



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

TEMA

“ESTUDIO E IMPLEMENTACIÓN DE UN SISTEMA GIS WEB PARA EL SISTEMA DE AVALÚOS Y CATASTROS DEL ILUSTRE MUNICIPIO DE IBARRA”

APLICATIVO

“SISTEMA GIS PARA LA ZONA REGULADA DEL CANTÓN DE IBARRA”

Autor: Cueva Cabrera Byron Mesias.

Director: Ing. Rea Mauricio.

Ibarra – Ecuador

2012

CERTIFICACIÓN

Certifico que la Tesis **“ESTUDIO E IMPLEMENTACIÓN DE UN SISTEMA GIS WEB PARA EL SISTEMA DE AVALÚOS Y CATASTROS DEL ILUSTRE MUNICIPIO DE IBARRA”** con el aplicativo **“SISTEMA GIS PARA LA ZONA REGULADA DEL CANTÓN DE IBARRA”** ha sido realizada en su totalidad por el señor: Cueva Cabrera Byron Mesias portador de la cédula de identidad número: 100295768-4.

.....
Ing. Rea Mauricio.

Director de la Tesis

CERTIFICACIÓN

Ciudad, 4 de abril de 2012

Señores

UNIVERSIDAD TÉCNICA DEL NORTE

Presente

De mis consideraciones.-

Siendo auspiciantes del proyecto de tesis del Egresado CUEVA CABRERA BYRON MESIAS con CI: 100295768-4 quien desarrolló su trabajo con el tema “ESTUDIO E IMPLEMENTACIÓN DE UN SISTEMA GIS WEB PARA EL SISTEMA DE AVALÚOS Y CATASTROS DEL ILUSTRE MUNICIPIO DE IBARRA” con el aplicativo “SISTEMA GIS PARA LA ZONA REGULADA DEL CANTÓN DE IBARRA”, me es grato informar que se han superado con satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del egresado CUEVA CABRERA BYRON MESIAS. Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra empresa/institución.

El egresado CUEVA CABRERA BYRON MESIAS puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,

Ing. Irving Reascos.

Director TIC.

ILUSTRE MUNICIPIO DE IBARRA



UNIVERSIDAD TÉCNICA DEL NORTE

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, CUEVA CABRERA BYRON MESIAS, con cedula de identidad Nro. 100295768-4, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, articulo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **“ESTUDIO E IMPLEMENTACIÓN DE UN SISTEMA GIS WEB PARA EL SISTEMA DE AVALÚOS Y CATASTROS DEL ILUSTRE MUNICIPIO DE IBARRA”** con el aplicativo **“SISTEMA GIS PARA LA ZONA REGULADA DEL CANTÓN DE IBARRA”**, que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte

.....

Firma

Nombre: CUEVA CABRERA BYRON MESIAS

Cédula: 100295768-4

Ibarra a los 18 días del mes de Julio del 2012



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente investigación:

DATOS DE CONTACTO	
CEDULA DE IDENTIDAD	100295768-4
APELLIDOS Y NOMBRES	CUEVA CABRERA BYRON MESIAS
DIRECCIÓN	Otavalo – Miguel Egas Cabezas – Barrio Central
EMAIL	byron7_cueva@yahoo.es
TELÉFONO FIJO	
TELÉFONO MOVIL	086238573

DATOS DE LA OBRA	
TITULO	“ESTUDIO E IMPLEMENTACIÓN DE UN SISTEMA GIS WEB PARA EL SISTEMA DE AVALÚOS Y CATASTROS DEL ILUSTRE MUNICIPIO DE IBARRA”
AUTOR	CUEVA CABRERA BYRON MESIAS
FECHA	04 DE ABRIL DEL 2012
PROGRAMA	PREGRADO
TITULO POR EL QUE	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTOR	ING. REA MAURICIO.

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, CUEVA CABRERA BYRON MESIAS, con cedula de identidad Nro. 100295768-4, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 143.

.....

Firma

Nombre: CUEVA CABRERA BYRON MESIAS.

Cédula: 100295768-4

Ibarra a los 18 días del mes de Julio del 2012

DEDICATORIA

A mis padres Manuel Mesias Cueva y Cecilia Cabrera a quien los quiero mucho y les agradezco por su comprensión y ayuda incondicional.

A mi padre por los tantos momentos cuando necesitaba un consejo y una mano para realizar las cosas que a veces se afrontan en la vida. Por ser perseverante y saber manejar nuestro hogar aun así cuando la vida se ponía dura y difícil, gracias por luchar y buscar cómo salir adelante.

A mi madre quien con su amor de madre ha brindado cariño y dedicación a nuestro hogar; los cuidados y por la ternura con la que me ha criado; y por el apoyo que siempre le aportado a mi padre en el trabajo.

A mi única hermana Claribel Cueva, a quien quiero mucho y con quien compartí muchos momentos especiales en mi infancia.

A una persona especial quien me apoyo y motivo a culminar mis estudios, quien me ha demostrado cariño y ha formado parte de mi desarrollo como ser humano y como profesional.

Byron Cueva...

AGRADECIMIENTO

A Dios por cuidarme y guiar mi camino.

A la Universidad Técnica del Norte por haberme acogido en sus aulas.

A la Facultad de Ingeniería en Ciencias Aplicadas y a todos sus docentes quienes me han sabido enseñar sus conocimientos y contribuir en mi desarrollo profesional.

Agradezco a mi director de Tesis, Ing. Mauricio Rea, quien me colaboró en todo momento en el desarrollo de este trabajo de grado.

Al Ilustre Municipio de Ibarra y a todas las personas que conforman la dirección de TIC's por haberme permitido realizar el presente trabajo.

A mis padres por el gran sacrificio depositado en mí para permitirme ser un profesional.

A familiares, amigos y a todas aquellas personas que de alguna manera u otra apoyaron mi carrera y colaboraron para la culminación de la misma.

Byron Cueva...

Tabla de Contenidos

CERTIFICACIÓN	I
CERTIFICACIÓN	II
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	III
AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	IV
DEDICATORIA.....	VI
AGRADECIMIENTO.....	VII
TABLA DE CONTENIDOS	VIII
ÍNDICE DE TABLAS.....	XIII
ÍNDICE DE FIGURAS.....	XVI
RESUMEN	XIX
SUMMARY	XX
1. CAPÍTULO I: INTRODUCCIÓN.	2
1.1. ANTECEDENTES.....	2
1.2. BASE LEGAL DE LA INSTITUCIÓN.....	3
1.2.1. <i>Misión</i>	3
1.2.2. <i>Visión</i>	3
1.2.3. <i>Objetivos Generales</i>	3
1.2.4. <i>Objetivos Específicos</i>	4
1.2.5. <i>Organigrama Estructural por Procesos</i>	5
1.2.6. <i>Dirección de Avalúos y Catastros</i>	6
1.2.6.1. <i>Misión</i>	6
1.2.6.2. <i>Funciones</i>	6
1.3. PROBLEMA.....	6
1.4. OBJETIVOS.....	7
1.4.1. <i>Objetivo General</i>	7
1.4.2. <i>Objetivos Específicos</i>	7
1.5. JUSTIFICACIÓN.....	8
1.6. ALCANCE.....	8
2. CAPÍTULO II: MARCO TEÓRICO.	11
2.1. SISTEMAS DE INFORMACIÓN GEOGRÁFICA.....	11
2.1.1. <i>Definición</i>	11
2.1.2. <i>Características</i>	11
2.1.3. <i>Componentes de un GIS</i>	12
2.1.4. <i>Funcionalidad</i>	12
2.1.5. <i>Aplicaciones de un GIS</i>	13
2.1.6. <i>Información Geográfica</i>	13
2.1.6.1. <i>Definición</i>	13
2.1.6.2. <i>Componentes de la Información Geográfica</i>	13

2.1.7.	<i>Modelo de Datos</i>	14
2.1.7.1.	Modelo Vectorial.....	14
	Elementos Geográficos.....	15
	Formatos de los Geodatos.....	16
2.1.7.2.	Modelo Raster.....	18
	Características de los formatos Raster	18
	Georeferencia de Imágenes.....	20
	Formatos de Imágenes	20
	Formatos de Compresión	21
2.1.7.3.	Orientada a Objetos	22
2.1.8.	<i>Sistema de Coordenadas y Proyección</i>	22
2.1.8.1.	Sistema de Coordenadas Geográficas.....	22
	Líneas de Referencia del globo terrestre. Unidades de Medida.....	23
	Esferas y Geoides	23
	Elipsoides.....	24
2.1.8.2.	El Datum.....	25
2.1.8.3.	Los Sistemas de Proyección.....	25
2.1.9.	<i>Sistema de Referencia WGS-84</i>	26
2.1.10.	<i>Sistema de Referencia Provisional South American Datum 1956 – PSAD56</i>	26
2.1.11.	<i>Captura de Información Geográfica</i>	27
2.1.12.	<i>Cartografía Temática</i>	28
2.1.12.1.	Mapa.....	28
2.1.12.2.	Semiología Gráfica.....	28
2.1.12.3.	Elementos Gráficos Básicos.....	28
2.2.	INTERCAMBIO DE DATOS ESPACIALES EN LA WEB.....	34
2.2.1.	<i>La Open Geospatial Consortium OGC</i>	34
2.2.2.	<i>Lenguajes de Marcado</i>	35
2.2.2.1.	XML (Extensible Markup Language).....	35
2.2.2.2.	HTML.....	36
2.2.2.3.	GML (Geography Markup Language).....	37
2.2.2.4.	KML (Keyhole Markup Language).....	38
2.2.2.5.	SVG (Scalable Vector Graphics).....	39
2.2.2.6.	GeoRSS	39
2.2.3.	<i>Formatos Vectoriales</i>	40
2.2.3.1.	GeoJSON.....	40
2.2.3.2.	WKT (Well Known Text).....	41
2.2.4.	<i>OGC Web Services (OWS)</i>	41
2.2.4.1.	<i>WMS (Web Map Services)</i>	42
2.2.4.2.	<i>WFS (Web Features Services)</i>	45
2.2.4.3.	<i>WCS (Web Coverage Service)</i>	48
2.3.	BASE DE DATOS.....	49
2.3.1.	<i>Base de Datos Relacional</i>	49
2.3.1.1.	PostgreSQL.....	49
2.3.2.	<i>Base de Datos Espacial</i>	50
2.3.2.1.	PostGIS.....	50
2.4.	SERVIDORES DE APLICACIONES.....	51
2.4.1.	<i>HTTP Apache</i>	51
2.4.2.	<i>Tomcat</i>	51
2.5.	LENGUAJES DE PROGRAMACIÓN.....	52
2.5.1.	<i>CSS</i>	52
2.5.2.	<i>PHP</i>	52
2.5.3.	<i>JAVASCRIPT</i>	53
2.5.4.	<i>PHP MAPSCRIPT</i>	54

2.5.5.	MAPFILE.....	54
2.6.	FRAMEWORKS.....	55
2.6.1.	Symfony 1.16.....	55
2.6.2.	Doctrine.....	56
2.6.3.	Ext JS.....	56
2.7.	METODOLOGÍA DE PROGRAMACIÓN EXTREMA XP.....	57
2.7.1.	Definición.....	57
2.7.2.	Actividades.....	57
2.7.2.1.	Codificar.....	57
2.7.2.2.	Hacer Prueba.....	58
2.7.2.3.	Escuchar.....	58
2.7.2.4.	Diseñar.....	58
2.7.3.	Ciclo de Vida.....	58
2.7.4.	Fases.....	58
2.7.4.1.	Fase de Exploración.....	58
2.7.4.2.	Fase de Planificación.....	59
2.7.4.3.	Fase de Iteración.....	59
2.7.4.4.	Fase de Producción.....	59
2.7.5.	Artefactos.....	59
2.7.5.1.	Historias de Usuario.....	59
2.7.5.2.	Tareas de Ingeniería.....	59
2.7.5.3.	Casos de Prueba de Aceptación.....	60
2.7.5.4.	Tarjeta CRC.....	60
2.8.	ALGUNAS DEFINICIONES.....	60
2.8.1.	Servidor de Mapas.....	60
2.8.2.	Clientes GIS Livianos.....	61
2.8.3.	Clientes GIS Pesados.....	61
2.8.4.	Norma ISO 19110: Metodología para la creación de catálogos de elementos.....	61
3.	CAPÍTULO III: ESTUDIO DE HERRAMIENTAS GIS OPEN SOURCE.....	63
3.1.	OPEN SOURCE (CÓDIGO ABIERTO).....	63
3.2.	SOFTWARE LIBRE.....	63
3.2.1.	Libertades del software libre.....	63
3.2.2.	Ventajas del Software Libre.....	64
3.3.	SERVIDORES DE MAPAS.....	65
3.3.1.	Cuadro de Compatibilidad.....	65
3.3.2.	Cuadro de Características Técnicas.....	67
3.3.3.	Cuadro de Funcionalidades y Características Principales.....	68
3.3.4.	Selección de Herramienta.....	69
3.4.	CLIENTES GIS LIVIANOS.....	69
3.4.1.	Cuadro de Compatibilidad.....	69
3.4.2.	Cuadro de Características Técnicas.....	71
3.4.3.	Cuadro de Funcionalidades y Características Principales.....	72
3.4.4.	Selección de Herramientas.....	73
3.5.	CLIENTES GIS PESADOS.....	74
3.5.1.	Cuadro de Compatibilidad.....	74
3.5.2.	Cuadro de Características Técnicas.....	76
3.5.3.	Cuadro de Funcionalidades y Características Principales.....	77
3.5.4.	Selección de Herramientas.....	77
3.6.	SERVIDORES DE CACHE GEOESPACIAL.....	78
3.6.1.	Cuadro de Compatibilidad.....	78

3.6.2.	<i>Cuadro de Características Técnicas</i>	78
3.6.3.	<i>Selección de Herramientas</i>	78
3.7.	PROPUESTAS DE SOFTWARE PARA EL DESARROLLO DE UN SISTEMA GIS.....	79
3.7.1.	<i>Software Privativo</i>	79
3.7.2.	<i>Software Libre</i>	79
4.	CAPÍTULO IV: DISEÑO Y DESARROLLO DEL APLICATIVO	81
4.1.	FASE DE EXPLORACIÓN.....	83
4.1.1.	<i>Roles</i>	83
4.1.2.	<i>Integrantes del Equipo XP</i>	84
4.1.3.	<i>Historias de Usuario</i>	84
4.1.4.	<i>Metáfora</i>	93
4.1.5.	<i>Diseño del Sistema</i>	93
4.1.5.1.	Arquitectura del Sistema.....	93
4.1.5.2.	Arquitectura Funcional.....	95
4.1.5.3.	Módulos del Sistema.....	96
4.1.6.	<i>Tareas</i>	96
4.1.6.1.	Módulo de Seguridad y Acceso al Sistema.....	97
4.1.6.2.	Módulo Visor de Mapas.....	97
4.1.6.3.	Módulo de Consultas y Búsquedas.....	104
4.1.7.	<i>Detalle de Historias de Usuario</i>	109
4.1.8.	<i>Tiempo Ideal</i>	111
4.1.9.	<i>Estimación de Esfuerzo</i>	112
4.1.9.1.	Estimaciones de historias de usuario en tiempo ideal.....	113
4.2.	FASE DE PLANIFICACIÓN.....	114
4.2.1.	<i>Velocidad del equipo de desarrollo</i>	114
4.2.2.	<i>Iteración por cada entrega</i>	114
4.2.3.	<i>Planificación por Historias</i>	115
4.2.4.	<i>Plan de Entregas</i>	117
4.2.5.	<i>Plan de Iteración</i>	117
4.2.5.1.	Iteración 1.....	118
4.2.5.2.	Iteración 2.....	118
4.2.5.3.	Iteración 3.....	119
4.2.5.4.	Iteración 4.....	120
4.2.5.5.	Iteración 5.....	120
4.3.	FASE DE ITERACIÓN.....	121
4.3.1.	<i>Diseño de pruebas de Aceptación</i>	121
4.3.1.1.	Módulo de Seguridad y Acceso al Sistema.....	121
4.3.1.2.	Módulo Visor de Mapas.....	122
4.3.1.3.	Módulo de Consultas y Búsquedas.....	126
4.3.2.	<i>Iteraciones</i>	134
4.3.2.1.	Primera Iteración.....	134
4.3.2.2.	Segunda Iteración.....	145
4.3.2.3.	Tercera Iteración.....	165
4.3.2.4.	Cuarta Iteración.....	172
4.3.2.5.	Quinta Iteración.....	176
4.3.3.	<i>Ejecución de Pruebas de Aceptación</i>	184
4.4.	FASE DE PRODUCCIÓN.....	188
4.4.1.	<i>Plan de Implantación</i>	188
4.4.1.1.	Requisitos para la implantación.....	188
4.4.1.2.	Preparación para la instalación.....	190
4.4.1.3.	Configuración del Aplicativo en un Ambiente de Producción.....	190

5. CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.	193
5.1. IMPACTOS.	193
5.2. CONCLUSIONES.	195
5.3. RECOMENDACIONES.	195
GLOSARIO	197
BIBLIOGRAFÍA	199
ANEXOS	203

Índice de Tablas

TABLA 1: ESTRUCTURA DE ALMACENAMIENTO RATER CON UNA ÚNICA COLUMNA DE NÚMEROS.....	18
TABLA 2: ORDEN EN EL QUE SE ALMACENAN LOS DATOS EN UN RASTER.	19
TABLA 3: PRINCIPALES ELIPSOIDES MEDIDOS EN LOS SIGLOS XIX Y XX.	24
TABLA 4: PARÁMETROS DE UNA PETICIÓN URL GETCAPABILITIES.....	43
TABLA 5: PARÁMETROS DE UNA PETICIÓN URL GETMAP.....	44
TABLA 6: PARÁMETROS DE UNA PETICIÓN URL GETFEATUREINFO.	45
TABLA 7: LIBERTADES DEL SOFTWARE LIBRE.	63
TABLA 8: COMPARATIVA DE SERVIDORES DE MAPAS, CUADRO DE COMPATIBILIDAD.....	66
TABLA 9: COMPARATIVA DE SERVIDORES DE MAPAS, CUADRO DE CARACTERÍSTICAS TÉCNICAS.	67
TABLA 10: COMPARATIVA DE SERVIDORES DE MAPAS, CUADRO DE FUNCIONALIDADES Y CARACTERÍSTICAS PRINCIPALES.	68
TABLA 11: COMPARATIVA DE GIS LIVIANOS, CUADRO DE COMPATIBILIDAD.	70
TABLA 12: COMPARATIVA DE GIS LIVIANOS, CUADRO DE CARACTERÍSTICAS TÉCNICAS.....	72
TABLA 13: COMPARATIVA DE GIS LIVIANOS, CUADRO DE FUNCIONALIDADES Y CARACTERÍSTICAS PRINCIPALES.	73
TABLA 14: COMPARATIVA DE GIS PESADOS, CUADRO DE COMPATIBILIDAD.	75
TABLA 15: COMPARATIVA DE GIS PESADOS, CUADRO DE CARACTERÍSTICAS TÉCNICAS.....	76
TABLA 16: COMPARATIVA DE GIS PESADOS, CUADRO DE FUNCIONALIDADES Y CARACTERÍSTICAS PRINCIPALES.	77
TABLA 17: COMPARATIVA DE CACHE GEOESPACIALES, CUADRO DE COMPATIBILIDAD.	78
TABLA 18: COMPARATIVA DE CACHE GEOESPACIALES, CUADRO DE CARACTERÍSTICAS TÉCNICAS.	78
TABLA 19: PROPUESTA PRECIOS CON SOFTWARE PRIVATIVO.	79
TABLA 20: PROPUESTA PRECIOS CON SOFTWARE LIBRE.	79
TABLA 21: HISTORIA DE CAMBIOS.	82
TABLA 22: ROLES DE XP.	84
TABLA 23: INTEGRANTES DEL EQUIPO XP.	84
TABLA 24: HISTORIA NRO. 1.....	85
TABLA 25: HISTORIA NRO. 2.....	86
TABLA 26: HISTORIA NRO. 3.....	86
TABLA 27: HISTORIA NRO. 4.....	86
TABLA 28: HISTORIA NRO. 5.....	87
TABLA 29: HISTORIA NRO. 6.....	87
TABLA 30: HISTORIA NRO. 7.....	88
TABLA 31: HISTORIA NRO. 8.....	88
TABLA 32: HISTORIA NRO. 9.....	89
TABLA 33: HISTORIA NRO. 10.....	89
TABLA 34: HISTORIA NRO. 11.....	89
TABLA 35: HISTORIA NRO. 12.....	90
TABLA 36: HISTORIA NRO. 13.....	90
TABLA 37: HISTORIA NRO. 14.....	90
TABLA 38: HISTORIA NRO. 15.....	91
TABLA 39: HISTORIA NRO. 16.....	91
TABLA 40: HISTORIA NRO. 17.....	92
TABLA 41: HISTORIA NRO. 18.....	92
TABLA 42: HISTORIA NRO. 19.....	92
TABLA 43: TAREA 1.1.	97
TABLA 44: TAREA 1.2.	97
TABLA 45: TAREA 2.1.	98
TABLA 46: TAREA 2.2.	98

TABLA 47: TAREA 2.3	98
TABLA 48: TAREA 2.4	98
TABLA 49: TAREA 3.1	99
TABLA 50: TAREA 3.2	99
TABLA 51: TAREA 3.3	99
TABLA 52: TAREA 3.4	99
TABLA 53: TAREA 3.5	100
TABLA 54: TAREA 3.6	100
TABLA 55: TAREA 3.7	100
TABLA 56: TAREA 3.8	100
TABLA 57: TAREA 4.1	101
TABLA 58: TAREA 4.2	101
TABLA 59: TAREA 4.3	101
TABLA 60: TAREA 4.4	102
TABLA 61: TAREA 4.5	102
TABLA 62: TAREA 5.1	102
TABLA 63: TAREA 5.2	102
TABLA 64: TAREA 6.1	103
TABLA 65: TAREA 6.2	103
TABLA 66: TAREA 7.1	103
TABLA 67: TAREA 7.2	104
TABLA 68: TAREA 8.1	104
TABLA 69: TAREA 8.2	104
TABLA 70: TAREA 8.3	105
TABLA 71: TAREA 9.1	105
TABLA 72: TAREA 9.2	105
TABLA 73: TAREA 9.3	105
TABLA 74: TAREA 10.1	106
TABLA 75: TAREA 10.2	106
TABLA 76: TAREA 11.1	106
TABLA 77: TAREA 12.1	107
TABLA 78: TAREA 13.1	107
TABLA 79: TAREA 14.1	107
TABLA 80: TAREA 15.1	108
TABLA 81: TAREA 16.1	108
TABLA 82: TAREA 17.1	108
TABLA 83: TAREA 18.1	108
TABLA 84: TAREA 18.2	109
TABLA 85: TAREA 19.1	109
TABLA 86: TAREA 19.2	109
TABLA 87: ESTIMACIÓN DE ESFUERZO, MÓDULO DE SEGURIDAD Y CONTROL DE ACCESO	112
TABLA 88: ESTIMACIÓN DE ESFUERZO, MÓDULO VISOR DE MAPAS	112
TABLA 89: ESTIMACIÓN DE ESFUERZO, MÓDULO DE CONSULTAS Y BÚSQUEDAS	113
TABLA 90: ESTIMACIÓN DE HISTORIAS DE USUARIO EN TIEMPO IDEAL	114
TABLA 91: PLANIFICACIÓN POR HISTORIAS	116
TABLA 92: PLAN DE ENTREGAS	117
TABLA 93: CASOS DE PRUEBA DE ACEPTACIÓN 1.1 DE HISTORIA NRO. 1	122
TABLA 94: CASOS DE PRUEBA DE ACEPTACIÓN 2.1 DE HISTORIA NRO. 2	123
TABLA 95: CASOS DE PRUEBA DE ACEPTACIÓN 3.1 DE HISTORIA NRO. 3	123
TABLA 96: CASOS DE PRUEBA DE ACEPTACIÓN 4.1 DE HISTORIA NRO. 4	124

TABLA 97: CASOS DE PRUEBA DE ACEPTACIÓN 5.1 DE HISTORIA NRO. 5.....	124
TABLA 98: CASOS DE PRUEBA DE ACEPTACIÓN 6.1 DE HISTORIA NRO. 6.....	125
TABLA 99: CASOS DE PRUEBA DE ACEPTACIÓN 6.2 DE HISTORIA NRO. 6.....	125
TABLA 100: CASOS DE PRUEBA DE ACEPTACIÓN 7.1 DE HISTORIA NRO. 7.....	126
TABLA 101: CASOS DE PRUEBA DE ACEPTACIÓN 7.2 DE HISTORIA NRO. 7.....	126
TABLA 102: CASOS DE PRUEBA DE ACEPTACIÓN 8.1 DE HISTORIA NRO. 8.....	126
TABLA 103: CASOS DE PRUEBA DE ACEPTACIÓN 8.2 DE HISTORIA NRO. 8.....	127
TABLA 104: CASOS DE PRUEBA DE ACEPTACIÓN 8.3 DE HISTORIA NRO. 8.....	127
TABLA 105: CASOS DE PRUEBA DE ACEPTACIÓN 9.1 DE HISTORIA NRO. 9.....	128
TABLA 106: CASOS DE PRUEBA DE ACEPTACIÓN 10.1 DE HISTORIA NRO. 10.....	129
TABLA 107: CASOS DE PRUEBA DE ACEPTACIÓN 11.1 DE HISTORIA NRO. 11.....	129
TABLA 108: CASOS DE PRUEBA DE ACEPTACIÓN 12.1 DE HISTORIA NRO. 12.....	130
TABLA 109: CASOS DE PRUEBA DE ACEPTACIÓN 13.1 DE HISTORIA NRO. 13.....	130
TABLA 110: CASOS DE PRUEBA DE ACEPTACIÓN 14.1 DE HISTORIA NRO. 14.....	131
TABLA 111: CASOS DE PRUEBA DE ACEPTACIÓN 15.1 DE HISTORIA NRO. 15.....	131
TABLA 112: CASOS DE PRUEBA DE ACEPTACIÓN 16.1 DE HISTORIA NRO. 16.....	132
TABLA 113: CASOS DE PRUEBA DE ACEPTACIÓN 17.1 DE HISTORIA NRO. 17.....	132
TABLA 114: CASOS DE PRUEBA DE ACEPTACIÓN 18.1 DE HISTORIA NRO. 18.....	133
TABLA 115: CASOS DE PRUEBA DE ACEPTACIÓN 19.1 DE HISTORIA NRO. 19.....	133
TABLA 116: TARJETA CRC: BASESFGUARDAUTHACTIONS.....	136
TABLA 117: TARJETA CRC: GiMAPLOGIC.....	159
TABLA 118: TARJETA CRC: GiGRUPOLOGIC.....	160
TABLA 119: TARJETA CRC: GiLAYERLOGIC.....	160
TABLA 120: TARJETA CRC: GiACCESOMANAGER.....	161
TABLA 121: TARJETA CRC: GiDASHBOARDACTION.....	162
TABLA 122: TARJETA CRC: GiSBARRA.PANEL.MAPPANEL.....	164
TABLA 123: TARJETA CRC: GiSBARRA.COMP.TREECAPAS.....	165
TABLA 124: TARJETA CRC: GiSBARRA.TOOLBAR.TOOLBARNAVMAP.....	166
TABLA 125: TARJETA CRC: GiGISACTIONS.....	169
TABLA 126: TARJETA CRC: CIUDADANOLOGIC.....	179
TABLA 127: TARJETA CRC: DATOSINGRESOLOGIC.....	180
TABLA 128: TARJETA CRC: DESCRIPCIONTERRENOPREDIOLOGIC.....	180
TABLA 129: TARJETA CRC: INFRAESTRSERVICIOSPREDIOLOGIC.....	181
TABLA 130: TARJETA CRC: UsZONIFICACIONUsoSUEOLOGIC.....	181
TABLA 131: TARJETA CRC: ZONIFICACIONIBARRAPREDIOSLOGIC.....	182
TABLA 132: TARJETA CRC: GiCATASTROSMANAGER.....	182
TABLA 133: TARJETA CRC: GiCATASTROSACTIONS.....	183
TABLA 134: EJECUCIÓN DE PRUEBAS DE ACEPTACIÓN.....	188
TABLA 135: REQUISITOS: BASE DE DATOS GEOGRÁFICA.....	189
TABLA 136: REQUISITOS: SERVIDOR DE MAPAS.....	189
TABLA 137: REQUISITOS: SERVIDOR DE APLICACIONES.....	189
TABLA 138: REQUISITOS: DEL CLIENTE.....	190
TABLA 139: IMPACTO: PORCENTAJES DE ACCESO Y CONOCIMIENTO DE LA INFORMACIÓN.....	193
TABLA 140: RESULTADO COMPARATIVO AL CONSULTAR LOS COLINDANTES DE UN PREDIO POR SEGUNDO.....	194

Índice de Figuras

FIGURA 1: ORGANIGRAMA ESTRUCTURAL POR PROCESOS DEL ILUSTRE MUNICIPIO DE IBARRA.....	5
FIGURA 2: INFORMACIÓN DE UN SISTEMA GIS.	11
FIGURA 3: MODELO DE DATOS GEOGRÁFICO.....	14
FIGURA 4: MODELO VECTORIAL.....	15
FIGURA 5: ELEMENTOS GEOGRÁFICOS DE UN MODELO VECTORIAL.	15
FIGURA 6: MODELO RASTER.	18
FIGURA 7: ESTRUCTURA DE LAS IMÁGENES RASTER.	18
FIGURA 8: PIRÁMIDE DE UNA IMAGEN RASTER.	20
FIGURA 9: ESFERAS Y GEOIDES.	24
FIGURA 10: TIPOS DE PROYECCIONES.	26
FIGURA 11: SISTEMA DE REFERENCIA WGS-84.....	26
FIGURA 12: FUENTES DE INFORMACIÓN GEOGRÁFICA.	27
FIGURA 13: MAPA DE SÍMBOLOS ÚNICOS.	29
FIGURA 14: MAPA CUALITATIVO.	30
FIGURA 15: MAPA DE ATRIBUTOS BINARIOS.	30
FIGURA 16: MAPA DE ATRIBUTOS ORDINALES.....	31
FIGURA 17: MAPA DE COROPLETAS.....	32
FIGURA 18: MAPA DE SÍMBOLOS GRADUADOS.....	32
FIGURA 19: MAPA DE DENSIDAD DE PUNTOS.....	33
FIGURA 20: ESTRUCTURA DE UN XML.....	35
FIGURA 21: ESTRUCTURA DE UN HTML.	37
FIGURA 22: DIAGRAMA UML QUE MUESTRA LA JERARQUÍA DE LA CLASE GML.....	38
FIGURA 23: ESTÁNDARES OGC.	42
FIGURA 24: OGC WMS.	43
FIGURA 25: OGC WFS.	46
FIGURA 26: ESTRUCTURA MAPFILE.	54
FIGURA 27: ARQUITECTURA DEL SISTEMA.	94
FIGURA 28: ARQUITECTURA FUNCIONAL DEL SISTEMA.....	95
FIGURA 29: MÓDULOS DEL SISTEMA.....	96
FIGURA 30: PLAN DE ITERACIÓN 1.....	118
FIGURA 31: PLAN DE ITERACIÓN 2.....	118
FIGURA 32: PLAN DE ITERACIÓN 3.....	119
FIGURA 33: PLAN DE ITERACIÓN 4.....	120
FIGURA 34: PLAN DE ITERACIÓN 5.....	120
FIGURA 35: DIAGRAMA RELACIONAL PARA EL CONTROL DE ACCESO AL SISTEMA.	135
FIGURA 36: INTERFAZ PARA EL INGRESO AL SISTEMA.	136
FIGURA 37: DIAGRAMA DE CLASES PARA EL MÓDULO DE SEGURIDAD Y ACCESO AL SISTEMA.	137
FIGURA 38: INSTALACIÓN GvSIG – INSTALACIÓN JDK.	140
FIGURA 39: INSTALACIÓN GvSIG – SELECCIÓN DEL LENGUAJE.	140
FIGURA 40: INSTALACIÓN GvSIG – INFORMACIÓN GENERAL.....	140
FIGURA 41: INSTALACIÓN GvSIG – ACUERDO DE LICENCIA.	141
FIGURA 42: INSTALACIÓN GvSIG – FIN DE INSTALACIÓN.	141
FIGURA 43: MIGRACIÓN - INICIO GvSIG.	142
FIGURA 44: MIGRACIÓN – CONFIGURACIÓN DE UN NUEVO PROYECTO.	142
FIGURA 45: MIGRACIÓN – CONFIGURACIÓN DE LA PROYECCIÓN.	142
FIGURA 46: MIGRACIÓN –VISTA PRINCIPAL DE NAVEGACIÓN.	143

FIGURA 47: MIGRACIÓN – VENTANA PARA AGREGAR CAPAS A LA VISTA ACTUAL.	143
FIGURA 48: MIGRACIÓN – ABRIR UN SHAPE.	143
FIGURA 49: MIGRACIÓN – NAVEGACIÓN SHAPE.	144
FIGURA 50: MIGRACIÓN – INGRESO DEL NOMBRE DE LA CAPA.	144
FIGURA 51: MIGRACIÓN – CONFIGURACIÓN DE LA CONEXIÓN DE LA BASE DE DATOS A MIGRAR.	144
FIGURA 52: MIGRACIÓN – PROCESO DE MIGRACIÓN DE SHAPE A UNA BASE DE DATOS POSTGIS.	145
FIGURA 53: MIGRACIÓN – VERIFICACIÓN DE LOS DATOS MIGRADOS EN EL PGADMIN.	145
FIGURA 54: PRUEBA DEL FUNCIONAMIENTO DEL SERVIDOR APACHE.	145
FIGURA 55: PRUEBA DEL SERVICIO DE MAPSERVER.	146
FIGURA 56: CONFIGURACIÓN DE CARPETAS PARA CONFIGURACIÓN DE MAPSERVER.	146
FIGURA 57: PRUEBA DEL SERVICIO WMS REQUEST GETCAPABILITIES.	151
FIGURA 58: PRUEBA DEL SERVICIO WMS REQUEST GETMAP.	151
FIGURA 59: GEOSERVER – CREACIÓN DE UN ESPACIO DE TRABAJO.	153
FIGURA 60: GEOSERVER – CREACIÓN DE UN STORE.	154
FIGURA 61: GEOSERVER – PUBLICACIÓN DE UNA CAPA.	155
FIGURA 62: PROBANDO REQUEST GETCAPABILITIES.	157
FIGURA 63: PROBANDO REQUEST GETMAP, A TRAVÉS DEL CLIENTE PESADO UDIG.	157
FIGURA 64: MODELO RELACIONAL PARA EL REGISTRO Y CONTROL DE CAPAS.	158
FIGURA 65: DISEÑO DE LA INTERFAZ DE LA APLICACIÓN.	159
FIGURA 66: ÁREA DE VISIÓN DE MAPAS.	162
FIGURA 67: TREE DE COLECCIÓN DE CAPAS.	164
FIGURA 68: BARRA DE HERRAMIENTAS DE NAVEGACIÓN.	165
FIGURA 69: BARRA DE LEYENDAS.	166
FIGURA 70: VENTANA PARA AGREGAR CAPAS.	167
FIGURA 71: HERRAMIENTA DE MEDICIÓN DE LA LONGITUD.	167
FIGURA 72: HERRAMIENTA DE MEDICIÓN DE ÁREA.	167
FIGURA 73: HERRAMIENTA DE BÚSQUEDA DE ZONAS.	168
FIGURA 74: HERRAMIENTA DE CONSULTA POR CALLES.	168
FIGURA 75: HERRAMIENTA DE CONSULTA LUGARES.	168
FIGURA 76: HERRAMIENTA DE CONSULTA DE LA INFORMACIÓN DE UN PREDIO.	170
FIGURA 77: MODELO RELACIONAL – FOTOS PREDIO.	170
FIGURA 78: MODELO RELACIONAL, PARA OBTENER LA INFORMACIÓN DEL PREDIO.	171
FIGURA 79: INTERFAZ PARA LA CONSULTA DE UN PREDIO.	172
FIGURA 80: RENDERIZACION SVG DE UN PREDIO SELECCIONADO.	172
FIGURA 81: INTERFAZ DE CONSULTA DE LA ZONIFICACIÓN.	172
FIGURA 82: MODELO RELACIONAL PARA LA CONSULTA DE LA ZONIFICACIÓN DE UN PREDIO.	173
FIGURA 83: INTERFAZ PARA LA CONSULTA DEL USO DE SUELO DE UN PREDIO.	173
FIGURA 84: MODELO RELACIONAL PARA LA CONSULTA DEL USO DEL SUELO.	173
FIGURA 85: VENTANA DESCRIPCIÓN DEL TERRENO.	174
FIGURA 86: MODELO RELACIONAL PARA LA CONSULTA DE LA DESCRIPCIÓN DEL TERRENO.	174
FIGURA 87: VENTANA INFORMACIÓN DE LA INFRAESTRUCTURA Y SERVICIOS DE UN PREDIO.	175
FIGURA 88: MODELO RELACIONAL PARA LA CONSULTA DE LA INFRAESTRUCTURA Y SERVICIOS.	175
FIGURA 89: VENTANA PARA LA CONSULTA DE LAS PROPIEDADES DE UN CIUDADANO.	176
FIGURA 90: VENTANA INFORMACIÓN DEL IMPUESTO PREDIAL.	176
FIGURA 91: REPORTE DEL IMPUESTO PREDIAL.	177
FIGURA 92: MODELO RELACIONAL PARA LA CONSULTA DE LOS IMPUESTOS PEDIALES.	177
FIGURA 93: VENTANA DE CONSULTA DE LOS COLINDANTES.	178
FIGURA 94: VENTANA PARA LA IMPRESIÓN DE UN CROQUIS.	178
FIGURA 95: VENTANA PARA LA CONSULTA DE LOS IMPUESTOS PEDIALES POR PARROQUIAS.	179
FIGURA 96: ENTORNO DE PRODUCCIÓN.	190

FIGURA 97: CUADRO ESTADÍSTICO DEL RESULTADO DEL ACCESO Y CONOCIMIENTO A LA INFORMACIÓN.	193
FIGURA 98: CUADRO ESTADÍSTICO DEL RESULTADO COMPARATIVO AL CONSULTAR LOS COLINDANTES DE UN PREDIO.	194

Resumen

El presente documento detalla todos los procesos que se siguieron para el desarrollo del Sistema GIS para la zona regulada de Ibarra (GIS IMI); para el Sistema de Catastros y Avalúos del Ilustre Municipio de Ibarra.

En la elaboración y desarrollo del Sistema GIS IMI, el documento presenta cinco capítulos. En cada uno se detalla los procesos, metodologías y conceptos que servirán de ayuda en el transcurso de la elaboración del Sistema.

En el capítulo uno, se hace una descripción de la institución a la cual se va aplicar el sistema, seguidamente en el capítulo dos se detallan conceptos principales que se involucran y son necesarios conocer en el desarrollo de la aplicación.

En el capítulo tres, se realiza un análisis de las herramientas para el desarrollo del sistema y se selecciona las más recomendables según la arquitectura a plantear.

El capítulo cuatro se desarrolla a través de las fases de la metodología de software XP (Programación Extrema), siendo esta una de las más utilizadas por los jefes de proyectos. Por último en el capítulo cinco se detallan las conclusiones a las que se ha llegado al realizar el Sistema.

Summary

This document details all the processes followed for the development of the GIS for the area regulated Ibarra (GIS IMI) for the System of Cadastre and Valuation of the Illustrious Municipality of Ibarra.

In the design and development of GIS System IMI, the document presents five chapters. In each detailed processes, methodologies and concepts that will be helpful during the development of the system.

In chapter one, is a description of the institution which will implement the system, then in chapter two main concepts are described that are involved and are necessary to know in the development of the application.

In chapter three, an analysis of tools for system development and select the most suitable according to the present architecture.

Chapter four is developed through the phases of the software methodology XP (Extreme Programming), this being one of the most used by project managers. Finally in chapter five details the conclusions he has come to make the system.

Capítulo I: Introducción

IBARRA *Hacia el Futuro*

CONTENIDO:

1. Antecedentes.
2. Base Legal de la Institución.
3. Problema.
4. Objetivos.
5. Justificación.
6. Alcance.

1. Capítulo I: Introducción.

1.1. Antecedentes.

El Ilustre Municipio de Ibarra es una Institución pública que tiene como misión “Planificar, regular, ejecutar y promover el desarrollo integral sostenible del cantón de Ibarra, a través de servicios de calidad eficientes y transparentes con la participación activa de la ciudadanía socialmente a fin de lograr un buen vivir”.

El catastro inmobiliario es un registro administrativo dependiente del Estado en el que se describen los bienes inmuebles rústicos, urbanos y de características especiales. Un sistema catastral permite unificar los criterios de representación y empleo de la información relacionada con el uso y tenencia de la tierra así como define el tratamiento de la cartografía y los datos alfanuméricos para que puedan ser manejados de manera eficiente e implementados en un sistema informativo automatizado.

Por otra parte la gestión de la información cartográfica asociada al territorio necesita de herramientas potentes de análisis de datos, tanto gráficos como alfanuméricos, estos datos representan variables y tienen una determinada distribución espacial.

La información catastral, especialmente la referida al valor de la superficie de las zonas habitadas, es un dato más asociado a las entidades territoriales y por tanto, es en principio susceptible de ser gestionado por Sistemas de Información Geográfica.

Los Sistemas de Información Geográfica se desarrollan por tanto, como una tecnología que permite la manipulación y el análisis de los datos geográficos, siendo de interés no sólo para científicos e investigadores sino también para organismos administrativos.

La información espacial, es el dato, definido como cualquier cosa que ocupe una posición definida en el espacio, que puede ser físicamente real (puente, río, bosque, calle, etc.). Se puede decir que es aquella información que es factible a ser resumida como imagen o un plano, es decir que no requiera texto para ser inteligible o utilizable.

El empleo de un Sistemas de Información Geográfica permite conocer la ubicación de los recursos en el contexto geográfico, al manejar información digital referente a topografía. Esta información debe estar ligada a tablas específicas de una base de datos.

Los Sistemas GIS facilitan el enlace de información geográfica con información descriptiva. A diferencia de un mapa tradicional, un GIS puede presentar sobre el mapa de una determinada región, de manera interactiva, varias capas que se superponen y que contienen información temática; por ejemplo, sobre recursos naturales, asentamientos humanos, educación, transporte, salud, agricultura, geología, etc.

1.2. Base Legal de la Institución.

1.2.1. Misión.

El municipio de Ibarra planifica, regula, ejecuta y promueve el desarrollo integral sostenible del cantón, a través de servicios de calidad eficientes y transparentes con la participación activa de la ciudadanía socialmente responsable a fin de lograr el buen vivir.

1.2.2. Visión.

Seremos un municipio líder en gestión con responsabilidad social, que garantice equidad, honestidad, trabajo y eficiencia por qué Ibarra se constituya en un cantón próspero, atractivo e incluyente, capital de los servicios y el conocimiento, referente del buen vivir en la región norte del Ecuador.

1.2.3. Objetivos Generales.

El Plan Estratégico del Municipio de Ibarra 2010-2014, por una Ibarra hacia el futuro, segura, productiva y social, determina cuatro temas estratégicos en base al análisis FODA, para identificar los cinco objetivos generales.

- 1. Identidad, Participación y Ciudadanía:** Construcción de nueva ciudadanía, multiétnica y pluricultural, con participación ciudadana, incrementando la seguridad, la protección del medio ambiente, la esperanza y la calidad de vida de los ciudadanos.
- 2. Recuperación Económica, Desarrollo y Competitividad:** Impulsar el crecimiento de la economía, a partir del mejoramiento de la competitividad y productividad, facilitando la instalación de actividades productivas.
- 3. Desarrollo Territorial:** Orientar el desarrollo físico y ambiental del municipio de forma que permita elevar la calidad de vida de sus habitantes, consolidar y desarrollar el espacio público de manera equitativa y mejorar la imagen urbana.

4. Desarrollo Institucional: Garantizar una administración municipal con responsabilidad social, bajo los principios de eficiencia, eficacia y transparencia, con procesos y procedimientos estandarizados.

1.2.4. Objetivos Específicos.

1. Construir una ciudad incluyente, solidaria, con la participación de los ibarreños e ibarreñas en todos los aspectos de la vida de la ciudad. Auspiciar la igualdad, la cohesión y la integración social y territorial, con respeto a los ciudadanos con capacidades especiales y de la tercera edad.
2. Aumentar la esperanza y la calidad de vida de la población, a través de un medio ambiente sano y sustentable, con acceso equitativo y seguro al agua, aire y suelo.
3. Promover los mecanismos necesarios para que Ibarra sea una ciudad segura, con sistemas adecuados de protección de la vida y bienes de las ciudadanas y los ciudadanos, prevención de riesgos y protección en caso de desastres, en base a mecanismos que incluyan la participación social.
4. Garantizar en coordinación con el Estado el acceso libre a una educación de calidad, en un ambiente seguro y confortable, con respeto a las culturas, las tradiciones y las etnias, que integre contenidos locales, de manera que se garanticen procesos identitarios a largo plazo, así como vincular la educación a los objetivos de desarrollo del cantón.
5. Consolidar estrategias que integren el deporte, el ejercicio y la recreación como partes fundamentales del mejoramiento de calidad de vida de los habitantes del cantón, fortaleciendo el uso de los espacios públicos y de encuentro común.
6. Promover el desarrollo de las culturas karanki, Imbayacuna, Cayambi, Natabuela, Awá, Afrochoteña y mestiza, así como de los aportes de los migrantes, a través de mecanismos de recuperación de memoria colectiva, difusión de los artes y saberes tradicionales, educación en contenidos propios de cada cultura, preservación de valores y potenciación de su desarrollo futuro, con respeto, equidad e integración.
7. Promover el desarrollo de las culturas karanki, Imbayacuna, Cayambi, Natabuela, Awá, Afrochoteña y mestiza, así como de los aportes de los migrantes, a través de mecanismos de recuperación de memoria colectiva, difusión de los artes y saberes tradicionales, educación en contenidos propios de cada cultura, preservación de valores y potenciación de su desarrollo futuro, con respeto, equidad e integración.
8. Proyectar a la comunicación como herramienta transversal en los procesos participativos, sociales, de salud y medio ambiente, de cultura, educación, deportes y recreación, utilizando para el efecto todas las tecnologías para mantener informados a los ciudadanos.

1.2.5. Organigrama Estructural por Procesos.

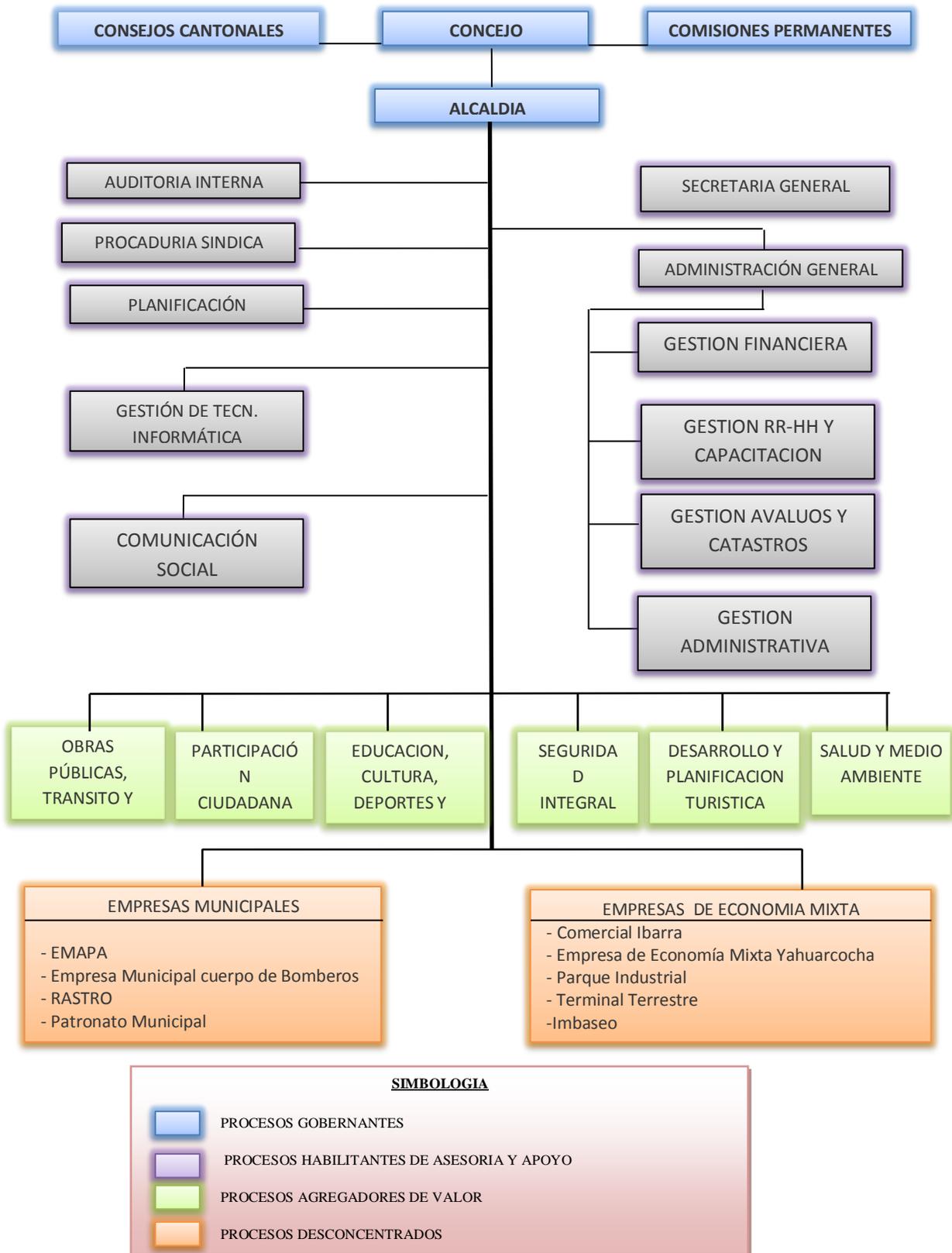


Figura 1: Organigrama estructural por procesos del Ilustre Municipio de Ibarra.
Fuente: Resolución administrativa – Gestión Organizacional por Procesos - IMI 2010

1.2.6. Dirección de Avalúos y Catastros.

1.2.6.1. Misión.

Administrar, mantener y actualizar el sistema catastral del Cantón Ibarra de manera integral, con fines de planificación, control físico espacial y determinación tributaria.

1.2.6.2. Funciones.

1. Determinar las políticas, objetivos y metas de la Dirección de Avalúos y Catastros en función de la visión y misión institucionales.
2. Establecer políticas de intervención catastral para la ejecución de planes, programas y proyectos catastrales, con sujeción a los procesos de modernización del estado.
3. Planificar los procesos de modernización de la Dirección de Avalúos y Catastros mediante la implementación integrada del sistema de información catastral y sistemas conexos.
4. Estructurar las funciones, actividades técnico-operativas y rutinas que deben ejecutarse anualmente en las Unidades de Avalúos y Catastros, en sus diferentes áreas de trabajo.
5. Elaborar el Plan Operativo Anual y Plan de Inversiones de la Dirección de Avalúos y Catastros en concordancia con el planeamiento anual municipal.
6. Reglamentar, controlar y supervisar la administración de los recursos humanos, materiales y equipos asignados a la Dirección, en coordinación con las Direcciones: Administrativa, Financiera, Jurídica, Planificación y Auditoría Interna municipales.
7. Asesorar a la máxima autoridad, concejo y comisiones de la municipalidad.
8. Estructurar y dirigir todas las actividades en el mantenimiento catastral de la información predial, cartográfica y de los sistemas que se utilizan para el efecto.
9. Coordinar y supervisar la información estadística legal, física, económica, tributaria inherente al catastro.
10. Planificar normas y procedimientos para la determinación y aplicación de la nomenclatura vial y domiciliaria.

1.3. Problema.

En la actualidad el Municipio de Ibarra cuenta con un sistema GIS, en el cual la información espacial se maneja a través de archivos planos, haciendo estos que las consultas de predios y análisis de la información geográfica sea lenta y dificultosa, además el acceso de los usuarios no es concurrente y es limitado solo a la red local del Municipio.

El sistema de Catastros y Avalúos hace uso de este sistema GIS, el cual a su vez funciona en un entorno de escritorio en Visual Basic, el registro y consulta de un predio y cálculos de impuestos se realizan directamente en este sistema, y a través de la interfaz GIS, se obtiene la clave catastral de un predio y se accede a la información del dueño.

El Sistema GIS actualmente funciona en un entorno de escritorio a través de un control Active X llamado MapObject y Visual Basic, lo cual limita mucho su rendimiento ya que toda la información geográfica se maneja a través de shapefiles, el cual es un formato vectorial de almacenamiento digital donde se guarda la localización de los elementos geográficos y los atributos asociados a ellos. El formato carece de capacidad para almacenar información topológica, es decir limita obtener las relaciones espaciales entre los diferentes elementos gráficos (topología de nodo/punto, topología de red/arco/línea, topología de polígono) y su posición en el mapa (proximidad, inclusión, conectividad y vecindad).

1.4. Objetivos.

1.4.1. Objetivo General.

Implementar un Sistema de Información Geográfica Web que permita la interacción con el Sistema de Catastros y Avalúos del Municipio de Ibarra, de acceso rápido y concurrente para los usuarios.

1.4.2. Objetivos Específicos.

1. Investigar el Módulo PostGIS, el cual permite añadir soporte de objetos gráficos a la base de datos PostgreSQL.
2. Seguir estándares de Sistemas de Información Geográfica, como la OGC (Open Geoespacial Consortium) que es una fundación que crea estándares abiertos.
3. Seguir la Norma ISO 19110 para infraestructura de los datos espaciales.
4. Mejorar el acceso a la información catastral del Ilustre Municipio de Ibarra, un acceso concurrente y reducción del tiempo de operación.
5. Realizar un diseño de la base de datos espacial y migrar la información espacial que se maneja actualmente a través de archivos shape.
6. Mantener la información geográfica centralizada en una base de datos permitiendo que la información sea transaccional y de acceso rápido.
7. Conocer los diferentes lenguajes de Marcado para la representación de datos geográficos y servicios de información geoespacial en la web.

8. Investigar sobre el sistema de coordenadas cartográficas mundial WGS84, el cual permite localizar cualquier punto en la Tierra.
9. Realizar las pruebas, validar e implementar el software para el Ilustre Municipio de Ibarra.

1.5. Justificación.

Un Sistema GIS permite mapear cualquier información almacenada en una plantilla o base de datos, que tenga un componente geográfico, que permita ver patrones, relacionales y tendencias, que no pueden verse en un formato de tabla o lista. Da una perspectiva totalmente nueva y dinámica de la información y ayuda a tomar mejores decisiones.

El Sistema GIS que actualmente se encuentra en funcionamiento está desarrollado con Visual Basic y MapObjects, lo cual lo limita a una funcionalidad cliente Servidor, en la red local del Ilustre Municipio de Ibarra y a su vez la información geográfica se encuentra almacenada en archivos Shape, lo cual complica realizar procesos de análisis geográfico sobre la misma.

Por ello se pretende el desarrollo de una herramienta, que permita mejorar el acceso a la información de la ubicación de manzanas, calles y predios de la ciudad, de una manera más sencilla y que permita establecer relaciones más precisas entre la información y la zona que se encuentra en análisis. Además que sirva como base para la integración con varios sistemas transaccionales que operan en el Ilustre Municipio de Ibarra.

La utilización de una base de datos espacial permite obtener información vectorial de la geografía a analizar, permitiendo a su vez realizar consultas más rápidas, por esta razón se pretende utilizar el Módulo PostGIS para Postgresql, que es un módulo que permite agregar funcionalidades de almacenamiento de objetos GIS en dicha base de datos. Además PostGIS sigue la especificación SFS (Simple Features Specification for SQL) del OGC (Open GIS Consortium).

1.6. Alcance.

El Sistema de Información Geográfica permitirá obtener información oportuna principalmente para el departamento de Avalúos y Catastros, aportando información locacional, coordenadas de longitud y latitud, propiedades, etc. del Manzanero, calles y predios.

Permitirá a su vez un rápido acceso a un sector en específico, a través de consultas, mapeo de la información para realizar un mejor análisis con la información centralizada dependiendo del parámetro de consulta.

Capas del Sistema:

- 1) **Capa de Calles.-** Permitirá la identificación, ubicación de las calles y consulta de cruces.
- 2) **Capa de Manzanas.-** Permitirá obtener la ubicación y consulta de las claves catastrales de cada zona.
- 3) **Capa de Predios.-** Permitirá consultar el pago de los impuestos prediales.
Diagnóstico y consulta de las propiedades que no han pagado a través de una interpretación geográfica diferenciada por tonos de colores.

Reportes y Ubicación:

Permitirá obtener reportes y ubicación rápida de una zona geográfica:

- Clave catastral del Predio.
- Informe del propietario de un predio.
- Características de un Predio.
- Impuesto a pagar del predio, atrasos, número de propiedades de un persona.
- Análisis sobre una zona.



Capítulo II: Marco Teórico

CONTENIDO:

1. Sistemas de Información Geográfica.
2. Intercambio de datos Espaciales en la Web.
3. Base de datos PostgrSQL y Postgis.
4. Servidores de Aplicaciones.
5. Lenguajes de Programación.
6. Frameworks.
7. Metodología XP.
8. Algunas definiciones.

2. Capítulo II: Marco Teórico.

A continuación se describen algunos conceptos que luego serán necesarios conocerlos para el desarrollo del Aplicativo.

2.1. Sistemas de Información Geográfica.

2.1.1. Definición.

Un GIS puede definirse como un conjunto de software, hardware, metodologías, datos y personas (usuarios), perfectamente integrados, de forma que hace posible la recolección, almacenamiento, procesamiento y análisis de datos georeferenciados, así como la producción de información de su aplicación, con el fin de resolver problemas de gestión y de planificación. [LIB 1].

2.1.2. Características.

Algunas de las características principales de un GIS son:

- Su propósito es la visualización de información geográfica expresada en forma de mapas. Para una determinada porción del territorio los datos se habrán de organizar, de manera inteligente, en diversa capas (layers), cada una conteniendo un aspecto o parte de la realidad, por ejemplo la topología, el parcelario, la hidrografía, los asentamientos, los centros de salud, el mobiliario urbano, etc.

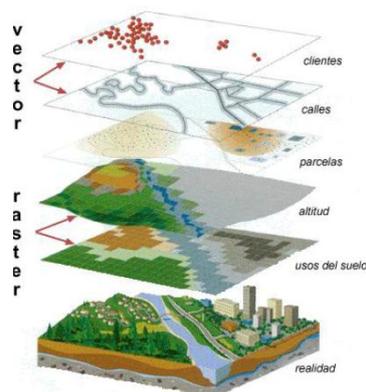


Figura 2: Información de un Sistema GIS.
Fuente: [WEB 64].

- La clave reside en la relación entre la posición un elemento geográfico, representado por puntos, líneas, polígonos y su información temática asociada.

- Ha de contener un elevado número de procedimientos de geometría computacional, de algoritmos informáticos, de métodos cuantitativos, etc.
- Almacena relaciones espaciales entre diferentes elementos.

2.1.3. Componentes de un GIS.

- **Usuarios:** Dependiendo de su especialización tendrán una exigencia distinta sobre el sistema.
- **Hardware:** Equipo informático, sobre el cual opera el GIS, servidores centralizados, operaciones aisladas o en red.
- **Software:** Programas informáticos que proveen las funciones y herramientas necesarias para visualizar, almacenar, consultar y analizar datos geográficos.
- **Datos:** Información geográfica y alfanumérica, la cual es la componente más importante de un GIS y es fundamental su correcta actualización. [LIB 1].
- **Métodos:** Son las reglas del negocio que limitan o regulan las prácticas de la organización.

2.1.4. Funcionalidad.

Las prestaciones que los GIS suelen proporcionar, son muy variables entre aplicaciones, podríamos enumerarlas de la siguiente manera:

- Entrada y captura de datos: entrada manual de datos, importación de otros archivos de datos digitales, conexión a dispositivos de captura de información. (por ejemplo GPS).
- Administración y organización informática de los archivos de geodatos.
- Edición, corrección, integración y geo procesamiento de los datos: modificación de la geometría, coordenadas, tablas de datos temáticos, generación de nuevas unidades espaciales (pixel, polígonos, etc.), normalización de datos dispares.
- Búsquedas o selecciones: consultas simples y complejas con criterios espaciales, temáticos o mixtos. [WEB 64].
- Obtención de datos derivados: ejecución de cálculos (simples o complejos) y medidas con los datos disponibles.
- Análisis: aplicación de técnicas diversas de estadística convencional y espacial, métodos de optimización, evaluación multicriterio, etc.
- Modelado: reconstrucción de aspectos de la realidad a partir de muestras o datos incompletos, obtención de similitudes de sistemas, predicciones, estimaciones, etc.
- Elaboración y visualización de mapas bi y tridimensionales, imágenes, gráficos y tablas, vuelos virtuales, etc.
- Servicio remoto de información geográfica bajo demanda de los usuarios.

- Impresión y exportación de mapas, gráficos, datos y realidad virtual del territorio.

2.1.5. Aplicaciones de un GIS.

Los GIS pueden ser aplicables a cualquier ámbito en el que sea necesario resolver un problema asociado a una variable espacial.

Es destacable también como los cambios socio-tecnológicos han contribuido a la popularización de los GIS, como la información visual, por lo que es muy habitual apoyar todo tipo de argumentos con datos geográficos. Por otro lado el intercambio masivo y libre de información geográfica a través del internet ha posibilitado referenciar una idea con una posición en la tierra.

Actualmente los campos de aplicación más usuales de los GIS son los siguientes:

- Planificación urbana y regional.
- Ingeniería de Transportes.
- Explotación de Recursos.
- Análisis de Mercados.
- Aplicaciones de Seguridad Pública.
- Aplicaciones de Sanidad Pública.
- Turismo.
- Prevención de Riesgos Naturales.

2.1.6. Información Geográfica.

2.1.6.1. Definición.

La información geográfica se refiere a la información que se puede ubicar en el espacio, sea con referencia a un sistema de coordenadas o a un orden topológico.

El conjunto de datos espaciales y no espaciales, constituyen la base de datos, que es el componente principal sobre el que se basan los análisis y resultados producidos con un Sistema de Información Geográfica. [WEB 75]

2.1.6.2. Componentes de la Información Geográfica.

Los datos que un GIS integra responden a tres componentes o dimensiones: la espacial, la temática y la temporal.

- La Componente Espacial: Alude a la información sobre la localización (coordenadas), las propiedades espaciales (tales como longitud, superficie, forma, etc.) y las relaciones espaciales.

- La Componente Temática: Alude a las características o atributos de las entidades reales, que se traducirán, bien como números, bien como texto o caracteres alfanuméricos, aunque éstos cabe representarlos también como códigos numéricos. Los datos deberán ser manejados con procedimientos acorde con su naturaleza, como a la hora de representarlos cartográficamente (uso de la simbología apropiada), como a la hora de realizar operaciones aritméticas o lógicas.
- La Componente de Dimensión Temporal: Esta nos recuerda que, por un lado, los datos están siempre referidos a un momento, por lo que se debe disponer de una perspectiva diacrónica del territorio para visualizar, identificar, cuantificar o modelar los cambios.

2.1.7. Modelo de Datos.

Un Modelo es una representación de la realidad y se genera mediante la selección y la simplificación de sus partes. Para que sea geográfico, el modelo ha de poseer un sistema de referenciación.

Existen dos formas de representar la información geográfica en un sistema informático, el modelo raster y el modelo vectorial. La diferencia entre uno y otro está en la manera de almacenar la información geográfica. Mientras que el modelo vectorial almacena las coordenadas de las formas geográficas que definen cada entidad, el modelo raster almacena una matriz de posiciones que adoptan el valor de la entidad que se representa en cada posición.

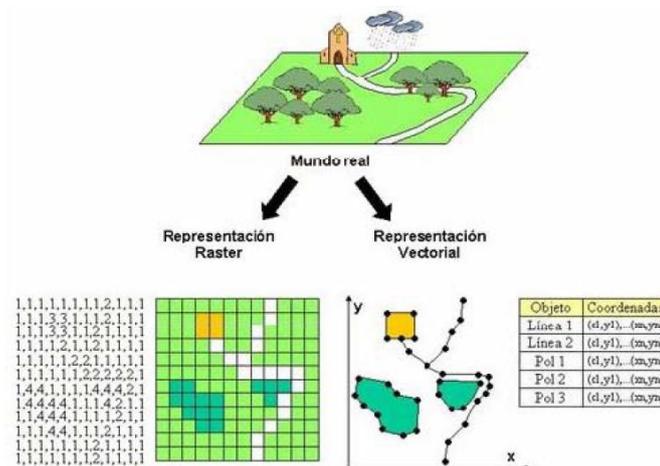


Figura 3: Modelo de datos Geográfico.
Fuente: [WEB 75]

2.1.7.1. Modelo Vectorial

Es la representación de los fenómenos geográficos mediante puntos, líneas y polígonos. Estos elementos se almacenan mediante el conjunto de coordenadas que forman su geometría.

Si por ejemplo se trata de un punto, solo se almacenan un par de coordenadas. Si se trata de objetos lineales, se almacena un conjunto de pares de coordenadas. Y si se trata de un objeto poligonal se almacena el conjunto de pares de coordenadas que define el contorno de dicho objeto. [WEB 76]

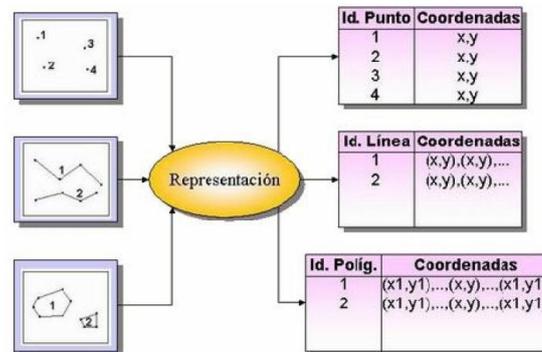


Figura 4: Modelo Vectorial.
Fuente: [WEB 76]

El modelo vectorial ha sido la base de la cartografía analógica clásica, puesto que resulta útil para representar entidades geográficas discretas, como edificios, carreteras o límites municipales.

Elementos Geográficos.

Los puntos, líneas y polígonos son los elementos geográficos básicos y basa su representación a través de vértices y segmentos.

- Vértice: Resulta del cruce de dos coordenadas X e Y, dependiendo sus valores de las características del sistema de referencia.
- Punto: Se representa por un único vértice, que se define geoméricamente, por sus valores X-Y.
- Línea: Están formadas por uno o varios segmentos articulados por vértices con sus respectivas coordenadas.
- Nodos: Son los vértices generados en los extremos, estos marcan el inicio y final de las líneas, y en consecuencia, el sentido de su dirección.
- Polígonos: Son líneas cerradas donde los nodos de inicio y final coinciden, delimitando en su interior un área.

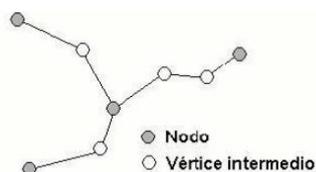


Figura 5: Elementos geográficos de un modelo vectorial.
Fuente: [WEB 76]

Los elementos vectoriales se convierten en objetos cuando su representación se asocia a entidades reales: una línea puede representar el discurrir de un río.

Formatos de los Geodatos

En cartografía digital, se entiende por formato la propuesta de los distintos desarrolladores para almacenar y tratar los datos digitales referidos a las entidades geográficas. La mayor parte de los formatos se basan en modelos genéricos de datos, esto es, en determinadas estructuras de distribución de los datos dentro de documentos. **[LIB 1]**

La Casa ESRI ha innovado, a lo largo de sus más de treinta años de existencia, tres formatos propios, para su cartografía vectorial.

- **Coberturas:** Está basado en el modelo arco-nodo y es un formato complejo que agrupa muchos archivos en un entorno de almacenamiento muy rígido y estructurado, y que diferencia claramente en distintas capas los elementos geográficos en función de su relación con las entidades geográficas (diferenciando por ejemplo, entre polígonos simples y regiones complejas).
- **Shapefiles:** Los shapefiles (archivos de forma), son muchos los datos geográficos disponibles en este formato, teniendo una estructura y configuración más simple que las coberturas puesto que no almacena todas las asociaciones topológicas entre los diferentes objetos y clases de objetos. Cada shapefile almacena elementos pertenecientes a una única clase (puntos, líneas, polígonos).

Tipos de Elementos que Contiene

Puede haber shapefiles de los siguientes tipos:

- Puntos o Multipuntos.
- Líneas, que pueden ser líneas simples o continuas.
- Polígonos, pueden ser de áreas, o áreas divididas en partes. Estos pueden superponerse, pero no pueden almacenar relaciones topológicas entre ellos.

Estructura de los SHAPEFILES

Los shapefiles se representan como un conjunto de ficheros, con el mismo nombre con diversas extensiones: tres ficheros básicos, dos índices espaciales, y dos índices de atributos. Todos ellos escritos en código binario, salvo el caso del fichero de base de datos que lo está en mayor parte en código ASCII.

Ficheros Básicos

Están compuestos por tres extensiones, el núcleo .shp (shape), el índice de los registros .shx (index shape) y las bases de datos con los atributos .dbf (database file).

- **.shp:** Almacena las características geográficas, con una lista de pares de coordenadas X-Y.
- **.shx:** Contiene un índice de cada registro, es decir, un registro del número de registros y la longitud de cada registro existente en el .shp.
- **.dbf:** Guarda la información de los atributos y sus características, conteniendo un registro de cada elemento de los .shp.

Índices Espaciales

Existe cuando se ejecutan operaciones como uniones o selecciones espaciales, dos son los archivos creados: .sbn y .sbx.

- **.sbn (spatial bin):** Divide el área de los elementos geográficos de los .shp en áreas rectangulares denominadas “bin” o recipientes. Cada recipiente contiene el número de registros de cada elemento que cae dentro de esa área.
- **.sbx (spatial bin index):** Contiene una longitud de cada registro existente en los .sbn.

Índices de Atributo

Estos se crean cuando se realiza una unión de tablas.

- **.ain (attribute index):** Contiene un índice por cada campo incluido en la unión.
- **Aih (attribute index header):** Contiene el nombre de cada campo que ha sido indexado y sirve como un directorio de valores contenidos en un archivo .ain.

• Bases de Datos (Geodatabase)

Una base de geodatos almacena cada elemento como una fila en una tabla. Un elemento vectorial es identificado en un campo denominado shape de la tabla, con los atributos en otros campos de la misma. Cada tabla almacena un conjunto de elementos con propiedades comunes a los que se les denomina feature class (elemento de clase). Junto a estos elementos, las bases de geodatos pueden almacenar raster, tabla de datos y referencia de otras tablas.

Los elementos en las Bases de Geodatos

Al ser posible personalizar los elementos creados, el número de elementos con propiedades comunes es ilimitado. La geometría básica de los elementos de clase está constituida por puntos, multipuntos, bifurcaciones de red (network junctions), líneas, bordes de red (network edges) y polígonos. Se puede además crear elementos con nuevas geometrías.

La Topología en las bases de Geodatos

La topología en una base de geodatos permite representar la forma geométrica entre elementos en un elemento de clase y entre diferentes elementos de clase. Las bases de geodatos amplían la capacidad de controlar las relaciones entre los elementos basados en geometría para posibilitar la basada en el comportamiento de las entidades geográficas.

2.1.7.2. Modelo Raster.

Es la representación del mundo real continua, mediante una superficie dividida en discretas celdas regulares, denominadas cuadrícula o grid, y donde cada celda tiene el nombre de pixel. Cada pixel representa una cualidad cuantificable de observación, la mínima, representada en cada localización mediante un tono o color, que se traduce a un valor numérico o nivel digital (ND). [WEB 76]

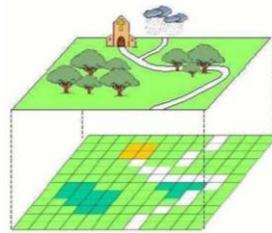


Figura 6: Modelo Raster.
Fuente: [WEB 65]

Características de los formatos Raster

Los formatos raster siguen las normas generales de las imágenes digitales.

Estructura de las imágenes raster

Tienen una estructura similar a un malla, normalmente regular, a cuyas celdas se le asigna un número (ND). Este número suele asociarse a un color, lo que conforma la apariencia visual de la imagen.

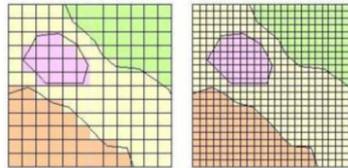


Figura 7: Estructura de las imágenes raster.
Fuente: [WEB 76]

La estructura de almacenamiento más simple es una única columna de números.

200	25	0	14	19
80	10	99	35	65
228	5	18	65	185
190	0	225	226	255

Tabla 1: Estructura de almacenamiento raster con una única columna de números.
Fuente: [LIB 1]

La forma de almacenar la información suele ser comenzando a contra por el pixel superior izquierdo, de izquierda a derecha y de arriba hacia abajo.

Nro Pixel	Valor
0	200
1	25
2	0
3	14
4	19
5	80
6	10
7	99
8	35
9	65
10	228
11	5
12	18
13	65
14	185
15	190
16	0
17	225
18	226
19	255

Tabla 2: Orden en el que se almacenan los datos en un raster.
Fuente: [LIB 1]

De esta forma, podríamos formar una relación de datos que, posteriormente, serán asignados a sus respectivas celdas. Luego se le asigna un color a cada número extrayendo de una paleta en escala de gris con 255 matices o con una paleta de color con diversas combinaciones.

Los números utilizados por las imágenes raster pueden ser reales, enteros, byte.

Las imágenes pueden almacenarse en archivos de tipo ASCII o binario o como binario empaquetado.

Medidas de los pixeles y resoluciones de las Imágenes.

- Cuantos más pequeños sean los pixeles, mayor nitidez darán las imágenes, aunque esto conlleva una mayor utilización de memoria y tiempo de manipulación, dado que cada pixel ocupa un determinado número de bits.
- La resolución de una imagen se mide por el número de píxeles que ocupa una pulgada, esta se expresa en puntos por pulgada (ppp, ppi o dpi)

La creación de pirámides.

Para acelerar la velocidad de visualización de los documentos raster con un número de píxeles, es posible crear una pirámide con distintos niveles de resolución. Esto permite que, al abrir una imagen raster, se muestre una capa de nivel de baja resolución, mientras se prepara el conjunto de los datos, además cada vez que se haga un zoom, aparecerá el nivel de resolución adecuado.

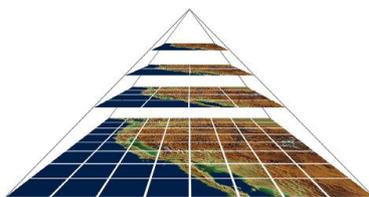


Figura 8: Pirámide de una imagen raster.
Fuente: [WEB 65]

Georeferencia de Imágenes

Para poder utilizar una imagen Raster en un GIS junto con otras capas de información, debemos asignarles coordenadas, esto es, georeferenciar las imágenes.

Formatos de Imágenes

MULTIBANDAS (.BIL, .BIP, .BSQ)

Entre los formatos más utilizados para almacenar la información de las imágenes, como por ejemplo las provenientes de los satélites Spot, se encuentran los formatos binarios con extensiones:

- **.bil (Band Interleaved by Line):** No posee datos en la cabecera ni en la cola de la imagen y se caracteriza por combinar la información de las bandas almacenando de forma contigua los píxeles cada línea de cada una de las bandas. Como no posee información incrustada, viene acompañado de un documento de texto que reflejan los metadatos de la imagen.
- **.bip (Band Interleaved by Pixel):** La información de cada banda se almacena de forma intercalada por píxel, esto es, valor de píxel 1 en cada banda.
- **.bsq (Band Sequential):** La información de cada banda se almacena consecutivamente, lo que permite una rápida diferenciación en las bandas.

ERDAS LAN (.LAN) Y GIS (.GIS)

Son formatos utilizados con las versiones 7.3 y 7.4 del programa ERDAS, durante mucho tiempo constituyó un estándar entre los formatos de imagen. Almacena imágenes de multibanda continuas.

- **.GIS:** Hace referencia a una banda temática en la cual los píxeles están divididos en categorías discretas.
- **.LAN:** Poseen información sobre la imagen alojada en la cabecera del documento.

ERDAS IMAGE (.IMG)

Es el formato nativo del programa actual ERDAS IMAGE, donde las imágenes se almacenan tanto en monobanda como en multibanda. Es un formato muy avanzado y complejo que admite profundidades de hasta 128 bits.

TIFF (.TIF) TAG IMAGE FILE FORMAT

Este formato se desarrolló en 1986 por las casa Aldus con el objetivo de que fuese fácilmente intercambiable entre plataformas, lo que se consigue con la utilización de etiquetas (Tags) en la cabecera de la imagen, con directorios de entrada de imagen (FID), que almacenan con códigos prefijados, los diversos datos necesarios para su interpretación. Desde la versión 6 se utiliza también como formato cartográfico denominado GeoTIF, al poder reservar varias de las etiquetas para describir, mediante códigos numéricos, el tipo de proyección, el sistema de coordenadas, el datum y la georreferenciación de la imagen.

WINDOWS BITMAP (.BMP)

Es un formato de imagen de mapa de bits (bitmap images) estándar de Windows en DOS y en ordenadores compatibles con Windows. las imágenes almacenadas pueden poseer una profundidad de bit de 1,4,8 ó 24 bits por pixel, lo que permite una mayor calidad desde el blanco y negro al color.

IMPELL BITMAP, FICHEROS COMPRIMIDOS (.RLC)

Asociado con el sistema de compresión run-length (RLC), es utilizado para escanear imágenes monocromáticas. El origen de las coordenadas de la imagen se sitúa en la parte superior, aumentando de izquierda a derecha y de arriba hacia abajo. El archivo se interpreta como una secuencia de 16 bits donde cada palabra corresponde a un número entero sin signo.

SUM RASTER (.RS, .RAS Y .SUN)

Almacena información en la cabecera de la imagen y se la somete a la compresión sin pérdida de la información RLE (Run-Length-Encode), almacenándose los datos en formato BIP.

Formatos de Compresión

Existen dos formas de comprimir documentos: con pérdida de información (loss) y con altos ratios de compresión, y sin pérdida de información (lossles) y con ratios moderados.

En las compresiones con pérdida de información tenemos:

- **JPG (.JPG):** Es una técnica establecida por Join Photographic Experts Group, de ahí su nombre, que alcanza ratios de 20:1 o superiores (lo que significa que una imagen de 20 Mb se puede comprimir en 1 Mb). Para alcanzar este ratio de compresión se desecha un número determinado de información redundante de la imagen, preservándose siempre sus aspectos esenciales, lo que implica pérdida de datos; por lo cual no es recomendable para procesos de análisis. Como mejora se ha incorporado un documento de texto con el mismo nombre de la imagen

comprimida y con extensión .JGW que almacena los metadatos de la imagen, permitiéndola ser referenciada en un sistema GIS.

- **MRSID (Multiresolution Seamless Image Database):** Es un compresor multiresolución, mediante pirámides con altos ratios, basado en el comportamiento de las ondas, lo que permite un rápido acceso a los datos en cualquier escala.

2.1.7.3. Orientada a Objetos

Esta representación es heredada de la programación orientada a objetos, por lo que comparte buena parte de la lógica y terminología de esa parcela de información. Un objeto se refiere a una entidad real, por ejemplo, un río, un autobús, etc. Su representación informática, aparte de vincularse a una determinada figura geométrica posee muchas posibilidades de comportarse.

Los objetos se agrupan en clases y subclases y cada una de ellas posee asociada un tipo de conducta bien definida. Dicho comportamiento en la realidad se traduce como operaciones que dicho objeto puede realizar.

2.1.8. Sistema de Coordenadas y Proyección.

La localización de los lugares en una superficie terrestre y su representación sobre un plano requiere de dos procesos:

- La construcción de un sistema de coordenadas geodésicas o geográficas, que asuman unas dimensiones bien definidas de la Tierra.
- La elección de un tipo de proyección que transforme su superficie tridimensional en plana.

2.1.8.1. Sistema de Coordenadas Geográficas.

Las coordenadas geográficas van a definir un sistema de referencia tridimensional de localización en la superficie de la Tierra. Esto requiere de tres elementos:

- Unidades de medida angulares desde el centro de la tierra (grados, radianes, etc.), que definan una red de meridianos y paralelos.
- Elección de un meridiano de origen, en muchos casos Greenwich desde los acuerdos internacionales de 1884.
- Elección de un datum. Este a su vez necesita la consideración de:
 - Un elipsoide que defina el achatamiento terrestre.
 - Un geoide que defina la forma aproximada de la Tierra.
 - Las Medidas que relacionan geoide y elipsoide en un lugar en concreto de la superficie de la Tierra.

Líneas de Referencia del globo terrestre. Unidades de Medida.

Un punto sobre la superficie terrestre está definido por sus valores de latitud, longitud y altitud. La latitud y la longitud son medidas angulares calculadas desde el centro de la Tierra. La altitud es medida desde la superficie del geoide. Para dar sentido a la medida de ángulos, se establecen dos planos de referencia que dividen la Tierra Horizontal y Vertical. **[LIB 1]**.

- El Plano Horizontal está marcado por el Ecuador, y cualquier punto situado al norte de este plano podrá ser medido en grados, denominado latitud N, o grados con signo positivo, y cualquier punto situado al sur será medido en grados de latitud S o grados de signo negativo, con un rango entre $+90^{\circ}\text{N}$ y -90°S .
- El plano vertical es artificial y muchas naciones, en distintos periodos de la historia, han mantenido uno o varios meridianos de origen en sus cartografías. Los grados hacia el este o hacia el oeste del meridiano de origen serán medidos en grados de longitud, grados con signo positivo hacia el este y con signo negativo hacia el oeste, con un rango $+180^{\circ}\text{E}$ y -180°O .

En esta retícula el punto de origen (0,0) sería el lugar de cruce del Ecuador con el meridiano de origen.

Esferas y Geoides

El sistema de coordenadas de la superficie de la Tierra puede ser representado sobre una superficie de esfera perfecta para facilitar los cálculos. Lo que es pertinente en muchas aproximaciones cartográficas, especialmente cuando se trabaja a escalas pequeñas o inferiores a 1:5.000.000. A estas escalas las irregularidades en la forma de la tierra y la diferencia de la altitud en su superficie son percibidas como insignificantes, bajo estas condiciones se considera que cada uno de los radios o semiejes de la esfera terrestre miden 6.370.997 m.

Si consideramos a la Tierra en su conjunto, adquiere una importancia grande a escalas mayores de 1:1.000.000, dando problemas de exactitud en la confección de mapas. Dos son la causa de esta divergencia: el achatamiento que ha sufrido la Tierra en sus Polos y la desigual distribución de las masas terrestres. La forma que adquiere la Tierra, considerando los factores anteriores, no es ya una esfera, sino más bien por su forma irregularmente única es un geoide, cuya medición se encarga la Geodesia. El Geoide no es una representación matemática, y se define como la superficie equipotencial del nivel medio del mar, lo que significa que cualquier punto ha de ser perpendicular con la dirección de la gravedad. **[LIB 1]**.

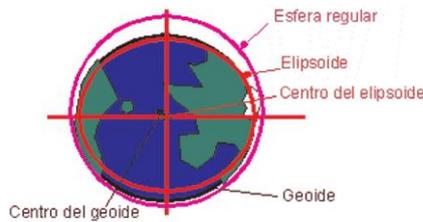


Figura 9: Esferas y Geoides.
Fuente: [WEB 78].

Elipsoides.

Para poder elaborar mapas con precisión se requiere de una superficie de referencia geoméricamente regular. Por ello, las observaciones del geoide se transfieren a una figura producida por una elipse que gira alrededor de su eje menor, obteniendo así una figura regular matemáticamente operable, como consecuencia de las dificultades que supone medir y conocer con precisión la forma exacta de la superficie de la tierra. [LIB 1].

Al no tener los dos ejes iguales, como en la esfera, la forma del elipsoide es definida mediante la relación entre los radios ecuatorial y polar. Los intentos de medir la longitud exacta de estos ejes, y de adaptar la curvatura del elipsoide, de una superficie terrestre concreta, realizada por distintos países o instituciones científicas, han dado lugar en los últimos doscientos años, diversas medidas de elipsoides.

Principales elipsoides medidos en los siglos XIX y XX	
Elipsoide	Semieje Mayor a (m)
Airy 1830	6.377.563,396
Everest 1830	6.377.276,345
Bessel 1841	6.377.397,155
Clarke 1866	6.378.206,400
Struve 1860	6.78.298,000
Clarke 1880	6.378.249,145
Hayford 1909 ó Internacional 1924	6.378.388,000
Krassovsky 1940	6.378.245,000
América del Sur 1969	6.378.160,000
WGS60	6.378.165,000
WGS66	6.378.145,000
WGS72	6.378.135,000
WGS84	6.378.137,000

Tabla 3: Principales elipsoides medidos en los siglos XIX y XX.
Fuente: [LIB 1]

Nombre del elipsoide y año de cálculo medida del semieje mayor e índice de aplanamiento.

Varios son los elipsoides globales medidos en los últimos cuarenta años, GRS67, GRS75, y GRS80 y los WGS60, WGS66, WGS72, WGS84, todos ellos con origen el centro de la Tierra.

2.1.8.2. El Datum.

Mientras que en la esfera utilizamos latitudes y longitudes geográfica, en el elipsoide serán geodésicas, y en el geoide astronómicas.

Cada sistema de referencia local dispone de un punto en donde se han hecho coincidir las verticales de las coordenadas astronómicas y geodésicas del geoide y del elipsoide, a este punto se lo denomina datum, y agrupa un conjunto de referentes que sirven para dar coherencia a todas las medidas tomadas sobre un determinado territorio. [LIB 1].

En cualquier otro punto

2.1.8.3. Los Sistemas de Proyección.

Coordenadas Cartográficas: Se denominan aquellas coordenadas terrestres ya sean geográficas, geodésicas o astronómicas que se encuentran sobre superficies en tres dimensiones, se las convierte al pasar a un mapa en coordenadas planas, en dos dimensiones.

La proyección es el proceso de transformación de coordenadas terrestres a coordenadas cartográficas. Este proceso requiere de la elección de un sistema de coordenadas cartesiano con orientación y punto de origen, y un proceso de transformación entre la esfera (superficie tridimensional de la Tierra 3D) y el plano (Mapa 2D), los cuales a su vez tendrán algún tipo de distorsión, que aumentara cuanto mayor superficie de territorio se represente.

La elección de la proyección más adecuada minimiza los efectos de la distorsión del mapa.

Existen varios tipos de proyecciones, la cuales pueden estar clasificadas según distintos criterios: según la superficie usada para realizar la proyección de la esfera terrestre (cilíndrica, cónica, etc.), según la distorsión que contienen y las características que mejor se conservan, etc.

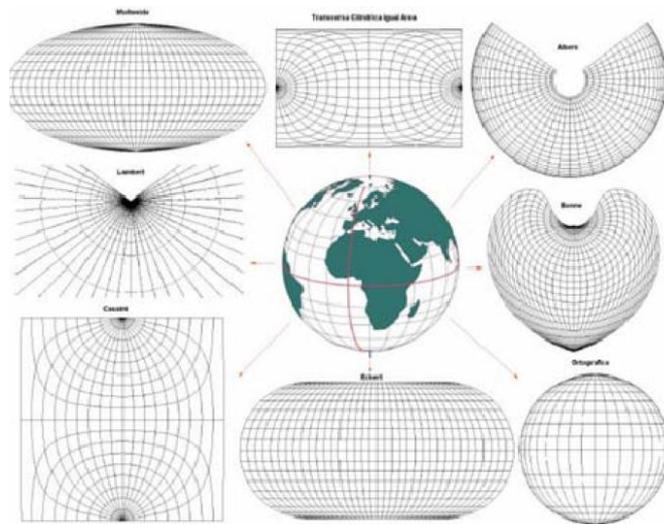


Figura 10: Tipos de proyecciones.
Fuente: [WEB 78].

2.1.9. Sistema de Referencia WGS-84

Es un sistema de referencia terrestre convencional. (“Conventional Terrestrial Reference System” CTRS). Se trata de un sistema de referencia geocéntrico fijo con la Tierra y orientado positivamente. Asociado al sistema cartesiano se considera un sistema de coordenadas geodésicas definidas por un elipsoide de revolución cuyo centro y eje de revolución coinciden respectivamente con el origen de coordenadas y eje OZ. [WEB 77].

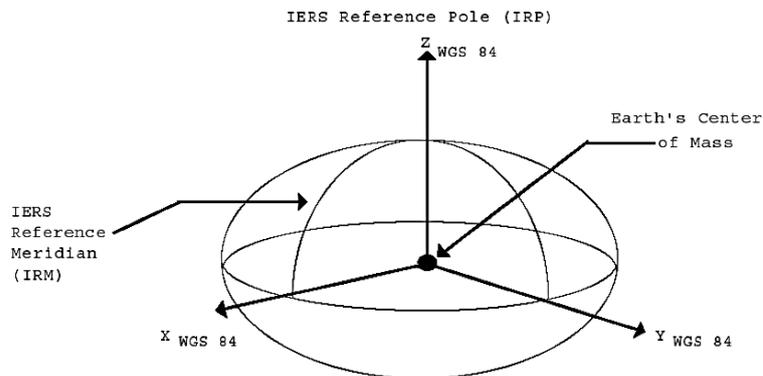


Figura 11: Sistema de Referencia WGS-84.
Fuente: [WEB 77].

2.1.10. Sistema de Referencia Provisional South American Datum 1956 – PSAD56.

Es un sistema de referencia local. Es un Sistema que tiene como elipsoide de referencia el Internacional de Hayford y como punto origen La Canoa ubicado en la República de Venezuela.

2.1.11. Captura de Información Geográfica.

Los datos geográficos pueden estar disponibles de múltiples formas, tanto analógicas como digitales (mapas, fotografías aéreas, imágenes de satélite, tablas, etc.). Una base de datos espacial puede ser construida de varios modos, o varios de estos modos simultáneamente:

- Adquirir los datos desde un suministrador especializado.
- Digitalizar datos analógicos.
- Interpolarse desde distintas observaciones puntuales a superficies continuas.



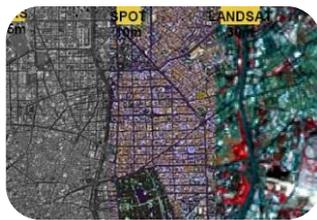
Cartografía Básica



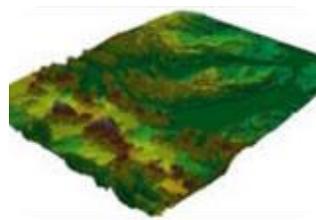
Cartografía Temática



Ortofoto



Imágen Satelital



Modelación Digital

Figura 12: Fuentes de Información Geográfica.
Fuente: [WEB 79].

Fuentes de Información geográfica.

En la captura de información geográfica es esencial prestar atención a los posibles errores cometidos durante el proceso de forma que se garantice la calidad de dichos datos y por extensión la calidad de los análisis y resultados posteriores.

Se pueden distinguir dos tipos de errores al trabajar con información geográfica en GIS:

- Errores debido al método de captura de información.
- Errores debido a la manipulación de la información dentro del propio GIS.

Ambos tipos de errores se pueden producir tanto en la información gráfica como en la información temática y tienen como consecuencia directa una baja calidad de los resultados finales.

2.1.12. Cartografía Temática.

2.1.12.1. Mapa.

El mapa es una de las formas de lenguaje que el ser humano tiene para representar el espacio que lo rodea de manera sistemática. Constituye una abstracción de la realidad mediante un croquis, esquemas y dibujos. [LIB 1].

El mapa tiene muchas ventajas a la hora de presentar la información espacial con colores, tramas, leyendas y además elementos cartográficos, en una misma hoja se puede transmitir y representar gran cantidad de información.

Simbolizar los datos implica la elección de los colores y símbolos que representarán los elementos, además de agruparlos y clasificarlos de acuerdo a sus valores.

2.1.12.2. Semiología Gráfica.

Ciencia que explica la manera correcta de utilizar toda la simbología que transmite información. Es una gramática visual, es un sistema de comunicación a través de reglas. [LIB 1].

Existe una ilimitada variedad de datos espaciales que pueden representarse mediante símbolos, éstos pueden clasificarse en tres:

- Símbolos de puntos: son individuales, se utilizan para representar datos posicionales.
- Símbolos de línea: No significa que el tipo de dato sea lineal o escalar, sino que la información los sea.
- Símbolos de zonas o área: Se extiende sobre una superficie del mapa, para atribuir o indicar cierto aspecto en común.

2.1.12.3. Elementos Gráficos Básicos.

Los símbolos, sean del tipo que sean, pueden tener valores distintos en función de su color, valor, tamaño, forma, espacio, orientación y ubicación. A este conjunto de aspectos se les denomina elementos gráficos básicos.

- El color es la apreciación visual más importante y compleja. La elección de los colores a utilizar es esencial para que el mapa esté bien presentado. Es aceptado, de modo generalizado, un juego de colores llamado **perceptivos**, que comprenden los tonos que la mayoría de la gente percibe como: el azul, verde, amarillo, rojo, marrón, negro y blanco. El resto de tonos unas personas los ven de un modo y otras de otro. Por ello se han fijado una serie de connotaciones simbólicas, es decir, asociar el color a sensaciones y estados de ánimo, como el

azul – agua y frío, verde – vegetación, amarillo – sequía y escasez de vegetación, rojo – calor y resaltar elementos importantes, etc.

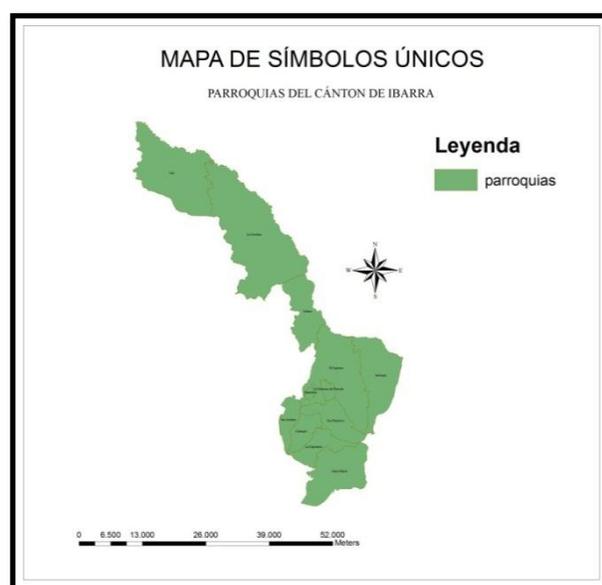
- El tamaño, se debe elegir, permitiendo que el mapa se muestre claro y legible, luego cuando se lean, se interpreten correctamente.
- Los elementos deben estar dispuesto de tal modo que su relación parezca lógica.

2.1.12.4. Mapas Temáticos.

Los mapas ayudan a tomar decisiones y resolver problemas, nos comunican una información compleja y de forma efectiva. La simbolización de los datos implica elegir colores y símbolos que representarán elementos geográficos, y clasificarlos según sus atributos. [LIB 1].

Mapa de Símbolos Únicos.

En este tipo de mapas se puede asignar un mismo símbolo a todos los objetos del mapa cartográfico.



**Figura 13: Mapa de Símbolos Únicos.
Fuente: Propia.**

Mapas Cualitativos.

La información que contienen las variables de carácter no numérico se emplean de forma cualitativa, estos permiten categorizar los atributos cualitativos. El mapa cualitativo es un tipo de representación cartográfica en el que cada valor de un atributo o característica es expresado de modo distinto.

Se suelen emplear para representar especialmente cuatro tipos de variables cualitativas:

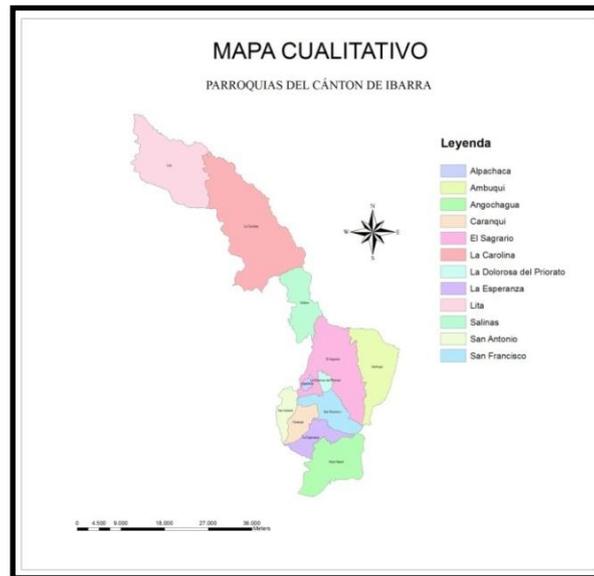


Figura 14: Mapa Cualitativo.
Fuente: Propia.

- **Atributos Categorizados:** En categorías o modalidades que son excluyentes entre sí y comprensivas de todo fenómeno. Dichas categorías podrían expresarse con nombres, símbolos, etiquetas, sin valores matemáticos y seleccionados discrecionalmente. Ejemplo: el uso del suelo.
- **Atributos Binarios o Dicotómicos:** Con valores tales como 0/1, sí/no. Ejemplo: resultados de una encuesta.

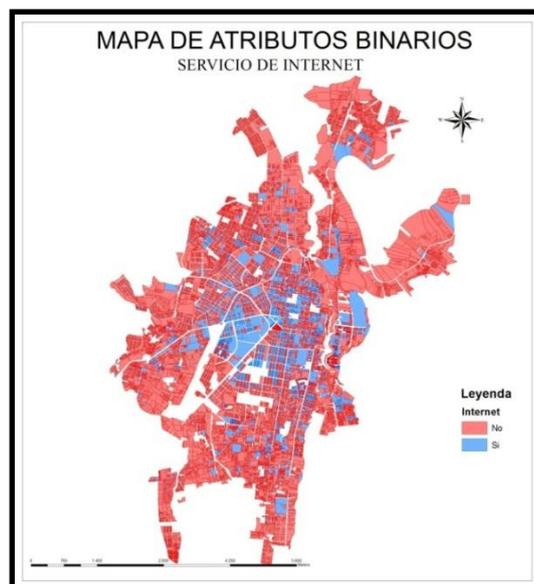


Figura 15: Mapa de Atributos Binarios.
Fuente: Propia.

- **Atributos Ordinales:** Son unidades espaciales ordenadas según un determinado criterio. Ejemplo: estado social.

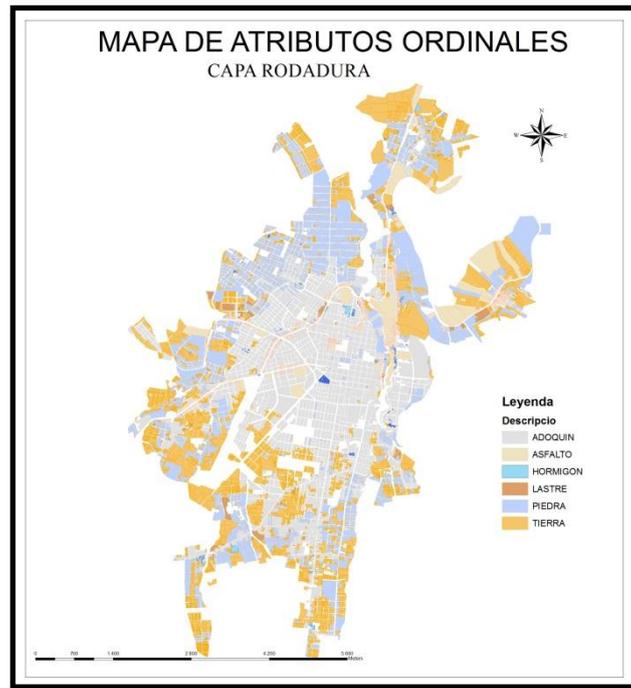


Figura 16: Mapa de Atributos Ordinales.
Fuente: Propia.

- **Atributos con una Jerarquía:** Atributos de símbolos ordenados a los individuos. Ejemplo: mapa con el resultado de una encuesta con preguntas que tienen varias posibilidades, que a su vez abren otros ítems.

Mapas Cuantitativos

Se utilizan los atributos de carácter numérico.

- **Mapa de coropletas o de graduaciones de color.**
Son mapas en los que se representa las distintas variaciones de una variable sobre un espacio dado, empleando colores o tramas diferentes. En este caso el espacio sobre el que se representan es un área o región. Los símbolos cambian de color en función de los valores que adopte la variable elegida. Son adecuados para representar datos en rangos, sean cualitativos o cuantitativos, o con algún tipo de progresión numérica, (medidas, proporciones, porcentajes, etc.), así como para variables de densidad o intensidad.

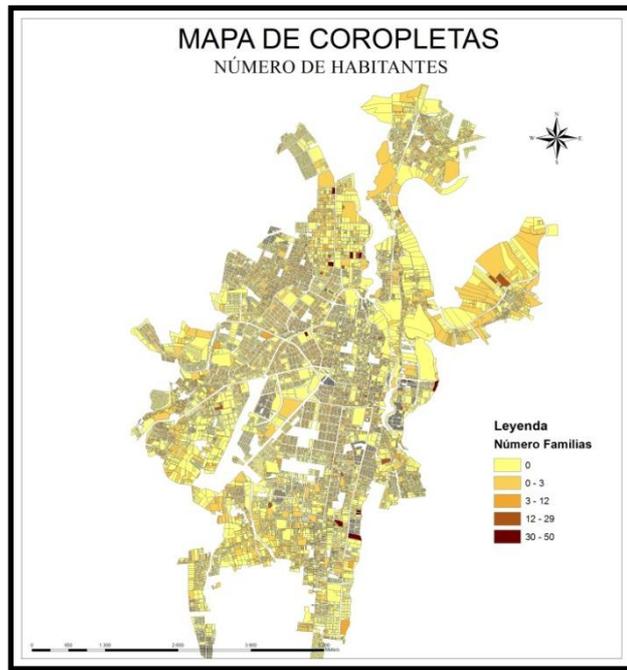


Figura 17: Mapa de Coropletas.
Fuente: Propia.

- **Mapa de Símbolos Graduados.**

Son similares a los anteriores, excepto porque se representan sobre elementos espaciales de tipo puntual o lineal. Permite variaciones en el tamaño de los símbolos de punto y en anchura de los símbolos de líneas. Son útiles para mostrar gradaciones y progresiones.

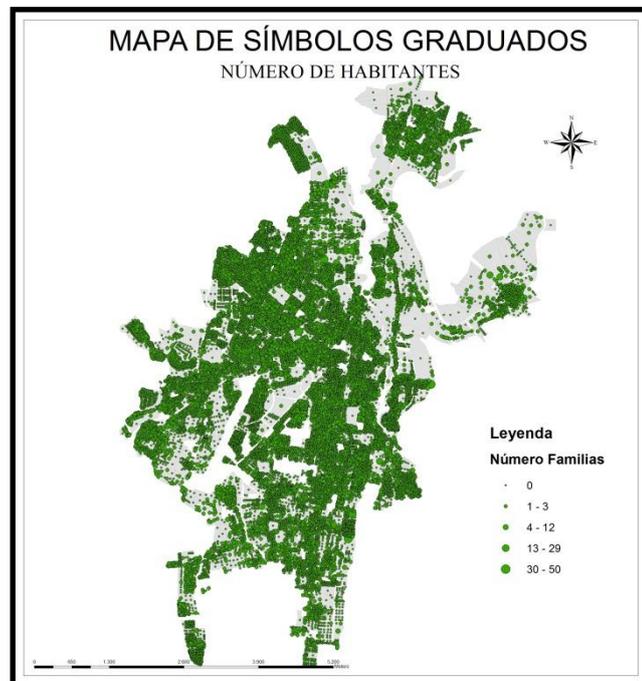


Figura 18: Mapa de Símbolos Graduados.
Fuente: Propia.

- **Mapa de Densidad de Puntos**

Utiliza puntos dentro de un polígono para representar un atributo. Cada uno de los puntos representa un valor específico y se reparten por el territorio de forma aleatoria. Es útil para representar la densidad de un suceso además de su cantidad. Por ejemplo un mapa de población, indicando la variación en el número de personas.

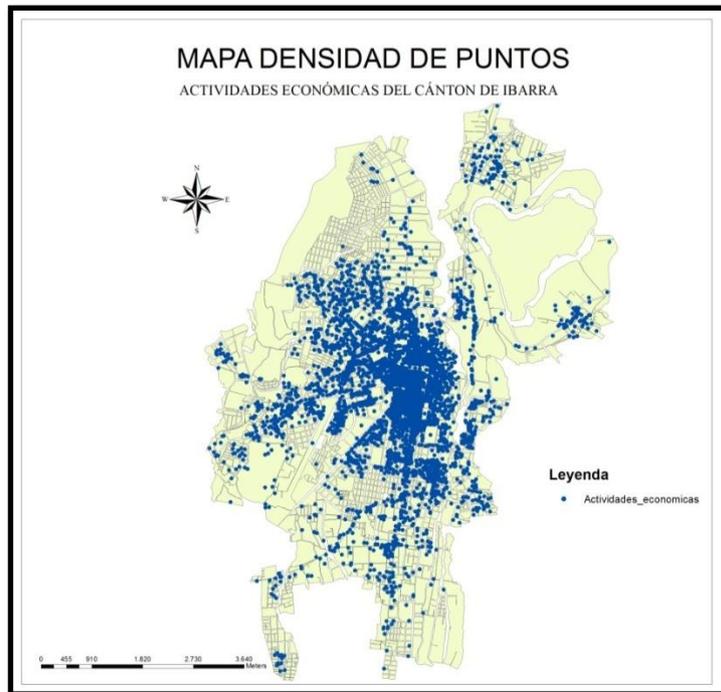


Figura 19: Mapa de Densidad de Puntos.
Fuente: Propia.

- **Métodos de clasificación de los Datos.**

Es la clasificación de un valor cuantitativo. Como:

- Intervalos basados en puntos de ruptura naturales.
- Intervalos basados en cuantiles: A cada intervalo se le asigna un mismo número de elementos: Identifica los intervalos buscando agrupamientos y patrones inherentes a los datos minimizando la variación dentro de cada clase.
- Intervalos iguales: Se agrupa en intervalos de igual amplitud o tamaño.
- Intervalos basados en la desviación estándar: La desviación estándar o típica muestra en qué grado cada uno de los valores difiere de la medida aritmética de todos los valores.

El método elegido dependerá de la naturaleza de los datos y el objetivo que se pretende obtener, ya que la elección del método determina la apariencia y el mensaje del mapa resultante.

Mapas de Múltiples Atributos

- **Mapa Multivariado**

Tiene la posibilidad de representar varias variables sobre un mismo mapa, el número dependerá del tipo de representación cartográfica que se elija, de las variables a utilizar, del entorno geográfico, etc.

- **Mapa con un gráfico**

Los mapas con gráficos son un medio para simbolizar datos y pueden incluir dos tipos: diagrama circular o de sectores, en cuyo caso, los círculos se pueden proporcionar según el peso de la variable, y gráfico de barras simples, compuestas y acumuladas que permiten comparar tanto valores absolutos como relativos y su distribución en el tiempo.

Otro Tipo de Mapas

- **Imágenes Raster.**

Obtenemos características raster de los mapas escaneados, de la fotografía de la superficie de la tierra, una foto aérea o una imagen satelital (ortofoto u ortoimagen).

- **Mapa de la Topografía o de Superficies Coloreadas.**

Los mapas topográficos son representaciones cartográficas de carácter lineal, sobre la base de isolíneas, es decir que tienen el mismo peso de la variable.

2.2. Intercambio de datos Espaciales en la Web.

2.2.1. La Open Geospatial Consortium OGC.

La OGC (Open Geospatial Consortium) nace en 1994, es una organización internacional sin fines de lucro, integrada por empresas, organismos públicos y Universidades, en un proceso de consenso para la definición de estándares abiertos e interoperables dentro de los Sistemas de Información Geográfica y de la World Wide Web, buscando acuerdos que posibiliten la interoperación de sistemas de geoprocésamiento y facilitar el intercambio de la información geográfica.

La OGC publica documentos técnicos de estándares y especificaciones que detallan los formatos que deben utilizar los desarrolladores en sus productos o servicios. En ellas se definen vocabularios, sintaxis y comandos comunes para lograr la comunicación e interoperabilidad entre clientes y servidores de distintas plataformas, formatos, software, marcas comerciales. [LIB 15].

2.2.2. Lenguajes de Marcado.

Un lenguaje de marcado es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación.

2.2.2.1. XML (Extensible Markup Language).

Es un estándar abierto y libre desarrollado por el consorcio W3C propuesto en 1996; primera versión en 1998. Es un meta-lenguaje que nos permite definir lenguajes de marcado adecuados a usos determinados o documentos que contienen datos estructurados. XML sirve como lenguaje común para el intercambio de información estructurada entre diferentes plataformas. [WEB 80].

Estructura:

Un XML contiene datos que se autodefinen, exclusivamente.



Figura 20: Estructura de un XML.

Fuente: [WEB 80].

Elementos de un XML

- Símbolos especiales: `<` `>` `''` `&` (`<`, `>`, `"`, `'`, `&`).
- Etiquetas (tags) y elementos: `<p>nombre</p>`
- Atributos (Attributes): `<elemento atributo="texto.otro="texto">`
- Elementos vacíos: `<nada></nada>` o `<nada />`

Tipo de Documentos XML

1. Documento Válido

Un documento XML es válido cuando está definido por un “Document Type Definition” (DTD), que es la gramática del documento que define qué tipos de elementos, atributos y entidades podría haber en el documento, además define el orden y ocurrencia de elementos.

2. Documento XML Bien-Formado

Un documento bien formado no tiene que adherirse a un DTD, pero debe seguir las siguientes reglas:

- Todo elemento debe tener una etiqueta de apertura y otra de cierre.
- Sólo un elemento raíz.
- Los valores de los atributos, siempre deben estar encerrados entre comillas simples o dobles.
- Es sensible al tipo de letra utilizado, es decir trata a las mayúsculas y minúsculas como símbolos diferentes.

2.2.2.2. HTML

HTML, siglas de HyperText Markup Language (lenguaje de marcado de hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de etiquetas, rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML [WEB 10].

Marcado HTML

Consta de varios componentes como: elementos y sus atributos, tipos de data, y la declaración de tipo de documento.

- **Elementos:** Son la estructura básica de HTML. Los elementos tienen dos propiedades básicas: atributos y contenido. Un elemento generalmente tiene una etiqueta de inicio (ejemplo: <nombre-de-elemento>) y una etiqueta de cierre (ejemplo: </nombre-de-elemento>). Los atributos del elemento están contenidos en la etiqueta de inicio y el contenido está ubicado entre las dos etiquetas (ejemplo:<nombre-de-elemento atributo=”valor”>Contenido</nombre-de-elemento>).

Algunos elementos, tales como
, no tienen contenido ni llevan una etiqueta de cierre.

- **Atributos:** La mayoría de los atributos de un elemento son pares nombre-valor, separados por un signo de igual “=” y escritos en la etiqueta de comienzo de un elemento, después del

nombre de éste. El valor puede estar rodeado por comillas dobles o simples, aunque ciertos tipos de valores pueden estar sin comillas en HTML (pero no en XHTML).



Figura 21: Estructura de un HTML.
Fuente: [WEB 10].

2.2.2.3. GML (Geography Markup Language)

Es una definición XML para expresar y comunicar features geográficas. GML constituye por tanto un lenguaje de modelado para sistemas geográficos, así como un formato de intercambio abierto para transacciones de información geográfica a través del internet. [WEB 81].

GML contiene un rico conjunto de primitivas que se utilizan para construir esquemas específicos de aplicación o lenguajes de aplicación. Estas primitivas incluyen:

- Feature
- Geometry
- Coordinate Reference System
- Topology
- Time
- Dynamic feature
- Coverage (including geographic images)
- Unit of measure
- Directions
- Observations
- Map presentation styling rules

Estructura:

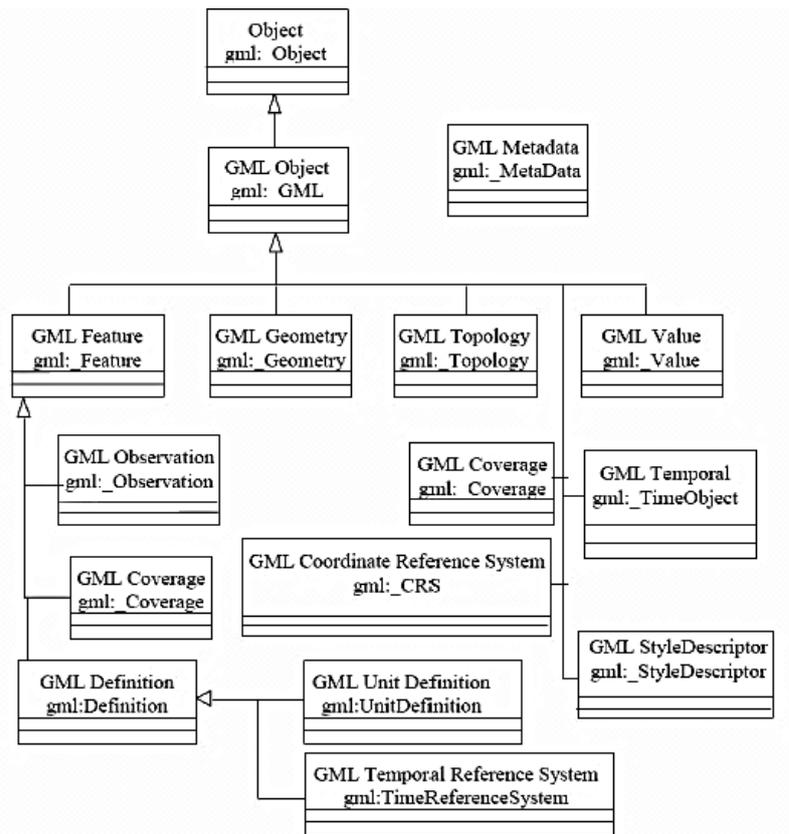


Figura 22: Diagrama UML que muestra la jerarquía de la clase GML.
Fuente: [WEB 81].

GML se diseñó a partir de la especificación abstracta producida por el grupo OpenGIS, ahora Open Geospatial Consortium, y de la serie de documentos ISO 19100.

GML fue aprobado formalmente como estándar OGC en el año 2001, se convirtió en estándar ISO en el año 2007. GML 3.2.16 es la versión más reciente del estándar conjunto OGC-ISO y actualmente se están generando las versiones 3.2.2 and 4.0.

2.2.2.4. KML (Keyhole Markup Language).

KML es una gramática XML, para transportar y visualizar representaciones de datos geográficos, basados en el internet, para el despliegue en el Navegador Earth, como un globo virtual 3D y navegadores de tejido o aplicaciones móviles en 2D.

KML fue desarrollado para el uso con Google Earht, y fue creado por Keyhole In, como una arquitectura para Google en el 2004, la versión 2.2 es adoptada como estándar internacional de la OGC.

El archivo de KML especifica un conjunto de características: las marcas del lugar, imágenes, los polígonos, modelos 3D, las descripciones textuales, etc..

KML es complementario a la mayoría de las normas de OGC existentes importantes incluso GML, WFS y WMS. Actualmente, KML 2.2 utiliza ciertos elementos de la geometría derivados de GML 2.1.2.

Ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Document>
    <Placemark>
      <name>New York City</name>
      <description>New York City</description>
      <Point>
        <coordinates>-74.006393,40.714172,0</coordinates>
      </Point>
    </Placemark>
  </Document>
</kml>
```

2.2.2.5. SVG (Scalable Vector Graphics).

Los Gráficos Vectoriales Escalables (SVG), es una especificación para describir gráficos vectoriales bidimensionales, tanto estáticos como animados, en formato XML.

SVG 1.1 se convirtió en una recomendación del W3C en septiembre del 2001, por lo que ya ha sido incluido de forma nativa en navegadores web del W3C.

El SVG permite tres tipos de objetos gráficos:

- Formas gráficas de vectores (camino consistentes en rectas y curvas, y áreas limitadas por ellos).
- Imágenes de mapa de bits /digitales
- Texto

Además SVG permite un juego amplio de manejadores de eventos, como “onMouseOver” y “onClick”, pueden ser asignados a cualquier objeto SVG. Debido a su compatibilidad y relación con otras normas Web.

2.2.2.6. GeoRSS

Es un conjunto de estándares para representar información geográfica mediante el uso de capas y está construido dentro de la familia de estándares RSS.

En las GeoRSS, el contenido consiste en puntos de interés georreferenciados y otras anotaciones y las fuentes se diseñan para generar mapas.

- GeoRSS-Simple: Es un formato muy ligero, este soporta las geometrías básicas como: point, line, box, poligon, circle, etc.
- GeoRSS-GML: es un GML, soporta grandes rasgos de elementos geográficos y sistemas de referencias.

GeoRSS puede utilizarse como RSS 1.0,2.0 y Atom, el último estándar IETF.

Ejemplo:

GeoRSS Simple usando Atom.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
xmlns:georss="http://www.georss.org/georss">
  <title>Earthquakes</title>
  <subtitle>International earthquake observation labs</subtitle>
  <link href="http://example.org"/>
  <updated>2005-12-13T18:30:02Z</updated>
  <author>
    <name>Dr. Thaddeus Remor</name>
    <email>tremor@quakelab.edu</email>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry>
    <title>M 3.2, Mona Passage</title>
    <link href="http://example.org/2005/09/09/atom01"/>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-08-17T07:02:32Z</updated>
    <summary>We just had a big one.</summary>
    <georss:point>45.256 -71.92</georss:point>
  </entry>
</feed>
```

2.2.3. Formatos Vectoriales.

2.2.3.1. GeoJSON.

Es un formato abierto, para codificación de una variedad de estructuras de datos geográficos. Se base en JSON.

GeoJSON permite que los datos sean almacenados de un forma legible para los humanos, es mucho más compacto que XML. Los tipos de datos soportados son: puntos, líneas, polígonos, multipuntos, multilínea, multipoligonos, colección de geometría y cuadros de limite, que se almacenan junto con la información de características y atributos. Además permite la especificación de un sistema de coordenadas geográficas.

Ejemplo:

```

{
  "type": "Feature",
  "id": "OpenLayers.Feature.Vector_314",
  "properties": {},
  "geometry": {
    "type": "Point",
    "coordinates": [97.03125, 39.7265625]
  },
  "crs": {
    "type": "OGC",
    "properties": {
      "urn": "urn:ogc:def:crs:OGC:1.3:CRS84"
    }
  }
}

```

2.2.3.2. WKT (Well Known Text).

Es una codificación diseñada para describir objetos espaciales expresados de forma vectorial. Los tipos de objetos soportados son: puntos, multipuntos, líneas, multilíneas, polígonos, multipolígonos y colección de geometrías. Su especificación ha sido promovida por la OGC. En WKT también se pueden especificar sistemas de referencia espacial describiendo el datum, geoide, sistema de coordenadas y proyección de los objetos espaciales.

Ejemplo:

Punto: POINT(30 50)

Línea: LINESTRING(1 1, 5 5, 10 10, 20 20)

Multilínea: LINESTRING((1 1, 5 5, 10 10, 20 20),(20 30, 10 15, 40 5))

Polígono simple: POLYGON ((0 0, 10 0, 10 10, 0 0))

Varios polígono en una sola geometría (multipolígono): POLYGON ((0 0, 10 0, 10 10, 0 10, 0 0),(20 20, 20 40, 40 40, 40 20, 20 20))

Geometrías de distinto tipo en un sólo elemento: GEOMETRYCOLLECTION(POINT(4 6),LINESTRING(4 6,7 10))

Punto vacío: POINT EMPTY

Multipolígono vacío: MULTIPOLYGON EMPTY

2.2.4. OGC Web Services (OWS).

La OWS son especificaciones de estándares internacionales por la OGC para la utilización de la información geográfica a través de la Web. Es el nombre genérico con el que se agrupan todos los estándares OGC Web Services (WMS, WFS, WCS, CSW, WPS, etc.). [LIB 15].

Todos los OWS se encuentran definidos en base a estándares de Internet: HTTP, URL, tipos MIME y XML; además todos poseen la operación GetCapabilities que devuelve la metadata del servicio.

La figura inferior proporciona un esquema general de arquitectura para los OWS. Este esquema identifica las clases genéricas de servicios que participan en diversas actividades de geoprocésamiento y localización.

Web Services Framework Of OGC Geoprocessing Standards

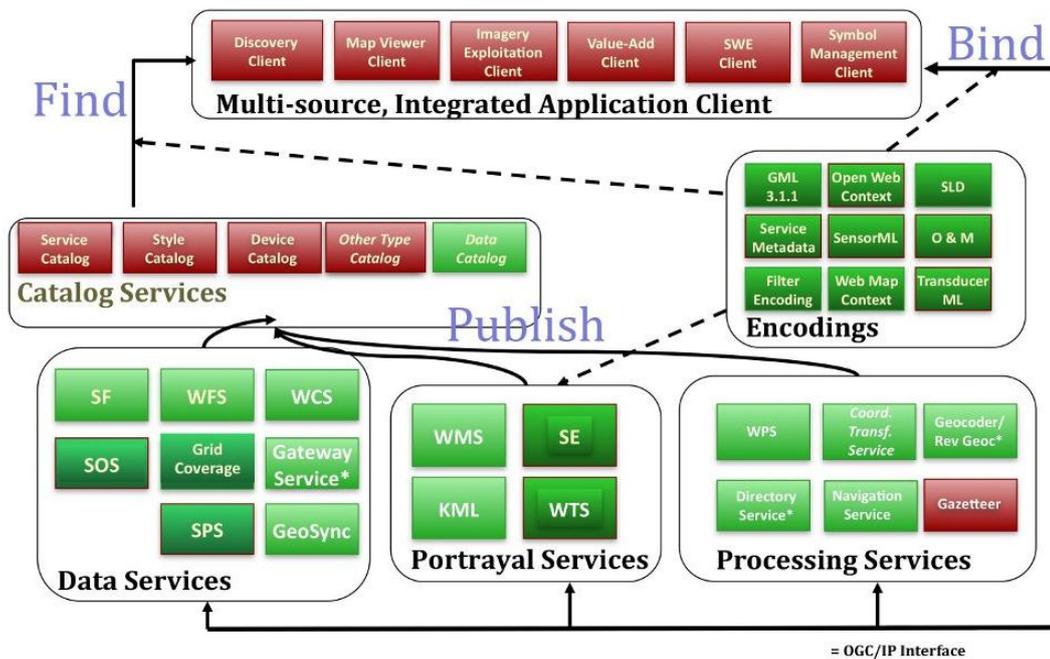


Figura 23: Estándares OGC.
Fuente: [WEB 74].

2.2.4.1. WMS (Web Map Services).

El servicio Web Map Services interactúa con los clientes a través de peticiones HTTP, produce mapas referenciados espacialmente de forma dinámica, a partir de información geográfica. Estos mapas son enviados a los clientes en forma de imagen digital.

Los mapas producidos por WMS se generan normalmente en un formato de imagen como PNG, GIF o JPEG, se utilizan los formatos transparentes para sobreponer capas de diferentes servidores, también se puede generar archivos TIFF que necesitan software adicional para su visualización y ocasionalmente se generan como gráficos vectoriales en formato SVG (Scalable Vector Graphics) o WebCGM (Web Computer Graphics Metafile). [LIB 9].

Funcionalidades:

- Obtener información del servicio y de la información geográfica que éste contiene.
- Solicitar mapas en diferentes formatos y leyendas de los objetos del mapa.
- Gestionar capas de información geográfica y estilos de visualización.

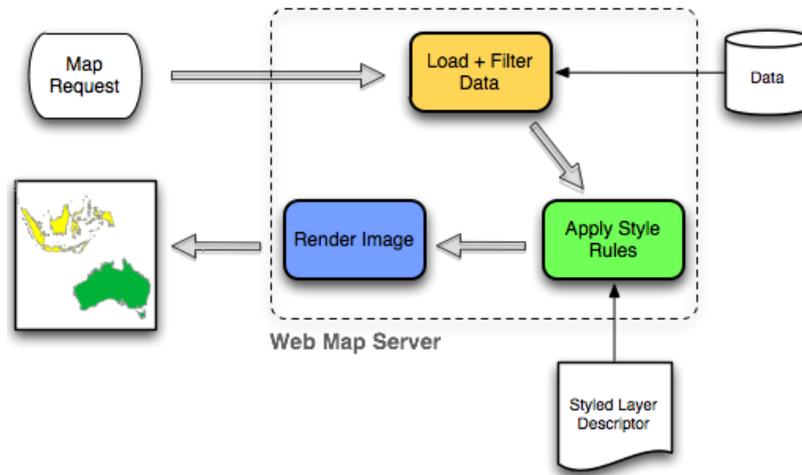


Figura 24: OGC WMS.
Fuente: [WEB 74].

Operaciones:

Las tres operaciones definidas por el estándar WMS son:

1. GetCapabilities (Obligatoria)

Permite obtener metadatos acerca del servicio web implementado, información de las funcionalidades soportadas por el servicio, información específica sobre las capas de información geográfica disponibles, y valores aceptables de los parámetros de petición. La información es devuelta al cliente codificada en un documento XML.

Parámetro Request	Obligatorio	Descripción
VERSION=versión	No	Versión del Servicio
SERVICE=WMS	Si	Tipo de Servicio
REQUEST=GetCapabilities	Si	Nombre de la opción Request.
FORMAT=MIME_type	No	Formato de salida del metadata del servicio.
UPDATESEQUENCE=string	No	Números o letras de secuencia para control de cache.

Tabla 4: Parámetros de una petición url GetCapabilities.
Fuente: [LIB 9].

2. GetMap (Obligatoria)

Esta funcionalidad genera un mapa a partir de un conjunto de entidades geográficas y de una descripción de las capas que lo componen junto con el estilo de visualización correspondiente a cada una de ellas. A través de esta operación, las aplicaciones cliente pueden obtener mapas, codificados en alguno de los formatos existentes para representar información gráfica (vectorial y raster), de forma que pueden ser visualizados directamente.

Parámetro Request	Obligatorio	Descripción
VERSION=1.3.0	Si	Versión del Servicio.
REQUEST=GetMap	Si	Nombre de la Opción Request.
LAYERS=layer_list	Si	Lista de las capas separadas por comas.
STYLES=style_list	Si	Nombre de un estilo por capa a solicitar, separados por coma.
CRS=namespace:identifier	Si	Sistema de Referencia
BBOX=minx,miny,maxx,maxy	Si	Rectángulo que define la extensión geográfica del mapa en el espacio especificado por el CRS.
WIDTH=output_width	Si	Ancho en pixeles del mapa a dibujar.
HEIGHT=output_height	Si	Alto en pixeles del mapa a dibujar.
FORMAT=output_format	Si	Formato de salida del mapa.
TRANSPARENT=TRUE FALSE	No	Fondo transparente del mapa (Por defecto su valor es FALSE)
BGCOLOR=color_value	No	Valor en Hexadecimal del color del fondo a aplicar (Por defecto su valor es 0xFFFFFFFF)
EXCEPTIONS=exception_format	No	Formato para reportar excepciones cuando existan por WMS. (Por defecto su valor es XML).
TIME=time	No	Valor del tiempo cuando la capa fue solicitada.
ELEVATION=elevation	No	Evaluación de la capa a ser requerida.
Other sample dimension(s)	No	Valor de otras dimensiones.

Tabla 5: Parámetros de una petición url GetMap.

Fuente: [LIB 9].

3. GetFeatureInfo (Opcional)

Es una operación opcional soportada únicamente en las capas en las que el atributo queryable está en 1 (verdadero), esta operación provee al cliente con información del elemento geográfico que está en el mapa, el cliente escoge un punto en el mapa del cual quiere obtener información, las capas que quiere investigar y el formato de la información devuelta.

Parámetro Request	Obligatorio	Descripción
VERSION=1.3.0	Si	Versión del Servicio
REQUEST=GetFeatureInfo	Si	Nombre de la opción Request
map request part	Si	Copia parcial de los parámetros de demanda que generaron el mapa para la información solicitada.
QUERY_LAYERS=layer_list	Si	Lista de una o más capas solicitadas, separadas por comas.
INFO_FORMAT=output_format	Si	Formato de respuesta de la información (MIME Type).
FEATURE_COUNT=number	No	Numero de objetos gráficos que tengan coincidencia en el punto a consultar, de los cuales retornará la información (Su valor por defecto es 1).
I=pixel_column	Si	Coordenada i en pixeles del objeto geográfico en el Mapa.
J=pixel_row	Si	Coordenada j en pixeles del objeto geográfico en el Mapa.
EXCEPTIONS=exception_format	No	El formato si ocurre una excepción (Por defecto es XML).

Tabla 6: Parámetros de una petición url GetFeatureInfo.
Fuente: [LIB 9].

2.2.4.2. WFS (Web Features Services).

WFS provee una interface que permite al cliente obtener y actualizar datos georeferenciados mediante el formato GML de múltiples servidores WFS usando el protocolo HTTP. Permite acceder a datos vectoriales y consultar todos los atributos de un fenómeno (feature) geográfico.

A través de estas interfaces se puede combinar, usar y manipular la información que está detrás de los mapas mediante la invocación de operaciones WFS sobre los elementos geográficos, las funciones en la especificación 1.1.0 permitidas son:

- Consultar elementos geográficos basados en parámetros espaciales y no espaciales.
- Crear una nueva instancia del elemento espacial.
- Obtener una descripción de las propiedades del elemento espacial.
- Borrar la instancia de un elemento espacial.
- Actualizar una instancia de un elemento espacial.
- Bloquear una instancia del elemento espacial. [LIB 10].

Requerimientos:

- Las interfaces deben estar definidas en XML.
- GML debe ser utilizado para expresar los elementos espaciales en la interfaz.
- El lenguaje de filtro debe estar definido en XML y ser derivado de CQL (Common Query Language).
- El almacenamiento de datos usado para guardar los elementos geográficos debería ser transparente a las aplicaciones cliente.
- Utilización del subconjunto XPath para representación de propiedades. [LIB 10].

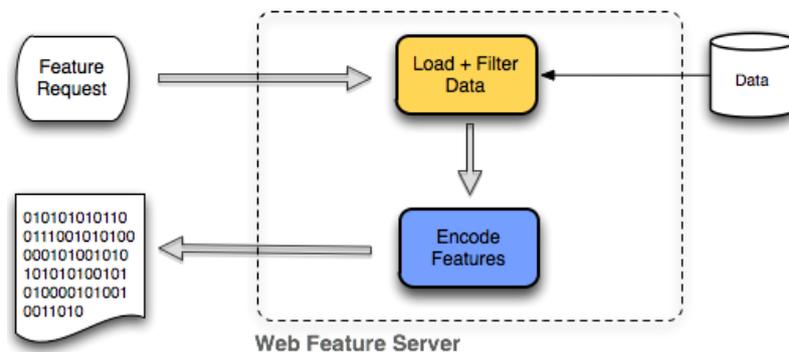


Figura 25: OGC WFS.
Fuente: [WEB 74].

Operaciones:

1. GetCapabilities (Obligatoria).

Devuelve un fichero XML con las características y definición del servicio y de los tipos de elementos geográficos que puede ofrecer y que operaciones soporta cada uno de ellos. [LIB 10].

2. DescribeFeatureType (Obligatoria).

Devuelve la estructura (campos y atributos) de cualquier tipo de elemento espacial que proporcione, en un determinado lenguaje [LIB 10], por ejemplo GML. Los parámetros son:

- TypeName: Nombre del tipo de entidad.
- OutputFormat: Formato del esquema, por defecto es GML.

3. GetFeature (Obligatoria).

Devuelve una instancia del elemento espacial, en formato GML. El cliente puede especificar que propiedades del elemento espacial, y restringirlas mediante una consulta, sea espacial o no espacial [LIB 10]. Los parámetros son:

- TypeName: Lista con los nombres de los tipos de entidades que se consultan. Similar al FROM de un SELECT, con un JOIN entre los tipos.

- **PropertyName:** Propiedad de un tipo de entidad que debe incluirse en la respuesta. Las propiedades obligatorias siempre se incluyen para que el documento se valide. Similar a los atributos de un SELECT de SQL.
- **Filter:** Permite imponer restricciones sobre las entidades que se incluyen en la respuesta, por ejemplo, condiciones geográficas (WITHIN, OVERLAP, etc.). Similar al WHERE de SQL.
- **OutputFormat:** Formato del esquema de descripción de la respuesta. Los tipos validos se obtienen mediante GetCapabilities.
- **ResultType:** Indica si debe devolver las entidades o la cantidad de las mismas. Similar al COUNT de SQL.
- **MaxFeatures:** Limita el número de entidades devueltas. Similar al LIMIT de SQL.

4. GetGmlObject (Opcional).

Un servicio WFS debería poder devolver la instancia de un elemento espacial mediante XLinks que contienen sus XML IDs [LIB 10]. Los parámetros son:

- **GmlObjectId:** Id del Objeto.
- **OutputFormat:** Formato de respuesta.
- **TraverseXlinkDepth:** Profundidad de búsqueda de los elementos referenciados mediante Xlinks.
- **TraverseXlinkExpiry:** Tiempo que debe esperar el servidor para obtener respuesta de otros servidores en consultas recursivas.

5. Transaction (Opcional).

El servicio WFS puede procesar las peticiones de transacción. Una petición de transacción está compuesta de una serie de operaciones para modificar el elemento espacial, estas pueden ser: create, update y delete. El servidor debe hacer un Commit al finalizar, o un Rollback si ocurrió algún error [LIB 10].

- **Insert:** Inserta elementos correspondientes al esquema utilizado, con la geometría expresada en GML.

El atributo idgen determina como se genera los gml:id para las nuevas entidades.

- **Update:** Modifica propiedades de elementos existentes.

Parámetros:

- **Property:** Permite indicar el nombre de la propiedad y el valor de reemplazo.
- **TypeName:** Permite indicar el tipo de entidad.
- **Filter:** Permite restringir las entidades afectadas por la modificación.
- **Delete:** Elimina elementos existentes.

Parámetros:

- TypeName: Permite indicar el tipo de entidad.
- Filter: Permite restringir las entidades afectadas por la modificación.

Como respuesta devuelve como mínimo el número de entidades creadas, modificadas y eliminadas.

6. LockFeature (Opcional).

El servicio WFS debe procesar una petición de bloqueo en una o más instancias de un tipo de elemento espacial mientras dure la transacción [LIB 10].

Tipos de WFS

Basando en las descripciones de las operaciones, tres tipos de web features services pueden ser definidas, en:

- **Basic WFS**

Este debe implementar las operaciones GetCapabilities, DescribeFeatureType y GetFeature. Se considera un Web Feature Services de solo lectura.

- **Xlink WFS**

Un Xlink WFS debe soportar todas las operaciones de Basic WFS y además la operación GetGmlObject.

- **Transaction WFS**

Transaction WFS debe soportar todas las operaciones de Basic WFS y además la operación Transaction, opcionalmente puede implementar las operaciones GetGmlObject y/o LockFeature.

2.2.4.3. WCS (Web Coverage Service).

Define la interfaz y operaciones que permiten el acceso interoperable a “coverages” geoespaciales, el término se refiere a contenido como: imágenes satelitales, fotografías aéreas, modelos digitales de elevación (DEM), y cualquier fenómeno espacial representado por valores en cada punto de medida.

WCS provee datos con su semántica original en lugar de solo imágenes, permitiendo su análisis e interpretación [LIB 13].

Operaciones:

1. GetCapabilities

Retorna un documento XML con la descripción del servicio y una breve descripción de los coverages que los clientes requieran.

2. DescribeCoverage

Devuelve un documento XML con la descripción de uno o más coverages proporcionados por el servidor.

3. GetCoverage

Retorna un coverage (valores de propiedades de un conjunto de ubicaciones geográficas), en el formato well-known coverage.

2.3. Base de Datos.

2.3.1. Base de Datos Relacional.

2.3.1.1. PostgreSQL.

Es un sistema de gestión de base de datos objeto-relacional, distribuido bajo la licencia BSD. Es el sistema de gestión de base de datos de código abierto más potente del mercado.

PostgreSQL utiliza un modelo cliente servidor y usa multiprocesos para garantizar la estabilidad del sistema, un fallo en uno de los procesos no afectará el resto y el sistema continuara funcionando.

PostgreSQL fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual. Es un sistema objeto-relacional, ya que incluye características de orientación a objetos, como herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. PostgreSQL se caracteriza por ser un sistema estable, de alto rendimiento, gran flexibilidad ya que funciona en la mayoría de los sistemas operativos.

Características:

- Garantiza el concepto de transacción (ACID).
- La implementación de SQL se adhiere al estándar ANSI-SQL 92/99.
- Tiene soporte completo para subqueries (incluye subselects en la cláusula FROM).
- Soporte para Replicación.
- Triggers, procedual lenguajes (PL/PgSQL,PL/R).
- Incluye soporte para el manejo de integridad de datos. Primary key, foreign key con acciones referenciales de cascada por update/delete, check constraints, unique constraint, not null constraint.
- Permite retornar resultados parciales de consultas con LIMIT/OFFSET.
- Proporciona acceso multi-usuario para potencialmente enormes cantidades de datos.
- Lenguaje de definición de datos DDL.

2.3.2. Base de Datos Espacial.

Las bases de datos espaciales, consisten en BBDD que en uno de sus campos contiene información espacial. De tal forma que tenemos por un lado la potencia del software GIS y por otro lado tendremos toda la potencia de las Bases de Datos.

Esta asociación, es muy beneficiosa ya que permite realizar consultas muy complejas y poder así extraer informaciones muy valiosas con una o dos sentencias sql.

2.3.2.1. PostGIS.

Es un módulo que añade soporte de objetos geográficos a la base de datos objeto-relacional PostgreSQL, convirtiéndola en una base de datos espacial para su utilización en GIS. Postgis ha sido desarrollado por Refraction Research como un proyecto open source y se encuentra publicado bajo la licencia GNU.

Su implementación está basada en geometrías e índices ligeros optimizados, para reducir el uso de disco y memoria. Utilizando geometrías ligeras ayuda al servidor a incrementar la cantidad de datos migrados desde el almacenamiento físico en el disco hacia la RAM.

Postgis puede ser construido en PostgreSQL 7.2 o superior.

Características.

- Certificado en 2006 por el Open Geospatial Consortium (OGC), lo cual garantiza la interoperabilidad con otros sistemas.
- Sigue el OpenGIS “Simple Features Specification for SQL” y se ha certificado como adherente al perfil “Types and Functions”.
- Permite tipos de datos geométricos para: points, linestrings, polygons, multipoints, multilinestrings, multipolygons y geometriccollection.
- Operadores espaciales para determinar las medidas geoespaciales como el área, distancia, longitud y perímetro.
- Operadores espaciales para determinar el conjunto de operaciones geoespaciales, como la unión, diferencia, diferencia simétrica y buffer.
- Índices espaciales de árboles R para rápidas consultas espaciales.

2.4. Servidores de Aplicaciones.

2.4.1. HTTP Apache.

Es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation [WEB 1].

Características:

- Altamente configurables.
- Bases de datos de autenticación y negociado de contenido.
- Modular.
- Código abierto.
- Multi-plataforma.
- Extensible.
- Fácil conseguir ayuda/suporte.
- La licencia Apache es una descendiente de la licencias BSD, no es GPL. Esta licencia permite hacer lo que quiera con el código fuente siempre y cuando se reconozca su trabajo.

2.4.2. Tomcat.

Funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems. Funciona en cualquier sistema operativo que disponga de la máquina virtual Java [WEB 22].

Características:

Tomcat 7.x

- Implementado de Servlet 3.0 JSP 2.2 y EL 2.2
- Mejoras para detectar y prevenir “fugas de memoria” en las aplicaciones web
- Limpieza interna de código
- Soporte para la inclusión de contenidos externos directamente en una aplicación web

2.5. Lenguajes de Programación.

2.5.1. CSS.

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. Es un lenguaje usado 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. La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML.

El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para navegadores [WEB 4].

Sintaxis:

CSS tiene una sintaxis muy sencilla, que usa unas cuantas palabras claves tomadas del inglés para especificar los nombres de sus selectores, propiedades y atributos.

Una hoja de estilos CSS consiste en una serie de reglas. Cada regla consiste en uno o más selectores y un bloque de estilos con los estilos a aplicar para los elementos del documento que cumplan con el selector que les precede. Cada bloque de estilos se define entre llaves, y está formado por una o varias declaraciones de estilo con el formato propiedad: valor; [WEB 5].

2.5.2. PHP.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor. Es un lenguaje de programación interpretado (Lenguaje de alto rendimiento), diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la 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+ [WEB 17].

Características:

- Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.
- Puede ser desplegado en la mayoría de los servidores web.
- La curva de aprendizaje es muy corta.
- Puede ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows.

- Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.
- El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su sitio web oficial.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).
- El programador puede aplicar cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable.

2.5.3. JAVASCRIPT.

Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a prototipos, imperativo, débilmente tipado y dinámico, desarrollado para incrementar las funcionalidades del lenguaje HTML [WEB 12].

Características:

- Se utiliza principalmente en el lado del cliente, 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.
- 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 Document Object Model (DOM).
- No es necesario declarar los tipos de variables que van a utilizarse.
- Maneja objetos dentro de nuestra página Web y sobre ese objeto podemos definir diferentes eventos. Dichos objetos facilitan la programación de páginas interactivas [WEB 11].
- Responde a eventos en tiempo real. Con esto podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor una página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario, etc.

2.5.4. PHP MAPSCRIPT.

PHP mapscript es un módulo para ampliar funcionalidades a PHP de manera que trabaje con MapServer. Este módulo le permitirá utilizar el lenguaje de scripting de gran alcance de PHP para crear y modificar de forma dinámica las imágenes de mapa de MapServer [WEB 15].

2.5.5. MAPFILE.

Es el archivo principal de configuración de MapServer. Es un archivo de texto, con extensión “.map”, en el que se incluye una serie de parámetros que definen las capas disponibles en el servicio, el estilo con que se representarán, su simbología, en que formato se generará la imagen, el sistema de referencia, etc [WEB 13].

Características:

- El archivo .map consta de varias secciones. Cada sección se inicia con el nombre de la sección y termina con la palabra END. El contenido de las secciones consiste en la definición de determinados parámetros del tipo atributo - valor.
- Para realizar comentarios debe introducirse delante de la línea el símbolo #.
- El orden de los parámetros no es sensitivo.
- Los colores son manejados mediante los tres canales RGB (rojo – verde – azul).
- La sección principal es el objeto map, la cual anida a otras secciones, como se observa en la siguiente figura.

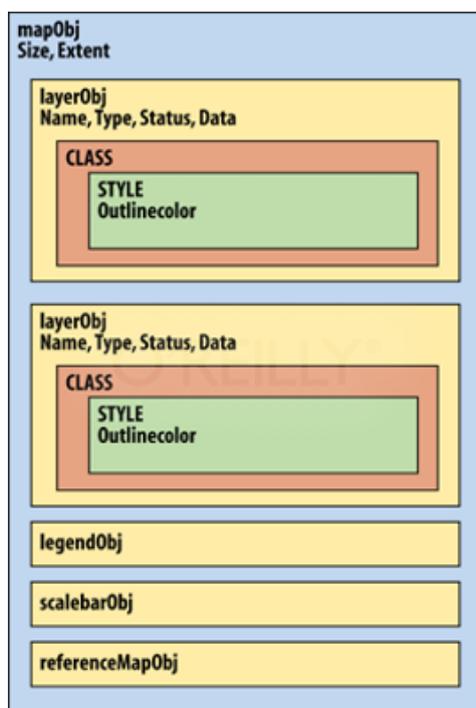


Figura 26: Estructura MapFile.
Fuente: [WEB 14].

2.6. Frameworks.

Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, con base a la cual otro proyecto de software puede ser más fácilmente organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto [WEB 16].

2.6.1. Symfony 1.16.

Es un framework PHP que facilita y optimiza el desarrollo de las aplicaciones web. Symfony se encarga de todos los aspectos comunes y aburridos de las aplicaciones web, dejando que el programador se dedique a aportar valor desarrollando las características únicas de cada proyecto. Symfony separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web [WEB 18].

Características:

- Fácil de instalar y configurar en sistemas Windows, Mac y Linux.
- Compatible solamente con PHP 5, para asegurar el mayor rendimiento y acceso a las características más avanzadas de PHP.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador solo debe configurar aquello que no es convencional.
- Preparado para aplicaciones empresariales, ya que se puede adaptar con facilidad a las políticas y arquitecturas propias de cada empresa u organización.
- Flexible hasta cualquier límite y extensible mediante un completo mecanismo de plugins.
- Publicado bajo licencia MIT de software libre y apoyado por una empresa comprometida con su desarrollo.
- Traducido a más de 40 idiomas y fácilmente traducible a cualquier otro idioma.
- El manejo de cache reduce el uso de banda ancha y la carga del servidor.
- La facilidad de soportar autenticación y credenciales facilita la creación de áreas restringidas y manejo de seguridad de los usuarios.
- El enrutamiento y las URLs inteligentes hacen amigable las direcciones de las páginas de la aplicación.
- Independiente del sistema gestor de bases de datos. Su capa de abstracción y el uso de Doctrine, permiten cambiar con facilidad de SGBD en cualquier fase del proyecto.

- Aunque utiliza MVC (Modelo Vista Controlador), tiene su propia forma de trabajo en este punto, con variantes del MVC clásico como la capa de abstracción de base de datos, el controlador frontal y las acciones.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Permite un mantenimiento muy sencillo.
- Está compuesto por una potente línea de comandos que facilitan generación de código, lo cual contribuye a ahorrar tiempo de trabajo.

2.6.2. Doctrine.

Doctrine es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un SGBD [WEB 6].

Características:

- Se necesita un bajo nivel de configuración para empezar un proyecto.
- Puede generar clases a partir de una base de datos existente y después el programador puede especificar relaciones y añadir funcionalidad extra a las clases autogeneradas.
- No es necesario generar o mantener complejos esquemas XML de base de datos como en otros frameworks.
- Otra característica importante de Doctrine es la posibilidad de escribir consultas de base de datos utilizando un dialecto de SQL denominado DQL (Doctrine Query Language) que está inspirado en Hibernate (Java).
- Soporte para datos jerárquicos.
- Soporte para hooks (métodos que pueden validar o modificar las escrituras y lecturas de la base de datos) y eventos para manejar la lógica de negocio relacionada.
- Herencia.
- Un framework de caché que utiliza diversos motores como memcached, SQLite o APC.
- Transacciones ACID.
- Diversos comportamientos del modelo (conjuntos anidados, internacionalización, log, índice de búsqueda);

2.6.3. Ext JS.

Es una biblioteca de JavaScript para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX, DHTML y DOM. Fue desarrollada por Sencha [WEB 8].

Características:

- Dispone de un conjunto de componentes (widgets) para incluir dentro de una aplicación web, como:
 - Cuadros y áreas de texto.
 - Campos para fechas.
 - Campos numéricos.
 - Combos.
 - Radiobuttons y checkboxes.
 - Editor HTML.
 - Árbol de datos.
 - Pestañas.
 - Barra de herramientas.
 - Menús al estilo de Windows.
 - Paneles divisibles en secciones.
 - Sliders.
- Elementos de datos (con modos de sólo lectura, datos ordenables, columnas que se pueden bloquear y arrastrar, etc.).
- Varios de sus componentes están capacitados para comunicarse con el servidor usando AJAX.
- También contiene numerosas funcionalidades que permiten añadir interactividad a las páginas HTML, como:
 - Cuadros de diálogo.
 - Quicktips para mostrar mensajes de validación e información sobre campos individuales.

2.7. Metodología de Programación Extrema XP.

2.7.1. Definición.

Es una metodología ágil basada en cuatro principios: simplicidad, comunicación, retroalimentación y valor. Además orientada por pruebas y refactorización; se diseñan las pruebas antes de programar la funcionalidad, el programador crea sus propios test de unidad. Xp busca la satisfacción del cliente tratando de mantener durante todo el tiempo su confianza en el producto.

2.7.2. Actividades.

2.7.2.1. Codificar.

Es necesario codificar y plasmar nuestras ideas a través del código.

2.7.2.2. Hacer Prueba.

Las pruebas dan la oportunidad de saber si lo implementado es lo que en realidad se tenía en mente. Las pruebas nos indican que nuestro trabajo funciona, cuando no podemos pensar en ninguna prueba que pudiese originar un fallo en nuestro sistema, entonces habremos acabado por completo.

2.7.2.3. Escuchar.

Si vamos hacer pruebas tenemos que preguntar si lo obtenido es lo deseado, y tenemos que preguntar a quién necesita la información. Tenemos que escuchar a nuestros clientes cuáles son los problemas de su negocio, debemos de tener una escucha activa explicando lo que es fácil y difícil de obtener, y la realimentación entre ambos nos ayudan a todos entender los problemas.

2.7.2.4. Diseñar.

El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben ser sencillos, si alguna parte del sistema es de desarrollo complejo, lo apropiado es dividirla en varias. Si hay fallos en el diseño o malos diseños, estos deben ser corregidos cuanto antes.

2.7.3. Ciclo de Vida.

El ciclo de vida de XP se enfatiza en el carácter iterativo e incremental del desarrollo, una iteración de desarrollo es un periodo de tiempo en el que se realiza un conjunto de funcionalidades determinadas que en el caso de XP corresponden a un conjunto de historias de usuario.

Las iteraciones son relativamente cortas ya que se piensa que entre más rápido se le entregue desarrollos al cliente, más retroalimentación se va a obtener y esto va a representar una mejor calidad del producto a largo plazo. Existe una fase de análisis inicial orientada a programar las iteraciones de desarrollo y cada iteración incluye diseño, codificación y pruebas [WEB 23].

2.7.4. Fases.

2.7.4.1. Fase de Exploración.

Es la fase en la que se define el alcance del proyecto. En esta fase el cliente define lo que necesita mediante la redacción de sencillas Historias de Usuario. Los programadores estiman los tiempos de

desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias, y podrían variar cuando se analice más en detalle en cada iteración.

Esta fase dura típicamente un par de semanas, y el resultado es una visión general del sistema [WEB 23].

2.7.4.2. Fase de Planificación.

La fase de planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberían implementarse las historias de usuario, y, asociadas a éstas, las entregas. El resultado de esta fase es el plan de Entregas [WEB 23].

2.7.4.3. Fase de Iteración.

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración termina sin errores es una medida clara de avance [WEB 23].

2.7.4.4. Fase de Producción.

Requiere prueba y comprobación extra del funcionamiento del sistema antes de que éste se pueda liberar al cliente. En esta fase, los nuevos cambios pueden todavía ser encontrados y debe tomarse la decisión de si se incluye o no en el reléase actual [WEB 23].

2.7.5. Artefactos.

2.7.5.1. Historias de Usuario.

Representa una breve descripción del comportamiento del sistema, emplea terminología del cliente sin lenguaje técnico, se realiza una por cada característica principal del sistema, se emplean para hacer estimaciones de tiempo y para el plan de entregas, reemplazan un gran documento de requisitos y presiden la creación de las pruebas de aceptación.

Difieren de los casos de uso porque son escritos por el cliente, no por los programadores, empleando terminología del cliente [WEB 24].

2.7.5.2. Tareas de Ingeniería.

Las tareas de ingeniería se usan para describir las tareas que se realizan sobre el proyecto. Las tareas pueden ser: desarrollo, corrección, mejora, etc.

Estas tareas tienen relación con una historia de usuario; se especifica la fecha de inicio y fin de la tarea, se nombra al programador responsable de cumplirla y describirnos que se trata de hacer en la tarea [WEB 24].

2.7.5.3. Casos de Prueba de Aceptación.

Permite confirmar que la historia ha sido implementada correctamente [WEB 24].

2.7.5.4. Tarjeta CRC.

Estas tarjetas se dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores. En ellas se expresa el diseño del sistema la sencillez de esta tarjeta hace que el diseño sea una tarea fácil [WEB 24].

Clase: Cualquier persona, cosa, evento, concepto, pantalla o reporte.

Responsabilidad: Son las cosas que conocen y las que realizan, sus atributos y métodos.

Colaboradores: Son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

2.8. Algunas Definiciones.

2.8.1. Servidor de Mapas.

Un servidor de mapas, más conocido por sus siglas IMS (Internet Map Server) provee mapas o cartografía a través del Internet, facilitando el proceso de publicación de Información Geográfica actualizada, en tiempo real y forma más barata a cualquier parte del mundo [WEB 65].

Características:

- Provee cartografía a través de la red tanto en modo vectorial como con imágenes.
- Los mapas son generados a partir de datos espaciales que se encuentran almacenados de forma local ó remota.
- Permiten al usuario la máxima interacción con la información geográfica. Por un lado el cliente accede a información en su formato original, de manera que es posible realizar consultas tan complejas como las que haría un GIS.
- Tienen la capacidad de integrar datos espaciales de diversas fuentes datos en una aplicación espacial.

- Contribuyen a facilitar de forma rápida y accesible datos espaciales a través de la web.

2.8.2. Clientes GIS Livianos.

Un cliente ligero para GIS es una aplicación en Internet con la que podemos visualizar y tratar la Información Geográfica a través de unas herramientas básicas de manipulación, navegación y análisis [WEB 3].

2.8.3. Clientes GIS Pesados.

Se denomina cliente pesado al programa “cliente” de una arquitectura cliente-servidor cuando la mayor carga de cómputo está desplazada hacia la computadora que ejecuta dicho programa. El programa cliente se califica como pesado cuando asume la mayor parte de las funcionalidades [WEB 2].

Características:

- Requiere la instalación o configuración a medida en el ordenador del usuario.
- La mayoría de los clientes pesados asumen todas las funcionalidades de la aplicación, dejando al programa servidor la gestión de transacciones contra una base de datos, es decir, el almacenamiento de información.
- Su riqueza en la interfaz de usuario, no está limitada por las características de un cliente universal, por ejemplo, un navegador web.
- Son capaces de acceder a todos los tipos de Geoinformación.
- Entregan entre un 60% a 80% de las posibles funciones de análisis de geoinformación existentes en la actualidad.

2.8.4. Norma ISO 19110: Metodología para la creación de catálogos de elementos.

La ISO 19110 proporciona un marco normativo para organizar y divulgar la clasificación de fenómenos del mundo real, asociados a una ubicación relativa a la tierra.

Especifica como la clasificación de fenómenos se organiza en un catálogo que se presenta a los usuarios, en la forma de un conjunto de datos geográficos, y define la metodología para catalogar diversos tipos de fenómenos.

Esta Norma Internacional es aplicable a la creación de catálogos de tipos de fenómenos que se representan en forma digital, aunque sus principios se pueden ampliar para la catalogación de otras formas de datos geográficos, como una base de datos espacial.



Capítulo III: Estudio de Herramientas GIS - Open Source

CONTENIDO:

1. Open Source.
2. Software Libre.
3. Servidores de Mapas.
4. Clientes GIS Livianos.
5. Clientes GIS Pesados.
6. Servidores de Cache Geoespacial.
7. Propuesta de software para el desarrollo de un Sistema GIS.

3. Capítulo III: Estudio de Herramientas GIS Open Source.

3.1. 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 compartir el código que a las cuestiones éticas y morales las cuales destacan en el llamado software Libre. [WEB 72].

3.2. Software Libre.

El software libre (en inglés free software) 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. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar distribuir, estudiar, modificar el software y distribuirlo modificado.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a software gratuito, ya que, conservando su carácter de libre, puede ser distribuido comercialmente. [WEB 71].

3.2.1. Libertades del software libre.

Un software es libre cuando garantiza las siguientes libertades.

Libertad	Descripción
0	La libertad de usar el programa, con cualquier propósito.
1	La libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades.
2	La libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo.
3	La libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

Tabla 7: Libertades del Software Libre.

Fuente: [WEB 71]

3.2.2. Ventajas del Software Libre.

- Bajo costo de adquisición: Se trata de un software económico ya que permite un ahorro de grandes cantidades en la adquisición de las licencias.
- Innovación tecnológica: esto se debe a que cada usuario puede aportar sus conocimientos y su experiencia y así decidir de manera conjunta hacia donde se debe dirigir la evolución y el desarrollo del software. Este es un gran avance en la tecnología mundial.
- Independencia del proveedor: al disponer del código fuente, se garantiza una independencia del proveedor que hace que cada empresa o particular pueda seguir contribuyendo al desarrollo y los servicios del software.
- Escrutinio público: esto hace que la corrección de errores y la mejora del producto se lleven a cabo de manera rápida y eficaz por cada uno de los usuarios que lleguen a utilizar el producto.
- Adaptación del software: esta cualidad resulta de gran utilidad para empresas e industrias específicas que necesitan un software personalizado para realizar un trabajo específico y con el software libre se puede realizar y con costes totales de operación (TCO) mucho más razonables.
- Lenguas: aunque el software se cree y salga al mercado en una sola lengua, el hecho de ser software libre facilita en gran medida su traducción y localización para que usuarios de diferentes partes del mundo puedan aprovechar estos beneficios.

3.3. Servidores de Mapas.

3.3.1. Cuadro de Compatibilidad.

Servidor	Fuentes de Datos		Publicación de OWS	Otros Estándares OGC	Solicitudes
	Vectorial	Ráster			
Deegree 	Base de Datos: Postgis, Oracle Spatial, ArcSDE. [WEB 59] Archivos: Shapefile. [WEB 33]	Base de Datos: Oracle GeoRaster Archivos: tiff, png, jpeg, gif, bmp, GeoTiff, EWC. [WEB 33].	WMS 1.1.0, 1.1.1, 1.3.0 WFS 1.0.0, 1.1.0 WCS 1.0.0 WFS -T [WEB 59]	SLD 1.0.0 CSW WPS GML KML [WEB 33]	HTTP GET HTTP POST Peticiones de Información de geometrías. [WEB 59].
GeoServer 	Base de Datos: Postgis, ArcSDE, DB2, Oracle Spatial, Mysql, SQL Server. Archivos: Shape. Otros: WFS externo. [WEB 57]	Base de Datos: Oracle GeoRaster. Archivos: GeoTiff, JPG, PNG, imágenes pyramid, formatos GDAL, Imágenes Mosaic. [WEB 57]	WMS WFS WFS-T WCS [WEB 57]	SDL GML [WEB 57]	HTTP POST
MapGuide Open Source 	Bases de Datos: Oracle Spatial, MySQL y Postgis. Archivos: SHP, SDF, ArcSDE, DWG. [WEB 61]	Archivos: JPG, PNG.	WMS 1.1.1 WFS		HTTP POST

<p>MapServer</p> 	<p>Base de datos: Oracle Spatial, Mysql, Postgis, ArcSDE. Archivos: shapefiles. [WEB 58]</p>	<p>Archivos: Tiff, GeoTiff, GDAL. [WEB 58]</p>	<p>WMS, WFS</p> <ul style="list-style-type: none"> • Servidor: 1.0.0, 1.0.7, 1.1.0, 1.1.1, 1.3.0. • Cliente: 1.0.0, 1.0.7, 1.1.0, 1.1.1 <p>WFS no transaccional (Cliente/Servidor) 1.0.0, 1.1.0 WCS 1.0.0, 1.1.0, 2.0.0. [WEB 58]</p>	<p>SDL 1.0.0 GML 2.1.2, 3.1.0.</p>	<p>HTTP GET HTTP POST</p>
<p>QGIS Server</p> 			<p>WMS 1.1.1, 1.3.0</p>	<p>SDL 1.0.0</p>	<p>HTTP GET</p>

Tabla 8: Comparativa de Servidores de Mapas, Cuadro de Compatibilidad.
Fuente: Propia.

3.3.2. Cuadro de Características Técnicas.

Servidor	Licencia	Versión	Plataformas	Lenguaje desarrollado	Servidor Web	Lenguajes de Scripting	Apoyo OSGEOS
Deegree	LGPL [WEB 59]	2.3 [WEB 59]	Windows Linux. [WEB 59]	Java (JRE or JSDK) versión 1.5.x [WEB 59]	Tomcat 5.5.x	Java Vb Script	SI (Graduado)
GeoServer	GPL v.2 [WEB 57]	2.1.3 [WEB 57]	Windows Linux Mac [WEB 57]	Java (JRE) versión 1.5.x [WEB 57]	Tomcat, Glassfish	GeoScript – GeoTools (Java, Python, JavaScript, Scala, and Groovy)	SI (Incubación)
MapGuide Open Source	LGPL [WEB 61]	2.2.0 [WEB 61]	Windows Linux [WEB 61]	.NET y C++, DWF y AJAX con DHTML [WEB 61]	Apache, IIS Web Server. [WEB 61]	PHP, .NET, Java y JavaScripts APIs. [WEB 38]	SI (Graduado)
MapServer	MIT Style. [WEB 58]	5.6.5. [WEB 58]	Windows Linux Mac Solaris. [WEB 58]	C [WEB 58]	Apache	PHP, Python, Perl, Ruby, Java y .NET [WEB 58]	SI
QGIS Server	GPL	1.7.0	Windows Linux MacOSX	C++	Es FastCGI/CGI	PyQGIS – Python	SI

**Tabla 9: Comparativa de Servidores de Mapas, Cuadro de Características Técnicas.
Fuente: Propia.**

3.3.3. Cuadro de Funcionalidades y Características Principales.

Funcionalidad/Característica	Servidor de Mapas				
	Deegree	GeoServer	MapGuide Open Source	MapServer	QGIS Server
Mapas Temáticos	SI (SLD). [WEB 59]	SI [WEB 57]	SI (A través de reglas) [WEB 61]	SI (Utilizando clases basadas en lógica de programación o en expresiones regulares). Y SDL [WEB 58]	SI
Simbología objetos de Imagen	SI (SLD). [WEB 59]	SI (SLD) [WEB 57]	SI	SI	SI
Transformación de coordenadas al vuelo	SI (Para más de 3000 Sistemas de Referencia de Coordenadas). [WEB 59]	SI (Soporta la mayoría de Bases de Datos de proyecciones EPSG). [WEB 57]	SI [WEB 61]	SI (para más de 1000 proyecciones proporcionadas por la librería Proj.4). [WEB 58]	NO
Representación por escala	SI	SI (Tiene la posibilidad de filtrar estilos dependientes del nivel de escala) [WEB 57]	SI	SI [WEB 39]	SI
Etiquetas de Features	SI	SI	SI	SI [WEB 39]	SI
Soporte entidades (features) Simples y Complejas	SI	SI	SI	SI [WEB 39]	SI

Tabla 10: Comparativa de Servidores de Mapas, Cuadro de Funcionalidades y Características Principales.
Fuente: Propia.

3.3.4. Selección de Herramienta.

Se ha seleccionado a MapServer como servidor de servicios WMS, WCS y WFS, por las siguientes razones:

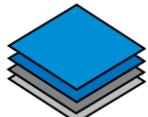
- Se encuentra desarrollado en C, la razón por la cual es más rápido al momento de renderizar un mapa.
- Tiene un módulo en PHP (MapScript), para el mayor control y generación de configuraciones dinámicas (MapFiles).
- Cuenta con simbología a través de expresiones regulares y SLD.
- Es compatible con la mayoría de bases de datos geoespaciales y formatos de información vectorial y raster.

Además, se utilizara GeoServer como servidor de servicios WFS-T, por ser más ligero que Deegree, ya que son los dos los únicos servidores que ofrecen este servicio. Además es compatible con GeoExt y OpenLayers.

3.4. Clientes GIS Livianos.

3.4.1. Cuadro de Compatibilidad.

Cliente	Formatos	Cliente de OGC Web Services	Fuentes de Datos
Geomajas 	WKT/WKB, JSON, GML, GeoJSON, SDL	WMS (1.1.0, 1.1.1) WFS (1.0.0, 1.1.0) WFS – T [WEB 47]	<ul style="list-style-type: none">• Capas comerciales: Google Maps, Yahoo.• Otros: Open Street Maps.
Mapbender	GeoRSS, GML	WMS (1.0.0, 1.1.0, 1.1.1) WFS-T (1.0.0, 1.1.0) [WEB 66]	No se encontro Información

			
GeoExt 	GeoJSON	WMS WFS WFS-T WCS	<ul style="list-style-type: none"> • Capas comerciales: Google Maps, Bing, Yahoo. Microsoft, MultiMap. • Otros: ArcGIS Images, MapGuide, MapServer, TileCache, Open Street Maps.
MapFish 	KML, GML [WEB 49]	WMS WFS [WEB 49]	<ul style="list-style-type: none"> • Capas comerciales: Google Maps, Bing, Yahoo. Microsoft, MultiMap. • Otros: ArcGIS Images, MapGuide, MapServer, TileCache, Open Street Maps.
OpenLayers 	Atom, ArcXML, KML, GeoJSON, WKT, GML, GeoRSS, SVG, VML, OSM, SLD, WMTS. [WEB 46]	WMS WFS WCS WFS-T [WEB 46]	<ul style="list-style-type: none"> • Capas comerciales: Google Maps, Bing, Yahoo. Microsoft, MultiMap. • Otros: ArcGIS Images, MapGuide, MapServer, TileCache, Open Street Maps. [WEB 40]
GeoMoose 	No se encontro Información	WMS WFS [WEB 50]	<ul style="list-style-type: none"> • Capas Comerciales: Google, VirtualEarth, YahooMaps. • Otros: Tilecache, ArcGIS REST, WMS. [WEB 50]

**Tabla 11: Comparativa de GIS Livianos, Cuadro de Compatibilidad.
Fuente: Propia.**

3.4.2. Cuadro de Características Técnicas.

Cliente	Licencia	Versión	Plataformas	Lenguaje		Extensible	Apoyo OSGEO
				Escrito	Programación		
Geomajas	AGPL v.3 [WEB 47]	1.6 [WEB 47]	Windows Linux Mac. [WEB 47]	Lenguaje: Java. Framework: Spring Framework, JTS Topology Suit, GeoTools, Hibernate espacial, GWT. [WEB 47]	Lenguaje: Java, Api: GWT (Google Web Toolkit) [WEB 34]	SI (Mecanismos de extensión orientada a plug-ins). [WEB 47]	NO
Mapbender	Licencia dual GNU General Public License y la licencia Simplified BSD license. [WEB 48]	2.7.2. [WEB 48]	Windows Linux Mac [WEB 48]	Lenguajes: PHP, Javascript, XML. [WEB 66]	Lenguaje: PHP, JavaScript. Api: JQuery y JQuery UI. [WEB 66]		SI (Graduado)
GeoExt	BSD	1.1	Windows Linux Mac	Lenguaje: Javascript Framework: Ext JS, OpenLayers.	Lenguaje: Javascript	SI	NO
MapFish	BSD License	2.2	Windows Linux Mac	Lenguaje: Javascript, Python. Framework: Ext JS, Openlayers, Pylons. Api: GeoExt.	Lenguaje: Java, Javascript, PHP, Python. Framework: Ext JS, Openlayers y	SI	NO (Incubación)

				Librerías: GeoAlchemy, Shapely. [WEB 37]	GeoExt. [WEB 37]		
OpenLayers	MIT-style license. [WEB 46]	2.11. [WEB 46]	Windows Linux Mac. [WEB 46]	Lenguaje: Javascript. [WEB 46]	Lenguaje: Javascript. [WEB 46]	SI (Altamente Extensible). [WEB 46]	SI
GeoMoose	MIT based license. [WEB 50]	2.4. [WEB 50]	Windows Linux Mac[WEB 50]	Lenguaje: JavaScript, PHP. [WEB 62]	Lenguaje: JavaScript, PHP. [WEB 62]	SI (Arquitectura Modular). [WEB 50]	NO (En incubación)

Tabla 12: Comparativa de GIS Livianos, Cuadro de Características Técnicas.
Fuente: Propia.

3.4.3. Cuadro de Funcionalidades y Características Principales.

Funcionalidad/Característica	Clientes Livianos					
	Geomajas	Mapbender	GeoExt	MapFish	OpenLayers	GeoMoose
Componente Cliente y Servidor	SI [WEB 47]	SI [WEB 66]	NO	SI	NO	SI [WEB 62]
Despliegue de Mapas	SI	SI	SI	SI	SI	SI
Administración	SI [WEB 47]	SI [WEB 66]	NO	NO	NO	NO
Widgets	SI [WEB 47]	SI [WEB 66]	SI	SI	NO	NO
Funciones de Edición Básica	SI [WEB 34]	SI [WEB 66]	SI	SI	SI	NO

Marcadores	NO	-	SI	SI	SI	NO
Popups	NO	-	SI	NO	SI	NO
Renderización de datos geográficos	NO	-	SI	SI	SI (SVG, VML, Canvas) [WEB 40]	SI
Dependencias de otros clientes	NO	-	OpenLayers	GeoExt OpenLayers [WEB 37]	No	OpenLayers [WEB 50]
Reproyección	SI [WEB 47]	-	SI	SI	SI [WEB 40]	SI [WEB 50]
Snapping	SI [WEB 47]	-	-	-	-	-
Posibilita la creación de un Aplicación RIA	SI	SI [WEB 66]	SI	SI [WEB 37]	SI	-
Store de datos Geográficos	NO	NO	SI	SI	NO	NO
Herramientas de Navegación	SI	SI [WEB 66]	SI	SI [WEB 37]	SI	SI [WEB 50]
Dependencia de Servidor de Mapas	NO	NO	NO	NO	NO	MapServer [WEB 50]

Tabla 13: Comparativa de GIS Livianos, Cuadro de Funcionalidades y Características Principales.
Fuente: Propia.

3.4.4. Selección de Herramientas.

Se ha seleccionado como cliente ligero a OpenLayers, por las siguientes razones:

- Es compatible con la mayoría de formatos de marcado y vectorial orientados a la web.
- Soporta los estándares OGC necesarios WMS, WFS, WCS, WFS-T, SLD, GML, KML.
- Permite renderización SVG.

- No es solamente un cliente ligero, sino que también es un API, que ayuda al desarrollo de aplicaciones GIS más complejas.
- Compatible con GeoExt y ExtJS.

Además se utilizara GeoExt como:

- Store de datos geográficos, para indexar un gráfico con su información.
- Controles Ricos en GIS.

3.5. Clientes GIS Pesados.

3.5.1. Cuadro de Compatibilidad.

Cliente	Fuentes de Datos y Formatos	Estándares OGC
GvSIG 	<ul style="list-style-type: none"> • Ráster: ecw, ENVI hdr, ERDAS img, (Geo)TIFF, GRASS. • Vectorial & CAD: shapefile, GML, KML, DGN, DXF, DWG. • Bases de datos geográficas: PostGIS, MySQL, Oracle, ArcSDE. • Datos remotos: ECWP, ArcIMS, estándares OGC. [WEB 54] 	WMS WFS WCS SLD WFS-G [WEB 54]
GRASS 	<ul style="list-style-type: none"> • Ráster: PNG, TIFF, VRML, • Vectorial: SVG, DXF, VTK • Bases de datos geográficas: SQLite, PostgreSQL. 	WFS WMS
Kosmo Desktop	<ul style="list-style-type: none"> • Ráster: ECW, MrSID, Tiff, JPG, GIF, PNG, BMP. 	WMS 1.0.0/1.1.0/1.3.0

	<ul style="list-style-type: none"> • Vectorial: Shapefiles, GML, DXF, DWG, DGN, CSV (por extensión). • Bases de datos geográficas: PostGIS, Oracle, MySQL. [WEB 55] 	WFS 1.0.0/1.1.0 SDL [WEB 55]
OpenJump GIS 	<ul style="list-style-type: none"> • Ráster: ECW, MrSID, JPG, TIFF, MIF. • Vectorial: Shapefiles, GML, DXF. • Base de datos geográfica: Postgis, ArcSDE, Oracle y MySQL. [WEB 56] 	GML 2 WMS WFS WFS-T SLD [WEB 56]
Quantum GIS 	<ul style="list-style-type: none"> • Ráster: • Vectorial: shapefiles, MapInfo, SDTS y GML. • Base de datos geográfica: Postgis. [WEB 52] 	WMS WFS (Completo) [WEB 52]
uDig 	<ul style="list-style-type: none"> • Ráster: ECW, MrSID, JPEG, PNG, TIFF. • Vectorial: Shapefiles. • Base de datos geográfica: ArcSDE, Oracle, DB2, Postgis. [WEB 69] 	WMS WFS WFS-T WMS-C SDL [WEB 69]

Tabla 14: Comparativa de GIS Pesados, Cuadro de Compatibilidad.
Fuente: Propia.

3.5.2. Cuadro de Características Técnicas.

Cliente	Licencia	Versión	Plataformas	Lenguaje Desarrollado
GvSIG	GPL v.2 [WEB 54]	1.11	Windows Linux MacOS X (Distribuciones no oficiales) [WEB 43]	Java. [WEB 43]
GRASS	GPL v.2 [WEB 53]	6.4.0RC6	Windows Linux MacOS X [WEB 53]	C, Python, Bourne Shell [WEB 53]
Kosmo Desktop	GPL v.2 [WEB 55]	2.0.1	Windows Linux [WEB 55]	Java [WEB 55]
OpenJump GIS	GPL [WEB 56]	1.5.1	Windows Linux Mac Unix [WEB 56]	Java [WEB 56]
Quantum GIS	GPL [WEB 52]	1.7.4	Windows Linux Mac Unix [WEB 52]	Python, C++ [WEB 42]
uDIG	LGPL v.2 [WEB 69]	1.2.2	Windows Linux Mac [WEB 69]	Java [WEB 68]

Tabla 15: Comparativa de GIS Pesados, Cuadro de Características Técnicas.

Fuente: Propia.

3.5.3. Cuadro de Funcionalidades y Características Principales.

Funcionalidad/Característica	Clientes Pesados					
	GvSIG	GRASS	Kosmo Desktop	OpenJump GIS	Quantum GIS	UDig
Edición Espacial y de Atributos (Información Vectorial).	SI	SI	SI	SI	SI	SI
Consulta y Edición de Información espacial almacenada en Postgis.	SI	NO	SI	SI	SI	Solo consulta
Exportación de información vectorial a Postgis.	SI	SI	SI	NO	SI (Plugin Spit)	NO
Operaciones de Geoprocesamiento.	SI [WEB 43]	SI	SI [WEB 55]	SI	SI	NO
Georeferenciación y reproyección	SI [WEB 43]		SI [WEB 55]		SI [WEB 42]	NO
Análisis y Consulta especial y de atributos (Información Vectorial).	SI	SI	SI	SI	SI	Solo Consulta
Consumo de OGC Web Services. (WMS y WFS)	SI	SI	SI	SI	SI	SI
Consumo de Servicio WMS-C	NO	NO	NO	NO	NO	SI

Tabla 16: Comparativa de GIS Pesados, Cuadro de Funcionalidades y Características Principales.
Fuente: Propia.

3.5.4. Selección de Herramientas.

De acuerdo a las prestaciones de estas herramientas se ha elegido a las herramientas para ser utilizadas de la siguiente manera:

GvSIG: Para migración de datos desde el Shape a la base de datos PostGIS, ya que esta herramienta migra los datos tal como se encuentra en el archivo shape sin alterar los atributos, además sigue el estándar: las claves primarias o id's los nombra con gid y la columna geométrica con el nombre the_geom.

Quantum GIS: Para prueba de servicios WMS, WFS y WFS-T, ya que es un cliente rápido en consulta de servicios OWS.

uDig: Para prueba de servicios WMS-C, esta es la única herramienta que permite consumir un servicio WMS en Cache.

Kosmo: Para edición de información geográfica en PostGIS, ya que cuenta con más herramientas de edición.

3.6. Servidores de Cache Geoespacial.

3.6.1. Cuadro de Compatibilidad.

Servidor	Formatos de Imagen	Servicios	Origen de Datos
Tile Cache	JPG, PNG	WMS-C, TMS [WEB 32]	MapServer, WMS, Mapnik [WEB 31]
GeoWebCache	JPG, PNG	WMS-C, TMS, WMTS [WEB 28]	WMS, Geoserver. [WEB 29]

Tabla 17: Comparativa de Cache Geoespaciales, Cuadro de Compatibilidad.

Fuente: Propia.

3.6.2. Cuadro de Características Técnicas.

Servidor	Licencia	Versión	Plataformas	Lenguaje Desarrollado	Servidor Web
Tile Cache	BSD-ish License [WEB 31]	2.11 [WEB 31]	Linux	Python [WEB 31]	Apache [WEB 31]
GeoWebCache	LGPL [WEB 28]	1.3-RC3 [WEB 28]	Windows, Linux	Java [WEB 29]	Tomcat

Tabla 18: Comparativa de Cache Geoespaciales, Cuadro de Características Técnicas.

Fuente: Propia.

3.6.3. Selección de Herramientas.

Se utilizara TileCache como servidor de WMS-C, ya que para instalar este servidor solo se necesita de python y Apache. Además es mucho más liviano.

3.7. Propuestas de software para el desarrollo de un Sistema GIS.

3.7.1. Software Privativo.

Tipo de Software	Software	Precios
Servidor de Mapas	ArcGIS Server	1209.00
Cliente Pesado	ARGIS 9.2 Desktop	1500.00
Base de Datos	AscSDE	740.00
	Microsoft SQL Server 2008 Standard Edition	4000.00
Cliente Ligero	ArcGIS Explorer	00.00
Sistema Operativo	Windows Server 2008	1240.00
Total		8689.00

Tabla 19: Propuesta precios con Software Privativo.

Fuente: Páginas oficiales de las Herramientas.

3.7.2. Software Libre.

Tipo de Software	Software	Precios
Servidor de Mapas	MapServer	00.00
	GeoServer	00.00
Cliente Pesado	gvSIG	00.00
	Quantum GIS	00.00
	uDIG	00.00
	Kosmo	00.00
Base de Datos	PostGIS	00.00
	PostgreSQL	00.00
Cliente Ligero	OpenLayers	00.00
	GeoExt	00.00
Sistema Operativo	Debian	00.00
Total		00.00

Tabla 20: Propuesta precios con Software Libre.

Fuente: Páginas oficiales de las Herramientas.

Capítulo IV: Diseño y Desarrollo del Aplicativo



CONTENIDO:

1. Fase de Exploración.
2. Fase de Planificación.
3. Fase de Iteración.
4. Fase de Producción.

4. Capítulo IV: Diseño y Desarrollo del Aplicativo.

El presente capítulo se desarrolla a través de las fases de la metodología de desarrollo Programación Extrema XP. Primeramente en la Fase de Exploración se realiza la recolección de historias de usuario detallando en ellas las necesidades del usuario, se propone la arquitectura del Sistema y de su funcionamiento; llegando así a describir las tareas necesarias para la realización del sistema y sus prioridades. En la Fase de planificación se realiza el Plan de Entregas, el cual cuenta con estimaciones de tiempo, para realizar los entregables en cada iteración; seguidamente se realiza la Fase de Iteración donde se detalla el trabajo realizado en cada tarea de las iteraciones necesarias para el desarrollo del Sistema, las mismas que tiene como salida un prototipo del sistema al cual se le aplica las pruebas de aceptación definidas por el usuario; por último cuando el Sistema cumple con todos los requerimientos y después de haber aprobado con todas las pruebas de aceptación se procede a la Fase de Producción, donde se detalla los procedimientos a seguir y requerimientos para que el sistema pueda funcionar en un entorno de Producción.

Sistema de Información Geográfica IMI “GIS IMI”

Programación Extrema

Cueva Cabrera Byron Mesias

Versión 1.0

Historia de Cambios

Fecha	Versión	Descripción	Autor
03/03/2011	1.0	Registro de Historias de Usuario	Byron Cueva
09/03/2011	1.0	Registro de Tareas de Ingeniería	Byron Cueva
11/03/2011	1.0	Fase de Planificación	Byron Cueva
08/04/2011	1.0	Registro Primera Iteración	Byron Cueva
17/05/2011	1.0	Registro Segunda Iteración	Byron Cueva
20/06/2011	1.0	Registro Tercera Iteración	Byron Cueva
06/07/2011	1.0	Registro Cuarta Iteración	Byron Cueva
20/07/2011	1.0	Registro Quinta Iteración	Byron Cueva
22/07/2011	1.0	Fase de Producción	Byron Cueva

Tabla 21: Historia de Cambios.

Fuente: Propia.

4.1. Fase de Exploración.

4.1.1. Roles.

Alcanzar las metas y construir lo que el cliente necesita es el objetivo primordial del equipo XP, y para eso se vale de una colaboración efectiva que se asume como compromiso por parte de cada uno de los integrantes. Cabe recalcar que XP sigue un proceso, pero que el empoderamiento a personas, guiados por buenas prácticas de desarrollo, maximizan los resultados.

A continuación se describen los roles:

Nombre	Descripción	Responsabilidad
Cliente (Customer)	Una persona o grupo de personas que tiene un sólido conocimiento del proceso del negocio actual y clarificará los problemas que el nuevo sistema ha de resolver.	<ul style="list-style-type: none">• Definir la metáfora del sistema.• Definir el valor de negocio.• Escribir historias de usuario.• Escribir o especificar las pruebas de aceptación.• Realizar conjuntamente con el equipo de desarrollo la planificación de las entregas.
Programador (Programmer)	Responsable del desarrollo del proyecto.	<ul style="list-style-type: none">• Estimar historias de usuario.• Realizar prototipos.• Escribir código.• Diseñar.• Participar en reuniones para la planificación.• Participar en reuniones diarias con el equipo.• Escribir pruebas unitarias.
Gestor (Manager)	Encargado de la gestión del proyecto internamente y hacia los clientes.	<ul style="list-style-type: none">• Aclarar obstáculos e interferencias que obstruya el trabajo del equipo de desarrollo.• Nivel alto de entendimiento del negocio, alineado con los valores del equipo.
Rastreador (Tracker)	Encargado del seguimiento y responsable de reunir métricas	<ul style="list-style-type: none">• Revisión de historias de usuario.• Revisión de tareas completadas.

	significativas para mostrar el estado del proyecto.	<ul style="list-style-type: none"> Comunicar el progreso del sistema.
Probador (Tester)	Encargado de realizar las pruebas de aceptación.	<ul style="list-style-type: none"> Realiza pruebas de aceptación, asegurando que el sistema funcione como se esperaba.
Entrenador (Coach)	Encargada de facilitar la comunicación dentro del equipo de desarrollo.	<ul style="list-style-type: none"> Proveer guías a los miembros del equipo de forma que aplique los lineamientos y prácticas necesarias en un desarrollo XP y se siga el proceso correctamente.

Tabla 22: Roles de XP.
Fuente: Propia.

4.1.2. Integrantes del Equipo XP.

El equipo XP está integrado por aquellas personas involucradas directamente en el desarrollo del sistema GIS IMI. A continuación se presenta una lista de los integrantes con su respectivo rol en el proyecto.

Nombre	Descripción	Rol XP
Lic. Sonia Bossano Subia.	Jefe del área de software de la dirección de TIC.	Rastreador y Entrenador
Ing. Manuel Lara Muñoz.	Analista de Sistemas de la Dirección de TIC.	Gestor y Probador
Ing. Javier Caiza.		Cliente y Probador
Byron Mesias Cueva Cabrera	Tesista	Programador

Tabla 23: Integrantes del Equipo XP.
Fuente: Propia.

4.1.3. Historias de Usuario.

Después de haber realizado las respectivas reuniones con el personal responsable de la Dirección de Catastros y Avalúos y TIC, a continuación se detallan las historias de usuario recopiladas. En estas se describen los requisitos que se diseñaron y programaron en la fase de iteración, además que servirán para describir la funcionalidad del sistema.

Cabe recalcar que pueden existir historias de usuario que son conocidas como “de valor de negocio cero”, ya que a primera vista no generan ningún beneficio para el usuario pero son indispensables

para el desarrollo del sistema. Tales historias de usuario pueden consistir en la instalación del ámbito de desarrollo, creación del esquema inicial de la base de datos, entre otros.

Las historias se han dividido y organizado en base a la estimación de que cada una de ellas que no requiera la inversión de más de 3 semanas de empleo en su desarrollo.

Nota: Los puntos de estimación de cada historia se calificara de acuerdo a la siguiente aclaración:

1 pto. = 1 Semana Ideal y 1/5 pto. = 1 día Ideal. Siendo 3 ptos. la puntuación más alta. La semana y día Ideal se describe más abajo en la fase de planificación.

Historia de Usuario	
Número: 1	Nombre Historia de Usuario: Seguridad y control de acceso.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Manuel Lara.	Puntos Estimados: 1 y 3/5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario del sistema, quiero ingresar al mismo a través de un usuario y contraseña, de forma que mi cuenta se encuentre segura y lo que realice no tenga relación con otros usuarios del sistema.	
Observaciones: Se debe utilizar la base de datos centralizada de usuarios del Municipio. Cada usuario tiene determinados accesos a los datos y funcionalidades del sistema, se asignara estos permisos a través de menús e ítems que contenga la aplicación.	

Tabla 24: Historia Nro. 1.

Fuente: Propia.

Historia de Usuario	
Número: 2	Nombre Historia de Usuario: Repositorio de datos geográficos.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Manuel Lara.	Puntos Estimados: 2 y 2/5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como administrador de sistemas, quiero un repositorio de datos con la información geográfica que se genera en la Dirección de Catastros y Avalúos, de forma que esta información se encuentre	

centralizada.
Observaciones: Debe existir en el repositorio roles de usuario como administrador, consultor y dibujante.

Tabla 25: Historia Nro. 2.
Fuente: Propia.

Historia de Usuario	
Número: 3	Nombre Historia de Usuario: Publicación de mapas en la web.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Manuel Lara.	Puntos Estimados: 3
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)
Descripción: Como administrador de sistemas, quiero que la información geográfica debidamente permitida pueda ser publicada a través de la web, de forma que la misma pueda visualizarse en un entorno web y que a futuro se pueda compartir con otras instituciones.	
Observaciones:	

Tabla 26: Historia Nro. 3.
Fuente: Propia.

Historia de Usuario	
Número: 4	Nombre Historia de Usuario: Despliegue de Información geográfica y colección de capas.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2 y 2/5
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero visualizar a la información catastral y que a su vez sea solamente sea visible al usuario que tenga permisos de verla. También deseo contar con la colección de capas que se encuentren visualizando, de forma que sea posible apagar y encender una capa de la navegación.	
Observaciones: Cada usuario debería tener permisos de visualización por cada capa.	

Tabla 27: Historia Nro. 4.
Fuente: Propia.

Historia de Usuario	
Número: 5	Nombre Historia de Usuario: Funciones de navegación y descriptor de leyendas.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 4/5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero navegar sobre el mapa y realizar acciones de aumento, disminución y centralizado de zoom sobre el mismo. También quiero contar con la descripción de la información que se encuentra visualizando, de tal forma que la navegación sea más fácil y amigable.	
Observaciones:	

Tabla 28: Historia Nro. 5.
Fuente: Propia.

Historia de Usuario	
Número: 6	Nombre Historia de Usuario: Adición y eliminación de capas.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 4/5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero agregar capas a la navegación, las mismas a las cuales tenga permisos de verlas o a su vez eliminarlas de la visualización actual, de tal forma que me permita montar una capa para poder realizar un análisis, ya que la información aumenta de acuerdo a los aspectos que se desean representar de una forma geográfica.	
Observaciones:	

Tabla 29: Historia Nro. 6.
Fuente: Propia.

Historia de Usuario	
Número: 7	Nombre Historia de Usuario: Estimación de longitud y área.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	

Usuario: Ing. Javier Caiza.	Puntos Estimados: 4/5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero una herramienta que me permita realizar una estimación de la longitud y área de una zona que desee medir.	
Observaciones:	

Tabla 30: Historia Nro. 7.
Fuente: Propia.

Historia de Usuario	
Número: 8	Nombre Historia de Usuario: Ubicación rápida de información catastral.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 1 y 3/5
Prioridad en el Negocio: Media (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero realizar búsquedas de la información catastral por clave catastral, calles o lugares principales, permitiéndome la ubicación rápida de la misma.	
Observaciones:	

Tabla 31: Historia Nro. 8.
Fuente: Propia.

Historia de Usuario	
Número: 9	Nombre Historia de Usuario: Consulta de información predial.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 4/5
Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero obtener información relevante sobre un predio al dar clic sobre el mismo, información como: su propietario, área, valor e historial de fotos, esta información será filtrada y visible dependiendo del permiso que el usuario tenga en la aplicación, de tal forma que no todos los usuario puedan ver información que sea sensible.	

Observaciones:

**Tabla 32: Historia Nro. 9.
Fuente: Propia.**

Historia de Usuario	
Número: 10	Nombre Historia de Usuario: Búsqueda de un predio.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero ubicar un predio, ya sea buscándolo por clave catastral, cédula o nombres y apellidos.	
Observaciones:	

**Tabla 33: Historia Nro. 10.
Fuente: Propia.**

Historia de Usuario	
Número: 11	Nombre Historia de Usuario: Consulta de la Zonificación de un Predio.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)
Descripción: Como usuario, quiero consultar la zonificación de un predio al dar clic sobre él.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

**Tabla 34: Historia Nro. 11.
Fuente: Propia.**

Historia de Usuario	
Número: 12	Nombre Historia de Usuario: Consulta de los usos de suelo de un predio.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5

Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero consultar los usos de suelo que se puede realizar sobre un predio.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

Tabla 35: Historia Nro. 12.
Fuente: Propia.

Historia de Usuario	
Número: 13	Nombre Historia de Usuario: Consulta de la descripción del terreno.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero consultar la descripción del terreno.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

Tabla 36: Historia Nro. 13.
Fuente: Propia.

Historia de Usuario	
Número: 14	Nombre Historia de Usuario: Consulta de la infraestructura y servicios.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Alta. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero consultar la infraestructura y servicios de un predio.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

Tabla 37: Historia Nro. 14.
Fuente: Propia.

Historia de Usuario	
Número: 15	Nombre Historia de Usuario: Reporte de las propiedades de un ciudadano.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Media. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero un reporte de las propiedades de un ciudadano.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

Tabla 38: Historia Nro. 15.
Fuente: Propia.

Historia de Usuario	
Número: 16	Nombre Historia de Usuario: Consulta del Impuesto Predial.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Media. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
Descripción: Como usuario, quiero consultar el impuesto predial de un predio, ya sea si este se ha pagado o se encuentre atrasado.	
Observaciones: La consulta se puede realizar si tiene permisos para hacerlo.	

Tabla 39: Historia Nro. 16.
Fuente: Propia.

Historia de Usuario	
Número: 17	Nombre Historia de Usuario: Consulta de los predios colindantes.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Media. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)

<p>Descripción:</p> <p>Como usuario, quiero consultar los colindantes asía un predio, que me permita distinguir cada uno de ellos.</p>
<p>Observaciones:</p> <p>La consulta se puede realizar si tiene permisos para hacerlo.</p>

Tabla 40: Historia Nro. 17.

Fuente: Propia.

Historia de Usuario	
Número: 18	Nombre Historia de Usuario: Impresión de Croquis de un predio.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 2/5
Prioridad en el Negocio: Media. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
<p>Descripción:</p> <p>Como usuario, quiero imprimir el croquis en formato pdf.</p>	
<p>Observaciones:</p> <p>La consulta se puede realizar si tiene permisos para hacerlo.</p>	

Tabla 41: Historia Nro. 18.

Fuente: Propia.

Historia de Usuario	
Número: 19	Nombre Historia de Usuario: Consulta del pago del impuesto predial por parroquias.
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre): No Aplica	
Usuario: Ing. Javier Caiza.	Puntos Estimados: 4/5
Prioridad en el Negocio: Media. (Alta / Media / Baja)	Riesgo en el Desarrollo: Bajo. (Alto / Medio / Bajo)
<p>Descripción:</p> <p>Como usuario, quiero realizar la consulta del pago del impuesto predial de una parroquia y por la fecha de vencimiento, de tal manera que pueda visualizar y distinguir cada uno de estos.</p>	
<p>Observaciones:</p> <p>La consulta se puede realizar si tiene permisos para hacerlo.</p>	

Tabla 42: Historia Nro. 19.

Fuente: Propia.

4.1.4. Metáfora.

A través del trabajo realizado durante la obtención de las historias de usuario con el personal responsable de la dirección de Catastros y Avalúos y TIC, se pudo determinar la metáfora del sistema descrita a continuación:

El sistema propuesto es una aplicación GIS, que permite el despliegue de la información catastral a través de mapas temáticos, generando una visión y análisis más amplio al unir el objeto geográfico con la información catastral de Ibarra. Además el sistema permite realizar búsquedas y ubicación por aspectos geográficos e información alfanumérica. Finalmente el sistema permite la obtención de la información que afecta o describe la realidad de un predio, se podría decir que un predio es la unidad de información más esencial al estudiar el castro de una ciudad.

4.1.5. Diseño del Sistema.

A partir de las historias de usuario se procede a diseñar la arquitectura del sistema, la arquitectura funcional y a determinar los módulos con los cuales contara el sistema.

4.1.5.1. Arquitectura del Sistema.

El sistema GIS IMI estará desarrollado a través de una tecnología Web, su lógica estará enfocada en el patrón MVC, conformada por:

Modelo: En esta capa se utilizara el ORM Doctrine, el cual permite obtener la información de una tabla a través de un objeto, permitiendo de esta manera un acceso a los datos de una forma más fácil y sencilla.

Controlador: Este es el control del flujo de las solicitudes realizadas a la aplicación, en esta capa se utilizara el framework Symfony 1.4.16, si bien este ya cuenta con el patrón MVC, pero este será modificado en la parte de la Vista, permitiéndole la compatibilidad con controles Ricos en la Web.

Vista: La parte visual asía el usuario estará desarrollada con los frameworks: Ext JS 3.4.0, GeoExt 1.1 y OpenLayers 2.11.

Ext JS nos permitirá crear una interfaz de escritorio, la misma que es de fácil uso y más atractiva asía el usuario. También nos permitirá optimizar el rendimiento a través del control de solicitudes manejadas por Ajax.

OpenLayers se utilizara para el consumo de los servicios OWS, visualización de mapas y generador de información geográfica en formato SVG.

GeoExt como store de datos (carga de datos espaciales en la vista en formato GeoJSON) y además para la generación de Widgets GIS.

Datos: Como base de datos Alfanumérica y Binaria se utilizara Postgresql, la misma que utilizan todos los sistemas del Municipio de Ibarra. También se utilizara Postgis que es un complemento para Postgresql, para el manejo de información geográfica.

Además del patrón MVC, se agregara otra capa para publicación de servicios OWS. En esta capa se utilizara: MapServer para publicación de WMS, WFS Básico y WCS; GeoServer para publicación de WFS Transaccional; y TileCache para la generación de WMS-C, que es un WMS en cache.

Esta arquitectura fue propuesta pensando en el enfoque que tiene la aplicación, permitiéndonos que sea extensible.

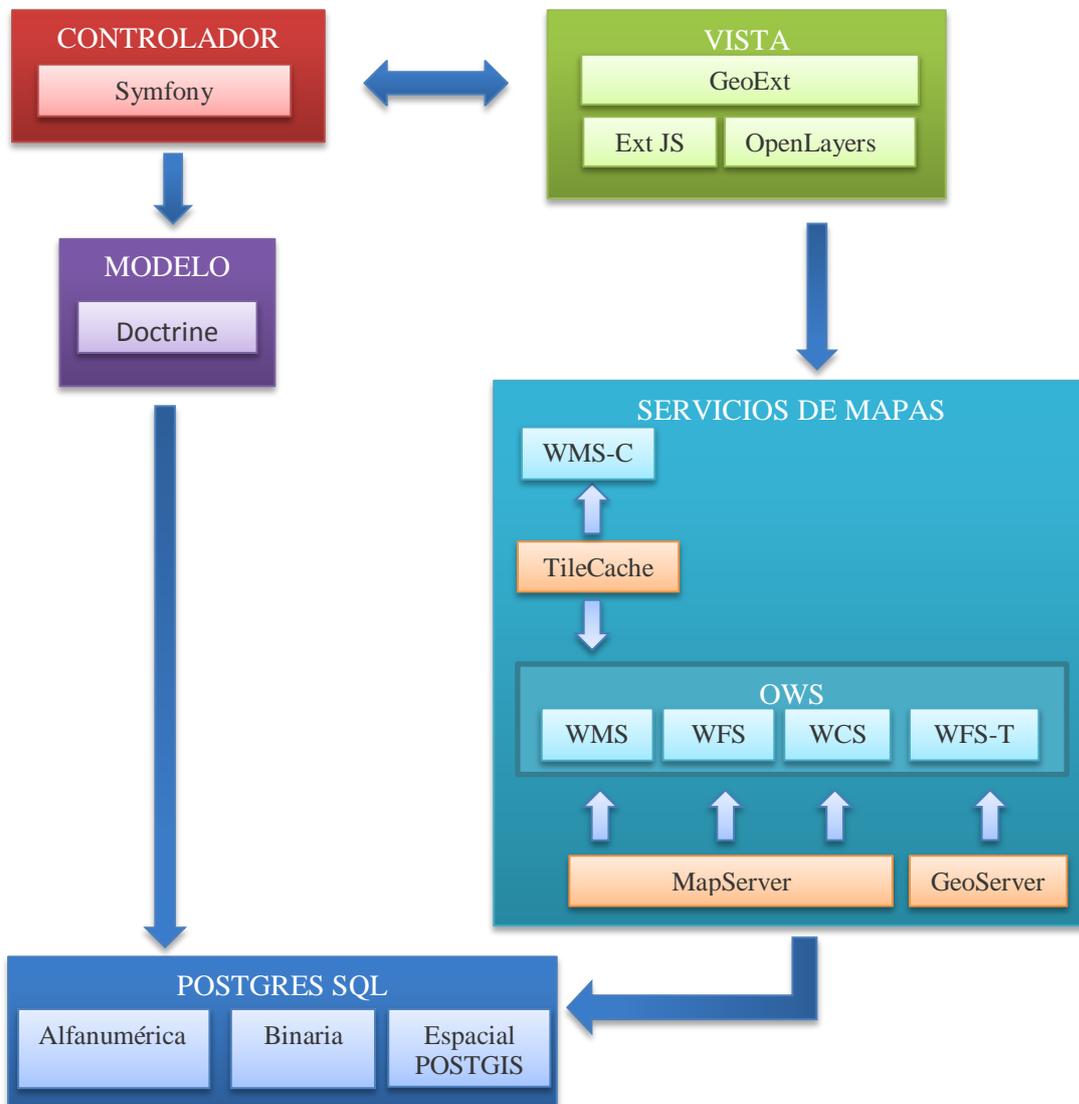


Figura 27: Arquitectura del Sistema.
Fuente: Propia.

4.1.5.2. Arquitectura Funcional.

El Sistema GIS IMI, será desarrollado sobre la plataforma Linux como sistema operativo Debian Squeeze, como servidor de aplicaciones Apache, Apache Tomcat como servidor de aplicaciones para el funcionamiento del Servidor de mapas GeoServer y por ultimo como repositorio de datos alfanuméricos y Binarios PostgreSQL y Postgis para la información Geográfica.

El flujo del funcionamiento del Sistema se centraliza en la publicación y consumo de los servicios de mapas, empezando por los repositorios de datos como son las bases de datos Alfanumérica, Binaria y Espacial. Los datos geográficos serán consultados por los servidores de mapas para la generación y renderización de mapas, los mismos que serán publicados al internet a través de OWS (OGC Web Services), luego estos servicios serán consumidos por la aplicación GIS IMI y clientes pesados en el internet. Por último la aplicación GIS IMI permitirá unir el objeto geográfico con su respectiva información, y esta será accesible a través de los navegadores más principales como Internet Explorer, Firefox, Google Chrome, Opera y Safari.

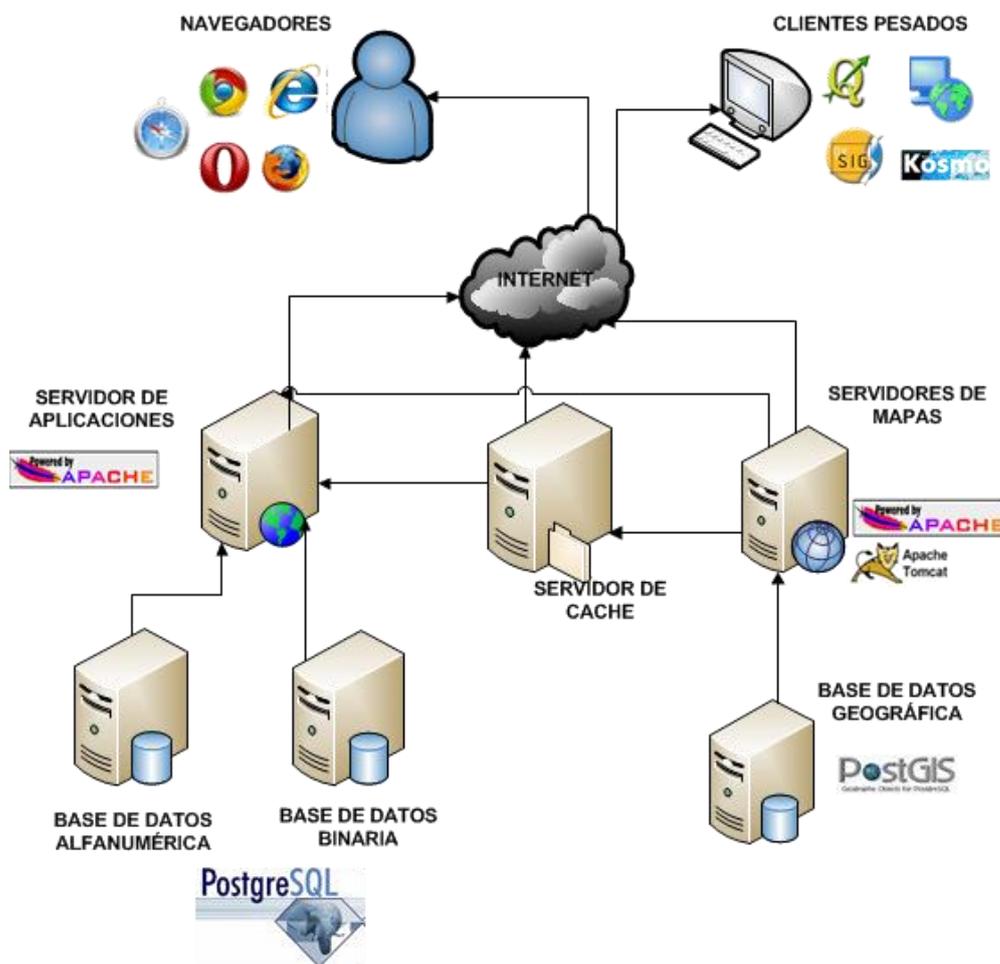


Figura 28: Arquitectura Funcional del Sistema.
Fuente: Propia.

4.1.5.3. Módulos del Sistema.

A partir de la descripción de las funciones y procesos obtenidos en las historias de usuario, se ha diseñado el Sistema GIS IMI teniendo como módulos principales los siguientes:

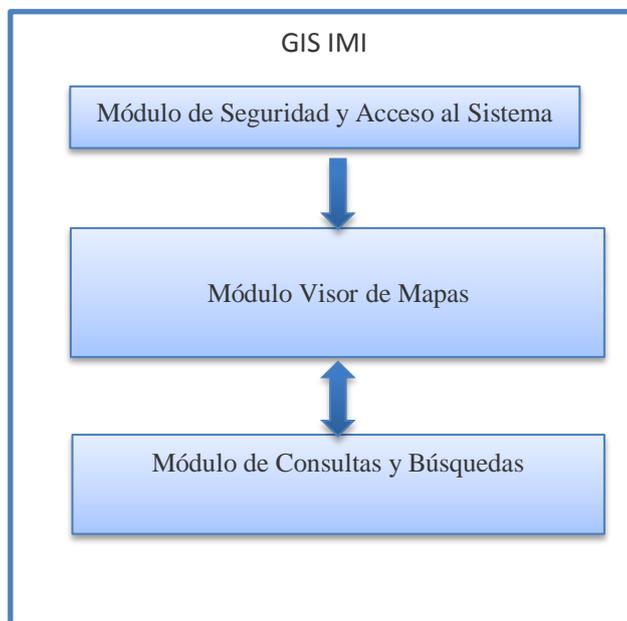


Figura 29: Módulos del Sistema.
Fuente: Propia.

Descripción de Módulos:

Módulo de Seguridad y Acceso al Sistema: En este módulo es donde el usuario se identifica y al momento de tener acceso al sistema, se asignan los diferentes permisos como menús, submenús e ítems.

Módulo Visor de Mapas: Este módulo es el principal de la aplicación el mismo que permite el despliegue y navegación de la información catastral, a través del consumo de los servicios OWS.

Módulo de Consultas y Búsquedas: Este módulo permite obtener la información que el objeto geográfico genera, es decir información descriptiva, a través de búsquedas por diferentes conceptos ya sean geográficos o información alfanumérica.

4.1.6. Tareas.

A continuación se realiza el análisis de requerimientos, dividiendo las historias de usuario, generadas por el usuario, en tareas pequeñas, las cuales no le llevarán a los desarrolladores más de un par de días en su implementación.

Cada tarea se ha clasificado dependiendo del módulo al cuál le pertenece la misma.

4.1.6.1. Módulo de Seguridad y Acceso al Sistema.

- **Historia Nro.: 1**

Tarea de Ingeniería	
Número Tarea: 1.1	Número de Historia: 1
Nombre Tarea: Diseñar la estructura de datos para el control de Acceso al Sistema.	
Tipo de Tarea : Diseño Desarrollo / Corrección / Mejora / Otra (especificar)	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se realiza el diseño de la base de datos para el registro de todos los datos del usuario, así como también para el control de la asignación de permisos, sobre el Sistema.	

Tabla 43: Tarea 1.1.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 1.2	Número de Historia: 1
Nombre Tarea: Crear la interfaz para el ingreso al Sistema.	
Tipo de Tarea : Desarrollo y Diseño	Puntos Estimados: 6/5
Programador Responsable: Byron Cueva.	
Descripción: Se diseña una ventana en la que se introducirá el código del usuario y la contraseña para el acceso al Sistema. A su vez la lógica de asignación de los permisos sobre el sistema al momento de ingresar al mismo.	

Tabla 44: Tarea 1.2.
Fuente: Propia.

4.1.6.2. Módulo Visor de Mapas.

- **Historia Nro.: 2**

Tarea de Ingeniería	
Número Tarea: 2.1	Número de Historia: 2
Nombre Tarea: Instalación y configuración del Módulo Postgis.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	

Descripción: Se instala y configura el motor de base de datos PostgreSQL con el módulo Postgis.

Tabla 45: Tarea 2.1.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 2.2	Número de Historia: 2
Nombre Tarea: Creación de la base de datos y roles de usuario.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Se crea la base de datos para el manejo de la información geográfica, así como también los roles de usuario y sus respectivos permisos.	

Tabla 46: Tarea 2.2.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 2.3	Número de Historia: 2
Nombre Tarea: Recopilación de Información catastral.	
Tipo de Tarea: Recopilación.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se realiza la recopilación de la información catastral.	

Tabla 47: Tarea 2.3.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 2.4	Número de Historia: 2
Nombre Tarea: Migración de la información geográfica.	
Tipo de Tarea: Migración.	Puntos Estimados: 1 y 3/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación del cliente pesado GvSIG, creación de esquemas y migración de las capas más principales de catastros a la base de datos.	

Tabla 48: Tarea 2.4.

Fuente: Propia.

- **Historia Nro.: 3**

Tarea de Ingeniería

Número Tarea: 3.1	Número de Historia: 3
Nombre Tarea: Instalación de Apache.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación y configuración de Apache.	

Tabla 49: Tarea 3.1.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 3.2	Número de Historia: 3
Nombre Tarea: Instalación y configuración de MapServer.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación del Servidor de Mapas MapServer.	

Tabla 50: Tarea 3.2.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 3.3	Número de Historia: 3
Nombre Tarea: Publicación del servicio WMS.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1 y 3/5
Programador Responsable: Byron Cueva.	
Descripción: Configuración del servicio WMS y publicación de capas principales.	

Tabla 51: Tarea 3.3.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 3.4	Número de Historia: 3
Nombre Tarea: Instalación de Tomcat.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación del servidor de aplicaciones Tomcat.	

Tabla 52: Tarea 3.4.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 3.5	Número de Historia: 3
Nombre Tarea: Instalación y configuración de Geoserver.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación y configuración del servidor de mapas Geosever.	

**Tabla 53: Tarea 3.5.
Fuente: Propia.**

Tarea de Ingeniería	
Número Tarea: 3.6	Número de Historia: 3
Nombre Tarea: Publicación del servicio WFS.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Configuración del servicio WFS y WFS-T; y publicación de capas.	

**Tabla 54: Tarea 3.6.
Fuente: Propia.**

Tarea de Ingeniería	
Número Tarea: 3.7	Número de Historia: 3
Nombre Tarea: Instalación y configuración de Tile Cache.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Instalación y configuración del servidor de teselas en cache Tile Cache.	

**Tabla 55: Tarea 3.7.
Fuente: Propia.**

Tarea de Ingeniería	
Número Tarea: 3.8	Número de Historia: 3
Nombre Tarea: Publicación del servicio WMSC.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Configuración y publicación de capas en WMSC.	

**Tabla 56: Tarea 3.8.
Fuente: Propia.**

- **Historia Nro.: 4**

Tarea de Ingeniería	
Número Tarea: 4.1	Número de Historia: 4
Nombre Tarea: Diseño de la estructura de datos para el control de capas.	
Tipo de Tarea: Diseño.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se diseña el modelo de datos para almacenar la configuración de las capas para que estas puedan ser desplegadas en el visor.	

Tabla 57: Tarea 4.1.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 4.2	Número de Historia: 4
Nombre Tarea: Diseño de la interfaz principal de la aplicación.	
Tipo de Tarea: Diseño.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se diseña la interfaz principal de la aplicación, para la navegación de la información catastral.	

Tabla 58: Tarea 4.2.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 4.3	Número de Historia: 4
Nombre Tarea: Programación de envío de datos de configuración de capas.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se programa las acciones principales para envío de configuración de las capas principales del visor de mapas.	

Tabla 59: Tarea 4.3.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 4.4	Número de Historia: 4
Nombre Tarea: Programación del ambiente de consumo de servicios WMS y WMSC.	

Tipo de Tarea: Diseño.	Puntos Estimados: 4/5
Programador Responsable: Byron Cueva.	
Descripción: Se programa el panel de mapa principal, de consumo de capas de origen WMS y WMSC.	

Tabla 60: Tarea 4.4.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 4.5	Número de Historia: 4
Nombre Tarea: Programación del tree de colección de capas.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se programa el panel tree para el control de encendido y apagado de capas en el visor.	

Tabla 61: Tarea 4.5.
Fuente: Propia.

- **Historia Nro.: 5**

Tarea de Ingeniería	
Número Tarea: 5.1	Número de Historia: 5
Nombre Tarea: Programación de funciones de navegación.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se programa las funciones principales de navegación.	

Tabla 62: Tarea 5.1.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 5.2	Número de Historia: 5
Nombre Tarea: Programación del panel descriptor de leyendas.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Programación del panel para la publicación de leyendas.	

Tabla 63: Tarea 5.2.
Fuente: Propia.

- **Historia Nro.: 6**

Tarea de Ingeniería	
Número Tarea: 6.1	Número de Historia: 6
Nombre Tarea: Diseño de la ventana para agregar capas y grupos de capas.	
Tipo de Tarea: Diseño.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño para la consulta de capas y grupos de capas disponibles, que puedan ser agregadas al visor de mapas.	

Tabla 64: Tarea 6.1.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 6.2	Número de Historia: 6
Nombre Tarea: Programación de la adición y eliminación de capas y grupos de capas.	
Tipo de Tarea: Diseño.	Puntos Estimados: 3/5
Programador Responsable: Byron Cueva.	
Descripción: Programación de las funcionalidades de adición y eliminación de capas, dependiendo de las capas que tenga el usuario permiso de visualizar.	

Tabla 65: Tarea 6.2.

Fuente: Propia.

- **Historia Nro.: 7**

Tarea de Ingeniería	
Número Tarea: 7.1	Número de Historia: 7
Nombre Tarea: Programación de la función de medición de longitud.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Programación de la función que permita realizar la medición de la longitud de una zona señalada en el mapa.	

Tabla 66: Tarea 7.1.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 7.2	Número de Historia: 7

Nombre Tarea: Programación de la función de medición de área.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Programación de la función de medición de área de una zona señalada en el mapa.	

Tabla 67: Tarea 7.2.
Fuente: Propia.

4.1.6.3. Módulo de Consultas y Búsquedas.

- **Historia Nro.: 8**

Tarea de Ingeniería	
Número Tarea: 8.1	Número de Historia: 8
Nombre Tarea: Diseño y programación de ubicación de zonas.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la funcionalidad de búsqueda de zonas, sectores, manzanas y predios a través de la clave catastral.	

Tabla 68: Tarea 8.1.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 8.2	Número de Historia: 8
Nombre Tarea: Diseño y programación de ubicación de calles.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la funcionalidad de búsqueda de calles y cruces de calles.	

Tabla 69: Tarea 8.2.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 8.3	Número de Historia: 8
Nombre Tarea: Diseño y programación de ubicación de lugares principales.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 4/5
Programador Responsable: Byron Cueva.	

Descripción: Diseño y programación de la funcionalidad de búsqueda de lugares principales.

Tabla 70: Tarea 8.3.

Fuente: Propia.

- **Historia Nro.: 9**

Tarea de Ingeniería	
Número Tarea: 9.1	Número de Historia: 9
Nombre Tarea: Configuración de la capa predial que permita solicitudes GetFeaturesInfo.	
Tipo de Tarea: Configuración.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Configuración de la capa predial para el soporte del request GetFeaturesInfo.	

Tabla 71: Tarea 9.1.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 9.2	Número de Historia: 9
Nombre Tarea: Diseño de la ventana de consulta de la información predial.	
Tipo de Tarea: Diseño.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño de la ventana para despliegue de la información del predio.	

Tabla 72: Tarea 9.2.

Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 9.3	Número de Historia: 9
Nombre Tarea: Programación de la función de consulta y carga de datos.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Programación de la función de consulta de datos y fotos del predio.	

Tabla 73: Tarea 9.3.

Fuente: Propia.

- **Historia Nro.: 10**

Tarea de Ingeniería	
Número Tarea: 10.1	Número de Historia: 10

Nombre Tarea: Diseño de la ventana para la búsqueda de un predio.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la ventana para la búsqueda de un predio, por los criterios de clave catastral, cédula y apellidos y nombres del propietario.	

**Tabla 74: Tarea 10.1.
Fuente: Propia.**

Tarea de Ingeniería	
Número Tarea: 10.2	Número de Historia: 10
Nombre Tarea: Programación de la función de dibujo de un predio.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Programación de dibujo de un predio.	

**Tabla 75: Tarea 10.2.
Fuente: Propia.**

- **Historia Nro.: 11**

Tarea de Ingeniería	
Número Tarea: 11.1	Número de Historia: 11
Nombre Tarea: Diseño de la ventana para consulta de la zonificación de un predio.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la ventana para el despliegue de la información de zonificación de un predio.	

**Tabla 76: Tarea 11.1.
Fuente: Propia.**

- **Historia Nro.: 12**

Tarea de Ingeniería	
Número Tarea: 12.1	Número de Historia: 12
Nombre Tarea: Diseño de la ventana para consulta del uso de suelo de un predio.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	

Descripción: Diseño y programación de la ventana para el despliegue de la información del uso de suelo de un predio.

Tabla 77: Tarea 12.1.
Fuente: Propia.

- **Historia Nro.: 13**

Tarea de Ingeniería	
Número Tarea: 13.1	Número de Historia: 13
Nombre Tarea: Diseño de la ventana para consulta de la descripción del terreno.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la ventana para el despliegue de la información de la descripción del terreno.	

Tabla 78: Tarea 13.1.
Fuente: Propia.

- **Historia Nro.: 14**

Tarea de Ingeniería	
Número Tarea: 14.1	Número de Historia: 14
Nombre Tarea: Diseño de la ventana para consulta de la infraestructura y servicios de un predio.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Diseño y programación de la ventana para el despliegue de la información de la infraestructura y servicios que posee un predio.	

Tabla 79: Tarea 14.1.
Fuente: Propia.

- **Historia Nro.: 15**

Tarea de Ingeniería	
Número Tarea: 15.1	Número de Historia: 15
Nombre Tarea: Reporte de las propiedades de un ciudadano.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Con los datos de los predios de un ciudadano se generara un reporte de cuantos y de	

cuales predios es propietario.

Tabla 80: Tarea 15.1.

Fuente: Propia.

- **Historia Nro. 16.**

Tarea de Ingeniería	
Número Tarea: 16.1	Número de Historia: 16
Nombre Tarea: Diseño y Programación de la ventana de consulta del impuesto predial.	
Tipo de Tarea: Diseño y Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Reporte con la información del impuesto predial de un predio.	

Tabla 81: Tarea 16.1.

Fuente: Propia.

- **Historia Nro. 17.**

Tarea de Ingeniería	
Número Tarea: 17.1	Número de Historia: 17
Nombre Tarea: Diseño y Desarrollo de la ventana de consulta de los colindantes.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Se diseña una ventana para la consulta de los predios colindantes.	

Tabla 82: Tarea 17.1.

Fuente: Propia.

- **Historia Nro. 18.**

Tarea de Ingeniería	
Número Tarea: 18.1	Número de Historia: 18
Nombre Tarea: Interfaz para la generación del croquis de un predio.	
Tipo de Tarea: Diseño.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Se diseña la interfaz para la generación del croquis de un predio.	

Tabla 83: Tarea 18.1.

Fuente: Propia.

Tarea de Ingeniería	
----------------------------	--

Número Tarea: 18.2	Número de Historia: 18
Nombre Tarea: Programación del reporte de impresión del croquis.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/5
Programador Responsable: Byron Cueva.	
Descripción: Generación de funcionalidades de impresión del mapa en formato pdf.	

Tabla 84: Tarea 18.2.
Fuente: Propia.

- **Historia Nro. 19.**

Tarea de Ingeniería	
Número Tarea: 19.1	Número de Historia: 19
Nombre Tarea: Generación de una capa dinámica para la consulta del impuesto predial por parroquia.	
Tipo de Tarea: Configuración.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Creación y configuración de una capa dinámica de consulta.	

Tabla 85: Tarea 19.1.
Fuente: Propia.

Tarea de Ingeniería	
Número Tarea: 19.2	Número de Historia: 19
Nombre Tarea: Diseño de la interfaz para la consulta de impuestos prediales por parroquia.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/5
Programador Responsable: Byron Cueva.	
Descripción: Desarrollo de una ventana, para la consulta del impuesto predial.	

Tabla 86: Tarea 19.2.
Fuente: Propia.

4.1.7. Detalle de Historias de Usuario.

El objetivo del presente punto es realizar una lista de las historias de usuario junto a cada una de sus tareas, especificadas por módulos. Así se pretende presentar un marco entendible de las historias de usuario y poder realizar la planificación posteriormente como lo señala la metodología XP.

- **Módulo de Seguridad y Acceso al Sistema.**
 - **Historia de Usuario 1:** Seguridad y Control de Acceso.
 - **Tarea 1:** Diseñar la estructura de datos para el control de Acceso al Sistema.
 - **Tarea 2:** Crear la interfaz para el ingreso al Sistema.
- **Módulo Visor de Mapas.**
 - **Historia de Usuario 2:** Repositorio de datos geográficos.
 - **Tarea 1:** Instalación y configuración del módulo Postgis.
 - **Tarea 2:** Creación de la base de datos y roles de usuario.
 - **Tarea 3:** Recopilación de la información catastral.
 - **Tarea 4:** Migración de la información geográfica.
 - **Historia de Usuario 3:** Publicación de mapas en la web.
 - **Tarea 1:** Instalación de Apache.
 - **Tarea 2:** Instalación y configuración de MapServer.
 - **Tarea 3:** Publicación del Servicio WMS.
 - **Tarea 4:** Instalación de Tomcat.
 - **Tarea 5:** Instalación y configuración de Geoserver.
 - **Tarea 6:** Publicación del servicio WFS.
 - **Tarea 7:** Instalación y configuración de Tile Cache.
 - **Tarea 8:** Publicación del servicio WMSC.
 - **Historia de Usuario 4:** Despliegue de información geográfica y colección de capas.
 - **Tarea 1:** Diseño de la estructura de datos para el control de capas.
 - **Tarea 2:** Diseño de la interfaz principal de la aplicación.
 - **Tarea 3:** Programación de envío de datos de configuración de capas.
 - **Tarea 4:** Programación del ambiente de consumo de servicios WMS y WMSC.
 - **Tarea 5:** Programación del tree de colección de capas.
 - **Historia de Usuario 5:** Funciones de navegación y descriptor de leyendas.
 - **Tarea 1:** Programación de funciones de navegación.
 - **Tarea 2:** Programación del panel descriptor de leyendas.
 - **Historia de Usuario 6:** Adición y eliminación de capas.
 - **Tarea 1:** Diseño de la ventana para agregar capas y grupos de capas.
 - **Tarea 2:** Programación de la adición y eliminación de capas y grupos de capas.
 - **Historia de Usuario 7:** Estimación de longitud y área.
 - **Tarea 1:** Programación de la función de medición de longitud.
 - **Tarea 2:** Programación de la medición de área.
- **Módulo de Consultas y Búsquedas.**
 - **Historia de Usuario 8:** Ubicación rápida de información catastral.
 - **Tarea 1:** Diseño y programación de ubicación de zonas.

- **Tarea 2:** Diseño y programación de ubicación de calles.
- **Tarea 3:** Diseño y programación de ubicación de lugares principales.
- **Historia de Usuario 9:** Consulta de Información predial.
 - **Tarea 1:** Configuración de la capa predial que permita solicitudes GetFeaturesInfo.
 - **Tarea 2:** Diseño de la ventana de consulta de la información predial.
 - **Tarea 3:** Programación de la función de consulta y carga de datos.
- **Historia de Usuario 10:** Búsqueda de un predio.
 - **Tarea 1:** Diseño de la ventana para la búsqueda de un predio.
 - **Tarea 2:** Programación de la función de dibujo de un predio.
- **Historia de Usuario 11:** Consulta de la zonificación de un predio.
 - **Tarea 1:** Diseño de la ventana para consulta de la zonificación de un predio.
- **Historia de Usuario 12:** Consulta de los usos de suelo de un predio.
 - **Tarea 1:** Diseño de la ventana para consulta del uso de suelo de un predio.
- **Historia de Usuario 13:** Consulta de la descripción de un terreno.
 - **Tarea 1:** Diseño de la ventana para consulta de la descripción del terreno.
- **Historia de Usuario 14:** Consulta de la infraestructura y servicios.
 - **Tarea 1:** Diseño de la ventana para consulta de la infraestructura y servicios de un predio.
- **Historia de Usuario 15:** Reporte de las propiedades de un ciudadano.
 - **Tarea 1:** Reporte de las propiedades de un ciudadano.
- **Historia de Usuario 16:** Consulta del impuesto predial.
 - **Tarea 1:** Diseño y programación de la ventana de consulta del impuesto predial.
- **Historia de Usuario 17:** Consulta de predios colindantes.
 - **Tarea 1:** Diseño y Desarrollo de la ventana de consulta de los colindantes,
- **Historia de Usuario 18:** Impresión del Croquis de un predio.
 - **Tarea 1:** Interfaz para la generación del croquis de un predio.
 - **Tarea 2:** Programación del reporte de impresión del croquis.
- **Historia de Usuario 19:** Consulta del pago del impuesto predial por parroquias.
 - **Tarea 1:** Generación de una capa dinámica para la consulta del impuesto predial por parroquia.
 - **Tarea 2:** Diseño de la interfaz para la consulta de impuestos prediales por parroquia.

4.1.8. Tiempo Ideal.

Es el tiempo necesario para elaborar una historia de usuario sin interrupciones, en el cual el desarrollo pueda sentirse completamente productivo.

Es así, como un día ideal de desarrollo corresponde para el equipo a una jornada de trabajo de 2 horas. En este caso se ha propuesto que una semana ideal es aquella que se trabaja de lunes a viernes, 2 horas diarias. La resolución del tiempo indicado se tomó en base al tiempo disponible por el programador.

4.1.9. Estimación de Esfuerzo

A continuación se presenta la estimación del esfuerzo por historia de usuario, tomando en cuenta el tiempo ideal necesario para cada historia de usuario.

- **Módulo de seguridad y control de acceso.**

Nro.	Historias de Usuarios	Puntos
1	Seguridad y Control de Acceso.	1 y 3/5

Tabla 87: Estimación de Esfuerzo, Módulo de Seguridad y Control de Acceso.
Fuente: Propia.

- **Módulo Visor de Mapas.**

Nro.	Historias de Usuario	Puntos
2	Repositorio de datos geográficos.	2 y 2/5
3	Publicación de mapas en la web	3
4	Despliegue de Información geográfica y colección de capas.	2 y 2/5
5	Funciones de navegación y descriptor de leyendas.	4/5
6	Adición y eliminación de capas.	4/5
7	Estimación de longitud y área.	4/5

Tabla 88: Estimación de Esfuerzo, Módulo Visor de Mapas.
Fuente: Propia.

- **Módulo de consultas y Búsquedas.**

Nro.	Historias de Usuario	Puntos
8	Ubicación rápida de información catastral.	1 y 3/5
9	Consulta de información predial.	4/5
10	Búsqueda de un predio.	2/5
11	Consulta de la Zonificación de un predio.	2/5
12	Consulta de los usos de suelo de un predio.	2/5
13	Consulta de la descripción del terreno.	2/5

14	Consulta de la infraestructura y servicios.	2/5
15	Reporte de las propiedades de un ciudadano.	2/5
16	Consulta del Impuesto Predial.	2/5
17	Consulta de los predios colindantes.	2/5
18	Impresión de Croquis de un predio.	2/5
19	Consulta del pago del impuesto predial por parroquias.	4/5

Tabla 89: Estimación de Esfuerzo, Módulo de Consultas y Búsquedas.
Fuente: Propia.

4.1.9.1. Estimaciones de historias de usuario en tiempo ideal.

Módulo	Nro	Historia	Tiempo ideal estimado		
			Semanas	Días	Horas
Seguridad y control de Acceso	1	Seguridad y Control de Acceso.	1.6	8	16
Visor de Mapas	2	Repositorio de datos geográficos	2.4	12	24
	3	Publicación de mapas en la web	3	15	30
	4	Despliegue de Información geográfica y colección de capas	2.4	12	24
	5	Funciones de navegación y descriptor de leyendas	0.8	4	8
	6	Adición y eliminación de capas	0.8	4	8
	7	Estimación de longitud y área	0.8	4	8
Consultas y Búsquedas	8	Ubicación rápida de información catastral	1.6	8	16
	9	Consulta de información predial	0.8	4	8
	10	Búsqueda de un predio	0.4	2	4
	11	Consulta de la Zonificación de un predio	0.4	2	4
	12	Consulta de los usos de suelo de un predio	0.4	2	4
	13	Consulta de la descripción del terreno	0.4	2	4
	14	Consulta de la infraestructura y servicios	0.4	2	4
	15	Reporte de las propiedades de un ciudadano	0.4	2	4
	16	Consulta del Impuesto Predial.	0.4	2	4

	17	Consulta de los predios colindantes.	0.4	2	4
	18	Impresión de Croquis de un predio.	0.4	2	4
	19	Consulta del pago del impuesto predial por parroquias.	0.8	4	8
TOTALES			18.6	93	186

Tabla 90: Estimación de Historias de Usuario en Tiempo Ideal.

Fuente: Propia.

4.2. Fase de Planificación.

La fecha de comienzo para el desarrollo del proyecto es el 14 de Marzo del 2011. En esta fase se detallan el número y orden de las iteraciones necesarias para cumplir con las Historias de Usuario, en las cuales se entregaran diferentes versiones del sistema acordado con el Rastreador (Tracker) del proyecto.

4.2.1. Velocidad del equipo de desarrollo.

En XP, la velocidad de desarrollo se refiere a la rapidez con la que el equipo de desarrollo puede trabajar implementando historias de usuario que han sido estimadas utilizando el tiempo ideal. Cuando se refiere a la velocidad de desarrollo, también se utiliza el término de esfuerzo ideal para denotar el tiempo que le es posible asignar a un desarrollador para implementar una funcionalidad del sistema en un tiempo determinado.

A continuación se muestra como se calcula la velocidad del equipo.

Programadores en el equipo: 1 persona.

Velocidad del equipo al mes (Esfuerzo ideal):

$$1 \text{ persona} \times \frac{4 \text{ semanas}}{1 \text{ persona}} = 4 \text{ semanas al mes}$$

Con esto se concluye que para el caso del presente proyecto, el equipo de desarrollo puede desarrollar 4 semanas ideales de desarrollo por cada mes de trabajo.

4.2.2. Iteración por cada entrega.

Una vez establecida la velocidad del equipo de desarrollo, se determina el número de iteraciones que se llevaran a cabo para la entrega del software.

En la sección 5.2.2., se obtuvo el total de semanas ideales (18.6 semanas) que serían necesarias para implementar la solución propuesta por el equipo de desarrollo. Tomando en cuenta este total, se procede a calcular el número de iteraciones de la siguiente manera.

Semanas ideales totales para implementar la solución: 18.6 semanas.

Velocidad del equipo de desarrollo al mes: 4 semanas.

Número de iteraciones para la entrega: $\frac{18.6 \text{ semanas}}{4 \text{ semanas}} = 4.65 \text{ iteraciones}$

Cabe señalar que una iteración corresponde para nuestro caso un mes ideal, en el cual se trabaja de lunes a viernes, 2 horas diarias y con 1 programador a cargo del desarrollo.

Con este resultado podemos decir que como mínimo, tendremos 5 iteraciones (redondeando a su inmediato superior), para implementar la solución.

4.2.3. Planificación por Historias.

En esta sección se escogen las historias de usuario que serán implementadas en cada iteración, y además se determina las fechas de inicio y fin para la ejecución de las mismas.

En este caso existen cuatro iteraciones, las mismas que tienen asignados a cada una de ellas, un grupo de historias de usuario de tal manera que se pueda entregar el valor al cliente en el menor tiempo posible.

Iteraciones	Nro	Historias	Inicio	Fin	Observaciones
Primera	1	Seguridad y Control de Acceso	14/03/2011	23/03/2011	
	2	Repositorio de datos geográficos	24/03/2011	08/04/2011	
Segunda	3	Publicación de mapas en la web	11/04/2011	29/04/2011	
	4	Despliegue de Información geográfica y colección de capas	02/05/2011	17/05/2011	
Tercera	5	Funciones de navegación y descriptor de leyendas	18/05/2011	23/05/2011	
	6	Adición y eliminación de capas	24/05/2011	27/05/2011	
	7	Estimación de longitud y área	30/05/2011	02/06/2011	
	8	Ubicación rápida de información catastral	03/06/2011	14/06/2011	
	9	Consulta de información predial	15/06/2011	20/06/2011	
Cuarta	10	Búsqueda de un predio	21/06/2011	22/06/2011	

	11	Consulta de la Zonificación de un predio	23/06/2011	24/06/2011	
	12	Consulta de los usos de suelo de un predio	27/06/2011	28/06/2011	
	13	Consulta de la descripción del terreno	29/06/2011	30/06/2011	
	14	Consulta de la infraestructura y servicios	01/07/2011	04/07/2011	
	15	Reporte de las propiedades de un ciudadano	05/07/2011	06/07/2011	
Quinta	16	Consulta del Impuesto Predial.	07/07/2011	08/07/2011	
	17	Consulta de los predios colindantes.	11/07/2011	12/07/2011	
	18	Impresión de Croquis de un predio.	13/07/2011	14/07/2011	
	19	Consulta del pago del impuesto predial por parroquias.	15/07/2011	20/07/2011	

Tabla 91: Planificación por Historias.

Fuente: Propia.

4.2.4. Plan de Entregas

Para la elaboración del plan de entrega, he tomado en cuenta que existirán cuatro iteraciones y que además se tiene previsto entregar todas las historias de usuario en la primera entrega del producto de software, siendo esta el alcance del proyecto.

Nro.	Nombre	Comienzo	Fin	Duración	Marzo			Abril		Mayo		Junio		Julio		
					01	11	14	08	11	17	18	20	21	06	07	20
1	Exploración	01/03/2011	11/03/2011	2 s.												
2	Iteración 1	14/03/2011	08/04/2011	4 s.												
3	Iteración 2	11/04/2011	17/05/2011	5 s.												
4	Iteración 3	18/05/2011	20/06/2011	5 s.												
5	Iteración 4	21/06/2011	06/07/2011	3 s.												
6	Iteración 5	07/07/2011	20/07/2011	3 s.												

Tabla 92: Plan de Entregas.
Fuente: Propia.

4.2.5. Plan de Iteración.

Para la planificación de las iteraciones, se debe planificar las tareas de ejecución de cada historia de usuario además de determinar las fechas de inicio y fin para ejecución de las mismas.

4.2.5.1. Iteración 1.

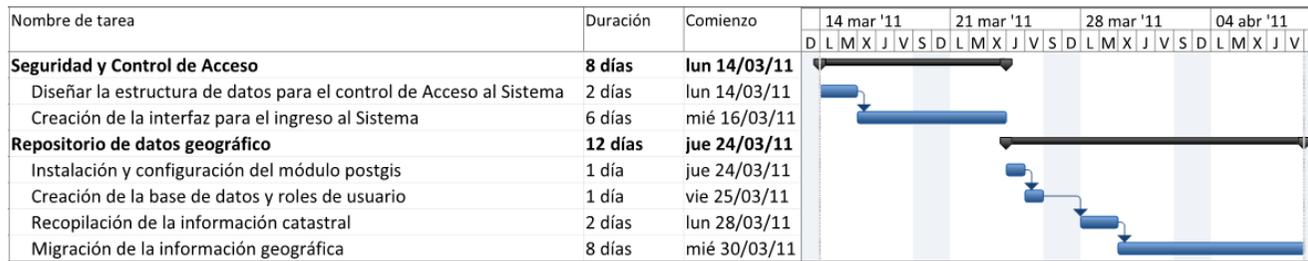


Figura 30: Plan de Iteración 1.
Fuente: Propia.

4.2.5.2. Iteración 2.

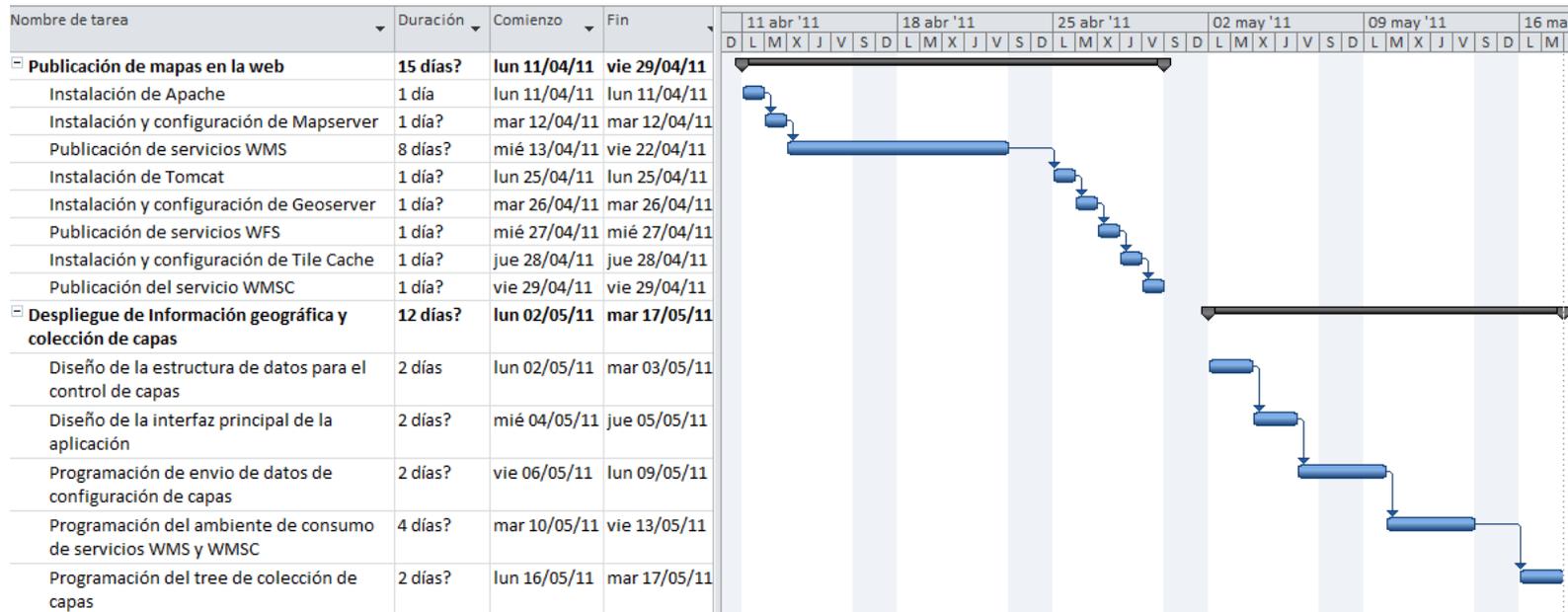


Figura 31: Plan de Iteración 2.
Fuente: Propia.

4.2.5.3. Iteración 3.

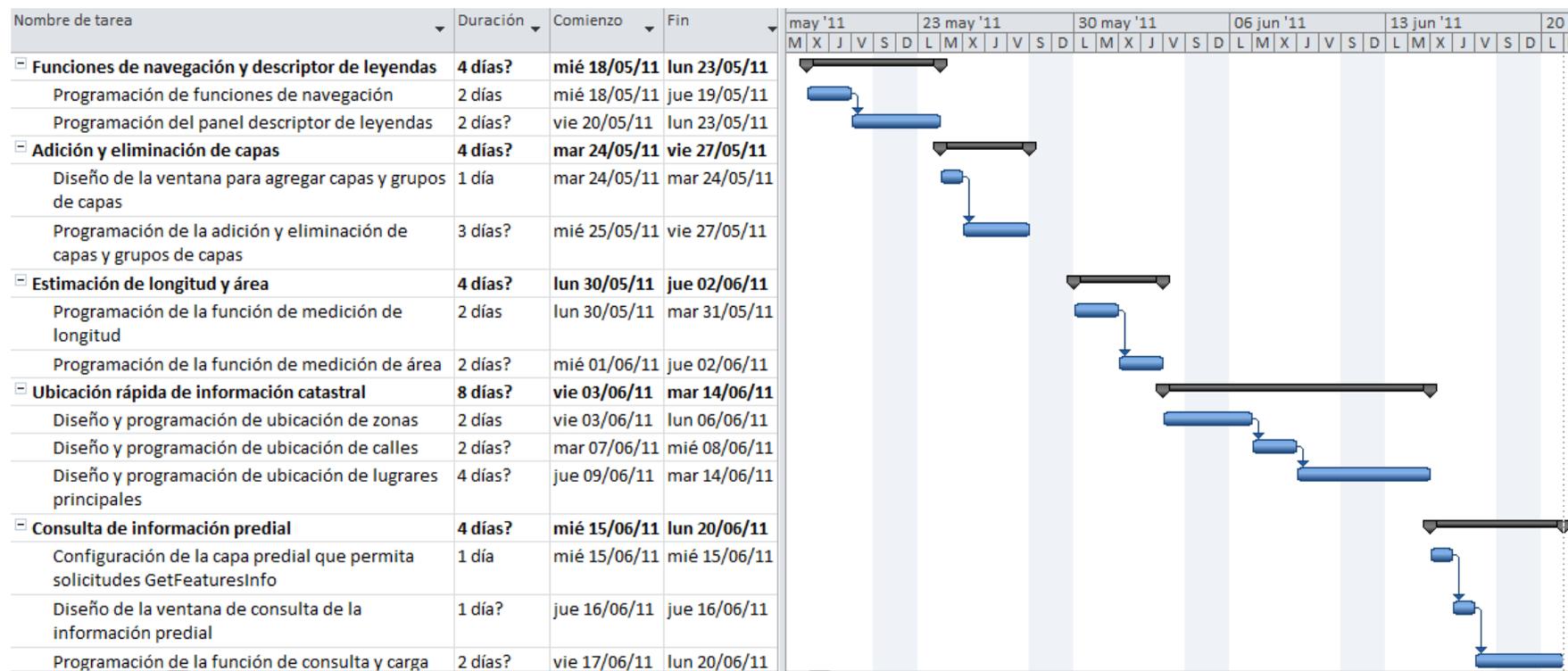


Figura 32: Plan de Iteración 3.

Fuente: Propia.

4.2.5.4. Iteración 4.

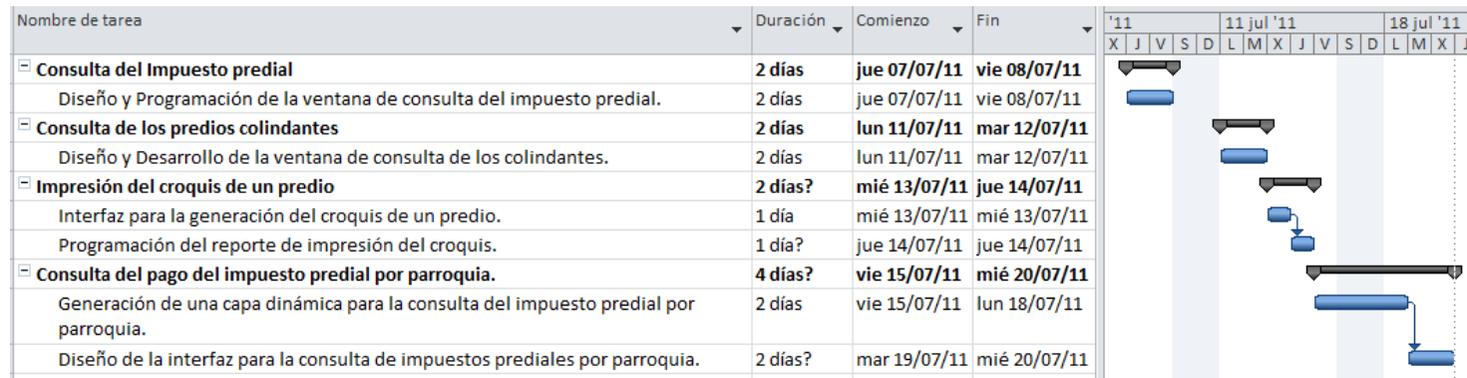


Figura 33: Plan de Iteración 4.
Fuente: Propia.

4.2.5.5. Iteración 5.

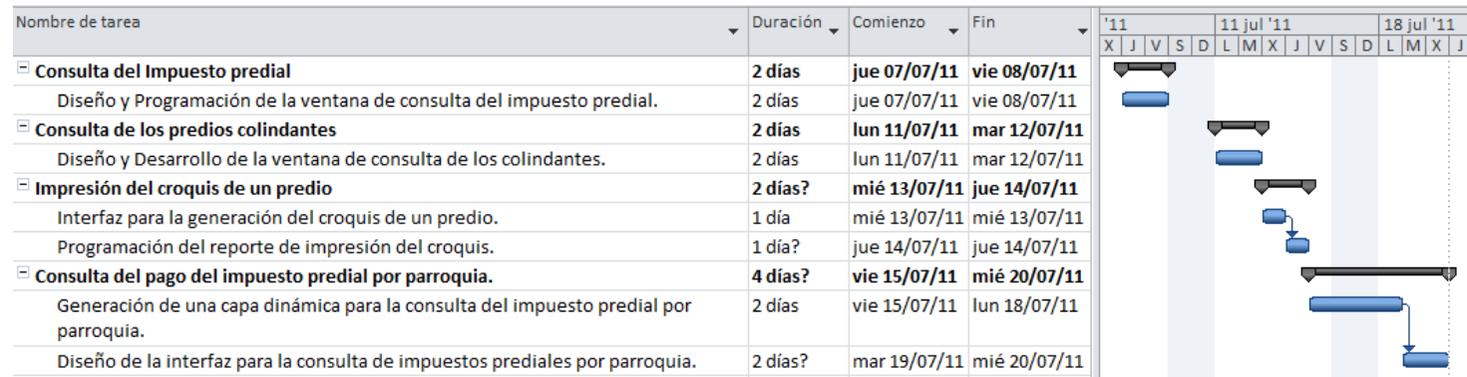


Figura 34: Plan de Iteración 5.
Fuente: Propia.

4.3. Fase de Iteración.

Una vez que se ha concluido con la planificación de las iteraciones que conformará la primera entrega del sistema, el equipo de desarrollo tiene en claro cuáles son los tiempos estimados para implementar las historias de usuario, las tareas correspondientes, y las dependencias que existen entre cada una de ellas.

Ahora bien, en esta sección, se pretende explicar el proceso que adoptó el equipo de desarrollo para implementar cada historia de usuario correspondiente a la entrega del sistema como tal. Es decir se explicará la ejecución de las tareas necesarias para complementar cada funcionalidad del sistema propuesto.

Cabe señalar que dada la circunstancia que la planificación inicial sirve únicamente como punto de partida para la implementación del sistema, la planificación real puede presentar algún cambio en las fechas de inicio y de fin durante la ejecución.

Se puede resumir que las actividades ejecutadas dentro de una iteración en XP son:

- Diseñar las pruebas de aceptación.
- Programar las historias de usuario.
- Refactorización de código y el diseño (cuando sea necesario).
- Ejecutar las pruebas de aceptación.

4.3.1. Diseño de pruebas de Aceptación.

El diseño de las pruebas de aceptación hace referencia a las prácticas de “Pruebas” y “Diseño Simple”, sugiriendo que antes de programar se deben diseñar pruebas concisas que garanticen el cumplimiento de los requerimientos solicitados por el cliente en cada historia de usuario pasando por un proceso de verificación, además esto permitirá prevenir errores de programación desde un principio.

En XP, el encargado de pruebas (tester) conjuntamente con el cliente son los encargados de diseñar las pruebas de aceptación.

A continuación se detallan las pruebas de aceptación ordenadas por módulos.

4.3.1.1. Módulo de Seguridad y Acceso al Sistema.

Caso de Prueba de Aceptación	
Código: 1.1	Historia de Usuario (Nro. y Nombre): 1 - Seguridad y Control de Acceso
Nombre: Caso de prueba para la historia; Seguridad y Control de Acceso al Sistema.	

<p>Descripción: Al momento de ingresar al sistema, espero observar la solicitud del usuario y la clave, para luego aparecerle el panel principal del sistema con las opciones a las cuales tiene permisos el usuario.</p>
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> • El nombre de usuario será escrito en mayúsculas y contendrá el primer apellido, seguido de la primera letra del primer nombre, ejemplo: Luis Maldonado, su usuario sería MALDONADOL.
<p>Entrada / Pasos de ejecución:</p> <ul style="list-style-type: none"> • Introducir el nombre de usuario. • Introducir la clave. • Clic en Ingresar.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • El sistema verifica si el usuario existe y si los datos ingresados son correctos. <p>Resultado 1:</p> <ul style="list-style-type: none"> • Si el ingreso es correcto, se consulta los permisos y se configura la aplicación para el ingreso. <p>Resultado 2:</p> <ul style="list-style-type: none"> • Si los datos son incorrectos, se informa al usuario que sus datos son incorrectos y no se permite el acceso.

Tabla 93: Casos de Prueba de Aceptación 1.1 de Historia Nro. 1.
Fuente: Propia.

4.3.1.2. Módulo Visor de Mapas.

Caso de Prueba de Aceptación	
Código: 2.1	Historia de Usuario (Nro. y Nombre): 2 – Repositorio de datos geográficos.
Nombre: Caso de prueba de la historia; Repositorio de datos geográficos.	
Descripción: La información geográfica de Catastros y Avalúos se encuentra en una base de datos centralizada.	
<p>Condiciones:</p> <ul style="list-style-type: none"> • El campo de geometría para cada capa debe llevar el nombre de “the_geom”, como un estándar. • Todas las capas deben estar proyectadas en el sistema de coordenadas PSAD 56 zona 17 Sur (EPSG 24877). 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Base de datos Geográfica. 	

<ul style="list-style-type: none"> • Esquema de Catastros. • Roles de usuarios para administrador, editor y consultor. • Capas migradas: parroquias, manzanas, vías, predial, propiedad horizontal, primer piso, segundo piso, tercer piso, cuarto piso y lugares principales (farmacias, hospitales, parques, etc.).
--

Tabla 94: Casos de Prueba de Aceptación 2.1 de Historia Nro. 2.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 3.1	Historia de Usuario (Nro. y Nombre): 3 – Publicación de mapas en la web.
Nombre: Caso de prueba para la historia; Publicación de mapas en la web.	
Descripción: La información geográfica permitida a ser publicación se encuentra disponible en un OWS según las necesidades (WMS, WFS o WMSC).	
Condiciones:	
<ul style="list-style-type: none"> • Todas las capas deben estar proyectadas en el sistema de coordenadas PSAD 56 zona 17 Sur (EPSG 24877). 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Servicio de mapas WMS, WFS y WMSC, escuchando solicitudes. • Capas publicadas: parroquias, manzanas, vías, predial, propiedad horizontal, primer piso, segundo piso, tercer piso, cuarto piso y lugares principales (farmacias, hospitales, parques, etc.). 	

Tabla 95: Casos de Prueba de Aceptación 3.1 de Historia Nro. 3.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 4.1	Historia de Usuario (Nro. y Nombre): 4 – Despliegue de Información geográfica y colección de capas.
Nombre: Caso de prueba para la historia; Despliegue de información geográfica y colección de capas.	
Descripción: Cuando se obtenga el acceso al sistema, espero ver un visor de navegación principal y una colección de capas con la información de las capas publicadas y permitidas a ser observadas por el usuario.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • El usuario debe haberse logeado primeramente para tener acceso al sistema. 	
Entrada / Pasos de ejecución:	

<ul style="list-style-type: none"> • Obtener acceso al sistema.
Resultado Esperado: <ul style="list-style-type: none"> • Visor de navegación y colección de capas desplegadas.

Tabla 96: Casos de Prueba de Aceptación 4.1 de Historia Nro. 4.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 5.1	Historia de Usuario (Nro. y Nombre): 5 – Funciones de navegación y descriptor de leyendas.
Nombre: Caso de prueba para la historia; Funciones de navegación y descriptor de leyendas.	
Descripción: Cuando se visualice el visor de mapas, espero contar con un conjunto de herramientas que permitan la navegación sobre el mismo, además un descriptor de la simbología de las capas que se encuentran visualizando.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • La visualización de la leyenda se desplegara dependiendo de la configuración de la capa, en la escala que se encuentre la misma. 	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Obtener acceso al sistema. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Herramientas de navegación: zoom al mapa, aumentar y disminuir zoom, etc. • Panel de simbología. 	

Tabla 97: Casos de Prueba de Aceptación 5.1 de Historia Nro. 5.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 6.1	Historia de Usuario (Nro. y Nombre): 6 – Adición y eliminación de capas.
Nombre: Caso de prueba para la Adición de capas.	
Descripción: Cuando se agregue o elimine una capa, espero que el visor permita reflejar rápidamente ese cambio.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Si el usuario no cuenta con permisos de visualización de una capa esta no se debe permitir agregar al visor de mapas. 	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Dar clic en la opción de adición de capas. 	

<ul style="list-style-type: none"> • Seleccionar la capa o grupo de capas a agregar. • Aceptar.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Se añade la nueva capa al panel de colección de capas. • La capa agregada se encuentra visible en el visor.

Tabla 98: Casos de Prueba de Aceptación 6.1 de Historia Nro. 6.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 6.2	Historia de Usuario (Nro. y Nombre): 6 – Adición y eliminación de capas.
Nombre: Caso de prueba para la Eliminación de capas.	
Descripción: Cuando se agregue o elimine una capa, espero que el visor permita reflejar rápidamente ese cambio.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • Si el usuario no cuenta con permisos de visualización de una capa esta no se debe permitir agregar al visor de mapas. 	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Seleccionar la capa o grupo, del panel de colección de capas. • Clic en la opción eliminar. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • La capa eliminada se pierde del panel de colección de capas. • La capa eliminada no se visualiza en el visor de mapas. 	

Tabla 99: Casos de Prueba de Aceptación 6.2 de Historia Nro. 6.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 7.1	Historia de Usuario (Nro. y Nombre): 7 – Estimación de longitud y área.
Nombre: Caso de prueba para la estimación de longitud.	
Descripción: Cuando seleccione las opciones de longitud, espero me permita medir la longitud de esta manera obtener una estimación de la misma.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Clic opción longitud. • Clic sobre el mapa y extender en el área que se desee medir. 	

Resultado Esperado:

- Medida de la longitud en unidad metros.

Tabla 100: Casos de Prueba de Aceptación 7.1 de Historia Nro. 7.**Fuente: Propia.**

Caso de Prueba de Aceptación	
Código: 7.2	Historia de Usuario (Nro. y Nombre): 7 – Estimación de longitud y área.
Nombre: Caso de prueba para la estimación del área.	
Descripción: Cuando seleccione las opciones de área, espero me permita medir el área y de esta manera obtener una estimación.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Clic opción área. • Clic sobre el mapa y extender en el área que se desee medir. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Medida del área en unidad metros cuadrados. 	

Tabla 101: Casos de Prueba de Aceptación 7.2 de Historia Nro. 7.**Fuente: Propia.****4.3.1.3. Módulo de Consultas y Búsquedas.**

Caso de Prueba de Aceptación	
Código: 8.1	Historia de Usuario (Nro. y Nombre): 8 – Ubicación rápida de información catastral.
Nombre: Caso de prueba para la ubicación rápida de información catastral por clave catastral.	
Descripción: Cuando realice una búsqueda por clave catastral, espero que el mapa se ubique en la zona, manzana, sector o predio.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar la clave catastral. • Clic buscar. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Ubicación de la zona o lugar en el mapa, dependiendo de la clave catastral. 	

Tabla 102: Casos de Prueba de Aceptación 8.1 de Historia Nro. 8.**Fuente: Propia.**

Caso de Prueba de Aceptación	
Código: 8.2	Historia de Usuario (Nro. y Nombre): 8 – Ubicación rápida de información catastral.
Nombre: Caso de prueba para la ubicación rápida de información catastral por calles.	
Descripción: Cuando realice una búsqueda por calles, espero que el mapa se dirija a la calle o el cruce de las calles ingresadas.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar calle uno. • Ingresar calle dos si es necesario. • Clic buscar. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Ubicación del Cruce de calles o calle ingresada. 	

Tabla 103: Casos de Prueba de Aceptación 8.2 de Historia Nro. 8.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 8.3	Historia de Usuario (Nro. y Nombre): 8 – Ubicación rápida de información catastral.
Nombre: Caso de prueba para la ubicación rápida de información catastral por lugares principales.	
Descripción: Cuando realice una búsqueda por lugares principales (parques, hospitales, etc.), espero que el mapa muestre la ubicación del lugar seleccionado.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Seleccionar el tipo de lugar (parque, farmacia, etc.). • Clic buscar. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Colección de los lugares (parques, farmacias, etc.). • Ubicación en el mapa del lugar seleccionado. 	

Tabla 104: Casos de Prueba de Aceptación 8.3 de Historia Nro. 8.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 9.1	Historia de Usuario (Nro. y Nombre): 9 – Consulta de información predial.

Nombre: Caso de prueba para la historia; Consulta de información predial.
Descripción: Cuando consulte la información de un predio, quiero elegir el predio a obtener la información en el visor de mapas, esperando obtener la información del propietario, valor del terreno y construcción; área del terreno, ubicación y el historial de fotos del predio.
Condiciones de Ejecución: <ul style="list-style-type: none"> • Un usuario puede realizar la consulta siempre y cuando tenga permiso de este tipo de consulta y permiso de consulta sobre la capa. • Debe permitir tener diferentes tipos de respuesta de información, ya que existe información que no debe ser visible asía todos los usuarios. • El usuario de consulta en este caso es un usuario del departamento de Catastros y Avalúos.
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Clic en la opción información. • Seleccionar la capa predial. • Clic sobre el predio.
Resultado Esperado: Como usuario de avalúos y catastros. <ul style="list-style-type: none"> • Información del propietario, valor del terreno y construcción; área del terreno, ubicación (barrio y calles) y el historial de fotos del predio. Otros usuario <ul style="list-style-type: none"> • Información solicitada o permitida al usuario.

Tabla 105: Casos de Prueba de Aceptación 9.1 de Historia Nro. 9.

Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 10.1	Historia de Usuario (Nro. y Nombre): 10 – Búsqueda de un predio.
Nombre: Caso de prueba para la historia; Búsqueda de un predio.	
Descripción: Cuando busque un predio, quiero buscarlo por clave catastral, cédula o apellidos y nombres del propietario; y espero tener una lista de coincidencias en la búsqueda para luego ubicarme en el mapa al elegir uno de los predios como resultado.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Seleccionar el criterio de consulta (clave catastral, cédula o nombres). • Ingresar la información a buscar. 	

<ul style="list-style-type: none"> • Clic en buscar.
<p>Resultado Esperado:</p> <p>Resultado 1 (Predio Dibujado):</p> <ul style="list-style-type: none"> • Información de todos los predios que cumplieron con la búsqueda. • Ubicación del predio en el mapa. <p>Resultado 2 (Predio no Dibujado):</p> <ul style="list-style-type: none"> • Información de todos los predios que cumplieron con la búsqueda. • Mensaje indicando que el predio se encuentra en actualización.

Tabla 106: Casos de Prueba de Aceptación 10.1 de Historia Nro. 10.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 11.1	Historia de Usuario (Nro. y Nombre): 11 – Consulta de la zonificación de un predio.
Nombre: Caso de prueba para la historia; Consulta de la zonificación de un predio.	
Descripción: Cuando consulte la zonificación de un predio, espero obtener información de la zonificación de la zona que cubre al predio de acuerdo a la reglamentación urbana.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Seleccionar la opción de consulta de zonificación. • Clic sobre el predio a consultar. 	
Resultado Esperado:	
Resultado 1 (En zona de Zonificación):	
<ul style="list-style-type: none"> • Información de la zonificación de la zona que cubre al predio. 	
Resultado 2 (No en zona de Zonificación):	
<ul style="list-style-type: none"> • Mensaje indicando que el predio no cuenta con esta información. 	

Tabla 107: Casos de Prueba de Aceptación 11.1 de Historia Nro. 11.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 12.1	Historia de Usuario (Nro. y Nombre): 12 – Consulta de los usos de suelo de un predio.
Nombre: Caso de prueba para la historia; Consulta de los usos de suelo de un predio.	
Descripción: Cuando consulte los usos de suelo de un predio, espero obtener la información del uso del suelo de la zona que cubre al predio, información de acuerdo a la reglamentación urbana.	

Condiciones de Ejecución:
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Seleccionar la opción de consulta de uso de suelo. • Clic sobre el predio a consultar.
Resultado Esperado: Resultado 1 (En zona de Zonificación): <ul style="list-style-type: none"> • Información del uso de suelo de la zona que cubre al predio. Resultado 2 (No en zona de Zonificación): <ul style="list-style-type: none"> • Mensaje indicando que el predio no cuenta con esta información.

**Tabla 108: Casos de Prueba de Aceptación 12.1 de Historia Nro. 12.
Fuente: Propia.**

Caso de Prueba de Aceptación	
Código: 13.1	Historia de Usuario (Nro. y Nombre): 13 – Consulta de la descripción del terreno.
Nombre: Caso de prueba para la historia; Consulta de la descripción del terreno.	
Descripción: Cuando consulte la descripción de un terreno, espero información de.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Seleccionar la opción de consulta de la descripción de un terreno. • Clic sobre el predio a consultar. 	
Resultado Esperado: <ul style="list-style-type: none"> • Información de la descripción del terreno. 	

**Tabla 109: Casos de Prueba de Aceptación 13.1 de Historia Nro. 13.
Fuente: Propia.**

Caso de Prueba de Aceptación	
Código: 14.1	Historia de Usuario (Nro. y Nombre): 14 – Consulta de la infraestructura y servicios.
Nombre: Caso de prueba para la historia; Consulta de la infraestructura y servicios.	
Descripción: Cuando consulte la infraestructura y servicios de un predio, espero la información de.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Seleccionar la opción de consulta de la infraestructura y servicios. 	

<ul style="list-style-type: none"> • Clic sobre el predio a consultar.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Información de la infraestructura y servicios de un predio.

Tabla 110: Casos de Prueba de Aceptación 14.1 de Historia Nro. 14.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 15.1	Historia de Usuario (Nro. y Nombre): 15 – Reporte de las propiedades de un ciudadano.
Nombre: Caso de prueba para la historia; Reporte de las propiedades de un ciudadano.	
Descripción: Cuando se despliegue el reporte, espero la información principal de todos los predios que la persona a consultar sea propietaria.	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
<ul style="list-style-type: none"> • Seleccionar el criterio de búsqueda. • Ingresar la información a buscar. • Elegir ciudadano. • Clic en imprimir. 	
Resultado Esperado:	
<ul style="list-style-type: none"> • Archivo pdf con la información principal de los predios que le pertenecen al ciudadano a buscar. 	

Tabla 111: Casos de Prueba de Aceptación 15.1 de Historia Nro. 15.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 16.1	Historia de Usuario (Nro. y Nombre): 16 – Consulta del Impuesto Predial.
Nombre: Caso de prueba para la historia; Consulta del Impuesto Predial.	
Descripción: Cuando realice una consulta del impuesto predial, espero se despliegue la información de los impuestos pagados y no pagados, con sus respectivas fechas, así como también el valor y el título.	
Condiciones de Ejecución:	
El usuario debe tener permisos para realizar esta consulta.	
Entrada / Pasos de ejecución:	

<ul style="list-style-type: none"> • Seleccionar la opción consulta de impuesto predial. • Dar clic sobre el predio. • Clic pdf.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Archivo pdf con la información del impuesto predial de los años transcurridos.

Tabla 112: Casos de Prueba de Aceptación 16.1 de Historia Nro. 16.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 17.1	Historia de Usuario (Nro. y Nombre): 17 – Consulta de los predios colindantes.
Nombre: Caso de prueba para la historia; Consulta de los predios colindantes.	
Descripción: Cuando realice una consulta de los predios colindantes, espero un listado de los colindantes, a su vez que estos se puedan identificar en el mapa con un color diferente.	
Condiciones de Ejecución: El usuario debe tener permisos para realizar esta consulta.	
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • Seleccionar la opción consulta de los predios colindantes. • Dar clic sobre el predio. 	
Resultado Esperado: <ul style="list-style-type: none"> • Una tabla con la información de los colindantes asía el predio de consulta. • Despliegue de predios colindantes con un color característico, para diferenciarlos del resto de predios. 	

Tabla 113: Casos de Prueba de Aceptación 17.1 de Historia Nro. 17.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 18.1	Historia de Usuario (Nro. y Nombre): 18 – Impresión del Croquis de un predio.
Nombre: Caso de prueba para la historia; Impresión del Croquis de un predio.	
Descripción: Cuando realice la impresión del croquis de un predio, espero que me permita elegir el zoom correcto.	
Condiciones de Ejecución: El usuario debe tener permisos para realizar esta consulta.	

<p>Entrada / Pasos de ejecución:</p> <ul style="list-style-type: none"> • Seleccionar la opción impresión del Croquis de un predio. • Dar clic sobre el predio. • Seleccionar el zoom deseado. • Digitar un nombre. • Digitar una descripción a la impresión. • Clic en pdf o imprimir.
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Un documento en formato pdf con el dibujo del croquis, nombre y descripción ingresada.

Tabla 114: Casos de Prueba de Aceptación 18.1 de Historia Nro. 18.
Fuente: Propia.

Caso de Prueba de Aceptación	
Código: 19.1	Historia de Usuario (Nro. y Nombre): 19 – Consulta del pago del impuesto predial por parroquias.
Nombre: Caso de prueba para la historia; Consulta del pago del impuesto predial por parroquias.	
Descripción: Cuando realice la consulta del pago del impuesto por parroquias, espero me despliegue los predios que han pagado o no con diferentes colores.	
<p>Condiciones de Ejecución:</p> <p>El usuario debe tener permisos para realizar esta consulta. Solo se pueden imprimir datos de hasta 5 años atrás.</p>	
<p>Entrada / Pasos de ejecución:</p> <ul style="list-style-type: none"> • Seleccionar la opción de consulta del pago del impuesto predial por parroquias. • Seleccionar la parroquia. • Seleccionar el año. • Clic en consultar. 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> • Ubicación de la parroquia. • Impresión de los predios que han pagado en color azul y no en color rojo. 	

Tabla 115: Casos de Prueba de Aceptación 19.1 de Historia Nro. 19.
Fuente: Propia.

4.3.2. Iteraciones.

4.3.2.1. Primera Iteración.

- **Módulo de Seguridad y Acceso al Sistema.**
 - **Historia de Usuario 1:** Seguridad y Control de Acceso.
 - **Tarea 1:** Diseñar la estructura de datos para el control de Acceso al Sistema.

Modelo Relacional:

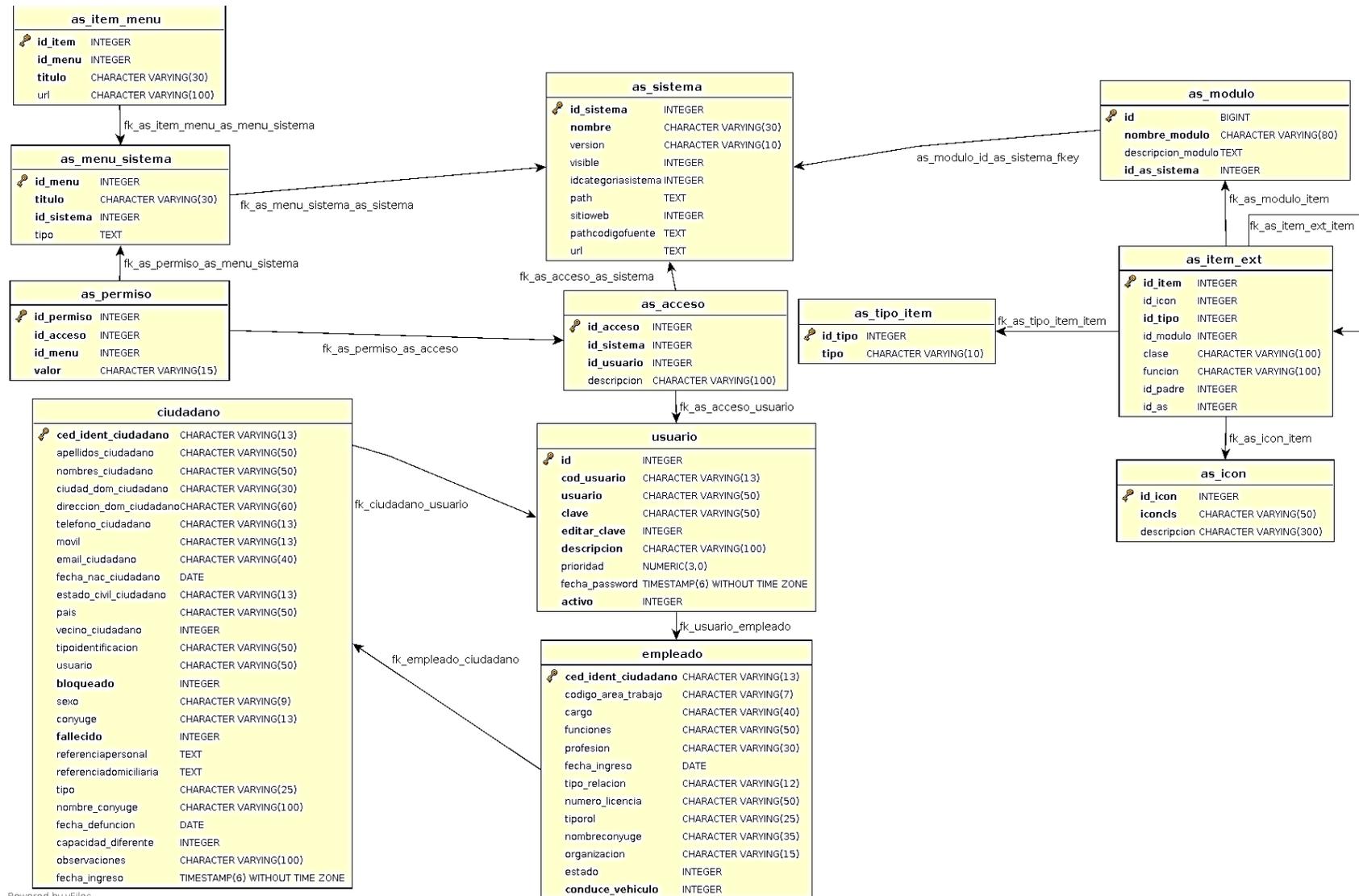


Figura 35: Diagrama Relacional para el control de acceso al sistema.
Fuente: Propia.

- **Tarea 2:** Crear la interfaz para el ingreso al Sistema.



Figura 36: Interfaz para el ingreso al sistema.
Fuente: Propia.

Tarjetas CRC:

Tarjeta CRC	
Nombre de la Clase: BasesfGuardAuthActions	
Descripción: Clase Controlador, permite el control de ingreso al sistema.	
Colaboradores	
Super Clase: sfActions (Clase de Symfony, es un controlador genérico).	
Requiere:	
Responsabilidades	
Métodos	
Nombre	Descripción
executeSignin	Permite el control de ingreso al sistema, si el usuario no se ha logueado se desplegara la pantalla de login, caso contrario permitirá el paso al sistema.
executeSignout	Permite salir del sistema y a su vez termina la sesión del usuario.
executeSecure	Permite el control de seguridad cuando un usuario intenta ingresar a una url del sistema sin haber pasado por la pantalla de ingreso (login).

Tabla 116: Tarjeta CRC: BasesfGuardAuthActions.
Fuente: Propia.

Esto instalara el motor de base de datos y un cliente para conectarse a la misma.

2. Cambiar la clave del usuario postgres del sistema, realizamos lo siguiente:
sudo passwd postgres
Seguidamente ingresamos la contraseña dos veces.
3. Cambiamos la contraseña del administrador de base de datos:
\$sudo su postgres
\$psql
\$alter user postgres with password 'su_password';
Si la contraseña se cambia correctamente devolverá el mensaje ALTER ROLE y luego para salir digitar:
\$q
4. Por defecto al instalar postgresql las conexiones se encuentran bloqueadas. Se debe editar el archivo postgresql.conf
sudo gedit /etc/postgresql/8.4/main/postgresql.conf
Buscar la línea:
#listen_addresses='localhost'
cambiarla por:
listen_addresses='*'
También buscar la línea:
#password_encryption=on
cambiarla por:
password_encryption=on
5. Ahora editamos el archivo pg_hba.conf:
sudo gedit /etc/postgresql/8.4/main/pg_hba.conf
Buscar las líneas:
"local" is for Unix domain socket connections only
local all all ident
Cambiarlas por:
"local" is for Unix domain socket connections only
local all all md5
6. Para que los cambios tengan efecto se debe reiniciar el servicio:
sudo /etc/init.d/postgresql-8.4 restart

Instalación y configuración del módulo PostGIS:

1. En una terminal digitamos:
sudo apt-get install postgis postgresql-8.4-postgis
2. Creamos un template para simplificar la creación de bases de datos espaciales:

Ingresamos con la cuenta postgres:

```
sudo su postgres.
```

Creamos una base de datos normal de PostgreSQL:

```
createdb -E UTF8 template_postgis
```

3. Habilitamos el lenguaje PL/SQL para la nueva base de datos, ya que la mayoría de funciones de PostGIS se escriben en este lenguaje:

```
createlang -d template_postgis plpgsql
```

4. Indicamos que es un template.

```
psql -d postgres -c "UPDATE pg_database SET datistemplate='true' WHERE datname='template_postgis';"
```

5. Cargamos las definiciones de objetos y funciones de PostGIS.

```
psql -d template_postgis -f /usr/share/postgresql/8.4/contrib/postgis-1.5/postgis.sql
```

```
psql -d template_postgis -f /usr/share/postgresql/8.4/contrib/postgis-1.5/spatial_ref_sys.sql
```

```
psql -d template_postgis -f /usr/share/postgresql/8.4/contrib/postgis_comments.sql
```

6. Damos permisos al usuario public, para que el template se encuentre accesible para cualquier usuario:

```
psql -d template_postgis -c "GRANT ALL ON spatial_ref_sys TO PUBLIC;"
```

- **Tarea 2:** Creación de la base de datos y roles de usuario.

Creación de la Base de datos geográfica:

En una terminal

```
$sudo su postgres
```

```
$createdb -U postgres -T template_postgis my_spatial_db
```

Creación de los roles de usuario:

- **Tarea 3:** Recopilación de la información catastral.

Se recopiló la información catastral en formato shape, con la ayuda de un representante de la dirección de Catastros y Avalúos.

- **Tarea 4:** Migración de la información geográfica.

Instalación del cliente pesado gvSIG:

1. Descargamos el paquete de gvSIG, la última versión desde el sitio oficial de GvSIG.
2. En una terminal, asignamos permisos de ejecución al paquete:
`Chmod +x gvSIG-1_11-linux-i586-withjre.bin`
3. En un terminal ejecutamos el paquete:
`./gvSIG-1_11-linux-i586-withjre.bin`
4. Nos pregunta si deseamos instalar el JDK, clic en continuar:

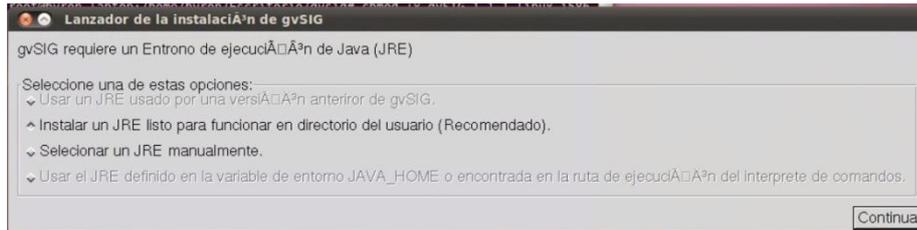


Figura 38: Instalación GvSIG – Instalación JDK.
Fuente: Propia.

5. Escogemos el idioma, y luego OK.

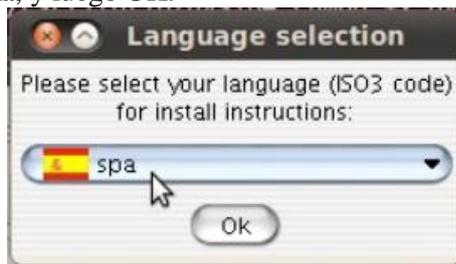


Figura 39: Instalación GvSIG – Selección del Lenguaje.
Fuente: Propia.

6. Nos despliega una información general del software, damos clic en siguiente:

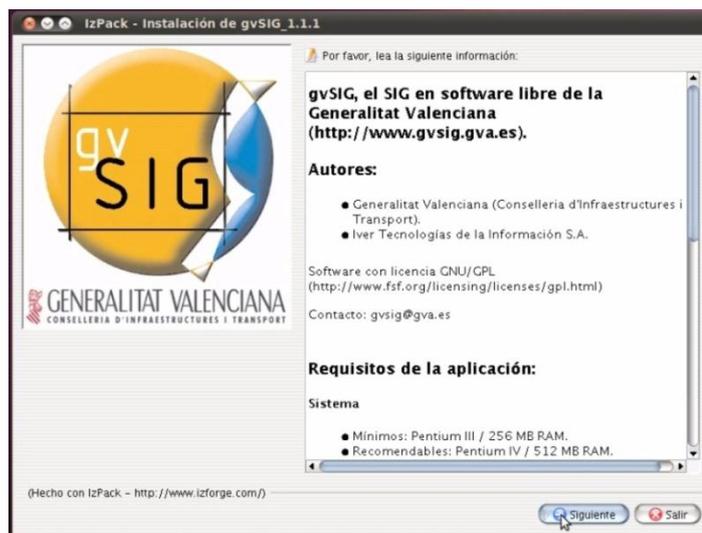


Figura 40: Instalación GvSIG – Información general.
Fuente: Propia.

7. Aceptamos la licencia, clic en siguiente hasta finalizar con la instalación:

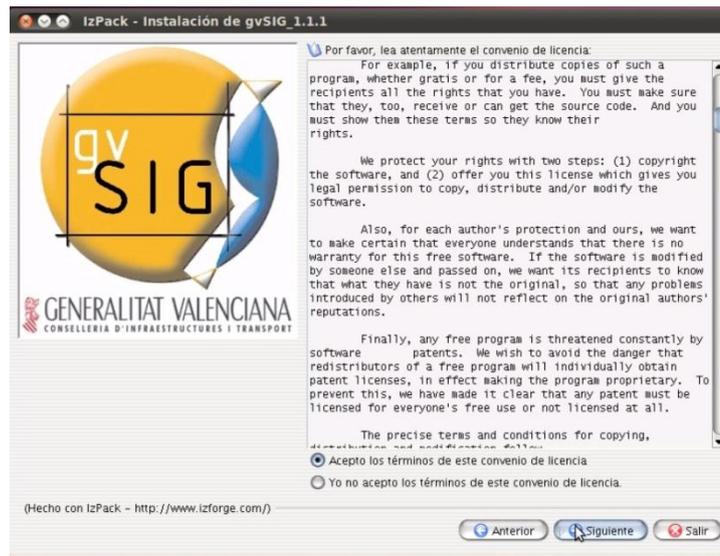


Figura 41: Instalación GvSIG – Acuerdo de Licencia.
Fuente: Propia.

8. Una vez terminada la instalación damos clic en hecho.



Figura 42: Instalación GvSIG – Fin de Instalación.
Fuente: Propia.

Migración de información catastral:

Los pasos a seguir para la migración de una capa desde un archivo shape a la base de datos geográfica son los siguientes:

1. Ingresamos a gvSIG.

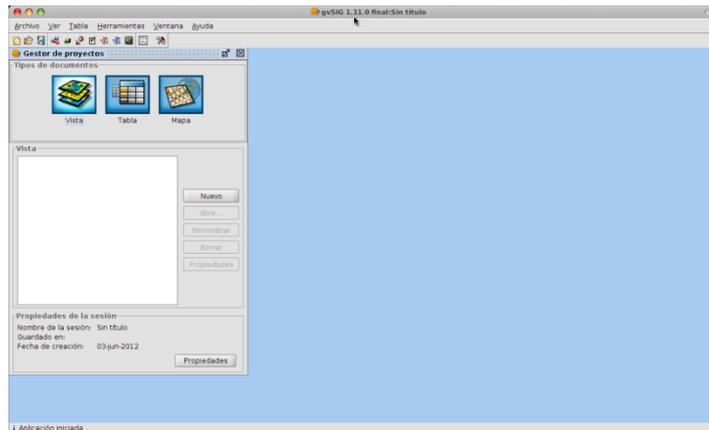


Figura 43: Migración - Inicio GvSIG.
Fuente: Propia.

2. Creamos un nuevo proyecto, clic en nuevo.
3. Seleccionamos el proyecto y damos clic en propiedades.

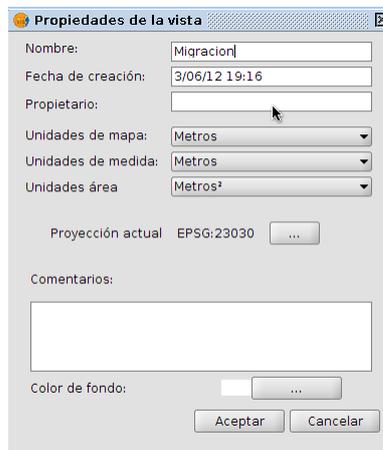


Figura 44: Migración – Configuración de un nuevo proyecto.
Fuente: Propia.

4. Cambiamos la proyección, clic en el botón Proyección actual.

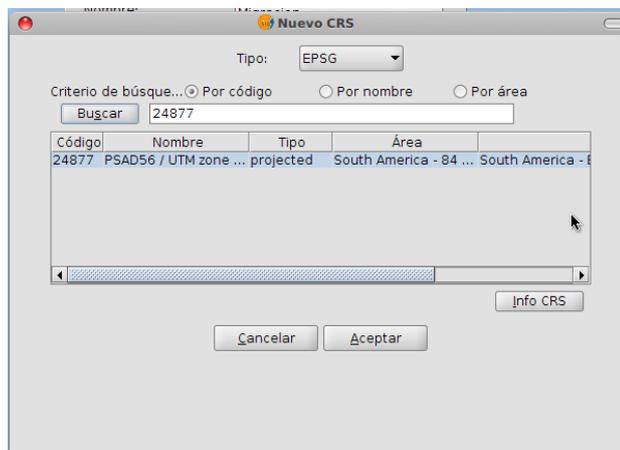


Figura 45: Migración – Configuración de la proyección.
Fuente: Propia.

En tipo escogemos EPSG, luego buscamos por código al 24877 que es la proyección de PSAD56 Zona 17 Sur, la misma que se va a utilizar para todas las capas del municipio. Aceptar hasta salir de las pantallas emergentes.

5. Clic en Abrir.



Figura 46: Migración – Vista principal de navegación.
Fuente: Propia.

Ampliamos la ventana para observar mejor.

6. En el menú Vista/Añadir capa/pestaña Archivo:

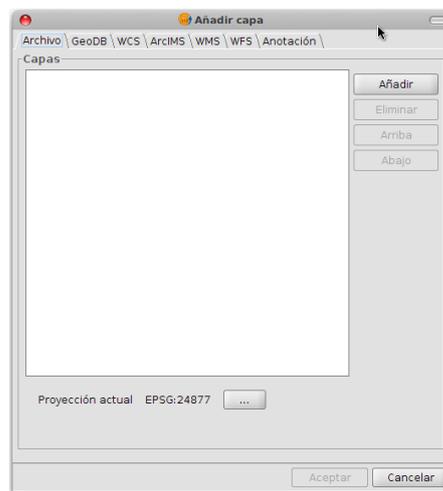


Figura 47: Migración – Ventana para agregar capas a la vista actual.
Fuente: Propia.

7. Clic en Añadir, buscamos la ubicación del archivo de la capa a migrar (en el ejemplo seleccionamos la capa predial):

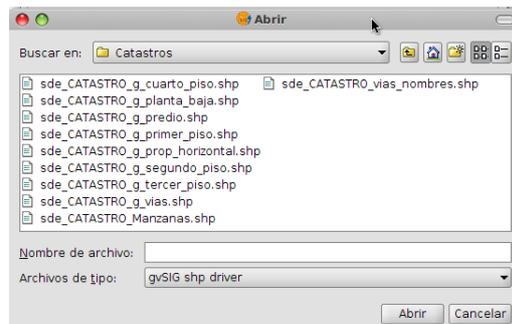


Figura 48: Migración – Abrir un shape.
Fuente: Propia.

Clic en abrir y luego aceptar.

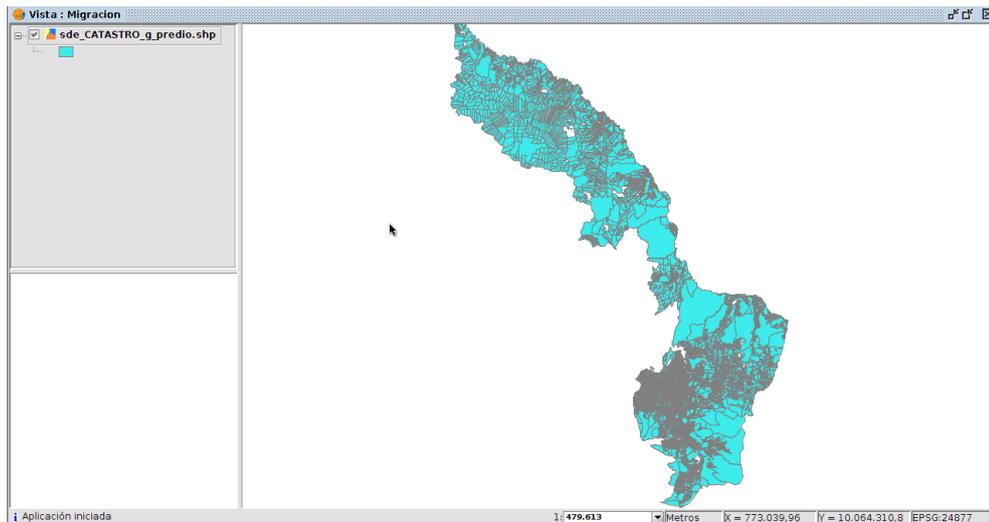


Figura 49: Migración – Navegación shape.
Fuente: Propia.

8. Seleccionamos la capa a exportar del árbol de capas y luego vamos a la opción en el menú: Capa/Exportar a.../Postgis.

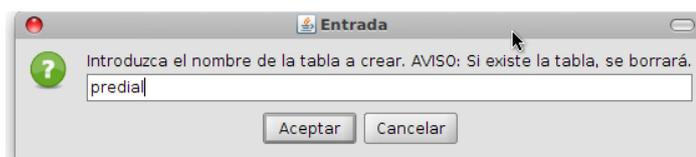


Figura 50: Migración – Ingreso del nombre de la capa.
Fuente: Propia.

9. Ingresamos un nombre a la tabla que va almacenar la capa, luego damos clic en Aceptar.
10. Nos pide que ingresemos la información de la conexión así la base de datos:

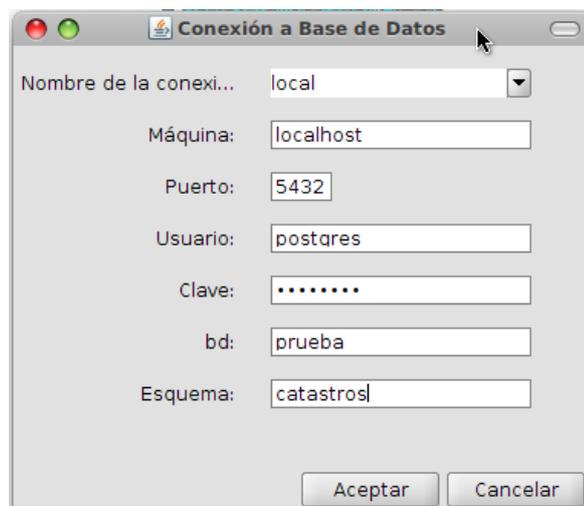


Figura 51: Migración – Configuración de la conexión de la base de datos a migrar.
Fuente: Propia.

11. Clic en aceptar, inicia el proceso de exportación, espere hasta que termine.

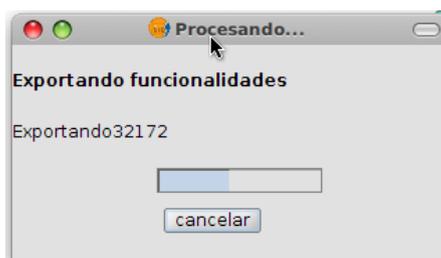


Figura 52: Migración – Proceso de migración de shape a una base de datos postGIS.
Fuente: Propia.

12. Luego podemos observar en la base datos que la tabla ya ha sido creada.

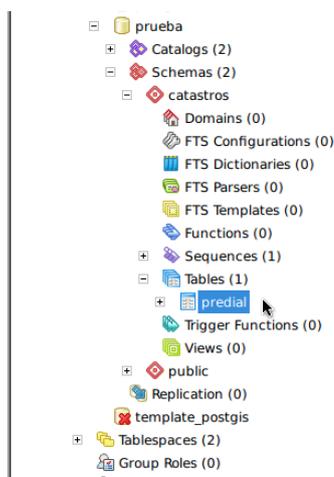


Figura 53: Migración – Verificación de los datos migrados en el pgadmin.
Fuente: Propia.

4.3.2.2. Segunda Iteración.

- **Historia de Usuario 3:** Publicación de mapas en la web.
 - **Tarea 1:** Instalación de Apache.

1. En un terminal ejecutamos:

```
sudo apt-get install apache2 apache2.2-bin apache2.2-common
```

2. Verificamos en un navegador con la dirección <http://localhost>

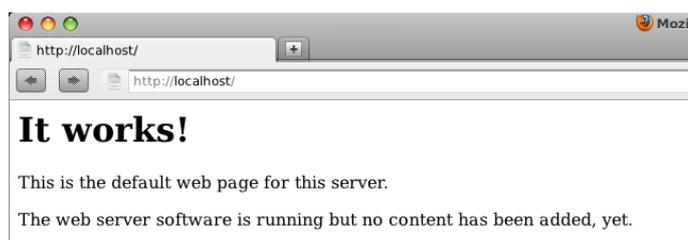


Figura 54: Prueba del funcionamiento del servidor Apache.
Fuente: Propia.

- **Tarea 2:** Instalación y configuración de MapServer.

1) En una terminal ejecutamos.

```
sudo apt-get install cgi-mapserver mapserver-bin mapserver-doc php5-mapscript python-mapscript
```

2) Verificamos si la instalación fue correcta.

```
/usr/lib/cgi-bin/mapserv -v
```

3) Reiniciamos apache.

```
/etc/init.d/apache2 restart
```

4) Verificamos en un navegador si el servicio está funcionando correctamente, con la dirección `http://localhost/cgi-bin/mapserv`, que es la dirección donde se publica el servicio por defecto.



Figura 55: Prueba del servicio de MapServer.
Fuente: Propia.

- **Tarea 3:** Publicación del Servicio WMS.

1. Primero creamos el repositorio o directorio donde se van a encontrar los archivos de configuración de MapServer, para la publicación del servicio WMS.

En este caso se creó en: `/servicios_wms/imiwms`

2. Debemos crear la siguiente estructura de archivos.

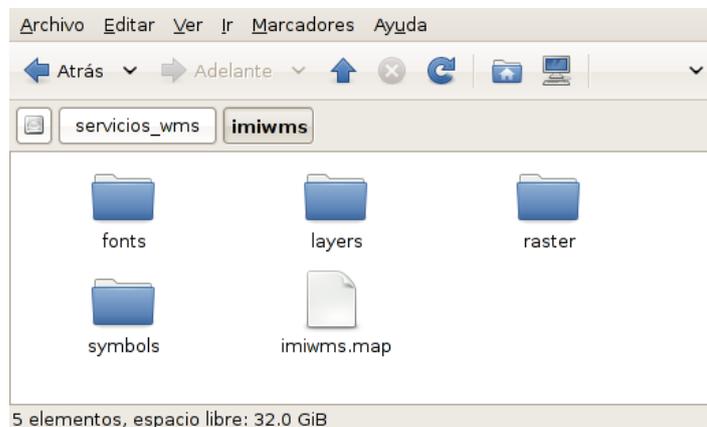


Figura 56: Configuración de carpetas para configuración de MapServer.
Fuente: Propia.

3. En la carpeta fonts guardamos todas las fuentes (.ttf) o tipos de letras y un archivo fonts.txt con una lista de pares con el nombre de la fuente y el nombre del archivo de la fuente.
4. En la carpeta symbols vamos a guardar todos los símbolos, en este caso dentro de esta vamos a crear un archivo symbols.sym, donde vamos a configurar la simbología para luego utilizarla en las capas o layers.

Parte del archivo symbol.sym

```

SYMBOLSET
  SYMBOL
    NAME "circle"
    TYPE ELLIPSE
    FILLED true
    POINTS 1 1 END
  END
  SYMBOL
    NAME "square"
    TYPE vector
    FILLED true
    POINTS 0 1 0 0 1 0 1 1 0 1 END
  END
END

```

5. En la carpeta raster se colocó todos los archivos .tif de las capas raster.
6. En la carpeta layers se creó todos los archivos .lay por cada capa o layer.

Predial.lay

Parte del archivo .lay

```

LAYER
  NAME "predial"
  STATUS ON
  TYPE POLYGON
  DUMP TRUE
  TRANSPARENCY 99
  DATA "the_geom from predial using unique gid using srid=24877"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres password=secret host=172.16.x.x
port=5432 dbname=base"
  PROCESSING "CLOSE_CONNECTION=DEFER"
  MAXSCALE -1.0
  MINSIZE -1.0
  SIZEUNITS pixels

  PROJECTION
    "init=epsg:24877"
  END

  TEMPLATE g_predio.html

  METADATA
    "wms_name" "Predial"

```

```

        "wms_title" "Predial"
        "wms_abstract" "Información Predial de la Ciudad de
Ibarra"
        "wms_keywordlist" "Predial"

        "wms_extent"          "783025.34801031          10017485.601
838304.829004297 10097177.2719919"
        "wms_srs" "EPSG:24877"

        "wms_formatlist" "image/png,image/jpeg"
        "wms_format" "image/jpeg"
        "wms_feature_info_mime_type" "application/vnd.ogc.gml"
        "wms_exceptions_format" "application/vnd.ogc.se_xml"
        "wms_dataurl_format" "application/vnd.ogc.gml"

        "wms_layer_group" "/predial"
        "wms_include_items" "clave_cata"
END

```

```

        LABELITEM "nro_predio"
        LABELCACHE ON
        LABELMAXSCALE 11491
        CLASSITEM tipo
        CLASS
        NAME "Urbano" #Nombre del estilo
        EXPRESSION /URBANO/ #valor que desea buscar el campo
        STYLE #estilo para los objetos que cumplen con la
propiedad
        #COLOR 175 202 219
        OUTLINECOLOR 234 141 19
        width 0.5
        END
        LABEL
        COLOR 0 0 0
        OUTLINECOLOR 255 255 255
        FONT "vera"
        TYPE TRUETYPE
        POSITION CC
        SIZE 6
        MINDISTANCE 300
        MINFEATURESIZE 5
        PARTIALS FALSE
        BUFFER 10
        END
        END
END # Layer

```

7. En el archivo .map (MapFile) vamos a configurar el servicio WMS o la instancia de un mapa en MapServer, en este punto se hace referencia a los archivos creados anteriormente.

Imiwms.map

Parte del Archivo .map.

```

MAP
        NAME "IMIWMS"

```

```

STATUS ON
EXTENT      781360.010000000093132      10017256.5399999991059303
840909.7900000000372529 10098464.1899999994784594
SYMBOLSET "symbols/symbols.sym"
FONTSET "fonts/fonts.txt"
IMAGECOLOR 255 255 255 #Color de Fondo de la imagen por defecto
TRANSPARENT ON
UNITS meters
IMAGETYPE "agg"

```

LEGEND

```

IMAGECOLOR 255 255 255
LABEL
    FONT "vera"
    ANGLE FOLLOW
    COLOR 0 0 0
    ENCODING "UTF-8"
    TYPE truetype
    SIZE 8

```

END

```

STATUS ON
TRANSPARENT ON
END

```

WEB

```

IMAGEPATH "tmp/"
IMAGEURL "tmp/"
METADATA
    "wms_encoding" "UTF-8"
    "wms_title" "Servicio WMS del Municipio de
Ibarra"
    "wms_name" "GIS Municipio de Ibarra"
    "wms_abstract" "Informacion geografica de la
ciudad de Ibarra"
    "wms_description" "Servicio de la informacion
catastral del Municipio de Ibarra"
    "wms_srs" " EPSG:24877"
    "wms_onlineresource"
"http://172.16.8.119/imiwms?"
    "wms_server_version" "1.1.1"
    "wms_formatlist" "image/png,image/jpeg"
    "wms_format" "image/png"
    "wms_feature_info_mime_type"
"application/vnd.ogc.gml"
    "wms_accessconstraints" "Esta informacion es
propiedad del Ilustre Municipio de Ibarra"
    "wms_contactorganization" "Ilustre Municipio de
Ibarra"

```

END

END

PROJECTION

```

"proj=utm"
"init=epsg:24877"
"ellps=intl"
"units=m"
"south"
"zone=17"
"no_defs"

```

```

END

OUTPUTFORMAT
    NAME "AGG_PNG" #nombre como parametros en openlayers
map_imagetype: 'AGG_PNG'
    DRIVER AGG/PNG
    IMAGEMODE RGB
    FORMATOPTION "QUANTIZE_FORCE=ON"
    FORMATOPTION "QUANTIZE_DITHER=OFF"
    FORMATOPTION "QUANTIZE_COLORS=256"
END
#Capas de Alarmas
INCLUDE "layers/catastros/predial.lay"
END # Map File

```

Nota: Para más información de cómo configurar el MapFile, revisar las especificaciones del MapFile en la página oficial de MapServer – www.mapserver.org.

8. Ahora configuramos el host virtual para el servicio.

- Editamos el archivo:
`nano /etc/apache2/sites-enabled/000-default`
- Escribimos la configuración del servicio:

```

ScriptAlias /imiwms /usr/lib/cgi-bin/mapserv
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        AddHandler cgi-script .py .cgi
        Options +ExecCGI -MultiViews
+SymLinksIfOwnerMatch
        Order allow,deny
        Allow from all
        SetEnvif Request_URI "/imiwms"
MS_MAPFILE=/servicios_wms/imiwms/imiwms.map
    </Directory>

```

9. Probando el servicio WMS:

Probando request GetCapabilities.

<http://172.16.8.119/imiwms?SERVICE=WMS&REQUEST=GetCapabilities>

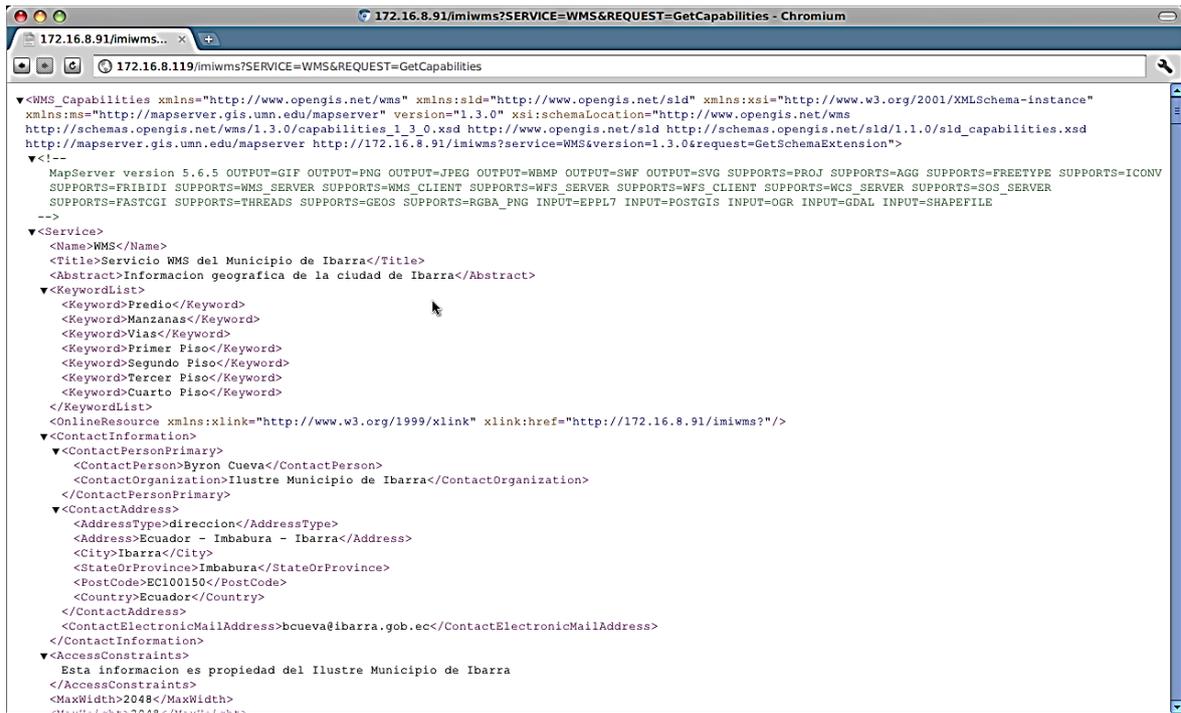


Figura 57: Prueba del servicio WMS Request GetCapabilities.
Fuente: Propia.

Probando request GetMap desde Quantum GIS.

http://172.16.8.119/imiwms?LAYERS=ct_predial&TRANSPARENT=FALSE&FORMAT=image%2Fjpeg&SERVICE=WMS&VERSION=1.1.1&REQUEST=GetMap&STYLES=&SRS=EPSG%3A24877&BBOX=783025.34801031,10017485.601,823629.17301031,10058089.426&WIDTH=256&HEIGHT=256 HTTP/1.1

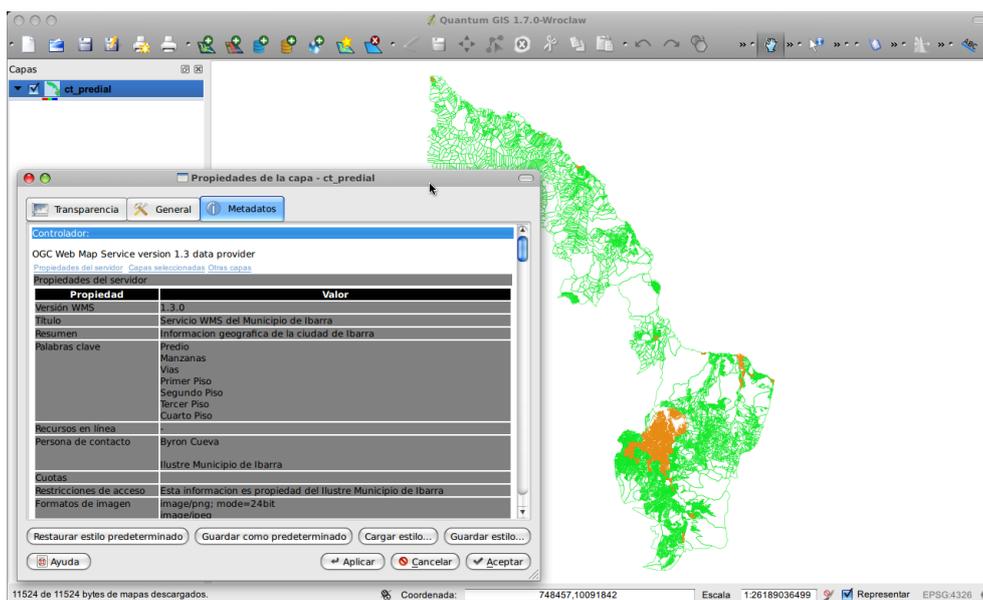


Figura 58: Prueba del servicio WMS Request GetMap.
Fuente: Propia.

- **Tarea 4:** Instalación de Tomcat.

1. Instalación del JDK.

- Descargar el binario `jdk-6u26-linux-x64.bin`.
- Copiar en la carpeta `/usr/lib/jvm`.
- Ejecutar el binario. Ejemplo: `./jdk-6u26-linux-x64.bin`.
- Configurar las variables de entorno en `/etc/bash.bashrc`.
`export JAVA_HOME=/usr/lib/jvm/carpeta_jdk`
`export JRE_HOME=$JAVA_HOME/jre`
`export PATH=$JAVA_HOME/bin:$PATH`

2. Descargar el archivo comprimido de instalación de Tomcat, de su página oficial, preferentemente la versión 7.

3. Descomprimirlo en `/usr/local/`.

4. Configurar la variable de entorno.

- `export CATALINA_HOME=/usr/local/apache-tomcat-7-x.x`

- **Tarea 5:** Instalación y configuración de Geoserver.

1. Descargar el archivo `geoserver.war`, de la página oficial de geoserver.

2. Copiar el archivo `war` en `/usr/local/carpeta_tomcat_7/webapps`, mientras este corriendo el servicio, para que la aplicación se publique.

3. Probar si está instalado en `http://localhost:8080/geoserver`.

- **Tarea 6:** Publicación del servicio WFS.

1. Crear un nuevo espacio de trabajo.

- a. Clic en `Workspace`.
- b. Clic en `Add new workspace`.
- c. Ingresar los dato y luego damos clic en `Submit`.

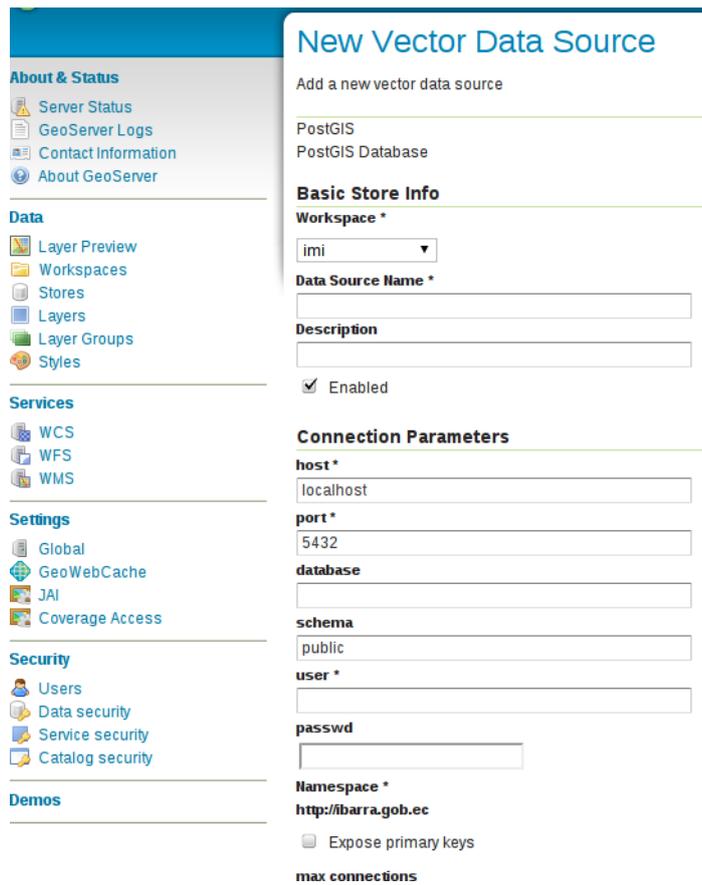


Figura 59: GeoServer – Creación de un espacio de trabajo.
Fuente: Propia.

2. Agregar un origen de datos PostGIS.
 - a. Clic en Store.
 - b. Clic en Add new data store.
 - c. Seleccione PostGIS,
 - d. Una vez ingresado los datos de la configuración de la conexión a la base, damos clic en guardar.

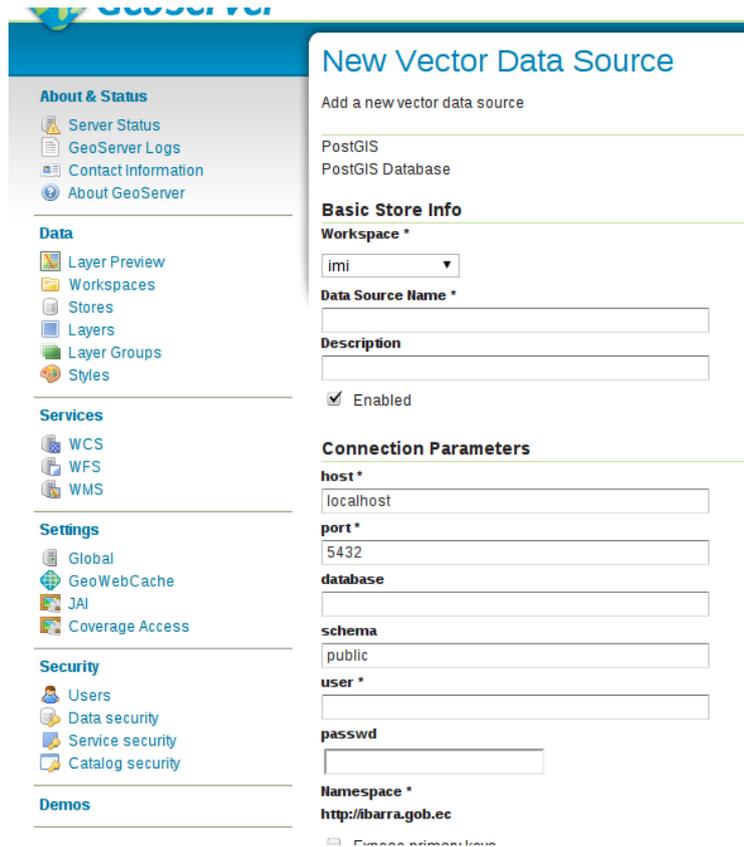


Figura 60: GeoServer – Creación de un Store.
Fuente: Propia.

3. Configurar Servicio WFS por capa.
 - a. Clic en Layers.
 - b. Clic en Add new resource.
 - c. Seleccionar el origen de datos.
 - d. Ingrese la información requerida y posteriormente guarde.

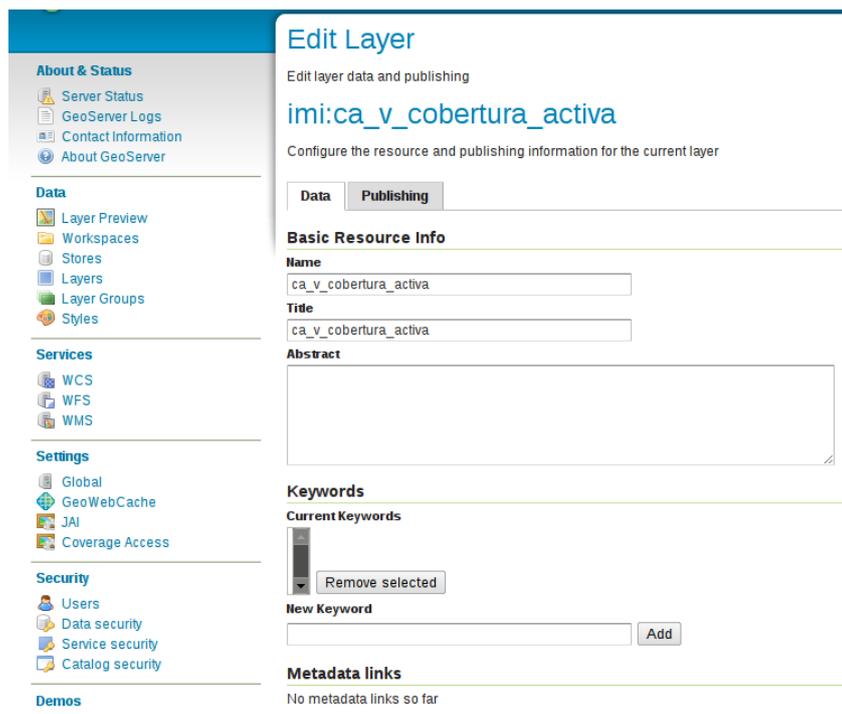


Figura 61: GeoServer – Publicación de una capa.
Fuente: Propia.

- **Tarea 7:** Instalación y configuración de Tile Cache.
 1. Ingresar a una terminal como usuario root.
 2. Instalar los siguientes paquetes:
`apt-get install libapache2-mod-python`
 3. Descargar el paquete tilecache-2.11.tar.gz del sitio oficial de TileCache.
 4. Descomprimir el paquete dentro del directorio /usr/lib/cgi-bin/.
`tar xzvf tilecache-2.11-tar-gz`
 5. Cambiar el nombre original de la carpeta descomprimida por tilecache, e ingresar en ella.
`mv tilecache-2.11 tilecache`
`cd tilecache`
 6. Cambiar la extensión del archivo tilecache de .cgi a .py, para ejecutarlo con el módulo python de apache.
`mv tilecache.cgi tilecache.py`

- **Tarea 8:** Publicación del servicio WMSC.
 1. Por cada capa que se publique a cache se debe adjuntar sus configuraciones en el archivo tilecache.cfg.

Capa Manzanas en cache:

```
[manzanas]
type=WMSLayer
url=http://
http://186.3.9.15/imiwms/imiwms?transparent=TRUE&MAP_IMAGETYPE=
agg&&VERSION=1.1.1
extension=png
srs=EPSG:24877
extent_type=loose
layers=manzanas
bbox=816842.629303537,1.00332569441673E7,824470.137493032,1.004
36811182391E7
resolutions=40.71942996796860825270,20.35971498398430412635,10.
17985749199215206318,5.08992874599607603159,2.54496437299803801
579,1.27248218649901900790,0.63624109324950950395,0.3$
maxresolution=29.794953865
size=256,256
```

2. Publicamos el servicio así la web a través de apache. Creamos el host virtual.

```
ScriptAlias /wmscache /usr/lib/cgi-bin/tilecache/tilecache.py
```

```
<Directory /usr/lib/cgi-bin/tilecache>
```

```
    AddHandler mod_python .py
```

```
    PythonHandler TileCache.Service
```

```
    PythonOption TileCacheConfig /usr/lib/cgi-bin/tilecache/tilecache.cfg
```

```
    PythonPath "[usr/lib/cgi-bin/tilecache/] + sys.path
```

```
    PythonDebug On
```

```
</Directory>
```

3. Prueba del servicio WMSC

Probando request GetCapabilities:

```
http://172.16.8.119/imiwmscache?REQUEST=GetCapabilities&TILED=true&SERVI
CE=WMS
```

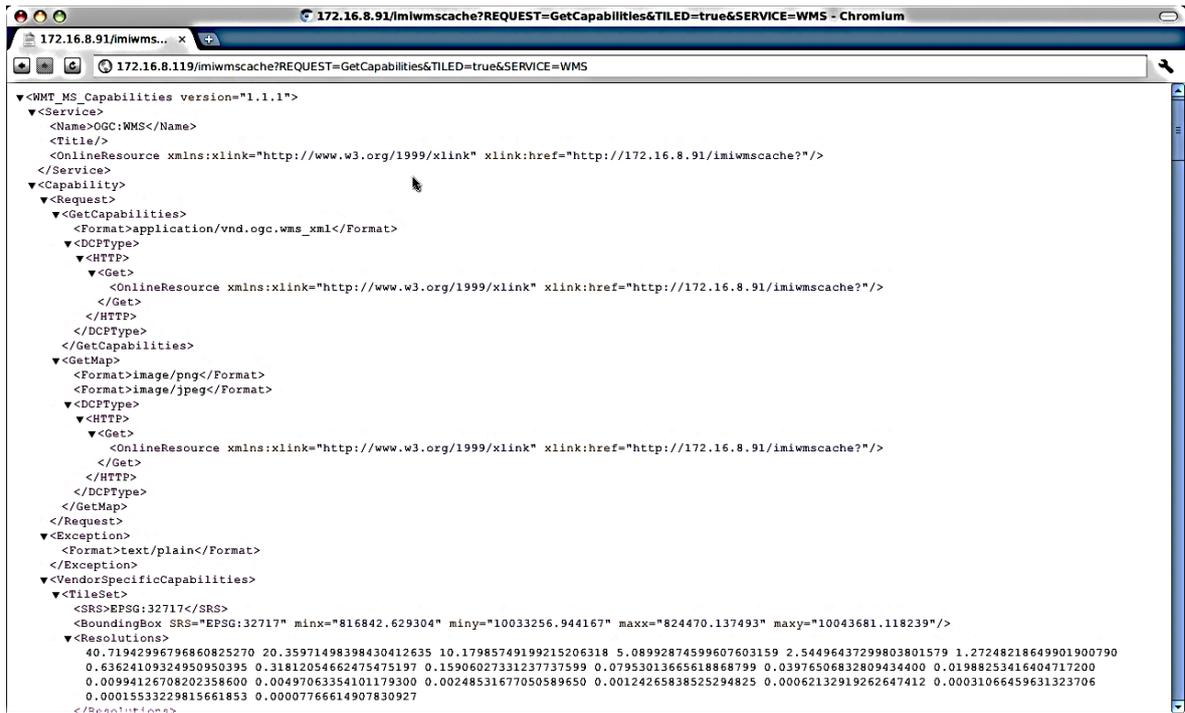


Figura 62: Probando request GetCapabilities.
Fuente: Propia.

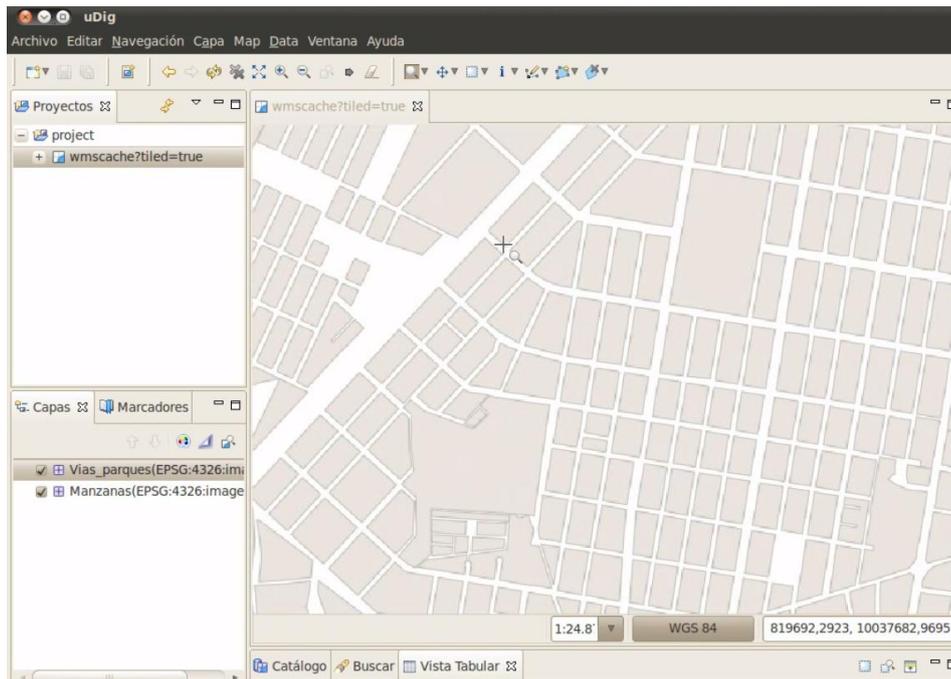


Figura 63: Probando request GetMap, a través del cliente pesado udig.
Fuente: Propia.

- **Historia de Usuario 4:** Despliegue de información geográfica y colección de capas.
 - **Tarea 1:** Diseño de la estructura de datos para el control de capas.

- **Tarea 2:** Diseño de la interfaz principal de la aplicación.

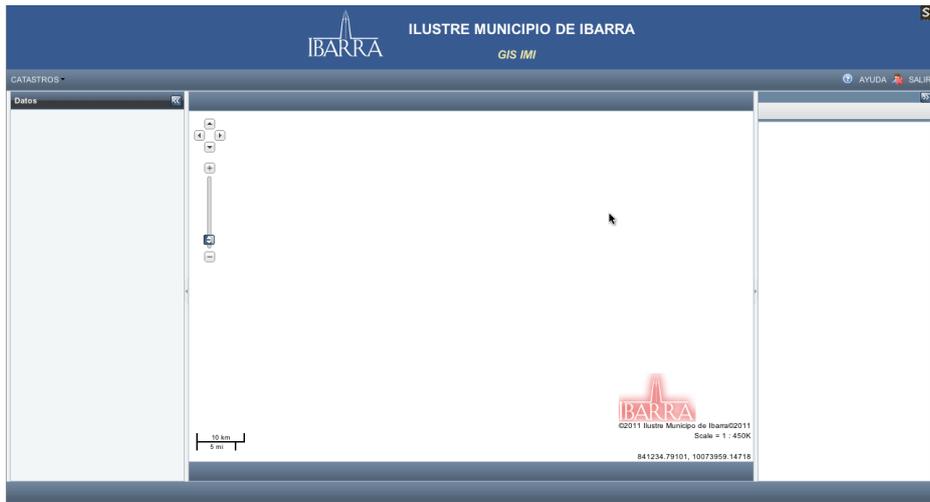


Figura 65: Diseño de la interfaz de la aplicación.

Fuente: Propia.

- **Tarea 3:** Programación de envío de datos de configuración de capas.

Tarjetas CRC

Tarjeta CRC	
Nombre de la Clase: GiMapLogic.	
Descripción: Clase Logic, lógica de la instancia del mapa de la aplicación.	
Colaboradores	
Super Clase:	
Requiere: GiMapTable (DAO), GiMap (DTO)	
Responsabilidades	
Métodos	
Nombre	Descripción
getMapPorId	Devuelve un objeto GiMap, con la información de configuración de la instancia del mapa.
getCapaBasePorIdMap	Devuelve un Array, con la configuración de la capa base o principal del Mapa.
getArrayMapPorIdMap	Devuelve un Array, con la información de la configuración de la instancia del mapa.

Tabla 117: Tarjeta CRC: GiMapLogic.

Fuente: Propia.

Tarjeta CRC	
Nombre de la Clase: GiGrupoLogic.	
Descripción: Clase Logic, lógica del grupo de capas.	
Colaboradores	
Super Clase:	
Requiere: GiGrupoTable (DAO), GiGrupo (DTO)	
Responsabilidades	
Métodos	
Nombre	Descripción
getGruposInicialesPorIdMap	Devuelve un Array, con la información de los grupos de capas iniciales del mapa.
getSubgrupoPorIdGrupo	Devuelve un Array, con la información de los subgrupos de capas de un grupo específico.

**Tabla 118: Tarjeta CRC: GiGrupoLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: GiLayerLogic.	
Descripción: Clase Logic, lógica de las capas o layers.	
Colaboradores	
Super Clase:	
Requiere: GiLayerTable (DAO), GiLayer (DTO)	
Responsabilidades	
Métodos	
Nombre	Descripción
getCapasPorIdGrupo	Devuelve un Array, con la información de las capas que le pertenecen a un grupo de capas.
getNodosCapasPorIdGrupo	Devuelve un Array, con la información de los nodos de capas que pertenecen a un grupo.

**Tabla 119: Tarjeta CRC: GiLayerLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: giAccesoManager	
Descripción: Clase Manager, esta permite la generación de la información de configuración del mapa y capas del visor.	
Colaboradores	
Super Clase:	
Requiere: Usuario, GiMapLogic, GiGrupoLogic, GiLayerLogic	
Responsabilidades	
Métodos	
Nombre	Descripción
getItems	Devuelve un Array, con la configuración de los ítems del menú al cual tiene permisos el usuario.
getMap	Devuelve un objeto GiMapa, con la configuración de la instancia actual del mapa.
getCapaBase	Devuelve un objeto GiLayer, con la configuración de la capa base del mapa que se encuentra instanciado.
getGiGrupos	Devuelve un Array, con la configuración del grupo de capas.
getCapasPorIdGrupo	Devuelve un Array, con la configuración de las capas que le pertenecen a un grupo.
getNodosPorGrupo	Devuelve la configuración de los nodos del árbol de capas.

**Tabla 120: Tarjeta CRC: giAccesoManager.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: giDashboardAction	
Descripción: Clase Controlador, esta controla el flujo de solicitudes de la información de configuraciones de capas.	
Colaboradores	

Super Clase: sfAction	
Requiere: giAccesoManager	
Responsabilidades	
Métodos	
Nombre	Descripción
executeGetCapasUsuario	Devuelve en formato JSON, la configuración de las capas, a las que tiene permiso el usuario.
executeGetCapasCompletas	Devuelve en formato JSON, la configuración del servicio de un grupo de capas.
executeGetIcons	Devuelve en formato JSON, la descripción de los iconos que se utilizan en el árbol de capas.
executeGetGruposUsuario	Devuelve en formato JSON, los grupos de capas que tiene permisos el usuario.
executeVistaActual	Devuelve un PDF, con la vista actual, del visor de mapas.

Tabla 121: Tarjeta CRC: giDashboardAction.
Fuente: Propia.

- **Tarea 4:** Programación del ambiente de consumo de servicios WMS y WMSC.

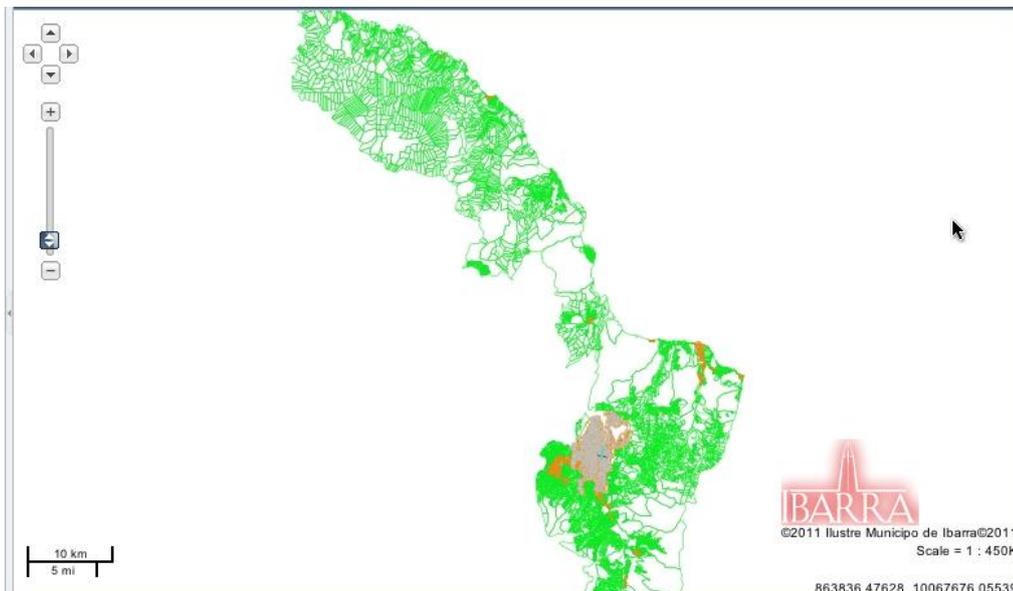


Figura 66: Área de visión de mapas.
Fuente: Propia.

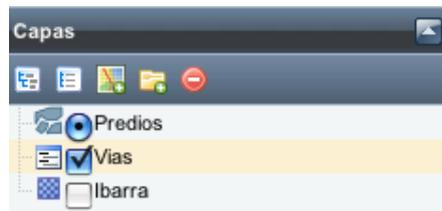
Tarjeta CRC.

Tarjeta CRC	
Nombre de la Clase: GisIbarra.panel.MapPanel.	
Descripción: Clase Vista, componente para el despliegue de las capas disponibles en el mapa.	
Colaboradores	
Super Clase: GeoExt.MapPanel.	
Requiere: GisIbarra.toolbar.ToolbarNavMap, GisIbarra.toolbar.ToolbarOptionsMap, OpenLayers.Map, OpenLayers.Layer.WMS.	
Responsabilidades	
Métodos	
Nombre	Descripción
crearMapa	Crea la instancia del mapa para el consumo de los servicios WMS, WMS-C, WCS.
agregarCapaVector	Agrega una capa tipo SVG al mapa.
agregarCapa	Agrega una capa tipo WMS o WMS-C.
agregarControl	Agregar un control al mapa.
eliminarControl	Elimina un control del mapa.
existeCapa	Verifica si existe una capa.
getLayerPorName	Busca una capa por el atributo nombre.
eliminarCapa	Elimina una capa de la instancia del mapa.
getRecordLayer	Devuelve un objeto GeoExt.data.LayerRecord, con los atributos de una capa.
getDataWMSFeaturesInfo	Devuelve un objeto, con la información de respuesta de un request GetFeaturesInfo.
handlerBeforeFeatureInfo	Evento que se ejecuta después de haber obtenido la información de un servicio WMS request GetFeaturesInfo.
updateFormats	Reconfigura el formato de renderizado SVG a un formato GeoJSON.
addVectInfo	Agrega una geometría SVG.
boxAGEometria	Zoom a un determinado extent de una geometría.
limpiaVectInfo	Limpia las geometrías SVG.

empiezaAnimaSelect	Activa la animación SVG.
animaSelect	Prende y apaga una geometría SVG.
terminaSelect	Desactiva la animación de una geometría SVG.
activaCapa	Hace visible a una capa en el mapa.

**Tabla 122: Tarjeta CRC: GisIbarra.panel.MapPanel.
Fuente: Propia.**

- **Tarea 5:** Programación del tree de colección de capas.



**Figura 67: Tree de colección de capas.
Fuente: Propia.**

Tarjeta CRC.

Tarjeta CRC	
Nombre de la Clase: GisIbarra.comp.TreeCapas.	
Descripción: Clase Vista, componente que permite observar de una forma organizada las capas disponibles en el mapa.	
Colaboradores	
Super Clase: Ext.Panel	
Requiere: Ext.tree.TreeLoader, Ext.tree.TreePanel, GisIbarra.menu.ContextMenuNodeLayer, GisIbarra.menu.ContextMenuNodeGrupo, GisIbarra.toolbar.ToolbarTreeCapas.	
Responsabilidades	
Métodos	
Nombre	Descripción
afterRender	Función que se ejecuta después de haberse renderizado el Componente.
getSelectedLayerRecord	Devuelve un objeto Ext.data.Record, con la información de la capa actualmente seleccionada.
agregarNodosLayer	Agrega nodos tipo capa al árbol.

agregarNodosGrupo	Agregar nodos tipo grupo al árbol.
getRootNodo	Devuelve el nodo raíz.
getActualNodo	Devuelve el nodo actualmente seleccionado.
getExisteNodo	Verifica si existe un nodo en específico.
esNodoGrupo	Verifica si un nodo es de tipo grupo.
buscarNodoPorText	Busca un nodo por el atributo text.
getNodeSeleccionado	Devuelve un objeto Ext.tree.TreeNode, nodo seleccionado.
eliminarNodo	Elimina un nodo del árbol, sea tipo capa o grupo.
ejecutaAccionActivaCheckLayer	Evento que se dispara cuando se activa o desactiva una capa.
expandirTodo	Expande el árbol.

Tabla 123: Tarjeta CRC: GisIbarra.comp.TreeCapas.
Fuente: Propia.

4.3.2.3. Tercera Iteración.

- **Historia de Usuario 5:** Funciones de navegación y descriptor de leyendas.
 - **Tarea 1:** Programación de funciones de navegación.



Figura 68: Barra de Herramientas de Navegación.
Fuente: Propia.

Tarjeta CRC.

Tarjeta CRC
Nombre de la Clase: GisIbarra.toolbar.ToolbarNavMap
Descripción: Clase Vista, toobal que permite la navegación sobre el visor de mapas
Colaboradores
Super Clase: Ext.Toolbar
Requiere: GisIbarra.toolbar.BuildItem, GisIbarra.config.ToolbarNavMap.tooltip, GeoExt.Action
Responsabilidades

Atributos	
Nombre	Descripción
buildbar	Objeto GisIbarra.toolbar.BuildItem
ct_navigation	Control de Navegación.
ct_zoom_in	Control para aumentar el zoom.
ct_zoom_out	Control para disminuir el zoom.
ct_nav_previous_action	Control para regresar a una vista anterior en el mapa.
ct_nav_next_action	Control para ir a la posterior navegación realizada en el mapa.
ct_max_extent	Control para ir a un al máximo extent permitido por el mapa.
ct_zoom_in_box	Control para aumentar el zoom a través de un cuadro.
ct_zoom_out_box	Control para disminuir el zoom a través de un cuadro.
Métodos	
Nombre	Descripción
createControls	Crea los controles.
activaNavegacion	Activa la navegación.

Tabla 124: Tarjeta CRC: GisIbarra.toolbar.ToolbarNavMap.
Fuente: Propia.

- **Tarea 2:** Programación del panel descriptor de leyendas.



Figura 69: Barra de Leyendas.
Fuente: Propia.

- **Historia de Usuario 6:** Adición y eliminación de capas.
 - **Tarea 1:** Diseño de la ventana para agregar capas y grupos de capas.

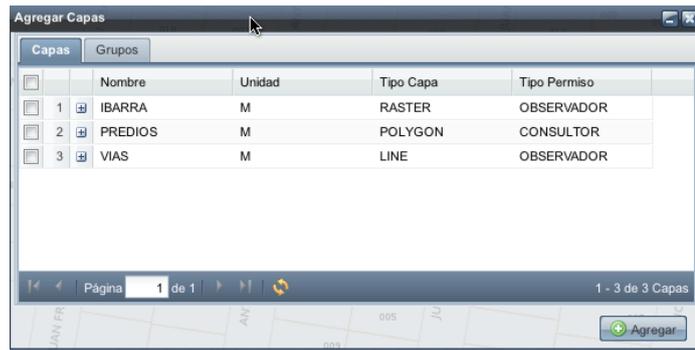


Figura 70: Ventana para agregar capas.
Fuente: Propia.

- **Tarea 2:** Programación de la adición y eliminación de capas y grupos de capas. Tarjetas CRC, tablas del 110 a la 114.
- **Historia de Usuario 7:** Estimación de longitud y área.
- **Tarea 1:** Programación de la función de medición de longitud.

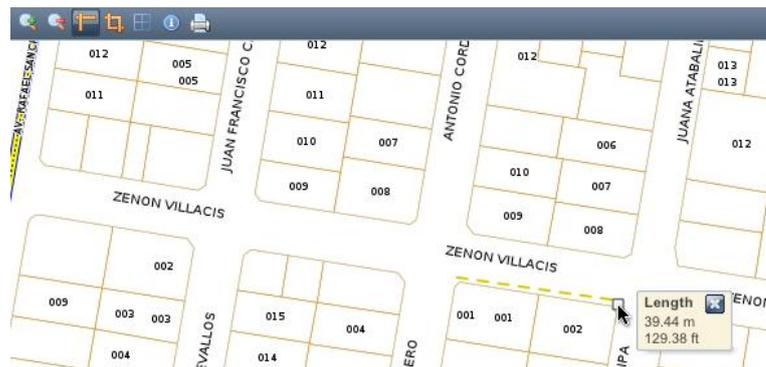


Figura 71: Herramienta de medición de la longitud.
Fuente: Propia.

- **Tarea 2:** Programación de la medición de área.



Figura 72: Herramienta de Medición de área.
Fuente: Propia.

- **Módulo de Consultas y Búsquedas.**

- **Historia de Usuario 8:** Ubicación rápida de información catastral.

- **Tarea 1:** Diseño y programación de ubicación de zonas.



Figura 73: Herramienta de búsqueda de Zonas.
Fuente: Propia.

- **Tarea 2:** Diseño y programación de ubicación de calles.



Figura 74: Herramienta de consulta por calles.
Fuente: Propia.

- **Tarea 3:** Diseño y programación de ubicación de lugares principales.



Figura 75: Herramienta de consulta lugares.
Fuente: Propia.

Tarjeta CRC.

Tarjeta CRC
Nombre de la Clase: giGisActions.
Descripción: Clase Controlador, permite el control de solicitudes de búsquedas de información

geográfica.	
Colaboradores	
Super Clase: sfActions.	
Requiere:	
Responsabilidades	
Métodos	
Nombre	Descripción
executeGetGeometriaPredio	Devuelve en GeoJSON la geometría de un predio.
executeGetCalles	Devuelve en JSON, la información de las calles.
executeGetGeometriaCalles	Devuelve en GeoJSON las geometrías de las calles.
executeGetGeometriaClave	Devuelve en GeoJSON la geometría de una zona, manzana o sector.

Tabla 125: Tarjeta CRC: giGisActions.
Fuente: Propia.

- **Historia de Usuario 9:** Consulta de Información predial.
 - **Tarea 1:** Configuración de la capa predial que permita solicitudes GetFeaturesInfo. En el archivo de configuración de la capa predial predial.lay se configuro lo siguiente, dentro de las etiquetas METADATA.

```

"wms_feature_info_mime_type" "application/vnd.ogc.gml"
"wms_dataurl_format" "application/vnd.ogc.gml"
"wms_include_items" "clave_cata"

"ows_include_items" "clave_cata"
"ows_exclude_items" ""
"ows_feature_info_mime_type" "application/vnd.ogc.gml"

"gml_include_items" "clave_cata"
"gml_exclude_items" ""
"gml_featureid" "gid"

```

Nota: Solo a través del servicio WMS, se obtiene la clave catastral del predio a partir de ella se obtiene la información alfanumérica y fotos del predio, con una posterior consulta a las diferentes bases de datos.

- **Tarea 2:** Diseño de la ventana de consulta de la información predial.

Figura 76: Herramienta de consulta de la Información de un predio.
Fuente: Propia.

- **Tarea 3:** Programación de la función de consulta y carga de datos.

Modelo relacional de la base de datos binaria.

predio_bin	
id	INTEGER
cod_catastral	CHARACTER VARYING(18)
fecha_hora	TIMESTAMP(6) WITHOUT TIME ZONE
usuario	CHARACTER VARYING(50)
fotografia	BYTEA

Figura 77: Modelo Relacional – Fotos Predio.
Fuente: Propia.

Modelo relacional de la base datos alfanuméricos.

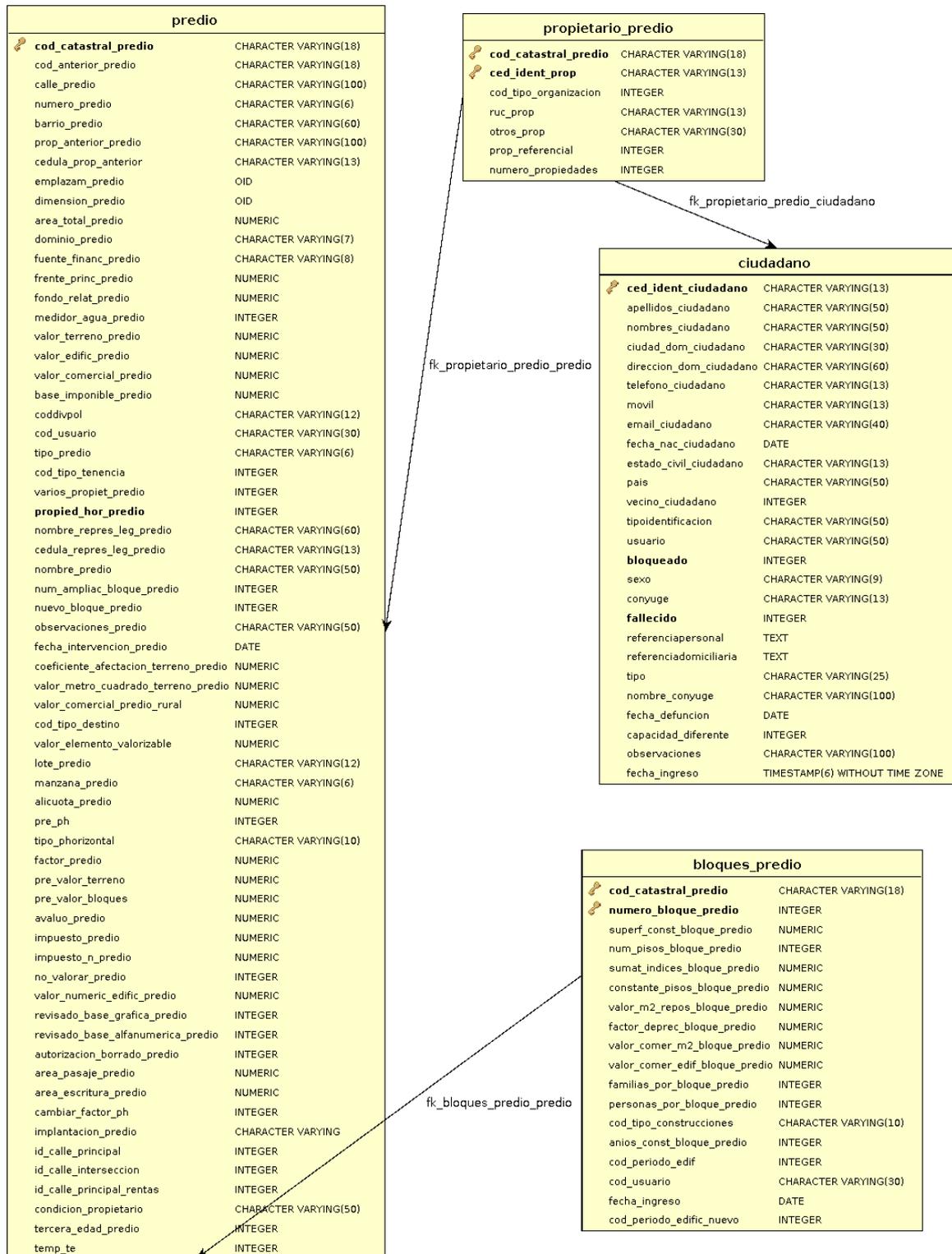


Figura 78: Modelo Relacional, para obtener la información del Predio.
Fuente: Propia.

4.3.2.4. Cuarta Iteración.

- **Historia de Usuario 10:** Búsqueda de un predio.
 - **Tarea 1:** Diseño de la ventana para la búsqueda de un predio.

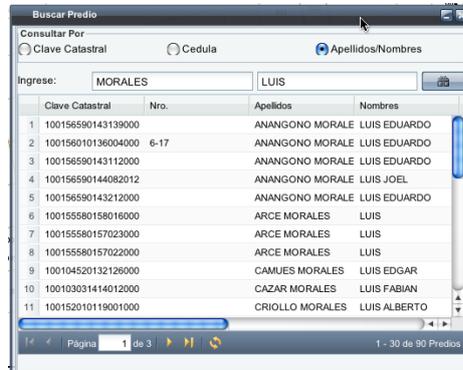


Figura 79: Interfaz para la consulta de un predio.
Fuente: Propia.

- **Tarea 2:** Programación de la función de dibujo de un predio.

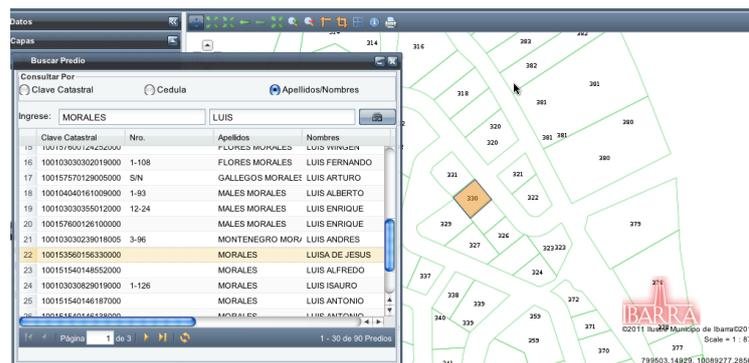


Figura 80: Renderización SVG de un predio seleccionado.
Fuente: Propia.

- **Historia de Usuario 11:** Consulta de la zonificación de un predio.
 - **Tarea 1:** Diseño de la ventana para consulta de la zonificación de un predio.



Figura 81: Interfaz de consulta de la Zonificación.
Fuente: Propia.

Modelo relacional

zonificacion_ibarra	
zona_tipo	CHARACTER VARYING(255)
forma_ocupacion	CHARACTER VARYING(255)
lote_minimo	NUMERIC
frente_minimo	NUMERIC
altura_pisos	NUMERIC
maxima_metros	NUMERIC
cos_porcentaje	NUMERIC
cus_porcentaje	NUMERIC
frente	NUMERIC
lateral_uno	NUMERIC
retiros_lateral_dos	NUMERIC
minimos_posterior	NUMERIC
entre_bloques	NUMERIC
densidades_bruta_neta	NUMERIC
f_quince	NUMERIC
observaciones	CHARACTER VARYING(255)

zonificacion_ibarra_predios	
clave_total	CHARACTER VARYING(18)
parroquia	CHARACTER VARYING(255)
zona	CHARACTER VARYING(255)
sector	CHARACTER VARYING(255)
manzana	CHARACTER VARYING(255)
r_uno	CHARACTER VARYING(255)
r_dos	CHARACTER VARYING(255)
zonificacion	CHARACTER VARYING(255)

Figura 82: Modelo Relacional para la consulta de la Zonificación de un predio.
Fuente: Propia.

- **Historia de Usuario 12:** Consulta de los usos de suelo de un predio.
 - **Tarea 1:** Diseño de la ventana para consulta del uso de suelo de un predio.

Compatibilidad	Uso
PRINCIPAL	VIVIENDA MEDIA DENSIDAD
PRINCIPAL	COMERCIO VECINAL
PRINCIPAL	COMERCIO SECTORIAL
PRINCIPAL	COMERCIO ZONAL
PRINCIPAL	INDUSTRIAL BAJO IMPACTO
PRINCIPAL	EQUIPAMIENTO SECTORIAL
RESTRINGIDO O CONDICIONADO	VIVIENDA ALTA DENSIDAD
RESTRINGIDO O CONDICIONADO	INDUSTRIAL MEDIANO IMPACTO
RESTRINGIDO O CONDICIONADO	EQUIPAMIENTO ZONAL
RESTRINGIDO O CONDICIONADO	EQUIPAMIENTO INFRAESTRUCTURA
PROHIBIDO	COMERCIO ESPECIAL
COMPATIBLE	VIVIENDA BAJA DENSIDAD
PROHIBIDO	COMERCIO RESTRINGIDO

Figura 83: Interfaz para la consulta del uso de suelo de un predio.
Fuente: Propia.

Modelo Relacional

us_zonificacion_uso_suelo	
cod_zonificacion_compatible	NUMERIC(18,0)
zona_tipo	CHARACTER VARYING(30)
cod_compatibilidad	CHARACTER VARYING(1)
cod_uso_suelo	CHARACTER VARYING(3)

us_zonificacion_compatible	
cod_zonificacion_compatible	INTEGER
zona_tipo	CHARACTER VARYING(10)
cod_compatible	CHARACTER VARYING(1)

us_actividades	
cod_uso_suelo	CHARACTER VARYING(3)
cod_padre_uso_suelo	CHARACTER VARYING(1)
descripcion_uso_suelo	CHARACTER VARYING(30)
estado_uso_suelo	CHARACTER VARYING(1)

us_compatible	
cod_compatibilidad	CHARACTER VARYING(1)
descripcion_compatibilidad	CHARACTER VARYING(30)

Figura 84: Modelo Relacional para la consulta del uso del suelo.
Fuente: Propia.

- **Historia de Usuario 13:** Consulta de la descripción de un terreno.
- **Tarea 1:** Diseño de la ventana para consulta de la descripción del terreno.



Figura 85: Ventana descripción del terreno.
Fuente: Propia.

Modelo Relacional

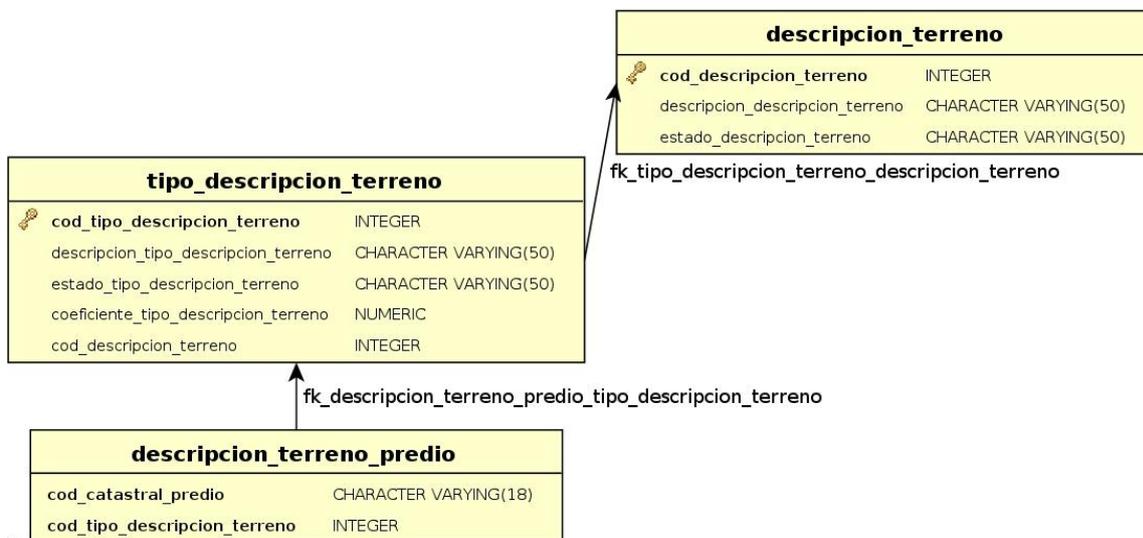


Figura 86: Modelo Relacional para la consulta de la descripción del terreno.
Fuente: Propia.

- **Historia de Usuario 14:** Consulta de la infraestructura y servicios.
- **Tarea 1:** Diseño de la ventana para consulta de la infraestructura y servicios de un predio.

Descripcion ▲	Tipo
Descripcion: ABASTECIMIENTO DE AGUA	
ABASTECIMIENTO DE AGUA	CONEXIÓN DOMICILIARIA
ABASTECIMIENTO DE AGUA	RED PÚBLICA
Descripcion: ALCANTARILLADO	
ALCANTARILLADO	RED COMBINADA
Descripcion: ENERGÍA ELÉCTRICA	
ENERGÍA ELÉCTRICA	ALUMBRADO
ENERGÍA ELÉCTRICA	CONEXIÓN DOMICILIARIA
ENERGÍA ELÉCTRICA	MEDIDOR
ENERGÍA ELÉCTRICA	RED DEFINITIVA
ENERGÍA ELÉCTRICA	TIENE RED PÚBLICA
Descripcion: MATERIAL VÍAS	
MATERIAL VÍAS	ADOQUIN
Descripcion: MEDIDOR DE AGUA	

Figura 87: Ventana Información de la infraestructura y servicios de un predio.
Fuente: Propia.

Modelo Relacional.

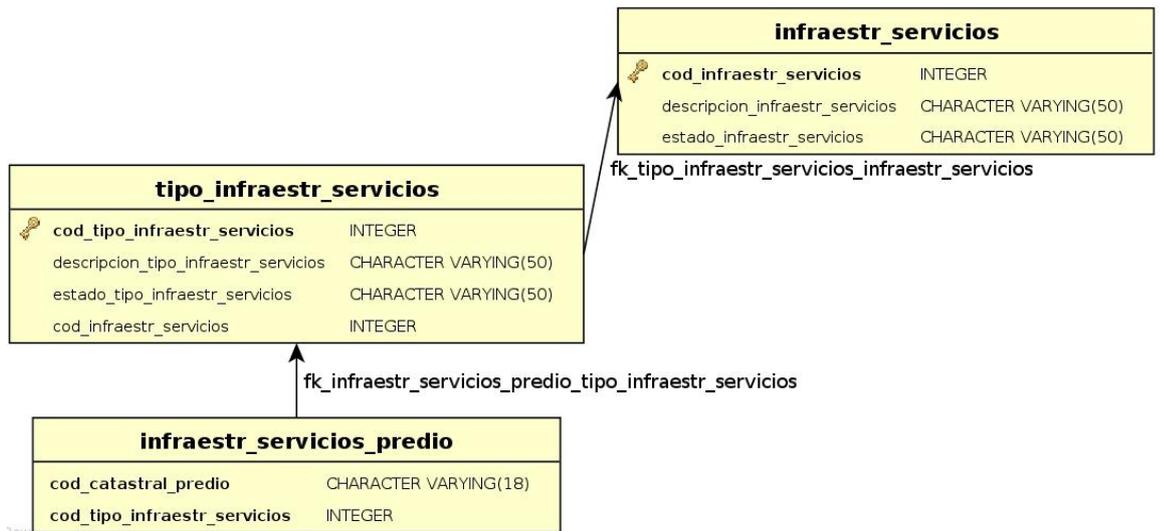


Figura 88: Modelo relacional para la consulta de la Infraestructura y Servicios.
Fuente: Propia.

- **Historia de Usuario 15:** Reporte de las propiedades de un ciudadano.
- **Tarea 1:** Reporte de las propiedades de un ciudadano.

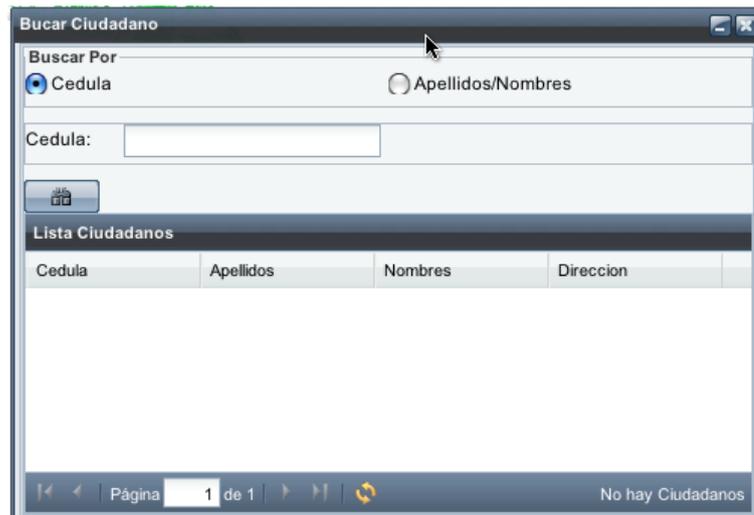


Figura 89: Ventana para la consulta de las propiedades de un ciudadano.
Fuente: Propia.

4.3.2.5. Quinta Iteración.

- **Historia de Usuario 16:** Consulta del impuesto predial.
- **Tarea 1:** Diseño y programación de la ventana de consulta del impuesto predial.

Información del Impuesto				
Predio: 100103030420013000				
C.I./RUP	Cod. Título	Nro. Título	Valor Título	Fecha Inc
1001052438	IP2	1734701	9.34	2012-01-1
1001052438	IP5	1550142	7.31	2011-01-1
1001052438	IP5	1376145	7.31	2010-01-1
1001052438	IP5	1353121	6.57	2009-01-1
1001052438	IP5	1202896	6.57	2008-01-1
1001052438	IP6	1052369	6.57	2007-01-1
1001052438	IP6	1	12.64	2006-01-1
1001052438	IP6	857904	6.57	2006-01-1
1001052438	IP5	751570	8.92	2005-01-1

Figura 90: Ventana Información del Impuesto Predial.
Fuente: Propia.

Reporte

Ilustre Municipio de Ibarra
Impuestos Prediales del Predio: 100103030420013000

Cédula/RUP	Cod. Titulo	Nro. Titulo	Valor Titulo	Fecha Ingreso	Fecha Vencimiento	Fecha Pago	Descuentos	Recargo	Intereses	Valor Exonerado	Estado
0001052438	IP2	1734701	9.34	2012-01-01	2012-12-31	2012-01-10	0.67	0	0	0	PAGADO
0001052438	IP5	1356142	7.31	2011-01-01	2011-12-31	2011-01-12	0.59	0	0	0	PAGADO
0001052438	IP5	1376145	7.31	2010-01-01	2010-12-31	2010-01-08	0.59	0	0	0	PAGADO
0001052438	IP5	1353121	6.57	2009-01-01	2009-12-31	2009-12-18	0	0.66	0	0	PAGADO
0001052438	IP5	1202896	6.57	2008-01-01	2008-12-31	2008-10-28	0	0.55	0	0	PAGADO
0001052438	IP6	1052369	6.57	2007-01-01	2007-12-31	2007-12-13	0	0.66	0	0	PAGADO
0001052438	IP6	5	12.64	2006-01-01	2006-12-31	2006-01-05	0	0	0	0	S/N
0001052438	IP6	857994	6.57	2006-01-05	2006-12-31	2006-09-21	0	0.49	0	0	PAGADO
0001052438	IP5	751570	8.92	2005-01-01	2005-12-31	2005-12-06	0	0.89	0	0	PAGADO

Figura 91: Reporte del impuesto predial.
Fuente: Propia.

Modelo Relacional

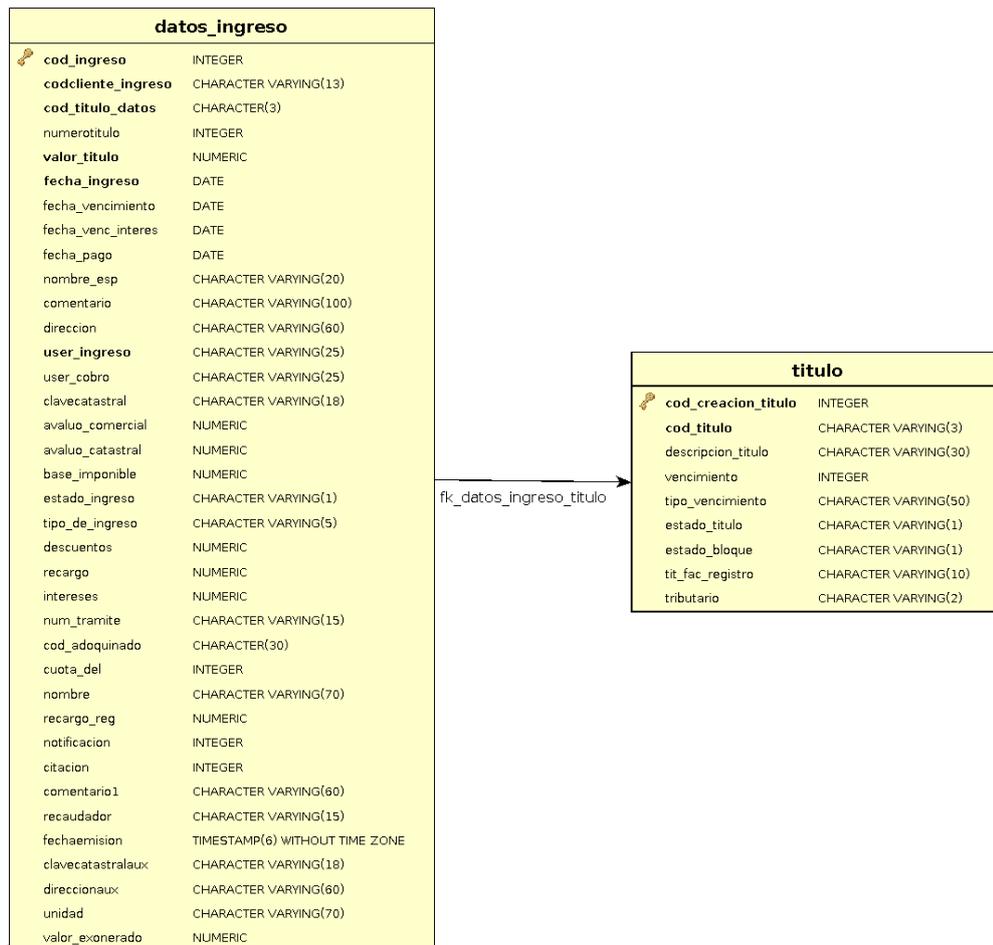


Figura 92: Modelo relacional para la consulta de los impuestos prediales.
Fuente: Propia.

- **Historia de Usuario 17:** Consulta de predios colindantes.
- **Tarea 1:** Diseño y Desarrollo de la ventana de consulta de los colindantes,



Figura 93: Ventana de consulta de los colindantes.
Fuente: Propia.

- **Historia de Usuario 18:** Impresión del Croquis de un predio.
- **Tarea 1:** Interfaz para la generación del croquis de un predio.



Figura 94: Ventana para la impresión de un croquis.
Fuente: Propia.

- **Tarea 2:** Programación del reporte de impresión del croquis.
- **Historia de Usuario 19:** Consulta del pago del impuesto predial por parroquias.

- **Tarea 1:** Generación de una capa dinámica para la consulta del impuesto predial por parroquia.

En la capa dinámica para consulta, se agregó un parámetro, el mismo que será enviado al momento de solicitar request GetMap del servicio WMS.

```
DATA "the_geom from (Select gid,nro_predio,clave_cata,the_geom from
gis.g_predio g left join gis.gis_datos_consulta p on
p.clavecatastral=clave_cata where p.estado='%estado%' and
p.session='%session%') as tabla using unique gid using srid=24877"
```

- **Tarea 2:** Diseño de la interfaz para la consulta de impuestos prediales por parroquia.



Figura 95: Ventana para la consulta de los impuestos prediales por parroquias.
Fuente: Propia.

Tarjetas CRC.

Tarjeta CRC	
Nombre de la Clase: CiudadanoLogic.	
Descripción: Clase Logic, lógica de los Ciudadanos.	
Colaboradores	
Super Clase:	
Requiere: CiudadanoTable (DAO), Ciudadano (DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
getCiudadanosPor	Devuelve un Array, con la información de los ciudadanos dependiendo del parámetro de búsqueda.

Tabla 126: Tarjeta CRC: CiudadanoLogic.
Fuente: Propia.

Tarjeta CRC	
Nombre de la Clase: DatosIngresoLogic.	
Descripción: Clase Logic, lógica del manejo de la información de los impuestos prediales.	
Colaboradores	
Super Clase:	
Requiere: DatosIngresoTable (DAO), DatosIngreso (DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
getImpuestoPredio	Devuelve un Array, con la información del impuesto predial.

**Tabla 127: Tarjeta CRC: DatosIngresoLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: DescripcionTerrenoPredioLogic.	
Descripción: Clase Logic, lógica del manejo de la información de la descripción de los terrenos.	
Colaboradores	
Super Clase:	
Requiere: DescripcionTerrenoPredioTable (DAO), DescripcionTerrenoPredio (DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
getDescripcionTerreno	Devuelve un Array, con la información de la descripción de un terreno.

**Tabla 128: Tarjeta CRC: DescripcionTerrenoPredioLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: InfraestrServiciosPredioLogic.	
Descripción: Clase Logic, lógica del manejo de la información de la infraestructura y servicios de un predio.	
Colaboradores	
Super Clase:	

Requiere: InfraestrServiciosPredioTable (DAO), InfraestrServiciosPredio (DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
getServiciosPredio	Devuelve un Array, con la información de la infraestructura y servicios de un predio.

**Tabla 129: Tarjeta CRC: InfraestrServiciosPredioLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: UsZonificacionUsoSueloLogic.	
Descripción: Clase Logic, lógica del manejo de la información de los usos de suelo de un predio.	
Colaboradores	
Super Clase:	
Requiere: UsZonificacionUsoSueloTable (DAO), UsZonificacionUsoSuelo (DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
getUsoSuelo	Devuelve un Array, con la información de los usos de suelo de un predio.

**Tabla 130: Tarjeta CRC: UsZonificacionUsoSueloLogic.
Fuente: Propia.**

Tarjeta CRC	
Nombre de la Clase: ZonificacionIbarraPrediosLogic.	
Descripción: Clase Logic, lógica del manejo de la información de la zonificación.	
Colaboradores	
Super Clase:	
Requiere: ZonificacionIbarraPrediosTable(DAO), ZonificacionIbarraPredios(DTO).	
Responsabilidades	
Métodos	
Nombre	Descripción
obtenerZonaPredio	Devuelve el nombre de la zona de un predio.

obtenerZonificacionPredio	Devuelve la información de la zonificación de un predio.
---------------------------	--

Tabla 131: Tarjeta CRC: ZonificacionIbarraPrediosLogic.
Fuente: Propia.

Tarjeta CRC	
Nombre de la Clase: giCatastrosManager.	
Descripción: Clase Manager, para el manejo de la información referente a Catastros.	
Colaboradores	
Super Clase:	
Requiere: ZonificacionIbarraPrediosLogic, DatosIngresoLogic, DescripcionTerrenoPredioLogic, CiudadanoLogic,	
Responsabilidades	
Métodos	
Nombre	Descripción
obtenerUsoSuelo	Devuelve un Array, con la información de los usos de suelo de un predio.
obtenerZonificacion	Devuelve un Array, con la información de la zonificación de un predio.
obtenerImpuestoPredio	Devuelve un Array, con la información del impuesto predial.
getDescripcionTerreno	Devuelve un Array, con la descripción de un terreno.
getCiudadanosPor	Devuelve un Array, con la información de los ciudadanos que cumplen con el parámetro de búsqueda.
getServiciosPredio	Devuelve un Array, con la información de los servicios e infraestructura de un predio.

Tabla 132: Tarjeta CRC: giCatastrosManager.
Fuente: Propia.

Tarjeta CRC
Nombre de la Clase: giCatastrosActions.

Descripción: Clase Controlador, permite el control de solicitudes de búsquedas de información, referente a Catastros.	
Colaboradores	
Super Clase: sfActions.	
Requiere: giCatastrosManager	
Responsabilidades	
Métodos	
Nombre	Descripción
executeGetPredioInfoPropietario	Devuelve en JSON, información del predio y del propietario.
executeGetFotoPredio	Devuelve en JPG, la foto de un predio.
executeGetDescFotoPredio	Devuelve en JSON, la descripción de las fotos de un predio.
executeBuscarPredioPor	Devuelve en JSON, la búsqueda de un predio.
executeGetColindantes	Devuelve en GeoJSON, la información de los colindantes de un predio.
executeGetZonificacion	Devuelve en JSON, la información de la Zonificación de un predio.
executeGetUsoSuelo	Devuelve en JSON, la información de los usos de suelo de un predio.
executeGetImpuesto	Devuelve en JSON, el impuesto predial.
executeImprimirCroquis	Devuelve en PDF, el croquis de un predio.
executeDescripcionTerreno	Devuelve en JSON, la descripción del terreno.
executeGetCiudadanosPor	Devuelve en JSON, la información de los ciudadanos.
executeGetImpuestoPredialPdf	Devuelve en PDF, información del impuesto predial.
executeGetServiciosPredio	Devuelve en JSON, información de la infraestructura y servicios de un predio

**Tabla 133: Tarjeta CRC: giCatastrosActions.
Fuente: Propia.**

4.3.3. Ejecución de Pruebas de Aceptación.

Una vez que el equipo de desarrollo ha implementado todas las historias de usuario que conformarán a la entrega del producto de software.

Previo a esta entrega, ahora es necesario realizar la ejecución de las pruebas de aceptación de modo que el cliente pueda comprobar que la implementación realizada ha cumplido con los requerimientos citados desde un principio.

Es así como a continuación se presenta la ejecución de las pruebas de aceptación para el Sistema GIS IMI.

GIS IMI					
Historia de Usuario		Código	Datos de Entrada	Resultado Esperado	Resultado de Prueba
Nro.	Nombre	Caso de Prueba			
1	Seguridad y Control de Acceso	1.1	<u>Caso 1</u> Usuario registrado. Usuario: CMANUEL Clave: manulc Clic ingresar.	Éxito	Éxito
			<u>Caso 2</u> Usuario no registrado. Usuario: ABC Clave: bha Clic ingresar.	Falla.	Falla
2	Repositorio de datos geográficos	2.1	<u>Caso:</u> Shapes.	Éxito	Éxito
3	Publicación de mapas en la web	3.1	<u>Caso:</u> Configuraciones.	Éxito	Éxito
4	Despliegue de Información geográfica y colección de capas	4.1	<u>Caso:</u> Acceso al Sistema.	Éxito	Éxito
5	Funciones de navegación y	5.1	<u>Caso:</u> Acceso al Sistema.	Éxito	Éxito.

	descriptor de leyendas				
6	Adición y eliminación de capas	6.1	Caso: Clic Agregar capas. Selección de capas: Propiedad Horizontal, Primer Piso y Segundo Piso. Clic Aceptar.	Éxito. Éxito.	Éxito. Éxito.
		6.2	Caso: Selección de capas visibles. Clic eliminar.	Éxito.	Éxito.
7	Estimación de longitud y área	7.1	Caso: Clic en la opción medir longitud.	Éxito.	Éxito.
		7.2	Caso : Clic en la opción medir área.	Éxito.	Éxito.
8	Ubicación rápida de información catastral	8.1	Caso: Clave catastral: 100104041 Clic buscar.	Éxito.	Éxito.
		8.2	Caso 1: Calle 1: Simón Bolívar. Calle 2: Pedro Moncayo. Clic buscar.	Éxito.	Éxito.
			Caso 2: Calle 1:-. Calle 2: Miguel Oviedo. Clic buscar.	Éxito.	Éxito.
			Caso 3: Calle 1: Av las Américas. Calle 2: Rio Chico. Clic buscar.	Falla.	Falla.
8.3	Caso: Lugares: Parques.	Éxito.	Éxito.		

			Clic buscar.		
9	Consulta de información predial	9.1	<u>Caso:</u> Clic en la opción información. Selección de capa Predial. Clic Predio con clave catastral: 100104040105016000.	Éxito.	Éxito.
10	Búsqueda de un predio	10.1	<u>Caso:</u> Predio Dibujado. Selección por Nombres. Apellidos/Nombres: Rodríguez/Ana. Clic buscar. Seleccionar predio. <u>Caso 2:</u> Predio no dibujado. Selección por Clave catastral. Clave Catastral: 100157010211067000 Clic buscar. Seleccionar predio.	Éxito. Falla.	Éxito. Falla.
11	Consulta de la Zonificación de un predio	11.1	<u>Caso 1:</u> En zona de Zonificación. Clic opción de consulta de Zonificación. Clic predio: 100104040607004000 <u>Caso 2:</u> No en una zona de Zonificación. Clic opción de consulta de Zonificación. Clic predio: 100103030106057000	Éxito. Falla.	Éxito. Falla.
12	Consulta de los	12.1	<u>Caso 1:</u>	Éxito.	Éxito.

	usos de suelo de un predio		<p>En zona de Zonificación.</p> <p>Clic opción de consulta de Usos de Suelo.</p> <p>Clic predio: 100104040607004000</p> <p>Caso 2:</p> <p>No en una zona de Zonificación.</p> <p>Clic opción de consulta de Usos de Suelo.</p> <p>Clic predio: 100103030106057000</p>	Falla.	Falla.
13	Consulta de la descripción del terreno	13.1	<p>Caso:</p> <p>Clic opción consulta de la descripción de un terreno.</p> <p>Clic predio: 100104040191007000</p>	Éxito.	Éxito.
14	Consulta de la infraestructura y servicios	14.1	<p>Caso:</p> <p>Clic Opción consulta de la infraestructura y servicios.</p> <p>Clic predio: 100104040191007000</p>		
15	Reporte de las propiedades de un ciudadano	15.1	<p>Caso:</p> <p>Buscar por Cédula.</p> <p>Nombres: Puma Tulcan / Rosa Matilde</p> <p>Selección.</p> <p>Clic en imprimir.</p>	Éxito.	Éxito.
16	Consulta del Impuesto Predial.	16.1	<p>Caso:</p> <p>Clic opción Impuesto.</p> <p>Clic Predio: 100104040192011000</p> <p>Clic en pdf.</p>	Éxito.	Éxito.
17	Consulta de los	17.1	<p>Caso:</p>	Éxito.	Éxito.

	predios colindantes.		Clic opción consulta de predios colindantes. Clic Predio: 100104040192011000		
18	Impresión de Croquis de un predio.	18.1	Caso: Clic opción imprimir croquis. Clic predio: 100104040192011000 Nombre: PREUBA. Descripción: Casa. Clic pdf.	Éxito.	Éxito.
19	Consulta del pago de impuestos predial por persona.	19.1	Caso: Clic opción consulta pago impuesto por parroquias. Parroquia: Ambuqui Año: 2010 Clic buscar.	Éxito.	Éxito.

Tabla 134: Ejecución de Pruebas de Aceptación.
Fuente: Propia.

4.4. Fase de Producción.

Llegando a este punto, el equipo de desarrollo ha implementado en su totalidad las iteraciones que conforman la entrega del Sistema GIS IMI y el producto se encuentra listo para la puesta en producción.

4.4.1. Plan de Implantación.

4.4.1.1. Requisitos para la implantación.

El equipo de desarrollo en base a un análisis técnico ha determinado los requisitos del Sistema GIS IMI son:

- Servidor de Base de datos Geográfica.

Característica	Valor
Memoria RAM	Mínimo 512

	Recomendable 3GB
Disco Duro	4GB Libres
Sistema Operativo	Debian o Ubuntu
Software Adicional	PostgreSQL, PostGIS, DBLINK

Tabla 135: Requisitos: Base de datos geográfica.

Fuente: Propia.

- Servidor de Mapas.

Característica	Valor
Procesador	Intel Xeon E5606
Memoria RAM	Mínimo 1GB Recomendable 4GB
Disco Duro	2GB Libres
Sistema Operativo	Debian o Ubuntu
Software Adicional	Apache, MapServer, Java JDK, Tomcat, Geoserver, Python, Tilecache.
Requisitos Adicionales	Salida directa al internet.

Tabla 136: Requisitos: Servidor de Mapas.

Fuente: Propia.

- Servidor de Aplicaciones.

Característica	Valor
Memoria RAM	Mínimo 512 Recomendable 2GB
Disco Duro	2GB Libres
Sistema Operativo	Debian o Ubuntu
Software Adicional	Apache, Php5, Symfony, Ext JS, GeoExt, OpenLayers, Python.

Tabla 137: Requisitos: Servidor de Aplicaciones.

Fuente: Propia.

- Requisitos del cliente.

Característica	Valor
Memoria RAM	Mínimo 512 Recomendable 2GB
Resolución del Cliente	800x600 mínimo. Recomendable 1024x768

Sistema Operativo	Windows o Linux
Software Adicional	Windows: Adobe Reader. Linux: Mozplugger. Navegadores: Firefox, Internet Explorer, Google Chrome, Opera, Safari.
Requisitos Adicionales	Conexión a internet o a la red local del Municipio.

Tabla 138: Requisitos: Del cliente.
Fuente: Propia.

4.4.1.2. Preparación para la instalación.

A continuación se realiza una verificación de los elementos del sistema, para que este tenga un buen funcionamiento en un ambiente de producción.

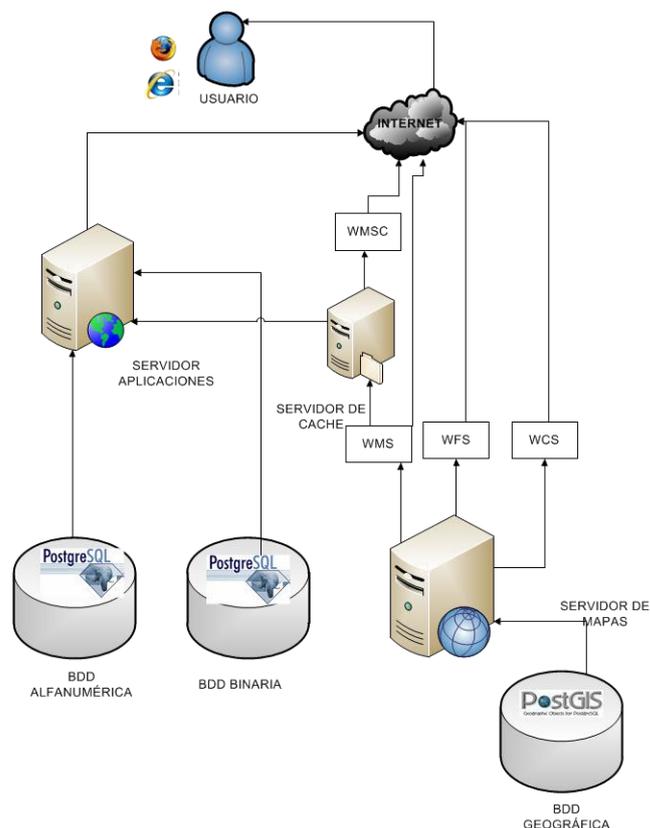


Figura 96: Entorno de Producción
Fuente: Propia.

4.4.1.3. Configuración del Aplicativo en un Ambiente de Producción.

- Configuración del Servidor Apache.
 - Instalación:

- En un terminal digitamos como usuario administrador: apt-get install apache2.
- Creación del Host Virtual.

```
NameVirtualHost *:80
Listen *:80
<VirtualHost *:80>
    DocumentRoot "/var/www/GisIbarra/web"
    DirectoryIndex index.php
    <Directory "/var/www/GisIbarra/web">
        Options FollowSymLinks
        AllowOverride All
        Allow from All
    </Directory>
    Alias /sf /var/www/libs/symfony-1.4.6-extjs/data/web/sf
    <Directory "/var/www/libs/symfony-1.4.6-extjs/data/web/sf">
        AllowOverride All
        Allow from All
    </Directory>
</VirtualHost>
```

- Configuración de la Aplicación.
 - Copiar la aplicación a /var/www.
 - Instalación del Framework Symfony.
 - Descargar el framework la versión 1.4.6.
 - Descomprimir el paquete.
 - Copiar en la carpeta /var/www/libs.
 - Configurar la ubicación del framework desde el aplicativo en el archivo GisIbarra/config/databases.yml, la línea.

Ejemplo de una conexión:

```
gis:
    class: sfDoctrineDatabase
    param:
        dsn: 'pgsql:host=172.17.4.108;dbname=bdgis'
        username: Usuariox
        password: xxxx
```



Capítulo V: Conclusiones y Recomendaciones

CONTENIDO:

1. Impactos.
2. Conclusiones.
3. Recomendaciones.

5. Capítulo V: Conclusiones y Recomendaciones.

5.1. Impactos.

- El Acceso y conocimiento de la información catastral por parte de los empleados de la Municipalidad ha aumentado notablemente como lo indica el siguiente gráfico.

Sistema	Acceso y conocimiento de la Información
GIS Anterior	13%
GIS IMI	35%

Tabla 139: Impacto: Porcentajes de Acceso y conocimiento de la Información.
Fuente: Propia (Datos de consumo de los servicios de Mapas).

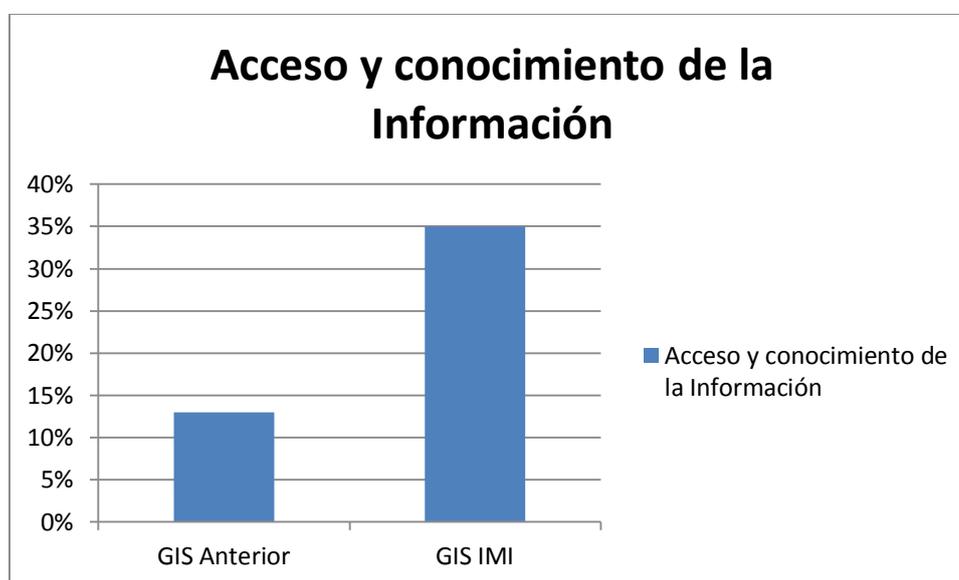


Figura 97: Cuadro estadístico del resultado del acceso y conocimiento a la información.
Fuente: Propia (Datos de consumo de los servicios de Mapas).

El 13% conforman solo el personal de las direcciones de Catastros y Avalúos, y Planificación, ya que estas direcciones eran las que generaban y registraban esta información. Gracias al nuevo sistemas GIS IMI se ha alcanzado a un 50% de empleados, ya que se ha ido integrando información de otras direcciones como es Participación Ciudadana, Obras públicas, y Tránsito y Transporte, con el fin de poder publicar desde la web información que es de su interés.

Además el hecho de ser un sistema web, esto hace posible que sea accesible desde el internet, y no solo se concentre en la red local del Municipio.

Otro aporte se ha realizado al Plan de Ordenamiento Territorial en 2011, el cual generó mucha información referente a uso y ocupaciones de suelo, estado de vías y aceras, etc. La misma información que gracias al Sistema GIS IMI fue accesible para otras personas e instituciones externas al Municipio.

También se ha aportado a la integración de información con EMAPA, a través de un acceso al Sistema. De esta forma EMAPA ha podido actualizar su información predial para luego tener una comunicación a través del código catastral.

Además los Bomberos utilizan el sistema para consultar ubicaciones de lugares de interés.

- Con el módulo PostGIS, es mucho más fácil realizar consultas de relaciones espaciales (Cruces, Unión, Intersección, Distancias, etc.), a través de SQL, sin la necesidad de utilizar un cliente pesado.

Ejemplo:

Consulta geográfica de los colindantes de Predios por segundo.

Al Consultar	GIS ANTERIOR	GIS IMI
1 Predio	7 s	3 s
2 Predios	13 s	5 s
5 Predios	49 s	13 s
8 Predios	74 s	17 s

Tabla 140: Resultado comparativo al consultar los colindantes de un predio por segundo.
Fuente: Propia (Consulta a la base de datos geográfica).

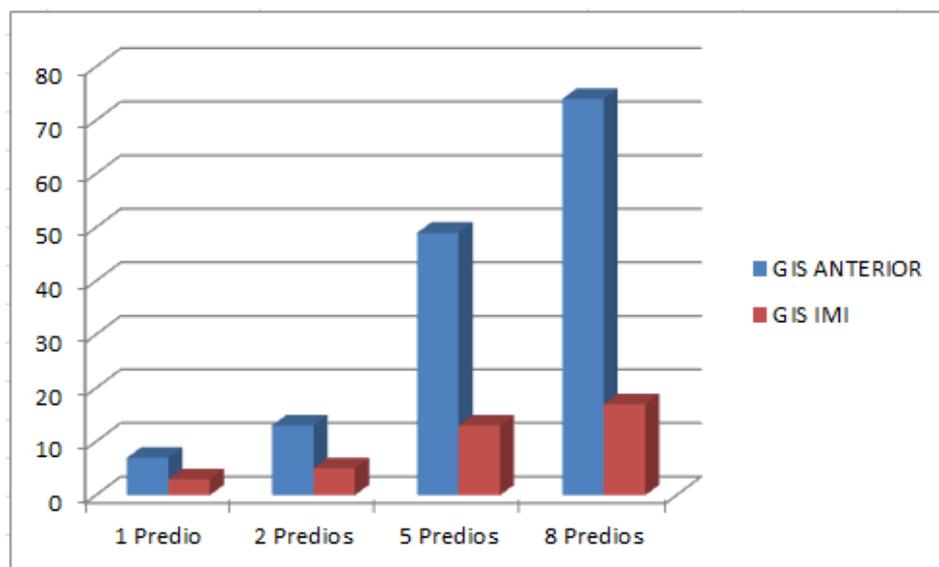


Figura 98: Cuadro estadístico del resultado comparativo al consultar los colindantes de un predio.
Fuente: Propia (Consulta a la base de datos geográfica).

Los datos y el gráfico anterior nos indican el tiempo en segundos transcurrido, en una consulta de intersección de objetos geográficos, al consultar que predios son colindantes a otro. Esta consulta se realizó sobre toda la capa predial tomando que esta contiene 74107 predios.

- Ahora se puede tener varios conceptos de consulta de información sobre un predio, ya que la información geográfica y alfanumérica se encuentra separada, mientras que anteriormente en un archivo shape se debía guardar los dos tipos de información en este formato.
- El GIS anterior visualizaba dos capas, el predial y las vías de Ibarra, ahora el sistema GIS IMI permite la adición de diferentes tipos de capas por ejemplo: Parroquias, Manzanas, Propiedad Horizontal, Primer Piso, Segundo Piso, Líneas de buses, etc.
- Facilidad al realizar cruces de información geográfica con la alfanumérica, a través de DBLINK y Secuencias Sql.

5.2. Conclusiones.

- Los estándares de la OGC permiten compartir información geográfica en el internet sin importar las plataformas o tecnologías de desarrollo, es la base para la interoperabilidad entre Sistemas de Información Geográfica.
- La utilización del módulo Postgis, hace que la información geográfica sea más manejable y de fácil análisis, a través del lenguaje SQL.
- La Arquitectura planteada que conforma: Servidor de Mapas, Servidor de Cache, Clientes Livianos y Pesados, es mucho más robusta y escalable que arquitecturas ya planteadas como por ejemplo: Mapserver y P.mapper (Servidor y Cliente respectivamente); siendo esta arquitectura la base para la construcción a futuro de un IDE (Infraestructura de Datos Espaciales).
- La utilización de software libre resulta beneficioso para la institución, dado que es un ahorro en el costo de licencias. Vea Sección 3.7.
- Un software desarrollado con XP, permite obtener documentada la información más relevante y de interés para un programador sin salirse a otros temas que alargan el entendimiento y el rápido análisis de un sistema ya desarrollado.
- Las tecnologías que tienen un enfoque RIA, permiten generar aplicaciones más livianas y que estas sean más amigables al usuario final.

5.3. Recomendaciones.

- Al momento de desarrollar un Sistema GIS, es un buen punto pensar en la infraestructura de hardware sobre la cual va funcionar, ya que son sistemas que requieren de muchos recursos de procesamiento y memoria.
- Es necesario guiar el desarrollo junto con los clientes o usuarios del sistema, dado que ellos tienen la experiencia en el manejo de la información y demás procesos.

- Hacer uso de buenas prácticas de programación y patrones de diseño, permiten implementar aplicaciones de fácil mantenimiento y sobre todo escalables.
- Utilizar estándares de consumo y publicación de información geográfica permitirá que a futuro esa información pueda ser compartida con otras instituciones que tengan interés sobre la misma.
- Una buena práctica es separar la información geográfica de la alfanumérica, ya que la información geográfica puede tener diferentes interpretaciones, las mismas que son solventadas a través de la información alfanumérica. A futuro los dos tipos de información son modificadas, es bueno contar con un medio para unir las, que me permita hacer referencia de la una a la otra.

GLOSARIO

API: Interfaz de Programación de Aplicaciones (Application Programming Interface) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Artefacto: Un artefacto es un producto tangible resultante del proceso de desarrollo de software. En ocasiones un artefacto puede referirse a un producto terminado, pero más habitualmente se refiere a la documentación generada a lo largo del desarrollo del producto.

CSS: El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets.

DBLINK: Funciones que permiten la conexión de dos bases de datos instalados en diferentes motores de PostgreSQL.

Escalabilidad: En telecomunicaciones y en ingeniería informática, la escalabilidad es la propiedad deseable de un sistema, una red o un proceso, que indica su habilidad para extender el margen de operaciones sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos.

Framework: Es un esquema (un esqueleto, un patrón) para el desarrollo y/o la implementación de una aplicación.

HTML: (Siglas de Hyper Text Markup Language) Lenguaje de marcado de hipertexto.

IDE: Infraestructura de datos Espaciales, es un conjunto de tecnologías, políticas y acuerdos institucionales; destinados a facilitar el acceso a la información espacial. Una base para la búsqueda, visualización, análisis y aplicación de datos espaciales a todos los niveles: administración, empresas, sectores sin ánimo de lucro, la universidad y los ciudadanos.

Java: Es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. La memoria es gestionada mediante un recolector de basura.

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.

JSON: Acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

Open Source: Código abierto, es el término con el que se conoce al software distribuido y desarrollado libremente.

ORM: Es el Mapeo Objeto-Relacional (Object-Relational Mapping) es una técnica de programación para convertir datos entre el lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia.

P.mapper: Es un framework para el desarrollo de aplicaciones GIS Web, en lenguaje php, dependiente de UMN Mapserver.

RIA: Rich Internet Application, son aplicaciones web ricas en el internet. Dan la posibilidad de que se pueda tener la mayoría de las características de las aplicaciones de escritorio tradicionales.

XML: Siglas en inglés de eXtensible Markup Language (lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).

Widget: Es un pequeño software que se emplea en la web y que puede ser instalado y ejecutado dentro de un navegador web por un usuario.

Usualmente son desarrollados en lenguajes DHTML, JavaScript o Flash.

BIBLIOGRAFÍA

LIBROS.

- **[LIB 1]:** Moreno Jiménez Antonio. (2da. Edición). *Sistemas y Análisis de la información Geográfica*.
- **[LIB 2]:** Ramón Jorge. (2009). *Ext JS 3.0 Cookbook*. Abhijeet Deobhakta.
- **[LIB 3]:** Open Geo. (2011). *Developing OGC Compliant Web Applications with GeoExt*. OpenPlans.
- **[LIB 4]:** Hazzard Erik. (2011). *OpenLayers 2.10 Beginner's Guide*. Aanchal Kumar.
- **[LIB 5]:** The MapServer Team. (2011) *MapServer Documentation Release 5.6.6*.
- **[LIB 6]:** GeoServer. (2011). *GeoServer User Manual Release 2.1.0*.
- **[LIB 7]:** Obe Regina, Hsu Leo. (2011). *PostGIS in Action*. Manning.
- **[LIB 8]:** *PostGIS 1.5.3 Manual*.
- **[LIB 9]:** De la Beaujardiere Jeff. (2002). *Web Map Service Implementation Specification*. Open Geospatial Consortium Inc.
- **[LIB 10]:** Vretanos Panagiotis A. (2005). *Web Feature Service Implementation Specification*. Open Geospatial Consortium Inc.
- **[LIB 11]:** Lalonde William. (2002). *Styled Layer Descriptor Implementation Specification*. Open Geospatial Consortium Inc.
- **[LIB 12]:** Portele Clemens. (2007). *OpenGIS Geography Markup Language (GML) Encoding Standard*. Open Geospatial Consortium Inc.
- **[LIB 13]:** Whiteside Arliss, Evans John D. (2008). *Web Coverage Service (WCS) Implementation Standard*. Open Geospatial Consortium Inc.
- **[LIB 14]:** Wilson Tim. (2008). *OGC KML*. Open Geospatial Consortium Inc.
- **[LIB 15]:** Percivall George. (2003). *OGC Reference Model*. Open Geospatial Consortium Inc.

PUBLICACIONES EN LÍNEA.

- **[WEB 1]:** *Servidor HTTP Apache*. http://es.wikipedia.org/wiki/Servidor_HTTP_Apache
- **[WEB 2]:** *Cliente pesado*. http://es.wikipedia.org/wiki/Cliente_pesado.
- **[WEB 3]:** (09/06/2009). *Comparación de clientes ligeros web para SIG*. http://wiki.osgeo.org/wiki/Comparaci%C3%B3n_de_clientes_ligeros_web_para_SIG.
- **[WEB 4]:** *CSS*. <http://www.librosweb.es/css/capitulo1.html>.
- **[WEB 5]:** *Hojas de estilo en cascada*.

- http://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada
- **[WEB 6]:** *Doctrine (PHP)*. http://es.wikipedia.org/wiki/Doctrine_%28PHP%29.
 - **[WEB 7]:** *Utilizando Doctrine como ORM en PHP*.
<http://web.ontuts.com/tutoriales/utilizando-doctrine-como-orm-en-php/>.
 - **[WEB 8]:** *Ext JS*. http://es.wikipedia.org/wiki/Ext_JS.
 - **[WEB 9]:** *Historia de Usuario*. http://es.wikipedia.org/wiki/Historias_de_usuario
 - **[WEB 10]:** *HTML*. <http://es.wikipedia.org/wiki/HTML>.
 - **[WEB 11]:** *JavaScript Características*.
http://www.uazuay.edu.ec/estudios/sistemas/lenguaje_iii/MAnualJavaScript/caracteristicas.htm
 - **[WEB 12]:** *JavaScript*. <http://es.wikipedia.org/wiki/JavaScript>.
 - **[WEB 13]:** *Map File*.
http://www.google.com.ec/url?sa=t&rct=j&q=referencia%20mapfile&source=web&cd=1&ved=0CFIQFjAA&url=http%3A%2F%2Fwww.freewebs.com%2FElk007007%2Fdownloads%2Freferencia_mapfile.pdf&ei=F7fTT5esEpHk6QHizrSTAw&usg=AFQjCNEINF_srI5baPeShMD_7CyaMH19rw&cad=rja
 - **[WEB 14]:** Steve Limem, Jeff McKenna, Jean-François Doyon. *MapFile*.
<http://mapserver.org/mapfile/index.html#mapfile>.
 - **[WEB 15]:** *MapScript*. <http://mapserver.org/mapscript/index.html#mapscript>
 - **[WEB 16]:** *Framework*. <http://es.wikipedia.org/wiki/Framework>.
 - **[WEB 17]:** *PHP*. <http://es.wikipedia.org/wiki/PHP>.
 - **[WEB 18]:** Javier Eguiluz. *Que es Symfony*. <http://www.symfony.es/que-es-symfony/>.
 - **[WEB 19]:** Paredes Manuel. (10-05-2009). *Que es Symfony*.
<http://www.tecnoretas.com/linux/que-es-symfony/>.
 - **[WEB 20]:** *Symfony*. <http://es.wikipedia.org/wiki/Symfony>.
 - **[WEB 21]:** *Symfony en pocas palabras*.
http://www.librosweb.es/symfony_1_0/capitulo1/symfony_en_pocas_palabras.html.
 - **[WEB 22]:** *Tomcat*. <http://es.wikipedia.org/wiki/Tomcat>.
 - **[WEB 23]:** *Ciclo de Vida y Fases*.
<http://programacion-extrema.wikispaces.com/5.+Ciclo+de+vida+y+fases>.
 - **[WEB 24]:** *Artefactos*. <http://programacion-extrema.wikispaces.com/7.+Artefactos>.
 - **[WEB 25]:** *Sistema de Información Geográfica*.
http://es.wikipedia.org/wiki/Sistema_de_Informaci%C3%B3n_Geogr%C3%A1fica
 - **[WEB 26]:** *ISO 19110:2005*. <http://www.construsur.net/index.php/catalogo/norma/iso/iso-19110:2005>.
 - **[WEB 27]:** *ISO 19110*. http://fr.wikipedia.org/wiki/ISO_19110.
 - **[WEB 28]:** *GeoWebCache*. <http://geowebcache.org/>.

- [WEB 29]: *GeoWebCache*. <http://opengeo.org/technology/geowebcache/>.
- [WEB 30]: *GeoWebCache*. <http://docs.geoserver.org/stable/en/user/geowebcache/index.html>.
- [WEB 31]: *TileCache*. <http://tilecache.org/>.
- [WEB 32]: *TileCache Getting Started*. <http://tilecache.org/docs/README.html>.
- [WEB 33]: *Deegree Info Sheet*. <http://www.osgeo.org/deegree>.
- [WEB 34]: *Geomajas*. <http://www.osgeo.org/geomajas>.
- [WEB 35]: *Geoserver*. <http://geoserver.org/display/GEOS/Welcome>.
- [WEB 36]: *MapBender Info Sheet*. <http://www.osgeo.org/mapbender>.
- [WEB 37]: *MapFish*. <http://www.osgeo.org/mapfish>.
- [WEB 38]: *MapGuide Open Source Project Info Sheet*. <http://www.osgeo.org/mapguide>
- [WEB 39]: *MapServer Info Sheet*. <http://www.osgeo.org/mapserver>.
- [WEB 40]: *OpenLayers Info Sheet*. <http://www.osgeo.org/openlayers>.
- [WEB 41]: *GRASS. Info Sheet*. <http://www.osgeo.org/grass>.
- [WEB 42]: *QGIS*. <http://www.osgeo.org/qgis>.
- [WEB 43]: *GvSIG Info Sheet*. <http://www.osgeo.org/gvsig>
- [WEB 44]: *Geotools*. <http://www.osgeo.org/geotools>
- [WEB 45]: *OSGEO*. <http://www.osgeo.org/>
- [WEB 46]: *OpenLayers*. http://live.osgeo.org/es/overview/openlayers_overview.html
- [WEB 47]: *Geomajas*. http://live.osgeo.org/es/overview/geomajas_overview.html
- [WEB 48]: *MapBender*. http://live.osgeo.org/es/overview/mapbender_overview.html
- [WEB 49]: *MapFish*. http://live.osgeo.org/es/overview/mapfish_overview.html
- [WEB 50]: *GeoMouse*. http://live.osgeo.org/es/overview/geomoose_overview.html
- [WEB 51]: *Postgis*. http://live.osgeo.org/es/overview/postgis_overview.html
- [WEB 52]: *QGIS*. http://live.osgeo.org/es/overview/qgis_overview.html.
- [WEB 53]: *GRASS*. http://live.osgeo.org/es/overview/grass_overview.html.
- [WEB 54]: *GvSIG*. http://live.osgeo.org/es/overview/gvsig_overview.html
- [WEB 55]: *Kosmo*. http://live.osgeo.org/es/overview/kosmo_overview.html.
- [WEB 56]: *OpenJump*. http://live.osgeo.org/es/overview/openjump_overview.html.
- [WEB 57]: *GeoServer*. http://live.osgeo.org/es/overview/geoserver_overview.html
- [WEB 58]: *MapServer*. http://live.osgeo.org/es/overview/mapserver_overview.html.
- [WEB 59]: *Deegree*. http://live.osgeo.org/es/overview/deegree_overview.html.
- [WEB 60]: *GeoTools*. http://live.osgeo.org/es/overview/geotools_overview.html.
- [WEB 61]: *MapGuide Open Source*.
http://live.osgeo.org/es/overview/mapguide_overview.html
- [WEB 62]: *GeoMouse*. <http://www.geomoose.org/>

- **[WEB 63]:** *Comparación de clientes ligeros web para SIG.*
<http://geotux.tuxfamily.org/index.php/es/component/k2/item/180-comparacion-de-clientes-ligeros-web-para-sig>
- **[WEB 64]:** Suárez Torres Alfonso, Suárez Villar Andrea, Vázquez Goyarzu Mónica, López Alberto. *Sistemas de Información Geográfica.* Pag: 6
- **[WEB 65]:** *Capítulo 4 Servidores de Mapas.* Pag: 2.
- **[WEB 66]:** Geo Bolivia. *MapBender.*
- **[WEB 67]:** *Udig.* <http://es.wikipedia.org/wiki/udig>.
- **[WEB 68]:** *Un GIS Framework para Eclipse.* <http://udig.refraction.net/home>.
- **[WEB 69]:** *User-friendly Desktop Internet GIG (uDig).*
http://live.osgeo.org/es/overview/udig_overview.html.
- **[WEB 70]:** *Open Source Mision.* <http://www.opensource.org/>
- **[WEB 71]:** *Software Libre.* http://es.wikipedia.org/wiki/Software_libre
- **[WEB 72]:** *Código Abierto.* http://es.wikipedia.org/wiki/C%C3%B3digo_abierto
- **[WEB 73]:** *Software libre y código abierto.*
http://es.wikipedia.org/wiki/Software_libre_y_de_c%C3%B3digo_abierto.
- **[WEB 74]:** *OpenGeo.* www.opengeo.org.
- **[WEB 75]:** *GIS.*
<http://www.geogra.uah.es/gisweb/1modulosespanyol/IntroduccionSIG/GISModule/GISTheory.doc>.
- **[WEB 76]:** *Modelo vectorial vs. raster.* www.sge.org/cartografia/sig2.pdf
- **[WEB 77]:** (2003). *Sistema de Referencia WGS-84.* [Wgs-84.pdf](#).
- **[WEB 78]:** Gilpérez Fraile Luis. (2005). *La importancia de consignar el Datum.* [AS15_Datum_en_los_mapas.pdf](#).
- **[WEB 79]:** *Generalidades del SIG.* [GIS.pdf](#).
- **[WEB 80]:** *XML.* <http://es.wiki.org/wiki/XML>.
- **[WEB 81]:** *GML.* <http://www.osgeo.org/gml>.



Anexos

CONTENIDO:

El contenido se encuentra en el CD adjunto.