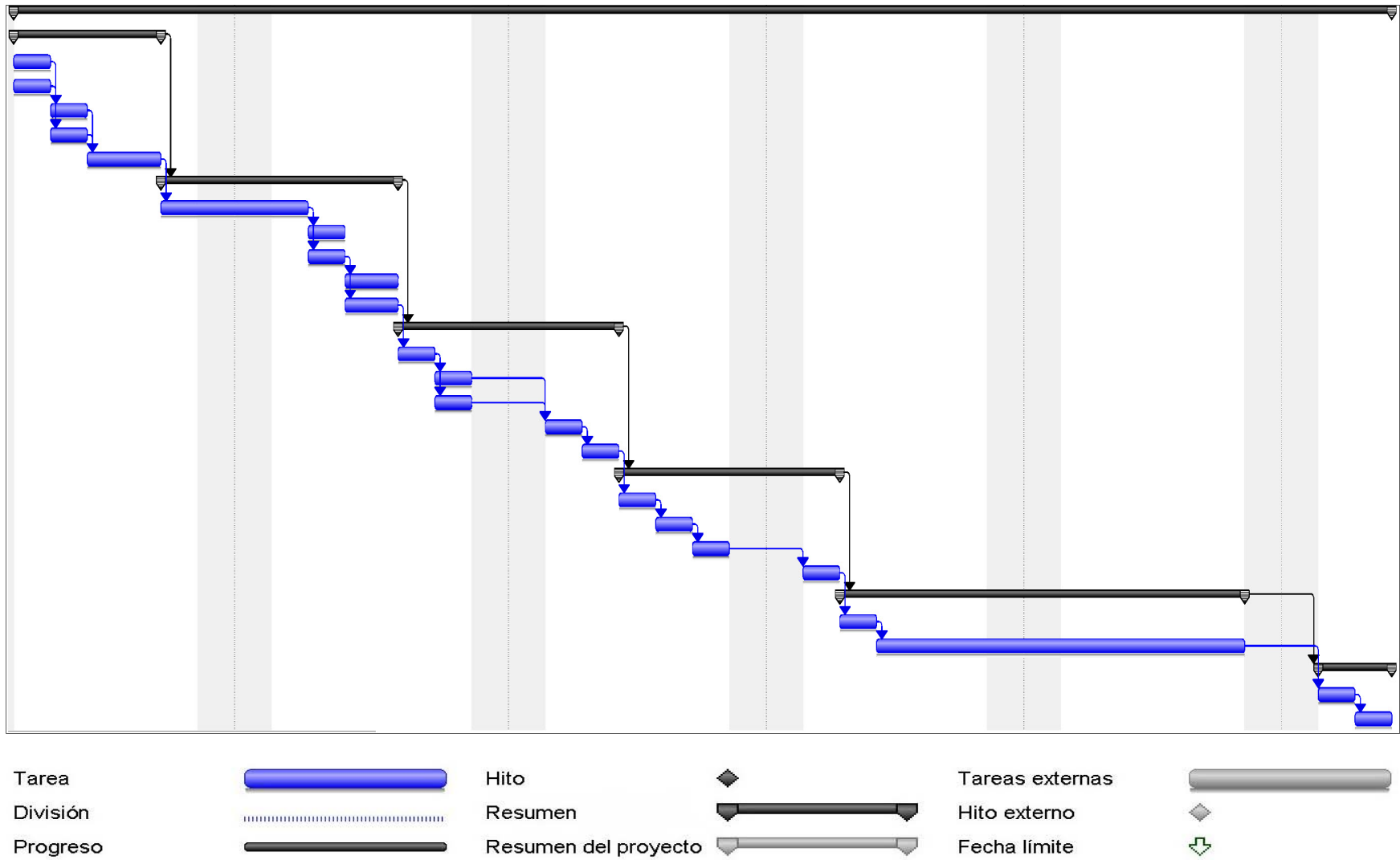


Figura 6.4. Diagrama de Gantt.

6.1.4. DIRECCIÓN

La dirección, o también llamada ejecución, es la actividad donde se pone en marcha el plan de implementación.

La responsabilidad de esta etapa recae sobre el director del proyecto de implementación, el cual está encargado de guiar el trabajo y tomar acciones que permita realizar en forma adecuada todas las actividades del plan de implementación.

- **Principios de la ejecución del plan de implementación:** Los principios que hay que tener presente durante la ejecución del plan de implementación son:
 - **Interés:** Coordinar los intereses del grupo y los intereses individuales de quienes participan en el logro del plan de implementación del nuevo sistema.
 - **Vía jerárquica:** Seguir los canales de comunicación establecidos en la organización del proyecto de implementación.
 - **Resolución de conflictos:** Los conflictos se deben de resolver de la forma más rápida posible, de modo que estos no entorpezcan el desarrollo normal de las actividades.

Aunque el director del proyecto de implementación es el responsable del cumplimiento del objetivo, todos los involucrados tienen el deber de cumplir a cabalidad sus tareas y responsabilidades, siendo el primero el encargado de inyectar la motivación y dinamismo que el grupo de trabajo requiera.

- **Costos de implementación:** Los costos asociados a la implementación del sistema SIZOO se han clasificado de la siguiente manera:
 - **Costos de recurso tecnológicos:** La Secretaría de Cultura de la Presidencia y el MUHNES cuentan con todo el recurso de hardware, software, de red de comunicación y conocimiento técnico para implementar el sistema SIZOO, no incurriendo en costos de este tipo.
 - **Costos de recurso humano:** La Secretaría de Cultura de la Presidencia cuenta con la Gerencia de Informática y Sistemas, que está conformada con personal técnico calificado, el cual puede asumir sin ninguna objeción, salvo el director del proyecto de implementación así lo requiera, los puestos definidos en la organización del proyecto. La implementación del sistema SIZOO se llevará a cabo en el horario normal laboral como un proyecto informático habitual.
 - **Costos de capacitación:** Los costos de capacitación del personal han sido en \$ 628.00¹.

Se recomienda que el grupo de trabajo haga las diligencias necesarias para contar con un presupuesto idóneo, que ayude a prevenir cualquier costo adicional en el que se incurra.

¹ Ver sección 6.2. Plan de capacitaciones.

6.1.5. CONTROL

El objetivo principal en esta actividad es medir el avance de cada una de las actividades que conforman el proceso de implementación del sistema SIZOO, comprobando que se está siguiendo el plan y que se están cumpliendo los objetivos de planificación.

Para ello es necesario monitorear cada una de las actividades para establecer puntos de control en los cuales se puedan detectar posibles desviaciones con la finalidad de realizar acciones que minimicen el impacto que pudiera ocasionar el retraso de una o más actividades.

Se deben incluir listas de tareas, reuniones e informes sobre el estado y avance en cada una de las tareas a ejecutar durante la implementación. A medida se vaya avanzando, se deben cumplir los puntos de control establecidos y con ello comprobar el avance de cada tarea del proyecto de implementación.

El Director del proyecto de implementación deberá llevar la continuidad del proyecto paso a paso, dirigiendo, organizando y controlando los tiempos necesarios a utilizar para el desarrollo de cada una de las etapas, este se hará responsable de los retrasos durante la implementación del sistema.

- **Establecimiento de puntos de control y monitoreo:** Consiste en establecer actividades de monitoreo sobre cada actividad del proceso de implementación, para evitar posibles desviaciones durante su ejecución.

Así mismo se deben incluir evaluaciones, las cuales pueden ejecutarse durante cada tarea de implementación y al finalizar las mismas. Dichas evaluaciones ayudaran a medir el logro de resultados y compararlos con lo planeado.

En caso de existir desviaciones que afecten el curso normal de una actividad, el director del proyecto en conjunto con el personal encargado de dicha actividad, serán los responsables de establecer medidas correctivas y coordinar la aplicación de éstas para garantizar el buen curso del proyecto de implementación.

También es necesario que se realicen reuniones cada cierto tiempo, en las cuales participen todos los integrantes del equipo de implementación, con el objetivo de medir los avances obtenidos a través de la ejecución de las diferentes actividades que constituyen el plan de implementación.

- **Estrategias de monitoreo y control:** Entre las estrategias de monitoreo y control del proyecto de implementación del sistema SIZOO tenemos:

- **Indicadores de gestión**¹: Los indicadores de gestión permiten la medición de lo planificado contra lo ejecutado hasta la fecha., siendo estos:

- **Indicador de duración de actividades (IDA)**: Permite conocer el grado de desviación entre el tiempo real de una actividad y el tiempo que se tenía programado para ésta. Esta se calcula de la siguiente manera:

$$IDA = \frac{\textit{Duración real de la actividad}}{\textit{Tiempo programado para la actividad}}$$

Dónde:

- ❖ $IDA \leq 1$, es aceptable. Indica que el tiempo invertido es acorde al tiempo planificado.
- ❖ $IDA \geq 1$, es inaceptable. Indica que deben tomarse medidas correctivas.

Entre las medidas correctivas tenemos:

- ❖ Re-planificación de actividades
 - ❖ Reducir el tiempo de ejecución de las tareas restantes.
 - ❖ Asignar más recursos a las actividades.
- **Indicador de actividades planificadas ejecutadas (IAPE)**: Permite conocer el grado de avance del proyecto de implementación. Se calcula de la siguiente manera:

$$IAPE = \frac{\sum \textit{Duración de las actividades ejecutadas}}{\sum \textit{Duración de todas las actividades planificadas}}$$

Donde el estándar de comparación es 100%. Mientras más cercano sea el indicador a este valor, mayor es el grado de avance en la ejecución de las actividades.

- **Formularios de control**: Para el cumplimiento de actividades del plan de implementación se llevará el seguimiento en cada una de las actividades del plan, el director del proyecto se auxiliará de informes de avances, los cuales contendrán información acerca de los logros obtenidos durante la ejecución de dichas tareas y que servirán para darle seguimiento a dicha ejecución. Estos serán presentados por los encargados de las diferentes tareas, y serán revisados y aprobados por el director del proyecto de implementación. Dichos informes serán representados por medio de formularios de control que permitirán controlar el avance de cada actividad realizada y aplicar medidas correctivas cuando fuese necesario.

El formulario propuesto se presenta en la figura 6.5.

¹ Medición del desempeño. Gerencia Informática. Carlos Ernesto García. Sexta Edición. Pág. 64.



Presidencia de la República de El Salvador

Implementación del Sistema Informático SIZOO Formulario de control de los tiempos de Actividades



Elaborado Por: A		Firma: B	
Fecha inicial: C		Fecha Final: D	
No.	Actividad	Tiempo programado	Tiempo real
E	F	G	H
Observaciones: I			
Página <i>n</i> de <i>m</i>			

Figura 6.5. Formulario de control de tiempo de las actividades.

Los elementos que componen el formulario de control de tiempo de actividades son:

- A. Nombre de la persona que lleno el formulario.
- B. Firma de la persona que lleno el formulario.
- C. Fecha de inicio del control de las actividades.
- D. Fecha de finalización del control de las actividades.
- E. Numero correlativo de la actividad.
- F. Nombre de la actividad.
- G. Tiempo estimado para realizar una actividad.
- H. Tiempo real que tomo realizar una actividad.
- I. Observaciones en caso de ser necesario.

6.2. PLAN DE CAPACITACIONES

6.2.1. PROPÓSITO

Especificar la manera en que deberá realizarse la capacitación de los usuarios en el uso del Sistema Informático para la Clasificación de la Zoología de El Salvador (SIZOO), de forma que se haga la inducción necesaria para que se haga un uso adecuado del sistema, y se obtengan los resultados que se esperan del mismo.

6.2.2. ALCANCE

La capacitación cubrirá al personal técnico de la Secretaría de Cultura y personal de dirección y operativo del Museo de Historia Natural de El Salvador (MUHNES).

Por medio de la capacitación se brindará la inducción necesaria para que el personal pueda utilizar correctamente las diferentes funciones que el sistema SIZOO proporciona.

Dicha capacitación será impartida por el personal de la Secretaría de Cultura de la República.

6.2.3. FUNCIONES QUE REQUIEREN CAPACITACIÓN

La implementación del sistema SIZOO requiere que el personal conozca la forma de utilizarlo correctamente para obtener los resultados esperados, en función de los siguientes aspectos:

- **Administración:** Esta función es parte importante del administrador del sistema, ya que debe conocer las diferentes opciones disponibles en el sistema SIZOO y la forma en que debe utilizarse.
- **Planificación:** Función que involucra al director del proyecto y personal técnico, ya que éstos deben formular la planificación con desempeño.
- **Desempeño:** Competencia del técnico de Capacitación, ya que es el encargado de los procesos de evaluación de desempeño.

6.2.4. PLANIFICACIÓN DE LA CAPACITACIÓN

Las actividades iniciales al periodo de capacitación, son las que ayudan a tener una mejor organización y ejecución para dicha fase, las cuales se mencionan a continuación:

1. **Elaboración previa de las actividades a desarrollar para la capacitación:** Se debe preparar con anticipación, cada tema a tratar dentro del contenido temático programado, de forma que cada tema esté lo más detallado y explicado posible, para evitar ambigüedades y confusiones en los usuarios a la hora de impartir la capacitación. Para ello se recomienda a las partes asistirse del Manual de Usuario y Manual Técnico del sistema SIZOO.
2. **Preparación de un escenario real para el desarrollo de la capacitación:** Se debe gestionar la utilización de las instalaciones a utilizar durante el periodo de capacitación; estas deben ser preparadas y acondicionadas con el mobiliario y equipo tecnológico necesario, de forma tal, que sean cómodas tanto para el usuario como para el que imparte la capacitación.
3. **Preparación del material necesario durante esta fase:** Es necesario realizar las gestiones pertinentes para la obtención del recurso necesario para el desarrollo de las capacitaciones, tales como: papelería para la reproducción del manual de usuario, plan de implementación; entre otros.
4. **Carga de datos:** Antes de impartir las capacitaciones a los usuarios, se debe realizar la carga inicial de datos al sistema, llenado de catálogos y creación de cuentas de usuario. Esto garantizará la correcta operatividad del sistema.

6.2.5. METODOLOGÍA DE CAPACITACIÓN

Las capacitaciones se llevarán a cabo en las instalaciones de la Secretaría de Cultura o del MUHNES, previo acuerdo de las partes, en jornadas de 4 horas en el transcurso del día laboral, que serán impartidas por el técnico de capacitación. Se recomienda que las instalaciones cuenten con:

- Espacio suficiente para albergar a aproximadamente 10 personas.
- Contar con aire acondicionado, dispensador de agua, baños accesibles, pantalla para proyecciones, y cualquier otro insumo que se considere necesario para que los asistentes se sientan cómodos.
- Acceso a internet, para mostrar las opciones del sistema que utilizan datos geo-referenciados.

Luego de transcurridas 2 horas de una temática, se recomienda dar un descanso a los asistentes de 15 minutos para esparcimiento de los asistentes.

Los técnicos de instalación y configuración deberán verificar, previo a las capacitaciones, que se encuentre listo el equipo que se utilizará. Este proceso no debe tomar más de un día.

Las capacitaciones se impartirán por grupos específicos de usuarios, de acuerdo a las características del perfil que les ha sido asignado dependiendo su rol.

En las capacitaciones se utilizará escenarios reales, de manera que los colaboradores se familiaricen a las funciones que utilizarán en la aplicación. Por tal razón, el técnico de capacitación preparará los diferentes escenarios que se considerarán indispensables en las capacitaciones, las cuales serán impartidas y desarrolladas con el apoyo del Manual de Usuario del sistema SIZOO.

Las capacitaciones serán expositivas y demostrativas, explicando cada una de las funciones del sistema. Esto se hará utilizando una computadora en la cual el sistema estará instalado previamente. La ejecución y demostración del sistema se hará utilizando el entorno de desarrollo Netbeans y el servidor local Glassfish. Esta debe cumplir los siguientes puntos:

- Exponer la forma correcta de utilizar el sistema SIZOO.
- Facilitar la comprensión de las funcionalidades del sistema SIZOO.
- Incentivar la utilización del sistema SIZOO.
- Disipar dudas y contestar preguntas del personal.

6.2.6. RECURSOS

Los recursos que se necesitan para impartir la capacitación son:

- **Recursos informáticos:** Los recursos informáticos necesarios son:
 - Computadora que cumpla los requerimientos mínimos de instalación del sistema SIZOO, tanto en hardware como en software.
 - El sistema SIZOO debe estar instalado en la computadora a utilizar en la capacitación. Es necesario que la computadora tenga instalado Netbeans 7.1. y el servidor de aplicaciones Glassfish 3.1.2.
 - Proyector.
 - Impresor a color.
- **Recurso humano:** Los perfiles del recurso humano con que cuentan la Secretaría de Cultura y el MUHNES pueden observarse en las tablas 6.1 a 6.4 de la sección anterior.

6.2.7. EJECUCIÓN DE LA CAPACITACIÓN DEL PERSONAL

Se debe hacer un listado de todo el personal que participará en las capacitaciones, siendo estos:

- **D:** Director del proyecto de Implementación.
- **A:** Administrador del Sistema.
- **C:** Técnico de Capacitación.
- **T:** Técnicos de instalación y configuración.
- **P:** Personal de pruebas (usuarios finales).

La temática a desarrollar se muestra en la tabla 6.9.

Jornada	Temática	Objetivos didácticos	Contenido	Personal	Duración (horas)
1	Configuración del sistema	Que los participantes conozcan en detalle los requerimientos y componentes internos que conforman el sistema a implementar, así como su instalación y configuraciones.	Características generales del sistema. Requerimientos de hardware y software. Estructura de la base de datos. Creación de la base de datos. Arquitectura del sistema. Instalación y configuración del sistema.	D, A, T	4
2	Administración del sistema	Que los participantes conozcan en detalle la gestión de la información del sistema dentro de los servidores de aplicación y base de datos.	Administración de usuarios, roles y privilegios. Gestión de datos del sistema. Auditoría del sistema.	D, A, T	4
3	Administración de la aplicación	Que los participantes sean capaces de administrar los usuarios que interactuaran con el sistema.	Acceso al sistema. Gestión de usuarios, roles y privilegios. Auditoría interna del sistema.	T, P	4
4	Gestión de geografía	Que los participantes sea capaces de administrar la información geográfica de El Salvador.	Gestión de departamentos. Gestión de municipios. Gestión de áreas protegidas.	T, P	4
5	Gestión de taxonomía	Que los participantes sean capaces de gestionar la información taxonómica de los animales.	Gestión de phylum. Gestión de clases y órdenes. Gestión de superfamilias y familias. Gestión de géneros y especies.	T, P	4
6	Gestión de recolecciones	Que los participantes sean capaces de gestionar la información de las recolecciones de especímenes y los recolectores	Gestión de recolectores. Gestión de recolecciones nacionales. Gestión de recolecciones internacionales.	T, P	4
7	Gestión de especímenes	Que los participantes sean capaces de gestionar la información de los animales, los especímenes de la colección y restauraciones de las muestras.	Gestión de animales. Gestión de especímenes. Gestión de restauraciones.	T, P	4
8	Consultas y reportes	Que los participantes sean capaces de realizar consultas y generar reportes con el sistema informático.	Consultas personalizadas. Consulta investigativa. Generación de reportes.	T, P	4
				Total	32

Tabla 6.9. Temática de la capacitación.

6.2.8. PRESUPUESTO DE CAPACITACIÓN

Para determinar el costo de la capacitación se considerarán los siguientes rubros:

1. **Recurso humano:** El recurso humano que intervendrá en las capacitaciones es personal que labora para la Secretaría de Cultura de la Presidencia, por lo que la participación del personal en esta actividad se encuentra remunerada, añadiendo también que es parte de sus funciones. Los costos de recurso humano pueden variar si el Director del Proyecto de Implementación del sistema SIZOO considera la necesidad de contratar a personal adicional que lo apoye en el cumplimiento del plan. El tiempo que el técnico de capacitación participará en la capacitación es de 32 horas aproximadamente. Estimando un salario normal de \$800.00, a \$ 5.00 la hora laboral, el costo de este recurso es:

$$5.00 \times 32 = \$ 160.00$$

2. **Material de apoyo:** Los costos del material de apoyo incluye la impresión de la documentación que acompaña al sistema SIZOO. Estimando un costo de impresión por hoja a color es de \$ 0.10, los costos del material de apoyo se muestran en la tabla 6.10.

Documento	Páginas	Costo unitario (\$)	Cantidad estimada	Total (\$)
Plan de implementación	40	4.00	2	8.00
Manual técnico	150	15.00	6	36.00
Manual de usuario	50	5.00	5	25.00
Total:				69.00

Tabla 6.10. Costos de documentación para la capacitación del sistema SIZOO.

3. **Otros costos:** Se puede incurrir en otros gastos, como papelería adicional, fotografías, acetatos, alimentación, entre otros, quedando estos a discreción de las partes involucradas en la coordinación de esta actividad. Estimando que cada hora de la capacitación genere un gasto aproximado de \$ 10.00, en 32 horas tenemos:

$$10.00 \times 32 = \$ 320.00$$

Al sumar todos los costos calculados de la capacitación tenemos:

$$\$ 160.00 + \$ 69.00 + \$ 320.00 = \$ 628.00$$

7. CONCLUSIONES

Luego del análisis de los procesos actuales del área de zoología del MUHNES, se identificaron las deficiencias provocadas al realizar los procesos de forma manual. Estos datos fueron analizados y se presentó una propuesta de cómo tendría que ser los procesos, incluyendo el personal involucrado.

Partiendo de este punto, se tomaron los requerimientos o necesidades que tienen que ser cubiertas por el nuevo sistema, siendo los más importantes la gestión de usuarios, gestión de taxonomías, animales, especímenes, recolectores, recolecciones, restauraciones de piezas y zonas geográficas.

La metodología usada en el desarrollo del sistema fue el análisis orientado a objetos, ya que provee los fundamentos para modelar la solución al problema de manera más sencilla y permitiendo agilizar el desarrollo en base a sus principios. Como ciclo de desarrollo del sistema, se eligió RUP dado que tiene un alto nivel de acoplamiento con el análisis orientado a objetos y sobre todo da flexibilidad y mayor participación del usuario. Añadido a esto, se usaron los patrones de diseño MVC y DAO, los cuales permiten, los cuales facilitaron la codificación del sistema, y que facilitara el mantenimiento y migración del sistema si las partes así lo acuerdan.

La base de datos de un sistema representa el dominio del mismo, y partiendo de que esté bien definido, se procedió a construir el resto de las piezas del sistema. Entre estas piezas tenemos las interfaces de usuario, que son el punto de contacto del usuario con el sistema. Un buen diseño que asegure facilidad de uso y eficiencia en las actividades garantiza la aprobación del usuario.

La calidad del sistema informático fue evaluada con el plan de pruebas, el cual definió la conclusión del desarrollo del sistema informático cuando fue superado y validado por los usuarios y por tanto listo para implementar. Es por esto que el plan de pruebas debe estar bien diseñado y fundamentado en estándares, que son los que definirán la calidad del sistema.

Cabe destacar que al implantarse el sistema informático la institución tendrá beneficios que se podrán ver reflejados en el manejo de la información, logrando así la preservación de la información, mejora en la calidad de atención prestada a los investigadores y facilidad en las búsquedas de información y generación de reportes.

8. RECOMENDACIONES

Para que el proyecto sobre el desarrollo de este y futuros sistemas informáticos funcionen de la mejor manera y se lleve a cabo con éxito, se hacen las siguientes recomendaciones:

- Para futuros proyectos, crear una relación de trabajo entre los interesados en el desarrollo del proyecto, sobre todo cuando abarca más de una unidad administrativa, y el equipo de trabajo para que el sistema que sea desarrollado posea la total aceptación para ser utilizado.
- Se ha constatado que no se cuenta con estándares bien definidos en la organización que pueden ser utilizados por el equipo de trabajo. Es recomendable contar con estos estándares para facilitar el mantenimiento de los futuros sistemas a implementar.
- Aunque los recursos tecnológicos hacen posible que el proyecto sea factible operativamente, se recomienda evaluar la mejora de estos recursos en un tiempo máximo de 2 años, debido a que existe la posibilidad de implementar nuevos proyectos en el servidor, además de los proyectos que ya han sido implementados y que consumen recursos de este servidor. Este tiempo lo recomienda Gordon E. Moore¹, cofundador de la empresa Intel.
- Debido a que todos los procesos son realizados por una sola persona, para que se vea reflejada la mejora en el manejo de información luego de haber sido implementado el sistema informático, se sugiere la contratación de más personal que se encargue de la digitación de todos los registros que ya se poseen hasta la fecha.
- La colaboración de la Secretaría de Cultura ha sido importante y efectiva, a pesar de esto, se sugiere mayor compromiso con futuros proyectos con el objetivo de obtener información oportunamente y así evitar retrasos y dificultades en el desarrollo normal y consecución de los proyectos informáticos.

¹ http://es.wikipedia.org/wiki/Ley_de_Moore

9. BIBLIOGRAFÍA

LIBROS

1. Pressman, Roger S.; Ingeniería del Software Un enfoque Practico; Mc Graw Hill, 6ª Edición, España, 2005.
2. García, Carlos Ernesto; Gerencia Informática; Informatik, 6ª. Edición, El Salvador, 2009.
3. Elmasri, Ramez; Fundamentos de Sistemas de Bases de Datos; Pearson, 5ª. Edición, España, 2007.
4. Kendall & Kendall; Análisis y diseño de sistemas; Pearson Educación, 6ª edición, México, 2005.

PÁGINAS WEB

1. Rojas Vera, M.E.; "Ciclos de vida – modelo en V"; (documento web); 2009.
< <http://spanishpmo.com/index.php/ciclos-de-vida-modelo-en-v/>>; 29/03/2012.
2. Álvarez, Itzcoalt; “Desarrollo Ágil con SCRUM”; (documento pdf); 2007.
<<http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:sg07.p02.scrum.pdf>>; 29/03/2012.
3. Solis, Camilo; “Metodologías de Desarrollo de SW”; (documento doc); 2007.
<<http://adonisnet.files.wordpress.com/2008/06/articulo-metodologia-de-sw-formato.doc>>;
29/03/2012.
4. Riveros, Brenda; “Programación estructurada vs Programación Orientada a Objetos”; (documento web); 2011.
<<http://mistock.lcompras.biz/paradigmas23/784-programacion-estructurada-vs-programacion-orientada-a-objetos>>; 29/03/2012.

5. Guevara, J; “Enfoque práctico basado en modelamiento UML”; (documento pdf); 2008.
< <http://es.scribd.com/doc/42190781/RUP-UML>>; 10/04/2012.
6. Bizagi; “BPMN”; (documento web); 2012;
<<http://wiki.bizagi.com/es/index.php?title=BPMN>>; 15/05/2012.
7. Medina, Eduardo; “Ingeniería del software I”; (documento web); 2010;
<<http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/>>; 28/05/2012.
8. López, Grace; “Pruebas de software”; (documento ppt); 2011;
< <http://www.slideshare.net/cliceduca/pruebas-de-software-2420588/>>; 28/05/2012.
9. Oracle; “JavaCode Conventions”; (documento pdf), 1997;
<<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>>; 10/06/2012.
10. ItesCam; “Implementación de sistemas”; (documento PDF); 2010
<<http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r79209.PDF>>; 18/10/2012
11. Wikipedia; “ISO/IEC 9126”; (documento web); 2010
< http://es.wikipedia.org/wiki/ISO/IEC_9126>; 05/09/2012
12. Clavijo, Paulo; “Presentación JUnit4”; (documento pdf); 2012
<<http://ebookbrowse.com/presentacion-junit4-pauloclavijo-mayo-2012-pdf-d350009540>>; 09/11/2012

10. GLOSARIO

- **B**

- **Base de Datos:** Es un sistema que almacena datos que están relacionados entre sí. Es un repositorio en donde guardamos información integrada que podemos almacenar y recuperar.
- **Biología:** es una ciencia porque se basa en la observación de la naturaleza y la experimentación para explicar los fenómenos relacionados con la vida.
- **Biólogo:** es un científico dedicado a producir resultados en el área de la biología a través del estudio de los organismos. Generalmente los biólogos estudian los organismos y su relación con el entorno que los rodea. Los biólogos del museo además de estudiar el ambiente de las especies, también estudian su fisiología, la cual varía según el clima y ubicación de donde se obtuvo la muestra.
- **Bit:** es el acrónimo Binary digit. (dígito binario). Un bit es un dígito del sistema de numeración binario. Mientras que en el sistema de numeración decimal se usan diez dígitos, en el binario se usan sólo dos dígitos, el 0 y el 1. Un bit o dígito binario puede representar uno de esos dos valores, 0 ó 1, y es considerado la menor forma de representación de los datos en informática.
- **Botánica:** es una rama de la biología y es la ciencia que se ocupa del estudio de los vegetales, bajo todos sus aspectos, lo cual incluye su descripción, clasificación, distribución, identificación y el estudio de su reproducción, fisiología, morfología, relaciones recíprocas, relaciones con los otros seres vivos y efectos provocados sobre el medio en el que se encuentran.
- **Botton-up:** es una estrategia de desarrollo que toma las partes individuales del diseñan con detalle y luego se enlazan para formar componentes más grandes, que a su vez se enlazan hasta que se forma el sistema completo.
- **bps:** los bits por segundo como unidad del Sistema Internacional de Unidades son utilizados para expresar la velocidad de transmisión de datos. Con frecuencia se usa en forma ambigua como bps, que para el Sistema Internacional significaría "bits por segundos". Por lo general se mide en Kbps (kilobits por segundo) o Mbps (megabits por segundo).
- **Byte:** es una secuencia de bits contiguos, cuyo tamaño depende del código de información o código de caracteres en que sea definido. Se usa comúnmente como unidad básica de almacenamiento de datos en combinación con los prefijos de cantidad, por ejemplo KiloByte (1024 bytes), MegaByte (2^{20} bytes) o GigaByte (2^{30} bytes).

- **C**

- **Caja negra:** en teoría de sistemas y física, se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. En otras palabras, de una caja negra nos interesará su forma de interactuar con el medio que le rodea entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace.

- **Casos de uso:** un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores. En el contexto de ingeniería del software, 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/u otros sistemas.
 - **Clase:** en biología, la clase es una categoría taxonómica situada entre el filo o división y el orden.
 - **Cerambícidos:** o escarabajos longicornios (Cerambycidae) son una familia de coleópteros polífagos provistos de llamativas antenas, casi siempre más largas que el cuerpo.
 - **Ciclo de vida del desarrollo de software:** es una estructura aplicada al desarrollo de un producto de software. Hay varios modelos a seguir para el establecimiento de un proceso para el desarrollo de software, cada uno de los cuales describe un enfoque diferente para diferentes actividades que tienen lugar durante el proceso.
- **D**
 - **DAO:** En software de computadores, un Data Access Object (DAO, Objeto de Acceso a Datos) es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo. El término se aplica frecuentemente al Patrón de diseño Object.
 - **Desarrollo en cascada:** es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior.
 - **Desarrollo en espiral:** Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a ninguna prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior.
 - **Diagrama de clases:** es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.
 - **Diagrama de interacción:** muestran las interacciones entre objetos ocurridas en un escenario del sistema.
 - **Diagrama entidad-relación:** es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

- **E**

- **Enfoque de Sistemas:** es un esquema metodológico que sirve como guía para la solución de problemas, en especial hacia aquellos que surgen en la dirección o administración de un sistema, al existir una discrepancia entre lo que se tiene y lo que se desea, su problemática, sus componentes y su solución.
- **Entomología:** es el estudio científico de los insectos. De cerca de las 1.3 millones de especies descritas, los insectos constituyen más de los dos tercios de todos los seres vivos conocidos y, además, tienen una larga historia fósil, ya que su aparición se remonta al Devónico, hace unos 400 millones de años. Tienen muchas formas de interacción con los humanos y con otras formas de vida en la Tierra; es así que la entomología se constituye una especialidad importante dentro de la zoología. La entomología incluye, con frecuencia, el estudio de otros artrópodos, como arácnidos, crustáceos y miriápodos, aunque esta extensión sea técnicamente incorrecta.
- **Especie:** en taxonomía se denomina especie (del latín species), o más exactamente especie biológica, a cada uno de los grupos en que se dividen los géneros. Una especie es la unidad básica de la clasificación biológica. Para su denominación se utiliza la nomenclatura binomial, es decir, cada especie queda inequívocamente definida con dos palabras, por ejemplo, Homo sapiens, la especie humana.

- **F**

- **Factibilidades:** el estudio de factibilidades es realizado por lo regular por una o dos personas que tiene conocimiento en técnicas de sistemas (analistas de sistemas), cuyo objetivo es indicar si un proyecto es viable o no lo es.
- **Factibilidad económica:** para iniciar con el desarrollo de esta factibilidad es necesario haber establecido la factibilidad técnica, esta comprende los beneficios que traerá la realización del proyecto. Se deben de hacer una serie de cuestionamientos para poder saber si es factible el desarrollo del sistema económicamente “¿los beneficios que se obtienen serán suficientes para aceptar los costos?, ¿los costos asociados con la decisión de no crear el sistema son tan grandes que se debe aceptar el proyecto?” sin duda este aspecto es el más importante en las empresas ya que los gerentes muchas veces no están dispuestos a solventar estos costos cuando no hay los suficientes fundamentos que los convenzan de que es necesario la realización del proyecto por los beneficios que se obtendrán.
- **Factibilidad operativa:** comprende la utilidad del sistema una vez se haya desarrollado e implantado en la empresa. ¿Será utilizado el sistema?, ¿existirá cierta resistencia al cambio por parte de los usuarios que dé como resultado una disminución de los posibles benéficos del proyecto?

- **Factibilidad técnica:** comprende los recursos técnicos necesarios para el desarrollo y puesta en marcha del proyecto (hardware, software, recurso humano). Estos recursos establecen si un proyecto es técnicamente factible de realizar ya que no todos los proyectos solicitados lo son. Sin embargo aquellos proyectos que son factibles deben ser incorporados en los planes de desarrollo de sistemas. Con una solicitud de aprobación de proyecto el cual puede comenzar inmediatamente a postergarse en otro momento.
 - **Fauna:** se refiere al conjunto de animales en sus diferentes clasificaciones, como mamíferos, reptiles, aves, etc., Para el conocimiento de la fauna, se parte del conocimiento taxonómico y de la distribución de las especies en los tres ambientes de vida terrestre, aguas continentales y aéreas.
 - **Fauna silvestre:** La Fauna Silvestre es un término que se refiere a los animales que normalmente no están domesticados, en otras palabras no criados por el hombre. Ellos son un recurso vivo que muere y es reemplazado por otro de su especie. Un animal, individualmente, no puede vivir más allá de su período normal de vida. Pero, si son manejados cuidadosamente, las poblaciones de fauna silvestre pueden ser conservadas prácticamente para siempre.
 - **Fisiografía:** descripción de las características físicas de a Tierra y de los fenómenos de la naturaleza que en ella se originan, en particular de las características aparentes, conspicuas o superficiales de la superficie terrestre y la vegetación.
 - **Fisiología:** estudia la interacciones de los elementos básicos de un ser vivo con su entorno y explicando el por qué de cada diferente situación en que se puedan encontrar estos elementos, como cambios de color, variación de tamaño con respecto a otra muestra de la misma especie, tipo de pelaje o de plumaje, etc., viendo como causa de estos cambios factores meteorológicos, cambios de temperatura, altura y otros.
- **G**
 - **Género:** en Taxonomía, el género es una categoría taxonómica que se ubica entre la familia y la especie; así, un género es un grupo de organismos que a su vez puede dividirse en varias especies (existen algunos géneros que son monoespecíficos, es decir, contienen una sola especie).
- **H**
 - **Hardware:** corresponde a todas las partes tangibles de un sistema informático; sus componentes son: eléctricos, electrónicos, electromecánicos y mecánicos.

- **Herpetología:** es la rama de la zoología que estudia a los reptiles y anfibios. El estudio de los anfibios beneficia a nuestro conocimiento del estado del ambiente, porque son muy sensibles a las perturbaciones de los ecosistemas, especialmente la contaminación, en parte por que su primer desarrollo se produce en ambientes acuáticos frecuentemente poco extensos o temporales. Algunos venenos y toxinas producidas por los reptiles y los anfibios son útiles en la medicina humana; por ejemplo, el estudio de los venenos de ciertas serpientes se investiga en busca de fármacos anticoagulantes.
 - **Hospedero:** en biología, se llama huésped, hospedador, hospedante y hospedero a aquel organismo que alberga a otro en su interior o lo porta sobre sí, ya sea en una simbiosis de parásito, un comensal o un mutualista.
- **I**
 - **Ictiología:** es una rama de la zoología dedicada al estudio de los peces. Esta incluye los osteíctios (peces óseos), los condriictios (peces cartilaginosos) tales como el tiburón y la raya y los agnatos (peces sin mandíbula). Se estima que hay alrededor de 32,200 especies descritas y que cada año son descritas oficialmente 250 nuevas especies.
 - **Interfaz de usuario:** es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.
 - **Invertebrado:** son todos aquellos animales que no se encuadran dentro de la categoría de los vertebrados. El nombre alude a que, a diferencia de estos últimos, carecen de columna vertebral y de esqueleto interno articulado. Agrupa al 95% de todas las especies animales.
- **J**
 - **Java:** es un lenguaje de programación orientado a objetos.
- **M**
 - **Malacología:** es la rama de la zoología encargada del estudio de los moluscos, el segundo grupo más grande de animales, en términos de especies descritas. Una división de la malacología, la conchiliología, se encarga del estudio de los moluscos con concha. Los campos de investigación de la malacología incluyen taxonomía, ecología, paleontología y evolución. Los conocimientos de la malacología se usan en aplicaciones médicas, veterinarias y agrarias.

- **Mastozoología:** también llamada mamiferología o teriología, es la ciencia que estudia los mamíferos. Este término se utiliza con frecuencia para la clasificación y diferenciación de ciertas especies de mamíferos que habitan la zona de Australia y Sudamérica de las del resto del mundo, debido a sus peculiares características fisiológicas (fundamentalmente en lo que se refiere a la gestación).
 - **Método de caja negra:** en teoría de sistemas y física, se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno.
 - **Modelo conceptual:** explica cuales son y cómo se relacionan los conceptos relevantes en la descripción del problema.
 - **Modelo de prototipos:** en Ingeniería de software, pertenece a los modelos de desarrollo evolutivo. El prototipo debe ser construido en poco tiempo, centrándose en una representación de aquellos aspectos del software que serán visibles para el cliente o el usuario final. Este diseño conduce a la construcción de un prototipo, el cual es evaluado por el cliente para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará.
 - **Modelo en V:** es una representación gráfica del ciclo de vida del desarrollo de sistemas. En él se resumen las principales medidas que deben adoptarse en relación con las prestaciones correspondientes en el marco del sistema informático de validación.
 - **Modelo físico:** es una descripción de la implementación de una base de datos en memoria secundaria: las estructuras de almacenamiento y los métodos utilizados para tener un acceso eficiente a los datos. Por ello, el diseño físico depende del SGBD concreto y el esquema físico se expresa mediante su lenguaje de definición de datos.
 - **Modelo lógico:** el objetivo del diseño lógico es convertir los esquemas conceptuales locales en un esquema lógico global que se ajuste al modelo de la base de datos sobre el que se vaya a implementar el sistema.
 - **Museo de Historia Natural:** Es una institución en servicio a la sociedad que se dedica a adquirir, conservar, investigar, exponer y exhibir diferentes tipos de objetos orgánicos e inorgánicos que forman parte de la historia de un país o región.
- **O**
 - **ORM:** El mapeo objeto-relacional (más conocido por su nombre en inglés, Object-Relational mapping, o sus siglas O/RM, ORM, y O/R mapping) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia. En la práctica esto crea una base de datos orientada a objetos virtual, sobre la base de datos relacional. Esto posibilita el uso de las características propias de la orientación a objetos (básicamente herencia y polimorfismo). Hay paquetes comerciales y de uso libre disponibles que desarrollan el mapeo relacional de objetos, aunque algunos programadores prefieren crear sus propias herramientas ORM.

- **Ornitología:** es la rama de la zoología que se dedica al estudio de las aves. Numerosos aspectos de la ornitología difieren de las disciplinas relacionadas, debido en parte a la alta visibilidad y el atractivo estético de las aves. Una de las diferencias más notables es la importancia y cantidad de estudios llevados a cabo por aficionados que trabajan dentro de los parámetros de la metodología científica.
- **Orden:** en biología, el orden es la categoría taxonómica entre la clase y la familia. En zoología, es una de las categorías taxonómicas de uso obligatorio, según el Código Internacional de Nomenclatura Zoológica. En taxonomía antigua era sinónimo de familia.

- **P**

- **Paleontología:** es la ciencia que estudia e interpreta el pasado de la vida sobre la Tierra a través de los fósiles. Se encuadra dentro de las Ciencias Naturales, posee un cuerpo de doctrina propio y comparte fundamentos y métodos con la Geología y la Biología, con las que se integra estrechamente.
- **Patrimonio arquitectónico:** Se entiende por un edificio, un conjunto de edificios o las ruinas de un edificio o de un conjunto de edificios que, con el paso del tiempo, han adquirido un valor mayor a su valor original, debido a factores culturales o emocionales, físico, tangible, histórico.
- **Patrimonio Cultural:** Son los todos los bienes y valores culturales que identifican nuestra nacionalidad o la identidad de nuestro país. como lo son las tradiciones, costumbres y los hábitos, así como el conjunto de bienes inmateriales y materiales, muebles e inmuebles, que poseen un especial interés histórico, artístico, estético, plástico, arquitectónico, urbano, arqueológico, ambiental, ecológico, lingüístico, sonoro, musical, audiovisual, fílmico, científico, testimonial, documental, literario, bibliográfico, museológico, antropológico y las manifestaciones, los productos y las representaciones de la cultura popular.
- **Peana:** Baza o apoyo para colocar encima una figura u otra cosa.
- **PH:** el potencial de hidrógeno es una medida de acidez o alcalinidad de una disolución.
- **Phylum:** El filo (phylum, plural phyla), tronco o tipo de organización es una categoría taxonómica situada entre el Reino y la Clase, y usada en el reino animal, reino protistas y dominio bacterias. En Botánica (reino Plantae), se emplea el término división en lugar de filo, siendo ambos términos equivalentes.
- **Plan de implementación:** es un documento formal el cual contiene la planeación, preparación del entorno, organización del personal, capacitación del personal y puesta en marcha de un proyecto.
- **Plan de pruebas:** permite especificar lo que desea probar y cómo ejecutar dichas pruebas.
- **Proceso Unificado Racional:** habitualmente resumido como RUP, es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

- **R**

- **Requerimientos:** es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Se usa en un sentido formal en la ingeniería de sistemas o la ingeniería de software.

- **S**

- **Salinidad:** es el contenido de sales minerales disueltas en un cuerpo de agua. Dicho de otra manera, es válida la expresión salinidad para referirse al contenido salino en suelos o en agua.
- **SCRUM:** es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software. Aunque SCRUM estaba enfocado a la gestión de procesos de desarrollo de software, puede ser utilizado en equipos de mantenimiento de software, o en una aproximación de gestión de programas.
- **Sinonimia:** en taxonomía sinonimia se refiere a la existencia de más de un nombre científico para un mismo taxón. El uso y la terminología de los sinónimos son diferentes para la Zoología y la Botánica.
- **Sistema Gestor de Base de Datos:** (SGBD) es un conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad.
- **Sistema informático:** es el conjunto de partes interrelacionadas, hardware, software y de recurso humano que permite almacenar y procesar información.
- **Software:** se conoce como software al equipamiento lógico o soporte lógico de un sistema informático, comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos, que son llamados hardware.
- **SQL:** El lenguaje de consulta estructurado o SQL (por sus siglas en inglés structured query language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en estas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar -de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre ella.

- **T**

- **Taxidermia:** se deriva de la palabra griega "taxis" que significa acondicionamiento y la palabra "dermis" el cual es piel. Es el arte de dar vida a las pieles de diferentes animales, utilizada desde tiempos antiguos por los cazadores para la conservación de los animales cazados como premio, ahora utilizados por los museos para exhibiciones animales al público.

- **Taxonomía:** es la ciencia de la clasificación. Habitualmente, se emplea el término para designar a la taxonomía biológica, la ciencia de ordenar a los organismos en un sistema de clasificación compuesto por una jerarquía de taxones anidados.
 - **Tipado:** característica de un lenguaje de programación que controla que no sean violados los tipos de datos en un código de programación.
 - **Top-down:** es una estrategia de desarrollo que formula un resumen del sistema, sin especificar detalles. Cada parte del sistema se refina diseñando con mayor detalle. Cada parte nueva es entonces redefinida, cada vez con mayor detalle, hasta que la especificación completa es lo suficientemente detallada para validar el modelo.
 - **Topografía:** es la ciencia que estudia el conjunto de principios y procedimientos que tienen por objeto la representación gráfica de la superficie de la Tierra, con sus formas y detalles, tanto naturales como artificiales.
 - **Tribu:** en biología, la tribu es una categoría taxonómica optativa, intermedia entre la familia y el género (o los sub o super correspondientes que existan), empleada para organizar las familias que contienen muchos géneros.
- **U**
 - **UML:** Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados
 - **URL:** en inglés uniform resource locator, es una secuencia de caracteres que se usa para nombrar recursos en Internet para su localización o identificación.
 - **Usuario:** es un conjunto de permisos y de recursos (o dispositivos) a los cuales se tiene acceso. Es decir, un usuario puede ser tanto una persona como una máquina, un programa, etc.
- **V**
 - **Variedad:** también conocida como nomenclatura trinomial (o, nombre trinomial) es una forma extendida de la nomenclatura binomial usada en biología para identificar a cada especie de ser vivo conocida, con el fin de referirse específicamente a subdivisiones menores de una especie determinada.
 - **Vertebrado:** es una categoría que comprende a los animales con espina dorsal o columna vertebral compuesta de vértebras. Incluye casi 62,000 especies animales actuales y muchos fósiles.

- **X**
 - **Xilófago:** es un término usado en ecología para describir los hábitos de un consumidor primario cuya dieta consiste principalmente (a menudo exclusivamente) en madera.
 - **XML:** son las siglas de Extensible Markup Language, una especificación/lenguaje de programación desarrollada por el W3C. XML es una versión de SGML, diseñado especialmente para los documentos de la web.

- **Z**
 - **Zoología:** es la disciplina biológica que se encarga del estudio de los animales.

ANEXOS

A. PLANIFICACIÓN DE RECURSOS

A continuación se presenta la cuantificación de los recursos a utilizar durante el desarrollo del sistema informático.

En la tabla A.1 se muestra el costo del recurso humano que intervendrá durante el periodo de desarrollo del proyecto.

Puesto	Cantidad de personas	Salario Individual	Salario Mensual	Costo Total
Director del Proyecto¹	1	\$900.00	\$ 900.00	\$7,200.00
Analista Programador¹	4	\$539.00	\$2,156.00	\$17,248.00
Costo Total				\$24,448.00

Tabla A.1. Costo total del recurso humano.

En la tabla A.2 se muestra el costo de recursos de computación que se utilizará durante el desarrollo del proyecto.

Equipo	Cantidad	Costo Individual	Costo Total
Computadoras	4	\$700.00	\$2,800.00
Servidor	1	\$800.00	\$ 800.00
Switch 8 puertos	2	\$18.00	\$ 36.00
Impresores	4	\$50.00	\$ 200.00
Costo Total			\$3,836.00

Tabla A.2. Costo total de recursos de computación.

Debido a que el software para analizar, diseñar, construir y programar así como la base de datos es software libre no se tendrá un costo de licencias para la creación del proyecto, como se muestra en la tabla A.3.

Software	Costo
Sistema gestor de base de datos	\$0.00
Herramientas para diseñar	\$0.00
Herramientas para programar	\$0.00
Costo Total	\$0.00

Tabla A.3. Costo total de licencias de software.

¹ García, C; "Gerencia Informática"; Sexta edición; 2009; página 3.

El costo estimado de consumibles durante el desarrollo del proyecto se muestra en la tabla A.4.

Materiales	Cantidad	Costo Individual	Costo Total
Papelería	10 Resmas	\$4.50	\$45.00
Cartucho nuevos	2 parejas blanco y color	\$25.00	\$50.00
Rellenado para cartuchos	5 kits	\$6.00	\$30.00
Gasolina	1 persona	\$320.00	\$320.00
Transporte	3 personas	\$30.00	\$90.00
Comida	4 personas	\$30.00	\$120.00
Costo Total			\$ 655.00

Tabla A.4. Costo total de consumibles.

El cálculo del consumo de energía eléctrica y agua potable se ha realizado en base al pliego tarifario publicado por la Superintendencia General de Electricidad y Telecomunicaciones (SIGET)¹ y Administración Nacional de Acueductos y Alcantarillados (ANDA)². Las fórmulas usadas para el consumo son:

$$\text{Consumo Total} = \text{Consumo por Hora} * \text{Cantidad de horas}$$

$$\text{Costo por Consumo} = \text{Consumo Total} * \text{Gasto Variable}$$

$$\text{Costo Total por Consumo} = \text{Consumo Fijo} + \text{Costo por Consumo}$$

Esta cuantificación de energía eléctrica y agua potable se puede observar en las tablas A.5 y A.6 respectivamente:

- Consumo por electricidad:

Cantidad de horas	Consumo por Hora	Consumo Total	Gasto variable por hora	Costo Por consumo	Costo Fijo	Costo Total por consumo
4	0.21 Kw	0.84 Kw	\$0.25	\$0.21	\$0.82	\$1.03

Tabla A.5. Costo total por energía eléctrica.

- Consumo por agua potable:

Cantidad de horas	Consumo por Hora	Consumo Total	Gasto variable por hora	Costo Por consumo	Costo Fijo	Costo Total por consumo
4	0.06 mts ³	0.24 mts ³	\$0.21	\$0.05	\$0.10	\$0.15

Tabla A.6. Costo total por agua potable.

¹ http://www.siget.gob.sv/attachments/1624_PLIEGOS%20TARIFARIOS%202012%20ABRIL%202012.pdf

² http://www.anda.gob.sv/index.php?option=com_content&view=article&id=91&Itemid=145

El costo por servicios básicos para el desarrollo del sistema informático se muestra en la tabla A.7.

Servicios	No. de personas	Costo individual	Costo Mensual	Costo Total
Teléfono Fijo	4	\$25	\$100.00	\$800.00
Celular	4	\$10	\$40.00	\$320.00
Internet	4	\$25	\$100.00	\$800.00
Agua Potable	4	\$0.15	\$12.00	\$96.00
Electricidad	4	\$1.03	\$82.40	\$659.20
Costo Total				\$2,675.20

Tabla A.7. Costo total por servicios básicos.

El costo total de recursos económicos estimados a utilizar durante el desarrollo del proyecto se muestra en la tabla A.8.

Recurso	Costo
Recurso humano	\$24,448.00
Recursos de computación	\$3,836.00
Licencias de software	\$0.00
Consumibles	\$655.00
Servicios básicos	\$2,675.00
Imprevistos (10%)	\$3,161.40
Costo Total	\$34,775.40

Tabla A.8. Costo total de recursos.

En la tabla A.9 se puede observar la distribución de los recursos durante las etapas de desarrollo del proyecto, teniendo en cuenta las horas a trabajar se han calculado al multiplicar los días utilizados para cada etapa por las 4 horas que cada uno de los miembros del grupo de trabajo dedicará a este proyecto.

Descripción de etapas	Horas a trabajar	No. de personas	Total horas-persona	Días utilizados	Porcentaje del tiempo	Costo Total
Fase de Iniciación	320	4	4	20	15%	\$5,216.31
Fase de Elaboración	864	4	4	54	40%	\$13,910.16
Fase de Construcción	752	4	4	47	35%	\$12,171.39
Fase de Transición	224	4	4	14	10%	\$3,477.54
Total				135	100%	\$34,775.40

Tabla A.9. Costo total distribuido por etapas.

Por último, en la tabla A.10 se muestran las etapas que se llevaran a cabo durante toda la ejecución del proyecto.

Descripción de etapas	Días utilizados	Porcentaje del tiempo
Pruebas	Durante todo el proyecto	Durante todo el proyecto
Gestión del cambio y configuraciones	Durante todo el proyecto	Durante todo el proyecto
Gestión del proyecto	Durante todo el proyecto	Durante todo el proyecto
Entorno	Durante todo el proyecto	Durante todo el proyecto

Tabla A.10. Etapas a realizar durante todo el proyecto.

B. ELECCIÓN DE METODOLOGÍA DE TRABAJO

Para el desarrollo del sistema informático, es necesario establecer la metodología a seguir, la cual servirá como una guía para cumplir con los objetivos del proyecto.

Luego de seleccionar la metodología más adecuada, se determinarán sus etapas así como el enfoque de desarrollo a seguir.

B.1. MODELO DEL CICLO DE VIDA

En la tabla B.1 se muestran tres ciclos de vida básicos para el desarrollo de proyectos, con sus ventajas y desventajas:

Modelo	Ventajas	Desventajas
Cascada	<ul style="list-style-type: none">• Buen desempeño con definición estable del producto y con metodología comprendida.• Funciona bien en productos complejos y bien comprendidos.• La planeación se puede hacer anticipadamente.	<ul style="list-style-type: none">• Algunos errores se aprecian en la construcción.• Se puede volver atrás, pero con un gran esfuerzo y elevando los costos.• Si los requerimientos del usuario varían, es muy difícil satisfacerlo si el proceso se encuentra en las últimas fases.
Espiral	<ul style="list-style-type: none">• Aumenta costos por trabajo, pero disminuye los riesgos del proyecto.• Incorpora objetivos de calidad.• Integra el desarrollo con el mantenimiento.	<ul style="list-style-type: none">• Es complicado, requiere un gerenciamiento cuidadoso.• Dificultad para definir la siguiente iteración.• Se puede caer en un desarrollo de nunca acabar.
Prototipo	<ul style="list-style-type: none">• Se incrementa la productividad del equipo de desarrollo, ya que el usuario se involucra lo suficiente.• Disminuye costos de mantenimiento del producto final.• El usuario reacciona mucho mejor ante el prototipo sobre el que puede experimentar.	<ul style="list-style-type: none">• Análisis insuficiente.• Tendencia a tratar de convertir el prototipo en el sistema de producción, sin mayores observaciones.• Necesidad de contar con herramientas de desarrollo de software que permitan el desarrollo rápido de las diferentes iteraciones del prototipo.

Tabla B.1. Comparación de ciclos de vida básicos.

Se concluye que ninguno de los modelos anteriores por si solos satisface las necesidades para el desarrollo del sistema informático, debido a las desventajas que pueden afectar negativamente el flujo normal del proyecto.

Sin embargo, la combinación de los modelos anteriores ha generado nuevas metodologías que han ganado popularidad, y que son muy usadas en la actualidad. En la tabla B.2 se muestran tres modelos de los más usados, con sus ventajas y desventajas respectivas:

Modelo	Ventajas	Desventajas
Modelo en V	<ul style="list-style-type: none"> • La relación entre las etapas de desarrollo y los distintos tipos de pruebas facilitan la localización de fallos. • Es un modelo sencillo y de fácil aprendizaje. • Especifica bien los roles de los distintos tipos de pruebas a realizar. • Los usuarios del modelo en V participan en el desarrollo y el mantenimiento del mismo. 	<ul style="list-style-type: none"> • Es difícil que el cliente exponga explícitamente todos los requerimientos. • Las pruebas pueden ser caras y, a veces, no lo suficientemente efectivas. • El producto final obtenido puede que no refleje todos los requisitos del usuario.
RUP	<ul style="list-style-type: none"> • Mitigación temprana de posibles riesgos altos. • Evaluación en cada fase que permite cambios de objetivos. • Progreso visible en las etapas tempranas. • El conocimiento adquirido en una iteración puede aplicarse de iteración a iteración • Los usuarios están involucrados continuamente. 	<ul style="list-style-type: none"> • La evaluación de riesgos es compleja. • Por el grado de complejidad del proyecto puede no resultar muy adecuado. • Requiere conocimientos del proceso y de UML.
SCRUM	<ul style="list-style-type: none"> • Entrega de un producto funcional al finalizar cada período. • Posibilidad de ajustar la funcionalidad en base a la necesidad de negocio del usuario. • Visualización del proyecto día a día. 	<ul style="list-style-type: none"> • No genera toda la evidencia o documentación de otras metodologías. • No es apto para los proyectos grandes. • Plantea un problema si el desarrollo está restringido por una fecha y un precio de entrega cerrados por contrato. • Tal vez sea necesario complementarlo con otras metodologías. • Se pone al usuario final en una situación incómoda para él.

Tabla B.2. Comparación de ciclos de vida usados en la actualidad.

Para la elección de la metodología más adecuada para el desarrollo del sistema informático, se ha definido los criterios que deben prevalecer durante el proyecto, los cuales son:

- **Permitir cambios en los requerimientos:** el modelo debe permitir adaptarse al cambio o surgimiento de requerimientos de parte del usuario.
- **Participación del usuario:** el usuario debe estar incluido en el equipo de trabajo sin entorpecer sus funciones cotidianas.
- **Manejo de los riesgos:** la metodología debe ayudar a mitigar los riesgos de forma temprana y continúa.
- **Flexibilidad en el desarrollo de tareas:** el modelo debe permitir el ajuste de las tareas según las necesidades del proyecto.
- **Monitoreo de avances:** la metodología debe brindar avances concretos (prototipos) durante las fases del proyecto.

En la tabla B.3 se han dividido los criterios de evaluación en críticos y no críticos, con su respectivo coeficiente de ponderación, los cuales fueron establecidos en consenso por el grupo de trabajo:

Criterio	Coficiente de ponderación
Críticos	75%
1. Permitir cambios en los requerimientos	30%
2. Participación del usuario	20%
3. Manejo de los riesgos	25%
No críticos	25%
4. Flexibilidad en el desarrollo de tareas	10%
5. Monitoreo de avances	15%

Tabla B.3. Criterios de evaluación.

En la tabla B.4 se muestra la evaluación de los modelos de ciclo de vida según los criterios definidos previamente:

Modelo	Críticos			No críticos	
	1	2	3	4	5
Modelo en V	x	✓	x	x	✓
RUP	✓	✓	✓	✓	✓
SCRUM	✓	x	✓	x	✓

Tabla B.4. Evaluación de los modelos de ciclo de vida de desarrollo de proyectos.

Los resultados obtenidos son:

- **Modelo en V:** 35%
- **RUP:** 100%
- **SCRUM:** 70%

Al analizar los resultados obtenidos, el modelo de ciclo de vida de desarrollo de proyectos que mejor se ajusta a las necesidades del proyecto es el **Proceso Unificado Racional (RUP)**.

B.2. ENFOQUE DE DESARROLLO

Para la selección del enfoque a utilizar durante el desarrollo del proyecto, se realizará una evaluación de las ventajas y desventajas del Enfoque Estructurado y el Orientado a Objetos, con lo cual se podrá elegir cuál es el más conveniente a utilizar para el desarrollo del proyecto.

En la tabla B.5 se muestran las ventajas y desventajas de ambos enfoques:

Enfoque	Ventajas	Desventajas
Estructurado	<ul style="list-style-type: none">• Simplicidad para la administración.• Los programas son más fáciles de entender debido a su estructura.• Reducción del esfuerzo en las pruebas.	<ul style="list-style-type: none">• Se obtiene un único bloque de programa, que cuando se hace demasiado grande puede resultar muy complejo manejarlo, aun con la utilización de módulos.• Poca flexibilidad en los cambios de requerimientos.
Orientado a Objetos	<ul style="list-style-type: none">• Proximidad de los conceptos de modelado respecto de las entidades del mundo real.• Reutilización del código generado.• Dinamismo en el manejo de los datos.• Facilidad en el mantenimiento y expansión.• Favorece el desarrollo iterativo del sistema.• Su diseño se realiza con UML	<ul style="list-style-type: none">• Es más largo por la cantidad de diagramas de diseño que conlleva.• Renuencia de muchos programadores a este enfoque.• Las aplicaciones consumen más recursos.

Tabla B.5. Comparación de enfoques de desarrollo.

Después de analizar las ventajas y desventajas de los enfoques anteriores, se concluye que:

- El enfoque orientado a objetos permite adaptarse a los cambios de requerimientos que puedan surgir en el desarrollo del proyecto.
- Tanto el enfoque orientado a objetos como el método RUP trabajan con UML, lo que facilitará su integración.

Por lo tanto se optará por el enfoque orientado a objetos para el desarrollo del proyecto.

B.3. ETAPAS DEL DESARROLLO DEL PROYECTO

El Proceso Unificado Racional (Rational Unified Process en inglés, y sus siglas RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (Unified Modeling Language en inglés, y sus siglas UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino que trata de un conjunto de metodologías adaptables al contexto y necesidades de cada organización, donde el software es organizado como una colección de unidades atómicas llamados objetos, constituidos por datos y funciones, que interactúan entre sí.

RUP se divide en cuatro fases, dentro de las cuales se realizan varias iteraciones según el proyecto y en las que se hace mayor o menos esfuerzo en las distintas actividades, tal como se muestra en la figura B.1:



Figura B.1. Esfuerzo en actividades según la fase del proyecto.

Dependiendo de la iteración del proceso, se puede realizar diferentes tipos de actividades en paralelo.

Las fases que componen RUP son:

1. **Fase de Iniciación:** fase donde se pone mayor énfasis en actividades del modelado del negocio y toma de requerimientos, definiendo la funcionalidad y capacidades del sistema a construir.
2. **Fase de Elaboración:** esta fase está orientada al desarrollo de la arquitectura del sistema, al refinamiento del modelo inicial, análisis y diseño.

3. **Fase de Construcción:** fase donde se hace énfasis en la construcción del sistema informático por medio de una serie de iteraciones. Por cada iteración, se refina el análisis y diseño de los requerimientos y se ejecutan las pruebas pertinentes. Además, se documenta el sistema construido como el manejo del mismo.
4. **Fase de Transición:** en esta fase el sistema informático se ha terminado según las especificaciones del usuario.

Las disciplinas conllevan los flujos de trabajo, los cuales son una secuencia de pasos para la culminación de cada disciplina. Las disciplinas de flujos de trabajo del proceso necesarias para la realización sistema informático son:

1. Modelado del Negocio
2. Requerimientos
3. Análisis y Diseño
4. Implementación
5. Pruebas
6. Despliegue

En la tabla B.6 se muestra el resumen de los principales productos de RUP y en qué momento deben iniciarse y terminarse usando una plantilla preestablecida. La ventaja de este modelo radica en que puede ser adoptado según las necesidades del sistema a construir, sin patrones fijos

Flujo	Productos	Inicio	Elaboración	Construcción	Transición
Administración del proyecto	Plan de desarrollo	I	R	R	R
	Caso de negocio	I			
	Lista de riesgos	I	R	R	R
Requisitos	Modelo de casos de uso	I	R		
	Visión	I	R		
	Especificación adicional	I	R		
	Glosario	I	R		
Análisis y diseño	Modelo de diseño		I	R	
	Documentación de la arquitectura SW		I		
Implementación	Modelo de implementación		I	R	R
Pruebas	Plan de pruebas		I	R	
Despliegue	Plan de despliegue				I

Tabla B.6. Principales producto en RUP, donde I = inicio, R = refinamiento.

Tanto las fases como las disciplinas serán adaptadas para el desarrollo del Sistema Informático de Clasificación de Zoología para el MUHNES.

- **MODELADO DEL NEGOCIO**

Esta disciplina tiene como objetivos comprender la estructura y la dinámica de la organización, comprender problemas actuales e identificar posibles mejoras, y comprender los procesos de registro y consulta de la colección de zoología del MUHNES. Para ello hay que llevar a cabo las siguientes tareas que se presentan en la tabla B.7:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Identificación de la oportunidad de mejora	<ul style="list-style-type: none"> • Entrevista • Observación 	<ul style="list-style-type: none"> • Videgrabadora 	<ul style="list-style-type: none"> • Equipo de trabajo • Jefe de unidad informática • Directora del museo
Análisis de situación actual	<ul style="list-style-type: none"> • Enfoque de sistemas 	<ul style="list-style-type: none"> • Editor de gráficos vectoriales 	<ul style="list-style-type: none"> • Equipo de trabajo
Planteamiento del problema	<ul style="list-style-type: none"> • Técnica de la caja negra • lluvia de ideas • diagrama de Ishikawa 	<ul style="list-style-type: none"> • Editor de gráficos vectoriales • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo
Estudio de factibilidad	<ul style="list-style-type: none"> • Análisis Costo/Beneficio 	<ul style="list-style-type: none"> • Calculadora • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo
Modelado del proceso	<ul style="list-style-type: none"> • Casos de uso 	<ul style="list-style-type: none"> • Editor de gráficos vectoriales 	<ul style="list-style-type: none"> • Equipo de trabajo

Tabla B.7. Tareas a realizar en la disciplina de modelado del negocio.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- Identificar el problema a resolver dentro del área de zoología del Museo de Historia Natural Saburo Hirao.
- Conocer el proceso actual de registro y clasificación de especies animales, así como las personas que intervienen en este.
- Comprobar si el proyecto a ejecutar es factible o no.

- **REQUERIMIENTOS**

El propósito de esta disciplina es establecer y mantener un acuerdo entre los usuarios y el equipo de desarrollo acerca de lo que debe hacer el sistema informático una vez finalizado. En la tabla B.8 se puede observar las tareas a realizar para la toma de requerimientos:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Recolección de requerimientos	<ul style="list-style-type: none"> • Entrevista • Cuestionario • Observación directa 	<ul style="list-style-type: none"> • Videgrabadora 	<ul style="list-style-type: none"> • Directora del museo • Jefe de unidad informática • Equipo de trabajo
Análisis del sistema	<ul style="list-style-type: none"> • Diagrama de casos de uso 	<ul style="list-style-type: none"> • Editor de gráficos vectoriales 	<ul style="list-style-type: none"> • Equipo de trabajo
Visión	<ul style="list-style-type: none"> • Entrevista 	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo • Directora del museo • Jefe de unidad informática
Glosario	<ul style="list-style-type: none"> • Entrevista • Cuestionario 	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo • Directora del museo • Jefe de unidad informática
Validación de requerimientos	<ul style="list-style-type: none"> • Revisión técnica formal • Lista de verificación • Glosario técnico 	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Directora del museo • Jefe de unidad informática • Equipo de trabajo

Tabla B.8. Tareas a realizar en la disciplina de requerimientos.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- Una descripción aprobada por los usuarios de los requerimientos funcionales o funcionalidad del sistema, así como los requerimientos no funcionales o características específicas que este debe de cumplir.
- Un glosario de términos que recoja el vocabulario propio del dominio del sistema.

- **ANÁLISIS Y DISEÑO**

En esta disciplina, se transforman los requerimientos a diseños del sistema, que estén acorde al entorno en el cual este funcionará. El análisis y diseño se ejecutará por cada requerimiento que necesite ser satisfecho (iteraciones). En la tabla B.9 se pueden observar las tareas que se tienen que ejecutar:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Definición de estándares de diseño	<ul style="list-style-type: none"> • Investigación bibliográfica 	<ul style="list-style-type: none"> • Libros • Sitios de Internet 	<ul style="list-style-type: none"> • Equipo de trabajo
Diseño de la base de datos	<ul style="list-style-type: none"> • Diagrama de flujo de datos • Modelo lógico • Modelo físico • Diccionario de datos 	<ul style="list-style-type: none"> • Herramienta de modelado de datos • Editor de gráficos vectoriales • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo
Diseño de la interfaz de usuario	<ul style="list-style-type: none"> • Diseño de salidas • Diseño de entradas • Diseño de procesos • Diseño de interfaces 	<ul style="list-style-type: none"> • Editor de páginas web 	<ul style="list-style-type: none"> • Equipo de trabajo
Validación de requerimientos	<ul style="list-style-type: none"> • Revisión técnica formal • Lista de verificación 	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Directora del museo • Jefe de unidad informática • Equipo de trabajo
Elaboración de documentación	-	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo

Tabla B.9. Tareas a realizar en la disciplina de análisis y diseño.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- Las especificaciones de diseño del sistema informático aprobadas por los usuarios.
- Los estándares a utilizar para llevar a cabo el diseño y la construcción del sistema informático.
- La documentación del diseño de cada una de las partes que conformaran el sistema informático.

- **IMPLEMENTACIÓN**

Esta disciplina se refiere a definir la organización del código, es decir, codificar los componentes en base al análisis y diseño previos. También incluye pruebas individuales a cada uno de los componentes para luego ser integrados en su totalidad. El término *Implementación* se refiere a la *Codificación e Integración* del sistema informático, no a la puesta en marcha del mismo. En la tabla B.10 se muestran las tareas a realizar durante esta disciplina:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Definición de estándares de programación y de base de datos	<ul style="list-style-type: none"> • Investigación bibliográfica 	<ul style="list-style-type: none"> • Libros • Sitios de Internet 	<ul style="list-style-type: none"> • Equipo de trabajo
Elaboración de plan de integración	<ul style="list-style-type: none"> • Investigación bibliográfica 	<ul style="list-style-type: none"> • Libros • Sitios de Internet 	<ul style="list-style-type: none"> • Equipo de trabajo
Construcción de prototipo	<ul style="list-style-type: none"> • Construcción de interfaces 	<ul style="list-style-type: none"> • Editor de páginas web 	<ul style="list-style-type: none"> • Equipo de trabajo
Construcción de base de datos	<ul style="list-style-type: none"> • Modelo relacional 	<ul style="list-style-type: none"> • Sistema gestor de base de datos • Herramienta de modelado de datos 	<ul style="list-style-type: none"> • Equipo de trabajo
Codificación de los subsistemas	<ul style="list-style-type: none"> • Programación orientada a objetos 	<ul style="list-style-type: none"> • Editor de páginas web • Servidor web • Lenguaje de programación web 	<ul style="list-style-type: none"> • Equipo de trabajo
Pruebas de subsistemas	<ul style="list-style-type: none"> • Pruebas unitarias 	<ul style="list-style-type: none"> • Plan de integración 	<ul style="list-style-type: none"> • Equipo de trabajo
Integración de subsistemas	<ul style="list-style-type: none"> • Pruebas de integración 	<ul style="list-style-type: none"> • Plan de integración 	<ul style="list-style-type: none"> • Equipo de trabajo

Tabla B.10. Tareas a realizar en la disciplina de implementación.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- Los estándares de programación y de base de datos a seguir durante el desarrollo del sistema informático.
- Una base de datos y subsistemas que se apeguen a los requerimientos aprobados por los usuarios.
- Prototipos de los subsistemas funcionales y validados.
- El plan de integración de los subsistemas.

- **PRUEBAS**

Esta disciplina actúa como un proveedor de servicios a las otras disciplinas, ya que busca asegurar la calidad de todos los componentes, verificando que todos los requerimientos se cumplan a cabalidad con el sistema informático integrado.

En la tabla B.11 se muestran las tareas a realizar en esta disciplina:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Pruebas unitarias	<ul style="list-style-type: none"> • Caja negra 	-	<ul style="list-style-type: none"> • Equipo de trabajo
Pruebas de integración	<ul style="list-style-type: none"> • Top-down 	-	<ul style="list-style-type: none"> • Equipo de trabajo
Pruebas de sistema	<ul style="list-style-type: none"> • Pruebas de recuperación • Pruebas de seguridad • Pruebas de resistencia • Pruebas de desempeño 	-	<ul style="list-style-type: none"> • Equipo de trabajo

Tabla B.11. Tareas a realizar en la disciplina de pruebas.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- El plan de pruebas a seguir para verificar la funcionalidad del sistema informático.
- La detección de errores de construcción y lógicos para ser corregidos.

- **DESPLIEGUE**

Esta disciplina tiene como objetivo asegurar que el sistema informático funcione en un ambiente similar al final, es decir, antes de ser entregado al usuario. Debido a los alcances del proyecto, el sistema informático no será puesto en producción, pero se desarrollara el plan de implementación para que sea puesto en marcha por la Gerencia Informática y Sistemas de la Secretaría de Cultura de la República de El Salvador.

En la tabla B.12 se pueden observar las tareas a seguir en esta disciplina:

Tarea	Técnicas	Herramientas y Equipo	Recursos
Pruebas del sistema informático	<ul style="list-style-type: none"> • Pruebas unitarias • Pruebas de integración • Pruebas de sistemas 	<ul style="list-style-type: none"> • Plan de pruebas 	<ul style="list-style-type: none"> • Equipo de trabajo
Manual técnico	-	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo
Manual de usuario	-	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo
Plan de implementación	-	<ul style="list-style-type: none"> • Procesador de texto 	<ul style="list-style-type: none"> • Equipo de trabajo

Tabla B.12. Tareas a realizar en la disciplina de despliegue.

Lo que se espera obtener al finalizar esta disciplina es lo siguiente:

- Un sistema informático funcional, confiable, aceptado y aprobado por los usuarios.
- El manual técnico que incluya la información para la instalación, desinstalación y configuración del sistema informático
- El manual de usuario que ayude a las personas a usar el sistema informático.
- El plan de implementación para poner en marcha el sistema informático.

C. PRUEBAS

C.1. PRUEBAS UNITARIAS

Pruebas unitarias del sistema informático SIZOO				
Tipo de prueba	Unitaria	Método de prueba	Caja negra	
Objetivo	Realizar las pruebas unitarias de caja negra a cada uno de los módulos que conforman el sistema, donde se compara que el valor conocido es igual o no al valor real.			
Correlativo	Paquete	Clase	Métodos probados	Resultados
001	muhnes.sizoo.modelo	PhylumTest	11	Exitoso
002	muhnes.sizoo.modelo	AcuaticoTest	14	Exitoso
003	muhnes.sizoo.modelo	AnimalTest	22	Exitoso
004	muhnes.sizoo.modelo	AreaprotegidaTest	8	Exitoso
005	muhnes.sizoo.modelo	BitacoraTest	14	Exitoso
006	muhnes.sizoo.modelo	ClaseTest	17	Exitoso
007	muhnes.sizoo.modelo	EspecimenTest	54	Exitoso
008	muhnes.sizoo.modelo	FamiliaTest	17	Exitoso
009	muhnes.sizoo.modelo	GeneroTest	13	Exitoso
010	muhnes.sizoo.modelo	InternacionalTest	12	Exitoso
011	muhnes.sizoo.modelo	MunicipioTest	16	Exitoso
012	muhnes.sizoo.modelo	NacionalTest	16	Exitoso
013	muhnes.sizoo.modelo	ObjetoTest	6	Exitoso
014	muhnes.sizoo.modelo	OperacionTest	6	Exitoso
015	muhnes.sizoo.modelo	OrdenTest	15	Exitoso
016	muhnes.sizoo.modelo	PrivilegioTest	12	Exitoso
017	muhnes.sizoo.modelo	RecoleccionTest	20	Exitoso
018	muhnes.sizoo.modelo	RecolectorTest	19	Exitoso
019	muhnes.sizoo.modelo	ResenyaTest	16	Exitoso
020	muhnes.sizoo.modelo	RestauracionTest	16	Exitoso
021	muhnes.sizoo.modelo	RolTest	6	Exitoso
022	muhnes.sizoo.modelo	RuralTest	10	Exitoso
023	muhnes.sizoo.modelo	SesionTest	10	Exitoso
024	muhnes.sizoo.modelo	SinonimiaTest	6	Exitoso
025	muhnes.sizoo.modelo	SuperfamiliaTest	13	Exitoso
026	muhnes.sizoo.modelo	TerrestreTest	10	Exitoso
027	muhnes.sizoo.modelo	UrbanoTest	12	Exitoso
028	muhnes.sizoo.modelo	UsuarioTest	18	Exitoso
029	muhnes.sizoo.modelo	ZonaTest	14	Exitoso
030	muhnes.sizoo.modelo	EspecieTest	17	Exitoso
031	muhnes.sizoo.publico	AuxiliarbeanTest.java	3	Exitoso
032	muhnes.sizoo.publico	AveBeanTest.java	3	Exitoso
033	muhnes.sizoo.publico	BusquedaBeanTest.java	5	Exitoso
034	muhnes.sizoo.publico	ContenidoBeanTest.java	16	Exitoso

Pruebas unitarias del sistema informático SIZOO				
Tipo de prueba	Unitaria	Método de prueba	Caja negra	
Objetivo	Realizar las pruebas unitarias de caja negra a cada uno de los módulos que conforman el sistema, donde se compara que el valor conocido es igual o no al valor real.			
Correlativo	Paquete	Clase	Métodos probados	Resultados
035	muhnes.sizoo.publico	InsectoBeanTest.java	3	Exitoso
036	muhnes.sizoo.publico	MamiferoBeanTest.java	3	Exitoso
037	muhnes.sizoo.publico	MoluscoBeanTest.java	3	Exitoso
038	muhnes.sizoo.publico	PezBeanTest.java	3	Exitoso
039	muhnes.sizoo.publico	ReptilBeanTest.java	3	Exitoso
040	muhnes.sizoo.controlador	AnimalBeanTest.java	38	Exitoso
041	muhnes.sizoo.controlador	AreaprotegidaBeanTest.java	15	Exitoso
042	muhnes.sizoo.controlador	BitacoraBeanTest.java	1	Exitoso
043	muhnes.sizoo.controlador	CPanelBeanTest.java	11	Exitoso
044	muhnes.sizoo.controlador	CRUDBeanTest.java	3	Exitoso
045	muhnes.sizoo.controlador	ClaseBeanTest.java	16	Exitoso
046	muhnes.sizoo.controlador	ConsultaBeanTest.java	22	Exitoso
047	muhnes.sizoo.controlador	CorreoBeanTest.java	12	Exitoso
048	muhnes.sizoo.controlador	DepartamentoBeanTest.java	11	Exitoso
049	muhnes.sizoo.controlador	EspecieBeanTest.java	16	Exitoso
050	muhnes.sizoo.controlador	EspecimenBeanTest.java	37	Exitoso
051	muhnes.sizoo.controlador	EstaticoBeanTest.java	13	Exitoso
052	muhnes.sizoo.controlador	ExportarBeanTest.java	3	Exitoso
053	muhnes.sizoo.controlador	FamiliaBeanTest.java	16	Exitoso
054	muhnes.sizoo.controlador	FichaBeanTest.java	38	Exitoso
055	muhnes.sizoo.controlador	GeneroBeanTest.java	16	Exitoso
056	muhnes.sizoo.controlador	ImportarBeanTest.java	23	Exitoso
057	muhnes.sizoo.controlador	MunicipioBeanTest.java	11	Exitoso
058	muhnes.sizoo.controlador	ObjetoBeanTest.java	11	Exitoso
059	muhnes.sizoo.controlador	OperacionBeanTest.java	11	Exitoso
060	muhnes.sizoo.controlador	OrdenBeanTest.java	16	Exitoso
061	muhnes.sizoo.controlador	PhylumBeanTest.java	13	Exitoso
062	muhnes.sizoo.controlador	PrivilegioBeanTest.java	16	Exitoso
063	muhnes.sizoo.controlador	RecoleccionBeanTest.java	29	Exitoso
064	muhnes.sizoo.controlador	RecolectorBeanTest.java	14	Exitoso
065	muhnes.sizoo.controlador	ReporteBeanTest.java	5	Exitoso
066	muhnes.sizoo.controlador	ReporteTaxonomiaBeanTest.java	3	Exitoso
067	muhnes.sizoo.controlador	RestauracionBeanTest.java	12	Exitoso
068	muhnes.sizoo.controlador	RolBeanTest.java	16	Exitoso

Pruebas unitarias del sistema informático SIZOO				
Tipo de prueba	Unitaria	Método de prueba	Caja negra	
Objetivo	Realizar las pruebas unitarias de caja negra a cada uno de los módulos que conforman el sistema, donde se compara que el valor conocido es igual o no al valor real.			
Correlativo	Paquete	Clase	Métodos probados	Resultados
069	muhnes.sizoo.publico	SesionBeanTest.java	12	Exitoso
070	muhnes.sizoo.publico	SuperfamiliaBeanTest.java	16	Exitoso
071	muhnes.sizoo.publico	UsuarioBeanTest.java	24	Exitoso
072	muhnes.sizoo.publico	ZonaBeanTest.java	32	Exitoso

C.2. PRUEBAS DE INTEGRACIÓN

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
001	PhylumBean, PhylumDao, PhylumDaoImpl, Phylum	Inserción	Inserción de datos en la entidad Phylum.	Inserción de datos exitosa.
002	PhylumBean, PhylumDao, PhylumDaoImpl, Phylum	Modificación	Modificación de datos en la entidad Phylum.	Modificación de datos exitosa.
003	ClaseBean, ClaseDao, ClaseDaoImpl, Clase	Inserción	Inserción de datos en la entidad Clase.	Inserción de datos exitosa.
004	ClaseBean, ClaseDao, ClaseDaoImpl, Clase	Modificación	Modificación de datos en la entidad Clase.	Modificación de datos exitosa.
005	OrdenBean, OrdenDao, OrdenDaoImpl, Orden	Inserción	Inserción de datos en la entidad Orden.	Inserción de datos exitosa.
006	OrdenBean, OrdenDao, OrdenDaoImpl, Orden	Modificación	Modificación de datos en la entidad Orden.	Modificación de datos exitosa.
007	SuperfamiliaBean, SuperfamiliaDao, SuperfamiliaDaoImpl, Superfamilia	Inserción	Inserción de datos en la entidad Superfamilia.	Inserción de datos exitosa.
008	SuperfamiliaBean, SuperfamiliaDao, SuperfamiliaDaoImpl, Superfamilia	Modificación	Modificación de datos en la entidad Superfamilia.	Modificación de datos exitosa.
009	FamiliaBean, FamiliaDao, FamiliaDaoImpl, Familia	Inserción	Inserción de datos en la entidad Familia.	Inserción de datos exitosa.
010	FamiliaBean, FamiliaDao, FamiliaDaoImpl, Familia	Modificación	Modificación de datos en la entidad Familia.	Modificación de datos exitosa.
011	GeneroBean, GeneroDao, GeneroDaoImpl, Genero	Inserción	Inserción de datos en la entidad Familia.	Inserción de datos exitosa.
012	GeneroBean, GeneroDao, GeneroDaoImpl, Genero	Modificación	Modificación de datos en la entidad Familia.	Modificación de datos exitosa.

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
013	EspecieBean, EspecieDao, EspecieDaoImpl, Especie	Inserción	Inserción de datos en la entidad Especie.	Inserción de datos exitosa.
014	EspecieBean, EspecieDao, EspecieDaoImpl, Especie	Modificación	Modificación de datos en la entidad Especie.	Modificación de datos exitosa.
015	ExportanBean, EspecieDao, EspecieDaoImpl	Exportación	Exportación de información taxonómica	Exportación de datos exitosa.
016	DepartamentoBean, DepartamentoDao, DepartamentoDaoImpl, Departamento	Inserción	Inserción de datos en la entidad Departamento.	Inserción de datos exitosa.
018	DepartamentoBean, DepartamentoDao, DepartamentoDaoImpl, Departamento	Modificación	Modificación de datos en la entidad Departamento.	Modificación de datos exitosa.
019	DepartamentoBean, DepartamentoDao, DepartamentoDaoImpl, Departamento	Eliminación	Eliminación de datos en la entidad Departamento.	Eliminación de datos exitosa.
020	MunicipioBean, MunicipioDao, MunicipioDaoImpl, Municipio	Inserción	Inserción de datos en la entidad Municipio.	Inserción de datos exitosa.
021	MunicipioBean, MunicipioDao, MunicipioDaoImpl, Municipio	Modificación	Modificación de datos en la entidad Municipio.	Modificación de datos exitosa.
022	MunicipioBean, MunicipioDao, MunicipioDaoImpl, Municipio	Eliminación	Eliminación de datos en la entidad Municipio.	Eliminación de datos exitosa.
023	AreaprotegidaBean, AreaprotegidaDao, AreaprotegidaDaoImpl, Areaprotegida	Inserción	Inserción de datos en la entidad AreaProtegida.	Inserción de datos exitosa.

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
024	AreaprotegidaBean, AreaprotegidaDao, AreaprotegidaDaoImpl, Areaprotegida	Modificación	Modificación de datos en la entidad AreaProtegida.	Modificación de datos exitosa.
025	AreaprotegidaBean, AreaprotegidaDao, AreaprotegidaDaoImpl, Areaprotegida	Eliminación	Eliminación de datos en la entidad AreaProtegida.	Eliminación de datos exitosa.
026	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Inserción	Inserción de datos en las entidades Zona y Urbano.	Inserción de datos exitosa.
027	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Modificación	Modificación de datos en las entidades Zona y Urbano.	Modificación de datos exitosa.
028	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Eliminación	Eliminación de datos en las entidades Zona y Urbano.	Eliminación de datos exitosa.
029	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Inserción	Inserción de datos en las entidades Zona y Rural.	Inserción de datos exitosa.
030	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Modificación	Modificación de datos en las entidades Zona y Rural.	Modificación de datos exitosa.
031	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Eliminación	Eliminación de datos en las entidades Zona y Rural.	Eliminación de datos exitosa.
032	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaReservaDao, ReservaDaoImpl, Reserva	Inserción	Inserción de datos en las entidades Zona y Reserva.	Inserción de datos exitosa.

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
033	AreaprotegidaBean, AreaprotegidaDao, AreaprotegidaDaoImpl, Areaprotegida	Modificación	Modificación de datos en la entidad AreaProtegida.	Modificación de datos exitosa.
034	AreaprotegidaBean, AreaprotegidaDao, AreaprotegidaDaoImpl, Areaprotegida	Eliminación	Eliminación de datos en la entidad AreaProtegida.	Eliminación de datos exitosa.
035	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Inserción	Inserción de datos en las entidades Zona y Urbano.	Inserción de datos exitosa.
036	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Modificación	Modificación de datos en las entidades Zona y Urbano.	Modificación de datos exitosa.
037	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaUrbanoDao, UrbanoDaoImpl, Urbano	Eliminación	Eliminación de datos en las entidades Zona y Urbano.	Eliminación de datos exitosa.
038	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Inserción	Inserción de datos en las entidades Zona y Rural.	Inserción de datos exitosa.
039	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Modificación	Modificación de datos en las entidades Zona y Rural.	Modificación de datos exitosa.
040	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaRuralDao, RuralDaoImpl, Rural	Eliminación	Eliminación de datos en las entidades Zona y Rural.	Eliminación de datos exitosa.
041	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaReservaDao, ReservaDaoImpl, Reserva	Inserción	Inserción de datos en las entidades Zona y Reserva.	Inserción de datos exitosa.

Pruebas de integración del sistema informático SIZOO					
Tipo de prueba		Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.				
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido	
042	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaReservaDao, ReservaDaoImpl, Reserva	Modificación	Modificación de datos en las entidades Zona y Reserva.	Modificación de datos exitosa.	
043	ZonaBean, ZonaDao, ZonaDaoImpl, ZonaReservaDao, ReservaDaoImpl, Reserva	Eliminación	Eliminación de datos en las entidades Zona y Reserva.	Eliminación de datos exitosa.	
044	RecolectorBean, RecolectorDao, RecolectDaoImpl, Recolector	Inserción	Inserción de datos en la entidad Recolector.	Inserción de datos exitosa.	
045	RecolectorBean, RecolectorDao, RecolectDaoImpl, Recolector	Modificación	Modificación de datos en la entidad Recolector.	Modificación de datos exitosa.	
046	RecoleccionBean, RecoleccionDao, RecoleccionDaoImpl, Recoleccion, NacionalDao, NacionalImpl, Nacional	Inserción	Inserción de datos en las entidades Recoleccion y Nacional.	Inserción de datos exitosa.	
047	RecoleccionBean, RecoleccionDao, RecoleccionDaoImpl, Recoleccion, NacionalDao, NacionalImpl, Nacional	Modificación	Modificación de datos en las entidades Recoleccion y Nacional.	Modificación de datos exitosa.	
048	RecoleccionBean, RecoleccionDao, RecoleccionDaoImpl, Recoleccion, InternacionalDao, InternacionalImpl, Internacional	Inserción	Inserción de datos en las entidades Recoleccion y Internacional.	Inserción de datos exitosa.	

Pruebas de integración del sistema informático SIZOO					
Tipo de prueba		Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.				
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido	
049	RecoleccionBean, RecoleccionDao, RecoleccionDaoImpl, Recoleccion, InternacionalDao, InternacionalImpl, Internacional	Modificación	Modificación de datos en las entidades Recoleccion y Internacional.	Modificación de datos exitosa.	
050	AnimalBean, AnimalDao, AnimalDaoImpl, Animal, EtiquetaDao, EtiquetaDaoImpl, Etiqueta, ResenyaDao, ResenyaDaoImpl, Resenya	Inserción	Inserción de datos en las entidades Animal, Resenya y Etiqueta.	Inserción de datos exitosa.	
051	AnimalBean, AnimalDao, AnimalDaoImpl, Animal, EtiquetaDao, EtiquetaDaoImpl, Etiqueta, ResenyaDao, ResenyaDaoImpl, Resenya	Modificación	Modificación de datos en las entidades Animal, Resenya y Etiqueta.	Modificación de datos exitosa.	
052	AnimalBean, AnimalDao, AnimalDaoImpl, Animal, SinonimiaDao, SinonimiaDaoImpl, Sinonimia	Modificación	Modificación de nombre científico en la entidad Animal	Modificación de nombre científico de animal exitosa.	
053	EspecimenBean, EspecimenDao, EspecimendaoImpl, Especimen, TerrestreDao, TerrestreDaoImpl, Terreste	Inserción	Inserción de datos en las entidades Especimen y Terrestre.	Modificación de datos exitosa.	

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
054	EspecimenBean, EspecimenDao, EspecimendaoImpl, Especimen, TerrestreDao, TerrestreDaoImpl, Terrestre	Modificación	Modificación de datos en las entidades Especimen y Terrestre.	Modificación de datos exitosa.
055	EspecimenBean, EspecimenDao, EspecimendaoImpl, Especimen, AcuaticoDao, AcuaticoDaoImpl, Acuatico	Inserción	Inserción de datos en las entidades Especimen y Acuatico	Inserción de datos exitosa.
056	EspecimenBean, EspecimenDao, EspecimendaoImpl, Especimen, AcuaticoDao, AcuaticoDaoImpl, Acuatico	Modificación	Modificación de datos en las entidades Especimen y Acuatico.	Modificación de datos exitosa.
057	RestauracionBean, RestauracionDao, RestauracionDaoImpl, Restauracion	Inserción	Inserción de datos en la entidad Restauracion.	Inserción de datos exitosa.
058	RestauracionBean, RestauracionDao, RestauracionDaoImpl, Restauracion	Modificación	Modificación de datos en la entidad Restauracion.	Modificación de datos exitosa.
059	ObjetoBean, ObjetoDao, ObjetoDaoImpl, Objeto	Inserción	Inserción de datos en la entidad Objeto.	Inserción de datos exitosa.
060	ObjetoBean, ObjetoDao, ObjetoDaoImpl, Objeto	Modificación	Modificación de datos en la entidad Objeto.	Modificación de datos exitosa.
061	ObjetoBean, ObjetoDao, ObjetoDaoImpl, Objeto	Eliminación	Eliminación de datos en la entidad Objeto.	Eliminación de datos exitosa.

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
062	OperacionBean, OperacionDao, OperacionDaoImpl, Operacion	Inserción	Inserción de datos en la entidad Operacion.	Inserción de datos exitosa.
063	OperacionBean, OperacionDao, OperacionDaoImpl, Operacion	Modificación	Modificación de datos en la entidad Operacion.	Modificación de datos exitosa.
064	OperacionBean, OperacionDao, OperacionDaoImpl, Operacion	Eliminación	Eliminación de datos en la entidad Operacion.	Eliminación de datos exitosa.
065	PrivilegioBean, PrivilegioDao, PrivilegioDaoImpl, Privilegio	Inserción	Inserción de datos en la entidad Privilegio.	Inserción de datos exitosa.
066	PrivilegioBean, PrivilegioDao, PrivilegioDaoImpl, Privilegio	Modificación	Modificación de datos en la entidad Privilegio.	Modificación de datos exitosa.
067	PrivilegioBean, PrivilegioDao, PrivilegioDaoImpl, Privilegio	Eliminación	Eliminación de datos en la entidad Privilegio.	Eliminación de datos exitosa.
068	RolBean, RolDao, RolDaoImpl, Rol	Inserción	Inserción de datos en la entidad Rol.	Inserción de datos exitosa.
069	RolBean, RolDao, RolDaoImpl, Rol	Modificación	Modificación de datos en la entidad Rol.	Modificación de datos exitosa.
070	RolBean, RolDao, RolDaoImpl, Rol	Eliminación	Eliminación de datos en la entidad Rol.	Eliminación de datos exitosa.
071	UsuarioBean, UsuarioDao, UsuarioDaoImpl, Usuario	Inserción	Inserción de datos en la entidad Rol.	Inserción de datos exitosa.

Pruebas de integración del sistema informático SIZOO				
Tipo de prueba	Integración	Método de prueba	Descendente	
Objetivo	Realizar las pruebas de integración del sistema para verificar si se realizan de forma correcta las operaciones deseadas.			
Correlativo	Clases	Operación a realizar	Resultado esperado	Resultado obtenido
072	UsuarioBean, UsuarioDao, UsuarioDaoImpl, Usuario	Modificación	Modificación de datos en la entidad Rol.	Modificación de datos exitosa.
073	UsuarioBean, UsuarioDao, UsuarioDaoImpl, Usuario	Eliminación	Eliminación de datos en la entidad Rol.	Eliminación de datos exitosa.
074	BitacoraBean, BitacoraDao, BitacoraDaoImpl, Bitacora	Inserción	Inserción de datos en la entidad Bitácora.	Inserción de datos exitosa.

C.3. PRUEBAS DE SISTEMA

Pruebas de sistema			
Fecha	21/02/2013	No. de caso de prueba	001
Tipo de prueba	Sistema	Subtipo de prueba	Recuperación
Objetivo	Verificar la correcta carga de información y restauración del sistema.		
Descripción	Se evaluaron los elementos de la base, esquemas, datos, respaldos y fallos en la red.		
Procedimiento	<ul style="list-style-type: none"> • Se prueba la carga de los esquemas de la base de datos. • Se verifican los datos que se tiene en los respaldos con los cargados. • Se realizan pruebas de fallo en la red en el momento de inserción de datos los cuales no se ingresaran para integridad de los datos. 		
Condiciones	<ul style="list-style-type: none"> • En escenarios sin permisos. • Fallos en la red. • Escenario exitoso. 		
Resultados			
Resultados esperados	Que el respaldo de los esquemas, datos y demás elementos de la base se haga de una manera que mantenga la integridad sin error alguno.		
Resultados obtenidos	<ul style="list-style-type: none"> • En la primera vez fallo por falta de permisos en Postgres. • Se probó nuevamente dando resultado exitoso en la carga tanto de esquemas como de datos. 		
Evaluación	Exitoso		
Observaciones	Verificar los permisos en la base de datos antes de cargar el respaldo de la misma.		

Pruebas de sistema			
Fecha	21/02/2013	No. de caso de prueba	002
Tipo de prueba	Sistema	Subtipo de prueba	Seguridad
Objetivo	Verificar la seguridad del sistema informático.		
Descripción	La prueba de seguridad consiste en tratar de ingresar sin tener permiso a diferentes pantallas.		
Procedimiento	<ul style="list-style-type: none"> • Probar el ingreso a páginas a través de direcciones completas. • Verificar la encriptación de los datos de usuario. • Comprobar que el menú sea generado a través de permisos en la base de datos. 		
Condiciones	<ul style="list-style-type: none"> • A través de diferentes direcciones IP. • Escenario de éxito. 		
Resultados			
Resultados esperados	Que solo usuarios registrados ingresen al sistema.		
Resultados obtenidos	<ul style="list-style-type: none"> • Se probó ingresando las URL de las paginas, con lo cual se podía ingresar al sistema. • Se probó la generación de permisos a través de la base, lo cual fue exitoso. • Se compro el cifrado de los datos del usuario. 		
Evaluación	Exitoso		
Observaciones	Se corrigió el acceso al sistema a través de URL evitando con esto que personas no registradas pudieran ingresar al sistema.		

Pruebas de sistema			
Fecha	21/02/2013	No. de caso de prueba	003
Tipo de prueba	Sistema	Subtipo de prueba	Resistencia
Objetivo	Verificar la resistencia del sistema informático.		
Descripción	Consiste en realizar ingresos masivos de la información, generación de reportes y instalación del sistema en diferentes sistemas operativos como lo son Debian y Windows.		
Procedimiento	<ul style="list-style-type: none"> • Importar grandes cantidades de datos desde la aplicación. • Instalar el sistema en sistemas operativos. • Elaboración del sistema. 		
Condiciones	<ul style="list-style-type: none"> • Diferentes sistemas operativos. • Falta de plugin para reportes. 		
Resultados			
Resultados esperados	Que el sistema funcione en cualquier sistema operativo, que genere los reportes a pesar de no tener el plugin de adobe instalado.		
Resultados obtenidos	<ul style="list-style-type: none"> • Se ingreso una carga masiva de datos que permitió verificar el funcionamiento de la importación. • Se instalo correctamente el sistema tanto en Debian como en Windows. • Cuando el plugin de adobe no estaba instalado se descarga el reporte generado. 		
Evaluación	Exitoso		
Observaciones	En todos los casos verificar que se tengan permisos de instalación.		

Pruebas de sistema			
Fecha	21/02/2013	No. de caso de prueba	004
Tipo de prueba	Sistema	Subtipo de prueba	Desempeño
Objetivo	Verificar el desempeño del sistema informático.		
Descripción	La prueba de desempeño consiste en una verificación final de cada una de las pruebas anteriores y verificar que se hayan corregido los fallos presentados.		
Procedimiento	<ul style="list-style-type: none"> • Carga masiva de datos • Verificación de permisos para el ingreso al sistema • Datos erróneos para probar que no se puedan ingresar. • Verificar tiempos en las fechas de ingreso. • Revisar que los datos presentados al público sean permitidos por el biólogo a cargo. • Revisión de los filtros utilizados para la consulta dinámica. 		
Condiciones	<ul style="list-style-type: none"> • Sin permisos para el ingreso de datos. • Escenarios exitosos. 		
Resultados			
Resultados esperados	Que el sistema cumpla con los requerimientos establecidos.		
Resultados obtenidos	<ul style="list-style-type: none"> • Se ingreso la carga masiva de datos. • Cuando se ingresaba datos erróneos, los mensajes de validación evitaban el ingreso de la información. • La información que es presentada al público es previamente mostrada al biólogo para aprobación de esta. • Se generaron diferentes consultas probando los filtros de información. 		
Evaluación	Exitoso		
Observaciones	Se corrigieron los errores dados en las pruebas anteriores, verificando tanto el respaldo, seguridad, la resistencia, y desempeño del sistema.		

Verificación de las pruebas de sistema		
Tipo de prueba	Resultado de la prueba	Observaciones
Seguridad	Exitoso	Los usuarios podrán ingresar a un sitio URL específico si han iniciado sesión en el sistema previamente.
Recuperación	Exitoso	Siempre verificar que se cuentan con los permisos respectico dentro de la base de datos.
Desempeño	Exitoso	El sistema funciona en los navegadores Internet Explorer, Mozilla Firefox y Google Chrome.
Resistencia	Exitoso	Los usuarios deben contar con los privilegios de administrador dentro del sistema operativo en uso.

D. ELECCIÓN DE MÉTODO DE CONVERSIÓN

Las estrategias de conversión definen la metodología a seguir en el proceso de convertir un sistema antiguo a uno nuevo. Al implementar alguna de estas estrategias es importante tomar en cuenta varios aspectos tales como el tipo de sistema a implementar, el recurso humano, el ambiente bajo el cual operará, entre otros.

Es por ello que se han seleccionado una serie de estrategias las cuales serán estudiadas y evaluadas para determinar la que mejor se adecue al sistema informático desarrollado.

La instalación del nuevo sistema puede realizarse según cinco métodos¹:

1. Conversión directa
2. Conversión paralela
3. Conversión gradual
4. Conversión de prototipo modular
5. Conversión distribuida

En la tabla D.1 se define cada uno de los métodos con sus ventajas y desventajas.

¹ Análisis y Diseño de Sistemas 6ª edición; Kendall & Kendall; página 636.

Método	Definición	Ventajas	Desventajas
Conversión directa	Consiste en implementar el nuevo sistema dejando de lado el sistema anterior, a partir del momento de su instalación.	<ul style="list-style-type: none"> • Es menos costoso que el paralelo, obteniendo los beneficios del sistema más rápidamente. • Los usuarios no tienen ninguna posibilidad de usar el sistema antiguo en lugar del nuevo, la adaptación es una necesidad. 	<ul style="list-style-type: none"> • Riesgo elevado, debido a que no se dispone del sistema anterior en caso de fallos del nuevo sistema. • Se requiere una planificación más cuidadosa, ya que pone a prueba en forma exigente al nuevo sistema y se debe establecer un mayor número de posibles hipótesis de conflictos con los procesos del nuevo sistema.
Conversión paralela	Una vez que el nuevo sistema está pronto y listo para operar, éste se instala y entra en funcionamiento, sin interrumpir el uso del sistema anterior.	<ul style="list-style-type: none"> • Es más seguro debido a que el sistema antiguo se mantiene en funcionamiento hasta que se haya validado el 100% del nuevo. • Ofrece un sentido de seguridad para los usuarios, quienes no están obligados a hacer un cambio abrupto al nuevo sistema. 	<ul style="list-style-type: none"> • Es más costoso, dado que se duplican los costos de ejecutar los dos sistemas al mismo tiempo y el agobio de los usuarios de duplicar virtualmente su carga de trabajo durante la conversión. • A menos que el sistema a ser reemplazado sea manual, es difícil hacer comparaciones entre las salidas del nuevo sistema y el viejo.
Conversión gradual o por fases o etapas	Introduce el nuevo sistema por etapas, es decir, aumentando gradualmente el volumen de transacciones manejadas por dicho sistema.	<ul style="list-style-type: none"> • Permite que los usuarios se involucren gradualmente con el sistema. • Los errores que pueden ocurrir tienen menos impacto ya que afectan a una cantidad relativamente pequeña de transacciones y no a su totalidad. 	<ul style="list-style-type: none"> • Debido a la conversión gradual, se debe tomar demasiado tiempo para colocar el nuevo sistema en el lugar. • Si ocurren problemas el inicio de la implementación, podrían difundirse apreciaciones negativas que perjudicaran el proceso en el futuro.
Conversión de prototipo modular	Se basa en la construcción de prototipos modulares y operacionales para cambiar el sistema antiguo al nuevo de manera gradual.	<ul style="list-style-type: none"> • Cada módulo se prueba completamente antes de ser utilizado. 	<ul style="list-style-type: none"> • Se debe de poner especial atención a las interfaces para que los módulos que se construyen realmente trabajen como un sistema.
Conversión distribuida	Se utiliza cuando se contemplan muchas instalaciones del mismo sistema en diferentes sitios geográficos de una empresa.	<ul style="list-style-type: none"> • Se pueden destacar y contener los problemas en lugar de afectar simultáneamente en todos los sitios. 	<ul style="list-style-type: none"> • Aunque la conversión es exitosa, cada sitio tendrá sus propias peculiaridades para trabajar y se deben manejar como corresponde.

Tabla D.1. Métodos de conversión.

Para la elección de la metodología más adecuada de conversión al sistema informático, se han definido los criterios que deben prevalecer durante este proceso, los cuales son mostrados en la tabla D.2.

Criterio	Descripción	Ponderación
1. Tiempo de implementación	El tiempo de puesta en marcha del sistema sea lo más pronto posible.	20%
2. Costos de implementación	Los costos de puesta en marcha del sistema no sean muy elevados.	20%
3. Adaptación de los usuarios a la estrategia de conversión	Los usuarios deben adaptarse al sistema paulatinamente.	15%
4. Examinar la confiabilidad de los resultados	Se puede verificar consistencia de la información que entrega el nuevo sistema con el antiguo.	20%
5. Tolerancia a fallos	El impacto de las fallas del nuevo sistema tiende a ser mínimo.	25%

Tabla D.2. Criterios de evaluación de las metodologías de conversión.

En la tabla D.3 se muestra la evaluación de las metodologías de conversión según los criterios definidos previamente:

Conversión	1	2	3	4	5
Directa	✓	✓	x	x	x
Paralela	✓	x	✓	✓	✓
Gradual	x	x	✓	✓	✓
Prototipo modular	x	x	✓	✓	✓
Distribuida	x	x	✓	x	✓

Tabla D.3. Evaluación de las metodologías de conversión

Los resultados obtenidos para cada uno de los métodos de conversión son:

- **Conversión directa:** 40%
- **Conversión paralela:** 80%
- **Conversión gradual o por fases o etapas:** 60%
- **Conversión de prototipo modular:** 60%
- **Conversión distribuida:** 40%

Al analizar los resultados obtenidos, el método de conversión que mejor se ajusta a las necesidades de implementación del sistema informático es la **Conversión Paralela**.