



# UNIVERSIDAD TÉCNICA DEL NORTE

## FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

### CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

# ARTÍCULO

# ESPAÑOL

## 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:** Byron Mesias Cueva Cabrera.

**Director:** Ing. Mauricio Rea.

Ibarra – Ecuador

## RESUMEN GENERAL

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.

# CAPÍTULO I

## 1. 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”.

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 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 de IMI.**

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 de IMI.**

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.

## **1.3. 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 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.

## **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.

# **CAPÍTULO II**

## **2. MARCO TEÓRICO.**

### **2.1. Sistema de Información Geográfica.**

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.01]

## 2.2. Intercambio de datos Espaciales en la Web.

### 2.2.1. La Open Geospacial 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.

### 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. Como: XML, HTML, GML, KML, SVG y GeORSS.

### 2.2.3. Formatos Vectoriales.

Son formatos para codificación de una variedad de estructuras de datos geográficos expresados de forma vectorial. Como: GeoJSON y WKT.

### 2.2.4. OGC Web Services (OWS).

Los 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.02]

#### 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.

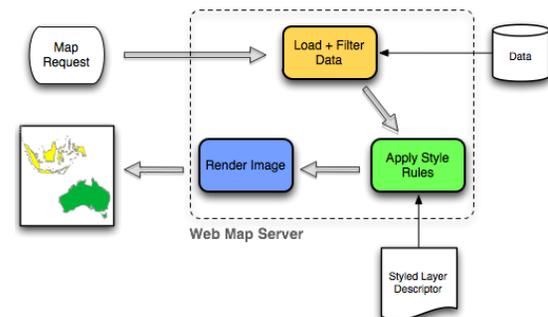


Figura 1 OCG WMS

Fuente: [www.01]

#### 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.

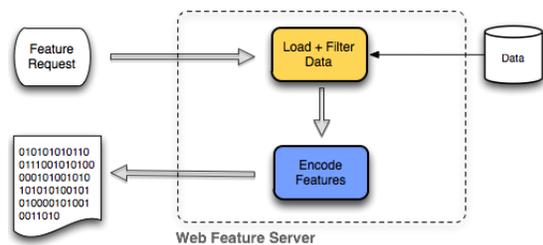


Figura 2 OCG WFS  
Fuente: [www.01]

### 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. [Lib.03]

## 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, 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.

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.

### 2.3.2. Base de Datos Espacial.

#### 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.

Su implementación está basada en geometrías e índices ligeros optimizados, para reducir el uso de disco y memoria.

## 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. [www.02]

### 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. [www.03]

## 2.5. Lenguajes de Programación.

### 2.5.1. CSS.

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. [www.04]

### 2.5.2. PHP.

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). [www.05]

### 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. [www.06]

### 2.5.4. 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. [www.07]

### 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. [www.08]

## 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. [www.09]

### 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. [www.10]

### 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. [www.11]

### 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. [www.12]

## 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 Pruebas.

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.

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 en 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. [www.13]

## 2.7.4. Fases.

### 2.7.4.1. Fase de Exploración.

En esta fase se define el alcance del proyecto. El cliente detalla 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. [www.13]

### 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. [www.14]

### 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.[www.13]

### 2.7.4.4. Fase de Producción.

Requiere pruebas 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. [www.13]

## 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. [www.14]

### 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 describimos que se trata de hacer en la tarea. [www.15]

### 2.7.5.3. Casos de Prueba de Aceptación.

Permite confirmar que la historia ha sido implementada correctamente. [www.15]

#### 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. [www.15]

### 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. [www.16].

#### 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. [www.17]

#### 2.8.3. Cliente 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.[www.18]

#### 2.8.4. Norma ISO 19110: Metodología para la creación de catálogo de fenómenos.

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.

## CAPÍTULO III

### 3. 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. [www.19].

#### 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.

### 3.3. Servidores de Mapas.

#### 3.3.1. Cuadro de Compatibilidad.

Servidor	Servidor OWS
Deegree	WMS, WFS, WCS, WFS -T
GeoServer	WMS, WFS, WFS-T, WCS
MapGuide	WMS, WFS
MapServer	WMS, WFS, WCS
QGIS Server	WMS

Tabla 2 Comparativa de Servidores de Mapas, Cuadro de Compatibilidad  
Fuente: Tesis (Resumen)

#### 3.3.2. Cuadro de Características Técnicas.

Servidor	Plataforma
Deegree	Windows, Linux.
GeoServer	Windows, Linux, Mac
MapGuide	Windows, Linux
MapServer	Windows, Linux, Mac, Solaris
QGIS Server	Windows, Linux, MacOSX

Tabla 2 Comparativa de Servidores de Mapas, Cuadro de Características Técnicas.  
Fuente: Tesis (Resumen)

#### 3.3.3. Cuadro de Funcionalidades y Características Principales.

Servidor	Mapas Temáticos	Simbología
Deegree	SI	SI
GeoServer	SI	SI
MapGuide	SI	SI
MapServer	SI	SI
QGIS Server	SI	SI

Tabla 3 Comparativa de Servidores de Mapas, Cuadro de Funcionalidades y Características Principales.  
Fuente: Tesis (Resumen)

#### 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, por lo 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	Cliente OWS
Geomajas	WMS, WFS, WFS - T
Mapbender	WMS, WFS-T
GeoExt	WMS, WFS, WFS-T, WCS
MapFish	WMS, WFS

OpenLayers	WMS, WFS, WCS, WFS-T
GeoMoose	WMS, WFS

Tabla 4 Comparativa de Clientes GIS Livianos, Cuadro de Compatibilidad  
Fuente: Tesis (Resumen)

### 3.4.2. Cuadro de Características Técnicas.

Cliente	Lenguaje Programación
Geomajas	Lenguaje: Java, Api: GWT (Google Web Toolkit)
Mapbender	Lenguaje: PHP, JavaScript. Api: JQuery y JQuery UI.
GeoExt	Lenguaje: Javascript
MapFish	Lenguaje: Java, Javascript, PHP, Python. Framework: Ext JS, Openlayers y GeoExt.
OpenLayers	Lenguaje: Javascript.
GeoMoose	Lenguaje: JavaScript, PHP.

Tabla 5 Comparativa de Clientes GIS Livianos, de Cuadro de Características Técnicas  
Fuente: Tesis (Resumen)

### 3.4.3. Cuadro de Funcionalidades y Características Principales.

Cliente	Dependencias de otros clientes
Geomajas	NO
Mapbender	-
GeoExt	OpenLayers
MapFish	Openlayers y GeoExt.
OpenLayers	NO
GeoMoose	Openlayers

Tabla 6 Comparativa de Clientes GIS Livianos, de Cuadro de Funcionalidades y Características Principales.  
Fuente: Tesis (Resumen)

### 3.4.4. Selección de Herramienta.

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
GvSIG	<ul style="list-style-type: none"> <li>• Ráster: ecw, ENVI hdr, ERDAS img, (Geo)TIFF.</li> <li>• Vectorial &amp; CAD: shapefile, GML, KML, DGN, DXF, DWG.</li> <li>• Bases de datos geográficas: PostGIS, MySQL, Oracle, ArcSDE.</li> </ul>

	<ul style="list-style-type: none"> <li>Datos remotos: ECWP, ArcIMS, estándares OGC.</li> </ul>
GRASS	<ul style="list-style-type: none"> <li>Ráster: PNG, TIFF.</li> <li>Vectorial: SVG, DXF.</li> <li>Bases de datos geográficas: SQLite, PostgreSQL.</li> </ul>
Kosmo Desktop	<ul style="list-style-type: none"> <li>Ráster: ECW, MrSID, Tiff, JPG, GIF, PNG, BMP.</li> <li>Vectorial: Shapefiles, GML, DXF, DWG, DGN, CSV (por extensión).</li> <li>Bases de datos geográficas: PostGIS, Oracle, MySQL.</li> </ul>
OpenJump	<ul style="list-style-type: none"> <li>Ráster: ECW, MrSID, JPG, TIFF, MIF.</li> <li>Vectorial: Shapefiles, GML, DXF.</li> <li>Base de datos geográfica: Postgis, ArcSDE, Oracle y MySQL.</li> </ul>
Quantum GIS	<ul style="list-style-type: none"> <li>Ráster: Tif, png</li> <li>Vectorial: shapefiles, MapInfo, SDTS y GML.</li> <li>Base de datos geográfica: Postgis.</li> </ul>
Udig	<ul style="list-style-type: none"> <li>Ráster: ECW, MrSID, JPEG, PNG, TIFF.</li> <li>Vectorial: Shapefiles.</li> <li>Base de datos geográfica: ArcSDE,</li> </ul>

	Oracle, DB2, Postgis.
--	-----------------------

Tabla 7 Comparativa de Clientes GIS Pesados, Cuadro de Compatibilidad

Fuente: Tesis (Resumen)

### 3.5.2. Cuadro de Características Técnicas.

Cliente	Plataformas
GvSIG	Windows, Linux, MacOS X
GRASS	Windows, Linux, MacOS X
Kosmo Desktop	Windows, Linux
OpenJump	Windows, Linux, Mac, Unix
Quantum GIS	Windows, Linux, Mac, Unix
Udig	Windows, Linux, Mac

Tabla 8 Comparativa de Clientes GIS Pesados, de Cuadro de Características Técnicas

Fuente: Tesis (Resumen)

### 3.5.3. Cuadro de Funcionalidades y Características Principales.

Cliente	Exportación de información vectorial a Postgis.
GvSIG	SI
GRASS	SI
Kosmo Desktop	SI
OpenJump	NO
Quantum GIS	SI (Plugin Spit)
Udig	NO

Tabla 9 Comparativa de Clientes GIS Pesados, de Cuadro de Funcionalidades y Características Principales.

Fuente: Tesis (Resumen)

### 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.

## CAPÍTULO IV

### 4. 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 ultimo 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 sistema pueda funcionar en un entorno de Producción.

#### 4.1. FASE DE EXPLORACIÓN.

##### Metáfora

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.

### Arquitectura del Sistema

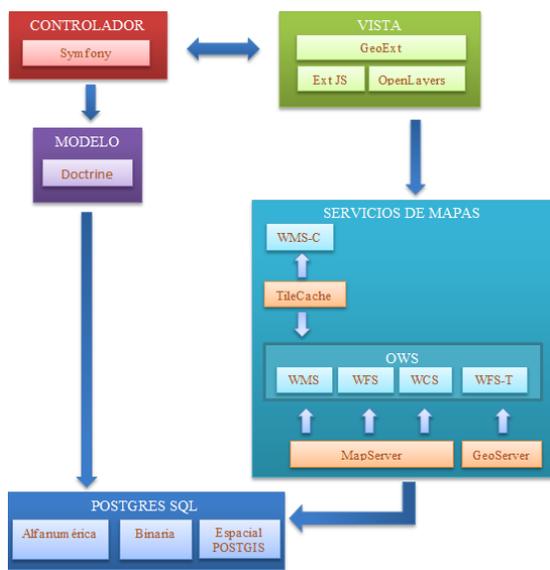


Figura 3 Arquitectura del Sistema Fuente: Propia

### Arquitectura Funcional del Sistema

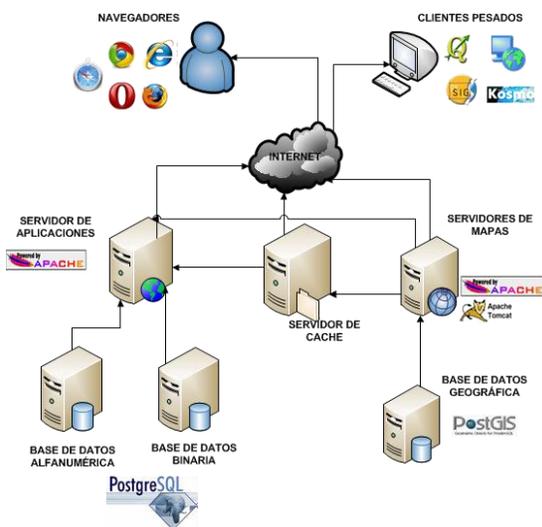


Figura 4 Arquitectura Funcional del Sistema Fuente: Propia

### 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 5 Módulos del Sistema Fuente: Propia

#### 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.

## Detalle de Historias de Usuario

- **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.

## 4.2. FASE DE PLANIFICACIÓN.

### Planificación por Historias.

Iteraciones	H. N°	Inicio	Fin
Primera	1	14/03/2011	23/03/2011
	2	24/03/2011	08/04/2011
Segunda	3	11/04/2011	29/04/2011
	4	02/05/2011	17/05/2011
Tercera	5	18/05/2011	23/05/2011
	6	24/05/2011	27/05/2011
	7	30/05/2011	02/06/2011
	8	03/06/2011	14/06/2011
	9	15/06/2011	20/06/2011
Cuarta	10	21/06/2011	22/06/2011

	11	23/06/2011	24/06/2011
	12	27/06/2011	28/06/2011
	13	29/06/2011	30/06/2011
	14	01/07/2011	04/07/2011
	15	05/07/2011	06/07/2011
Quinta	16	07/07/2011	08/07/2011
	17	11/07/2011	12/07/2011
	18	13/07/2011	14/07/2011
	19	15/07/2011	20/07/2011

Tabla 10 Planificación por Historias.

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.

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.

#### 4.4. FASE DE PRODUCCIÓN.

A continuación se realiza una verificación de los elementos del sistema, para que este tenga un buen funcionamiento en un ambiente de producción.

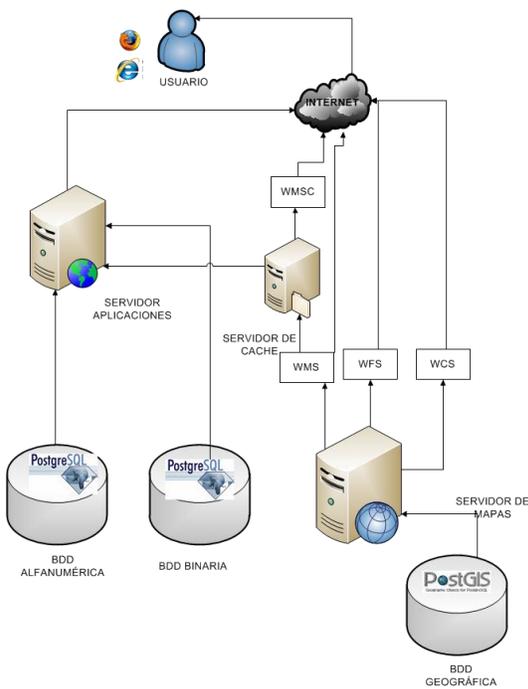


Figura 6 Entorno de producción  
Fuente: Propia

## 5. 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 11 Acceso y conocimiento de la Información.  
Fuente: Propia.

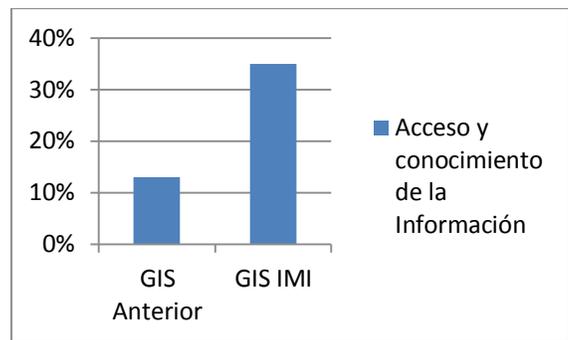


Figura 7 Acceso y conocimiento de la Información.  
Fuente: Propia

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 35% de empleados, ya que se ha ido integrando información de otras direcciones como es Participación Ciudadana, Obras públicas, y Transito y Trasporte, 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 10 Consulta de los colindantes de un predio.  
Fuente: Propia.

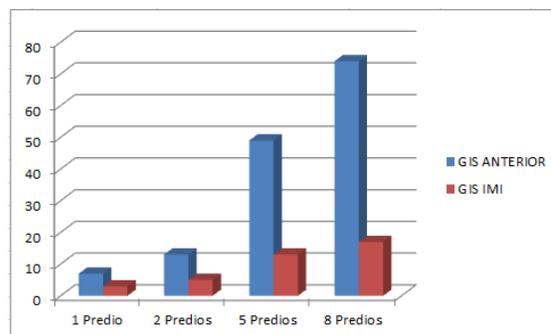


Figura 8 Consulta de los colindantes de un Predio.  
Fuente: Propia

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.

## 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 utilización de software libre resulta beneficioso para la institución, dado que es un ahorro en el costo de licencias.
- 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.

### 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.

## 6. BIBLIOGRAFIA Y REFERENCIAS

### 6.1. SITIOS WEB

[www.01]  
OpenGEO Estándares OGC.  
[www.opengeo.org](http://www.opengeo.org)

[www.02]  
Servidor HTTP Apache.  
[http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache)

[www.03]  
Tomcat  
<http://es.wikipedia.org/wiki/Tomcat>.

[www.04]  
CSS  
<http://www.librosweb.es/css/capitulo1.html>.

[www.05]  
PHP  
<http://es.wikipedia.org/wiki/PHP>.

[www.06]  
JavaScript  
<http://es.wikipedia.org/wiki/JavaScript>

[www.07]  
MapScript  
<http://mapserver.org/mapscript/index.html#mapscript>

[www.08]

MapFile  
[http://www.google.com.ec/url?sa=t&rct=j&q=referencia%20mapfile&source=web&cd=1&ved=0CFIQFjAA&url=http%3A%2F%2Fwww.free-webs.com%2Felk007007%2Fdownloads%2Freferencia\\_mapfile.pdf&ei=F7fTT5esEpHk6QHlzsTAW&usg=AFQjCNEINF\\_srI5baPeShMD\\_7CyaMH19rw&cad=rja](http://www.google.com.ec/url?sa=t&rct=j&q=referencia%20mapfile&source=web&cd=1&ved=0CFIQFjAA&url=http%3A%2F%2Fwww.free-webs.com%2Felk007007%2Fdownloads%2Freferencia_mapfile.pdf&ei=F7fTT5esEpHk6QHlzsTAW&usg=AFQjCNEINF_srI5baPeShMD_7CyaMH19rw&cad=rja)

[www.09]  
Framework  
<http://es.wikipedia.org/wiki/Framework>

[www.10]  
Que es Symfony.  
<http://www.symfony.es/que-es-symfony/>.

[www.11]  
Doctrine (PHP)  
[http://es.wikipedia.org/wiki/Doctrine\\_%28PHP%29](http://es.wikipedia.org/wiki/Doctrine_%28PHP%29)

[www.12]  
Ext JS  
[http://es.wikipedia.org/wiki/Ext\\_JS](http://es.wikipedia.org/wiki/Ext_JS).

[www.13]  
Ciclo de Vida y Fases  
<http://programacion-extrema.wikispaces.com/5.+Ciclo+de+vida+y+fases>

[www.14]  
Ciclo de Vida y Fases  
<http://programacion-extrema.wikispaces.com/5.+Ciclo+de+vida+y+fases>

[www.15]  
Artefactos  
<http://programacion-extrema.wikispaces.com/7.+Artefactos>

[www.16]  
Capítulo 4 Servidores de Mapas

[http://es.wikipedia.org/wiki/Cliente\\_pesado](http://es.wikipedia.org/wiki/Cliente_pesado)

[www.17]  
Comparación de clientes ligeros en la Web  
[http://wiki.osgeo.org/wiki/Comparaci%C3%B3n\\_de\\_clientes\\_ligeros\\_web\\_para\\_SIG](http://wiki.osgeo.org/wiki/Comparaci%C3%B3n_de_clientes_ligeros_web_para_SIG).

[www.18]  
Cliente Pesado  
[http://es.wikipedia.org/wiki/Cliente\\_pesado](http://es.wikipedia.org/wiki/Cliente_pesado).

[www.19]  
Codigo Abierto  
[http://es.wikipedia.org/wiki/C%C3%B3digo\\_abierto](http://es.wikipedia.org/wiki/C%C3%B3digo_abierto)

## 6.2. LIBROS

[Lib.01]  
Moreno Jiménez Antonio. (2da. Edición).  
*Sistemas y Análisis de la información Geográfica.*

[Lib.02]  
The MapServer Team. (2011) *MapServer Documentation Release 5.6.6.*

[Lib.03]  
Whiteside Arliss, Evans John D. (2008). *Web Coverage Service (WCS) Implementation Standard.* Open Geospatial Consortium Inc.