

UNIVERSIDAD DE LAS FUERZAS ARMADAS “ESPE”

DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

**“ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UNA GUÍA
INTERACTIVA Y SITIO WEB PARA LAS ÁREAS PROTEGIDAS DEL
ECUADOR”**

PREVIA A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERA EN SISTEMAS E INFORMÁTICA

POR: LUCÍA PATRICIA TRUJILLO MONTALVO

SANGOLQUÍ, AGOSTO DE 2012.

DECLARACIÓN

Yo, Lucía Patricia Trujillo Montalvo, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad de las FUERZAS ARMADAS ESPE, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Lucía Patricia Trujillo Montalvo

CERTIFICACIÓN

Certificamos que el presente trabajo, fue realizado en su totalidad por la Srta. Lucía Patricia Trujillo Montalvo como requerimiento parcial a la obtención del Título de **INGENIERA EN SISTEMAS E INFORMÁTICA**, bajo nuestra supervisión.

Sangolquí, Agosto de 2012

Ing. César Villacís

DIRECTOR

Ing. Germán Ñacato

CODIRECTOR

DEDICATORIA

Dedico este trabajo a mi Madre, que desde siempre con su esfuerzo y dedicación, supo brindarme su apoyo incondicional, enseñanzas y valores con los cuales he podido obtener mis logros y me han hecho luchar para cumplir mis metas y objetivos.

Para ti Mami, con todo mi amor, mil gracias por todo.

Lucía Patricia Trujillo Montalvo.

AGRADECIMIENTO

Agradezco a mis padres por darme la oportunidad de seguir una carrera y culminarla exitosamente.

A Dios por sus bendiciones me han permitido guiar mi vida en todo momento.

A mis profesores que con su noble labor han aportado los conocimientos adquiridos.

A mi Director de Carrera Ing. Mauricio Campaña, por el apoyo incondicional en todo el proceso.

A mi Director de Tesis Ing. César Villacís, a mi Codirector Ing. Germán Ñacato, a mi Informante Ing. Carlos Caizaguano, por su apoyo, tiempo y paciencia para el desarrollo del tema.

Mil Gracias.

Lucía Patricia Trujillo Montalvo

ÍNDICE

RESUMEN	1
ABSTRACT	2
CAPÍTULO 1	3
GENERALIDADES	3
1.1 TÍTULO	3
1.2 INTRODUCCIÓN	3
1.3 PLANTEAMIENTO DEL PROBLEMA	3
1.4 JUSTIFICACIÓN	4
1.5 OBJETIVOS	4
1.5.1 OBJETIVO GENERAL	4
1.5.2 OBJETIVOS ESPECÍFICOS.....	5
1.6 ALCANCE	5
1.7 METODOLOGÍA	6
1.8 HERRAMIENTAS	6
1.8.1 HARDWARE	6
1.8.2 SOFTWARE	7
1.9 FACTIBILIDAD	7
1.9.1 OPERATIVA.....	7
1.10 FONDO DE ÁREAS PROTEGIDAS	8
1.10.1. ORÍGENES.....	8
1.10.2. MISIÓN	8
1.10.3. VISIÓN	8
1.10.4. CARACTERÍSTICAS PRINCIPALES.....	8
1.10.5. PRINCIPIOS Y VALORES	9
1.11 CENTRO DE INFORMACIÓN AMBIENTAL	9
1.11.1. OBJETIVOS.....	9
1.11.2 FUNCIONES	10
1.11.3. INFORMACIÓN DISPONIBLE	11

CAPÍTULO 2	12
MARCO TEÓRICO	12
2.1 CONCEPTOS BÁSICOS	12
2.1.1. SOFTWARE	12
2.1.2. SOFTWARE LIBRE	12
2.1.3. OPEN SOURCE	12
2.1.4. LENGUAJES DE PROGRAMACIÓN	13
2.2 EXTREME PROGRAMMING (XP)	13
2.2.1 ALGUNAS BASES DE LA PROGRAMACIÓN EXTREMA	13
2.2.3 LOS PASOS A SEGUIR EN UN PROYECTO	14
2.2.4 PRÁCTICAS BÁSICAS DE LA PROGRAMACIÓN EXTREMA	15
2.3 INTRODUCCIÓN A LAS APLICACIONES HIPERMEDIA.....	16
2.4. HIPERMEDIA Y BASES DE DATOS ORIENTADAS A OBJETOS	19
2.4.1. METODOLOGÍAS DE DESARROLLO DE SISTEMAS HIPERMEDIALES.....	19
2.5. INGENIERÍA WEB	20
2.5.1 ESTRATOS DE LA INGENIERÍA WEB	22
2.5.2 PROCESO	22
2.5.3. MÉTODOS.....	22
2.5.4 HERRAMIENTAS Y TECNOLOGÍAS.....	24
2.5.5 EL PROCESO DE INGENIERÍA WEB	24
2.5.6 DEFINIR EL MARCO DE TRABAJO	24
2.5.7 MEJORES PRÁCTICAS EN LA INGENIERÍA WEB.	26
2.5.8 FORMULACIÓN Y PLANEACIÓN PARA INGENIERÍA WEB.....	27
2.5.9. MODELO DE ANÁLISIS PARA APLICACIONES WEB	29
2.5.10 MODELO DE DISEÑO PARA APLICACIONES WEB	34
2.6 HTML.....	40
2.7 MYSQL.....	41
2.7.1 APLICACIONES.....	42
2.7.2 USO DE MYSQL.....	42
2.7.3 CARACTERÍSTICAS DE MYSQL.....	43
2.8 MODELOS DE BASES DE DATOS	43
2.8.1 BASES DE DATOS JERÁRQUICAS.....	43
2.8.2 BASE DE DATOS DE RED	44

2.8.3 BASES DE DATOS TRANSACCIONALES	44
2.8.4 BASES DE DATOS RELACIONALES	44
2.8.5 BASES DE DATOS MULTIDIMENSIONALES	45
2.8.6 BASES DE DATOS ORIENTADAS A OBJETOS.....	46
2.9 PROGRAMACIÓN ORIENTADA A OBJETOS (POO).....	46
2.9.1 CÓMO SE PIENSA EN OBJETOS.....	47
2.9.2 CLASES EN POO	48
2.9.3 OBJETOS EN POO	48
2.10 PHP.....	50
2.10.1. HISTORIA	50
2.10.2 USOS DE PHP	50
2.10.3 CARACTERÍSTICAS DE PHP.....	51
2. 11 JAVASCRIPT.....	52
2.12 AJAX	53
2.12.1 INTRODUCCIÓN A AJAX.....	53
2.13 RUP	56
2.13.1. ENTREGA DEL SOFTWARE OPERATIVO CON CONFIANZA	57
2.13.2. CONTROL DE NUEVAS HERRAMIENTAS Y TECNOLOGÍAS	57
2.13.3. CARACTERÍSTICAS Y BENEFICIOS.....	57
2.13.4. SISTEMAS OPERATIVOS Y PLATAFORMAS DE HARDWARE APROPIADAS	58
2.14 CSS	58
2.14.1 SOPORTE DE CSS EN LOS NAVEGADORES	59
2.14.2 FUNCIONAMIENTO BÁSICO DE CSS	59
2.15 UML.....	59
2.16 GENERADOR DE CÓDIGO CREACOD	60
2.16.1 COMO GENERA CÓDIGO CREACOD	61
<i>CAPÍTULO 3</i>	<i>62</i>
<i>ANÁLISIS Y DISEÑO DEL WEBSITE Y GUÍA INTERACTIVA</i>	<i>62</i>
3.1. ANÁLISIS DE SITUACIÓN ACTUAL	62
3.2 MENÚ HORIZONTAL DEL WEB SITE ACTUAL.....	63
3.3 FORMULACIÓN	64

3.4 ESPECIFICACIÓN DE REQUERIMIENTOS	64
3.4.1 INTRODUCCIÓN.....	64
3.4.2 ESPECIFICACIONES DE CASOS DE USO	65
3.5. DIAGRAMAS DE SECUENCIA	98
3.5.1 DIAGRAMA DE SECUENCIA PARA ADMINISTRADOR - GESTIÓN DE CONTENIDO E INFORMACIÓN	98
3.5.2. DIAGRAMA DE SECUENCIA – USUARIO.....	99
3.7 HISTORIAS DE USUARIO	101
3.7.1 HISTORIA 1 (H1)	101
3.7.2 HISTORIA 2 (H2)	102
3.7.3 HISTORIA 3 (H3)	103
3.7.4 HISTORIA 4 (H4)	104
3.7.5 HISTORIA 5 (H5)	105
3.7.6 HISTORIA 6 (H6)	106
3.7.7 HISTORIA 7 (H7)	107
3.7.8 HISTORIA 8 (H8)	108
3.7.9 TABLA DE HISTORIAS DE USUARIO	109
3.8 PLAN DE ENTREGA	110
3.9 PLANTEAMIENTO	111
3.9.1 ITERACIÓN PRIMERA:.....	111
3.9.2 ITERACIÓN SEGUNDA:.....	113
3.10 INCIDENCIAS	121
3.11 TARJETAS C.R.C.	122
3.8 PROTOTIPOS	126
3.8.1 PANTALLA PRINCIPAL.....	126
3.8.2 CREACIÓN DE CLASES.....	127
3.8.3 PÁGINA WEB PRINCIPAL	128
3.8.4 PÁGINA DE INFORMACIÓN SOBRE ÁREAS PROTEGIDAS.....	129
3.8.5 PANTALLA DE FLORA.....	130
3.8.6 PANTALLA DE FAUNA	132
3.9.1 Creación de la Base de Datos.....	133
3.9.2 CREACIÓN DE LAS PANTALLAS DE CONTROL USANDO EL PROGRAMA CREACOD.....	135
3.9.2.10 CREACIÓN DEL CD MULTIMEDIA.....	140
3.9.3 PRUEBAS DEL PORTAL	144

3.9.1 PRUEBAS DE CONTENIDO	144
3.9.2 PRUEBAS DE INTERFAZ DE USUARIO	145
3.9.3 PRUEBAS DE NAVEGACIÓN	145
3.9.4 PRUEBAS DE COMPONENTES.....	145
3.9.5 PRUEBAS DE CONFIGURACIÓN	145
3.9.6 PRUEBAS DE SEGURIDAD	146
3.9.7 PRUEBAS DE DESEMPEÑO	146
CAPÍTULO 4.....	147
CONCLUSIONES, RECOMENDACIONES Y BIBLIOGRAFÍA.....	147
4.1 CONCLUSIONES.....	147
4.2 RECOMENDACIONES	147
4.3 Bibliografía:.....	148
ANEXOS.....	<i>¡Error! Marcador no definido.</i>
A.1 PANTALLAS DE GESTIÓN	<i>¡Error! Marcador no definido.</i>
A.1.1 PANTALLA PARA GESTIÓN DE ÁREAS.....	<i>¡Error! Marcador no definido.</i>
A.1.2 PANTALLA DE GESTIÓN PARA ÁREA TIPO	<i>¡Error! Marcador no definido.</i>
A.1.3 PANTALLAS DE GESTIÓN PARA FLORA.....	<i>¡Error! Marcador no definido.</i>
A.1.4 PANTALLAS DE GESTIÓN DE FAMILIA PARA FLORA Y FAUNA	<i>¡Error! Marcador no definido.</i>
A.1.5 PANTALLAS DE GESTIÓN PARA FAUNA.....	<i>¡Error! Marcador no definido.</i>
A.1.6 PANTALLAS DE GESTIÓN PARA GEOGRAFÍA	<i>¡Error! Marcador no definido.</i>
A.1.7 PANTALLAS DE GESTIÓN PARA HIDROGRAFÍA	<i>¡Error! Marcador no definido.</i>
A.1.8 PANTALLAS DE GESTIÓN PARA TURISMO	<i>¡Error! Marcador no definido.</i>
A.1.9 PANTALLAS DE GESTIÓN PARA TURI-GRUP	<i>¡Error! Marcador no definido.</i>
A.1.10 PANTALLAS DE GESTIÓN PARA VESTIMENTA	<i>¡Error! Marcador no definido.</i>
A.1.11 PANTALLAS PARA GESTIÓN DE REGIÓN (VESTIMENTA, PROVINCIA, CLIMA, ETNIA)	<i>¡Error! Marcador no definido.</i>
Marcador no definido.	
A.1.12 PANTALLAS PARA GESTIÓN DE PROVINCIA	<i>¡Error! Marcador no definido.</i>
A.1.13 PANTALLAS DE GESTIÓN PARA CLIMA	<i>¡Error! Marcador no definido.</i>
A.1.14 PANTALLAS DE GESTIÓN PARA ETNIA.....	<i>¡Error! Marcador no definido.</i>
A.2 MENÚ DE USUARIO WEB	<i>¡Error! Marcador no definido.</i>
A.2.1 INICIO.....	<i>¡Error! Marcador no definido.</i>
A.2.2 TODAS LAS ÁREAS.....	<i>¡Error! Marcador no definido.</i>
A.2.4 ¿QUIENES SOMOS?.....	<i>¡Error! Marcador no definido.</i>
A.2.5 ACERCA DE.....	<i>¡Error! Marcador no definido.</i>

A.3 PANTALLAS DE NAVEGACIÓN PARA EL USUARIO BARRA LATERAL..... ¡Error! Marcador no definido.

A.3.1 ÁREAS POR TIPO ¡Error! Marcador no definido.

A.3.2 MAPA DE ÁREAS PROTEGIDAS..... ¡Error! Marcador no definido.

A.3.3 ATRIBUTOS..... ¡Error! Marcador no definido.

A.4 diagramas físicos y lógicos del sistema ¡Error! Marcador no definido.

A.5. DICCIONARIO DE DATOS (VER EN EL ARCHIVO DIGITAL DESDE LA PÁGINA 201 A LA PÁGINA 325) ¡Error! Marcador no definido.

ÍNDICE DE GRÁFICOS

<i>Figura 1. 1 Información Disponible de áreas protegidas en SNAP.</i>	11
<i>Figura 2. 1Relación que debe existir entre los integrantes del proyecto</i>	23
<i>Figura 2. 2 Fase de Construcción de una Aplicación WEB.</i>	26
<i>Figura 2. 3Ejemplo de diagrama de secuencia</i>	32
<i>Figura 2. 4 Ejemplo de Modelo de Objetos, Modelo Navegacional, utilizados en el proceso de desarrollo orientado a la WEB</i>	35
<i>Figura 2. 5 Pirámide del diseño IWEB</i>	36
<i>Figura 2. 6 Tecnologías agrupadas bajo el concepto de AJAX</i>	53
<i>Figura 2. 7 Comparación Gráfica del Modelo tradicional de aplicación web y del nuevo modelo propuesto por AJAX.</i>	54
<i>Figura 2. 8 Comparación entre las comunicaciones síncronas de las aplicaiones web tradicionales y las comunicaciones asíncronas de las aplicaciones AJAX.</i>	55
<i>Figura 2. 9 Ejemplo de código de plantilla A.S.P.</i>	61
<i>Figura 3. 1 Menú Horizontal de navegación en el Web Site actual.</i>	63
<i>Figura 3. 2 Diagrama de Casos de uso para el Administrador del Sistema.</i>	66
<i>Figura 3. 3 Diagrama de casos de uso para Usuario dentro del Web Site.</i>	83
<i>Figura 3. 4 Diagrama de Casos de Uso de Usuario.</i>	99
<i>Figura 3. 5 Base de Datos del Sistema SNAP.</i>	100
<i>Figura 3. 6 Tabla con Listado de Historias de Usuario.</i>	109
<i>Figura 3. 7 Plan de Entrega</i>	110
<i>Figura 3. 8 TAREA 1 para la Iteración 1</i>	111
<i>Figura 3. 9 TAREA 2 para la Iteración 1</i>	112
<i>Figura 3. 10 TAREA 1 para la Iteración 2</i>	113
<i>Figura 3. 11 TAREA 2 para la Iteración 2</i>	114

<i>Figura 3. 12 TAREA 3 para la Iteración 2</i>	115
<i>Figura 3. 13 TAREA 4 para la Iteración 2</i>	116
<i>Figura 3. 14 TAREA 1 para la Iteración 3</i>	117
<i>Figura 3. 15 TAREA 2 para la Iteración 3</i>	118
<i>Figura 3. 16 TAREA 1 para la Iteración 4</i>	119
<i>Figura 3. 17 TAREA 1 para la Iteración 5</i>	120
<i>Figura 3. 18 TAREA 2 para la Iteración 5</i>	121
<i>Figura 3. 19 Tarjeta “Mapa Áreas Protegidas”</i>	122
<i>Figura 3. 20 Tarjeta “Base de Datos”</i>	123
<i>Figura 3. 21 Tarjeta “Página Web”</i>	124
<i>Figura 3. 22 Tarjeta “CD”</i>	125
<i>Figura 3. 23 Prototipo para pantalla de acceso</i>	126
<i>Figura 3. 24 Prototipo de las clases del Sistema.</i>	127
<i>Figura 3. 25 Prototipo de la página Web Principal.</i>	128
<i>Figura 3. 26 Prototipo De página Web con información de Áreas Protegidas</i>	129
<i>Figura 3. 27 Prototipo de Pantalla de Flora</i>	130
<i>Figura 3. 28 Prototipo de información sobre Flora.</i>	131
<i>Figura 3. 29 Prototipo Galería fotográfica de Flora</i>	131
<i>Figura 3. 30 Prototipo de Pantalla de Fauna.</i>	132
<i>Figura 3. 31 Prototipo de Galería Fotográfica de Fauna.</i>	132
<i>Figura 3. 32 Pantalla de Mysql Front, para generación de base de datos.</i>	133
<i>Figura 3. 33 Estado de la Importación.</i>	133
<i>Figura 3. 34 Base de Datos Creada en Mysql Front.</i>	134
<i>Figura 3. 35 Campos de la Base de datos creada en Mysql Front.</i>	134
<i>Figura 3. 36 Pantalla inicial del programa Crea Cod.</i>	135
<i>Figura 3. 37 Pantalla de registro para crear un nuevo proyecto.</i>	135
<i>Figura 3. 38 Pantalla de Conexión con la Base de Datos</i>	136
<i>Figura 3. 39 Pantalla de configuración de los campos de la Base de Datos.</i>	136
<i>Figura 3. 40 Pantalla de configuración de propiedades de los campos de la Base de Datos.</i>	137
<i>Figura 3. 41 Pantalla de configuración de propiedades de Tabla.</i>	137
<i>Figura 3. 42 Pantalla de Propiedades de Columna.</i>	138
<i>Figura 3. 43 Pantalla de Creación de Plantilla de Gestión de Información</i>	138
<i>Figura 3. 44 Pantalla de configuración de Controles.</i>	139
<i>Figura 3. 45 Pantalla Final del Proyecto Configurado.</i>	139
<i>Figura 3. 46 Pruebas del Sistema</i>	144

ÍNDICE DE FIGURAS DE ANEXOS

<i>Figura A. 1 Pantalla de Gestión para búsqueda de Áreas</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 2 Listado de búsqueda de Áreas al presionar botón Asterisco.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 3 Pantalla para crear una nueva Área.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 4 Pantalla para ingreso de Datos del Área nueva</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 5 Pantalla de búsqueda de tipos de Área</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 6 Pantalla que despliega el listado de los tipos de Áreas presionando el botón *</i>	<i>¡Error! Marcador no definido.</i>
<i>no definido.</i>	
<i>Figura A. 7 Pantalla para crear nuevo tipo de Área.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 8 Pantalla de Búsqueda de Flora</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 9 Listado de la Flora Existente</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 10 Pantalla para ingreso de datos de Nueva Flora.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 11 Pantalla para búsqueda de Familia de Flora o Fauna.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 12 Listado de Familias de Flora y Fauna Existentes</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 13 Pantalla para ingreso de Información de nueva familia de Flora o Fauna.</i>	<i>¡Error! Marcador no</i>
<i>definido.</i>	
<i>Figura A. 14 Pantalla de búsqueda de Fauna.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 15 Listado de Fauna existente.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 16 Pantalla de creación de nuevo ítem de Fauna.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 17 Pantalla de búsqueda de Geografía.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 18 Pantalla con listado de Geografía Existente.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 19 Pantalla para crear una nueva Geografía.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 20 Pantalla de búsqueda de Hidrografía.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 21 Listado de Hidrografía.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 22 Pantalla para crear nueva Hidrografía.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 23 Pantalla de Búsqueda de Turismo.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 24 Listado de Turismo.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 25 Pantalla para crear nuevo Turismo.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 26 Búsqueda de Tipo de Turismo.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 27 Listado de Tipo de Turismo.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 28 Pantalla para crear nuevo Tipo de Turismo</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 29 Pantalla de Búsqueda de Vestimenta.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 30 Listado de Vestimenta existente.</i>	<i>¡Error! Marcador no definido.</i>
<i>Figura A. 31 Pantalla para ingresar nueva vestimenta.</i>	<i>¡Error! Marcador no definido.</i>

Figura A. 32 Pantalla para búsqueda de Regiones para Vestimenta, Provincia, Clima, Etnia. **¡Error! Marcador no definido.**

Figura A. 33 Pantalla con lista de Regiones para Vestimenta, Provincia, Clima y Etnia. **¡Error! Marcador no definido.**

Figura A. 34 Pantalla para ingresar una nueva Región. **¡Error! Marcador no definido.**

Figura A. 35 Pantalla de Búsqueda de Provincia **¡Error! Marcador no definido.**

Figura A. 36 Listado de Provincias. **¡Error! Marcador no definido.**

Figura A. 37 Listado para Crear Nueva Provincia. **¡Error! Marcador no definido.**

Figura A. 38 Pantalla de Búsqueda de Clima. **¡Error! Marcador no definido.**

Figura A. 39 Listado de Clima. **¡Error! Marcador no definido.**

Figura A. 40 Pantalla para crear nuevo Clima. **¡Error! Marcador no definido.**

Figura A. 41 Pantalla de Búsqueda de Etnia. **¡Error! Marcador no definido.**

Figura A. 42 Listado de Etnias existente. **¡Error! Marcador no definido.**

Figura A. 43 Pantalla para ingreso de nueva Etnia. **¡Error! Marcador no definido.**

Figura A. 44 Pantalla de Inicio del Web Site **¡Error! Marcador no definido.**

Figura A. 45 Pantalla con listado de todas las Áreas. **¡Error! Marcador no definido.**

Figura A. 46 Pantalla de Área por tipo **¡Error! Marcador no definido.**

Figura A. 47 Pantalla con Información Sobre los desarrolladores del Sistema. **¡Error! Marcador no definido.**

Figura A. 48 Información sobre la Guía de Áreas Protegidas. **¡Error! Marcador no definido.**

Figura A. 49 Menú Lateral de la Guía Interactiva. **¡Error! Marcador no definido.**

Figura A. 50 Área Nacional de Recreación **¡Error! Marcador no definido.**

Figura A. 51 Parque Nacional **¡Error! Marcador no definido.**

Figura A. 52 Refugio de Vida Silvestre **¡Error! Marcador no definido.**

Figura A.53 Reserva Biológica **¡Error! Marcador no definido.**

Figura A.54 Reserva de Producción de Fauna **¡Error! Marcador no definido.**

Figura A.55 Reserva Ecológica **¡Error! Marcador no definido.**

Figura A.56 Reserva Geobotánica **¡Error! Marcador no definido.**

Figura A.57 Reserva Marina **¡Error! Marcador no definido.**

Figura A.58 Mapa de Áreas Protegidas **¡Error! Marcador no definido.**

Figura A.59 Barra Lateral Atributos **¡Error! Marcador no definido.**

Figura A.60 Flora **¡Error! Marcador no definido.**

Figura A.61 Fauna **¡Error! Marcador no definido.**

Figura A.62 Geografía **¡Error! Marcador no definido.**

Figura A.63 Hidrografía **¡Error! Marcador no definido.**

Figura A.64 Turismo **¡Error! Marcador no definido.**

Figura A.65 Vestimenta. **¡Error! Marcador no definido.**

Figura A.66 Provincia **¡Error! Marcador no definido.**

Figura A.67 Clima

¡Error! Marcador no definido.

Figura A.68 Etnia

¡Error! Marcador no definido.

Figura A. 69 Modelo Lógico del Sistema

¡Error! Marcador no definido.

Figura A. 70 Modelo Físico del Sistema parte 1

¡Error! Marcador no definido.

Figura A. 71 Modelo Físico del Sistema parte 2

¡Error! Marcador no definido.

Figura A. 72 Modelo Físico del Sistema parte 3

¡Error! Marcador no definido.

RESUMEN

El Ministerio del Ambiente, basándose en la problemática actual de la protección y mantenimiento de la diversidad biológica, así como de recursos naturales y culturales asociados; decide insertar un programa de culturización sobre la preservación de recursos naturales que conforman las Áreas Protegidas.

Un área protegida es una superficie de tierra y/o mar especialmente consagrada a la protección y el mantenimiento de la diversidad biológica, así como de recursos naturales y los recursos culturales asociados, y manejada a través de medios jurídicos u otros medios eficaces.

Actualmente el Ministerio del Ambiente no cuenta con un sistema o Guía informativa de estas áreas, siendo de gran interés por parte de sus autoridades el generar una Guía interactiva para difusión nacional e internacional de éstas riquezas, por este motivo, se ha previsto diseñar un sistema que permita enseñar y conocer más sobre las Áreas Protegidas del Ecuador.

Este proyecto de Tesis, tiene como objetivo principal analizar, diseñar desarrollar e implementar una Guía sobre Áreas Protegidas que permitan a ecuatorianos y extranjeros conocer más a fondo sobre el tema de Áreas protegidas y todo lo que ello abarca con el fin de difundir las riquezas que se quieren conservar de Ecuador.

El proyecto consta de dos secciones, la primera de Gestión de la información, la misma que será manejada por un administrador, el cual se encargará de añadir, eliminar, o cambiar la información referente a Áreas Protegidas del Sistema rSnap. La siguiente sección es la página Web, en la cual el usuario Web, podrá interactuar y utilizar la información contenida en el sitio.

Palabras Clave:

- Área Protegida , Programación Extrema , Snap , Área Nacional de Recreación , Parque Nacional , Refugio de Vida Silvestre , Reserva Biológica , Reserva de Producción de Fauna , Reserva Ecológica , Reserva Geobotánica , Reserva Marina.

ABSTRACT

The Ministry of Environment based on the current problems of the protection and maintenance of biological diversity and of natural and cultural resources associated; decides to insert a program of acculturation on the preservation of natural resources that make up the protected ÁREAs.

A protected ÁREA is an ÁREA of land and / or sea especially dedicated to the protection and maintenance of biological diversity and of natural and associated cultural resources, and managed through legal or other effective means.

Currently the Ministry of Environment has no guidance system or information of these ÁREAs is of great interest from the authorities create an interactive guide for national and international dissemination of these riches, for this reason, it is planned to design a system allowing teach and learn more about the protected ÁREAs of Ecuador.

This thesis project has as main objective to analyze, design, develop and implement guidance on Protected ÁREAs that allow Ecuadorians and foreigners better understand the issue of protected ÁREAs and all that it encompasses to spread the wealth are to be retained in Ecuador.

The project consists of two sections, the first of Information Management, the same that will be managed by an administrator, who is responsible for adding, deleting, or changing information concerning rSnap Protected ÁREAs System. The next section is the Web page in which the Web user, you can interact and use the information contained on the site.

Keywords: Protected ÁREA, Extreme Programming, Snap, National Recreation ÁREA, National Park, Wildlife Refuge, Biological Reserve, Fauna Production Reserve, Ecological Reserve, Reserve Geobotanical, Marine Reserve.

CAPÍTULO 1

GENERALIDADES

1.1 TÍTULO

“Análisis, diseño, construcción e implementación de una Guía Interactiva y sitio Web para las áreas protegidas del Ecuador”

1.2 INTRODUCCIÓN

Un área protegida es una superficie de tierra y/o mar especialmente consagrada a la protección y el mantenimiento de la diversidad biológica, así como de recursos naturales y los recursos culturales asociados, y manejada a través de medios jurídicos u otros medios eficaces.

Las áreas protegidas del Ecuador son actualmente 44, y están repartidas en todo el territorio, siendo la región amazónica la que más las abarca.

Actualmente el Ministerio del Ambiente no cuenta con un sistema o Guía informativa de estas áreas, siendo de gran interés por parte de sus autoridades el generar una Guía interactiva para difusión nacional e internacional de éstas riquezas.

1.3 PLANTEAMIENTO DEL PROBLEMA

El Ministerio del Ambiente basándose en la problemática actual de la protección y el mantenimiento de la diversidad biológica, así como de recursos naturales y los recursos culturales asociados; decide realizar un programa de culturización sobre la preservación de los recursos naturales de nuestro país en especial de aquellos que conforman las Áreas Intangibles.

Debido a la riqueza existente tanto en flora, fauna y riqueza cultural se debe realizar una detenida selección de cada área asociada con su riqueza.

Se cree indispensable la creación de una Guía interactiva de las áreas protegidas con el fin de familiarizar a turistas nacionales y extranjeros con las maravillas que pueden encontrar en nuestro país y concienciar a la juventud a conservar los valiosos tesoros los cuales no se preservan y se perderían.

1.4 JUSTIFICACIÓN

El Ministerio del Ambiente en su afán de lograr la conservación de la variedad de especies nativas de nuestro país y preocupándose en concienciar a la juventud cree prudente la creación de esta Guía sobre Áreas Protegidas como un aporte a las riquezas del país.

Los beneficios obtenidos serán:

- Material didáctico para estudiantes a nivel Nacional e internacional.
- Promoción de la Guía de Áreas Naturales a otras culturas.
- Autogestión para seguir realizando los programas de conservación de las especies.
- Mejora de la imagen del Ecuador a nivel internacional

1.5 OBJETIVOS

1.5.1 OBJETIVO GENERAL

- Desarrollar e implementar una Guía Interactiva sobre Gestión de Áreas protegidas del Ecuador para el Ministerio del Ambiente y su departamento de Conservación de Áreas Protegidas SNAP.

1.5.2 OBJETIVOS ESPECÍFICOS

- Describir la situación actual del Sitio Web del Ministerio del Ambiente con respecto al manejo de información sobre áreas protegidas.
- Analizar y diseñar el Sitio Web y la Guía Aplicando la Metodología XP e Ingeniería WEB.
- Construir el Sitio Web y la Guía aplicando la Metodología XP e Ingeniería WEB.
- Realizar pruebas de la Aplicación correspondiente.

1.6 ALCANCE

Se ve necesaria la creación de una Guía Interactiva para Áreas protegidas del Ecuador que maneje información cultural, educativa y científica de interés nacional e internacional, que permita dar a conocer las riquezas del país.

El desarrollo de la Guía de Áreas Protegidas se lo realizará con herramientas Multimedia de libre difusión, inicialmente se ha creído conveniente desarrollar una Guía que permita el levantamiento de información de especies de flora, fauna y demás riquezas que comprende un Área Protegida.

El sistema contemplará los siguientes elementos:

- La estructura o metodología utilizada es XP (Xtreme Programming), la cual permite agilidad en el desarrollo de Aplicaciones.
- Ingeniería Web compuesta por componentes orientados a objetos y componentes convencionales (estructurales).
- El sistema será híbrido, contemplando para su desarrollo los componentes:
 - Orientados a objetos
 - PHP y clases PHP
 - Convencionales
 - CSS
 - Javascript
 - Agentes Truvoices y CSAgents

Para el levantamiento del sitio Web se utilizará como lenguaje de programación a PHP, con instrucciones en lenguaje Javascript y agentes de ayuda.

Para el desarrollo de la Guía interactiva, se transformará el código PHP a lenguaje HTML, utilizando además los agentes y lenguaje Javascript.

1.7 METODOLOGÍA

Para el desarrollo de la Guía de Áreas Protegidas y su sitio WEB, se ha optado por seguir las normas del modelo XP junto con Ingeniería WEB.

“La ingeniería Web se la puede definir como el proceso utilizado para crear, implantar y mantener aplicaciones y sistemas Web de alta calidad. Esta breve definición lleva a abordar un aspecto clave de cualquier proyecto como es determinar qué tipo de proceso es más adecuado en función de las características del mismo”¹.

1.8 HERRAMIENTAS

1.8.1 HARDWARE

- Computador de prestaciones actuales:
- Intel Corei5 1.8 Ghz
- 3 Gb Memoria RAM
- 100 Gb Disco duro
- Monitor 15” Widescreen
- Computador Servidor de prestaciones actuales:
- Las características del servidor estarán a cargo del Ministerio del Ambiente.

¹hraulvargasq.blogspot.com/2009/06/ingenieria-web.htm

1.8.2 SOFTWARE

- Microsoft Windows XP, versión Professional
- Macromedia Dreamweaver CS3
- Microsoft Office 2007
- Power Designer 6 Data Architect
- MySQL 5
- Apache y PHP
- CD Autorun Creator
- GetRight

1.9 FACTIBILIDAD

1.9.1 OPERATIVA

La factibilidad de este proyecto se inicia con la necesidad existente en el Ministerio del Ambiente de tener una Guía sobre Áreas Protegidas, dentro de las cuales constan : Parques Nacionales, Reservas Ecológicas, Reservas Biológicas, Reservas Marinas, Parques Nacionales de Recreación, la cual permita difundir este tema, tanto en un sitio web y con un CD interactivo; el mismo que servirá para enseñanza Local y para envíos al extranjero para las personas que no puedan acceder al Internet; y la elaboración de un renovado sitio Web para el SNAP (Sistema Nacional de Áreas Protegidas) del Ministerio del Ambiente en que manejará información de las áreas protegidas.

Esta Guía será muy útil ya que permitirá dar un aporte valioso que mejorará los conocimientos e la juventud y la difusión de este tema a nivel internacional.

Este proyecto está auspiciado en su totalidad por el Ministerio del Ambiente y su programa de Áreas Protegidas SNAP.

1.10 FONDO DE ÁREAS PROTEGIDAS

1.10.1. ORÍGENES

El fondo ambiental Nacional (FAN) es una institución privada, sin fines de lucro, que nace del interés de un grupo de líderes de diferentes sectores de la sociedad ecuatoriana (gobierno, sector productivo, académico, y organizaciones no gubernamentales ambientalistas), con el fin de identificar un mecanismo idóneo para apoyar al financiamiento de la gestión ambiental.

1.10.2. MISIÓN

Apoyar Al financiamiento de la gestión ambiental tendiente al desarrollo sustentable del Ecuador.

1.10.3. VISIÓN

El Fondo Ambiental Nacional es una organización líder en el diseño e implementación de estrategias y mecanismos financieros, y se constituye en un punto de encuentro y concertación de voluntades y acciones que apoyan la gestión ambiental y la conservación del Ecuador en el marco del desarrollo sustentable.

1.10.4. CARACTERÍSTICAS PRINCIPALES

- El Fondo Ambiental Nacional:
- Administra y canaliza recursos.
- Apoya al financiamiento de las prioridades ambientales del país, por medio de una alianza estratégica con el Ministerio del Ambiente del Ecuador.
- Es un mecanismo voluntario para la canalización de recursos desde identidades públicas o privadas, nacionales o internacionales.
- Posee un Directorio multisectorial y multidisciplinario, que asegura la participación en la toma de decisiones de diferentes actores.

1.10.5. PRINCIPIOS Y VALORES

Las actividades del FAN se orientan a generar y fortalecer cambios hacia el desarrollo sustentable mediante los siguientes principios y valores institucionales:

1. Planificar actividades que correspondan a políticas ambientales del país.
2. Generar y fortalecer procesos participativos mediante alianzas estratégicas entre diferentes actores, en un marco de equidad y respeto.
3. Enfocar acciones en el más amplio concepto de gestión ambiental, realizando una distribución justa y equitativa de sus costos y beneficios.
4. Realizar acciones, procesos e inversiones, con la debida responsabilidad ambiental y social.
5. Realizar con integridad, transparencia y eficiencia todos los procesos y una adecuada rendición de cuentas de la gestión institucional.
6. Lograr la excelencia en la gestión, caracterizada por el compromiso, profesionalismo y calidad humana de directivos y Equipo Técnico en un proceso de mejoramiento continuo.

1.11 CENTRO DE INFORMACIÓN AMBIENTAL

1.11.1. OBJETIVOS

1. Administrar e integrar la información ambiental del Ecuador.
2. Recopilar, sistematizar y difundir información sobre biodiversidad del Ecuador en general; bosques y vegetación protectores; calidad ambiental en el país; indicadores ambientales y de biodiversidad.
3. Desarrollar herramientas que permitan la integración y difusión de la información ambiental a través de medios escritos, electrónicos como internet, imágenes, mapas, y otros afines.
4. Facilitar información espacial y alfanumérica hacia dentro del MAE así como a entidades públicas y privadas para facilitar la toma de decisiones en el campo ambiental.

5. Promover mecanismos de estandarización en la información ambiental, como medio de interrelacionar a los sectores civiles, privados, grupos comunitarios y gubernamentales.
6. Desarrollar, coordinar, y ejecutar programas de capacitación para el Ministerio del Ambiente, especialmente en el campo de manejo de bases de datos, sistemas de información geográficos, y sensores remotos. El desarrollo de estas actividades podrá ser cumplida con los recursos propios del Ministerio o la colaboración de personal fuera del Ministerio bajo la coordinación de los técnicos del CIAM.

1.11.2 FUNCIONES

1. Asesorar a las Autoridades del Ministerio del Ambiente en materia de información ambiental para la toma de decisiones.
2. Asegurar que la información ambiental contribuya a la conservación y uso sostenible de la biodiversidad, al manejo del recurso forestal, al monitoreo de la calidad ambiental del país
3. Optimizar el uso de la cooperación internacional y fortalecer la capacitación a los funcionarios del Ministerio, en el manejo de la información ambiental.
4. Participar en la elaboración de la Estrategia Nacional de Ordenamiento Territorial y los planes seccionales, con las demás dependencias del Ministerio del Ambiente y Entidades Nacionales involucradas en éste proceso.

1.11.3. INFORMACIÓN DISPONIBLE

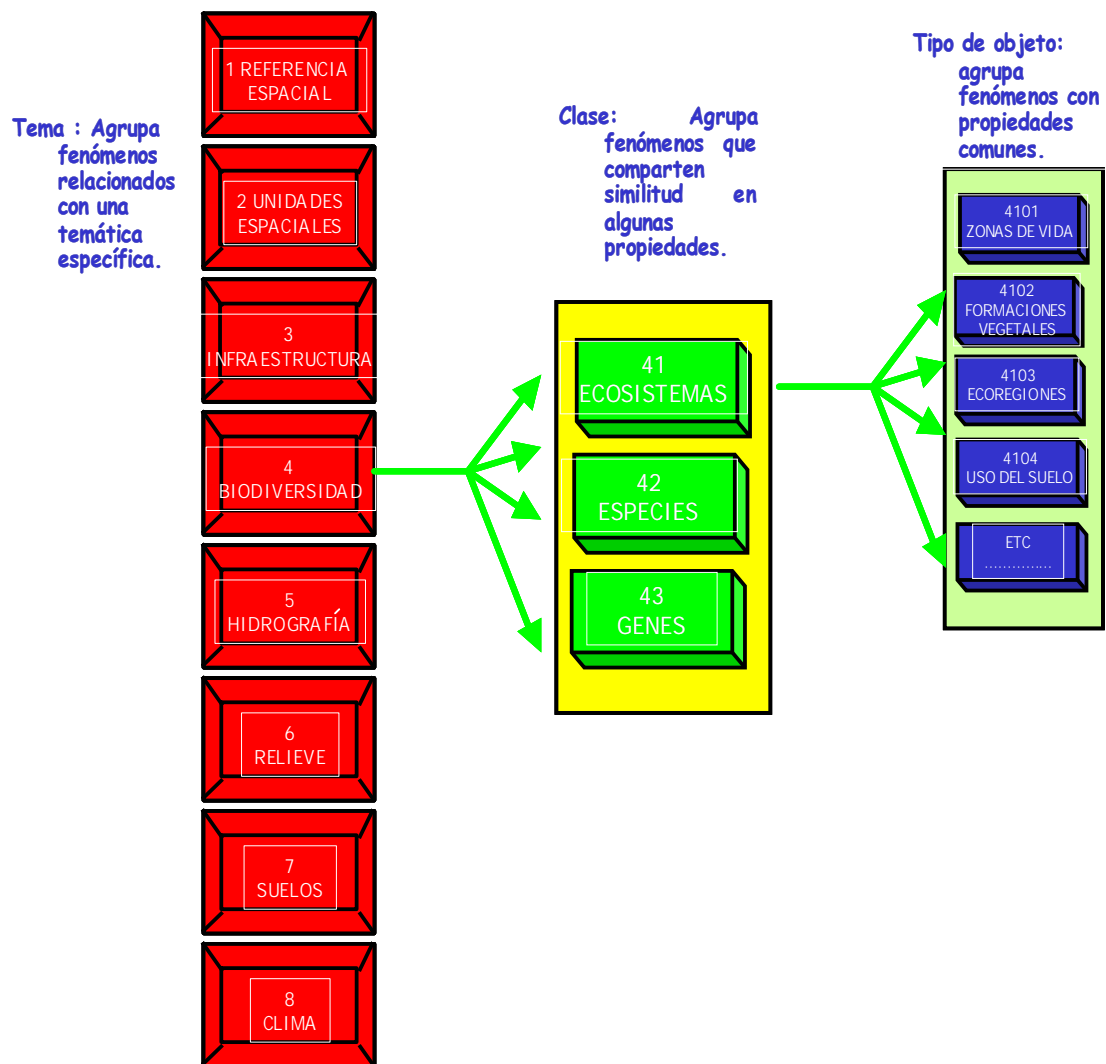


Figura 1. 1 Información Disponible de áreas protegidas en SNAP.

La información disponible se encuentra estratificada (figura 1.1) por:

Temas que agrupa fenómenos desarrollados con una temática específica. Los temas se encuentran agrupados en clases, las mismas que comparten similitudes en algunas propiedades; y a su vez las clases se encuentran asociadas a objetos que agrupan a fenómenos que poseen propiedades iguales.

CAPÍTULO 2

MARCO TEÓRICO

2.1 CONCEPTOS BÁSICOS

2.1.1. 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”².

2.1.2. SOFTWARE LIBRE

“El software libre (en inglés free software, aunque esta denominación también se confunde a veces con "gratis" por la ambigüedad del término "free" en el idioma inglés, por lo que también se usa "libre software" y "logical libre") es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado, y redistribuido libremente”³.

2.1.3. OPEN SOURCE

“Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de poder acceder al código, que a las cuestiones éticas y morales las cuales se destacan en el software libre”⁴.

²es.wikipedia.org/wiki/Software

³puntoabierto.net/wp.../03/material_softwarelibre_ext.pdf

⁴www.sanmarcux.org/index.php/software-libre/39.../core

2.1.4. LENGUAJES DE PROGRAMACIÓN

Un lenguaje de programación es un idioma artificial diseñado para expresar procesos que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación”⁵.

2.2 EXTREME PROGRAMMING (XP)

“La Programación Extrema (PX), mejor conocida por su nombre en inglés Extreme Programming (XP), es una de las llamadas Metodologías Ágiles de desarrollo de software más exitosas de los tiempos recientes”⁶.

2.2.1 ALGUNAS BASES DE LA PROGRAMACIÓN EXTREMA

La Programación extrema inicia, porque es imposible realizar un análisis tan detallado que capture toda la información antes de empezar a programar.

Al iniciar el desarrollo, se presentan más requerimientos por parte del usuario y se comienzan a detectar los errores cometidos en el diseño, esto genera que se pierda tiempo en realizar la nueva documentación y rediseño del Sistema.

⁵www.buenasTAREAs.com › Inicio › Tecnología

⁶ Extreme Programming (XP) – AnOverview
ExtremeProgramming.aspx.htm

http://www.c-sharpcorner.com/UploadFile/sakthivel_a/ExtremeProgramming11282005054653AM/ExtremeProgramming.aspx

En lugar de desechar todo lo ya realizado, se busca una forma adecuada para aprovechar las partes buenas y desechar de manera fácil las partes que no se ajustan al sistema.

“Básicamente la idea de la programación extrema consiste en trabajar estrechamente con el cliente, haciéndole mini-versiones con mucha frecuencia (cada dos semanas). En cada mini-versión se debe hacer el mínimo de código y lo más simple posible para que funcione correctamente. El diseño se hace sobre la marcha, haciendo un mini-diseño para la primera mini-versión y luego modificándolo en las siguientes mini-versiones. Además, no hay que hacer una documentación para el diseño, no hay mejor documentación que el mismo código. El código, por tanto, también se modifica continuamente de mini-versión en mini-versión, añadiéndole funcionalidad y extrayendo sus partes comunes”⁷.

2.2.3 LOS PASOS A SEGUIR EN UN PROYECTO

En un proyecto usando programación extrema se siguen los siguientes pasos:

El desarrollador junto con el cliente definen que es lo que se requiere hacer para la Aplicación, esto se realiza mediante "historias de usuario". “Una historia de usuario en un texto de una o dos frases en las que se dice algo que debe hacer el sistema. Es más extensa que un requisito (que suele ser una frase corta) y menos que un caso de uso (que puede ser de una o dos páginas). Se evalúa para cada historia de usuario el tiempo que puede llevar, que debe ser corto, de aproximadamente una semana. Un programador puede estimar con cierta fiabilidad un trabajo que le lleve unos días, pero la estimación es menos fiable si es de un plazo superior a una semana. Si es más largo, hay que partir la historia en otras más pequeñas. Luego se ordenan en el orden en que se van a desarrollar y se establecen las mini-versiones, de forma que cada mini-versión implementa varias de las historias de usuario”⁸.

⁷ www.chuidiang.com/ood/metodologia/extrema.php

⁸ www.chuidiang.com/ood/metodologia/extrema.php

2.2.4 PRÁCTICAS BÁSICAS DE LA PROGRAMACIÓN EXTREMA

“Para que todo esto funcione, la programación extrema se basa en trece "prácticas básicas" que deben seguirse al pie de la letra.

1. Equipo completo: Forman parte del equipo todas las personas que tienen algo que ver con el proyecto, incluido el cliente y el responsable del proyecto.
2. Planificación: Se hacen las historias de usuario y se planifica en qué orden se van a hacer y las mini-versiones. La planificación se revisa continuamente.
3. Test del cliente: El cliente, con la ayuda de los desarrolladores, propone sus propias pruebas para validar las mini-versiones.
4. Versiones pequeñas: Las mini-versiones deben ser lo suficientemente pequeñas como para poder hacer una cada pocas semanas. Deben ser versiones que ofrezcan algo útil al usuario final y no trozos de código que no pueda ver funcionando.
5. Diseño simple: Hacer siempre lo mínimo imprescindible de la forma más sencilla posible. Mantener siempre sencillo el código.
6. Pareja de programadores: Los programadores trabajan por parejas (dos delante del mismo ordenador) y se intercambian las parejas con frecuencia (un cambio diario).
7. Desarrollo guiado por las pruebas automáticas: Se deben realizar programas de prueba automática y deben ejecutarse con mucha frecuencia. Cuantas más pruebas se hagan, mejor.
8. Mejora del diseño: Mientras se codifica, debe mejorarse el código ya hecho con el que se tiene y que sea susceptible de ser mejorado. Extraer funcionalidades comunes, eliminar líneas de código innecesarias, etc.
9. Integración continua: Deben tenerse siempre un ejecutable del proyecto que funcione y en cuanto se tenga una nueva pequeña funcionalidad, debe recompilarse y probarse. Es un error mantener una versión congelada dos meses mientras se hacen mejoras y luego integrarlas todas de golpe. Cuando falle algo, no se sabe qué es lo que falla de todo lo que se haya metido.

10. El código es de todos: Cualquiera puede y debe tocar y conocer cualquier parte del código. Para eso se hacen las pruebas automáticas.
11. Normas de codificación: Debe haber un estilo común de codificación (no importa cual), de forma que parezca que ha sido realizado por una única persona.
12. Metáforas: Hay que buscar unas frases o nombres que definan cómo funcionan las distintas partes del programa, de forma que sólo con los nombres se pueda uno hacer una idea de qué es lo que hace cada parte del programa. Un ejemplo claro es el "recolector de basura" de java. Ayuda a que todos los programadores (y el cliente) sepan de qué se esta hablando y que no haya mal entendidos.
13. Ritmo sostenible: Se debe trabajar a un ritmo que se pueda mantener indefinidamente. Esto quiere decir que no debe haber días muertos en que no se sabe qué hacer y que no se deben hacer un exceso de horas otros días. Al tener claro semana a semana lo que debe hacerse, hay que trabajar duro en ello para conseguir el objetivo cercano de terminar una historia de usuario o mini-versión⁹.

2.3 INTRODUCCIÓN A LAS APLICACIONES HIPERMEDIA¹⁰

“El término HIPERMEDIA, combinación de los conceptos HIPERtexto y multiMEDIA, hace referencia a una tecnología de construcción de hiperdocumentos que permite a los lectores encontrar fácilmente la información que realmente necesitan, de la manera que ellos decidan, a través de enlaces establecidos por el autor entre los diferentes elementos de información multimedia (texto, sonido, imagen, vídeo, etc.) que conforman el documento”.

“El término Hipertexto se refiere a un documento hipertexto donde todos los elementos de información contienen únicamente texto. También suele confundirse con el término Multimedia, cuando éste, en realidad, hace referencia a sistemas que contienen y presentan texto, imágenes, sonido, video, etc. pero sin enlaces entre estos elementos de información”.

⁹ www.chuidiang.com/ood/metodologia/extrema.php

¹⁰ biblo.una.edu.ve/docu.7/bases/anali/texto/Sirviente.pdf

“Un hiperdocumento se compone, en principio, de dos tipos de elementos: por un lado estarían los nodos, o "contenedores" de la información multimedia; y por otro los enlaces, que interconectan los nodos permitiendo otras alternativas de navegación por la información diferentes del clásico acceso secuencial "desde la primera a la última línea". Existe, no obstante, un tercer componente fundamental denominado ancla. Se trata de un mecanismo para señalar puntos incluidos en el interior de los nodos que sirven de origen o de destino a un determinado enlace entre nodos.”

“Los Sistemas Hipermediales son los entornos que ofrecen a los usuarios todos los mecanismos para la creación, manipulación y consulta de hiper documentos. “

“Debido a la complejidad evidente de los documentos hipermedia, desde los orígenes de esta tecnología se ha intentado establecer un modelo universal de hiper documento que permita su percepción desde diferentes niveles de abstracción para facilitar el desarrollo de estándares de interface entre niveles que garanticen la portabilidad de los documentos generados. “

“De todo lo anterior se evidencian algunas consideraciones importantes a tener en cuenta respecto a la tecnología hipermedia que pueden aconsejar la incorporación del paradigma de la orientación a objetos en los diferentes aspectos de dicha tecnología:

- La información que se maneja en los sistemas hipermediales es muy heterogénea, lo cual es evidente debido al componente multimedia en la tecnología.
- Los hiper documentos se estructuran en bloques documentales (nodos) autónomos a los que se accede a través de enlaces. Se va a requerir un gran espacio de almacenamiento para la información. Se pueden crear hiperespacios de navegación muy complejos debido a las posibilidades de interconexión de nodos.
- El usuario es el elemento más importante de un sistema hipermedial. De tal forma que si se le ignora durante el proceso de diseño del hiperdocumento es muy probable nunca lo utilice si no satisface plenamente sus necesidades de información.

- También hay que cuidar especialmente la interface del sistema con el usuario para hacerlo atractivo. “

“Los recientes modelos que están desarrollándose, incorporan el paradigma de la orientación a objetos con el fin de mejorar las prestaciones de los sistemas de hipertexto e hipermedia. Esto lo hacen mediante el uso de Sistemas de Gestión de Bases de Datos Orientados a Objeto (SGBDOO) para almacenar la información heterogénea, aplicando alguna norma estándar para estructurar el contenido de un hiperdocumento y adoptando un enfoque de ingeniería de software con el fin de diseñar un modelo previo que recoja las necesidades de los usuarios. Un modelo orientado a objetos permite una representación gráfica del hipertexto para representar la estructura estática de la información, un modelo dinámico para los aspectos temporales del comportamiento del sistema y un modelo funcional para representar los procesos que transforman la información del sistema. Lo normal es utilizar software de autor o herramientas de edición para crear hipertextos, pero es preciso antes un análisis conceptual tanto de los elementos estructurales, como de la navegación y de la interfaz que se le presentará al usuario”¹¹.

¹¹edutec2004.lmi.ub.es/pdf/42.pdf

2.4. HIPERMEDIA Y BASES DE DATOS ORIENTADAS A OBJETOS

“Los Sistemas de Gestión de Bases de Datos tradicionales, y en particular los relacionales, adolecen de falta de capacidad semántica y de soporte para la abstracción estructural necesarios para el almacenamiento de la información hipermedia”. Así, un SGBDOO (Sistema de Gestión de Base de Datos Orientada a Objetos) sería el modelo ideal para el nivel de almacenamiento de hiperdocumentos, ya que podría soportar los requisitos de estos sistemas, como la necesidad de almacenamiento de información multimedia no estructurada (secuencias de vídeo, sonidos, imágenes, etc.) y el establecimiento de enlaces complejos entre objetos documentales”¹².

2.4.1. METODOLOGÍAS DE DESARROLLO DE SISTEMAS HIPERMEDIALES.

Habitualmente el desarrollo de Sistemas Hipermediales se realiza con la utilización directa de herramientas de autor a nivel de implementación, descuidando el proceso previo de análisis y diseño abstracto de los aspectos estructurales, de navegación y de interfaz con el usuario, dejando de esta forma fuera de la documentación, haciendo complicado el mantenimiento futuro del sistema.

La nueva tendencia en programación es la de desarrollar basándose en el sistema hipermedial con un enfoque de proceso de ingeniería de software, por lo que ya se han propuesto diferentes metodologías, como HDM o RMM, que establecen la necesidad de considerar un diseño previo a la construcción del sistema.

En la actualidad, han empezado a aparecer metodologías que asumen la orientación a objetos como paradigma de diseño al igual que ha ocurrido en el ámbito de la Ingeniería del Software. “En el caso hipermedia, el enfoque OO (Orientado a Objetos) en los modelos es muy útil debido al gran nivel de abstracción que ofrece y a sus mecanismos de composición (generalización,

¹²openaccess.uoc.edu/webapps/o2/.../35026tfc.pdf - España

clasificación y agregación) que facilitan el modelado de la estructura hipermedial. Esto queda reflejado en las dos metodologías más conocidas de desarrollo orientado a objetos de sistemas hipermediales como son EORM y OOHDM.”¹³

2.5. INGENIERÍA WEB ¹⁴

La Ingeniería de la Web es la aplicación de metodologías que permiten un alto desempeño Web.

“La Ingeniería de la Web hace referencia a las metodologías, técnicas y herramientas que se utilizan en el desarrollo de Aplicaciones Web complejas y de gran dimensión en las que se apoya la evaluación, diseño, desarrollo, implementación y evolución de dichas aplicaciones.

La Ingeniería de la Web es multidisciplinar y aglutina contribuciones de diferentes áreas: arquitectura de la información, ingeniería de hipermedia/hipertexto, ingeniería de requisitos, diseño de interfaz de usuario, usabilidad, diseño gráfico y de presentación, diseño y análisis de sistemas, ingeniería de software, ingeniería de datos, indexado y recuperación de información, testeo, modelado y simulación, despliegue de aplicaciones, operación de sistemas y gestión de proyectos.

Los sistemas y aplicaciones basadas en Web ofrecen un completo arreglo de contenido y funcionan a una amplia población de usuarios finales. La ingeniería Web es el proceso con el que se crean las WebApps de alta calidad (Aplicaciones Web)”¹⁵.

La Ingeniería Web no intenta copiar a la ingeniería de software, pero utiliza conceptos y principios fundamentales como desarrollo de software y programación.

¹³ repositorio.espe.edu.ec/bitstream/21000/401/1/T-ESPE-025005.pdf

¹⁴ Definición de Ingeniería de la Web extraída de webengineering.org

¹⁵ es.wikipedia.org/wiki/Ingeniería_web

“Los principales aspectos de la Ingeniería de la Web incluyen, entre otros, los siguientes temas”¹⁶ :”

- Diseño de procesos de negocio para aplicaciones Web
- Herramientas CASE para aplicaciones Web
- Generación de código para aplicaciones Web
- Desarrollo Web colaborativo
- Modelado conceptual de aplicaciones Web
- Diseñó de Modelos de datos para sistemas de información Web
- Ingeniería Web empírica
- Entornos de desarrollo de aplicaciones Web integrados
- Herramientas de autor para contenido multimedia
- Pruebas de rendimiento de aplicaciones basadas en Web
- Personalización y adaptación de aplicaciones Web
- Modelado de procesos para aplicaciones Web
- Herramientas y métodos de prototipado
- Control de calidad y pruebas de sistemas
- Ingeniería de requisitos para aplicaciones Web
- Aplicaciones para la Web Semántica
- Factorías de software para la Web
- Métodos, herramientas y automatización de pruebas para aplicaciones Web
- Aplicaciones Web móviles y ubicuas
- Usabilidad de aplicaciones Web
- Accesibilidad para la Web
- Metodologías de diseño Web
- Formación en ingeniería de la Web
- Diseño de interfaces de usuario
- Métricas para la Web, estimación de costes y medición
- Gestión de proyectos Web y gestión de riesgos
- Desarrollo y despliegue de servicios Web”

¹⁶**Pressman Roger S.** Ingeniería del Software Un enfoque práctico, Capítulo I, Aplicación de la Ingeniería Web.

2.5.1 ESTRATOS DE LA INGENIERÍA WEB

“El desarrollo de aplicaciones Web incorpora métodos de proceso especializados, métodos de ingeniería de software adaptados a características de desarrollo de las aplicaciones Web y un conjunto de importantes tecnologías que permitan un correcto desarrollo de las mismas.

Los procesos, métodos y tecnologías (herramientas) proporcionan un enfoque en estratos de la IWeb (Ingeniería Web) que es conceptualmente idéntico a los estratos de la ingeniería de software”¹⁷.

2.5.2 PROCESO

“Los modelos de procesos Web adoptan la filosofía de desarrollo ágil, enfatizando un enfoque de desarrollo riguroso que incorpora rápidos ciclos de desarrollo.

El problema todavía debe analizarse, debe desarrollarse un diseño, la implementación debe preceder en una forma incremental y se debe iniciar un enfoque organizado de prueba. Dichas actividades de marco de trabajo se deben definir dentro de un proceso que:

- Adopte el cambio
- Aliente la creatividad y la independencia del equipo de desarrollo y fortalezca la interacción con el usuario.
- Construya sistemas que utilicen pequeños equipos de desarrollo.
- Subraye el desarrollo evolutivo o incremental mediante el uso de cortos ciclos de desarrollo”¹⁸.

2.5.3. MÉTODOS

Los métodos utilizados para la IWeb abarcan un conjunto de labores técnicas que permiten al ingeniero Web comprender, caracterizar y luego construir una aplicación Web de alta calidad.

¹⁷www.scribd.com/doc/.../Estratos-de-la-ingenieria-web

¹⁸www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

Se clasifican de la siguiente manera:

Métodos de comunicación: Definen el enfoque con que se facilita la comunicación entre, los integrantes del equipo figura 3:

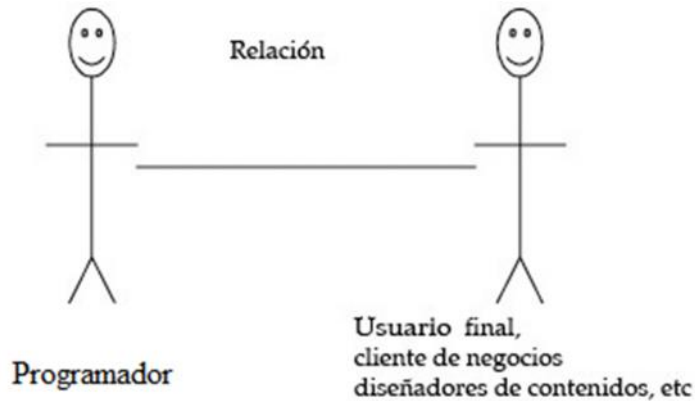


Figura 2. 1 Relación que debe existir entre los integrantes del proyecto

Las técnicas de comunicación son importantes durante la recolección de requisitos.

Métodos de análisis de requisitos: Se aplica este método, para obtener bases que permitan comprender el contenido que debe estar en la Aplicación Web, las funciones que tendrá el usuario final y los modos de navegación de cada clase de usuario por la aplicación.

Métodos de diseño: Incluyen varias "técnicas de diseño que abordan el contenido, la aplicación y la arquitectura de información, así como el diseño de interfaz y la estructura de navegación de las Aplicaciones Web (WebApps)"¹⁹.

Método de prueba: "Incorporan revisiones técnicas formales tanto de contenido y el modelo de diseño como de una amplia variedad de técnicas de pruebas que abordan conflictos al nivel de componentes. Pruebas de la navegación, pruebas de facilidad de uso, pruebas de seguridad y pruebas de configuración"²⁰.

¹⁹www.scribd.com/doc/.../Estratos-de-la-ingenieria-web

²⁰www.scribd.com/doc/.../Estratos-de-la-ingenieria-web

2.5.4 HERRAMIENTAS Y TECNOLOGÍAS

“Las tecnologías abarcan un amplio conjunto de descripción de contenido y lenguaje de modelación por ejemplo: HTML, VRML, XML, etc. lenguajes de programación por ejemplo java, php, jsp, etc. recursos de desarrollo basados en componentes por ejemplo corba, com, activeX, .net, etc. navegadores, herramientas multimedia, herramientas de auditoría de sitio, herramientas de conectividad de base de datos, herramientas de seguridad, servidores y utilidades de servidor, y herramientas de administración y análisis de sitio”²¹.

2.5.5 EL PROCESO DE INGENIERÍA WEB

Es un proceso que el tiempo de desarrollo y ciclos de vida de los productos para un proyecto son muy cortos y por su complejidad se vuelven imprescindibles.

Los más destacados aspectos que añaden dificultad a la gestión son:

- Alto porcentaje de contratación a nuevo personal,
- Para el desarrollo se requiere personal técnico y usuarios (personal no técnico) trabajando en conjunto,
- El equipo de desarrollo es difícil de conseguir ya que se requiere que el personal debe estar en capacidad para manejar diversos temas como: software basado en componentes, redes, diseño de arquitectura y navegación, diseño gráfico y de interfaces, lenguajes y estándares en Internet, test de aplicaciones Web.

2.5.6 DEFINIR EL MARCO DE TRABAJO

El principio básico de la Ingeniería es la adaptabilidad, refiriéndose a la organización del equipo de trabajo, la comunicación entre los miembros del proyecto, las actividades y tareas a realizarse, la recopilación de información, la creación de información, y la metodología empleada para realizar un proyecto de calidad que deben adaptarse a los desarrolladores del Proyecto, al tiempo de entrega y las restricciones solventando cualquier problema que se desea resolver

²¹www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

Para definir un marco de trabajo para la Ingeniería Web, se debe tener en cuenta que:

- Las aplicaciones Web, se entregan de manera incremental,
- Los cambios ocurrirán frecuentemente,
- Los plazos son cortos.

Comunicación con el cliente: La comunicación con el cliente se caracteriza por medio de dos grandes tareas:

- **El análisis del negocio:** Define el contexto empresarial-organizativo para las WebApps y otras aplicaciones de negocio.
- **La formulación:** Es una actividad de recopilación de requisitos que involucran a todos los participantes.

Planeación: Se planifica el plan del proyecto para incremento de la Aplicación Web, definiendo tareas y los plazos para realizar cada una, respetando los plazos establecidos para la entrega final del proyecto.

Modelado: El análisis y diseño de la ingeniería del software se adaptan al desarrollo de las Aplicaciones Web, juntándose para generar la Aplicación Web.

Construcción: “Las herramientas y las tecnologías de Ingeniería Web se aplican para construir la Aplicación Web que se ha modelado. Una vez que se construye el incremento de Aplicación Web se dirige a una serie de pruebas rápidas para asegurar que se descubran los errores en el diseño”²².

Despliegue: “Las WebApps se configuran para su ambiente operativo, se entrega a los usuarios finales y luego comienza un período de evaluación. La retroalimentación acerca de la evaluación para realizar los procesos respectivos”²³.

²²www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

²³www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

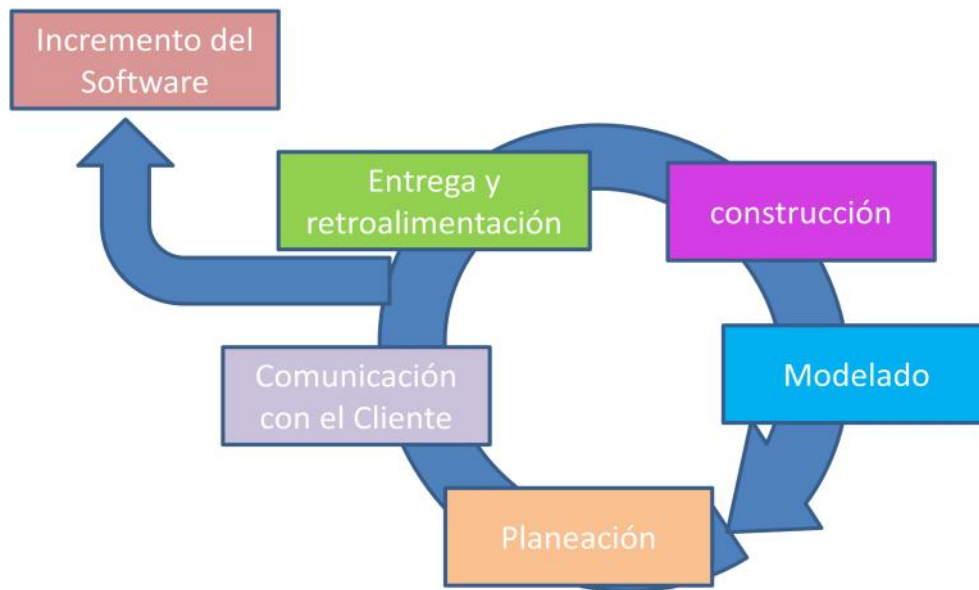


Figura 2. 2 Fase de Construcción de una Aplicación WEB²⁴.

2.5.7 MEJORES PRÁCTICAS EN LA INGENIERÍA WEB.

Se debe destinar tiempo a entender las necesidades del negocio y lo que el cliente requiere que haga el producto, incluso si los detalles de Aplicaciones Web son escasos.

Realizar una descripción de posibles escenarios en los que deben interactuar los usuarios con la Aplicación Web.

Realizar una planificación del Proyecto.

Incluir tiempo de modelado de la Aplicación Web.

“Revisar la consistencia y calidad de los modelos.

Utilizar herramientas y tecnología que permitan construir el sistema con tantos componentes reutilizables.

No apoyarse en usuarios anteriores para depurar la webApp”²⁵

²⁴www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

²⁵www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III.

2.5.8 FORMULACIÓN Y PLANEACIÓN PARA INGENIERÍA WEB²⁶

2.5.8.1 FORMULACIÓN DE SISTEMAS BASADOS EN WEB

La formulación de los sistemas para Aplicaciones Web, se basa en las necesidades y objetivos del proyecto y la información relacionada. "Los clientes y los ingenieros Web deben definir el contenido requerido, discutir la funcionalidad específica, características específicas e identificar la forma en que los usuarios finales interactuarán con las Aplicaciones Web, todo esto es formulación y recopilación de requisitos."²⁷

2.5.8.2 PREGUNTAS DE FORMULACIÓN:

En esta etapa es bueno realizar y responder las siguientes preguntas.

- ¿Cuál es la principal motivación para las WebApps?
- ¿Cuáles son los objetivos que deben satisfacer las WebApps?
- ¿Quién usará la WebApp?

“Si tenemos una respuesta correcta a estas preguntas implican metas específicas para el sitio Web.

En general se identificas dos categorías de metas²⁸:

Metas informativas: “indican una intención de proporcionar contenido información específicos al usuario final²⁹.”

Metas aplicables: “Indican la habilidad para realizar alguna TAREA dentro de la WebApp³⁰.”

²⁶Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

²⁷Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

²⁸Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

²⁹Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

³⁰Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

2.5.8.3 RECOPIACIÓN DE REQUISITOS PARA WEBAPPS

La recopilación de requerimientos propuestos para la Ingeniería de software son los siguientes:

- Identificar requisitos de contenido.
- Identificar requisitos funcionales.
- Definir escenarios de interacción para diferentes clases de usuarios.

“Los siguientes pasos de la recopilación de requisitos se dirigen para lograr estos objetivos:

1. Pedir a los clientes que definan una categoría de usuario y describan cada categoría.
2. Comunicarse con los clientes para definir los requisitos básicos para las WebApps.
3. Analizar la información recopilada y utilizar la información para realizar un seguimiento a los clientes.
4. Definir casos de uso que describan escenarios de interacción para cada clase de usuario”³¹.

³¹Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

2.5.9. MODELO DE ANÁLISIS PARA APLICACIONES WEB

2.5.9.1 REQUISITOS PARA EL ANÁLISIS DE LAS WEBAPP

“El análisis de requisitos para las WebApps abarca tres grandes tareas:

1. Formulación.
2. Recopilación de requisitos.
3. Modelado de análisis”³²

2.5.9.2 LA JERARQUÍA DE USUARIO

Las categorías de los usuarios finales se identifican como parte de las tareas reformulación y de recopilación de requisitos. Las categorías de usuarios son relativamente limitados y no necesitan una representación UML. Sin embargo, cuando crece el número de categorías de usuario, a veces es aconsejable desarrollar una jerarquía de usuarios.

2.5.9.3 DESARROLLO DE CASOS DE USO

“Los casos de uso se desarrollan para cada categoría de usuario descrita en la jerarquía de usuario. En el contexto de la ingeniería Web, el caso de uso en sí mismo es relativamente informal: un párrafo narrativo que describe una interacción específica entre el usuario y la WebApp”³³.

2.5.9.4 AFINACIÓN DEL MODELO DE CASO DE USO

“A la par que se crean los diagramas de casos de uso para cada categoría de usuario, se desarrolla una vista superior de los requisitos de la WebApp observables de manera externa.

³²Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo II, 517-543, Sexta Edición McGrawHill.

³³www.docstoc.com/.../Modelado-de-Analisis-para-Aplicaci...

Cada paquete se valora para garantizar que sea:

- Comprensible.
- Cohesivo
- Librementemente acoplados
- Jerárquicamente superficial.

Puesto que el análisis y modelado de actividades son actividades iterativas”³⁴.

2.5.9.5 EL MODELADO DE ANÁLISIS PARA WEBAPPS

“Se basa en la información que contienen los casos de uso desarrollados para la aplicación. Se identifican el contenido que presentará la WebApp y se extraen las funciones que se desarrollarán a partir de las descripciones de caso de uso. Finalmente los requisitos específicos de la implementación se deben desarrollar de modo que el ambiente y la infraestructura que apoyan la WebApps puedan construirse.

Cuatro actividades de análisis, cada una con su aporte a la creación de un modelo de análisis completo son:

- Análisis de contenido
- Análisis de interacción
- Análisis de funciones
- Análisis de configuración

El modelo es si mismo contiene elementos estructurales y dinámicos. Los elementos estructurales identifican las clases de análisis y los objetivos de contenido que se requieren para crear una WebApp que satisfaga las necesidades del cliente.

Los elementos dinámicos del modelo de análisis describen como interactúan los elementos estructurales, entre ellos y con los usuarios finales”³⁵.

³⁴www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

³⁵www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

2.5.9.6 EL MODELO DE CONTENIDO

“El modelo de contenido contiene elementos estructurales que proporcionan una importante visión de los requisitos de contenido para una WebApp. Además incluye todas las clases de análisis: entidades visibles para el usuario que se crean o manipulan conforme éste interactúa con la WebApp.

Al igual que otros elementos del modelo de análisis, el modelo de contenido se deriva a partir de un examen cuidadoso de los casos de uso desarrollados para la WebApp”³⁶.

2.5.9.7 DEFINICIÓN DE OBJETOS DE CONTENIDO

“Un objeto de contenido puede ser una descripción textual de un producto, un artículo que describa un evento noticioso

Los objetos de contenido se extraen en los casos de uso al examinar la descripción del escenario para referencias directas e indirectas al contenido”³⁷.

2.5.9.8 RELACIONES DE JERARQUÍA DE CONTENIDO

“El modelo de contenido puede contener diagramas de relación de entidades o árboles de datos que bosquejan las relaciones entre los objetos de contenido o la jerarquía de éste que mantiene una WebApp”³⁸.

2.5.9.9 EL MODELO DE INTERACCIÓN

Este modelo de interacción comprende cuatro elementos:

- Casos de uso
- Diagramas de secuencia
- Diagramas de estado
- Prototipo de interfaz de usuario

³⁶ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

³⁷ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

³⁸ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

Casos de Uso: Un caso de uso se modela para todos los procesos que la WebApp debe llevar a cabo. Los procesos se describen dentro del caso de uso por una descripción textual o una secuencia de pasos ejecutados. Los Diagramas de Actividad se pueden usar también para modelar escenarios gráficamente. Una vez que el comportamiento de la aplicación está captado de esta manera, los casos de uso se examinan y amplían para mostrar qué objetos se interrelacionan para que ocurra este comportamiento.

Diagrama de Secuencia: “Un diagrama de Secuencia muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren”³⁹.

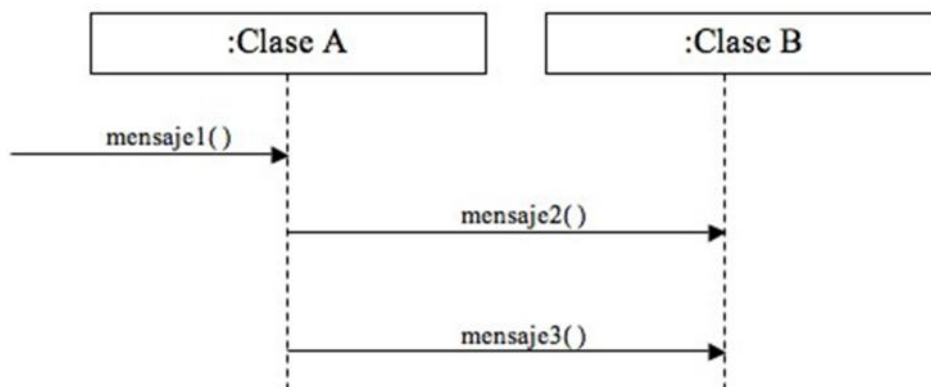


Figura 2. 3 Ejemplo de diagrama de secuencia

Diagramas de Estado: “El comportamiento en tiempo real de cada clase que tiene comportamiento dinámico y significativo, se modela usando un Diagrama de Estado. El diagrama de actividad puede ser usado también aquí, esta vez como

³⁹es.scribd.com/doc/52731260/13/Diagrama-de-Secuencia

una extensión del diagrama de estado, para mostrar los detalles de las acciones llevadas a cabo por los objetos en respuesta a eventos internos. El diagrama de actividad se puede usar también para representar gráficamente las acciones de métodos de clases”⁴⁰.

Prototipo de interfaz de usuario: “Algunas propuestas se basan en obtener de la definición de requisitos prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea de la estructura de la interfaz del sistema con el usuario. Esta técnica tiene el problema de que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final.

Puesto que las herramientas de desarrollo de la WebApp son abundantes y funcionalmente poderosas, es mejor crear prototipos de la interfaz mediante tales herramientas”⁴¹.

El modelo funcional: “El modelo funcional aborda dos elementos de procesamiento de la WebApp y cada uno representa un grafico diferente de la abstracción de procedimiento:

Funcionalidad observable respecto al usuario y que entrega al usuario final de WebApp.

Las operaciones dentro de las clases de análisis que implementan comportamientos asociados con la clase”⁴².

El modelo de configuración: “Las WebApps se deben diseñar e implementar de forma que se acomoden a una diversidad de ambientes, tanto del lado del servidor como del cliente. Se deben especificar el hardware del servidor y el ambiente del sistema operativo. Si las WebApps deben tener acceso a una gran base de datos o ínter operar con las aplicaciones corporativas existentes en el lado del servidor, se deben especificar las interfaces apropiadas, los protocolos de comunicación y la información”⁴³.

⁴⁰ mmc.geofisica.unam.mx/LuCAS/...modelado.../x101.html

⁴¹ eqaula.org/eva/file.php/2022/_000028_.pdf

⁴² <http://www.docstoc.com/docs/43814981/Modelado-de-Analisis-para-Aplicaciones-Web>

⁴³ <http://www.docstoc.com/docs/43814981/Modelado-de-Analisis-para-Aplicaciones-Web>

Análisis de navegación: “Uno de los aspectos más importantes en los sistemas de información en las WebApps es el de la navegación. La gran mayoría de las propuestas metodológicas para sistemas WebApp resaltan este aspecto ofreciendo modelos que permitan diseñarlo e implementarlo asegurando la calidad del resultado. Sin embargo, analizando dichos modelos y técnicas y viendo los resultados de diferentes estudios comparativos, se puede observar que este aspecto, en la mayoría de las propuestas, se trata solamente en las últimas fases del ciclo de vida, principalmente en diseño e implementación.

Los mecanismos de navegación se definen como parte del diseño. En esta etapa, los desarrolladores deben considerar requisitos de navegación globales”⁴⁴.

2.5.10 MODELO DE DISEÑO PARA APLICACIONES WEB ⁴⁵

2.5.10.1 TEMAS DE DISEÑO PARA INGENIERÍA WEB

“Cuando se aplica el diseño dentro del contexto de la ingeniería Web, se deben considerar cuestiones tanto genéricas como específicas. El diseño debe ser muy específico”⁴⁶.

2.5.10.2 DISEÑO DE CALIDAD DE UNA WEBAPP

“En la práctica, la calidad de los sitios web se ha evaluado de una manera “ad hoc”, basada en el sentido común y en la experiencia de los desarrolladores. El estudio de la calidad de productos y procesos de desarrollo para la Web es muy reciente y todavía no se dispone de métodos de evaluación ampliamente difundidos para este tipo de entorno.

Por lo tanto, existe la gran necesidad de metodologías efectivas para la obtención de aplicaciones Web de calidad. Actualmente, hay dos vertientes de metodologías de desarrollo para la Web: la comunidad de Ingeniería de Software y la comunidad de Hipermedia. Estas metodologías carecen sin embargo de métricas que puedan ser aplicadas a los modelos intermedios

⁴⁴ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

⁴⁵ Pressman Roger S. Ingeniería del Software Un enfoque práctico, Capítulo IV, 566-603 Sexta Edición McGrawHill.

⁴⁶ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III



Figura 2. 4 Ejemplo de Modelo de Objetos, Modelo Navegacional, utilizados en el proceso de desarrollo orientado a la WEB

Otras características de diseño de calidad Web son:

- Seguridad.
- Disponibilidad.
- Escalabilidad.
- Tiempo en el mercado ⁴⁷

2.5.10.3 METAS DE DISEÑO

“Una de las metas de diseño de una WebApp es abarcar lo siguiente:

- Simplicidad , Consistencia, Identidad, Robustez, Navegabilidad, Apariencia visual, Compatibilidad ⁴⁸

2.5.10.4 PIRÁMIDE DEL DISEÑO IWEB

El diseño conduce a un modelo que contiene una mezcla adecuada de estética, contenido y tecnología. La pirámide de la figura2.5, muestra las diferentes actividades del diseño:

⁴⁷ www.fe.up.pt/ipc/suporte/praticas/JISBD-01.pdf

⁴⁸ www.utpl.edu.ec/ecc/wiki/index.php/Sistemas_III

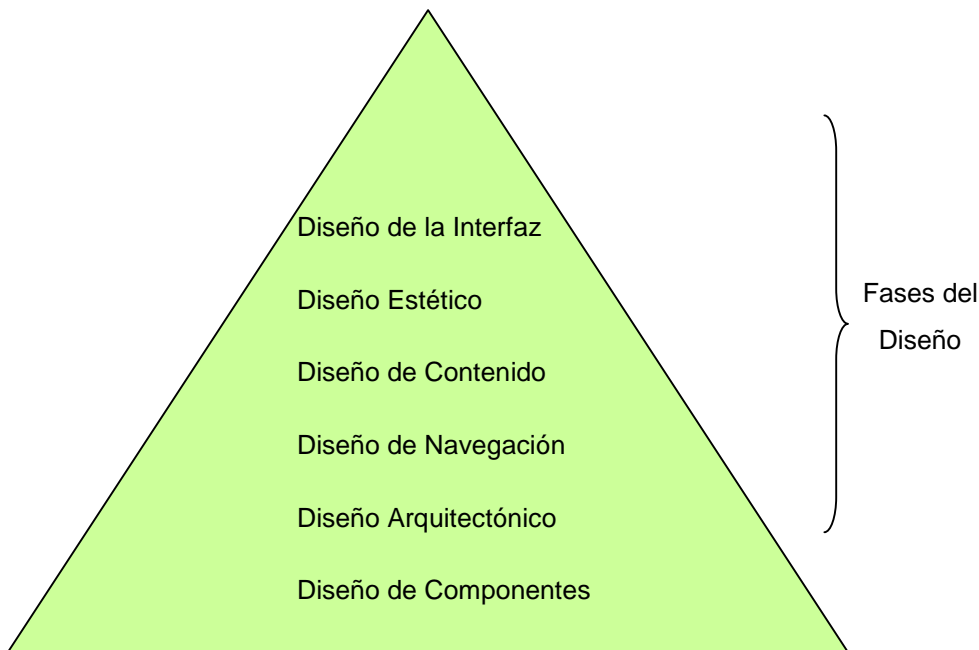


Figura 2. 5 Pirámide del diseño IWEB ⁴⁹

- **Diseño de la Interfaz:** “Describe la estructura y organización de la interfaz del usuario. Incluye una representación de la plantilla de pantalla.
- **Diseño Estético:** Describe la apariencia de la WebApp. Incluye esquemas de color, tamaño de texto, fuente y ubicación, uso de gráficos.
- **Diseño de Contenido:** Define la estructura y bosquejo de todo el contenido. Establece las relaciones entre los objetos de contenido.
- **Diseño de Navegación:** Define las rutas de navegación al contenido y a las funciones de la WebApp y su habilitación a los usuarios de acuerdo a sus niveles de acceso.
- **Diseño Arquitectónico:** Identifica la estructura hipermedial global para la WebApp.
- **Diseño de Componentes:** Desarrolla la lógica de procesamiento detallado que se requiere para implementar componentes funcionales. Está dividido en:
 - Componentes basados en Clases: Se basan en la información obtenida del análisis y en el modelo arquitectónico creado. Se concentran en la

⁴⁹ Figura: Pressman, Ingeniería del Software – Un Enfoque Práctico, Sexta Edición, Pág: 573

elaboración de clases de análisis (específicas del dominio del problema), su definición e infraestructura.

- Componentes Convencionales: Son patrones o grupos lógicos que permiten mantener el dominio funcional de los sistemas. Se utilizan diagramas de flujo para seguir la secuencia de los mismos.
- **Diseño de la interfaz de la WebApp:** Toda interfaz de usuario debe presentar las siguientes características:
 - Fácil de usar, Fácil de aprender, Fácil de navegar, Intuitiva, Consistente, Eficiente, Libre de errores, Funcional.

Debe ofrecer al usuario final una experiencia satisfactoria y gratificante.

Flujo de trabajo en el diseño de la interfaz: Las siguientes tareas representan un flujo de trabajo rudimentario para el diseño de la interfaz WebApp.

1. Revisar la información contenida en el modelo de análisis y refinarla conforme se requiera.
2. Desarrollar un bosquejo aproximado de la plantilla de la interfaz de la WebApp.
3. Correlacionar los objetivos del usuario con acciones específicas de la interfaz.
4. Definir un conjunto de tareas de usuario que estén asociadas con cada acción.
5. Elaborar bosquejos con imágenes de la pantalla para cada acción de la interfaz.
6. Refinar la plantilla de la interfaz y los bosquejos con el uso de entradas desde el diseño estético.
7. Identificar los objetos de la interfaz del usuario que se requieren para implementarla.
8. Desarrollar una representación de procedimiento de la interacción del usuario con la interfaz.
9. Desarrollar una representación del comportamiento de la interfaz.
10. Describir la plantilla de la interfaz para cada estado.
11. Refinar y revisar el modelo de diseño de la interfaz.

- **Diseño estético:** El diseño estético, también llamado diseño gráfico, es un esfuerzo artístico que complementa los aspectos técnicos de la ingeniería Web. Sin él una WebApp puede ser funcional, pero sin atractivo.

- **Diseño de contenido:** El diseño de contenido se enfoca en dos asuntos de diseño diferentes, cada uno lo abordan individuos con distintos conjuntos de habilidades. El diseño de contenido desarrolla una representación de diseño para los objetos de contenido y representa los mecanismos que se requieren para que establezcan sus relaciones uno con otro.

- **Objetos de contenido:** La relación entre objetos de contenido, defina como parte del modelo de análisis WebApp y los objetos de diseño que representan contenido es análoga a la relación entre las clases de análisis y los componentes de diseño.

Un objeto de contenido tiene atributos que incluyen información específica de contenido y atributos específicos de implementación que se especifican como parte del diseño.

- **Diseño arquitectónico:** El diseño arquitectónico está enlazado con las metas establecidas para la WebApp, el contenido que se presentará, los usuarios que la vistan y la filosofía de navegación que se establezca.

La arquitectura de contenido se basa en la forma en que los objetos de contenido se estructuran para su presentación y navegación.

- **Arquitectura de contenido:** Se centra en la definición de la estructura hipermedia global de la WebApp. El diseño se puede elegir de cuatro diferentes estructuras de contenido:
 - Estructuras lineales.
 - Estructuras en retícula.
 - Las estructuras jerárquicas.
 - La estructura en red.

- **Diseño de navegación:** El diseñador debe definir las rutas de navegación que habiliten para la ruta de los usuarios el acceso al contenido y las funciones de las WebApps. Para ello se debe:

Identificar la semántica de navegación para diferentes usuarios del sitio

Definir la mecánica que logra la navegación.

- **Semántica de navegación:** Durante las etapas iniciales del diseño, de navegación se valora la arquitectura de contenido de la WebApp para determinar una o más FdN (Funciones de Navegación) para cada caso de uso. Como se anotó anteriormente, una FdN (Funciones de Navegación) identifica los nodos de navegación y los vínculos que permiten la navegación entre ellos.
- **Sintaxis de navegación:** Conforme el diseño se lleva a cabo se define la mecánica de navegación, entre sus posibilidades tenemos:
 - Vinculo de navegación individual.
 - Barra de navegación horizontal.
 - Columna de navegación vertical.
 - Pestañas.
 - Mapas del sitio⁵⁰.

⁵⁰www.buenasTAREAs.com › Inicio › Tecnología

2.6 HTML

“El HTML, Hyper Text MarkupLanguage (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomo dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX (el editor vi o ed) o el que ofrece MS Office (Word). El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o .html

Estos documentos pueden ser mostrados por los visores o "browsers" de páginas Web en Internet, como Netscape Navigator, Mosaic, Opera y Microsoft Internet Explorer.

También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo, texto que vuela desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página.

Un documento escrito en HTML contiene las siguientes etiquetas en el siguiente orden:

Ejemplo:

```
<HTML>
```

```
<HEAD>
```

```
<TITLE> Título de mi página de Internet </TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<H1><CENTER>Primerapagina</CENTER></H1>
```

```
<HR>51
```

2.7 MYSQL

“MySQL es la base de datos open source más popular y, posiblemente, mejor del mundo. Su continuo desarrollo y su creciente popularidad, haciendo de MySQL un competidor cada vez más directo de gigantes en la materia de las bases de datos como Oracle

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

Existen muchos tipos de bases de datos, desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL, como base de datos relacional, utiliza múltiples tablas para almacenar y organizar la información.

MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

⁵¹Pablo Ravioli (20 de Marzo de 2012) Lenguaje de programación para páginas web. Recuperado(3 de julio-2012)<http://www.monografias.com/trabajos7/html/html.shtml>

También es muy destacable, la condición de open source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

MySQL es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto”⁵².

2.7.1 APLICACIONES.

“MySQL es muy utilizado en aplicaciones web, como Drupal o php Bulletin Board, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación”⁵³.

2.7.2 USO DE MYSQL

“MySQL es muy popular en aplicaciones web, y es componente de las plataformas LAMP, MAMP, WAMP, entre otras. MySQL suele combinarse con el popular lenguaje PHP”⁵⁴.

⁵²<http://www.espestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm/JoséManuelPerez/> 16-08-2005/recuperado 24/04/2012/¿Que es Mysql?

⁵³ <http://es.wikipedia.org/wiki/MySQL>

⁵⁴ Diccionario de Informatica(2011);Recuperado(2012-06-2012), Información y significado de MySQL, <http://www.alegsa.com.ar/Dic/mysql.php>

2.7.3 CARACTERÍSTICAS DE MYSQL

“Las características de MySql son:

- MySQL está escrito en C y C++
- Emplea el lenguaje SQL para consultas a la base de datos.
- MySQL Server está disponible como freeware bajo licencia GPL.
- MySQL Enterprise es la versión por suscripción para empresas, con soporte las 24 horas.
- Trabaja en las siguientes plataformas: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Microsoft Windows (95, 98, ME, NT, 2000, XP y Vista)”⁵⁵.

2.8 MODELOS DE BASES DE DATOS⁵⁶

“Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una "descripción" de algo conocido como contenedor de datos (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos”⁵⁷:

2.8.1 BASES DE DATOS JERÁRQUICAS

“Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

⁵⁵www.alegsa.com.ar/Dic/mysql.php

⁵⁶es.wikipedia.org/wiki/Base_de_datos

⁵⁷es.wikipedia.org/wiki/Base_de_datos

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos⁵⁸.

2.8.2 BASE DE DATOS DE RED

“Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales⁵⁹.

2.8.3 BASES DE DATOS TRANSACCIONALES

“Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales⁶⁰.

2.8.4 BASES DE DATOS RELACIONALES

“Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la

⁵⁸es.wikipedia.org/wiki/Base_de_datos

⁵⁹es.wikipedia.org/wiki/Base_de_datos

⁶⁰es.wikipedia.org/wiki/Base_de_datos

teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que representarían las tuplas, y campos (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años 80 la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión⁶¹.

2.8.5 BASES DE DATOS MULTIDIMENSIONALES

“Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como creación de Cubos OLAP. Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos relacional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos

⁶¹es.wikipedia.org/wiki/Base_de_datos

de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar⁶².

2.8.6 BASES DE DATOS ORIENTADAS A OBJETOS

“Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los objetos completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

- Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.
- Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.
- Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones⁶³.

2.9 PROGRAMACIÓN ORIENTADA A OBJETOS (POO)

“La POO es un paradigma de programación (o técnica de programación) que utiliza objetos e interacciones en el diseño de un sistema.

⁶²es.wikipedia.org/wiki/Base_de_datos

⁶³es.wikipedia.org/wiki/Base_de_datos

Durante años, los programadores se han dedicado a construir aplicaciones muy parecidas que resolvían una y otra vez los mismos problemas. Para conseguir que los esfuerzos de los programadores puedan ser utilizados por otras personas se creó la POO. Que es una serie de normas de realizar las cosas de manera que otras personas puedan utilizarlas y adelantar su trabajo, de manera que consigamos que el código se pueda reutilizar”⁶⁴.

2.9.1 CÓMO SE PIENSA EN OBJETOS

“Pensar en términos de objetos es muy parecido a cómo lo haríamos en la vida real. Por ejemplo vamos a pensar en un coche para tratar de modelizarlo en un esquema de POO. Diríamos que el coche es el elemento principal que tiene una serie de características, como podrían ser el color, el modelo o la marca. Además tiene una serie de funcionalidades asociadas, como pueden ser ponerse en marcha, parar o aparcar.

Pues en un esquema POO el coche sería el objeto, las propiedades serían las características como el color o el modelo y los métodos serían las funcionalidades asociadas como ponerse en marcha o parar.

Por poner otro ejemplo vamos a ver cómo modelizaríamos en un esquema POO una fracción, es decir, esa estructura matemática que tiene un numerador y un denominador que divide al numerador, por ejemplo $3/2$.

La fracción será el objeto y tendrá dos propiedades, el numerador y el denominador. Luego podría tener varios métodos como simplificarse, sumarse con otra fracción o número, restarse con otra fracción, etc.

Estos objetos se podrán utilizar en los programas, por ejemplo en un programa de matemáticas harás uso de objetos fracción y en un programa que gestione un taller de coches utilizarás objetos coche. Los programas Orientados a objetos utilizan muchos objetos para realizar las acciones que se desean realizar y ellos

⁶⁴webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

mismos también son objetos. Es decir, el taller de coches será un objeto que utilizará objetos coche, herramienta, mecánico, recambios, etc.”⁶⁵

2.9.2 CLASES EN POO

Las clases son declaraciones de objetos, también se podrían definir como abstracciones de objetos. Esto quiere decir que la definición de un objeto es la clase. Cuando se programa un objeto y define sus características y funcionalidades en realidad lo que se hace es programar una clase.

2.9.2.1 PROPIEDADES EN CLASES

“Las propiedades o atributos son las características de los objetos. Cuando definimos una propiedad normalmente especificamos su nombre y su tipo. Nos podemos hacer a la idea de que las propiedades son algo así como variables donde almacenamos datos relacionados con los objetos”⁶⁶.

2.9.2.2 MÉTODOS EN LAS CLASES

“Son las funcionalidades asociadas a los objetos. Cuando estamos programando las clases las llamamos métodos. Los métodos son como funciones que están asociadas a un objeto”⁶⁷.

2.9.3 OBJETOS EN POO

“Un objeto es una entidad provista de métodos o mensajes a los cuales responde (comportamiento); atributos con valores concretos (estado); y propiedades (identidad).

Para crear un objeto se tiene que escribir una instrucción especial que puede ser distinta dependiendo el lenguaje de programación que se emplee, pero será algo parecido a esto.

```
MiObjeto = new Objeto()
```

⁶⁵webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

⁶⁶webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

⁶⁷webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

Con la palabra `new` especificamos que se tiene que crear una instancia de la clase que sigue a continuación. Dentro de los paréntesis podríamos colocar parámetros con los que inicializar el objeto de la clase Objeto”⁶⁸.

2.9.3.1 ESTADOS EN OBJETOS

“Cuando tenemos un objeto sus propiedades toman valores. Por ejemplo, cuando tenemos un coche la propiedad `color` tomará un valor en concreto, como por ejemplo rojo o gris metalizado. El valor concreto de una propiedad de un objeto se llama estado.

Para acceder a un estado de un objeto para ver su valor o cambiarlo se utiliza el operador punto.

```
miCoche.color = rojo
```

El objeto es `miCoche`, luego colocamos el operador punto y por último el nombre de la propiedad a la que deseamos acceder. En este ejemplo estamos cambiando el valor del estado de la propiedad del objeto a rojo con una simple asignación”⁶⁹.

2.9.3.2 MENSAJES EN OBJETOS

“Un mensaje en un objeto es la acción de efectuar una llamada a un método. Por ejemplo, cuando le decimos a un objeto coche que se ponga en marcha estamos pasándole el mensaje “ponte en marcha”.

Para mandar mensajes a los objetos utilizamos el operador punto, seguido del método que deseamos invocar.

```
miCoche.ponerseEnMarcha()
```

En este ejemplo pasamos el mensaje `ponerseEnMarcha()`. Hay que colocar paréntesis igual que cualquier llamada a una función, dentro irían los parámetros”⁷⁰

⁶⁸webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

⁶⁹webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

⁷⁰webdelprofesor.ula.ve/ingenieria/.../pr2/practicaPOO.pdf

2.10 PHP

“PHP, es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor. PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser embebido dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno”⁷¹.

2.10.1. HISTORIA

“PHP fue originalmente diseñado en Perl, seguidos por la escritura de un grupo de CGI binarios escritos en el lenguaje C por RasmusLerdorf en el año 1994 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. El 8 de junio de 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio FormInterpreter para crear PHP/FI”⁷².

2.10.2 USOS DE PHP

“Los principales usos del PHP son los siguientes:

- Programación de páginas web dinámicas, habitualmente en combinación con el motor de base de datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.

⁷¹es.wikipedia.org/wiki/PHP

⁷²es.wikipedia.org/wiki/PHP

- Programación en consola, al estilo de Perl o Shell scripting.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y Qt/GTK+, lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado⁷³.

2.10.3 CARACTERÍSTICAS DE PHP

2.10.3.1 VENTAJAS “

- Es un lenguaje multiplataforma
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

2.10.3.2 DESVENTAJAS

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización, unicode, etc.

⁷³es.wikipedia.org/wiki/PHP

- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características promueve la creación de código desordenado y complejo de mantener.
- Está diseñado especialmente para un modo de hacer aplicaciones web que es ampliamente considerado problemático y obsoleto (mezclar el código con la creación de la página web)⁷⁴.

2. 11 JAVASCRIPT

“JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador, aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DocumentObjectModel (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a

⁷⁴es.wikipedia.org/wiki/PHP

funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML”⁷⁵.

2.12 AJAX⁷⁶

2.12.1 INTRODUCCIÓN A AJAX

“Ajax no es una tecnología en sí mismo. En realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes.

Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías”⁷⁷.

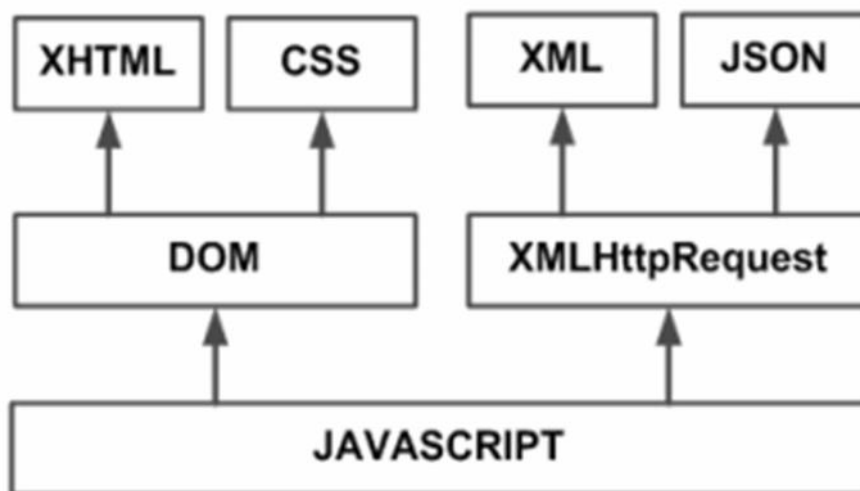


Figura 2. 6 Tecnologías agrupadas bajo el concepto de AJAX⁷⁸

⁷⁵es.wikipedia.org/wiki/JavaScript

⁷⁶ WWW.LIBROSWEB.ES , [INTRODUCCIÓN A AJAX](#)

⁷⁷ WWW.LIBROSWEB.ES , [INTRODUCCIÓN A AJAX](#)

“Desarrollar aplicaciones AJAX requiere un conocimiento avanzado de todas y cada una de las tecnologías anteriores.

En las aplicaciones web tradicionales, las acciones del usuario en la página (pinchar en un botón, seleccionar un valor de una lista, etc.) desencadenan llamadas al servidor. Una vez procesada la petición del usuario, el servidor devuelve una nueva página HTML al navegador del usuario.

En el siguiente esquema, la Figura 2.7 de la izquierda muestra el modelo tradicional de las aplicaciones web. La imagen de la derecha muestra el nuevo modelo propuesto por AJAX⁷⁹:

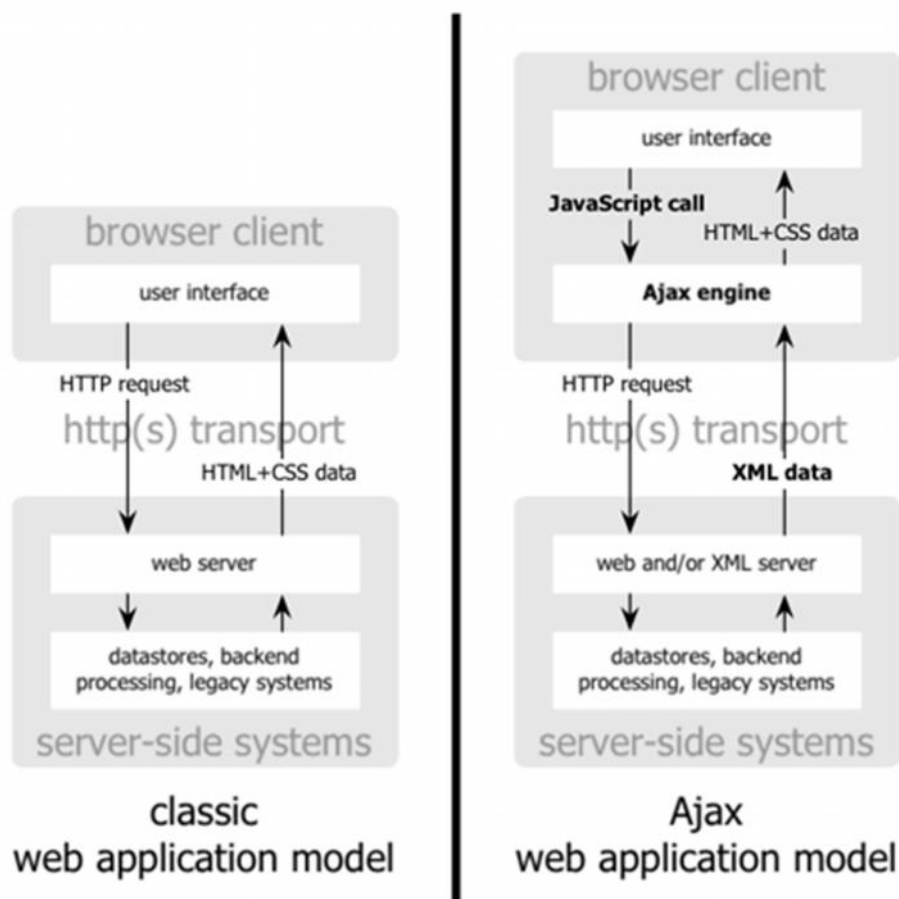


Figura 2. 7 Comparación Gráfica del Modelo tradicional de aplicación web y del nuevo modelo propuesto por AJAX⁸⁰.

⁷⁸ WWW.LIBROSWEB.ES › [INTRODUCCIÓN A AJAX](#)

⁷⁹ WWW.LIBROSWEB.ES › [INTRODUCCIÓN A AJAX](#)

“Esta técnica tradicional para crear aplicaciones web funciona correctamente, pero no crea una buena sensación al usuario. Al realizar peticiones continuas al servidor, el usuario debe esperar a que se recargue la página con los cambios solicitados. Si la aplicación debe realizar peticiones continuas, su uso se convierte en algo molesto.

Las aplicaciones construidas con AJAX eliminan la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor. La nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor.

La Figura 2.8 muestra la diferencia más importante entre una aplicación web tradicional y una aplicación web creada con AJAX.⁸¹

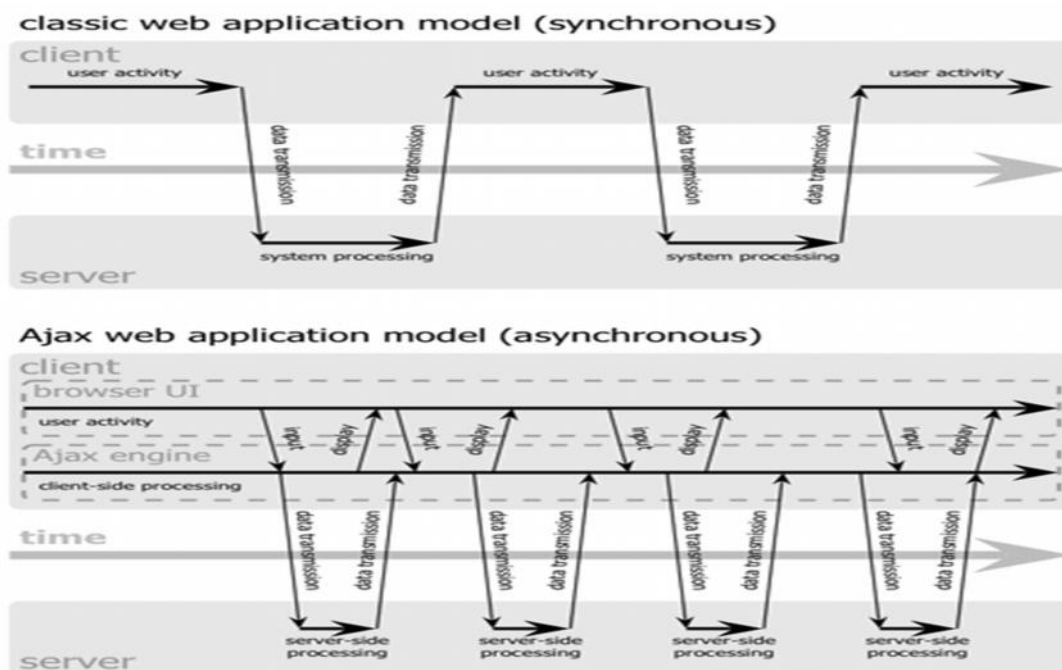


Figura 2. 8 Comparación entre las comunicaciones sincrónicas de las aplicaciones web tradicionales y las comunicaciones asíncronas de las aplicaciones AJAX⁸².

⁸⁰ www.librosweb.es › [Introducción a AJAX](#), Imagen original creada por AdaptivePath y utilizada con su permiso

⁸¹ www.librosweb.es › [Introducción a AJAX](#)

“La imagen superior muestra la iteración síncrona propia de las aplicaciones web tradicionales. La imagen inferior muestra la comunicación asíncrona de las aplicaciones creadas con AJAX

Las peticiones HTTP al servidor se sustituyen por peticiones JavaScript que se realizan al elemento encargado de AJAX. Las peticiones más simples no requieren intervención del servidor, por lo que la respuesta es inmediata. Si la interacción requiere una respuesta del servidor, la petición se realiza de forma asíncrona mediante AJAX. En este caso, la interacción del usuario tampoco se ve interrumpida por recargas de página o largas esperas por la respuesta del servidor.

Las aplicaciones más conocidas basadas en AJAX son:

- Gestores de correo electrónico: Gmail, Yahoo Mail, Windows Live Mail.
- Cartografía: Google Maps, Yahoo Maps, Windows Live Local.
- Aplicaciones web y productividad: Google Docs, Zimbra, Zoho.
- Otras: Netvibes [metapágina], Digg [noticias], Meebo [mensajería], 30 Boxes [calendario], Flickr[fotografía]”⁸³.

2.13 RUP ⁸⁴

“Rational Unified Process es una infraestructura flexible de desarrollo de software que proporciona prácticas recomendadas probadas y una arquitectura configurable.

Las mejores prácticas del Rational Unified Process, (RUP), son un conjunto de procesos web-enabled de ingeniería de software que dan Guía para conducir las actividades de desarrollo del equipo. Como una plataforma de procesos que abarca todas las prácticas de la industria, el RUP permite seleccionar fácilmente el conjunto de componentes de proceso que se ajustan a las necesidades específicas del proyecto. Se podrán alcanzar resultados predecibles unificando el

⁸²www.librosweb.es › [Introducción a AJAX](#), Imagen original creada por AdaptivePath y utilizada con su permiso

⁸³www.librosweb.es › [Introducción a AJAX](#)

⁸⁴www.rational.com.ar/herramientas/rup.html

equipo con procesos comunes que optimicen la comunicación y creen un entendimiento común para todas las tareas, responsabilidades y artefactos. Desde un único sitio web centralizado de intercambio, el Software Rational, las plataformas, herramientas y expertos de dominios proveen los componentes de proceso necesarios para el éxito”⁸⁵.

2.13.1. ENTREGA DEL SOFTWARE OPERATIVO CON CONFIANZA

“El RUP mantiene al equipo enfocado en producir incrementalmente software operativo a tiempo, con las características requeridas y con la calidad requerida. Las mejores prácticas probadas en la industria, contenidas en el RUP, incorporan las lecciones aprendidas de cientos de líderes de la industria y miles de proyectos. Ya no hay necesidad de re-inventar soluciones a desafíos de la ingeniería de software bien conocidos. Siguiendo el acercamiento al desarrollo iterativo del RUP, es posible entregar a tiempo y con confianza el software”⁸⁶.

2.13.2. CONTROL DE NUEVAS HERRAMIENTAS Y TECNOLOGÍAS

“La plataforma del RUP (Rational Unified Process) permite controlar nuevas herramientas y tecnologías en un único ambiente a través de contenido Plug-In customizado, mentores de herramientas y ayuda. Los Plug-Ins tecnológicos permiten actualizar el proceso de desarrollo y customizarlo a medida que la tecnología, herramientas y plataformas evolucionan. Para controlar completamente las nuevas tecnologías e incrementar la eficiencia en el uso de las herramientas, RUP provee mentores específicos on-line para las mismas que muestran como implementarlas en el nuevo ambiente”⁸⁷.

2.13.3. CARACTERÍSTICAS Y BENEFICIOS

“No existen dos proyectos de desarrollo de software que sean iguales. Cada uno tiene prioridades, requerimientos, y tecnologías muy diferentes. Sin embargo, en

⁸⁵www.rational.com.ar/herramientas/rup.html

⁸⁶www.rational.com.ar/herramientas/rup.html

⁸⁷www.rational.com.ar/herramientas/rup.html

todos los proyectos, se debe minimizar el riesgo, garantizar la predictibilidad de los resultados y entregar software de calidad superior a tiempo. Rational Unified Process, o RUP, es una plataforma flexible de procesos de desarrollo de software que ayuda proveyendo guías consistentes y personalizadas de procesos para todo el equipo de proyecto”⁸⁸.

2.13.4. SISTEMAS OPERATIVOS Y PLATAFORMAS DE HARDWARE APROPIADAS

“Los sistemas Operativos y plataformas recomendadas son:

- HP-UX , Linux ,Sun Solaris, Windows 2000, Windows NT, Windows XP”⁸⁹

2.14 CSS ⁹⁰

“CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados "documentos semánticos"). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc.

Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página”⁹¹.

⁸⁸www.rational.com.ar/herramientas/rup.html

⁸⁹www.rational.com.ar/herramientas/rup.html

⁹⁰daw-fiec.pbworks.com/w/page/16963466/CSS

2.14.1 SOPORTE DE CSS EN LOS NAVEGADORES

“El trabajo del diseñador web siempre está limitado por las posibilidades de los navegadores que utilizan los usuarios para acceder a sus páginas. Por este motivo es imprescindible conocer el soporte de CSS en cada uno de los navegadores más utilizados del mercado.

Internamente los navegadores están divididos en varios componentes. La parte del navegador que se encarga de interpretar el código HTML y CSS para mostrar las páginas se denomina motor. Desde el punto de vista del diseñador CSS, la versión de un motor es mucho más importante que la versión del propio navegador”⁹².

2.14.2 FUNCIONAMIENTO BÁSICO DE CSS

“Antes de la adopción de CSS, los diseñadores de páginas web debían definir el aspecto de cada elemento dentro de las etiquetas HTML de la página”⁹³.

2.15 UML

“Lenguaje Unificado de Modelado (LUM o UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). 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.

⁹¹ daw-fiec.pbworks.com/w/page/16963466/CSS

⁹² daw-fiec.pbworks.com/w/page/16963466/CSS

⁹³ daw-fiec.pbworks.com/w/page/16963466/CSS

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas⁹⁴.

2.16 GENERADOR DE CÓDIGO CREACOD

“CreaCod es un sistema desarrollado con la finalidad de ayudar al programador de bases de datos a crear clases (librerías) de control de cualquier tipo de bases de datos, usando para ello cualquier lenguaje de programación. Además, CreaCod puede generar las pantallas, ya sean GUI o WEB, con la final de gestionar la información contenida (Agregar, Editar, Eliminar, Borrar).

CreaCod utiliza, para la generación de los archivos de control y gestión de la base de datos, plantillas entendibles al usuario, las cuales se pueden modificar, o

⁹⁴es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado

incluso ser creadas desde cero, ya que están diseñadas con el lenguaje propio de programación”⁹⁵.

2.16.1 COMO GENERA CÓDIGO CREACOD

“CreaCod genera el código fuente a partir de plantillas previamente establecidas, las cuales pueden ser editadas por el usuario, ya que el código es muy similar al usado en el lenguaje de programación que se utilice”⁹⁶.

```
9 Class {TABLE_NAMCLS}
10
11 Private m_DataBase
12 Private RS
13 Public m_Utills
14 Private m_Code
15 Private m_ColEnabled
16 Private m_ColOrder
17 Private m_LastStringSQL
18
19 {BEGIN}
20 Public {COLUMN_CLSOBJ}
21 {END}
22
23 {BEGIN}
24 Private m_{COLUMN_NAMFUN}, m_Change{COLUMN_NAMFUN}
25 {END}
26
27 Private Sub Class_Initialize()
28     Set RS = Server.CreateObject("adodb.Recordset")
```

Figura 2. 9 Ejemplo de código de plantilla A.S.P.⁹⁷

⁹⁵ Sistema realizado por Eduardo Chávez Reina como tema de Tesis "Análisis, Diseño y Desarrollo de un Generador de Código fuente para Gestión de Información de MySql, SQL Server y Access para los lenguajes Java, PHP y ASP" para ESPE

⁹⁶ Sistema realizado por Eduardo Chávez Reina como tema de Tesis para ESPE

⁹⁷ Figura tomada de Sistema realizado por Eduardo Chávez Reina como tema de Tesis "Análisis, Diseño y Desarrollo de un Generador de Código fuente para Gestión de Información de MySql, SQL Server y Access para los lenguajes Java, PHP y ASP" para ESPE

CAPÍTULO 3

ANÁLISIS Y DISEÑO DEL WEBSITE Y GUÍA INTERACTIVA

3.1. ANÁLISIS DE SITUACIÓN ACTUAL

Actualmente la página Web del Ministerio del Ambiente ha incorporado al SNAP (Sistema Nacional de Áreas Protegidas) a su Web Site, lo llaman punto verde y el mismo posee información básica de las Áreas Protegidas del Ecuador.

El Web Site, no posee información interactiva de especies de Flora y Fauna, Etnias, Sitios de interés para Turismo, Geografía, Hidrografía, Clima, etc. Que son de interés para los visitantes Nacionales y Extranjeros.

La información que consta es muy específica y estática, lo cual no es de mayor interés para los visitantes y está desplegada en dos páginas, en la primera consta información general de las Áreas y en la segunda constan archivos anexados como archivos pdf's.

El Web Site no interactúa con otro sistema para la obtención de datos, para las Áreas Protegida la información disponible está incluida en el mismo sitio, para su libre uso.

El Web Site, actualmente está hecho en PHP.

3.2 MENÚ HORIZONTAL DEL WEB SITE ACTUAL

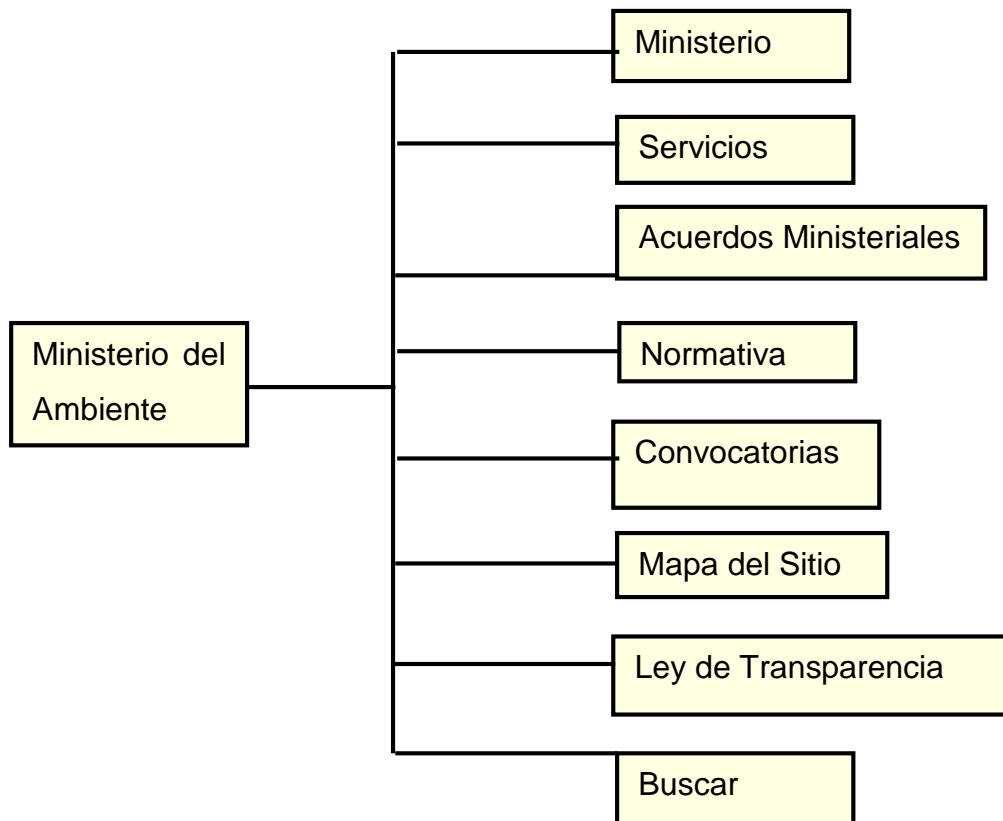


Figura 3. 1 Menú Horizontal de navegación en el Web Site actual.

3.3 FORMULACIÓN

Luego del Análisis de los requerimientos solicitados por la SNAP para el Web Site y el Cd interactivo de Áreas Protegidas se obtienen los siguientes requerimientos:

- Material didáctico para estudiantes a nivel Nacional e internacional.
- Promoción de la Guía de Áreas Naturales a otras culturas.
- Autogestión para seguir realizando los programas de conservación de las especies.
- Mejora de la imagen del Ecuador a nivel internacional
- Aportar a mejorar la calidad del web Site existente de Áreas Protegidas

3.4 ESPECIFICACIÓN DE REQUERIMIENTOS

La factibilidad de este proyecto se inicia con la necesidad existente en el Ministerio del Ambiente de tener una Guía sobre Áreas Protegidas, dentro de las cuales constan : Parques Nacionales, Reservas Ecológicas, Reservas Biológicas, Reservas Marinas, Parques Nacionales de Recreación, la cual permita difundir este tema, tanto en un sitio web y con un CD interactivo; el mismo que servirá para enseñanza Local y para envíos al extranjero para las personas que no puedan acceder al Internet; y la elaboración de un renovado sitio Web para el SNAP (Sistema Nacional de Áreas Protegidas) del Ministerio del Ambiente en que manejará información de las áreas protegidas.

3.4.1 INTRODUCCIÓN

Esta Guía y página Web, serán muy útil ya que permitirán dar un aporte valioso que mejorará los conocimientos de la juventud y la difusión de este tema a nivel internacional.

La información contenida en el Web Site y la Guía interactiva de Áreas Protegidas, permitirán al Ministerio del Ambiente a través del SNAP, actualizar, ingresar y eliminar datos de flora, fauna, etnias e información existente para las Áreas Protegidas del Ecuador

3.4.2 ESPECIFICACIONES DE CASOS DE USO

ADMINISTRADOR:

1. ADM_CU_RSNAPO1: Ingreso De Usuario
2. ADM_CU_RSNAPO2: Gestión De Información
3. ADM_CU_RSNAPO3: Gestión De Atributos
4. ADM_CU_RSNAPO4: Gestión De Áreas
5. ADM_CU_RSNAPO5: Gestión De Contenido
6. ADM_CU_RSNAPO6: Gestión De Vestimenta
7. ADM_CU_RSNAPO7: Gestión De Clima
8. ADM_CU_RSNAPO8: Gestión De Geografía
9. ADM_CU_RSNAPO9: Gestión De Provincia
10. ADM_CU_RSNAPO10: Gestión De Turismo
11. ADM_CU_RSNAPO11: Gestión De Etnias
12. ADM_CU_RSNAPO12: Gestión De Fotografías (Galerías)
13. ADM_CU_RSNAPO13: Gestión De Fauna
14. ADM_CU_RSNAPO14: Gestión De Flora
15. ADM_CU_RSNAPO15: Gestión De Hidrografía

**ESPACIO EN
BLANCO
INTENCIONAL**

3.4.2.1 DIAGRAMA DE CASOS DE USO DE contenido de información

Este diagrama muestra lo que puede realizar en el sistema el Usuario Administrador, el que se encargará de Gestionar los diferentes tipos de información y contenidos para el Web Site.

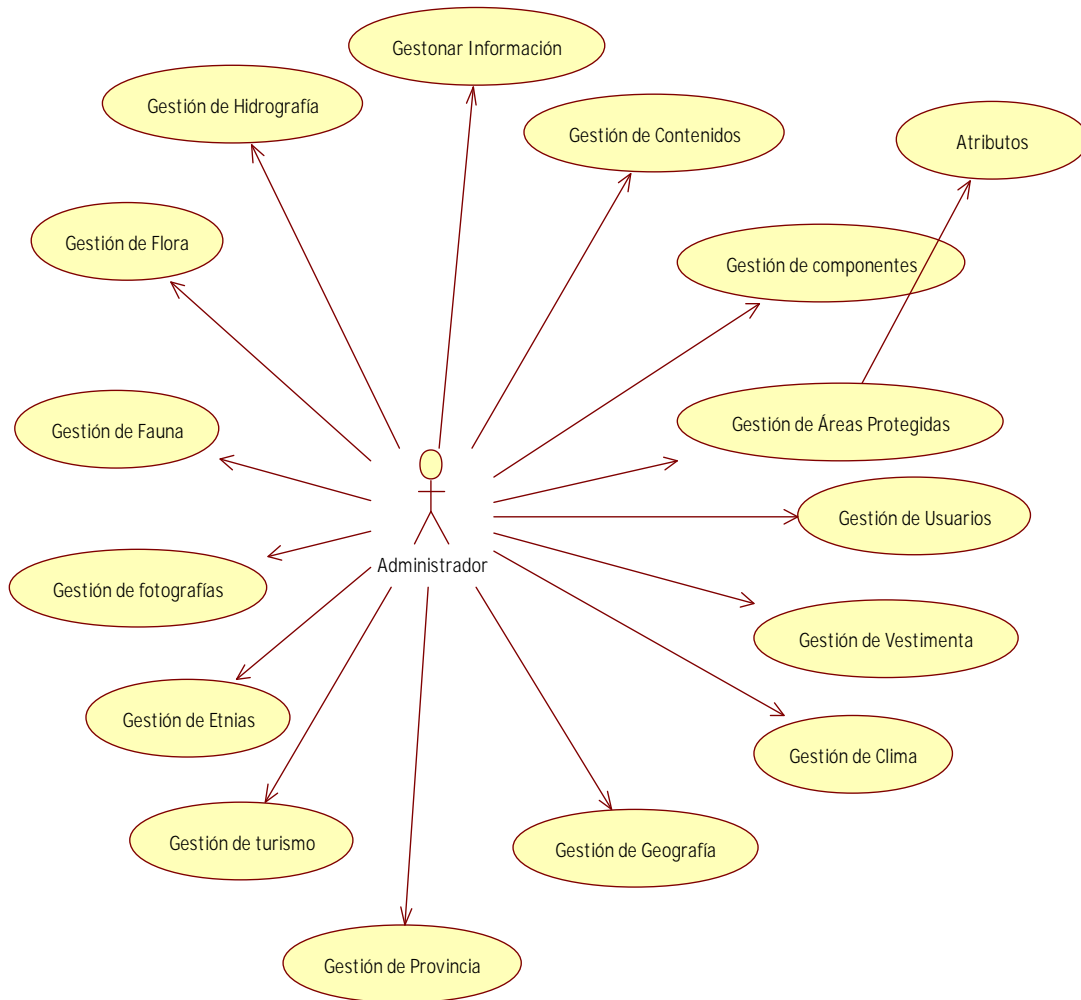


Figura 3. 2 Diagrama de Casos de uso para el Administrador del Sistema.

3.4.2.1.1 INGRESO DE USUARIO

ADM_CU_RSAP_01: Ingreso De Usuario			
Resumen:	Permite el ingreso de Usuario		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Login Usuario ○ LoginPassword 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema muestra los controles necesarios para que el usuario pueda ingresar los datos de : <ul style="list-style-type: none"> ○ Login Usuario ○ LoginPassword </td> </tr> </table>	Crear	El sistema muestra los controles necesarios para que el usuario pueda ingresar los datos de : <ul style="list-style-type: none"> ○ Login Usuario ○ LoginPassword
Crear	El sistema muestra los controles necesarios para que el usuario pueda ingresar los datos de : <ul style="list-style-type: none"> ○ Login Usuario ○ LoginPassword 		
Pre-condiciones			
Notas: El sistema tiene guardados registros de Usuario.			
Validaciones			
El sistema valida login y password, si es usuario incorrecto presentará el mensaje de que no puede ingresar presentando nuevamente la pantalla de ingreso.			

3.4.2.1.2 GESTIÓN DE INFORMACIÓN

ADM_CU_RSNAPO2: Gestión De Información			
Resumen:	Actualizar la Información existente en los archivos Inicio, Acerca de, ¿Quiénes somos? Y ayuda		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Los 68archivosAbout, Main y Help. 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema muestra la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda </td> </tr> </table>	Crear	El sistema muestra la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda
Crear	El sistema muestra la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema actualiza la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda </td> </tr> </table>	Modificar	El sistema actualiza la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda
Modificar	El sistema actualiza la información de texto de las pantallas: <ul style="list-style-type: none"> o Inicio o Acerca de o ¿Quiénes Somos? o Ayuda 		

.

Pre-condiciones
Notas: El sistema tiene guardados registros de Información.

.

Validaciones
El sistema no permite que el campo quede vacío.

.

3.4.2.1.3 GESTIÓN DE ATRIBUTOS

ADM_CU_RSNAPO_03: Gestión De Atributos			
Resumen:	Modificar los atributos existentes en el sistema.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema muestra los controles necesarios para actualizar la información de los atributos de: <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia </td> </tr> </table>	Crear	El sistema muestra los controles necesarios para actualizar la información de los atributos de: <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia
Crear	El sistema muestra los controles necesarios para actualizar la información de los atributos de: <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar la información de los atributos de : <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia </td> </tr> </table>	Modificar	El sistema permite modificar la información de los atributos de : <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia
Modificar	El sistema permite modificar la información de los atributos de : <ul style="list-style-type: none"> ○ Área ○ Geografía ○ Hidrografía ○ Clima ○ Vestimenta ○ Etnias ○ Turismo ○ Provincia 		
Pre-condiciones			
Notas: El sistema tiene guardados registros de Atributo.			
Validaciones			

3.4.2.1.4 GESTIÓN DE ÁREAS

ADM_CU_RSAP_04: Gestión De Áreas		
Resumen:	Permite realizar cambios en la información de Áreas	
Prioridad:	Esencial	
Actores directos:	Administrador	
Escenarios		
Tipo de escenario	Descripción	
Principal	Se deben establecer los valores de: ○ Áreas	
Secundario	Crear	El sistema permite crear una: ○ Área
Secundario	Modificar	El sistema permite modificar la información de : ○ Área
Pre-condiciones		
Notas: El sistema tiene guardados registros de Área.		
Validaciones		
El sistema alertará en el caso que exista el Área.		

3.4.2.1.5 GESTIÓN DE CONTENIDO

ADM_CU_RSNAPO5: Gestión De Contenido			
Resumen:	El programa permitirá actualizar el contenido de información general de: Áreas, Clima, Provincia, Geografía, Hidrografía, Turismo, Etnias, Flora, Fauna		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite crear nueva contenido de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna </td> </tr> </table>	Crear	El sistema permite crear nueva contenido de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna
Crear	El sistema permite crear nueva contenido de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar el contenido existente de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna </td> </tr> </table>	Modificar	El sistema permite modificar el contenido existente de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna
Modificar	El sistema permite modificar el contenido existente de: <ul style="list-style-type: none"> ○ Áreas ○ Clima ○ Provincia ○ Geografía ○ Hidrografía ○ Turismo ○ Etnias ○ Flora ○ Fauna 		
Pre-condiciones			
Notas: El sistema tiene guardados registros de contenido.			
Validaciones			

3.4.2.1.6 GESTIÓN DE VESTIMENTA

ADM_CU_RSAP_06: Gestión De Vestimenta		
Resumen:	Permite ingresar una nueva vestimenta	
Prioridad:	Esencial	
Actores directos:	Administrador	
Escenarios		
Tipo de escenario	Descripción	
Principal	Se deben establecer los valores de: ○ Vestimenta	
Secundario	Crear	El sistema permite ingresar una nueva: ○ Vestimenta
Secundario	Modificar	El sistema permite modificar un registro existente de: ○ Vestimenta
Pre-condiciones		
Notas: El sistema tiene guardados registros de Vestimenta.		
Validaciones		

3.4.2.1.7 GESTIÓN DE CLIMA

ADM_CU_RSAP_07: Gestión De Clima		
Resumen:	Realizar la Gestión de Clima	
Prioridad:	Esencial	
Actores directos:	Administrador	
Escenarios		
Tipo de escenario	Descripción	
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Clima 	
Secundario	Crear	El sistema permite ingresar un nuevo registro de : <ul style="list-style-type: none"> o Clima
Secundario	Modificar	El sistema permite modificar los registros existentes de : <ul style="list-style-type: none"> o Clima
.		
Pre-condiciones		
Notas:		
El sistema tiene guardados registros de Clima		
.		
Validaciones		
El sistema alertará en el caso que ya exista el Clima.		
.		

3.4.2.1.8 GESTIÓN DE GEOGRAFÍA

ADM_CU_RSAP_08: Gestión De Geografía	
Resumen:	Permite ingresar una nueva Geografía.
Prioridad:	Esencial
Actores directos:	Administrador
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: o Geografía
Secundario	Crear El sistema permite crear un nuevo registro de: o Geografía
Secundario	Modificar El sistema permite modificar un registro existente de: o Geografía
Pre-condiciones	
Notas: El sistema tiene guardados registros de Geografía.	
Validaciones	
El sistema alertará en el caso que exista la Geografía.	

3.4.2.1.9 GESTIÓN DE PROVINCIA

ADM_CU_RSNAPO9: Gestión De Provincia	
Resumen:	Permite ingresar el registro de una nueva Provincia
Prioridad:	Esencial
Actores directos:	Administrador
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: o Provincia
Secundario	Crear El sistema permite ingresar una nueva: o Provincia
Secundario	Modificar El sistema permite modificar una : o Provincia
Pre-condiciones	
Notas: El sistema tiene guardados registros de Provincia.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté incorrecto	

3.4.2.1.10 GESTIÓN DE TURISMO

ADM_CU_RSNAAP_10: Gestión De Turismo			
Resumen:	Permite crear, modificar, actualizar Turismo		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Turismo 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite crear un nuevo registro con: <ul style="list-style-type: none"> o Turismo </td> </tr> </table>	Crear	El sistema permite crear un nuevo registro con: <ul style="list-style-type: none"> o Turismo
Crear	El sistema permite crear un nuevo registro con: <ul style="list-style-type: none"> o Turismo 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Turismo </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Turismo
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Turismo 		
Pre-condiciones			
Notas:			
El sistema tiene guardados registros de Turismo			
Validaciones			
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto			

3.4.2.1.11 GESTIÓN DE ETNIAS

ADM_CU_RSNAAP_11: Gestión De Etnias			
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Etnias 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Etnias </td> </tr> </table>	Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Etnias
Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Etnias 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Etnias </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Etnias
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Etnias 		

.

Pre-condiciones
Notas: El sistema tiene guardados registros de Etnias.

.

Validaciones
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto

.

3.4.2.1.12 GESTIÓN DE FOTOGRAFÍAS (GALERÍAS)

ADM_CU_RSNAIP_12: Gestión De Fotografías (Galerías)			
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Galerías 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> o Galerías </td> </tr> </table>	Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> o Galerías
Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> o Galerías 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Galerías </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Galerías
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> o Galerías 		
.			
Pre-condiciones			
Notas:			
El sistema tiene guardados registros de Galerías.			
.			
Validaciones			
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto			
.			

3.4.2.1.13 GESTIÓN DE FAUNA

ADM_CU_RSAP_13: Gestión De Fauna			
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Fauna 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Fauna </td> </tr> </table>	Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Fauna
Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Fauna 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Fauna </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Fauna
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Fauna 		
Pre-condiciones			
El sistema tiene guardados registros de Fauna			
Validaciones			
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto			

3.4.2.1.14 GESTIÓN DE FLORA

ADM_CU_RSAP_14: Gestión De Flora			
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Flora 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Flora </td> </tr> </table>	Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Flora
Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Flora 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Flora </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Flora
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Flora 		
.			
Pre-condiciones			
Notas:			
El sistema tiene guardados registros de Flora.			
.			
Validaciones			
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto			
.			

3.4.2.1.15 GESTIÓN DE HIDROGRAFÍA

ADM_CU_RSAP_15: Gestión De Hidrografía			
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.		
Prioridad:	Esencial		
Actores directos:	Administrador		
Escenarios			
Tipo de escenario	Descripción		
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Hidrografía 		
Secundario	<table border="1"> <tr> <td>Crear</td> <td>El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Hidrografía </td> </tr> </table>	Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Hidrografía
Crear	El sistema permite ingresar un nuevo registro de: <ul style="list-style-type: none"> ○ Hidrografía 		
Secundario	<table border="1"> <tr> <td>Modificar</td> <td>El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Hidrografía </td> </tr> </table>	Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Hidrografía
Modificar	El sistema permite modificar un registro existente de: <ul style="list-style-type: none"> ○ Hidrografía 		
.			
Pre-condiciones			
Notas:			
El sistema tiene guardados registros de Hidrografía			
.			
Validaciones			
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto			
.			

USUARIO NAVEGADOR:

1. USR_CU_RSNAP_01: Usuario Navegador
2. USR_CU_RSNAP_02:Ver Áreas
3. USR_CU_RSNAP_03:Ver Flora
4. USR_CU_RSNAP_04:Ver Fauna
5. USR_CU_RSNAP_05:Ver Hidrografía
6. USR_CU_RSNAP_06:Ver Geografía
7. USR_CU_RSNAP_07: Ver Clima
8. USR_CU_RSNAP_08:Ver Provincia
9. USR_CU_RSNAP_09:Ver Turismo
- 10.USR_CU_RSNAP_10:Ver Vestimenta
- 11.USR_CU_RSNAP_11:Ver Etnias
- 12.USR_CU_RSNAP_12:Ver Galerías
- 13.USR_CU_RSNAP_13:Ver Contenido
- 14.USR_CU_RSNAP_14:Ver Información

**ESPACIO EN
BLANCO
INTENCIONAL**

3.4.2.2 DIAGRAMA DE CASOS DE USO PARA EL USUARIO NAVEGADOR

El Usuario Navegador Web, podrá realizar las actividades de navegación, búsqueda, consulta, visita de Galerías y seguimiento de links dentro del Web Site.



Figura 3. 3 Diagrama de casos de uso para Usuario dentro del Web Site.

3.4.2.2.1 USUARIO NAVEGADOR

USR_CU_RSNAPO1: Usuario Navegador	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Usuario
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.2 VER ÁREAS

USR_CU_RSNAPO2: Ver Áreas	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Áreas
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.3 VER FLORA

USR_CU_RSNAPO2: Ver Flora	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Flora
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas:	
El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.4 VER FAUNA

USR_CU_RSNAPO2: Ver Fauna	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Fauna
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.5 VER HIDROGRAFÍA

USR_CU_RSNAPO2: Ver Hidrografía	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Hidrografía
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.6 VER GEOGRAFÍA

USR_CU_RSNAPO6: Ver Geografía	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Geografía
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas:	
El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.7 VER CLIMA

USR_CU_RSNAPO7: Ver Clima	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Clima
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.8 VER PROVINCIA

USR_CU_RSNAPO8: Ver Provincia	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Provincia
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.9 VER TURISMO

USR_CU_RSNAPO9: Ver Turismo	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Turismo
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.10 VER VESTIMENTA

USR_CU_RSNAp_10: Ver Vestimenta	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> o Vestimenta
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> o Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> o Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.11 VER ETNIAS

USR_CU_RSNAAP_11: Ver Etnias	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Etnias
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> ○ Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> ○ Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.12 VER GALERÍAS

USR_CU_RSNAp_12: Ver Galerías	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Galerías
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> ○ Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> ○ Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.13 VER CONTENIDO

USR_CU_RSNAp_13: Ver Contenido	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Contenido
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> ○ Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> ○ Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.4.2.2.14 VER INFORMACIÓN

USR_CU_RSNAp_14: Ver Información	
Resumen:	Indicar al programa la base de datos que se conectará para el proyecto.
Prioridad:	Esencial
Actores directos:	Usuario
Escenarios	
Tipo de escenario	Descripción
Principal	Se deben establecer los valores de: <ul style="list-style-type: none"> ○ Información General
Secundario	Navegar <ul style="list-style-type: none"> El sistema permite navegar por el menú y desplegar la información al: <ul style="list-style-type: none"> ○ Usuario
Secundario	Consultar <ul style="list-style-type: none"> El sistema permite realizar consultas : <ul style="list-style-type: none"> ○ Usuario
Pre-condiciones	
Notas: El sistema deberá encontrar las distintas plantillas de bases de datos soportadas.	
Validaciones	
El sistema alertará en el caso que algún parámetro esté vacío o sea incorrecto	

3.5. DIAGRAMAS DE SECUENCIA

3.5.1 DIAGRAMA DE SECUENCIA PARA ADMINISTRADOR - GESTIÓN DE CONTENIDO E INFORMACIÓN

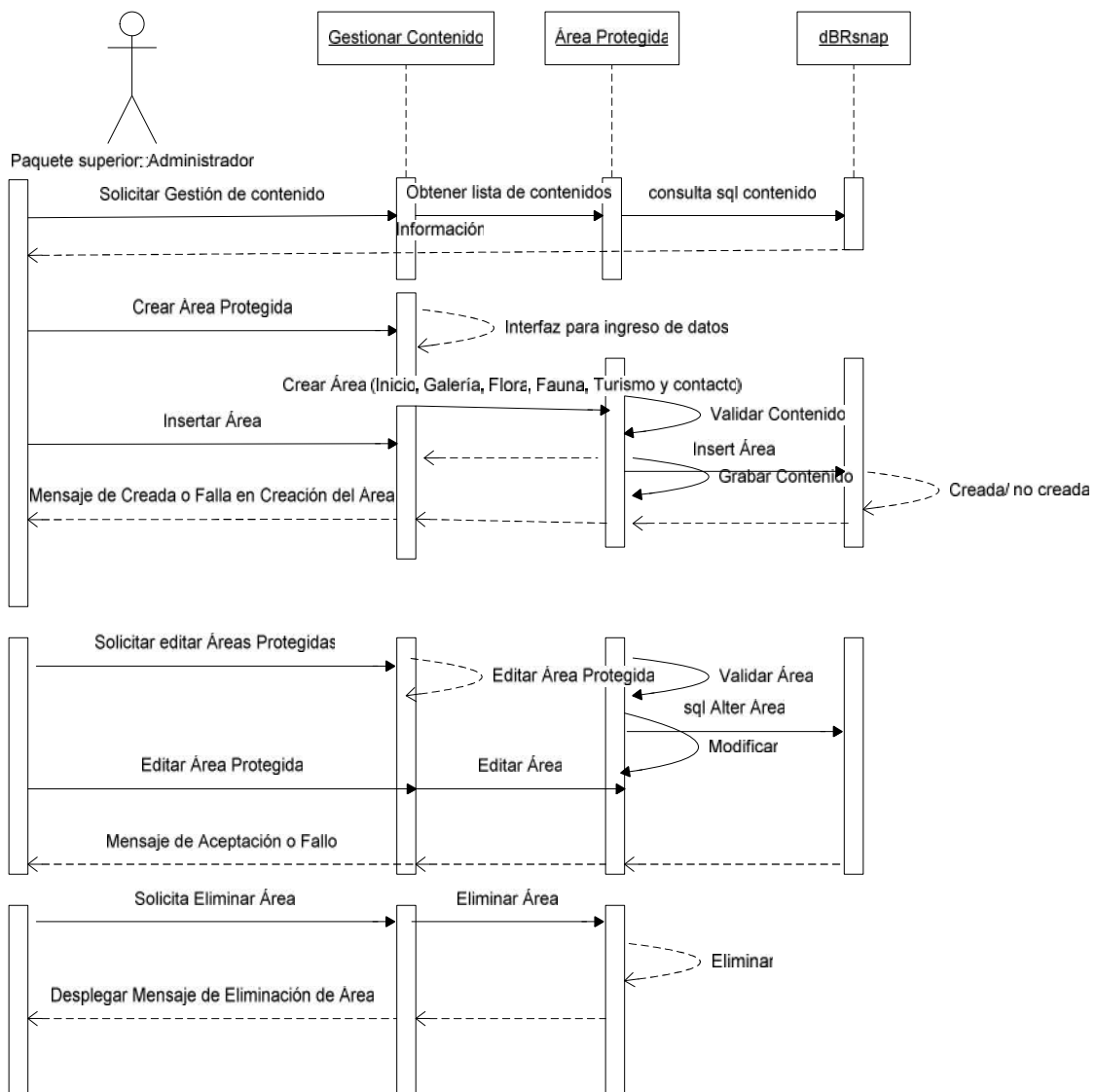


Figura3.3 Diagrama de Secuencia de Administrador.

3.5.2. DIAGRAMA DE SECUENCIA – USUARIO

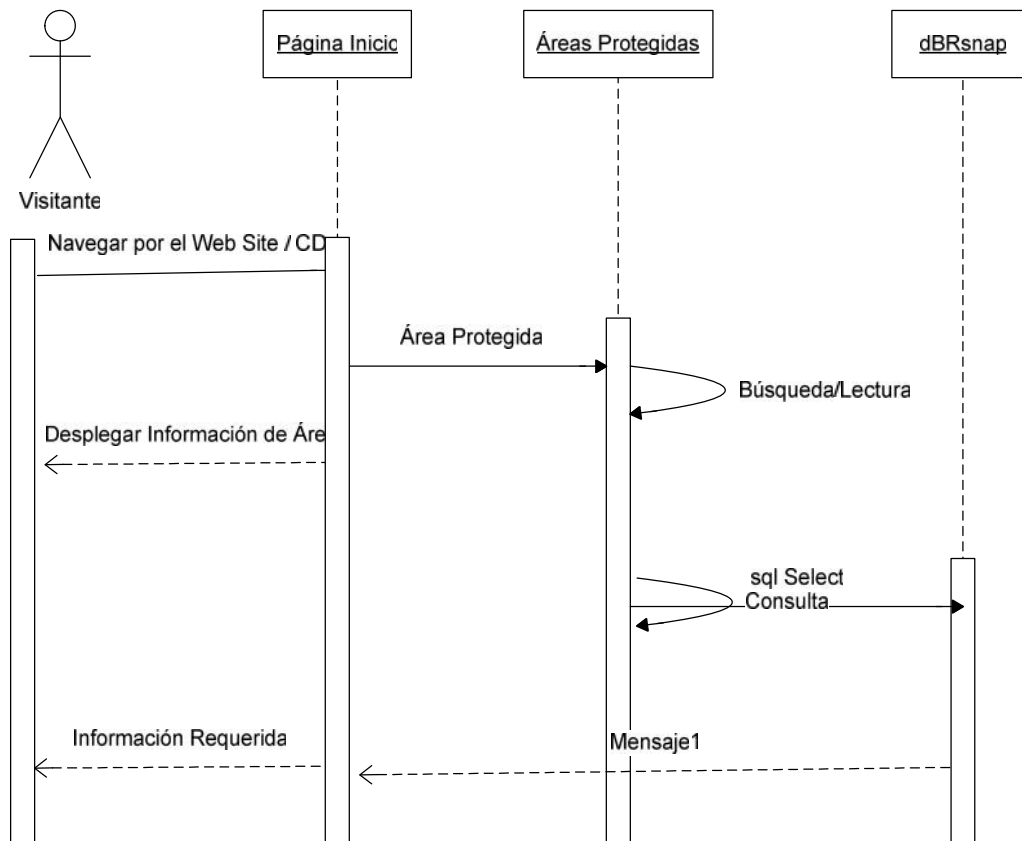


Figura 3. 4 Diagrama de Casos de Uso de Usuario.

3.6 DISEÑO CONCEPTUAL

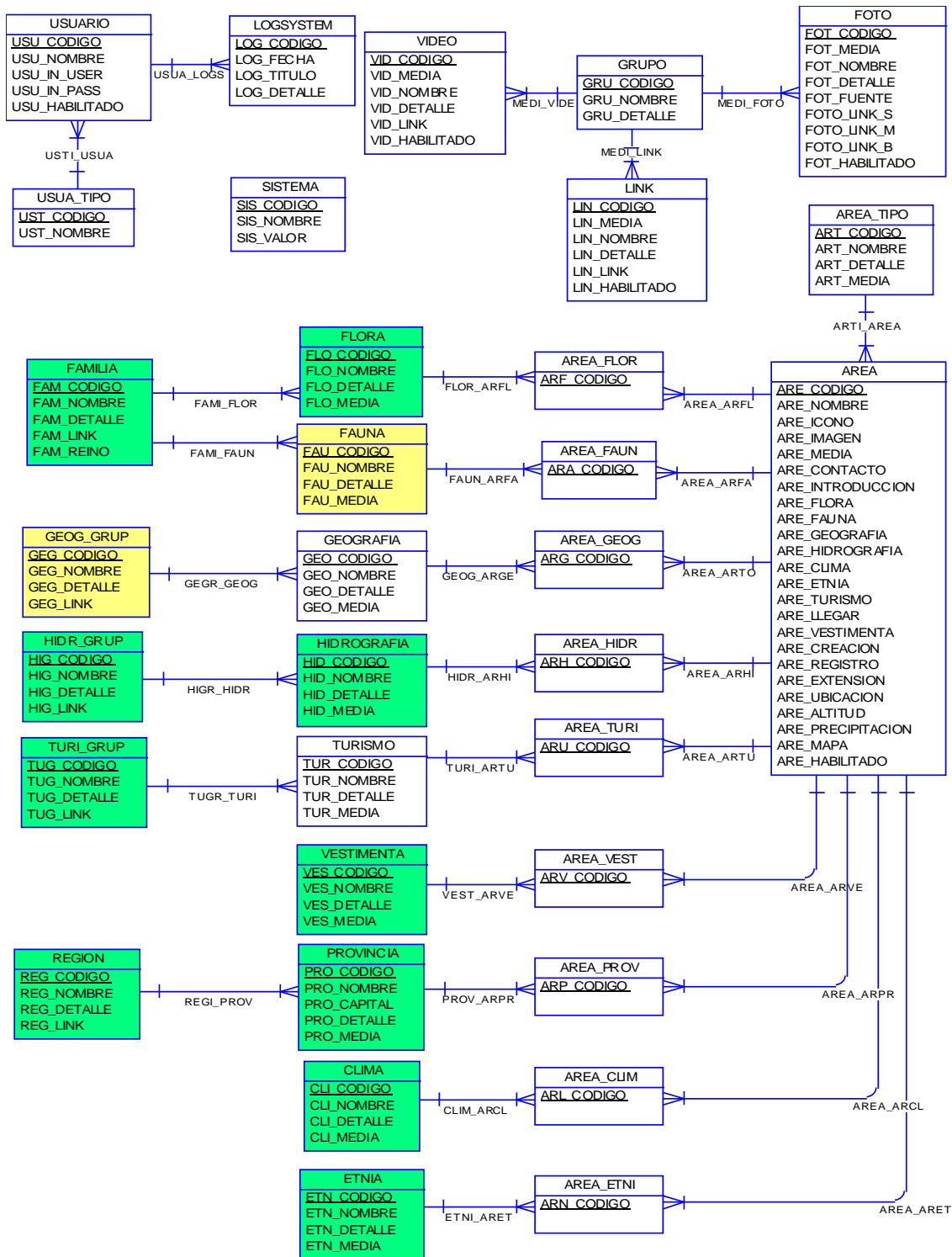


Figura 3. 5 Base de Datos del Sistema SNAP.

3.7 HISTORIAS DE USUARIO

3.7.1 HISTORIA 1 (H1)

Descripción:

El cliente solicita que en el Sistema aparezca el logo del proyecto y la fotografía que identifique a cada una de las páginas que se visite.

Historia de Usuario	
Numero: 1	Usuario : Programador
Nombre: Información Gráfica	
Prioridad : ALTA	Riesgo en Desarrollo: Bajo
Puntos Estimados : 10	Iteración asignada : 1
Programador responsable : Patricia Trujillo M.	
Descripción: El cliente solicita que se le presente diseños posibles para la página web, en donde se muestre el logo del proyecto y una fotografía para describir la información que se está viendo.	
Observaciones: Los datos necesarios serán proporcionados por el CIAM y por las investigaciones realizadas por la tesista.	

3.7.2 HISTORIA 2 (H2)

Descripción: Se necesita un Mapa de las zonas de Áreas Protegidas, que se muestre en alguna parte de las pantallas.

Historia de Usuario	
Numero: 2	Usuario : Programador
Nombre: Mapa de áreas protegidas	
Prioridad : ALTA	Riesgo en Desarrollo: MEDIA
Puntos Estimados : 4	Iteración asignada : 1
Programador responsable : Patricia Trujillo M.	
Descripción: Se necesita un Mapa de las zonas de Áreas Protegidas, que se muestre en alguna parte de las pantallas.	
Observaciones: El Mapa será proporcionado por el CIAM y modificado para el sistema.	

3.7.3 HISTORIA 3 (H3)

Descripción: El programa deberá tener como información básica de cada zona la extensión, ubicación y las especies de animales y plantas que habitan en dicha región, algo que hable del Proyecto y enseñe las Áreas. Y que la pantalla enseñe información General de cada parte que se abra y que tenga fotos de Flora y Fauna.

Historia de Usuario	
Numero: 3	Usuario : Programador
Nombre: Información Requerida por Áreas.	
Prioridad : ALTA	Riesgo en Desarrollo: MEDIA
Puntos Estimados : 10	Iteración asignada : 1
Programador responsable : Patricia Trujillo M.	
Descripción: El programa deberá tener como información básica de cada zona la extensión, ubicación y las especies de animales y plantas que habitan en dicha región, algo que hable del Proyecto y enseñe las Áreas. Y que la pantalla enseñe información General de cada parte que se abra y que tenga fotos de Flora y Fauna.	
Observaciones:	

3.7.4 HISTORIA 4 (H4)

Descripción: Se creará la base de datos a partir de los datos existentes en el CIAM, así como información que tengan publicada en las páginas del Ministerio del Ambiente y SNAP.

Historia de Usuario	
Numero: 4	Usuario : Programador
Nombre : Base de Datos	
Prioridad : ALTA	Riesgo en Desarrollo: Bajo
Puntos Estimados : 10	Iteración asignada : 2
Programador responsable : Patricia Trujillo M.	
Descripción: Se creará la base de datos a partir de los datos existentes en el CIAM, así como información que tengan publicada en las páginas del Ministerio del Ambiente y SNAP.	
Observaciones: La base de datos debe tener un campo para poder relacionar con las fotografías.	

3.7.5 HISTORIA 5 (H5)

Descripción: El sitio Web debe tener el mapa de áreas protegidas, la descripción de cada área, y las especies de flora y fauna.

Historia de Usuario	
Numero: 5	Usuario : Programador
Nombre : Información estática para Sitio Web	
Prioridad : ALTA	Riesgo en Desarrollo: ALTA
Puntos Estimados : 10	Iteración asignada : 3
Programador responsable : Patricia Trujillo M.	
Descripción: El sitio Web debe tener una barra con la descripción de cada área, que es la información que se presenta en el Ministerio del Ambiente, y las especies de flora y fauna, tipo Wikipedia.	
Observaciones:	

3.7.6 HISTORIA 6 (H6)

Descripción: Una vez obtenida la información se creará la aplicación que servirá para el sitio web.

Historia de Usuario	
Numero: 6	Usuario : Programador
Nombre : Formato Página Web	
Prioridad : ALTA	Riesgo en Desarrollo: ALTA
Puntos Estimados : 10	Iteración asignada : 3
Programador responsable : Patricia Trujillo M.	
Descripción: El sitio Web debe tener fondo verde para que Identifique que es un sitio Ecológico.	
Observaciones:	

3.7.7 HISTORIA 7 (H7)

Descripción: Las ayudas, facilitarán al usuario navegar por el sitio Web y acceder a la información de la Guía interactiva guardada en el CD o DVD.

Historia de Usuario	
Numero: 7	Usuario : Programador
Nombre : Contenidos	
Prioridad : BAJA	Riesgo en Desarrollo: BAJA
Puntos Estimados : 7	Iteración asignada : 4
Programador responsable : Patricia Trujillo M.	
Descripción: La información que se debe presentar es la que aparece en la página del Ministerio del Ambiente, añadiendo la información de Flora y Fauna para cada Área.	
Observaciones:	

3.7.8 HISTORIA 8 (H8)

Descripción: Se necesita que la información esté guardada en un CD o DVD, para permitir al usuario acceder al Mapa, elegir el área protegida y acceder a la información del área, así como las especies de flora y fauna que habitan en dicha área,

Historia de Usuario	
Numero: 8	Usuario : Programador
Nombre : CD	
Prioridad : ALTA	Riesgo en Desarrollo: ALTA
Puntos Estimados : 10	Iteración asignada : 5
Programador responsable : Patricia Trujillo M.	
Descripción: Se necesita que la información esté guardada en un CD o DVD, para permitir al usuario ver las áreas protegidas y acceder a la información del área, y ver las especies de flora y fauna que habitan en dicha área.	
Observaciones: El CD debe tener las fotografías de las especies de flora y fauna.	

3.7.9 TABLA DE HISTORIAS DE USUARIO

Núm	Nombre	Prioridad	Riesg o	Esfuerz o	Iteració n
1	DATOS DEL SISTEMA				1
2	MAPA DE ÁREAS PROTEGIDAS				1
3	INFORMACIÓN BÁSICA				1
4	BASE DE DATOS				2
5	INFORMACIÓN PARA SITIO WEB				3
6	SITIO WEB				3
7	AYUDAS				4
8	CD				5

Figura 3. 6 Tabla con Listado de Historias de Usuario.

3.9 PLANTEAMIENTO

3.9.1 ITERACIÓN PRIMERA: Se realiza una reunión preliminar para definir los requerimientos del usuario, recopilar la información necesaria, definir características de la página Web y la información que deberá contener.

TAREA 1

TAREA	
Número de TAREA : 1	Número de Historia : 6
Nombre de TAREA : Requerimientos para el Web Site	
Tipo de TAREA: Investigación	Puntos Estimados : 5
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Buscar algunas plantillas de Sitios Web, para presentar al Cliente y que escoja cuál le gusta.	

Figura 3. 8 TAREA 1 para la Iteración 1

TAREA 2

TAREA	
Número de TAREA : 2	Número de Historia : 1,2
Nombre de TAREA: Agregar al sitio Web un logo, mapa y fotografías.	
Tipo de TAREA: Investigación	Puntos Estimados : 5
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Realizar un logo y un mapa que sea agradable al cliente y buscar imágenes referentes a Áreas Protegidas.	

Figura 3. 9 TAREA 2 para la Iteración 1

3.9.2 ITERACIÓN SEGUNDA: Se depurará la información de la Base de Datos, y se creará el nuevo diseño de la Base de Datos compatible con el Sistema, con la información necesaria.

TAREA 1

TAREA	
Número de TAREA : 1	Número de Historia : 4
Nombre de TAREA : Creación de base de datos	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Implementar la base de datos con la información perteneciente a las áreas protegidas de la región amazónica.	

Figura 3. 10 TAREA 1 para la Iteración 2

TAREA 2

TAREA	
Número de TAREA : 2	Número de Historia : 4
Nombre de TAREA : Ingreso de información	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Ingresar la información detallada en los diferentes campos pertenecientes a las clases de la Base de datos.	

Figura 3. 11 TAREA 2 para la Iteración 2

TAREA 3

TAREA	
Número de TAREA : 3	Número de Historia : 4
Nombre de TAREA : Relacionar las clases de la base de datos	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Relacionar las clases pertenecientes a la base de datos para áreas protegidas tomando en cuenta las descripciones proporcionadas por el CIAM.	

Figura 3. 12 TAREA 3 para la Iteración 2

TAREA 4

TAREA	
Número de TAREA : 4	Número de Historia : 4
Nombre de TAREA : Pruebas de la base de datos	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Realizar pruebas de la base de datos creada.	

Figura 3. 13 TAREA 4 para la Iteración 2

3.9.3 ITERACIÓN TERCERA: Se buscará información referente a Áreas Protegidas, para presentarle al cliente y decida qué información va en la página web.

TAREA 1

TAREA	
Número de TAREA : 1	Número de Historia : 3
Nombre de TAREA: Información requerida por Áreas.	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Se presentará varias opciones con la información básica de las Áreas Protegidas para que el cliente decida el formato que quedará para el web Site.	

Figura 3. 14 TAREA 1 para la Iteración 3

TAREA 2

TAREA	
Número de TAREA : 2	Número de Historia : 5
Nombre de TAREA : Información estática de la página web	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Se decidirá junto con el cliente que contenido se colocará como información estática de la página web.	

Figura 3. 15 TAREA 2 para la Iteración 3

3.9.4 ITERACIÓN CUARTA: Se analizará la información del sitio web del Ministerio del Ambiente, para definir que contenidos se colocarán en la página web del SNAP.

TAREA 1

TAREA	
Número de TAREA : 1	Número de Historia : 4
Nombre de TAREA : Contenidos para la Página Web	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Crear los contenidos para la barra informativa de la página web.	

Figura 3. 16 TAREA 1 para la Iteración 4

3.9.5 ITERACIÓN QUINTA: Se creará un CD con la información de las Áreas Protegidas de la Región Amazónica.

TAREA 1

TAREA	
Número de TAREA : 1	Número de Historia : 5
Nombre de TAREA : Diseño de CD	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
<p>Descripción:</p> <p>Diseñar con la información depurada de la base de datos los contenidos para el CD de Áreas Protegidas, tanto informativos como fotográficos. Seleccionar el diseño que permita cumplir con los requerimientos del CIAM.</p>	

Figura 3. 17 TAREA 1 para la Iteración 5

TAREA 2

TAREA	
Número de TAREA : 2	Número de Historia : 5
Nombre de TAREA : Creación de CD	
Tipo de TAREA: Desarrollo	Puntos Estimados :
Fecha de Inicio :	Fecha Fin :
Programador responsable : Patricia Trujillo M.	
Descripción: Crear el CD o DVD con la información de Áreas Protegidas.	

Figura 3. 18 TAREA 2 para la Iteración 5

3.10 INCIDENCIAS

- Cada una de las iteraciones fueron especificadas para realizar la Guía interactiva multimedia de áreas protegidas.
- Se utilizó como medio de información adicional la información correspondiente a la página web del Ministerio del ambiente.

3.11 TARJETAS C.R.C.

Mapa de Áreas Protegidas	
<ul style="list-style-type: none">• Archivo que contiene el Mapa en las diferentes áreas protegidas.• En el Mapa se debe tomar en cuenta solo las zonas que comprenden las áreas protegidas del Ecuador.	

Figura 3. 19 Tarjeta “Mapa Áreas Protegidas”

Base de Datos	
<ul style="list-style-type: none"> • Creación de base de datos. • Establece las entidades y relaciones de la base de datos. • Las clases son : <ul style="list-style-type: none"> ○ USUA_TIPO ○ USUARIO ○ LOGSYSTEM ○ SISTEMA ○ VIDEO ○ GRUPO ○ LINK ○ FOTO ○ ÁREA_TIPO ○ ÁREA ○ FAMILIA ○ FLORA ○ FAUNA ○ ÁREA_FLOR ○ ÁREA_FAUN ○ GEOG_GRUP ○ GEOGRAFÍA ○ ÁREA_GEOG ○ HIDR_GRUP ○ HIDROGRAFIA ○ ÁREA_HIDR ○ TURI_GRUP ○ TURISMO ○ ÁREA_TURI ○ REGION ○ VESTIMENTA ○ ÁREA_VEST ○ PROVINCIA ○ ÁREA_PROV ○ CLIMA ○ ÁREA_CLIM ○ ETNIA ○ ÁREA_ETNI • Retorna la base de datos. 	

Figura 3. 20 Tarjeta “Base de Datos”

Página Web	
<ul style="list-style-type: none">• Diseño del Sitio Web.• Se debe tomar en cuenta los siguientes requerimientos:<ul style="list-style-type: none">○ Información básica del área protegida.○ Información de especies existentes de Flora y Fauna de cada área.○ Sitios de interés turísticos• Esta página deberá ser creada con el programa “DreamWeaver” y para la gestión de la misma se utiliza el programa “Crea Cod”.	

Figura 3. 21 Tarjeta “Página Web”

CD	
<ul style="list-style-type: none">• Diseño del CD.• Se debe tomar en cuenta los siguientes requerimientos:<ul style="list-style-type: none">○ Información básica del área protegida.○ Información de especies existentes de Flora y Fauna de cada área.○ Cada clasificación de especie debe estar sub clasificada y asociada a una fotografía.• Esta página deberá ser creada con el programa “MiniCreator”.	

Figura 3. 22 Tarjeta “CD”

3.8 PROTOTIPOS

3.8.1 PANTALLA PRINCIPAL

USUARIO	Administrador
PASSWORD	*****

Figura 3. 23 Prototipo para pantalla de acceso

**ESPACIO EN
BLANCO
INTENCIONAL**

3.8.2 CREACIÓN DE CLASES

NOMBRE DE CLASE				
Información 1				
Información 2				
Información 3				
VER	AGREGAR	EDITAR	ELIMINAR	REFRESCAR

Figura 3. 24 Prototipo de las clases del Sistema.

**ESPACIO EN
BLANCO
INTENCIONAL**

3.8.3 PÁGINA WEB PRINCIPAL

CIAM Programa de Áreas Protegidas	
Información General	
¿Quienes somos?	
Áreas Protegidas	
Flora	
Fauna	

Figura 3. 25 Prototipo de la página Web Principal.

**ESPACIO EN
BLANCO
INTENCIONAL**

3.8.4 PÁGINA DE INFORMACIÓN SOBRE ÁREAS PROTEGIDAS

NOMBRE DE ÁREA PROTEGIDA			
Información General		Descripción:	
¿Quiénes somos?			
Áreas Protegidas			
Flora			
Fauna			
		FLORA	FAUNA

Figura 3. 26 Prototipo De página Web con información de Áreas Protegidas

3.8.5 PANTALLA DE FLORA


CIAM Programa de Áreas Protegidas FLORA	
Información General	Descripción :
¿Quienes somos?	
Áreas Protegidas	
Flora	
Fauna	
	

Figura 3. 27 Prototipo de Pantalla de Flora

3.8.5.1 PANTALLA DE INFORMACIÓN DE FLORA



CIAM Programa de Áreas Protegidas FLORA	
Información General	 
¿Quienes somos?	
Áreas Protegidas	
Flora	
Fauna	

Figura 3. 28 Prototipo de información sobre Flora.

3.8.5.2 GALERÍA FOTOGRÁFICA



Figura 3. 29 Prototipo Galería fotográfica de Flora

3.8.6 PANTALLA DE FAUNA



CIAM Programa de Áreas Protegidas FAUNA	
Información General	 
¿Quienes somos?	
Áreas Protegidas	
Flora	
Fauna	

Figura 3. 30 Prototipo de Pantalla de Fauna.

3.8.6.1 PANTALLA DE GALERÍA FOTOGRÁFICA

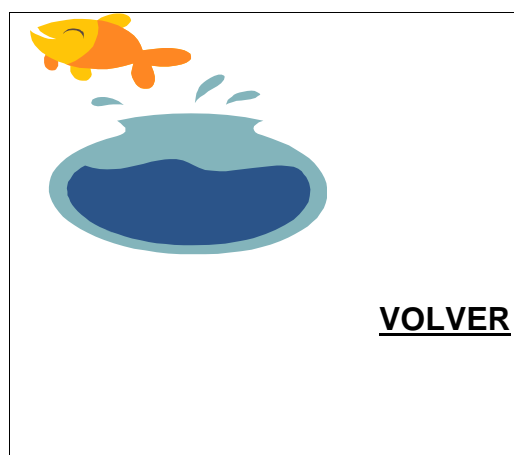


Figura 3. 31 Prototipo de Galería Fotográfica de Fauna.

3.9 CREACIÓN DEL PROYECTO

A continuación se detalla los pasos realizados para elaborar el proyecto:

3.9.1 CREACIÓN DE LA BASE DE DATOS

Para crear la base de datos se utiliza el programa MySQL Font, el cual es un Ide de MySQL, se abre el programa, se selecciona SQL Editor, en la opción File, se selecciona ImportSql File.

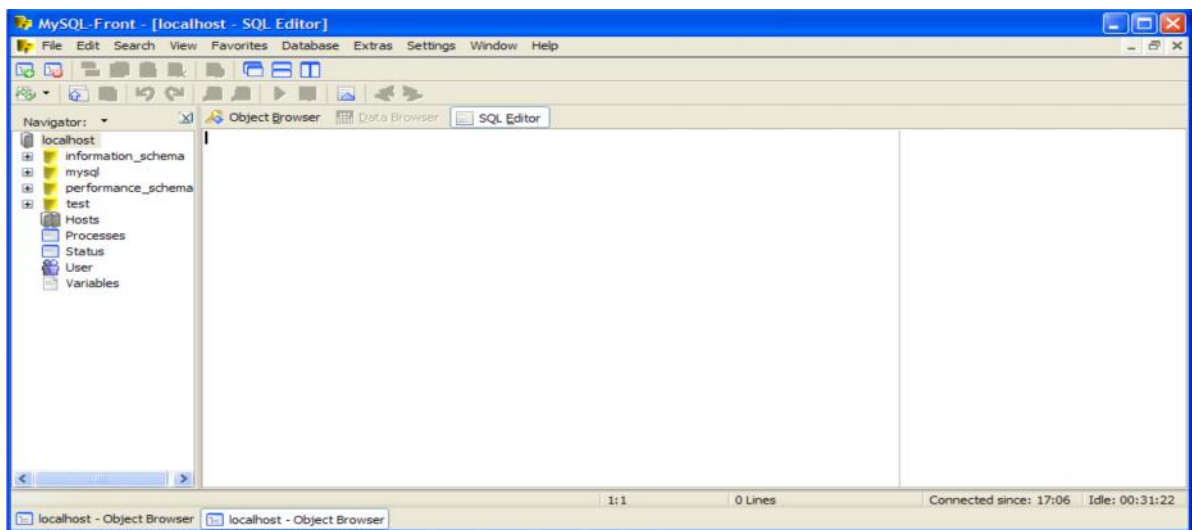


Figura 3. 32 Pantalla de MySQL Front, para generación de base de datos.

3.9.1.2 CUADRO DE ESTADO DE LA IMPORTACIÓN DE LA BASE DE DATOS EN MYSQL FRONT.

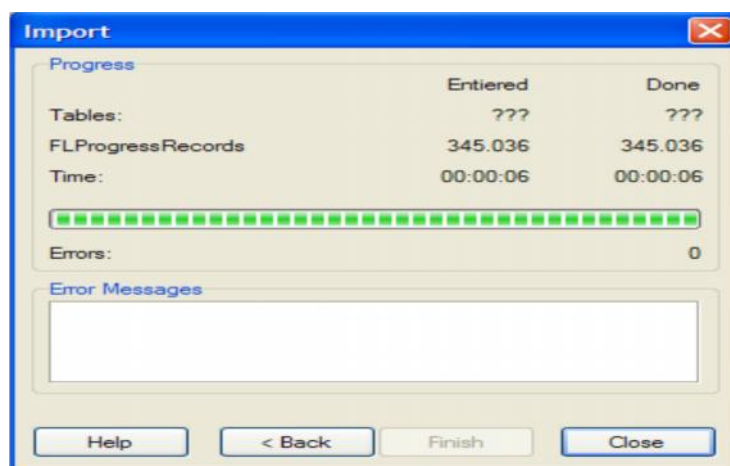


Figura 3. 33 Estado de la Importación.

3.9.1.3 PANTALLA CON LA BASE DE DATOS CREADA.

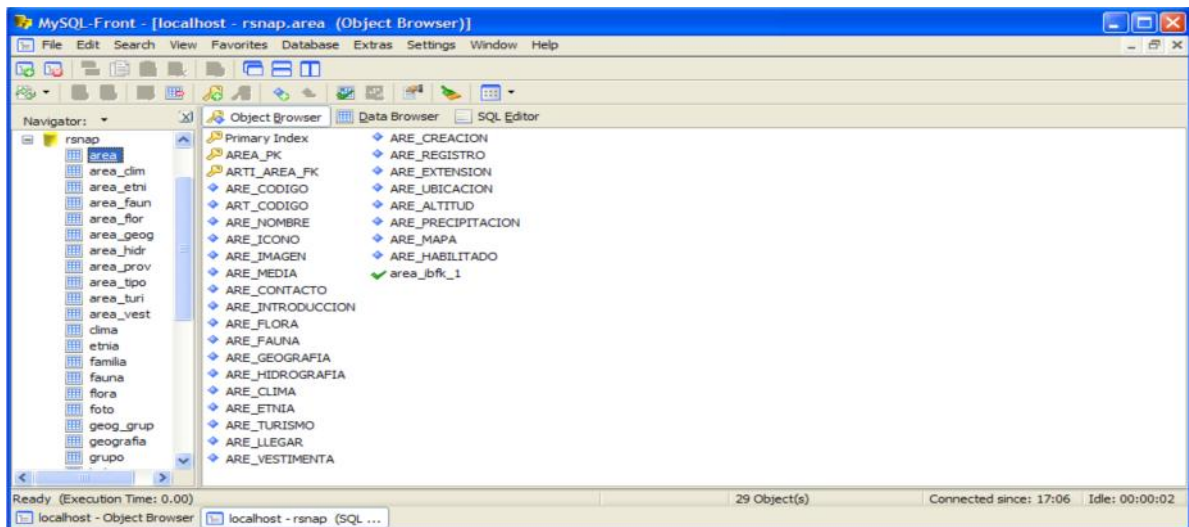


Figura 3. 34 Base de Datos Creada en MySQL Front.

3.9.1.4 CAMPOS DE LA BASE DE DATOS

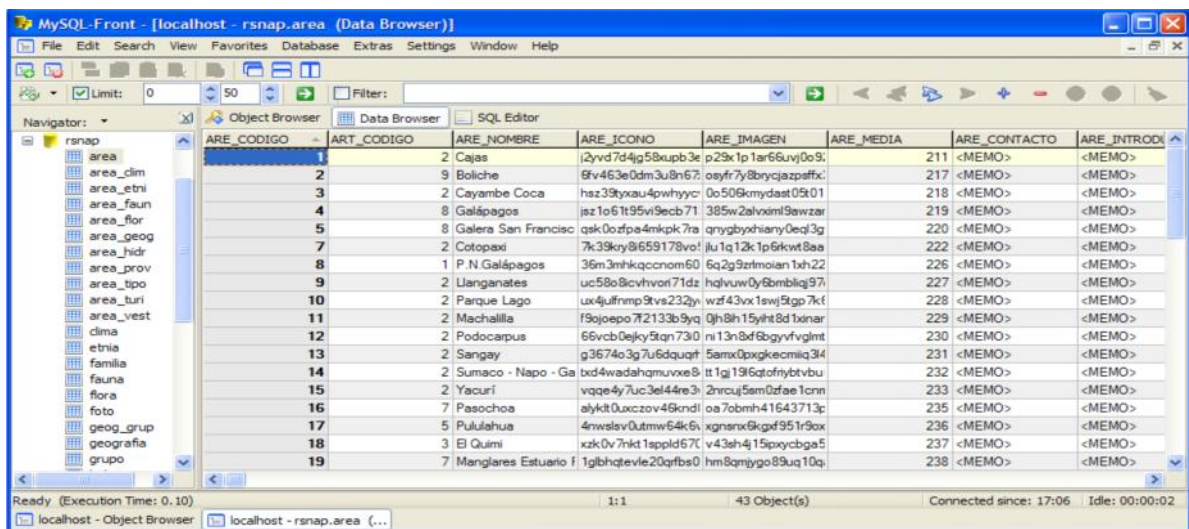


Figura 3. 35 Campos de la Base de datos creada en MySQL Front.

3.9.2 CREACIÓN DE LAS PANTALLAS DE CONTROL USANDO EL PROGRAMA CREACOD.

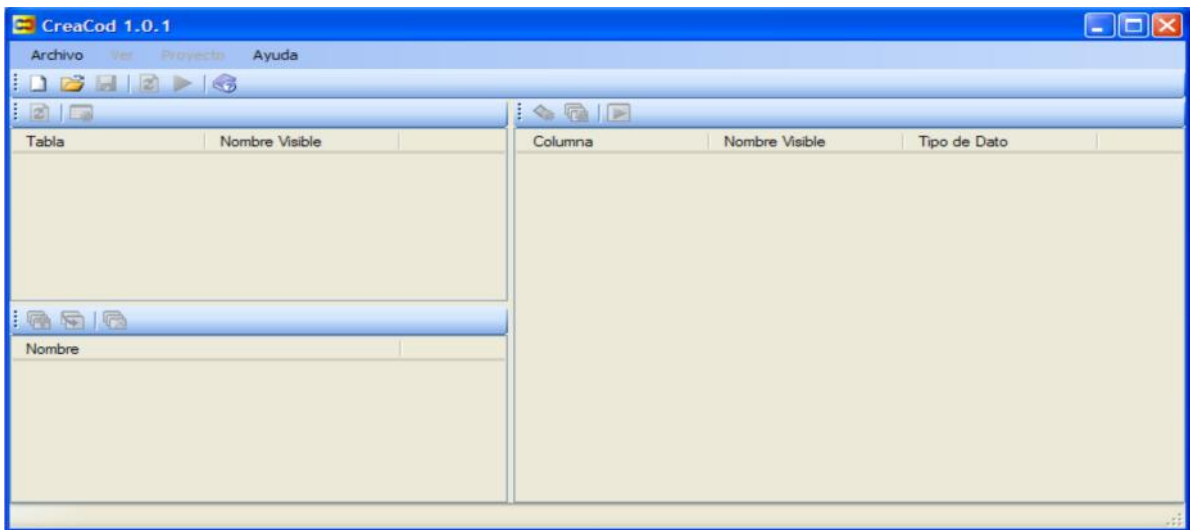


Figura 3. 36 Pantalla inicial del programa Crea Cod.

3.9.2.1 REGISTRO DE LA INFORMACIÓN PARA LA CREACIÓN DEL NUEVO PROYECTO EN EL PROGRAMA CREACOD.

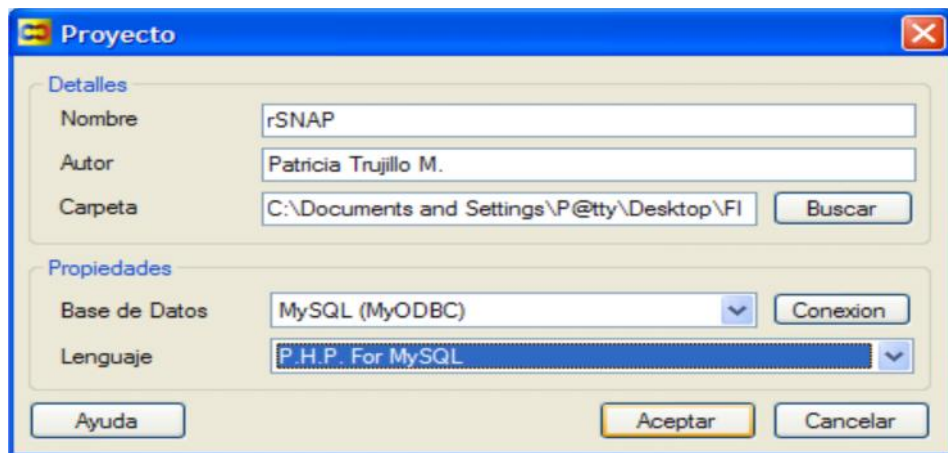


Figura 3. 37 Pantalla de registro para crear un nuevo proyecto.

3.9.2.2 PANTALLA DE CONEXIÓN A LA BASE DE DATOS CON CREACOD.

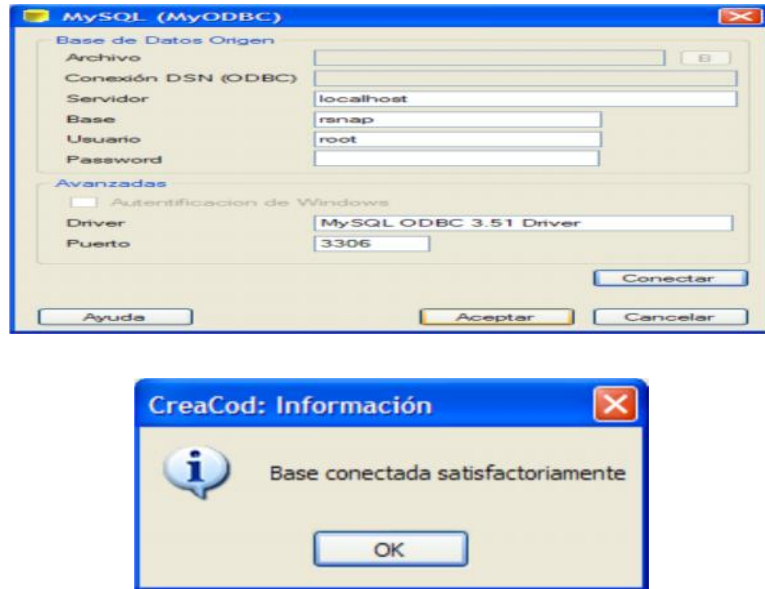


Figura 3. 38 Pantalla de Conexión con la Base de Datos

3.9.2.3 CONFIGURACIÓN DE PARÁMETROS DE LA BASE DE DATOS.

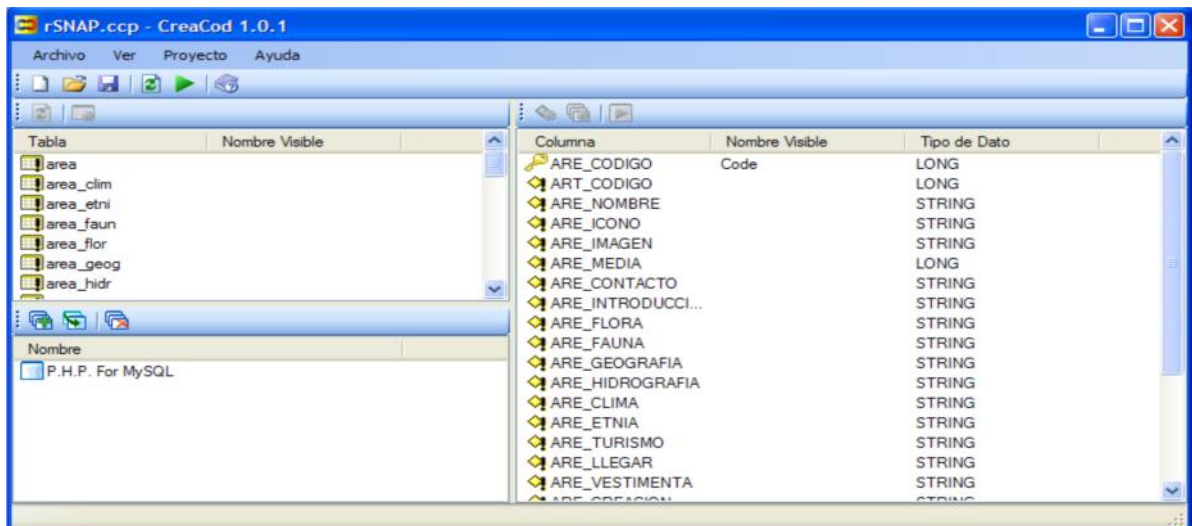


Figura 3. 39 Pantalla de configuración de los campos de la Base de Datos.

3.9.2.4 TABLA DE CONFIGURACIÓN DE LAS PROPIEDADES DE LA BASE DE DATOS.

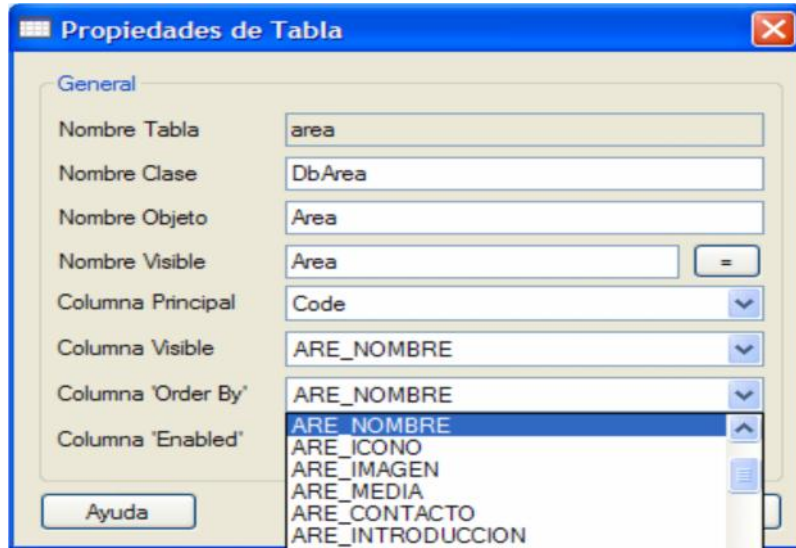


Figura 3. 40 Pantalla de configuración de propiedades de los campos de la Base de Datos.

3.9.2.5 PANTALLA DE PROPIEDADES DE TABLA.

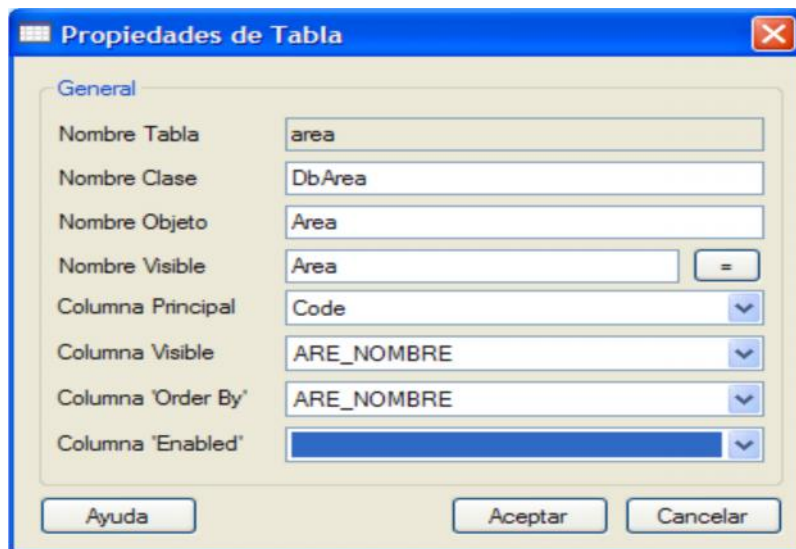
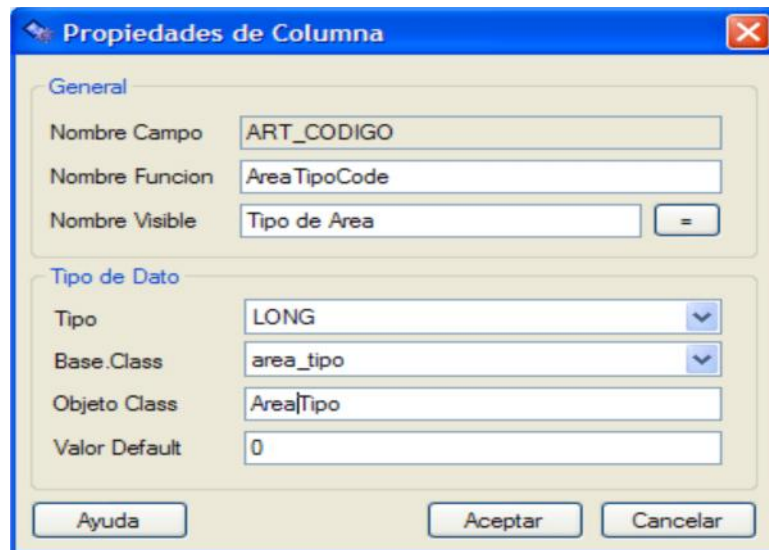


Figura 3. 41 Pantalla de configuración de propiedades de Tabla.

3.9.2.6 PANTALLA DE CREACIÓN DE PROPIEDADES DE COLUMNA.



The screenshot shows a dialog box titled "Propiedades de Columna" with a close button in the top right corner. It is divided into two sections: "General" and "Tipo de Dato".

General section:

- Nombre Campo: ART_CODIGO
- Nombre Funcion: AreaTipoCode
- Nombre Visible: Tipo de Area (with an equals sign button to its right)

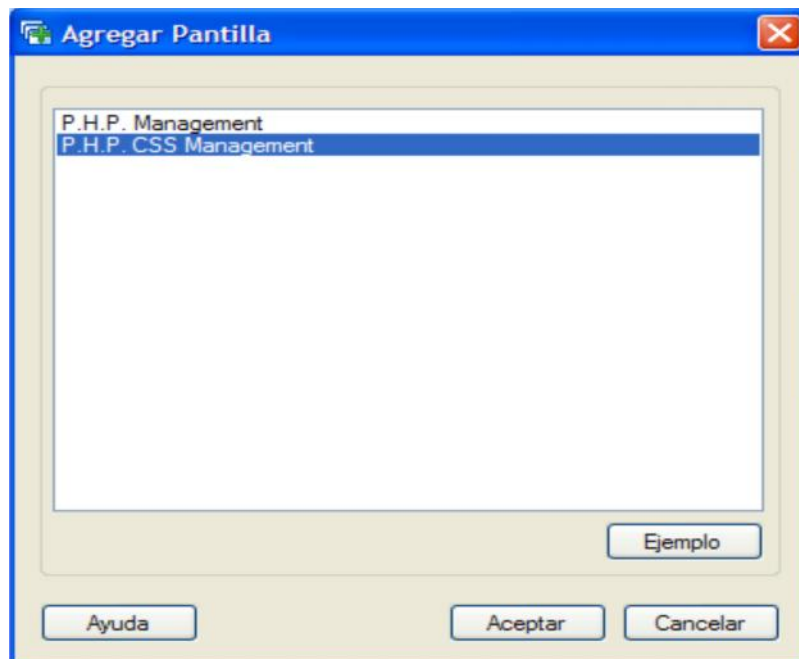
Tipo de Dato section:

- Tipo: LONG (dropdown menu)
- Base.Class: area_tipo (dropdown menu)
- Objeto Class: AreaTipo
- Valor Default: 0

At the bottom, there are three buttons: "Ayuda", "Aceptar", and "Cancelar".

Figura 3. 42 Pantalla de Propiedades de Columna.

3.9.2.7 PANTALLA DE SELECCIÓN DE LA PLANTILLA DE GESTIÓN DE INFORMACIÓN.



The screenshot shows a dialog box titled "Agregar Pantilla" with a close button in the top right corner. It features a list box containing two items:

- P.H.P. Management
- P.H.P. CSS Management (highlighted with a blue background)

Below the list box is a button labeled "Ejemplo". At the bottom of the dialog, there are three buttons: "Ayuda", "Aceptar", and "Cancelar".

Figura 3. 43 Pantalla de Creación de Plantilla de Gestión de Información

3.9.2.8 PANTALLA DE CONFIGURACIÓN DE CONTROLES DE PLANTILLA DE GESTIÓN.

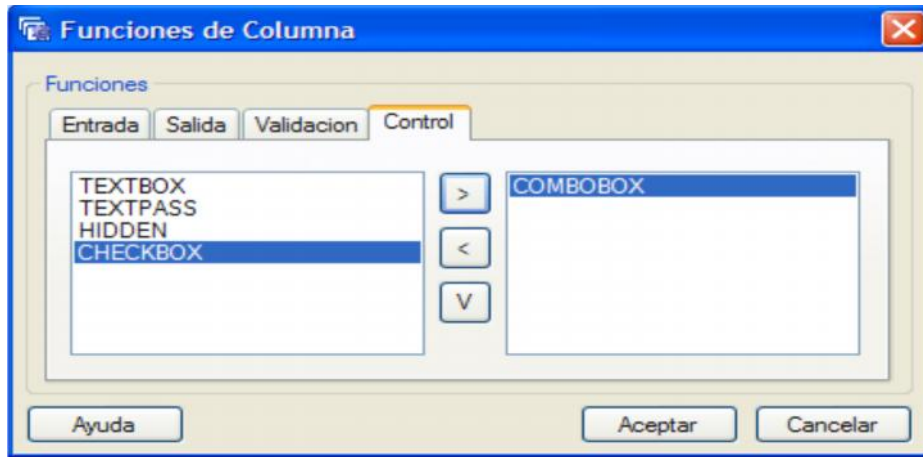


Figura 3. 44 Pantalla de configuración de Controles.

3.9.2.9 PANTALLA DE PROYECTO FINALIZADO CON TODAS LAS CONFIGURACIONES REALIZADAS.

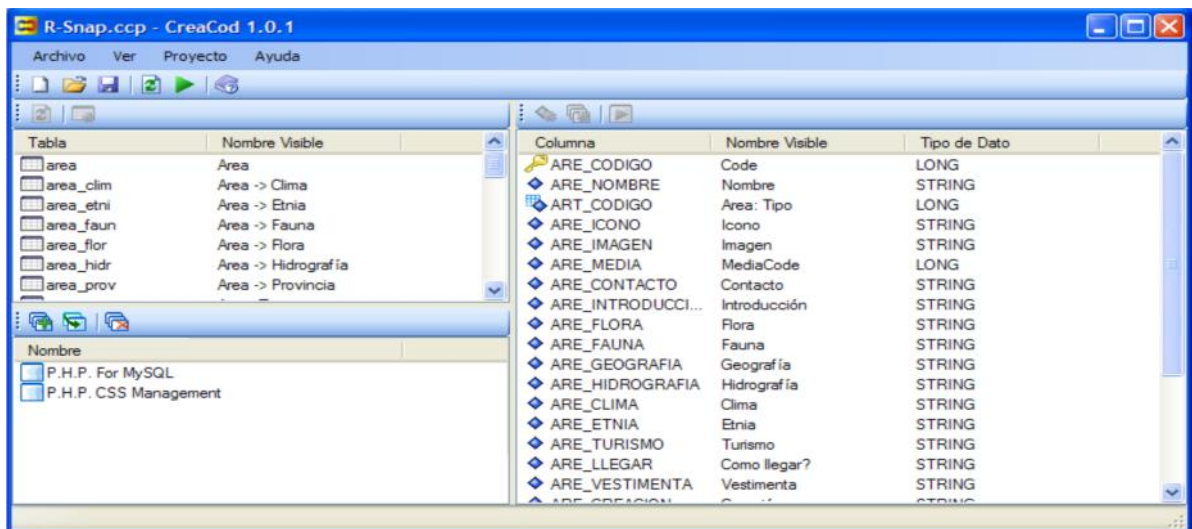


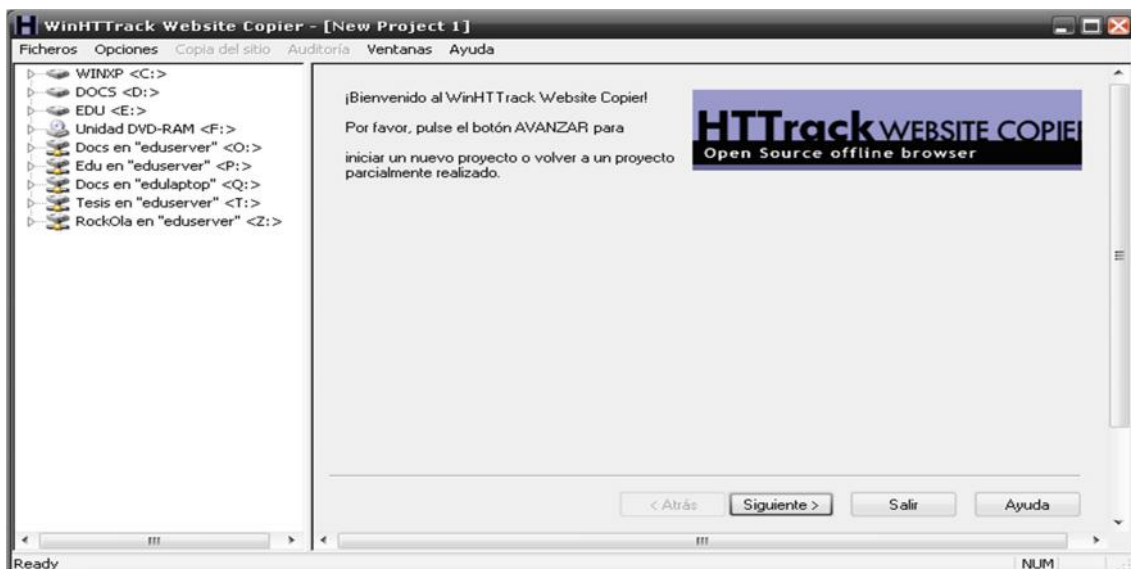
Figura 3. 45 Pantalla Final del Proyecto Configurado.

3.9.2.10 CREACIÓN DEL CD MULTIMEDIA

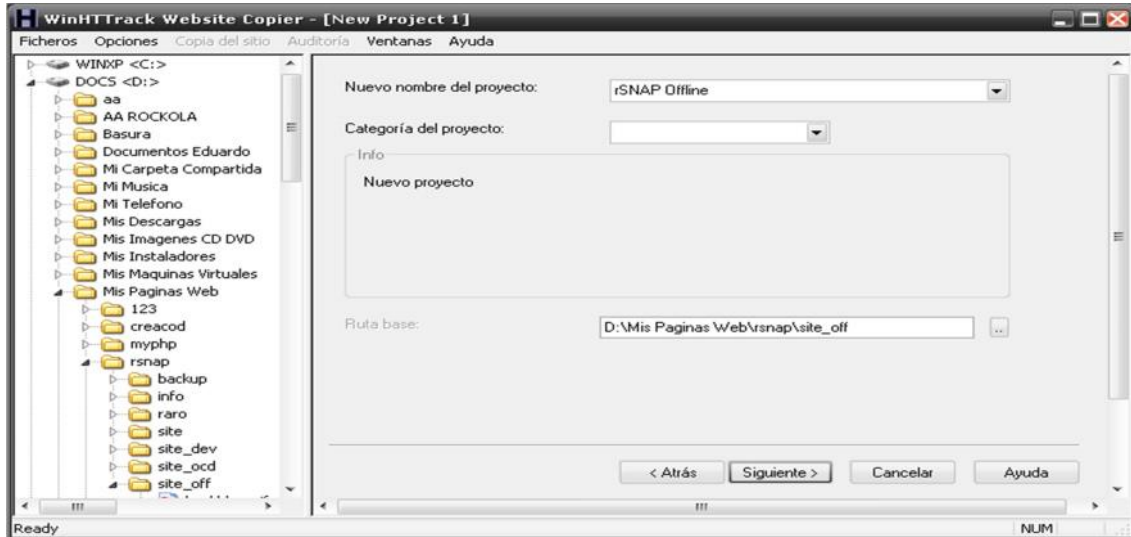
Abrir el sitio web de rSnap y copiar la dirección URL



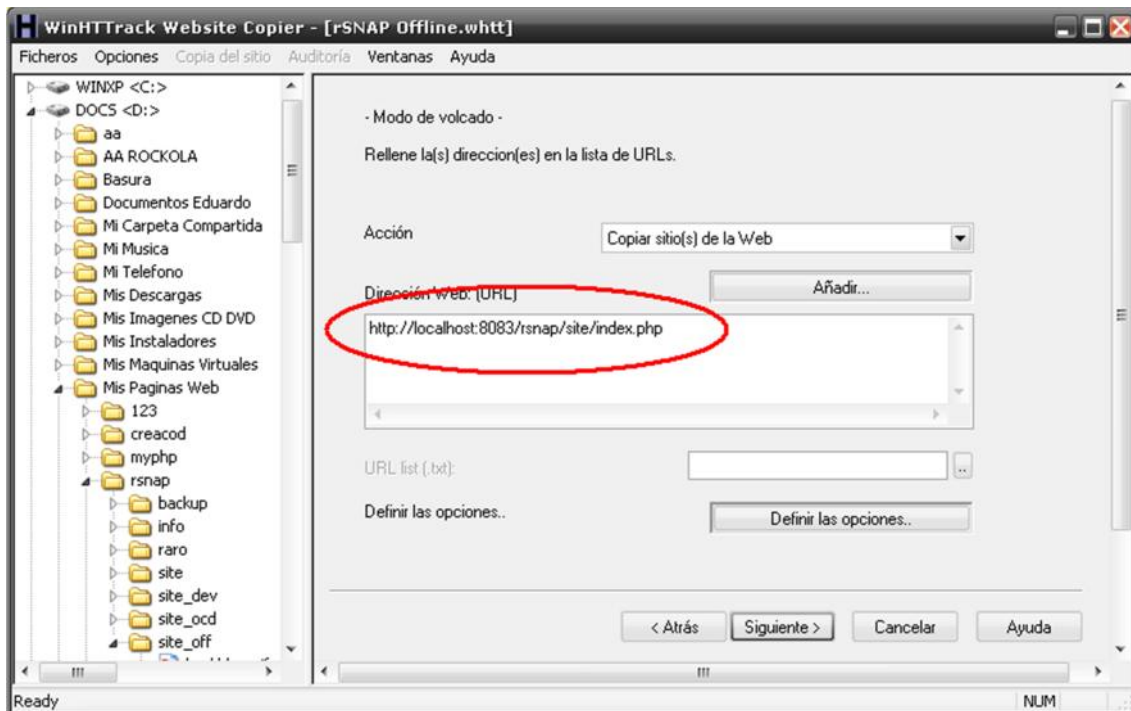
Abrir el programa WinHTTrackWebsiteCopier, presionar botón “Siguiete”



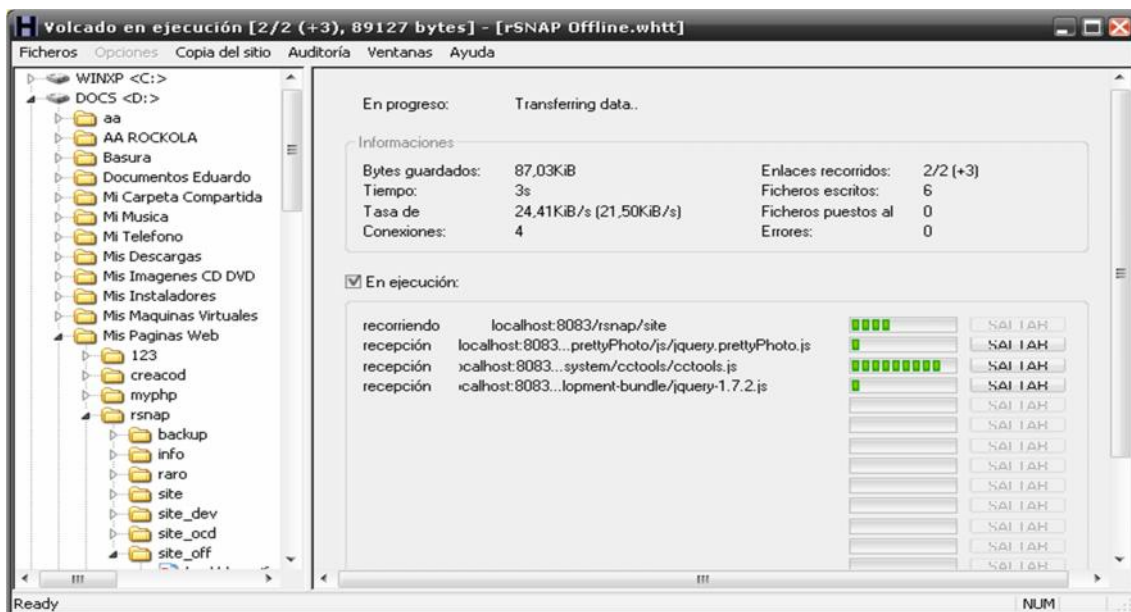
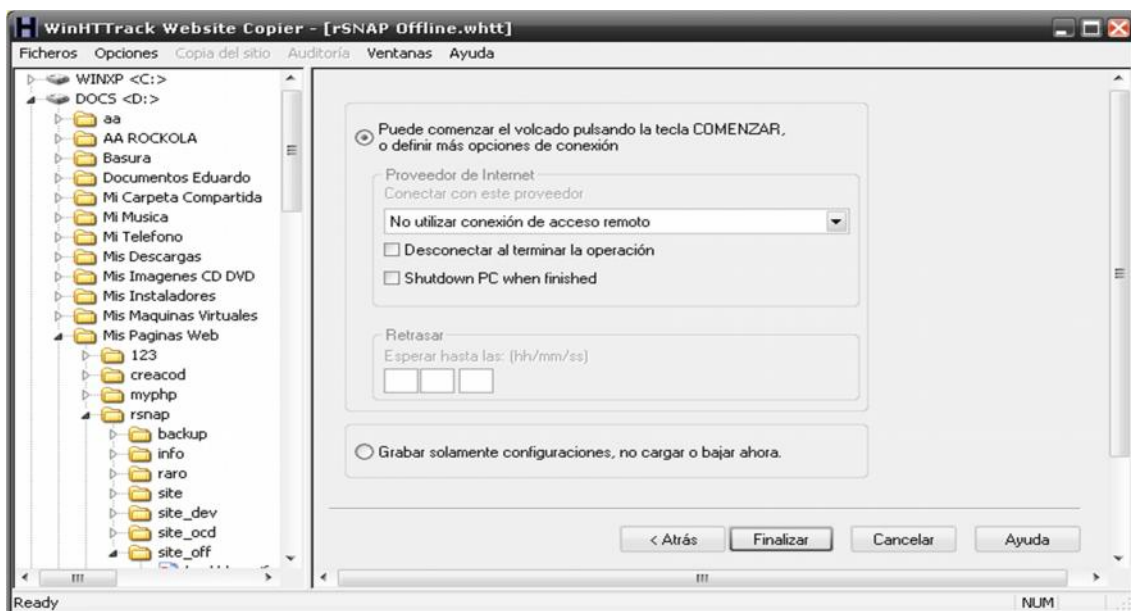
Se debe poner un nombre al proyecto y la carpeta donde se almacenarán los archivos, presionar el botón “Siguiente”.



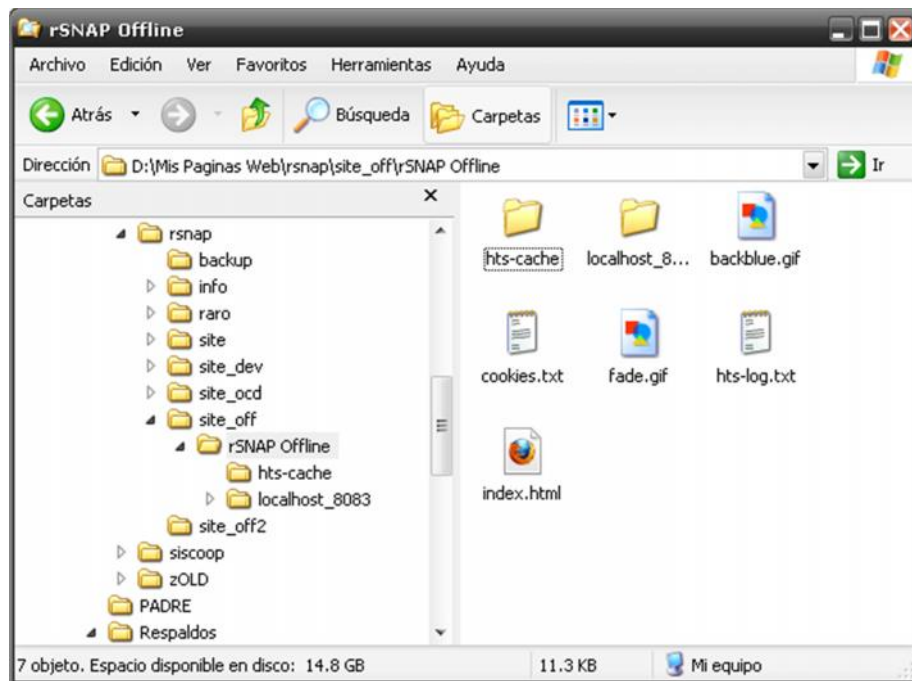
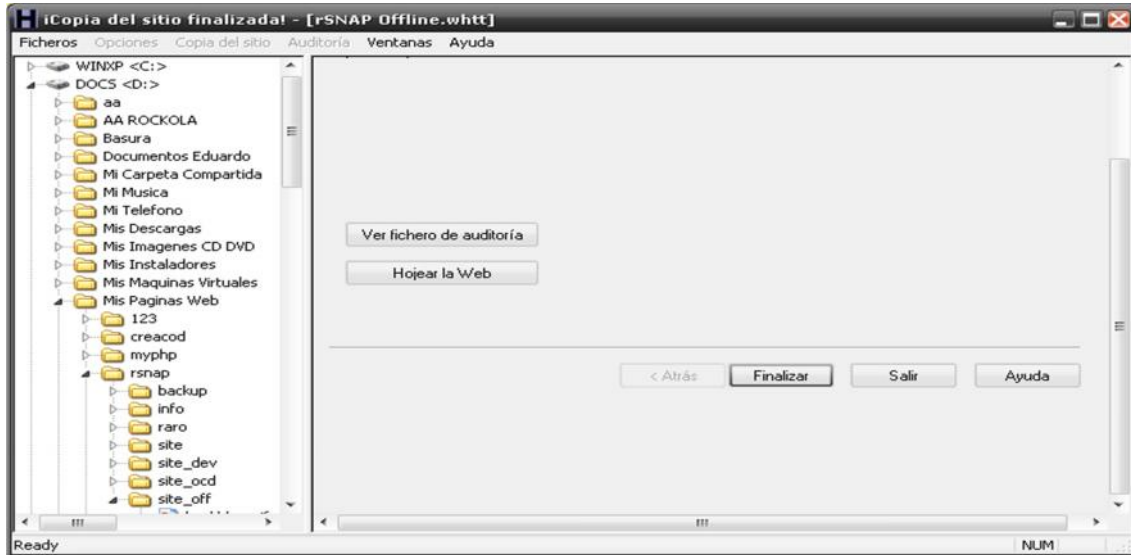
Se coloca la dirección URL del sitio web, presionar el botón “Siguiente”



Presionar el botón "Iniciar", el sistema empezará a crear el CD offline



El sistema informará cuando se haya completado la transferencia de archivos, ir a la carpeta destino del proyecto y copiar los archivos generados a un CD/DVD.



3.9.2 PRUEBAS DE INTERFAZ DE USUARIO

En esta prueba se validó la facilidad de uso del Web Site, grado de facilidad en interacción con el usuario, se evaluaron el menú horizontal y la barra vertical con la información disponible en el portal, la estética de las pantallas, la resolución de las imágenes, el contenido de cada ítem.

3.9.3 PRUEBAS DE NAVEGACIÓN

Para realizar las Pruebas de Navegación, se ha analizado con los usuarios del CIAM, que los vínculos creados funcionen adecuadamente y concuerden con el contenido, y que todos los enlaces sean amigables y comprensibles para el usuario.

3.9.4 PRUEBAS DE COMPONENTES

Las Pruebas de Componentes, han permitido realizar pruebas con ingreso de información en las pantallas de Gestión, validando los datos existentes y los nuevos para cada tipo de dato ingresado, la información se guarda en el servidor sin inconvenientes y no existe pérdida de datos.

3.9.5 PRUEBAS DE CONFIGURACIÓN

Se ha analizado los dos tipos de conflictos que podrían existir en una aplicación Web y se darían en el Cliente y en el Servidor.

Los Clientes (Usuarios), han utilizado el sistema en los siguientes navegadores: Mozilla Firefox 14.0.1, Internet Explorer 9.0, Google Chrome

- Luego de realizar la navegación en los mismos se ha demostrado que se mantiene la estética y la navegabilidad, sin corromper los datos existentes.
- Se ha comprobado que la rapidez en la obtención de la información al realizar una consulta, no ha bajado su desempeño con un promedio de respuesta inferior a 1 minuto.
- La Galería de imágenes permite navegar con facilidad y despliega la información adecuadamente.

Se ha instalado la Aplicación Web en el Servidor del CIAM y funciona adecuadamente, se han realizado validaciones de conexión con la Base de Datos.

3.9.6 PRUEBAS DE SEGURIDAD

La Gestión de la Aplicación, se maneja con seguridades de login y password, los mismos que permiten mantener la integridad de la información que posee la Aplicación.

3.9.7 PRUEBAS DE DESEMPEÑO

Las Pruebas de desempeño fueron calculadas en base a la siguiente fórmula⁹⁹:

$P=N \times T \times D$ Donde,

N, es el número de usuarios concurrentes.

T, es el número de transacciones en línea por usuario por unidad de tiempo.

D, es la carga de datos procesada por el servidor por transacción.

Para nuestro estudio se realizó una prueba con 20 usuarios, cada usuario realiza una transacción cada 2 minutos, y la carga de datos procesada por cada transacción realizada en el servidor, es de 87Kbytes, por lo tanto, queda de la siguiente manera:

$$P= (20 \times 0.5 \times 512) / 60$$

$$P=85.33 \text{ Kbytes / seg.}$$

Las pruebas de carga de datos procesadas están basadas en una velocidad de 512 k con conexión wireless.

⁹⁹ Como Probar Aplicaciones Web, Ingeniería Web, Capítulo 20 Pressman.

CAPÍTULO 4

CONCLUSIONES, RECOMENDACIONES Y BIBLIOGRAFÍA

4.1 CONCLUSIONES



- Para el desarrollo de la Guía Interactiva, se realizaron investigaciones sobre Software Libre, con el objetivo de cumplir los requisitos para software en entidades del Gobierno.
- La Utilización de Software Libre, permite la reducción de costos en el Desarrollo de la Guía Interactiva de Áreas Protegidas.
- El Uso de la Metodología XP, proporciona flexibilidad en la programación ya que se puede hacer o deshacer con facilidad cualquier función creada.
- El Generador de Código Crea Code, permite al programador optimizar tiempo de programación, al generar código fuente.
- Al Generar un CD con la Guía Interactiva, permite a un mayor número de personas conocer sobre Áreas Protegidas del Ecuador, ya que no se requiere conexión a Internet.


4.2 RECOMENDACIONES


- Se debería difundir el uso de Software libre en las entidades del estado y en los establecimientos educativos en los diferentes niveles, para la creación de proyectos.
- Utilizar Programación Xtrema, para este tipo de proyectos, ya que facilita al programador la creación, modificación y eliminación de funciones sin necesidad de primero tener el modelo definido del Sistema.
- Emplear el Sistema generador de código Crea Cod, ya que facilita al programador al Generar el código Fuente en varios lenguajes de programación y permite ahorro de tiempo en el desarrollo de aplicaciones.


4.3 BIBLIOGRAFÍA:


LIBROS:

 SCHWABE, D. ROSI, G.; BARBOSA, S. Systematic Hypermedia Application Design with OOHDM. *Proceedings of the ACM International Conference on Hypertext (Hypertext'96)*. Washington DC, March, 16-20, 1996.  <http://www-di.inf.puc-rio.br/schwabe/papers/TAPOSRevised.pdf> [Volver]


 SCHWABE, D. BARBOSA, S. "An Object Oriented Approach to Web-Based Applications Design. En: *Theory and Practice of Object Systems (TAPOS)*, October 1998.


 CARIDAD, Mercedes y MOSCOSO, Purificación. *Los sistemas de hipertexto e hipermedios*. Fundación Germán Sánchez Ruipérez, 1991.


 CASANOVA, M., TUCHERMAN, L. et al. "The nested context model for hyperdocuments". *Proceedings ACM Conference on Hypertext and Hypermedia - Hypertext'91*. ACM Press, New York, 1991.


 CAMPBELL, B., GOODMAN, and J "HAM: A general purpose hypertext abstract machine". *CACM*, Vol. 31, Nº 7, July 1988.


 NIELSEN, J. *Hypertext and Hypermedia*. Oxford: Oxford Academic Press, 1990.

 PARUNAK, H. "Don't link me in: set based hypermedia for taxonomic reasoning". *Proceedings 3th ACM Conference on Hypertext and Hypermedia - Hypertext'91*. ACM Press, New York, 1991.


 SHNEIDERMAN, B. *Designing the user interface: Strategies for effective Human-Computer Interaction*. 3th ed. Massachusetts: Addison-Wesley, 1998.

 TOMPA, F. "A Data Model for Flexible Hypertext Database Systems". *ACMTOIS* Vol 7. Nº 1, January 1989.


 Van DAM, Andries. Hypertext '87 keynote address. *Communications of the ACM*, 31, July 1988, 887-895.


 **Pressman Roger S.** Ingeniería del Software Un enfoque práctico, Sexta Edición McGrawHill, Capítulo I, páginas 503-516


WEB:


 GÜEL, Natacha; SCHWABE, Daniel; VILAIN, Patricia. *Modeling Interactions and Navigation in Web Applications*. <http://www-di.inf.puc-rio.br/schwabe//papers/WWWCM00Expanded.pdf>


 OMG. Object Management Group. <http://www.omg.org/>

 SILVA, Darío Andrés. MERCERAT, Silvia. *Construyendo aplicaciones web con una metodología de diseño orientada a objetos*. http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf


 AMBLER, Scott. *Home Page*. <http://www.ambysoft.com/scottAmbler.html>


 BALASUBRAMANIAN, V. "State of the Art Review of Hypermedia: Issues and Applications", 1995.

 BIANCHINI, Adelaide. *Modelo referencial de hipermedio, basado en teoría de grafos para minimizar el problema de la desorientación del usuario*. <http://www ldc.usb.ve/~abianc/publicaciones.html> [Volver]


 BLAT, Josep. *Hypermedia/Multimedia Systems (2004). Introduction: systems, applications and models*. <http://www.iaa.upf.es/~jblat/material/doctorat/introduction.html>

 CONKLIN, Jeff. "Hypertext: An Introduction and Survey". *IEEE Computer*, September 1987. <http://cs.aue.aau.dk/~kirstin/f7s2005/pdf/conklin.pdf>


 DE BRA. *The Architecture of Hypertext Systems*. Eindhoven University of Technology. <http://www.wis.win.tue.nl/>

 DONDO, Agustín. *Programación orientada a objetos*. Programación en castellano. http://programacion.com/articulo/dondo_poo/

 Especificaciones UML. <http://www.popkin.com/>

 MARTÍNEZ SÁNCHEZ, José Manuel. HILERA GONZÁLEZ, José Ramón.
"Modelado de documentación multimedia e hipermedia" Cuadernos de Documentación
Multimedia, núm. 6-7, 1997-1998.

<http://www.ucm.es/info/multidoc/multidoc/revista/cuad6-7/artmulti.htm>


 MUKHERJEA, S., FOLEY, J. "Visualizing the World Wide Web with a Navigational
View Builder". *Proceedings Third International Conference on the World Wide Web -
WWW*

1995. <http://www.igd.fhg.de/www/www95/proceedings/papers/44/mukh/mukh.html>


 NAVARRETE TERRASA, Toni. *Modelos Hipermedia*. Verano del 2000.

<http://www.iaa.upf.es/~berenguer/cursos/interact/treballs/navarrete/modelos.pdf>

 OMG (Object Management Group). <http://www.omg.org/>

 ROVIRA, Cristòfol. *La orientación a objetos en el diseño de hipertextos para la
enseñanza-*

aprendizaje. <http://www.ucm.es/info/multidoc/multidoc/revista/num8/rovira.html>


 Universidad de Murcia. Departamento de Información y Documentación.
*Hipertexto y World Wide Web: Representación de información en entornos
navegacionales.*


<http://www.um.es/gtiweb/fjmm/sarisite/2002-Trans-T1/SARI-trans-T1-p1.PPT>


 De VOCHT, J. "Experiments for the Characterization of Hypertext Structures".


Master's Thesis, Eindhoven University of Technology. April 1994.


<http://wwwis.win.tue.nl/~debra/joep>

 Gamma, Y., Helm, R., Johnson R. y Vlissides, J. (1995). *Design Patterns:
Elements of reusable object-oriented software*. USA: Addison Wesley.

 German, D. (2003). *The Object Oriented Hypermedia Design Method*.
[Documento en línea]. Disponible: <http://www.telemidia.puc-rio.br/oohdm/oohdm.html>. [Consulta: 2005, Febrero 05].

 Koch, N. (2002). *Ingeniería de Requisitos en Aplicaciones para la Web—Un estudio comparativo*. [Documento en línea]. Disponible: <http://www.lsi.us.es/docs/informes/LSI-2002-4.pdf>. [Consulta: 2005, Febrero 05].

 Sánchez, M. (s. f.). *Interfaz de Usuario en el Desarrollo de un Simulador de Conducción* [Documento en línea]. http://www.cc3.net/sanchez/Sanchez_M.htm. [Consulta 2004, Diciembre 24].

 Silva, D. y Mercerat, B. (2001). *Construyendo aplicaciones web con una metodología de diseño orientada a objetos*. [Documento en línea]. Disponible: www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art_c.pdf. [Consulta 2004, Diciembre 24].

 Juan Manuel Barruffaldi www.viait.com.ar Pág. 2 de 5

 <http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>

 <http://www.di.uniovi.es/~cueva/asignaturas/doctorado/2004/IngWeb.html>

 http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_la_Web


 <http://www.marinaramos.net/docu/tema3.pdf>

 file:///D:/iweb/Sistemas_III.htm

Extreme Programming (XP) – An Overview

ExtremeProgramming.aspx.htm

http://www.csharpcorner.com/UploadFile/sakthivel_a/ExtremeProgramming11282005054653AM/ExtremeProgramming.aspx

 Desarrollo de software: probando Extreme Programming

Desarrollo de software probando XP.htm

<http://aikon.com.ve/metodologias-desarrollo-software-extreme-programming/>

 Ejemplo de desarrollo software utilizando la metodología XP

Ejemplo de desarrollo software con XP.html

<http://www.dsic.upv.es/assignaturas/facultad/lsi/ejemploxp/index.html>

 Prácticas Básicas de XP.

programacion-extrema2.shtml

<http://www.monografias.com/trabajos51/programacion-extrema/programacion-extrema2.shtml>