

# Tesis de Ingeniería en Sistemas Computacionales "Estudio e Implementación de un sistema GIS Web para el Sistema de Avalúos y Catastros del Ilustre Municipio de Ibarra" 2010-2012

## Implementación de un sistema GIS Web para el Sistema de Avalúos y Catastros del Ilustre Municipio de Ibarra

Byron Cueva Cabrera  
Universidad Técnica del Norte  
Carrera de Ingeniería en Sistemas Computacionales  
Byron7\_cueva@yahoo.es

***Abstract.** The investigation is focused in the standards published by the Open Geospatial Consortium OGC, the OGC Web Services (OWS) that are specifications for the use of the geographical information through the Web, it allows that the systems have bigger interoperability. OWS is the generic name with which you/they group all the standards (WMS, WFS, WCS, CSW, WPS, etc.). When developing the system you model through architecture MVC, also architecture guided to applications GIS, with its elements that are: server of maps, light client, heavy client and cache server, allowing us to obtain a more robust architecture.*

### 1 Instrucciones

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.

#### 1.1 Herramientas para el desarrollo del Software

##### Desarrollo.

**Symfony 1.4.6:** 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

**Ext JS 3.4:** Es una biblioteca de JavaScript para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX, DHTML y DOM. Fue desarrollada por Sencha.

**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.

##### 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. En el proyecto se utilizó como servidores de mapas a: MapServer y GeoServer.

##### Cientes 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. Para el proyecto se utilizó a: OpenLayers y GeoExt.

### **Cientes 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. En el proyecto se utilizó: GvSIG, Kosmo, Qgis y Udig.

## **3 Proceso de Desarrollo**

La implementación del sistema se encuentra amparada por una metodología de desarrollo de software ágil, Programación Extrema XP cual se encuentra 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.

### **3.1 Artefactos de la Metodología de Software**

#### **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.

#### **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.

#### **Casos de Prueba de Aceptación.**

Permite confirmar que la historia ha sido implementada correctamente.

#### **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.

**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.

## **3.2. Arquitectura del Sistema**

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

**Modelo:** En esta capa se utiliza 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 utiliza el framework Symfony 1.6, si bien este ya cuenta con el patrón MVC, pero este fue 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 permite crear una interfaz de escritorio, la misma que es de fácil uso y más atractiva asía el usuario. También nos permite optimizar el rendimiento a través del control de solicitudes manejadas por Ajax.

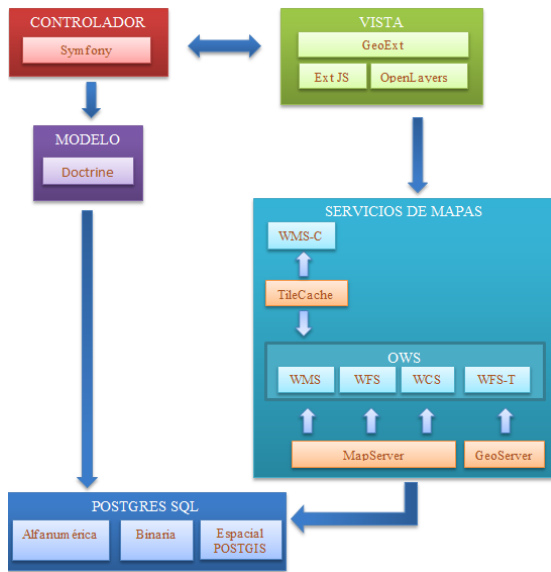
OpenLayers se utilizó para el consumo de los servicios OWS, visualización de mapas y generador de información geográfica en formato SVG.

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 utilizó Postgresql, la misma que utilizan todos los sistemas. 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 utilizó: 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.



### 3.3 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:



**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. 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.
- Las tecnologías que tienen un enfoque RIA, permiten generar aplicaciones más livianas y que estas sean más amigables al usuario final.

## Referencias

- [1] Moreno Jiménez Antonio. (2da. Edición). *Sistemas y Análisis de la información Geográfica*.
- [2] Ramón Jorge. (2009). *Ext JS 3.0 Cookbook*. Abhijeet Deobhakta..
- [3] Open Geo. (2011). *Developing OGC Compliant Web Applications with GeoExt*. OpenPlans.