

TRABAJO FINAL DE CARRERA

Aplicación web para la monitorización en el espacio-tiempo de enfermedades.

11/06/2012

Universitat Oberta de Catalunya

Alumno: Ivan Marques Marques

Dirigido por: Laura Díaz Sanchez



Universitat Oberta
de Catalunya

Licencias

Memoria del trabajo final de carrera, Trabajo práctico i Video-presentación (Creative Commons)

Este TFC está sujeto a una licencia de Reconocimiento 3.0 No adaptada de Creative Commons, podéis copiarlo, distribuirlo i transmitirlo públicamente siempre que se cite al autor y la obra.

La licencia completa puede ser consultada en:

<http://creativecommons.org/licenses/by/3.0/deed.ca>

*A mi pareja, Eva, por aguantar esos momentos de estrés y dedicación a este trabajo.
A mis padres, Diego y Marga, por aquel día que a pesar de la distancia que nos iba a
separar, decidieron darme el empujón necesario para empezar mi nueva vida.
A mi consultora, Laura Díaz, por sus ánimos y consejos diarios.*

1. Resumen

Este documento forma parte del proyecto final de carrera de la carrera de ingeniería técnica en informática de sistemas. En el mismo se realizara un estudio sobre los sistemas SIG (Sistemas de información geográfica) y como aplicarlos en casos específicos. Concretamente como aplicar los conocimientos adquiridos sobre las tecnologías SIG en la realización de un análisis y la visualización sobre un mapa interactivo de los datos proporcionados por el ASPB (Agencia de Salud Publica de Barcelona).

Podemos dividir este trabajo en 3 etapas siguiendo el ciclo de vida clásico (análisis – diseño - implementación):

En primer lugar, se ha realizado un amplio análisis previo sobre las tecnologías SIG y su implicación en el mundo actual. Esta primera etapa ha sido de las más complejas debido al gran desconocimiento existente sobre dicha materia. Durante el desarrollo de la misma se han adquirido las diferentes competencias necesarias para llevar el proyecto a buen fin.

En una segunda etapa se ha realizado el diseño funcional de la aplicación, el análisis de los requisitos representado mediante casos de uso y diagrama de secuencia, así como el estudio de las diversas herramientas utilizadas y su justificación ante posibles variantes. Durante esta etapa se han analizado diversas tecnologías (*ArcGIS, OpenMaps, Geoserver, etc.*) comparando los beneficios que otorgan cada una de ellas y eligiendo la que mejor se adapta a nuestro propósito. De igual forma se han analizado diferentes entornos de trabajo y herramientas de desarrollo (Eclipse, Flex Developer, etc.) con el fin de agilizar la etapa de implementación.

Finalmente, la etapa de implementación puede dividirse en diversas sub-etapas (implementación – pruebas - mantenimiento) durante las cuales se ha realizado una aplicación en su primera versión beta. En esta última etapa se ha puesto en práctica el conocimiento adquirido y se ha desarrollado las primeras versiones beta de la aplicación. Durante este proceso codificaremos los WS (*Web Services*), la base de datos y la aplicación y probaremos su funcionamiento.

Contenido

Licencias	2
1. Resumen.....	4
2. Índice de Ilustraciones.....	7
3. Descripción del proyecto.....	8
3.1 Introducción	8
3.2 Objetivos	8
3.3 Alcance	9
3.4 Enfoque del trabajo.....	9
4. Organización y etapas del proyecto	10
4.1 Metodología de trabajo	10
4.2 Relación de las tareas a realizar	10
4.2.1 Pac 1	10
4.2.2 Pac 2	10
4.2.3 Pac 3	11
4.2.4 Pac 4	12
4.2.5 Debate Virtual	13
5. Sistemas de información geográfica	14
5.1 ¿Qué es un sistema de información geográfica?	14
5.2 Introducción al Web-GIS	15
5.2.1 Arquitectura Cliente-servidor.....	17
5.3 Datos espaciales.....	18
5.3.1 Representación de datos espaciales	18
5.3.2 Almacenamiento de datos en sistemas SIG	19
5.4 Estándares de los SIG	20
5.4.1 Geographic Markup Language (GML)	21
5.4.2 Web Feature Service (WFS).....	21
5.4.3 Web Map Service (WMS)	22
6. Tecnología utilizada.....	22
6.1 PostGIS	22
6.2 GeoServer.....	23
6.3 GeoExt (OpenLayers).....	24
7. Análisis.....	25
7.1 Análisis funcional.....	25

7.1.1	Casos de uso	25
7.1.2	Diagramas de secuencia de la aplicación	26
7.1.3	Diagramas de estado y Diagrama de Actividad de la aplicación.	30
8.	Diseño Técnico e implementación de la Aplicación Web-GIS.	32
8.1	Base de datos espacial	33
8.1.1	Diseño BD y modelado de datos	34
8.1.2	Carga de datos.....	35
8.1.3	Futuro mantenimiento de los datos.....	35
8.2	GeoServer, servicios web de tratamiento y gestión de datos.	36
8.3	Aplicación cliente	37
8.3.1	Llamada y uso del servicio web WMS	38
8.3.2	Llamada y uso del servicio web WFS.....	39
8.3.3	Creación de la interfaz de usuario IU	40
9.	Conclusiones.....	42
10.	Futuras ampliaciones del trabajo	43
11.	Glosario	44
12.	Bibliografía.....	46
12.1	Documentos	46
12.2	URL's.....	46
13.	Anexos	47

2. Índice de Ilustraciones

Ilustración 1 - Diagrama de Grantt.....	13
Ilustración 2 - Capas de sistemas SIG	14
Ilustración 3 - Arquitectura de un SIG según OGC	16
Ilustración 4 Diagrama de flujo para el desarrollo de aplicaciones WEB-GIS	17
Ilustración 5 - Modelo Vista Controlador.....	18
Ilustración 6 - Representación geográfica (Vectorial vs Raster)	19
Ilustración 7 - Esquema genérico del aplicativo.....	22
Ilustración 8 - Estructura Base de Datos PostgreSQL.....	23
Ilustración 9 Diagrama de casos de uso de la aplicación	26
Ilustración 10 Diagrama de Secuencia de la aplicación	27
Ilustración 11 Diagrama de secuencia Activar/Desactivar Capas	27
Ilustración 12 Diagrama de secuencia de función Zoom	28
Ilustración 13 Diagrama de secuencia Función Full Extended	28
Ilustración 14 Diagrama de secuencia Función Pan.....	29
Ilustración 15 Diagrama de secuencia de Identificador de Objeto	29
Ilustración 16 Diagrama de secuencia de filtrado de información	30
Ilustración 17 Diagrama de estados del servicio.....	31
Ilustración 18 Diagrama de estado del cliente.....	31
Ilustración 19 Diagrama de Actividad de la aplicación.....	32
Ilustración 20 Arquitectura base de la aplicación	32
Ilustración 21 - Modelo de datos de la BBDD	34
Ilustración 22 - Carga de datos en BBDD espacial.....	35
Ilustración 23 - Uso del aplicativo para el mantenimiento de los datos.....	36
Ilustración 24 - Esquema global de un servicio web mediante GeoServer	37
Ilustración 25 - Petición WMS del tipo "GetMap"	39
Ilustración 26 - Peticion WMS del tipo "GetFeatureInfo".....	39
Ilustración 27 - Interfaz por defecto de OpenLayer	41
Ilustración 28 - Codigo OpenLayer	41

3. Descripción del proyecto

Este capítulo expone el marco general en que se ha desarrollado el presente Trabajo Final de Carrera (TFC). A continuación se presentan los objetivos que se pretende alcanzar, la planificación de las principales tareas y la metodología utilizada para su correcta ejecución.

3.1 Introducción

Desde hace diversos años se utiliza el sistema de Vigilancia Epidemiológica (VE)¹ para la monitorización de las enfermedades y su comportamiento en el espacio y en el tiempo para detectar cambios y patrones que harían saltar algún tipo de alarma. En la actualidad, la VE es una herramienta básica para los programas de prevención y control sanitarios, y por lo tanto, tiene en cuenta todas las actividades necesarias para un control eficiente sobre las enfermedades. Pero esto solo es útil si puede ofrecer la información reconocida en el momento y en el lugar que se necesite.

Actualmente muchos sistemas de monitorización de enfermedades son capaces de producir informes textuales con gráficos que permiten describir las diversas incidencias del sector, pero en la actualidad aun no explotan de forma completa los beneficios de las técnicas geo-espaciales.

Este es el caso de la Agencia de Salud Pública de Barcelona (ASPB) encargada de la monitorización de enfermedades como la tuberculosis o el SIDA. La ASPB recoge datos desde hace más de 20 años, pero sin explotar totalmente las técnicas geo-espaciales. Debido a esto no es posible dar una perspectiva geográfica y proporcionar estudios avanzados como la monitorización de enfermedades en el espacio y en el tiempo con el fin de obtener datos referentes a la aparición o propagación de enfermedades.

Este proyecto se centrará en mejorar esta ausencia y por lo tanto mejorar el sistema VE mediante el uso de técnicas SIG (Sistemas de Información Geográfica), de esta forma se intentará facilitar las tareas de interpretación y control por parte de los expertos de la ASPB.

3.2 Objetivos

El objetivo principal de este trabajo es la creación de una aplicación WEB SIG interactiva, que permita a los expertos de la ASPB visualizar los datos de vigilancia de enfermedades de forma gráfica mediante un geo-procesamiento previo.

Una vez concluido el trabajo de final de carrera deberíamos haber alcanzado de forma eficaz los siguientes objetivos:

- Comprender los conceptos de la tecnología SIG, su metodología y sobretodo conocer los beneficios de implementar dicha tecnología en diversos problemas sociales.
- Conocer la estructura de los diferentes tipos de datos.
- Conocer la arquitectura y los componentes necesarios para desarrollar aplicaciones WEB SIG, así como el lenguaje de programación necesario para su implementación.
- Saber localizar, generar y manipular datos geográficos.

¹ Según la Organización Mundial de la Salud (OMS), la vigilancia epidemiológica es el escrutinio permanente y la observación activa de la distribución y propagación de las infecciones y de los factores relacionados con gran exactitud de calidad y cantidad para un control eficaz.

- Saber elaborar un proyecto referente a tecnologías SIG (organización, planificación, capacidad de síntesis).
- Demostrar conocimientos prácticos con herramientas de análisis espacial y transformación como *ESRI ArcGIS Online*, *GeoServer*, *OpenLayers*.
- Demostrar conocimientos prácticos en la elaboración de un portal WEB SIG y en la creación de las bases de datos necesarias para la elaboración del proyecto.

3.3 Alcance

Se considera dentro del alcance de este proyecto los siguientes puntos clave:

- Realización de un **estudio previo de la tecnología SIG** y su metodología, así como los diferentes conceptos relacionados con la misma (topología, cartografía, generación de datos, etc.).
- Realización de **un estudio previo de los sistemas VE**, así como su potencial dentro del sector de la sanidad.
- **Recopilación de toda la información necesaria** (datos de la enfermedad a llevar a estudio).
- **Creación de la base de datos** necesaria para la muestra de datos.
- Redacción de la memoria final del proyecto y elaboración de la presentación final (video).

3.4 Enfoque del trabajo

El trabajo realizado va a seguir las fases propias de un proyecto de Ingeniería de *Software*.

Fase de Análisis en la que se establecen los requisitos del cliente y se especifican los casos de uso asociados a los requerimientos funcionales.

Fase de diseño en la que se especificara la tecnología elegida, la arquitectura del trabajo y el diseño de la aplicación y la base de datos.

Fase de implementación en la cual realizaremos la versión beta de nuestra aplicación mediante las tecnologías elegidas (*Geoserver*, *Openlayers*) utilizando lenguajes de programación como Java y *Javascript* y utilizando como IDE Eclipse desarrollaremos todo el código fuente necesario para implementar los diferentes requerimientos.

Fase de pruebas, donde se realizaran las pruebas unitarias para comprobar el correcto funcionamiento de la aplicación.

4. Organización y etapas del proyecto

Los sistemas SIG son una materia un poco desconocida debido a su especialidad, a su ausencia dentro del plan de estudios ordinarios de la carrera de informática y a su ámbito general en diversas especialidades como la sociología, la topografía, la ingeniería, etc. Debido a estos inconvenientes el esfuerzo para entender los conceptos SIG ha sido mayor de lo esperado.

4.1 Metodología de trabajo

Una vez alcanzada la idea general del proyecto y aprendido los conocimientos básicos para realizar el análisis funcional. Se ha dividido el trabajo en las siguientes etapas que detallaremos durante el transcurso de este documento:

- Análisis funcional.
- Diseño Técnico y implementación de la aplicación
 - Carga de la información y los datos necesarios para la creación de una base de datos geo-espacial.
 - Creación de un servicio web que proporcione los diferentes datos almacenados.
 - Creación de una aplicación web, que nos permita visualizar y gestionar la información.
- Pruebas unitarias de la aplicación.

Gracias a esta estructura nos podemos centrar en el modelo *divide and conquer*² y realizar un análisis, diseño y implementación de cada uno de ellos. De esta forma obtendremos un producto reutilizable.

4.2 Relación de las tareas a realizar

A continuación vamos a proceder a describir cada una de las tareas que deberán realizarse durante el transcurso de este proyecto, así como su fecha estimada de inicio y fin.

4.2.1 Pac 1

Creación del documento "Plan de trabajo"		Horas: 18h
Inicio: 03/03/2012	fin: 13/03/2012	Hojas:
Planificación del proyecto, cálculos de tiempos y requerimientos y elaboración del documento "Plan de trabajo".		

Entrega del documento "Plan de trabajo"		Horas:
Inicio: 13/03/2012	fin: 13/03/2012	Hojas:

4.2.2 Pac 2

Configuración del equipo		Horas: 2h
Inicio: 14/03/2012	fin: 14/03/2012	Hojas:
Puesta a punto del equipo que se utilizara durante el proyecto, realización de algunos test y comprobación de su correcto funcionamiento.		

² En la cultura popular, *divide y vencerás* hace referencia a un refrán que implica resolver un problema difícil, dividiéndolo en partes más simples tantas veces como sea necesario, hasta que la resolución de las partes se torna obvia. La solución del problema principal se construye con las soluciones encontradas.

Descarga y Instalación del Software necesario	Horas: 5h
Inicio: 15/03/2012 fin: 16/03/2012	Hojas:
Descarga del diferente software necesario para la elaboración del proyecto y la instalación del mismo en el equipo.	

Estudio y adecuación con la herramienta Flex	Horas: 10h
Inicio: 17/03/2012 fin: 18/03/2012	Hojas:
Planificación del proyecto, cálculos de tiempos y requerimientos y elaboración del documento "Plan de trabajo".	

Estudio exhaustivo de la tecnología SIG	Horas: 25h
Inicio: 19/03/2012 fin: 28/03/2012	Hojas: 4 páginas máximo
Estudio de la tecnología SIG y su posterior implementación en un entorno WEB.	

Obtención de los datos necesarios para el estudio	Horas: 20h
Inicio: 25/03/2012 fin: 04/04/2012	Hojas: 4 páginas máximo
Buscar y descargar los diferentes datos que nos servirán como muestrario. En un principio estos datos serán suministrados por la propia ASPB.	

Elaboración del borrador referente a la PAC 2	Horas: 10h
Inicio: 04/04/2012 fin: 07/04/2012	Hojas: 10 páginas máximo
Elaboración del borrador de la PAC 2, revisión ortográfica, presentación del documento.	

Entrega del borrador PAC2	Horas:
Inicio: 07/04/2012 fin: 07/04/2012	Hojas:
Entrega del borrador referente a la práctica 2.	

Corrección del borrador	Horas: 10h
Inicio: 07/04/2012 fin: 15/04/2012	Hojas:
Correcciones comentadas por la profesora sobre el borrador.	

Entrega de la PAC 2	Horas:
Inicio: 15/04/2012 fin: 15/04/2012	Hojas:

4.2.3 Pac 3

Elaboración de los requerimientos del proyecto	Horas: 30h
Inicio: 15/04/2012 fin: 22/04/2012	Hojas: 3 paginas máximo
Descripción de los requerimientos funcionales (relación entre procesos, actores, datos, MVC) y no funcionales del proyecto (herramientas de desarrollo, plataformas de explotación).	

Elaboración de un modelo conceptual del aplicativo	Horas: 5h
Inicio: 22/04/2012 fin: 24/04/2012	Hojas: 3 paginas máximo
Realización de un modelo conceptual mediante el lenguaje UML para definir el proyecto (clases, atributos, asociaciones).	
Diseño lógico y físico de la base de datos	Horas: 10h
Inicio: 24/04/2012 fin: 27/04/2012	Hojas: 3 paginas máximo
Creación de un primer diseño lógico de la base de datos mediante UML y posteriormente elaborar un diseño físico mas específico de la herramienta utilizada como base de datos.	
Creación de la Base de Datos y inserción de datos	Horas: 15h
Inicio: 27/04/2012 fin: 01/05/2012	Hojas: 0
Creación de la base de datos mediante la herramienta MySQL y el modelo físico descrito en el apartado anterior y su posterior inserción de los datos obtenidos de la ASPB.	
Elaboración de las consultas de la BBDD.	Horas: 40h
Inicio: 01/05/2012 fin: 13/05/2012	Hojas: 3 paginas máximo
Implementación de todas las consultas y filtros necesarios para el tratamiento de la información dentro de la BBDD.	
Elaboración del código necesario para la implementación	Horas: 40h
Inicio: 13/05/2012 fin: 27/05/2012	Hojas: 3 paginas máximo
Implementación del código necesario para el portal web y la visualización y uso de la herramienta.	
Elaboración del borrador referente a la PAC 3	Horas: 5h
Inicio: 27/05/2012 fin: 29/05/2012	Hojas: 20 páginas máximo
Elaboración del borrador de la PAC 2, revisión ortográfica, presentación del documento.	
Entrega del borrador PAC3	Horas:
Inicio: 29/05/2012 fin: 29/05/2012	Hojas:
Entrega del borrador referente a la práctica 2.	
Corrección del borrador	Horas: 10h
Inicio: 29/05/2012 fin: 03/06/2012	Hojas:
Correcciones comentadas por la profesora sobre el borrador.	
Entrega de la PAC 3	Horas:
Inicio: 03/06/2012 fin: 03/06/2012	Hojas:
4.2.4 Pac 4	
Realización de la presentación (PPT y video)	Horas: 20h
Inicio: 03/06/2012 fin: 06/06/2012	Hojas: 10 paginas máximo
Creación de una presentación a partir de diapositivas y un video demostrativo del uso de la aplicación.	

Finalización de la memoria	Horas: 20h
Inicio: 06/06/2012 fin: 08/06/2012	Hojas: 60 páginas máximo
Finalización de la memoria, juntar todas las entregas y elaborar el documento final. Verificar ortografía, semántica y presentación del mismo.	

Entrega del borrador PAC4 (memoria)	Horas:
Inicio: 08/06/2012 fin: 08/06/2012	Hojas:
Entrega del borrador referente a la práctica 2.	

Corrección del borrador final	Horas: 10h
Inicio: 09/06/2012 fin: 10/06/2012	Hojas:
Correcciones comentadas por la profesora sobre el borrador.	

Entrega de la PAC 4 (Memoria final)	Horas:
Inicio: 11/06/2012 fin: 11/06/2012	Hojas:

4.2.5 Debate Virtual

Realización de un debate virtual con el comité de evaluación	Horas: 10h
Inicio: 27/06/2012 fin: 29/06/2012	Hojas:
Durante 3 días existe la posibilidad de recibir diferentes preguntas del comité de evaluación mediante un debate virtual.	

Se han marcado en rojo las tareas que se consideran clave, así como los periodos de entrega que no deberían ser alterados durante el transcurso del proyecto.

Debido a diversos motivos como el trabajo diario y la relación con la familia y amigos, el tiempo disponible a la realización de este proyecto se divide en 2 horas diarias entre semana y a 5 horas diarias los fines de semana.

En la siguiente figura se puede observar el diagrama de Gantt mediante el cual se ha establecido el tiempo y los recursos invertidos en cada una de las tareas especificadas en el punto anterior.

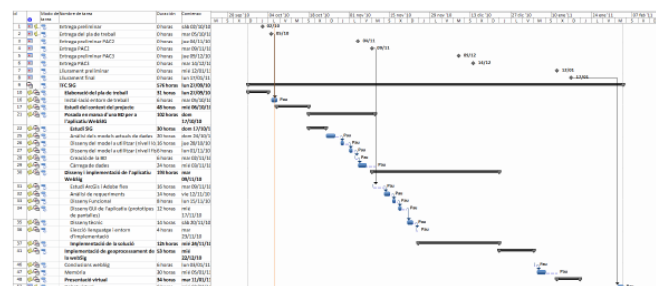


Ilustración 1 - Diagrama de Grantt

5. Sistemas de información geográfica

La información geográfica puede ser consultada, transferida, manipulada, procesada y representada utilizando numerosas herramientas software. En este capítulo se verán algunas de ellas, así como la tecnología y los estándares que emplean.

5.1 ¿Qué es un sistema de información geográfica?

Un Sistema de Información Geográfica (SIG o GIS, en su acrónimo inglés *Geographic Information System*) es una integración organizada de *hardware*, *software* y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y gestión geográfica. También puede definirse como un modelo de una parte de la realidad referido a un sistema de coordenadas terrestre y construido para satisfacer unas necesidades concretas de información. En el sentido más estricto, es cualquier sistema de información capaz de integrar, almacenar, editar, analizar, compartir y mostrar la información geográficamente referenciada. En un sentido más genérico, los SIG son herramientas que permiten a los usuarios crear consultas interactivas, analizar la información espacial, editar datos, mapas y presentar los resultados de todas estas operaciones.

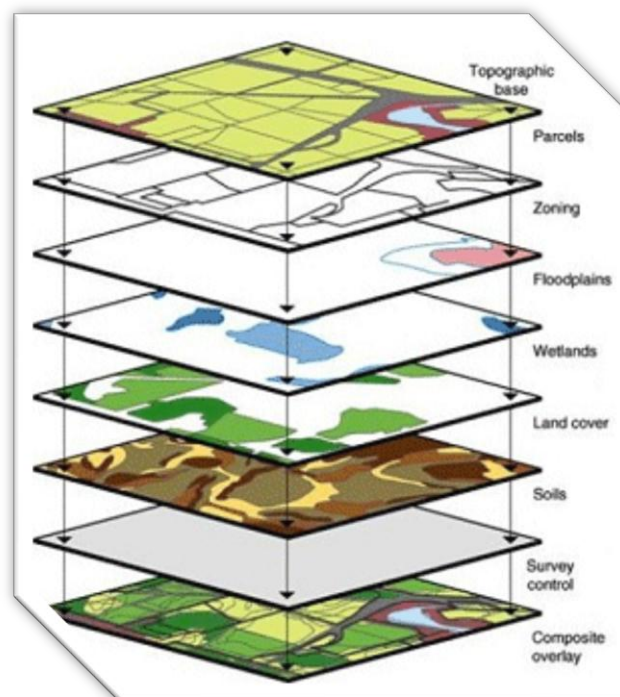


Ilustración 2 - Capas de sistemas SIG

El SIG funciona como una base de datos con información geográfica (datos alfanuméricos) que se encuentra asociada por un identificador común a los objetos gráficos de un mapa digital. De esta forma, señalando un objeto se conocen sus atributos e, inversamente, preguntando por un registro de la base de datos se puede saber su localización en la cartografía.

La razón fundamental para utilizar un SIG es la gestión de información espacial. El sistema permite separar la información en diferentes capas temáticas (véase Ilustración 2 - Capas de sistemas SIG) y las almacena independientemente, permitiendo trabajar con ellas de manera

rápida y sencilla, facilitando al profesional la posibilidad de relacionar la información existente a través de la topología de los objetos, con el fin de generar otra nueva que no podríamos obtener de otra forma.

La tecnología de los Sistemas de Información Geográfica puede ser utilizada para investigaciones científicas, la gestión de los recursos, gestión de activos, la arqueología, la evaluación del impacto ambiental, la planificación urbana, la cartografía, la sociología, la geografía histórica, el marketing, la logística por nombrar unos pocos. Por ejemplo, un SIG podría permitir a los grupos de emergencia calcular fácilmente los tiempos de respuesta en caso de un desastre natural, el SIG puede ser usado para encontrar los humedales que necesitan protección contra la contaminación, o pueden ser utilizados por una empresa para ubicar un nuevo negocio y aprovechar las ventajas de una zona de mercado con escasa competencia.

Las principales cuestiones que puede resolver un Sistema de Información Geográfica, ordenadas de menor a mayor complejidad, son:

- **Localización:** preguntar por las características de un lugar concreto.
- **Condición:** el cumplimiento o no de unas condiciones impuestas al sistema.
- **Tendencia:** comparación entre situaciones temporales o espaciales distintas de alguna característica.
- **Rutas:** cálculo de rutas óptimas entre dos o más puntos.
- **Pautas:** detección de pautas espaciales.
- **Modelos:** generación de modelos a partir de fenómenos o actuaciones simuladas.

Por ser tan versátiles, el campo de aplicación de los Sistemas de Información Geográfica es muy amplio, pudiendo utilizarse en la mayoría de las actividades con un componente espacial. La profunda revolución que han provocado las nuevas tecnologías ha incidido de manera decisiva en su evolución.

5.2 Introducción al Web-GIS

En la actualidad existen un gran número de trabajos que han sido desarrollados para trabajar con tecnología *Web-Mapping*, en donde todos éstos hacen referencia a las aplicaciones de mapeo y cartografía proporcionadas por un Sistema de Información Geográfica (SIG), y que pueden manipularse a través de una interfaz Web.

Con estas aplicaciones se puede acceder, manipular, analizar y recuperar datos geo-espaciales, en algún formato específico o contenido en una base de datos espacial a través de Internet o una Intranet.

Hoy en día, la tendencia en el área de la Geocomputación se ha enfocado al desarrollo de SIG en Internet, lo cual ha sido denominado como *Web-GIS*³. A través de esta tecnología los usuarios de Internet tienen la posibilidad de acceder a aplicaciones SIG desde sus navegadores, sin efectuar un pago por el uso del software SIG propietario.

³ http://es.wikipedia.org/wiki/Web_Mapping

La gran mayoría de los proyectos actuales de Web-GIS y aplicaciones relacionadas con servidores de mapas están dirigidos a la cartografía interactiva. Esta es una de las áreas más importantes que involucran "pay-for-use"⁴ servicios de *Mapping* y SIG.

La característica más común entre las aplicaciones SIG actuales es la de operar sobre datos espaciales, utilizando una arquitectura Cliente-Servidor y realizando accesos por medio de protocolos de Internet o Intranet.

La arquitectura de *Web-Mapping* (Web-GIS) puede describirse como un modelo Web Cliente-Servidor. Actualmente existen varios organismos internacionales para la definición de estándares de representación, consulta y visualización de información geográfica. Así tanto el *Open Geospatial Consortium* (OGC) como el *Technical 212* de ISO han establecido diferentes estándares para servicios y lenguajes de representación de información geográfica.

La arquitectura de un SIG según ambos organismos es la siguiente:

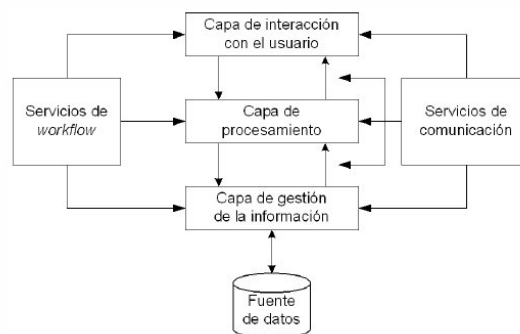


Ilustración 3 - Arquitectura de un SIG según OGC

En la Ilustración 3 - Arquitectura de un SIG según OGC puede observarse las diferentes capas que forman la arquitectura básica de un sistema SIG:

- **Capa de gestión de la información:** es la encargada del almacenamiento y procesamiento básico.
- **Capa de procesamiento de la información:** es la encargada del análisis espacial.
- **Capa de interacción con el usuario:** es la encargada de la visualización y de la interacción con el usuario.
- **Servicios de comunicación:** define lenguajes y servicios para conectar las diferentes capas.
- **Servicios de workflow:** define servicios y protocolos para enlazar e integrar diversos componentes.

Asimismo se puede observar como las diferentes capas se comunican de forma directa para ofrecer la información solicitada por el cliente. De esta forma, el mapa actualizado con la nueva información solicitada es enviado al navegador embebido en una página HTML.

⁴ Muchos de los servicios actuales de SIG utilizan el sistema "pay-for-use" donde el precio del servicio depende del uso realizado.

Finalmente, para el desarrollo y establecimiento de un servidor de *Web-GIS* se debe considerar el diagrama mostrado en la siguiente figura (véase Ilustración 4 Diagrama de flujo para el desarrollo de aplicaciones WEB-GIS), en ella se pueden observar las diferentes etapas para llevar a buen puerto una aplicación Web-GIS.

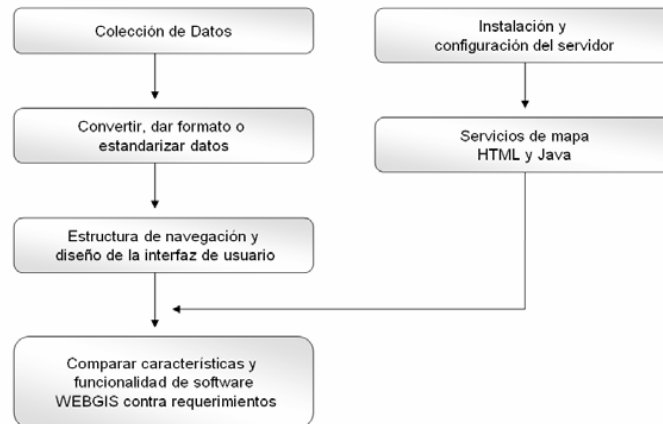


Ilustración 4 Diagrama de flujo para el desarrollo de aplicaciones WEB-GIS

5.2.1 Arquitectura Cliente-servidor.

La mayoría de aplicaciones SIG implementan la arquitectura comúnmente conocida como cliente-servidor.

La arquitectura cliente-servidor(C/S) es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En la arquitectura C/S el remitente de una solicitud es conocido como cliente. Sus características son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación.
- Espera y recibe las respuestas del servidor.
- Por lo general, puede conectarse a varios servidores a la vez.
- Normalmente interactúa directamente con los usuarios finales mediante una interfaz gráfica de usuario.

Al receptor de la solicitud enviada por el cliente se conoce como servidor. Sus características son:

- Al iniciarse esperan a que lleguen las solicitudes de los clientes, desempeñan entonces un papel pasivo en la comunicación.
- Tras la recepción de una solicitud, la procesan y luego envían la respuesta al cliente.

- Por lo general, aceptan conexiones desde un gran número de clientes (en ciertos casos el número máximo de peticiones puede estar limitado).
- No es frecuente que interactúen directamente con los usuarios finales.

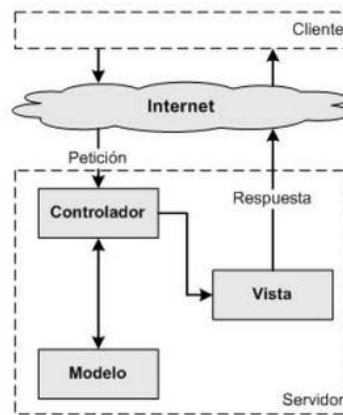


Ilustración 5 – Arquitectura C/S (Modelo Vista Controlador)

En la Ilustración 5 – Arquitectura C/S (Modelo Vista Controlador se hace referencia a una estructura básica de la arquitectura Cliente-Servidor, pero entrando en más de detalle en la parte servidor donde se utilizara la arquitectura Modelo-Vista-Controlador (MVC).

5.3 Datos espaciales

Una base de datos espacial es una base de datos con extensiones para el almacenamiento, consulta y manipulación de información geográfica y datos espaciales, también conocidas como bases de datos geográficas. A estas bases de datos accederán los servidores de información geográfica para almacenar, modificar, obtener y ofrecer los datos a petición de las aplicaciones cliente.

Nuestro trabajo se va a realizar sobre una base de datos espacial *PostgreSQL* con su extensión *PostGIS*.

5.3.1 Representación de datos espaciales

Actualmente existen dos formas de representar los datos en un SIG: *raster* y vectorial (véase Ilustración 6 - Representación geográfica (Vectorial vs *Raster*)). Aunque en este proyecto nos centraremos en la representación vectorial, vamos a realizar una breve introducción a ambos.

Los SIG que se centran en el manejo de datos en formato vectorial son más populares en el mercado. No obstante, los SIG *raster* son muy utilizados en estudios que requieran la generación de capas continuas, necesarias en fenómenos no discretos; también en estudios medioambientales donde no se requiere una excesiva precisión espacial (contaminación atmosférica, distribución de temperaturas, localización de especies marinas, análisis geológicos, etc.).

Raster:

Un tipo de datos *raster* es, en esencia, cualquier tipo de imagen digital representada en mallas. El modelo de SIG *raster* o de retícula se centra en las propiedades del espacio más que en la precisión de la localización. Divide el espacio en celdas regulares donde cada una de ellas representa un único valor.

Vectorial:

En los datos vectoriales, el interés de las representaciones se centra en la precisión de localización de los elementos geográficos sobre el espacio y donde los fenómenos a representar son discretos, es decir, de límites definidos. Cada una de estas geometrías está vinculada a una fila en una base de datos que describe sus atributos.

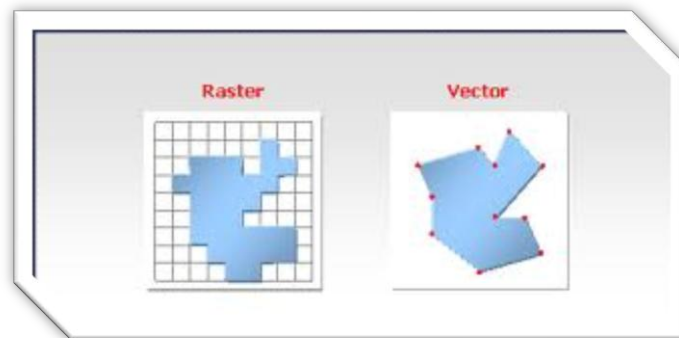


Ilustración 6 - Representación geográfica (Vectorial vs Raster)

Como hemos comentado en este proyecto nos centraremos en el almacenamiento de datos de tipo vectorial. Para modelar digitalmente las entidades del mundo real se utilizan tres elementos geométricos: el punto, la línea y el polígono.

- **Puntos:** Los puntos se utilizan para las entidades geográficas que mejor pueden ser expresadas por un único punto de referencia. En otras palabras: la simple ubicación. Por ejemplo, las localizaciones de los pozos, picos de elevaciones o puntos de interés. Los puntos transmiten la menor cantidad de información de estos tipos de archivo y no son posibles las mediciones. También se pueden utilizar para representar zonas a una escala pequeña. Por ejemplo, las ciudades en un mapa del mundo estarán representadas por puntos en lugar de polígonos.
- **Líneas o polilíneas:** Las líneas unidimensionales o polilíneas son usadas para rasgos lineales como ríos, caminos, ferrocarriles, rastros, líneas topográficas o curvas de nivel. De igual forma que en las entidades puntuales, en pequeñas escalas pueden ser utilizados para representar polígonos. En los elementos lineales puede medirse la distancia.
- **Polígonos:** Los polígonos bidimensionales se utilizan para representar elementos geográficos que cubren un área particular de la superficie de la tierra. Estas entidades pueden representar lagos, límites de parques naturales, edificios, provincias, o los usos del suelo. Por ejemplo, los polígonos transmiten la mayor cantidad de información en archivos con datos vectoriales y en ellos se pueden medir el perímetro y el área.

5.3.2 Almacenamiento de datos en sistemas SIG

Una vez comentadas las técnicas de representación gráfica nos centraremos en la elección del almacenamiento de datos. Los datos geográficos son más complejos y tienen múltiples tipos de representación como hemos visto anteriormente. Debido a esta complejidad los sistemas SIG normalmente se apoyan en sistemas de gestión de base de datos (SGBD) tradicionales aunque ligeramente adaptados (modificados) con la finalidad de poder almacenar y gestionar de forma más eficiente los datos espaciales. Estas bases de datos espaciales (*spatial database*)

son un sistema administrador de bases de datos que maneja datos existentes en un espacio o datos espaciales.

Base de datos espacial

En este tipo de bases de datos es imprescindible establecer un cuadro de referencia (un SRE, *Sistema de Referencia Espacial*) para definir la localización y relación entre objetos, ya que los datos tratados en este tipo de bases de datos tienen un valor relativo, no un valor absoluto. Los sistemas de referencia espacial pueden ser de dos tipos: geo-referenciados (aquellos que se establecen sobre la superficie terrestre. Son los que normalmente se utilizan, ya que es un dominio manipulable, perceptible y que sirve de referencia) y no geo-referenciados (son sistemas que tienen valor físico, pero que pueden ser útiles en determinadas situaciones).

La construcción de una base de datos geográfica implica un proceso de abstracción para pasar de la complejidad del mundo real a una representación simplificada que pueda ser procesada por el lenguaje de las computadoras actuales. Este proceso de abstracción tiene diversos niveles y normalmente comienza con la concepción de la estructura de la base de datos, generalmente en capas; en esta fase, y dependiendo de la utilidad que se vaya a dar a la información a compilar, se seleccionan las capas temáticas a incluir.

Ficheros Shapefile

Por otro lado, en este proyecto se utilizara otro formato de almacenamiento de datos espaciales denominado *Shapefile*. El formato ESRI Shapefile (SHP) es un formato de archivo informático propietario de datos espaciales desarrollado por la compañía ESRI, quien crea y comercializa software para Sistemas de Información Geográfica como Arc/Info o ArcGIS. Originalmente se creó para la utilización con su producto ArcView GIS, pero actualmente se ha convertido en formato estándar de facto para el intercambio de información geográfica entre SIG por la importancia que los productos ESRI tienen en el mercado SIG y por estar muy bien documentado.

Un *Shapefile* 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. Un *Shapefile* es un formato multiarchivo, es decir está generado por varios ficheros informáticos. El número mínimo requerido es de tres y tienen las extensiones siguientes:

- **.shp** - es el archivo que almacena las entidades geométricas de los objetos.
- **.shx** - es el archivo que almacena el índice de las entidades geométricas.
- **.dbf** - es la base de datos, en formato dBASE, donde se almacena la información de los atributos de los objetos.

5.4 Estándares de los SIG

El *Open Geospatial Consortium* (OGC) fue creado en 1994 y agrupa a 372 organizaciones públicas y privadas. Su fin es la definición de estándares abiertos e interoperables dentro de los Sistemas de Información Geográfica y de la *World Wide Web*. Persigue acuerdos entre las diferentes empresas del sector que posibiliten la interoperación de sus sistemas de geoprocetamiento y facilitar el intercambio de la información geográfica en beneficio de los usuarios.

Los principales estándares publicados por el OGC que hemos empleado en el desarrollo de la aplicación son los siguientes:

5.4.1 Geographic Markup Language (GML)

Sub-lenguaje de XML descrito como una gramática en XML Schema para el modelaje, transporte y almacenamiento de información geográfica. Su importancia radica en que a nivel informático se constituye como una lengua franca para el manejo y trasvase de información entre los diferentes software que hacen uso de este tipo de datos, como los Sistemas de Información Geográfica.

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 no contiene información específica sobre cómo se debe hacer la visualización de los datos representados. Para ello se utilizan estilos que se relacionan a GML y se describen en otros sub-lenguajes de XML. Otras extensiones manejadas por GML incluyen SMIL para definir elementos de interacción y Xpointer para representar metadatos.

5.4.2 Web Feature Service (WFS)

La especificación del OGC del estándar WFS permite a un cliente acceder de manera uniforme a los elementos geográficos, codificados en diferentes formatos de salida como GML o KML, almacenados en un servidor desde diferentes servicios web. Esta especificación define interfaces para operaciones de acceso y manipulación de información de entidades geográficas, a través del protocolo HTTP.

Se usa WFS cuando se quieren llevar a cabo acciones como:

- Realizar peticiones de un conjunto de datos y recuperación de elementos geográficos.
- Conocer la definición de un elemento geográfico como nombre o tipo de las propiedades del elemento.
- Agregar nuevos elementos al conjunto de datos.
- Borrar elementos del conjunto de datos.
- Actualizar elementos del conjunto de datos.
- Bloquear elementos para la prevención de modificaciones.

Las operaciones obligatorias definidas por el estándar son las siguientes:

- **DescribeFeatureType:** Ofrece la estructura (campos y atributos) de cualquiera de los tipos de fenómenos que el servicio puede ofrecer, mediante su descripción en esquemas XML, siendo éstos esquemas GML válidos.
- **GetFeature:** Devuelve instancias de fenómenos (objetos individuales). En formato GML. Además, el cliente debe tener la posibilidad de solicitar las propiedades del fenómeno que desea y de realizar tanto consultas espaciales como no espaciales.
- **GetCapabilities:** Ofrece información acerca de las características del servicio, de los tipos de fenómenos que puede ofrecer y las operaciones que soporta cada uno de ellos.

5.4.3 Web Map Service (WMS)

El estándar *Web Map Service* (WMS) definido por el OGC produce mapas de datos referenciados espacialmente, de forma dinámica a partir de información geográfica. Este estándar internacional define un mapa como una representación de la información geográfica en forma de un archivo de imagen digital conveniente para la exhibición en una pantalla de ordenador. Los mapas producidos por WMS se generan normalmente en un formato de imagen como PNG, GIF o JPEG.

El estándar define tres operaciones:

- **GetCapabilities:** Devuelve metadatos e información del servicio.
- **GetMap:** Construye y devuelve un mapa como una imagen cuyos parámetros geográficos y dimensionales han sido bien definidos.
- **GetFeatureInfo:** Devuelve información acerca de los objetos representados en la imagen.

6. Tecnología utilizada

En este capítulo realizaremos una breve introducción a las diferentes tecnologías elegidas para el desarrollo del trabajo, como puede verse en la Ilustración 7 - Esquema genérico del aplicativo.

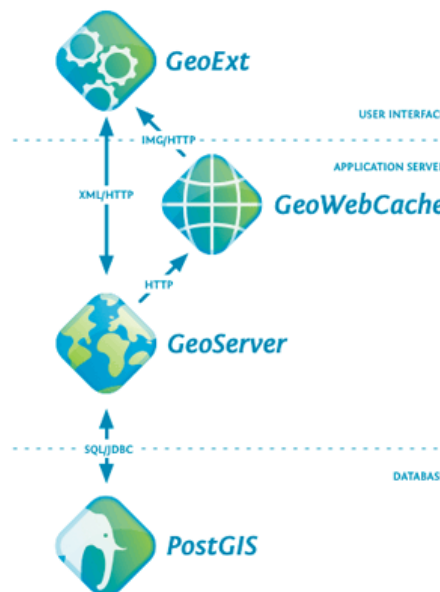


Ilustración 7 - Esquema genérico del aplicativo

6.1 PostGIS

PostGIS fue desarrollado por la firma Canadiense *Refractions Research Inc.*, como un proyecto de investigación en tecnología de bases de datos espaciales de código abierto (GNU).

PostGIS proporciona soporte para el manejo de datos geográficos en la base de datos relacional de *PostgreSQL* y la habilita para el manejo de datos espaciales como puede observarse en la Ilustración 8 - Estructura Base de Datos PostgreSQL, permitiéndole ser usada como respaldo de base de datos espacial para los Sistemas de Información Geográfica.

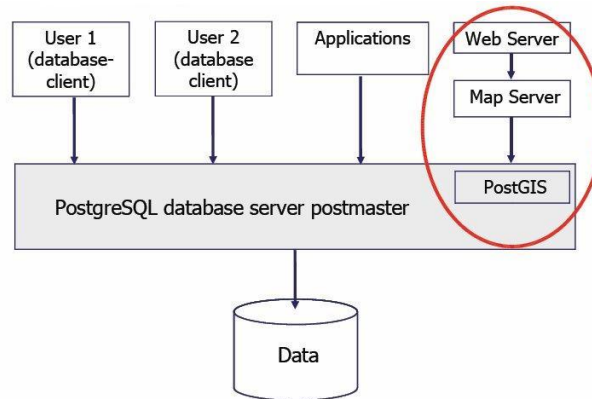


Ilustración 8 - Estructura Base de Datos PostgreSQL

Actualmente adopta el estándar del OGC y se convierte en un modelo espacial basado en objetos con las siguientes características:

- Dos formas de representar los objetos espaciales, la *Well-know text* (WKT) o representación de los objetos en modo texto, como por ejemplo, "POINT (0 0 0)" y la *Well-know binary* (WKB) o formato binario. Las dos formas guardan información del tipo de objeto y sus coordenadas.
- Requiere que los objetos incluyan el identificador del sistema de referencia espacial (SRID), el cual es requerido cada vez que se inserta una instancia en un objeto espacial.
- Tres tablas para definir los objetos espaciales, una que define el objeto es si, y otras dos tablas de metadatos, una define el sistema de referencia espacial llamada *spatial_ref_sys*, y la otra define el atributo geométrico llamada *geometry_column*.

6.2 GeoServer

GeoServer es un programa de servidor de mapas *Open Source*⁵ escrito en java que permite compartir y editar datos geoespaciales, está distribuido bajo licencia *General Public License* (GPL), el proyecto está dirigido por la comunidad *GeoServer* y está construido en base a la librería *Geotools*⁶.

Los servidores de mapas permiten al usuario la máxima interoperabilidad con la información geográfica. Por un lado, el usuario o cliente accede a la información en su formato original, de manera que es posible realizar consultas tan complejas como las que haría un SIG. Un servidor de mapas funciona enviando, a petición del cliente, desde su navegador de internet, una serie de páginas HTML con una cartografía asociada en formato de imagen. En la actualidad los

⁵ Código abierto (Open source) 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.

⁶ GeoTools es una biblioteca SIG de código libre que permite desarrollar soluciones adaptadas a los estándares. Proporciona una implementación de las especificaciones del Open Geospatial Consortium según van apareciendo. <http://geotools.org/>

servidores de mapas implementan tecnologías que permiten el intercambio de formatos e información siguiendo las especificaciones *Web Map Service (WMS)*, *Web Coverage Service (WCS)* y *Web Feature Service (WFS)*, creadas por el OGC.

GeoServer tiene certificación de la OGC en los estándares WCS 1.0, WMS 1.1.1 y WFS 1.0, además sirve de referencia para la implementación de los mismos.

Utiliza tecnología Java J2EE y puede ser portado a cualquier contenedor de servlets, soportando arquitectura 64-bit. Incluye seguridad integrada y una herramienta web de configuración y administración.

6.3 GeoExt (OpenLayers)

OpenLayers hace que sea fácil mostrar un mapa dinámico en cualquier página web, ya que puede mostrar mapas y marcadores cargados de cualquier fuente, es totalmente gratuito, de código abierto y publicado bajo la licencia BSD.

La idea básica de *OpenLayers* consiste en separar las herramientas de mapeo de los datos del mapa, de esta manera todas las herramientas puedan operar sobre todas las fuentes de datos.

Se puede definir a *OpenLayer* como una biblioteca *JavaScript* para visualizar mapas en la mayoría de navegadores modernos, no del lado del servidor, sino del lado del cliente, que nos permite elaborar nuestros propios mapas haciendo uso de su propia base de información cartográfica o usando su estructura para hacer uso de otros servicios y poder colocarlos en nuestra web.

Además, *OpenLayers* implementa métodos estándar de la industria para el acceso a los datos geográficos, tales como el OGC, *Web Map Services (WMS)* y *Web Feature Service (WFS)*, *OpenLayers* está escrito en *JavaScript* orientado a objetos, utilizando componentes de *Prototype.js*⁷ y de la biblioteca Rico⁸.

⁷ <http://www.prototypejs.org/>

⁸ <http://openrico.org/>

7. Análisis

Una vez definidos los objetivos del trabajo y realizada una pincelada previa sobre la teoría de los SIG y las diferentes tecnologías relacionadas con estos sistemas de procesamiento de información geográfica, procedemos a realizar un estudio de los requisitos que deben solventar nuestra aplicación.

EL objetivo principal de la aplicación es la creación de una herramienta web que permita visualizar de forma rápida un mapa interactivo y visualizar en él, los diferentes casos de enfermedades en el territorio de Barcelona. Mediante esta aplicación tiene que ser posible visualizar en el espacio-tiempo los diferentes casos y la influencia que ha tenido sobre ellos factores ajenos como puede ser el barrio de residencia, el nivel de contaminación, etc.

La información relativa a los diferentes casos de tuberculosis en el municipio de Barcelona nos ha sido proporcionada por la Agencia de Salud Pública de Barcelona (ASPB), <http://www.aspb.com> y los datos cartográficos referentes a los barrios de Barcelona nos ha sido facilitada por la profesora Laura Diaz.

7.1 Análisis funcional

En la presente sección se describe el modelado de la aplicación Web-GIS, haciendo uso de diversos diagramas UML. De igual forma solo se presentan los diagramas principales del modelado.

En el desarrollo de esta primera fase de la aplicación sólo existirá un tipo de usuario ya que no va a ser necesario estar registrado para acceder a la interfaz⁹. Vamos a explicar las funcionalidades de la aplicación a través de los diagramas de casos de uso y diagramas de secuencia, para tener una idea general de la aplicación y facilitar su comprensión.

7.1.1 Casos de uso

Se listan a continuación las operaciones y funciones principales de la aplicación WEB-GIS de propósito específico:

- **Despliegue de la información espacial.** Este módulo se encarga de establecer enlaces a los datos geográficos; así como de presentar y desplegar la información geográfica organizada por capas.
- **Activar y desactivar las capas de información.** Esta función se encarga de activar, visualizar una o más capas; así como también de desactivarlas, no visualizar las capas, con el objetivo de analizar y representar los datos de acuerdo al caso de estudio.
- **Funciones de navegación en el mapa: zoom, pan y full extent.** Las funciones de navegación básicas: acercamiento, alejamiento, desplazamiento y extensión, están presentes como opciones de visualización, éstas son en conjunto una herramienta de visualización que permite a los usuarios analizar los datos desde diferentes perspectivas.
- **Filtrado de información.** Esta función nos permite filtrar y acotar las búsquedas dentro del aplicativo. Para poder obtener información más precisa o eliminar información innecesaria para el estudio actual.

⁹ Se ha planteado la creación de un sistema de registro y control de usuarios, pero se ha dejado para un posible evolutivo futuro.

- **Identificación de objetos geográficos.** La tarea de esta función es recuperar los atributos que componen un objeto espacial (individual) dentro de una capa de datos espacial.

Los procesos se describen de forma visual por medio de un diagrama de casos de uso en la Ilustración 9 Diagrama de casos de uso de la aplicación.

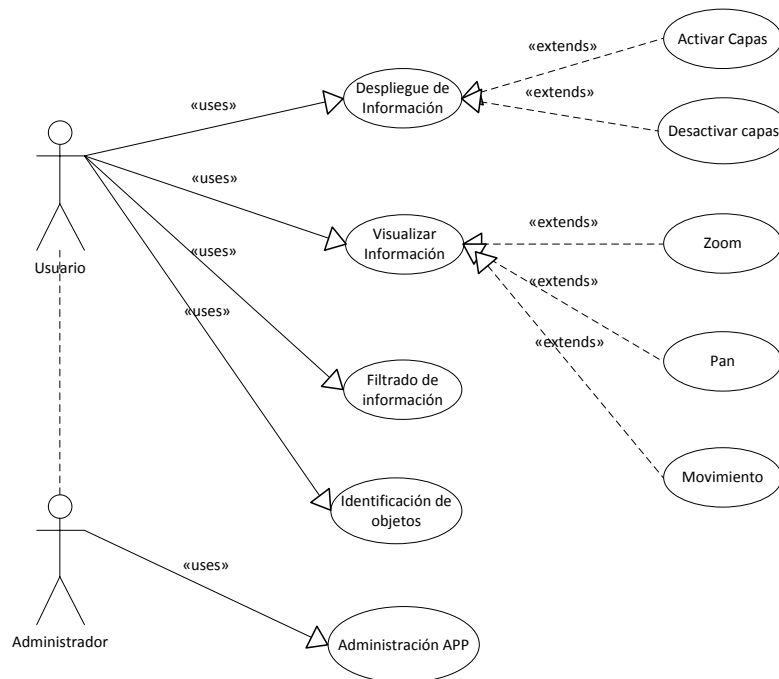


Ilustración 9 Diagrama de casos de uso de la aplicación

7.1.2 Diagramas de secuencia de la aplicación

A continuación definiremos los diferentes diagramas de secuencia relacionados con cada uno de los casos de uso definidos en el diagrama anterior.

Despliegue de información y capas

Este módulo tiene la función de efectuar la solicitud al sistema (Servidor de Mapas + Base de datos espacial) y posterior visualización de la información espacial por parte del cliente (aplicación web + usuario). En este proceso existe una relación directa con la base de datos espacial y los archivos geográficos ya que a partir de estos se obtiene la información solicitada (véase Ilustración 10 Diagrama de Secuencia de la aplicación).

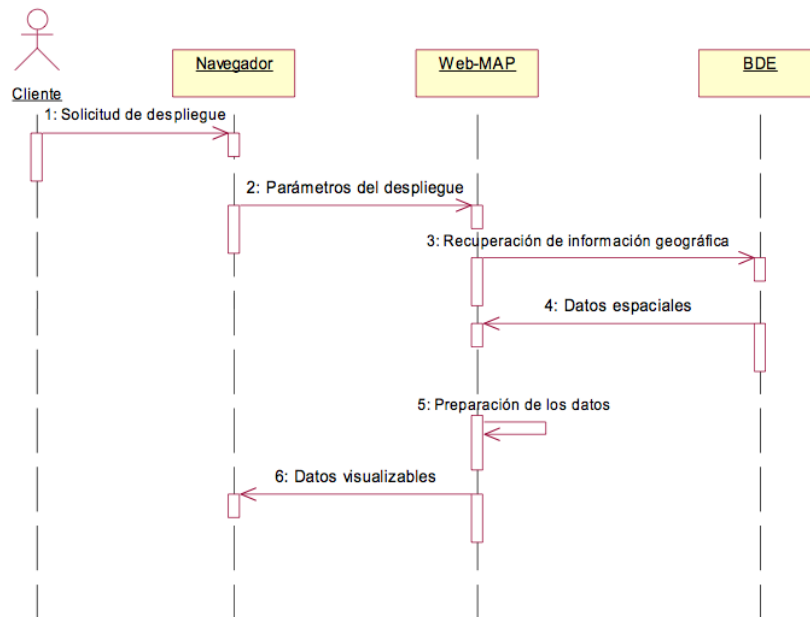


Ilustración 10 Diagrama de Secuencia de la aplicación

Activar y desactivar capas

Este proceso tiene la función de activar y desactivar las diferentes capas de datos para que puedan ser manipuladas y se obtenga la información necesaria de forma más visual (véase Ilustración 11 Diagrama de secuencia Activar/Desactivar Capas).

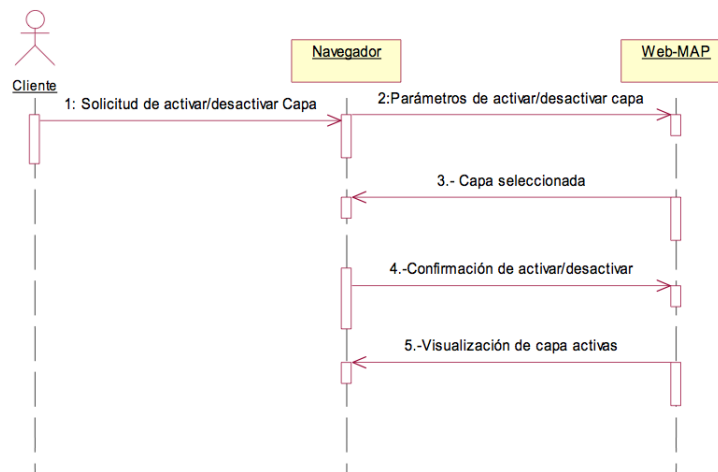


Ilustración 11 Diagrama de secuencia Activar/Desactivar Capas

Función de Zoom

Este es el proceso encargado del acercamiento de la imagen que se visualiza sobre el área de trabajo, el cual puede ser visualizado en la Ilustración 12 Diagrama de secuencia de función Zoom.

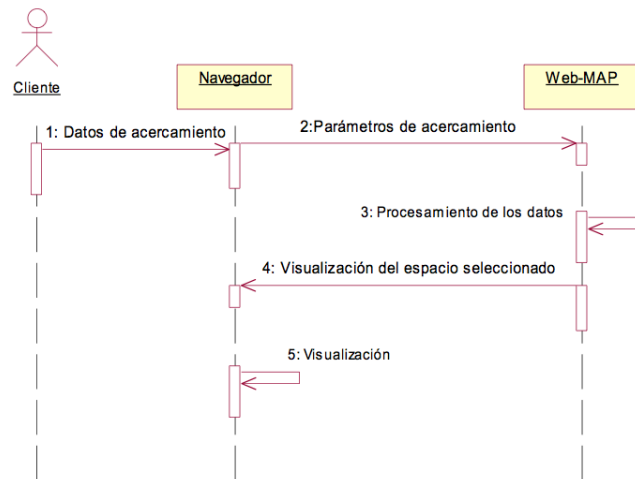


Ilustración 12 Diagrama de secuencia de función Zoom

Función Full Extended

Esta es la función encargada de mostrar el espacio total de la imagen geográfica (véase Ilustración 13 Diagrama de secuencia Función Full Extended).

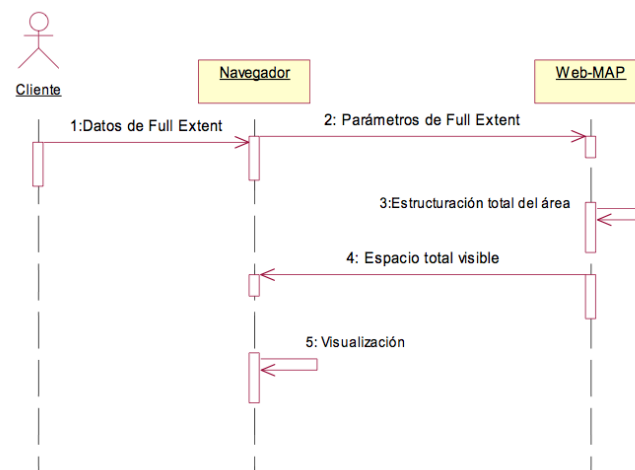


Ilustración 13 Diagrama de secuencia Función Full Extended

Función Pan

Esta función se encarga del desplazamiento que se tenga en ese instante, sin modificar el nivel de detalle de visualización de los datos geográficos. (Véase Ilustración 14 Diagrama de secuencia Función Pan).

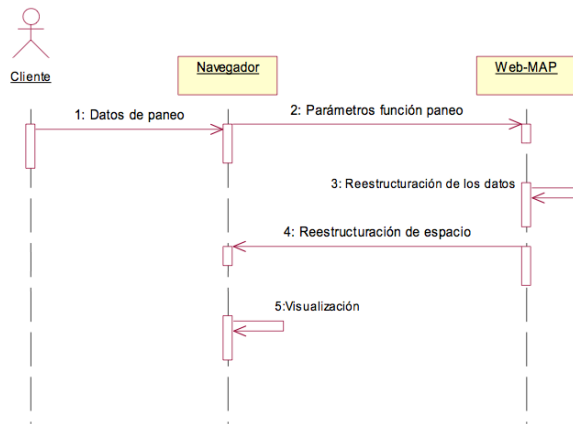


Ilustración 14 Diagrama de secuencia Función Pan

Función Identificación de Objetos

Proceso encargado de la búsqueda y despliegue de atributos que corresponden a un objeto geográfico, los cuales son almacenados en una capa de información o en la BBDD (véase Ilustración 15 Diagrama de secuencia de Identificador de Objeto).

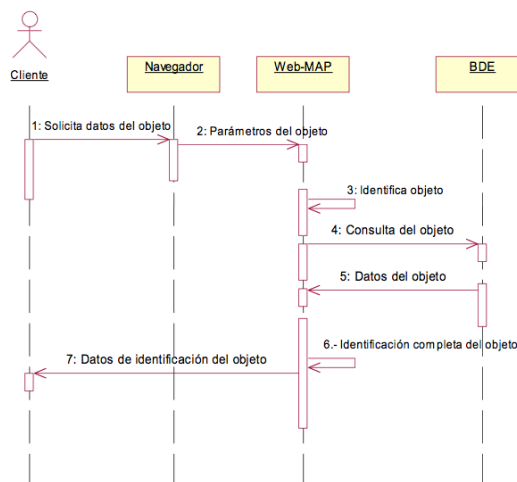


Ilustración 15 Diagrama de secuencia de Identificador de Objeto

Función Filtrar información

Proceso encargado del filtrado de información representada en pantalla. Mediante este proceso podemos filtrar los datos obtenidos del sistema. Formulando preguntas como “¿Cuántos casos existen en un distrito concreto?”, “¿En qué distritos hay mayor número de casos?”, etc. (véase Ilustración 16 Diagrama de secuencia de filtrado de información).

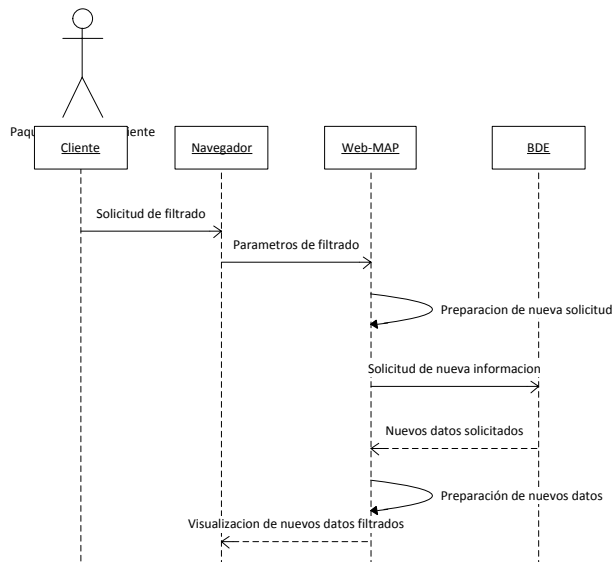


Ilustración 16 Diagrama de secuencia de filtrado de información

Para la realización de este punto se han utilizado las herramientas ofrecidas por *GeoServer*, mediante el uso del WFS que nos ofrece podemos realizar el filtrado de información para representar por ejemplo solo los casos de tuberculosis en el distrito de Gracia o comparar visualmente sobre el mapa dos distritos y su potencial número de casos.

7.1.3 Diagramas de estado y Diagrama de Actividad de la aplicación.

El modelo estático del sistema consiste de dos diagramas principales:

- El diagrama de estados.
- El diagrama de actividad.

El primero presenta los estados por los que pasa internamente las clases principales del Web-GIS y el segundo presenta de manera general el funcionamiento estático de la aplicación (véase Ilustración 19 Diagrama de Actividad de la aplicación).

En un principio el servicio carga un mapa del área metropolitana de Barcelona y ofrece al usuario la posibilidad de cargar los datos de las enfermedades, en una primera versión de la aplicación, solamente los casos de tuberculosis.

El usuario puede tomar la decisión (véase Ilustración 17 Diagrama de estados del servicio) de visualizar los casos (objetos) sobre el mapa y por otro lado activar/desactivar las diferentes capas de información, el servicio trata las peticiones del usuario solicitando la información y generando una respuesta visual sobre el mapa.

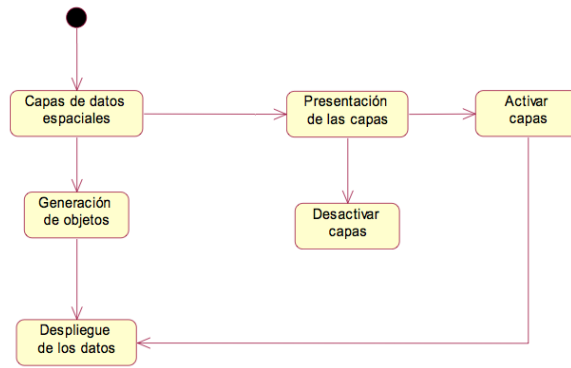


Ilustración 17 Diagrama de estados del servicio

Una vez cargada la información se presentan al usuario diferentes opciones de visualización de la información como puede ser la identificación de casos concretos con los datos más relevantes del mismo o el filtrado y visualización de datos en respuesta a determinadas especificaciones (véase Ilustración 18 Diagrama de estado del cliente).

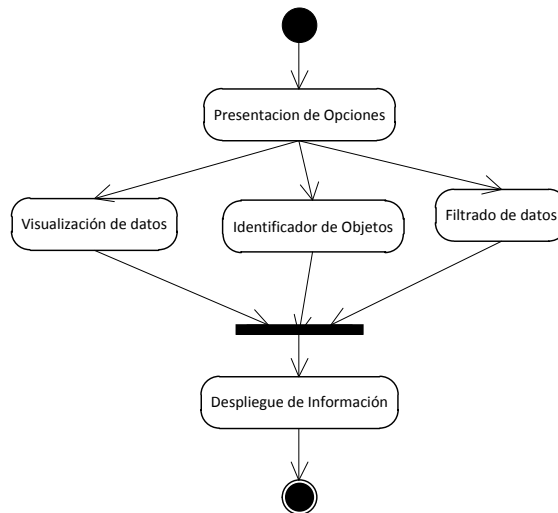


Ilustración 18 Diagrama de estado del cliente

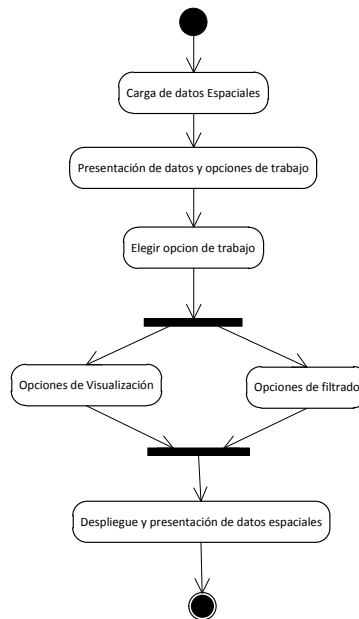


Ilustración 19 Diagrama de Actividad de la aplicación

8. Diseño Técnico e implementación de la Aplicación Web-GIS.

Una vez definidos los requisitos y realizado el análisis funcional de los diferentes aspectos del aplicativo procederemos a realizar el Diseño Técnico (DT) y la implementación de una versión beta del mismo.

En primer lugar realizaremos una pincelada sobre la Arquitectura elegida para este proyecto (Cliente-Servidor) y cada uno de los componentes que intervendrán durante el desarrollo de la aplicación. En el esquema siguiente (véase Ilustración 20 Arquitectura base de la aplicación) se puede observar la estructura que tendrá la aplicación por completo y las tecnologías utilizadas en cada sección.

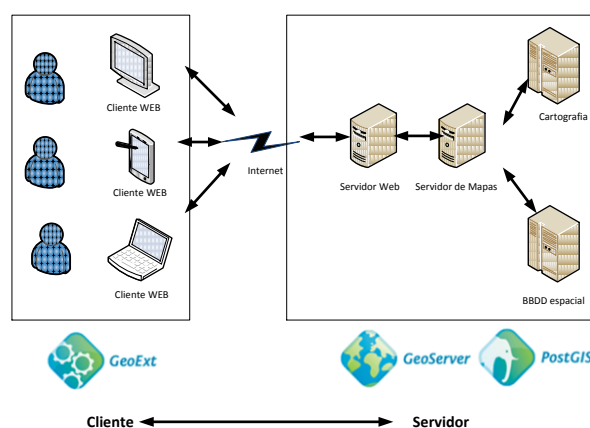


Ilustración 20 Arquitectura base de la aplicación

En este caso se ha optado por instalar y configurar los diferentes servicios y tecnologías por separado para obtener así una mayor personalización, pero para agilizar trabajo se podría optar por la siguiente suite *OpenGeo*¹⁰ que ya incorpora todas las herramientas necesarias para llevar a cabo el proyecto.

En primer lugar, hay que crear una base de datos espacial sobre *PostGIS*, que incluirá los datos ofrecido por el ASPB y los *Shapesfiles* de los distritos de Barcelona, para ello usaremos la interfaz gráfica de *PostgreSQL*, *pgAdmin III*, con *PostGIS* simplemente almacenamos en la base de datos la información geográfica que vamos a mostrar en los mapas, es el que alimenta al servidor de mapas.

En segundo lugar, cargamos nuestra base de datos espacial en *GeoServer* para poder ofrecer, más tarde, los servicios de WMS y WFS a *OpenLayers*, para conectarnos con *PostGIS* simplemente hay que especificar los parámetros de conexión con la base de datos, *GeoServer* consultará la tabla *geometry_columns* para identificar qué tablas de base de datos tienen datos georreferenciados y nos las mostrará, para que elijamos la que queramos. Para almacenarlas crearemos un espacio de nombres llamado ASPB y un almacén de datos, *DataStore*, aquí cargamos todas nuestras capas o *FeatureType*.

GeoServer se encarga de servir los mapas por protocolos estándar de forma que puedan ser visualizados por clientes tanto ligeros como pesados.

El proyecto *GeoServer* es una implementación totalmente transaccional de la especificación OGC que documenta como ofrecer información geográfica de una manera estándar a través de un servidor web.

Una vez explicadas las pautas básicas de la aplicación podemos observar que se divide en tres sectores y a continuación entraremos en detalle en cada uno de ellos:

8.1 Base de datos espacial

El primero de ellos es la gestión y el almacenamiento de los datos, como hemos comentado anteriormente este es uno de los puntos más relevantes de un SIG. Aunque en esta aplicación en concreto los requisitos no son muy elevados necesitamos de igual forma un SGBD adaptable a este tipo de datos.

En un principio, los datos geo-espaciales que vamos a tratar en este proyecto se basan en los elementos geográficos básicos, principalmente el punto y las líneas, ocasionalmente según el sistema de filtrado de los datos se pueden llegar a utilizar el elemento polígono, para representar áreas sobre el mapas.

Se ha optado por utilizar la combinación de *PostgreSQL*¹¹ + *PostGIS*¹² para la realización de un SGBD adaptado a la información geo-espacial. Aunque podría haberse utilizado un SGBD tradicional y hacer los cálculos geo-espaciales mediante código en el servicio web. Esto no

¹⁰ OpenGeo es una suite Open Source que incluye todas las herramientas necesarias para llevar a cabo un proyecto Web-GIS <http://workshops.opengeo.org/stack-intro/install-suite.html>

¹¹ <http://www.postgresql.org/>

¹² <http://postgis.refrations.net/>

sería conveniente en un trabajo donde se pretende explicar el potencial de los sistemas SIG y reduciría la escalabilidad y reutilización de esta base de datos.

Existen numerosas opciones en el mercado que nos permitirían realizar las funciones necesarias para el tratamiento geo-espacial (ArcSDE de ESRI, Spatial de Oracle, Geomedia, etc), aunque podríamos haber realizado un estudio de cualquiera de ellas y serian totalmente válidas para este proyecto, se ha elegido la combinación mencionada anteriormente porque los datos facilitados por ASBP que utilizaremos en el desarrollo de esta aplicación ya vienen configurados mediante esa herramienta.

PostGIS como hemos definido anteriormente es un complemento de código libre de un SGBD bastante conocido y con una gran solvencia contrastada, de esta forma *PostGIS* solo tiene que dedicarse a implementar la información geográfica.

8.1.1 Diseño BD y modelado de datos

Para la realización del proyecto la universidad nos ha proporcionado unos datos de ejemplo ya exportados desde un SGBD. Estos datos son más antiguos y puede que no sirvan a la hora de realizar un estudio exhaustivo, pero si para realizar las pruebas necesarias de la aplicación. Por otro lado, existe la posibilidad de actualizar dichos datos con los proporcionados por la ASBP mucho más actualizados, aunque estos datos tendrían que convertirse en datos geo-espaciales mediante el proceso de geocodificación¹³ para obtener así una referencia necesaria para representar la información sobre un mapa.

La base de datos obtenida como ejemplo de la información ya geocodificada del ASBP tiene el siguiente modelo de datos y se centra básicamente alrededor de una tabla principal “Casos” que contiene prácticamente toda la información necesaria como puede verse en la Ilustración 21 - Modelo de datos de la BBDD.

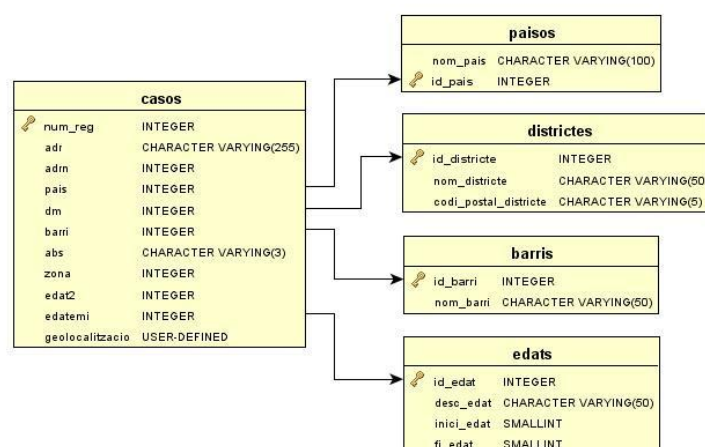


Ilustración 21 - Modelo de datos de la BBDD

¹³ Existen servicios web públicos para geocodificar las direcciones postales. Algunos ejemplos pueden relevantes son los sistemas de geocodificación de Yahoo! y Google.

8.1.2 Carga de datos

Posiblemente durante la implementación de la aplicación sea necesario crear diversas vistas para la obtención de datos más específicos, como puede ser la latitud y la longitud de cada punto.

Debido que la base de datos ha sido proporcionada ya con el formato de *PostgreSQL*, simplemente ha sido necesario configurar una nueva base de datos a partir de una plantilla de *PostGIS* y ejecutar el script origen. De esta forma obtenemos ya una base de datos geo-espacial que podemos administrar fácilmente desde la aplicación *pgAdmin III* (véase Ilustración 22 - Carga de datos en BBDD espacial).

Por otra parte, los datos cartográficos recibidos en formato “shp”, aunque *GeoServer* tiene la capacidad de tratar este tipo de archivo de datos de forma nativa, se ha decidido importar todos los datos a *PostGIS*, para unificar a un único sistema de almacenamiento de información.

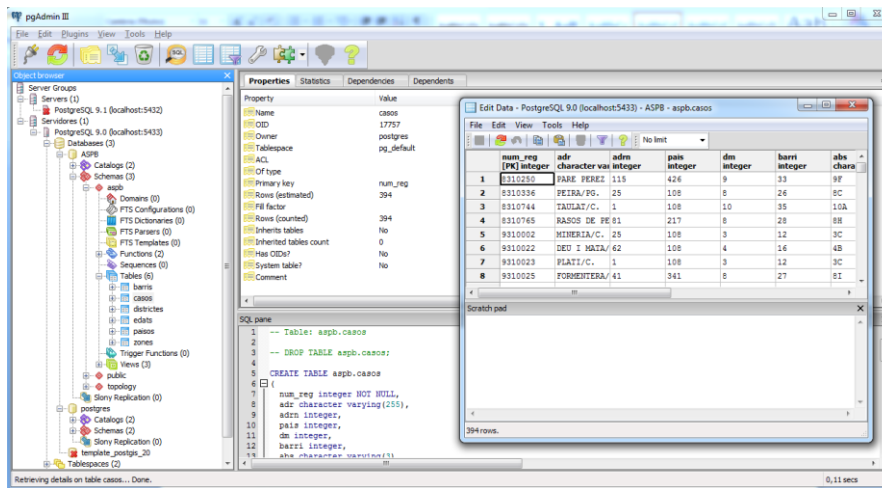


Ilustración 22 - Carga de datos en BBDD espacial

8.1.3 Futuro mantenimiento de los datos

Aunque los datos disponibles son más que suficientes para elaborar el proyecto y generar las pruebas unitarias y funcionales del mismo, es posible que en un futuro próximo tengan que ser actualizadas o renovadas.

Para realizar este proceso disponemos de un aplicativo (véase Ilustración 23 - Uso del aplicativo para el mantenimiento de los datos.) que nos permite geocodificar las direcciones postales utilizando un servicio web como los ofrecidos por Yahoo! o Google.

La idea principal de este aplicativo es ofrecer la posibilidad de introducir datos de forma masiva sin tener nociones de programación. El uso del aplicativo es muy simple y consta de las siguientes etapas:

- Se cargan los datos desde la base de datos existente.
- Se seleccionan los diversos registros que se quieran geocodificar. Este proceso utiliza los servicios de geocodificación ofrecidos por Yahoo!
- Es posible comprobar los datos resultantes mediante los mapas ofrecidos por Yahoo! o Google.

- En caso de ser necesario se ofrece la posibilidad de modificar manualmente los parámetros (latitud, longitud, etc).
- Finalmente se guarda todo el proceso de nuevo en la base de datos.

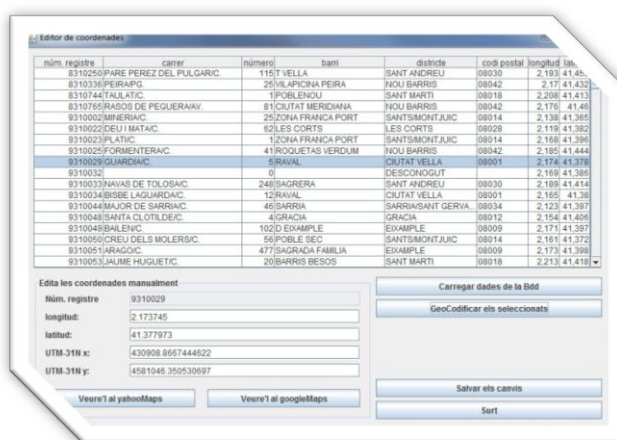


Ilustración 23 - Uso del aplicativo para el mantenimiento de los datos.

8.2 GeoServer, servicios web de tratamiento y gestión de datos.

Esta parte de la aplicación es crucial y se encargara de hacer de intermediario y realizar las comunicaciones necesarias entre la aplicación cliente y la base de datos geo-espacial. Las funciones principales de esta herramienta son:

- Realizar operaciones constantes de tipo CRUD¹⁴ contra el SGBD elegido.
- Responder mediante uno o varios servicios web¹⁵ de tipo REST¹⁶, se ha elegido este tipo de servicio web en lugar de los servicios más extensos de tipo SOAP¹⁷ por su fácil configuración e interpretación a las consultas (peticiones vía http) realizadas por la aplicación cliente y responder a las mismas con la información obtenida del SGBD.

Así mediante la instalación de *GeoServer*¹⁸ y su correspondiente configuración que pasaremos a describir a continuación obtenemos todos los servicios web necesarios para nuestra aplicación.

Aunque la especificación de los detalles de la siguiente Ilustración 24 - Esquema global de un servicio web mediante GeoServer ofrecida por el OGC es más detallada de lo necesario si nos centramos en el flujo de datos representados por la línea verde puede observarse el flujo principal de nuestra aplicación (IUWeb – Servidor – BBDD) y podemos observar que mediante *GeoServer* podemos satisfacer todas nuestras necesidades.

¹⁴ Create, Read, Update and Delete.

¹⁵ Un servicio web (en inglés, *Web service*) es una pieza de software que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

¹⁶ *Representational State Transfer*. <http://es.wikipedia.org/wiki/REST>

¹⁷ *Simple Object Access Protocol*. <http://es.wikipedia.org/wiki/SOAP>

¹⁸ *GeoServer* - Un servidor de código abierto escrito en Java - permite a los usuarios compartir y editar datos geospaciales.

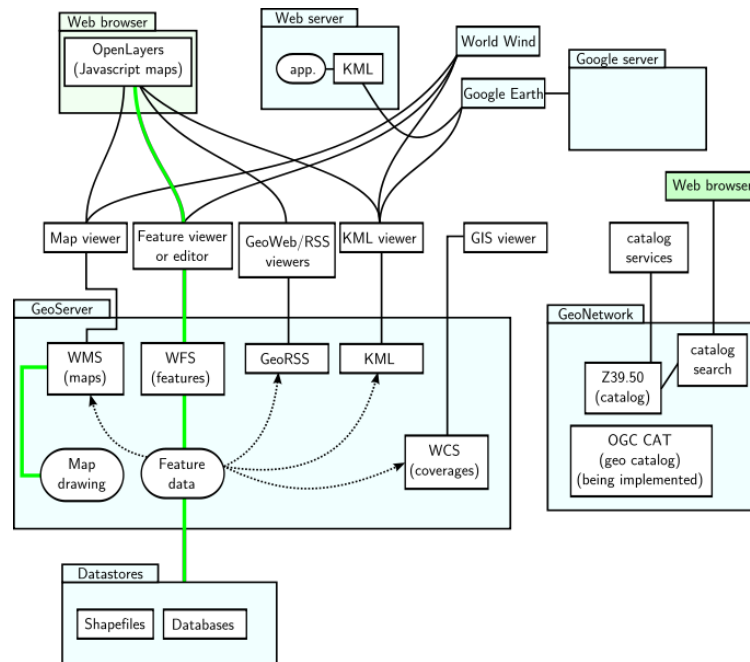


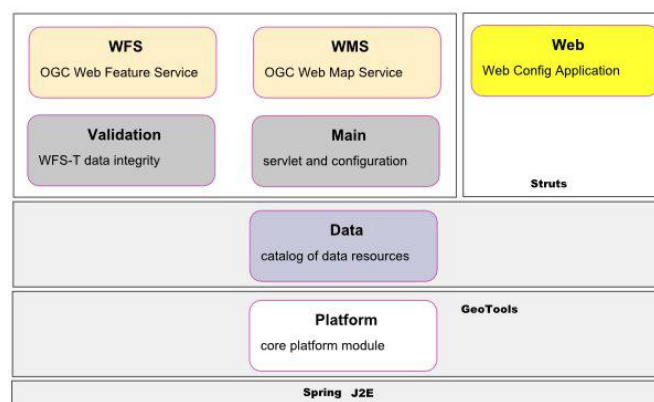
Ilustración 24 - Esquema global de un servicio web mediante GeoServer

Como hemos comentado si seguimos la línea verde en la anterior ilustración se puede ver el flujo principal de nuestra aplicación.

EL usuario desde un cliente Web, se conecta al aplicativo y realiza una serie de operaciones y consultas. *GeoServer* mediante los diferentes servicios configurados WMS y WFS nos permite gestionar estas consultas y realizar las peticiones necesarias a nuestra Base de datos espacial.

Una vez obtenida la información es *GeoServer* quien se encarga de gestionar dicha información y generar las imágenes (mapas) correspondientes para ser visualizados mediante el cliente web.

La aplicación *GeoServer* está formada por una serie de módulos:



8.3 Aplicación cliente

La aplicación cliente es la parte más visual de todo el proyecto y será la herramienta que utilizara el usuario final para realizar las consultas, gestionar la información geo-espacial y visualizarla sobre una representación gráfica.

El objetivo principal del proyecto es la de crear una web interactiva i dinámica para la gestión de datos espaciales y por lo tanto la parte visual conocida como Interfaz de usuario (UI) tiene una gran relevancia. La idea principal es realizar esta etapa mediante las nuevas tecnologías web (HTML5, jQuery, etc.) para su posible acceso desde cualquier terminal móvil.

Para el desarrollo de esta sección del proyecto se han utilizado diferentes tecnologías. Aunque el mercado ofrece diversas opciones para el desarrollo de la parte cliente como puede ser la API de ArcGIS para Flex o la API para Javascript de la misma empresa. Otra opción y la elegida en este proyecto por su carácter *Open Source* ha sido la API de *OpenLayer*, junto a un *framework*¹⁹ de javascript llamado "GeoExt" que nos permitirá realizar un entorno mucho más profesional.

La comunicación servidor-cliente cumple los estándares del OGC que documenta como ofrecer información geográfica de una manera estándar a través de un servidor web y establece la comunicación con los servidores, por lo general, las peticiones a los servicios se realizan en forma de petición HTTP GET, o en su caso HTTP POST. La respuesta a peticiones WMS del tipo **GetCapabilities** será un fichero XML y la respuesta a una petición tipo **GetMap** tendrá el formato gráfico o vectorial solicitado si la petición tiene éxito, si la petición falla enviará un fichero XML describiendo el error.



Dentro de los estándares de OGC que vamos a utilizar en esta aplicación podemos destacar tres especificaciones importantes: GML, WMS y WFS.

- **GML (Geography Markup Language):** Lenguaje basado en XML mediante el cual se establece el estándar para modelar, transferir y almacenar información de contenido Geográfico.
- **WMS:** Especificaciones que estandarizan el envío y recepción de información geoespacial, es tan sencillo como escribir en un navegador web la dirección o URL a la que apunta el servicio demandado, el resultado siempre será el mismo, una imagen digital en cualquiera de los formatos utilizados en la web (PNG, GIF, JPEG).
- **WFS:** Especificaciones de implementación que permiten obtener y actualizar datos geoespaciales desde una base de datos a través de Internet.

8.3.1 Llamada y uso del servicio web WMS

WMS nos ofrece una serie de parámetros que son pasados vía web, junto con el nombre del recurso. Estos parámetros son los siguientes:

- **Request:** es el más importante, y en el que nos vamos a centrar puesto que es el que indica que acción tomará. Puede tener los siguientes valores:
 - **GetMap:** obtiene una imagen.
 - **GetCapabilities:** determina las características del recurso WMS.
 - **GetFeatureInfo:** permite consultar el mapa.
- **Versión:** indica la versión del servicio WMS según el OGC.
- **Layers:** especifica las capas de mapa de visualizar.
- **SRS:** Indica la proyección cartográfica.
- **Width:** Ancho de la imagen a obtener.
- **Height:** Altura de la imagen a obtener.
- **Bbox:** Coordenadas de un rectángulo necesarias para realizar un zoom.

Un ejemplo sería la llamada request con el valor GetMap, esto nos permitiría obtener una imagen de la zona solicitada con información concreta. Esta llamada es la más común en los servicios WMS.

```
http://localhost:8080/geoserver/wms?  
bbox=516323,4251040,517864,4252195&  
format=image/png&  
  
request=GetMap&  
layers=aspb:distritos&  
width=400&  
height=300&  
srs=EPSG:23030
```

Ilustración 25 - Petición WMS del tipo "GetMap"

Otro ejemplo sería la consulta mediante el parámetro request "GetFeatureInfo" esta consulta es de gran interés ya que nos permite obtener información sobre elementos del mapa. Una vez realizada la consulta podemos obtener la respuesta en diferentes formatos: texto plano, GML y HTML.

```
http://localhost:8080/geoserver/wms?  
format=image/png&  
request=GetFeatureInfo&  
layers=aspb:distritos&  
bbox=516323,4251040,517864,4252195&  
width=400&  
height=300  
&srs=EPSG:23030  
&info_format=gml
```

Ilustración 26 - Petición WMS del tipo "GetFeatureInfo"

8.3.2 Llamada y uso del servicio web WFS

Para poder controlar los resultados del documento generado el protocolo WFS nos ofrece también una serie de parámetros que son pasados vía web junto con el nombre del recurso. Estos parámetros son los siguientes:

- **Request:** Puede tener los siguientes valores:
 - **GetFeature:** obtiene elementos del mapa en forma de documento estructurado.
 - **GetCapabilities:** determina las características del recurso WFS.
- **Versión:** indica la versión del servicio WFS según el OGC.
- **TypeName:** especifica la capa de mapa que queremos obtener.
- **SRS:** Indica la proyección cartográfica.
- **Bbox:** Coordenadas de un rectángulo necesarias para acotar el área de estudio.

De manera general el protocolo que sigue este estándar es el siguiente:

- La aplicación cliente solicita el documento de capacidad del servicio, que describe todas las operaciones que soporta el servicio WFS y una lista de todos los tipos de características que puede servir.
- La aplicación cliente, opcionalmente, realiza una petición al servidor para obtener la definición de uno o más de los tipos de elementos a servir.
- La aplicación cliente genera una petición basándose en la definición de los tipos de elementos.
- La petición se envía a un servidor web.
- El WFS se invoca para leer y servir la petición.
- Cuando el WFS ha completado de procesar la petición genera un informe de estado y lo devuelve al cliente.

8.3.3 Creación de la interfaz de usuario IU

La interfaz es la parte de la aplicación en la cual los usuarios interactúan, aquí vamos a definir la forma en la que van a aparecer los objetos de la interfaz que activarán la navegación y el resto de las funcionalidades de la aplicación.

Para la creación y desarrollo del visor hemos utilizado *OpenLayers*, librería en *JavaScript* que tiene la cualidad de poder acceder a información geoespacial ya sea de servidores que cumplan con el estándar WMS y WFS del *Open Geospatial Consortium* como de los principales servicios comerciales que hay en la actualidad.

Todo el código de la aplicación está escrito en HTML (*HyperText Markup Lenguaje*), es el lenguaje que se utiliza para crear páginas Web que puedan ser interpretadas por los navegadores de Internet.

El uso principal de *JavaScript* es añadir comportamiento dinámico, como modificación de elementos, interacción con el usuario etc., a las páginas en HTML, que de otra forma serían estáticas.

OpenLayers tiene una interfaz por defecto (véase Ilustración 27 - Interfaz por defecto de OpenLayer), que es la que vamos a utilizar, sin embargo, es posible añadir un número de opciones de configuración a través de la clase de controles, como una barra para ajustar el nivel de zoom, posición del cursor, etc.

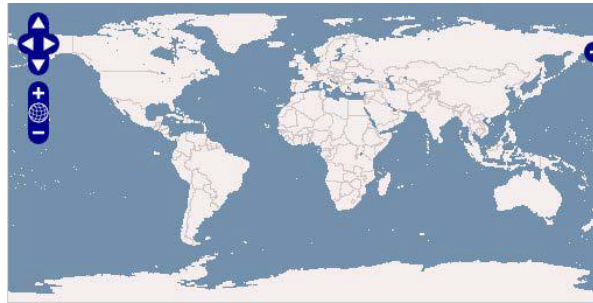


Ilustración 27 - Interfaz por defecto de OpenLayer

Se realizara una única pantalla dividida en dos secciones. La intención principal es que el usuario no tenga que desplazarse por menús complejos y llegue a perder de vista el mapa con la información representada.

Por tanto se dividirá la pantalla en dos secciones una barra lateral con las opciones de filtraje y búsqueda y una sección principal donde se visualizara mediante la interfaz por defecto de *OpenLayer* el mapa con toda la información.

Utilización de la API de OpenLayer para la visualización de datos

A continuación explicaremos algunas formas de representar y manipular datos a partir de la API de *OpenLayer* y los WMS generados mediante *GeoServer*.

OpenLayers, básicamente utiliza un sistema de paquetes a través del directorio base, creando una jerarquía de objetos, que en producción se comprime en un único fichero .js, las clases básicas de *OpenLayers* son:

- **Map.js:** Es el objeto central de *OpenLayers* y contiene a todos los demás.
- **Layer.js:** Cada capa hereda los métodos básicos de esta capa.
- **Control.js:** Controles del mapa, tienen una relación 1-1 con los *handlers* y generalmente son los elementos que se suelen personalizar.
- **Handler.js:** Son los manejadores de eventos, están asociados a los eventos típicos de la web como son: "Click", "MouseWheel", "Hover"...

En el siguiente bloque de código (véase Ilustración 28 - Código OpenLayer) se puede observar la invocación e inicialización de un mapa mediante *OpenLayer*.

```
function init(){  
    map = new OpenLayers.Map( 'map', {  
        maxExtent: new  
        OpenLayers.Bounds(567920.01,4411366.8,816289.4,4763637.1) ,  
        maxResolution: 'auto',  
        units: "m",  
        projection: "EPSG:23030"}  
    }  
}
```

Ilustración 28 - Código OpenLayer

9. Conclusiones

En primer lugar, es importante mencionar que a pesar de la satisfacción de alcanzar el objetivo planteado en este trabajo, también se ha alcanzado otro objetivo no menos importante. Durante el desarrollo del mismo se han adquirido grandes conocimientos antes desconocidos sobre las tecnologías y herramientas utilizadas y en especial sobre los sistemas de información geográfica. Aunque la mayoría de los conocimientos adquiridos han sido teóricos, algunos otros se han puesto en práctica alcanzando la experiencia necesaria para llevar a cabo cualquier proyecto de características similares.

La aplicación Web diseñada durante la realización de este trabajo creo que representa una herramienta de gran utilidad para la monitorización de enfermedades. Permitiendo a los expertos del ASPB conocer, de manera rápida y fácil, los casos existentes y su distribución en el municipio de Barcelona y la posible relación con otros datos cartográficos de interés.

Se puede afirmar que el conjunto de herramientas (*PostGis*, *GeoServer* y *OpenLayers*), gracias a su carácter *Open Source*, es una estupenda combinación tecnológica para el desarrollo de aplicaciones en entornos web, permitiendo romper con las limitaciones de acceso del software propietario y permitiendo elaborar aplicaciones de gran calidad y una elegante interfaz.

Otro aspecto muy importante es que la aplicación desarrollada se comunica con los servicios web a través de estándares, y utiliza estándares de estilos y filtros, todos ellos transparentes a los usuarios.

Para finalizar creo que también deberíamos tener en cuenta que el desarrollo de la aplicación se ha realizado en todo momento con la premisa de ser escalable. Con esto se consigue que el mantenimiento y ampliación de la aplicación sea factible en cualquier momento y no altere al uso diario de los usuarios.

Una vez explicada la situación final, puede considerarse que este proyecto ha cumplido en un 100% los objetivos y funcionalidades planteados, pudiendo ser implementado de forma eficaz.

10. Futuras ampliaciones del trabajo

Una vez cumplido el objetivo del trabajo que es ofrecer a partir de los sistemas de información geográfica otro punto de vista a los expertos de la ASPB a la hora de analizar los datos existentes sobre la enfermedad de la tuberculosis, se plantean a continuación una serie de posibles mejoras a realizar sobre la aplicación para hacerla más interesante en un futuro.

Algunas de ellas aunque no las únicas podrían ser:

- Introducir datos de otras enfermedades, para poder realizar el mismo tipo de estudio sobre ellas.
- Ampliar la información existente sobre las enfermedades (por ejemplo una ampliación de la ventana temporal) y representarla de forma visual.
- Dar la posibilidad de generar gráficos o tablas de datos a partir de los cálculos y filtrajes realizados mediante los sistemas SIG.
- Combinar más datos geográficos que puedan considerarse de interés para los expertos de la ASPB (renta per cápita de las zonas, densidad de población, nivel de estudios o índice de delincuencia).
- Dar la posibilidad de generar polígonos sobre el mapa y obtener la información de los casos existentes en ellos.
- Aumentar los criterios de selección y filtraje.

Debido que el trabajo se ha realizado desde su inicio con la intención de que sea escalable en un futuro, todas estas mejoras pueden ir introduciéndose a corto/largo plazo sin tener que redefinir todo el proyecto desde cero.

11. Glosario

ArcGIS: es el nombre de un conjunto de productos de software en el campo de los Sistemas de Información Geográfica o SIG. Producido y comercializado por Esri, bajo el nombre genérico ArcGIS se agrupan varias aplicaciones para la captura, edición, análisis, tratamiento, diseño, publicación e impresión de información geográfica. Estas aplicaciones se engloban en familias temáticas como ArcGIS Server, para la publicación y gestión web, o ArcGIS Móvil para la captura y gestión de información en campo.

CRUD: En computación CRUD es el acrónimo de Crear, Obtener, Actualizar y Borrar (del original en inglés: *Create, Read, Update and Delete*). Es usado para referirse a las funciones básicas en bases de datos o la capa de persistencia en un sistema de software.

Escalabilidad: Capacidad de aumentar o mejorar el funcionamiento de una aplicación informática sin comprometer el funcionamiento actual de la misma.

RIA: Las *rich Internet applications*, o RIA (en español "aplicaciones de Internet enriquecidas"), son aplicaciones web que tienen la mayoría de las características de las aplicaciones de escritorio tradicionales. Estas aplicaciones utilizan un navegador web estandarizado para ejecutarse y por medio de complementos o mediante una máquina virtual se agregan las características adicionales. Las RIA surgen como una combinación de las ventajas que ofrecen las aplicaciones web y las aplicaciones tradicionales. Buscan mejorar la experiencia del usuario.

FLEX: Entorno de desarrollo para la creación de aplicaciones RIA de la empresa de software Adobe Systems.

JDBC: Del Inglés Java Database Connectivity, es un estándar para la creación de librerías utilizadas para la conexión entre el lenguaje java y cualquier base de datos relacional.

SGBD: Sistema de gestión de base de datos. Programas destinados a la gestión de base de datos relacionales, de sus datos, usuarios, accesos, procedimientos y funciones.

Eclipse: Entorno de desarrollo para la creación de aplicaciones diversas, en este proyecto se utilizara este IDE para la creación de los servicios web.

Geocodificación: Proceso para la obtención de las coordenadas geográficas a partir de una dirección postas o cualquier referencia textual.

Geoprocesamiento: Procesamiento de datos geográficos utilizando diversas herramientas geográficas.

Sistema de coordenadas: Conjunto de parámetros que permiten definir cualquier punto, en este caso de la superficie de la tierra, mediante un punto de referencia y un conjunto finito de valores.

Latitud: La latitud es la distancia angular entre la línea ecuatorial (el ecuador), y un punto determinado del planeta, medida a lo largo del meridiano en el que se encuentra dicho punto.

Se abrevia con lat. Según el hemisferio en el que se sitúe el punto, puede ser latitud norte o sur.

Longitud: La longitud, abreviada long., en cartografía, expresa la distancia angular entre un punto dado de la superficie terrestre y el meridiano que se tome como 0° (es decir el meridiano base), tomando como centro angular el centro de la Tierra; habitualmente en la actualidad el meridiano de Greenwich.

API: Interfaz de programación de aplicaciones o API (del inglés *Application Programming Interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas (también denominadas vulgarmente "librerías").

SOAP: (siglas de *Simple Object Access Protocol*) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por David Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros y está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

REST: La Transferencia de Estado Representacional (*Representational State Transfer*) o REST es una técnica de arquitectura software para sistemas hipermedia distribuidos como la *World Wide Web*. El término se originó en el año 2000, en una tesis doctoral sobre la web escrita por Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP y ha pasado a ser ampliamente utilizado por la comunidad de desarrollo.

12. Bibliografía

12.1 Documentos

Alfons Bataller Díaz, I. (2012) “*Gestió i desenvolupament de projectes*”, Universitat Oberta de Catalunya

Nita Sáenz Higuera, I. (2012) “*Redacció de textos científicotècnics*”, Universitat Oberta de Catalunya

Roser Beneito Montagut, I. (2012) “*Presentació de documents i elaboració de presentacions*”, Universitat Oberta de Catalunya

12.2 URL's

Instituto cartográfico de Cataluña, <http://www.icc.es/>

Instituto cartográfico Nacional, <http://www.ing.es/>

Servicio de Epidemiología de Barcelona, Agencia de Salud Pública (ASPB), <http://www.aspb.es>

OGC, *Open Geospatial Consortium*. <http://www.opengeospatial.org/standards>

Gabriel Ortiz, <http://foro.gabrielortiz.com/>

GeoServer, <http://geoserver.org/>

OpenLayers, <http://openlayers.org/>, <http://trac.osgeo.org/openlayers/wiki/TitleIndex>

OpenLayers, *JavaScript Mapping Library*, <http://dev.openlayers.org/releases/OpenLayers-2.10/doc/apidocs/files/OpenLayers-js.html>

OpenLayers manual, <http://www.ingemoral.es/pages3/manualOpenLayers.html>

Implement REST Web Services in Java, <http://www.dosideas.com/noticias/java/314-introduccion-a-los-servicios-web-restful.html>

Apuntes de Java, RESTful la forma más ligera de crear un WebServices, <http://www.apuntesdejava.com/2010/11/restful-la-forma-mas-ligera-de-hacer.html>

REST vs. SOAP: www.taranfx.com/rest-vs-soap-using-http-choosing-the-rightwebservice-protocol

Introduction to *PostGIS*, <http://workshops.opengeo.org/postgis-intro/>

Manuel Martín Martín, “*Manual PostGIS*”, traducción del texto de Paul Ramsey a lengua castellana. <http://postgis.refractory.net/documentation/postgis-spanish.pdf>

Servicios web de Google Maps, <https://developers.google.com/maps/documentation/webservices/index?hl=es>

API de codificación geográfica de Google, <https://developers.google.com/maps/documentation/geocoding/?hl=es>

Yahoo! Maps Web Services – Geocoding API, <http://developer.yahoo.com/maps/rest/V1/geocode.html>