

Universidad de las Ciencias Informáticas

Facultad 4



**Título: Plataforma de servicios web para el procesamiento
y gestión de información geoespacial**

Trabajo de Diploma para optar por el título de:
Ingeniero en Ciencias Informáticas

Autor: Eliana Bárbara Ril Valentin

Tutor: MSc. Rafael Rodríguez Puente

Ciudad de La Habana, Cuba

Mayo de 2008

“Año 50 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de la presente tesis y autorizo a la Facultad # 4 de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los _____ días del mes de mayo de 2008.

Eliana Bárbara Ril Valentin

MSc. Rafael Rodríguez Puente

Agradecimientos

A la Revolución, por la oportunidad de servirle

A nuestro Comandante en Jefe, por hacer realidad mis sueños

A la UCI, por ser mi segunda casa, por hacerme crecer

A mi novio y amigo en estos 5 años de carrera, Rafa, no solo por su certera guía, su perseverancia y exigencia, sino por el amor y todo el esfuerzo que le robo

A ti, que no te he mencionado y sonríes cuando llegas a esta línea, por existir

Dedicatoria

*A mi familia, en especial a mi mamá, abu y Zule
por ser mi fuerza, mi inspiración, mi soporte, mi consuelo
hoy y siempre*

RESUMEN

Actualmente la tendencia para elaborar aplicaciones informáticas está haciendo énfasis hacia el incremento de la interoperabilidad y la creación de aplicaciones multiplataforma.

Con este Trabajo de Diploma se pretende desarrollar un conjunto de servicios web que garanticen el procesamiento, obtención y representación de la información geoespacial. Disminuyendo así los tiempos de desarrollo de funcionalidades particulares para cada una de las aplicaciones de manera independiente. Los servicios web permiten la construcción de aplicaciones multiplataforma, con ventajas de personalización, redistribución, interoperabilidad y reutilización de mejoras de desarrollo que ofrecen estas aplicaciones.

Para su desarrollo se modelaron los flujos de trabajo de Requerimientos, Análisis y Diseño e Implementación, teniendo en cuenta el Proceso de Desarrollo Unificado (RUP) y el lenguaje de modelado UML. Las tecnologías y herramientas utilizadas fueron: servidor web Apache, PostgreSQL, PHP, MapServer, todas siguiendo los lineamientos de arquitectura trazados por la Universidad de las Ciencias Informáticas (UCI), de software libre. Se aplicó además la arquitectura en capas.

Logrando de esta manera dar un paso de avance en la automatización del procesamiento, obtención y representación de la información geoespacial, al permitir la creación de un sistema capaz de facilitar el proceso de toma de decisiones en la UCI.

PALABRAS CLAVE

Servicios web, información geoespacial, interoperabilidad, software libre, aplicaciones multiplataforma, sig.

TABLA DE CONTENIDOS

RESUMEN.....	I
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	6
1. Introducción.....	6
2. Tecnologías de información geoespacial sobre la web en el mundo.....	6
3. Tecnologías de información geoespacial sobre la web en Cuba.....	9
4. Tecnologías de información geoespacial sobre la web y el software libre.....	11
5. Tecnologías Actuales.....	12
5.1 Servicios web.....	12
5.2 Arquitectura orientada a servicios.....	15
6. Metodología a utilizar.....	17
6.1 Proceso Unificado del Rational (RUP).....	17
7. Herramientas a utilizar.....	18
7.1 Herramienta CASE: Visual Paradigm-UML.....	18
7.2 Servidor de mapa: MapServer.....	19
7.3 Lenguaje de programación: PHP.....	20
7.4 Sistema Gestor de Bases de Datos: PostgreSQL.....	21
8. Conclusiones.....	23
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	24
1. Introducción.....	24
2. Propuesta de sistema.....	24
3. Modelo de dominio.....	25
3.1 Definición de las entidades y conceptos fundamentales.....	25
3.2 Representación del Modelo de dominio.....	26
4. Especificación de los requisitos de software.....	27
4.1 Requerimientos funcionales.....	27
4.2 Requerimientos no funcionales.....	28
5. Modelo del sistema.....	29
5.1 Definición de los actores.....	29
5.2 Definición de los casos de uso.....	29
5.2.1 Descripción de los casos de uso.....	29
5.2.2 Diagrama de casos de uso del sistema.....	30
5.2.3 Descripción detallada de los casos de uso.....	31
6. Conclusiones.....	44

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	45
1. Introducción.....	45
2. Modelo de análisis.....	45
2.1 Diagramas de clases de análisis.....	46
3. Modelo de diseño.....	47
3.1 Arquitectura del sistema	48
3.2 Patrones de diseño.....	49
3.3 Diagramas de clases del diseño.....	51
3.4 Descripción de las clases del diseño.....	54
3.5 Diagramas de interacción.....	57
4. Diseño de la base de datos.....	57
4.1 Diagrama de clases persistentes.....	58
4.2 Modelo de datos.....	59
4.3 Descripción de las tablas de la base de datos	59
5. Conclusiones.....	60
CAPÍTULO 4: IMPLEMENTACIÓN.....	61
1. Introducción.....	61
2. Modelo de despliegue.....	61
3. Modelo de implementación.....	62
4. Conclusiones.....	64
CONCLUSIONES GENERALES.....	65
RECOMENDACIONES.....	66
REFERENCIAS BIBLIOGRÁFICAS.....	67
BIBLIOGRAFÍA CONSULTADA.....	70
ANEXO I ESTÁNDARES ISO TC 211.....	75
ANEXO II DIAGRAMAS DE SECUENCIA.....	82
ANEXO III DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATOS.....	92
GLOSARIO DE TÉRMINOS Y ACRÓNIMOS.....	95

INTRODUCCIÓN

El significativo desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) ha conllevado a la creación de nuevas herramientas en los más diversos campos, que permiten mejorar el proceso de toma de decisiones en toda la sociedad. Se ha demostrado que en dichos procesos entre el 80 y el 90% de la información que se maneja tiene una componente espacial. Los negocios cada vez exigen crear aplicaciones más complejas, con menos tiempo y presupuesto que antes. Crear estas aplicaciones, requiere en muchos casos de funcionalidades ya antes implementadas como parte de otros sistemas.

En la Universidad de las Ciencias Informáticas (UCI) existe la necesidad de representar información geoespacial. Debido a su gran extensión territorial y continuo crecimiento, cada vez se hace más complejo el manejo y localización de la información para los que conviven en ella, por ejemplo: ubicar un edificio de la residencia, un laboratorio específico, o un docente determinado, entre otros. Además a esto se suma, que es continuamente visitada por personalidades interesadas en conocer detalles de su estructura geográfica y hoy sólo se les puede mostrar de forma muy general con La Maqueta ubicada en el centro.

La mayoría de los proyectos productivos se encuentran aislados entre sí, provocando que se dupliquen funcionalidades e información. Los resultados que se obtienen se convierten en aplicaciones aisladas, que pueden ser usadas por diferentes entidades.

Actualmente no existen servicios web estándares que posibiliten la integración de todas las aplicaciones que se desarrollan para la representación de la información geoespacial. Además, todo el que necesita representar dicha información debe tener conocimientos de Geomática.

Luego de haberse realizado un profundo análisis de las dificultades existentes surge el

siguiente **problema de la investigación**: La Universidad de las Ciencias Informáticas no cuenta hoy con una plataforma de servicios web que le permita el procesamiento, obtención y representación de la información geoespacial.

Con este Trabajo de Diploma se pretende desarrollar una **Plataforma de servicios web** que brinde una serie de funcionalidades propias de los Sistemas de información geográfica (SIG) y permita a los desarrolladores de software (sin necesidad de tener conocimientos de geomática) crear aplicaciones que se encarguen del procesamiento, obtención y representación de la información geoespacial, seleccionando cuál de las funcionalidades de las que se les brinda desean utilizar. Disminuyendo así los tiempos de desarrollo de funcionalidades particulares para cada una de las aplicaciones de manera independiente y adicionando la interoperabilidad para el trabajo con la plataforma de servicios propuesta.

En correspondencia con lo formulado, el **objeto de estudio** estará enmarcado en los procesos de gestión y procesamiento de la información geoespacial para su representación.

Las tecnologías de información geoespacial sobre la web sería la parte del objeto de estudio que se va a investigar, es decir, el **campo de acción**.

Apoyados en lo expuesto anteriormente y para guiar la investigación se plantea la siguiente **hipótesis**: Si la UCI contara con una plataforma de servicios web para el procesamiento, obtención y representación de la información geoespacial, entonces los desarrolladores no necesitarían tener conocimientos de geomática para incluir información geoespacial en sus aplicaciones y dicha información estaría centralizada.

Para lograr esto se ha trazado como **objetivo general** desarrollar un conjunto de servicios web que garanticen el procesamiento, obtención y representación de la información geoespacial en la Universidad de las Ciencias Informáticas.

Para poder cumplir el objetivo anteriormente planteado se proponen las siguientes

Tareas de la Investigación:

- Analizar la información existente relacionada con las tecnologías de información geoespacial sobre la web.
- Seleccionar la metodología de análisis y diseño, que facilite y garantice la creación con calidad de la plataforma de servicios.
- Evaluar las tendencias actuales en cuanto a las herramientas informáticas más utilizadas.
- Seleccionar las herramientas que se ajustan a las necesidades de la investigación.
- Desarrollar un conjunto de servicios web que faciliten la gestión de mapas, utilizando la metodología de desarrollo de software escogida.

Como todo proceso investigativo se utilizaron métodos para el desarrollo de la investigación, se tuvieron en cuenta los métodos de investigación científica, que presentamos a continuación:

Métodos Teóricos:

Analítico – sintético: Este método se utiliza para analizar la situación problemática y determinar posibles variantes de solución.

Hipotético – deductivo: A partir de la hipótesis, se elaboran conclusiones acerca de la factibilidad de la utilización de una interfaz de servicios para la manipulación de mapas digitales que brinde la posibilidad de integrar los datos de las diferentes aplicaciones que conviven en una comunidad de desarrollo de software, sin tener conocimientos de la

Geomática y tecnologías asociadas.

Modelación: Se crean modelos a través de la metodología RUP.

Métodos Empíricos:

Entrevista: Se realizan entrevistas con el objetivo de precisar el problema a resolver y para asumir determinados criterios que son imprescindibles para realizar el desarrollo del sistema.

Estudiar y documentarse sobre el tema de los servicios web y su desarrollo en el mundo para poder aplicar esta tecnología en el trabajo.

El presente documento está estructurado por cuatro capítulos, los cuales abarcan todo lo relacionado con la investigación realizada, análisis y diseño de la aplicación propuesta, así como las especificaciones de la implementación de la misma.

Capítulo 1. Fundamentación Teórica: En este capítulo se realiza un análisis del estado del arte a nivel internacional y nacional de las tecnologías de información geoespacial sobre la web, que brindan la funcionalidad de gestionar información geográfica. Además, se tratan aspectos fundamentales para la comprensión del sistema que se desea modelar, los conceptos más importantes, las tendencias actuales, las técnicas y tecnologías, entre otros aspectos de vital importancia para el desarrollo de esta investigación.

Capítulo 2. Características del sistema: Se aborda el entorno donde se implantará el sistema y se presentan los requerimientos funcionales y no funcionales que este debe cumplir. Además se realiza una propuesta de sistema a través de la definición de los casos de uso que guiarán todo el proceso de desarrollo. Así como una descripción detallada de los mismos.

Capítulo 3. Análisis y Diseño del sistema: Se modelan los diagramas de clases del

análisis y diseño, los diagramas de secuencia y de clases persistentes que propone la metodología empleada, así como el Modelo de Datos y la descripción de forma general de las clases utilizadas. Además se especifican los patrones para el diseño de la aplicación y la arquitectura que soportará la misma.

Capítulo 4. Implementación: Refleja los aspectos relacionados con la construcción de la solución propuesta. Se representa el modelo de implementación, o sea, los diagramas de componentes y el diagrama de despliegue.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1. Introducción

En el presente capítulo se brinda una visión general de los aspectos relacionados con las tecnologías de información geoespacial sobre la web, a través del análisis de sus disponibilidades a escala mundial y nacional. También se muestra una breve panorámica de las características de cada tipo de herramientas y tecnologías actuales utilizados para la confección del sistema. Además se ofrece una pequeña descripción de la metodología de análisis y diseño escogida, así como los conceptos necesarios para el estudio de los servicios web que son necesarios para entender la propuesta de solución.

2. Tecnologías de información geoespacial sobre la web en el mundo

El continuo crecimiento de los volúmenes de información y el dinamismo que caracteriza a las empresas en la actualidad, han provocado que el procesamiento de datos espaciales y la posibilidad de extraer información a partir de ellos se haya convertido en un factor importante para el apoyo a la toma de decisiones.

Diferentes grupos a nivel nacional (EE.UU., Canadá, España), regional (Europa) e internacional (ISO, FAO, UNESCO), se han dedicado a la generación de estándares relacionados con la información geográfica. Entre los más importantes se encuentran los liberados por OpenGIS, tales como:

1. Especificaciones abstractas y de implementación del Servicio Web de Mapas (WMS)
2. Especificaciones de implementación del Servicio de Catálogo
3. Especificaciones de implementación del Servicio Web de Coberturas (WCS)

4. Especificaciones de implementación del Servicio Web de Elementos (WFS)

En el caso de la especificación Web Map Service (WMS) o Servicios Web de Mapas brinda una interfaz común para consultar y acceder a capas de mapas en un servidor de mapas. Los clientes y servidores que se adhieren al estándar OGC/WMS pueden comunicarse entre sí, sin importar la arquitectura o implementación particular de cada uno. El software cliente obtiene del servidor imágenes y código XML de georeferenciación, como resultado de los criterios enviados al servidor en las peticiones de acuerdo al estándar.[CCIDEP 2008]

Un servicio WMS brinda las siguientes funcionalidades:

1. *GetCapabilities* (Información del servidor, retorna un documento XML que describe las capas temáticas de información disponible y las proyecciones que soporta)
2. *GetMap* (Generación de imágenes de mapa, retorna la imagen del mapa en formato gif, jpeg, swf o png)
3. *GetFeatureInfo* (Consulta de atributos, text/plain y text/html únicamente, retorna información sobre el objeto geográfico en XML, GML, HTML, o texto)

Existen varios servidores y librerías para el manejo de mapas que cumplen con el estándar WMS, como son: MapServer, GeoServer, GeoTools entre otros; los mismos son muy utilizados a nivel mundial, principalmente MapServer y solucionan el problema de compartir los datos aunque se hace engorroso su uso, ya que para utilizar la funcionalidad principal que brindan (GetMap), el desarrollador debe de pasar un significativo número de parámetros (nombre del mapa, nombre de cada una de las capas que se quiere mostrar, qué porción del mapa se desea mostrar minx, miny, maxx, maxy, entre otros), lo que implica además de hacer el trabajo complejo, tener conocimientos de Geomática.

La iniciativa más reciente, OGC Web Services Phase 3 (OWS-3), está encargada de la

definición de una arquitectura que permita la integración e interoperabilidad de diferentes servicios OGC descritos o expuestos como servicios web. También el OGC está impulsando diversos experimentos en el marco de los servicios de geo-procesamiento como el experimento Web Process Service Interoperability Experiment (WPS IE), llevado a cabo para testear la viabilidad de las especificaciones de interfaces OWS para permitir servicios de geo-procesamiento a través de Internet.

Por otra parte el Comité Técnico 211 de la ISO (ISO/TC 211) está trabajando en más de 30 estándares (*Ver Anexo I*) para la adquisición, manipulación, administración, análisis, acceso, presentación y transferencia de la información entre diferentes usuarios y sistemas sin que esto dependa de su ubicación geográfica, algunos de los cuales ya se encuentran disponibles. La importancia de la definición de estos estándares radica en obtener mejores resultados, basado en la organización de la información a nivel internacional de manera adecuada, que posibilite la incorporación de aplicaciones, utilización y el almacenamiento de los datos correspondientes a las diferentes temáticas de una región determinada.

Si bien la ISO/TC 211 se orienta hacia estándares abstractos de información geográfica, OGC aporta especificaciones de implementación, lo cual constituye una de las claves de su popularidad. Por otra parte, está integrado por los líderes mundiales de tecnologías de información y de las tecnologías geoespaciales, lo cual asegura su implementación en la industria.[*Delgado 2005*]

El consorcio OpenGIS y la ISO/TC 211 trabajan en estrecha colaboración por lo que se espera que los estándares y especificaciones generados por ambos grupos sean compatibles. De hecho, en varios casos, las especificaciones del OGC se han convertido en estándares ISO. De igual forma, el OGC ha tomado diversas definiciones de la ISO, tal como han sido generadas.[*Convencion 2003*]

3. Tecnologías de información geoespacial sobre la web en Cuba

El tratamiento geográfico de la información en los últimos decenios ha cobrado un auge vertiginoso a escala mundial, cada día con mayores posibilidades de aplicación gracias al desarrollo de los Sistemas de Información Geográfica. Las expectativas creadas sobre estos sistemas también están presentes en Cuba, con sus correspondientes limitaciones y paradojas.

Es así como surge en el 2005, la idea de crear la Infraestructura de Datos Espaciales de la República de Cuba (IDERC), la cual abarca las políticas, tecnologías, estándares y recursos humanos necesarios para la efectiva recolección, administración, acceso, entrega y utilización de los datos espaciales a nivel nacional en función de la toma de decisiones económicas, políticas y sociales.*[Delgado 2004]*

Como punto de acceso nacional a la información geográfica de la IDERC, se crea el Portal Geoespacial de Cuba, con aplicaciones que permiten el acceso a los datos geoespaciales fundamentales y temáticos (muy pobres en estos momentos), acceso a un catálogo para la búsqueda de metadatos, un sistema que nos brinda un Diccionario Geográfico y por último, los Mapas de Población del Anuario Estadístico de Cuba. Actualmente estos sistemas presentan problemas de accesibilidad y la mayor parte del tiempo no se encuentran disponibles.

Dentro de los incipientes resultados de nuestro país en esta esfera, la Universidad de las Ciencias Informáticas no se ha quedado atrás. Teniendo en cuenta que la extensión territorial del centro no ha dejado de crecer desde su creación, el dinamismo que lo caracteriza y que cada vez se hace más complejo el manejo y localización de la información, desde hace unos años se han venido realizando estudios sobre la creación de sistemas que se encarguen de gestionar la información geográfica.

Han sido cuatro los Trabajos de Diploma que sobre este tema se han presentado:

1. *Sistema de Información Geográfica sobre la WEB de la UCI:* Permite mostrar las funcionalidades básicas de un SIG y hacer búsquedas por edificios.
2. *Sistema de gráficos estadísticos y mapas temáticos:* Ofrece la creación de gráficos estadísticos, por ejemplo: un gráfico de barras que muestre las llamadas telefónicas realizadas por mes. Además permite crear mapas temáticos, como son: cantidad de estudiantes suspensos por edificios, por ciento de estudiantes varones o hembras por edificios, entre otros.
3. *Modelación de Servicios Web para el procesamiento y la representación de información geoespacial:* Abarca sólo hasta la modelación de algunos servicios web que brindan funcionalidades muy básicas propias de un SIG.
4. *Modelación del subsistema de control de la información geoespacial en línea:* Abarca hasta la modelación de una herramienta que permitiera el control de la información geoespacial disponible para los diferentes sistemas que necesiten hacer uso de la misma, mediante la utilización de servicios web.

Existe en la Universidad una aplicación web que permite realizar localizaciones y brinda la información que haya disponible sobre el lugar localizado por el usuario, pero este sistema no brinda una interfaz para exportar sus funcionalidades, por lo que los demás sistemas existentes en la universidad y del país en general, no pueden hacer uso de las mismas. Además, si se quisiera adicionarle nuevas funcionalidades, o hacer uso del código existente para incorporarlo a otros sistemas, los desarrolladores necesitarían de una serie de conocimientos mínimos de Geomática, los cuales, para muchos de estos desarrolladores, no son imprescindibles de conocer. Actualmente dicha aplicación no se encuentra disponible.

Se debe señalar que en este curso surgió el polo productivo referente a la Geomática en la facultad 9 y es el encargado en la universidad de todo lo relacionado con los Sistemas de Información Geográfica, en estos momentos está dando sus primeros pasos investigativos.

4. Tecnologías de información geoespacial sobre la web y el software libre

En los años 80, las empresas de programación comenzaron a obligar a sus clientes a firmar acuerdos de licencia por la utilización de los programas que vendían. En contraposición, en 1984 Richard Stallman propuso el concepto de software libre.

La revolución que ha supuesto la filosofía del software libre ha llegado también a las tecnologías de información geoespacial y actualmente existen varios proyectos serios, tal es el caso de: GRASS GIS, Jump, gvSIG, uDIG, Quantum GIS, SAGA, entre otros; que pretenden ser una alternativa viable a programas comerciales como ArcView, Geomedia o IDRISI.

Otro factor importante es la fundación de instituciones encargadas de definir y velar por la aplicación de estándares independientes de los fabricantes en el ámbito de los datos geoespaciales. Tal es el caso de Open Geospatial Consortium (OGC) que engloba más de 250 empresas, instituciones y organizaciones. La estandarización tiene numerosas ventajas, sobre todo para los usuarios de los estándares, ya que implica que no es necesario depender de un único fabricante y permite migrar de un fabricante a otro sin grandes problemas, o intercambiar información con usuarios de otras plataformas. Actualmente los grandes fabricantes se están viendo forzados a adaptar sus programas a estos estándares ya que, sobre todo, por parte de instituciones públicas, se está empezando a exigir que los programas se ciñan a estándares públicos y abiertos. Sin embargo, ha sido en los proyectos de software libre donde estos estándares han tenido más repercusión y dichos proyectos se han basado desde sus inicios en dichos estándares, por lo que podría decirse que llevan esa ventaja frente a las alternativas comerciales.

Por otro lado estarían los servidores de datos geográficos como MapServer, GeoServer o las Geodatabases¹ como PostGIS y MySQL Spatial.

¹ La Geodatabase es un modelo que permite el almacenamiento físico de la información geográfica, ya sea en archivos dentro de un sistema de ficheros o en una colección de tablas en un Sistema Gestor de Bases de Datos

Estos servidores de mapas libres, al igual que los SIG de código abierto, brindan la posibilidad de reutilizar el código de las funcionalidades más importantes que debe tener un SIG, pero no brindan un mecanismo para que esas funcionalidades queden centralizadas, de forma tal que si se mejora el código de las mismas, o se hace alguna actualización, se modifique la funcionalidad en un sólo lugar; esto entorpecería el desarrollo y evolución de los sistemas que hagan uso de estas funcionalidades.

5. Tecnologías Actuales

5.1 Servicios web

Un servicio web (en inglés Web Service) es una colección de protocolos y estándares que sirven para intercambiar datos y funciones entre aplicaciones independientes de la plataforma o el fabricante.[*OPTIC 2007*]

Realizan funciones que pueden ser de todo tipo: desde simples peticiones hasta complejos procesos de negocio. Una vez publicado un servicio web, otras aplicaciones pueden hacer uso de la funcionalidad que brinda el mismo.

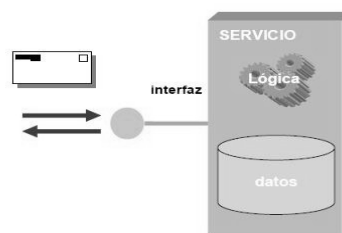


Figura #1.1 Visión interna de los servicios

Entre las ventajas de los servicios web se encuentran:[*CAS 2001*]

- Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.

- Los servicios web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios web pueden aprovecharse de los sistemas de seguridad (firewall) sin necesidad de cambiar las reglas de filtrado.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Permiten la interoperabilidad entre plataformas de distintos fabricantes por medio de protocolos estándares.

Los objetivos principales que persiguen los servicios web son la interoperabilidad y la integración. Esta última permite obtener la información solicitada en tiempo real, agilizando el proceso de toma de decisiones. Para soportar dicha interoperabilidad entre diferentes máquinas en una red los servicios web están basados en tecnologías estándares: XML, WSDL, SOAP o UDDI.

XML: No ha nacido solo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto y hojas de cálculo y casi cualquier cosa imaginable, [OPTIC 2007] es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores, tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

WSDL: Un documento WSDL proporciona la información necesaria al cliente para interactuar con el servicio web. Es extensible y se puede utilizar para describir, prácticamente, cualquier servicio de red, incluyendo SOAP sobre HTTP e incluso protocolos que no se basan en XML como DCOM sobre UDP [Terés 2007]. Dichos documentos definen

los servicios como colecciones de puntos finales de red o puertos.

Entre los parámetros que se establecen en el WSDL se encuentran:

- URL del servicio
- Nombre del servicio
- Nombre de la funcionalidad
- Parámetros que requiere
- Parámetro que devuelve
- Puerto de comunicación

De esta forma un desarrollador de aplicaciones a partir de la descripción anterior puede utilizar un servicio sin necesidad de saber como está implementado internamente.

SOAP: Trabaja con la infraestructura de Internet. No hay que hacer configuraciones especiales en routers, firewall, proxy servers, para que SOAP trabaje. Usando HTTP se vuelve muy fácil de desarrollar.

Varios proveedores y organizaciones han soportado la implementación de SOAP. El consorcio W3C² ha definido a SOAP como una especificación, y ninguna organización o un simple proveedor tendrán el control sobre SOAP, por lo tanto se ha convertido en otro estándar de Internet, tales como HTML, XML, HTTP y SMTP.

Ha recibido un increíble apoyo por parte de la industria, pues es el primer protocolo de su tipo que ha sido aceptado prácticamente por todas las grandes compañías de software del mundo. Compañías que en raras ocasiones cooperan entre sí están ofreciendo su apoyo a

² El [Consortio World Wide Web](#) (W3C) es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web .

este protocolo. Algunas de las mayores compañías que soportan SOAP son Microsoft, IBM, SUN, Microsystems, SAP y Ariba.

En el núcleo de los servicios web se encuentra SOAP, que proporciona un mecanismo estándar de empaquetar mensajes. SOAP ha recibido gran atención debido a que facilita una comunicación estilo RPC entre un cliente y un servidor remoto.[*Terés 2007*]

UDDI: A través de este estándar, se puede publicar y descubrir información de una empresa y de sus servicios. Lo más importante es que una UDDI contiene información sobre las interfaces técnicas de los servicios de una empresa. A través de un conjunto de llamadas basadas en SOAP, se puede interactuar con una UDDI tanto en tiempo de diseño como de ejecución para descubrir datos técnicos de los servicios que permitan invocarlos y utilizarlos.

Su principal objetivo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL, en los que se describen los requisitos del protocolo y los formatos del mensaje solicitado para interactuar con los servicios web del catálogo de registros.[*E A 2006*]

Es por ello, que esta tecnología nos brinda la posibilidad de ofrecer interoperabilidad y estandarización de las funcionalidades que se implementarán en este trabajo. Además, permite tener el código de las funcionalidades básicas centralizado, de forma que se hace muy cómodo el mantenimiento del mismo. Los usuarios de los servicios web pueden hacer uso de los mismos sin importar el lenguaje de programación y la plataforma en el cual estén desarrollando.

5.2 Arquitectura orientada a servicios

La Arquitectura Orientada a Servicios (SOA), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos de software del usuario.[*Barron 2007*]

La mayoría de las definiciones de SOA identifican la utilización de servicios web

empleando SOAP y WSDL en su implementación, no obstante se puede implementar una SOA utilizando cualquier tecnología basada en servicios.

Los servicios web se han convertido en el estandarte de SOA, ya que esta tecnología posee un conjunto de características que permiten cubrir todos los principios de la orientación a servicios.[Barco 2006] La definición de la interfaz encapsula las particularidades de una implementación, lo que la hace independiente del fabricante, del lenguaje de programación o de la tecnología de desarrollo. Con esta arquitectura, se pretende que los componentes software desarrollados sean muy reusables, ya que la interfaz se define siguiendo un estándar.

En la figura 1.2 que se muestra a continuación, se presenta la estructura básica de funcionamiento de una arquitectura SOA tradicional.

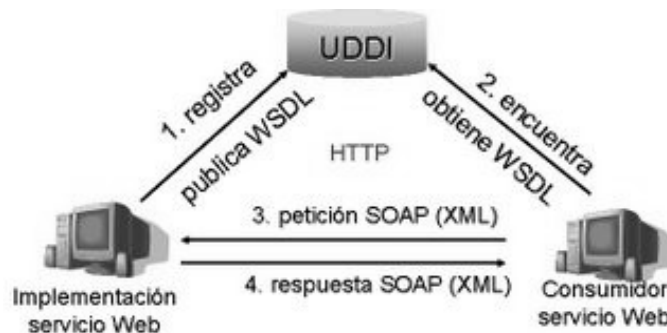


Figura # 1.2 Arquitectura SOA

Se puede observar la existencia de tres roles claramente diferenciados:

- Cliente del servicio: Es el que solicita la ejecución del servicio web y por lo tanto el que lo consume.
- Proveedor del servicio: Es el encargado de implementar el servicio web y ofrecerlo a los clientes.

- Registro del servicio: Es un repositorio donde se almacenan las descripciones de los servicios, para que así los clientes puedan buscar el servicio web que mejor se adapte a sus necesidades.

Desde el punto de vista práctico de la utilización de una arquitectura orientada a servicios en entornos geoespaciales, la principal ventaja radica en la evidente independencia de los sistemas a utilizar los servicios expuestos, sin que esto conlleve a analizar especificidades internas de las tecnologías de manipulación de mapas existentes o utilizadas en el desarrollo de la interfaz, permitiendo incorporar un valor añadido a las aplicaciones que utilicen estos servicios.

6. Metodología a utilizar

Una metodología para el desarrollo de un proceso de software nos brinda mayor organización a la hora de desarrollar un sistema y contribuye a que se obtenga un producto eficiente. Es por esto que se hace necesaria la selección de una metodología que guíe el proceso de desarrollo del sistema.

6.1 Proceso Unificado del Rational (RUP)

RUP presenta una forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo), pretende implementar las mejores prácticas en Ingeniería de Software, posee una administración de requisitos, tiene en cuenta el control de cambios y la verificación de la calidad del software, es explícita en la definición de artefactos (son los productos tangibles del proceso) y su trazabilidad, contempla la relación causal de los artefactos creados desde los requerimientos hasta la implementación y las pruebas, facilitando el seguimiento y control del proyecto, incluye roles (papel que desempeña una persona en un determinado momento), se caracteriza por ser iterativo e incremental, está centrado en la arquitectura y guiado por los casos de uso. Estas son características que la distinguen de

otras metodologías de desarrollo.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.[*Mendoza 2004*]

Debido a las facilidades que aporta, se escoge RUP como la metodología de desarrollo para llevar a cabo este proyecto.

7. Herramientas a utilizar

La creciente popularidad que las aplicaciones basadas en SIG han cobrado en los últimos años, ha propiciado el desarrollo de un enorme número de herramientas de software para el manejo de información espacial de código abierto para cubrir prácticamente cualquier necesidad.

7.1 Herramienta CASE: Visual Paradigm-UML

La herramienta CASE seleccionada fue el Visual Paradigm por sus características: ofrece soporte multiplataforma, brinda un entorno agradable de creación de diagramas UML, fácil de usar, proporciona excelentes facilidades de interoperabilidad con otras aplicaciones, fue creado para el ciclo vital completo del desarrollo del software que lo automatiza y acelera, permitiendo la captura de requisitos, análisis, diseño e implementación.

Brinda capacidades, tales como: generación del código, ingeniería inversa y generación de informes; tiene la capacidad de crear el esquema de clases a partir de una base de datos y viceversa, está diseñado para usuarios interesados en sistemas de software de gran escala, teniendo en cuenta el paradigma de programación orientada a objetos, además apoya los estándares más recientes de las notaciones de Java y UML, incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y

vean en tiempo real los cambios hechos por sus compañeros, es un producto de calidad que soporta aplicaciones web, ofrece la disponibilidad de múltiples versiones para cada necesidad y la de integrarse con los principales entornos de desarrollo. Por último, es muy fácil de instalar y de utilizar.

7.2 Servidor de mapa: MapServer

MapServer, fue desarrollado originalmente por la Universidad de Minnesota (UMN). Actualmente es organizado por el proyecto TerraSip, financiado por la NASA, entre la UMN y el consorcio de intereses de la administración del suelo. *[Alvarez, Cisneros 2006]*

Como servidor de mapas ha demostrado a partir de estadísticas y estudios realizados, que el mismo ofrece mejores rendimientos y prestaciones, con respecto a otros servidores de mapas Open Source.

Provee un conjunto de librerías que permiten a los desarrolladores la manipulación de la cartografía. Obtiene un rendimiento significativamente superior cuando se utiliza en combinación con PostGIS/PostgreSQL. *[Herrera, Lopez, Barreras 2006]*

Tiene dentro de sus características principales que puede ejecutarse en plataformas Linux/Apache y Windows 95 o superior/Apache/IIS y es considerado como el producto de software libre más maduro y popular para el desarrollo de aplicaciones web de información georreferenciada. *[Alvarez, Cisneros 2006]*

En nuestro caso específico se utilizará MapServer como servidor de mapas, solamente para implementar dos servicios, ya que el mismo facilita la generación de imágenes a partir de un mapa, y la obtención de información asociada a una coordenada geográfica.

Este servidor de mapas es el más utilizado a nivel mundial, posee un alto rendimiento y

tiene una amplia comunidad de desarrollo mejorando sus funcionalidades y añadiendo nuevas continuamente, por lo que se brindarán además, todas las funcionalidades de este servidor para los usuarios que estén capacitados para utilizarlas.

7.3 Lenguaje de programación: PHP

PHP (acrónimo de PHP: Hypertext Preprocessor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.[VAN DER HENST 2001]

Una de sus características más potentes es su soporte para un gran número de servidores de bases de datos, tales como: InterBase, MySQL, Oracle, ODBC, DB2, Informix, SQLite, PostgreSQL; lo cual permite la creación de aplicaciones web robustas, ofrece la integración con varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa, desde generar documentos en pdf hasta analizar código XML, utiliza su propio sistema de administración de recursos, provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo “.ini”, brinda una amplia documentación en el sitio oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda, permite las técnicas de Programación Orientada a Objetos así como crear los formularios para la web, no requiere definición de tipos de variables ni manejo detallado del bajo nivel, se pueden hacer grandes cosas con pocas líneas de código, lo que hace que merezca la pena utilizarlo, no soporta directamente punteros, como el caso de C/C++, de forma que no existen problemas de depuración provocados por los mismos.

Otra característica a tener en cuenta es la conectividad. PHP dispone de una amplia gama de librerías, y agregarle extensiones es muy fácil. Esto le permite ser utilizado en muchas áreas diferentes, tales como encriptado, gráficos, XML y otras.[Jimenez 2006]

Además de todas las ventajas expuestas con anterioridad que brinda PHP, hay dos razones fundamentales por la que se escogió para implementar los servicios web:

- Es un lenguaje multiplataforma.
- Es software libre, por lo que se presenta como una alternativa de fácil acceso para todos.

7.4 Sistema Gestor de Bases de Datos: PostgreSQL

PostgreSQL es un proyecto de código abierto que está ampliamente considerado como el sistema gestor de bases de datos de código abierto más avanzado del mundo.[Worsley, Drake 2001]

Sus ventajas respecto a otros sistemas de bases de datos son destacables:

- Es altamente extensible, puesto que no hay costo asociado a la licencia del software, el código fuente está disponible para todos.
- Es multiplataforma, está disponible en casi cualquier Unix (34 plataformas en la última versión estable).
- Diseñado para ambientes de alto volumen, pues usa una estrategia de almacenamiento llamada MVCC para conseguir una mejor respuesta.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, sobre redes (MAC, IP), cadenas de bits, etc. También permite la creación de tipos propios.
- Ofrece herramientas gráficas de alta calidad para administrar las Bases de Datos (ej. pgAdmin y Data Architect) y para hacer diseños de las mismas (Tora, Data Architect)
- Posee una gran escalabilidad: es capaz de ajustarse al número de procesadores y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta, o sea, su arquitectura puede ser continuamente ampliada de acuerdo con la demanda de los

usuarios.

- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos.

A partir de los requerimientos originados por nuestro problema en cuestión y los resultados alcanzados por el gestor de bases de datos PostgreSQL, se decide optar por el mismo para el almacenamiento de la información geográfica, para esto es necesario incorporar al gestor de bases de datos soporte para estructura de datos espaciales, utilizando para esto PostGIS.

PostGIS es una extensión al sistema gestor de bases de datos objeto-relacional PostgreSQL para su utilización en SIG. Está publicado bajo licencia GNU y puede ser usado en las plataformas Windows, GNU/Linux y Mac OS X. Actualmente existen numerosos productos informáticos que pueden utilizarlo como apoyo para su base de datos, entre ellos está MapServer a quien sirve de fuente de datos. Soluciona los problemas de flexibilidad que surgen cuando se utilizan otros formatos de almacenamiento de información digital. Define nuevos tipos de datos, crea dos tablas con información relevante al sistema (proyección de los datos y columna que posee la información geográfica) y define también las funciones de manejo de información como procedimientos almacenados.

Con PostGIS podemos usar todos los objetos que aparecen en la especificación OpenGIS como puntos, líneas, polígonos, multilíneas, multipuntos, y colecciones geométricas.[*Martín 2006*]

La integración de PostGIS y PostgreSQL incorpora a la base de datos la posibilidad de la definición de tipos de datos espaciales y además permite vincular fácilmente la información asociada a estos tipos de datos con los objetos geográficos. Esta solución es adecuada para el trabajo que se propone, cumpliendo con las reglamentaciones del Ministerio de Informática y las Comunicaciones acerca de la política de utilización del Software Libre.

8. Conclusiones

En el presente capítulo se ha tratado de forma general lo relacionado con el estado del arte a nivel internacional y nacional del objeto de estudio propuesto, arribando a las siguientes conclusiones:

1. Los servidores y librerías para el manejo de mapas existentes no dan solución al problema planteado.
2. Los SIG actuales, no brindan la posibilidad de acceder a funcionalidades de forma remota, sin tener conocimientos mínimos de Geomática.
3. Los SIG de código abierto estudiados, no permiten la reutilización de código sin hacer engorrosa la actualización y el mantenimiento del software que haga uso de sus funcionalidades.
4. Se hace necesario desarrollar un conjunto de servicios web, que solucionen el problema planteado, haciendo uso de las funcionalidades que ofrecen algunos de los sistemas y servidores de mapas analizados en el presente capítulo.
5. Se realizó un análisis de las tecnologías que serán utilizadas a lo largo del desarrollo del sistema propuesto.
6. Se fundamentaron las elecciones del lenguaje de programación, el servidor de mapas, el Sistema Gestor de Bases de Datos, la herramienta CASE y la metodología que se tendrá en cuenta para la elaboración de los servicios que se proponen.
7. Una vez conocidas las herramientas óptimas y los conceptos claves para el desarrollo y comprensión del sistema que se presenta se comenzará a desarrollar la propuesta de solución.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

1. Introducción

En el presente capítulo se describen las características que presenta el sistema, para esto se estudiarán los procesos del negocio relacionados con el objeto de estudio, a partir del análisis de éstos se puede percibir que debido a la falta de información existente y a la poca estructuración de dichos procesos, se necesita definir conceptos que se pueden agrupar en un Modelo de dominio para que sea más entendible el contexto en el que se ubica el sistema y poder construirlo eficientemente.

Además se enumeran los requerimientos funcionales y no funcionales de la plataforma de servicios web y se logra identificar mediante un Diagrama de casos de uso y las descripciones detalladas de cada uno, las relaciones de los actores con el sistema y las secuencias de acciones con las que interactúan.

2. Propuesta de sistema

Como se ha mostrado anteriormente los procesos de gestión y procesamiento de la información geoespacial para su representación constituye el objeto de estudio de este trabajo de diploma. Dentro de este objeto como campo de acción se encuentran las tecnologías de información geoespacial sobre la web.

Este trabajo se concentra en una propuesta de sistema basado en servicios web, que sirva como herramienta de apoyo en el proceso de toma de decisiones. Esta serie de funcionalidades que se estarán implementando responde a la necesidad de encontrar una fórmula que permita a diferentes sistemas informáticos acceder, sin importar la plataforma en la que han sido desarrollados, a la información geográfica disponible utilizando la web. Permitiendo el desarrollo de aplicaciones para el manejo de este tipo de información de forma

centralizada y que logre erradicar el duplicado innecesario de dicha información en la Universidad de las Ciencias Informáticas.

La plataforma manejará principalmente el trabajo con mapas geográficos, mediante la creación, modificación, obtención, eliminación, entre otras operaciones de los mismos y de las diferentes capas y objetos que lo conforman.

3. Modelo de dominio

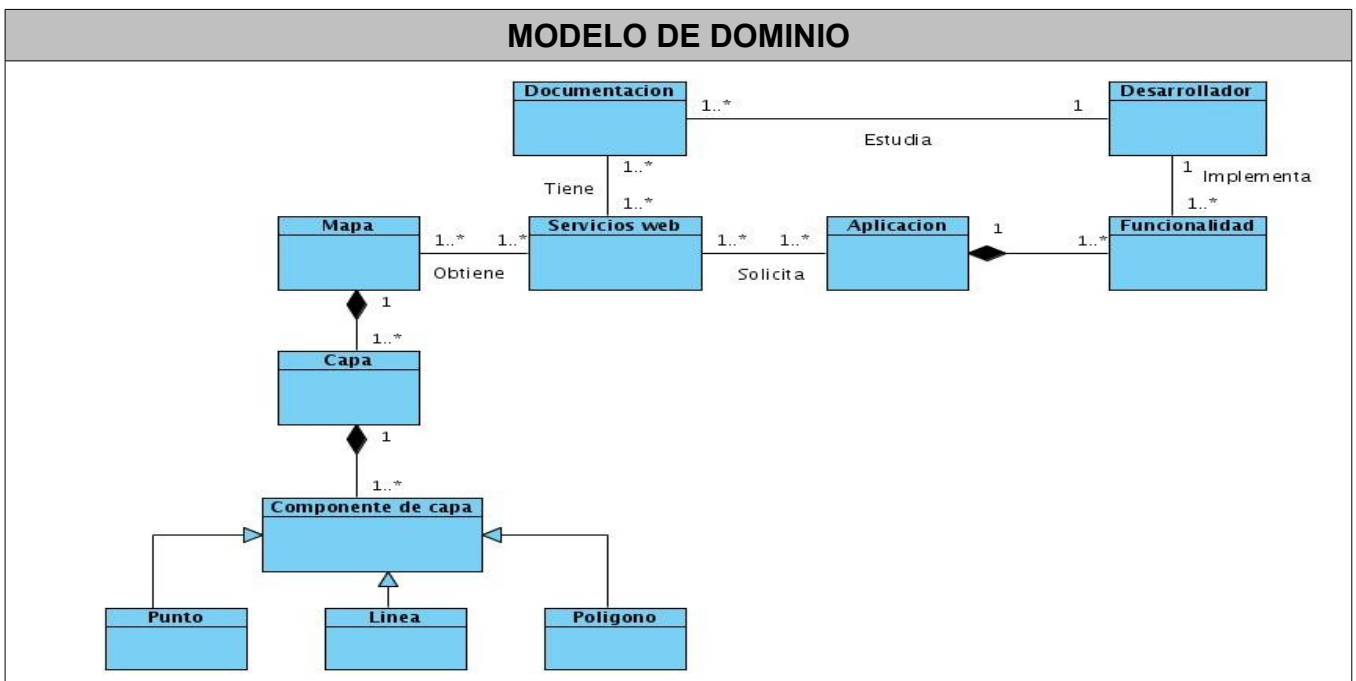
Este modelo nos permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto ayuda a dichos usuarios, clientes, desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se desarrolla. Además va a contribuir posteriormente a identificar algunas clases que se utilizarán en dicho sistema. Primeramente vamos a definir los principales conceptos que se utilizarán en el diagrama.

3.1 Definición de las entidades y conceptos fundamentales

Concepto	Descripción
Mapa	Es un conjunto de capas que representan de forma gráfica y métrica una porción de territorio sobre una superficie bidimensional. [Palacín 2007]
Capa	Entidades gráficas organizadas en planos de visualización que contienen la información de los puntos en la pantalla (o píxeles) para su representación. Conjunto de elementos de una misma geometría.
Componente de capa	Un objeto que puede comportarse como un punto, una línea o un polígono.
Punto	Abstracción de un objeto de cero dimensiones representado por un par de coordenadas X,Y. [Escobar, Hunter, Bishop, Zerger 1999] Representa una entidad geográfica demasiado pequeña para ser representada como una línea o como una superficie y que tiene localizaciones concretas.
Línea	Es una sucesión continua de puntos. Representan objetos que son largos

	pero muy angostos para dibujarlos como polígonos.[<i>EduTEKA 2005</i>]
Polígono	Entidad utilizada para representar superficies. Se define por las líneas que forman su contorno y por un punto interno que lo identifica. Los polígonos tienen atributos que describen al elemento geográfico que representan.
Servicios web	Colección de protocolos y estándares que sirven para intercambiar datos y funciones entre aplicaciones independientes de la plataforma o el fabricante.[<i>OPTIC 2007</i>]
Aplicación	Es cualquier sistema que se encarga de gestionar información geográfica.
Funcionalidad	Acciones que es capaz de realizar cualquier aplicación para gestionar información geográfica.
Desarrollador	Profesional que forma parte de un equipo para el desarrollo y mantenimiento de Aplicaciones de Información Geográfica.
Documentación	Toda la documentación (WSDL, manuales, tutoriales, entre otros) que brinda información sobre el uso de los servicios web.

3.2 Representación del Modelo de dominio



4. Especificación de los requisitos de software

La correcta especificación de requisitos determina en gran medida el éxito de un proyecto de desarrollo de software. A continuación se muestra la especificación de los requisitos tanto funcionales como no funcionales de la plataforma de servicios web que se propone.

4.1 Requerimientos funcionales

Los requerimientos funcionales especifican acciones que el sistema debe ser capaz de realizar, capacidades o condiciones que debe cumplir. En nuestro caso se debe:

RF1: Gestionar mapa.

- 1.1. Crear un mapa nuevo.
- 1.2. Obtener lista de mapas disponibles.
- 1.3. Obtener un mapa existente.
- 1.4. Modificar un mapa existente.
- 1.5. Adicionar capas a un mapa existente.
- 1.6. Eliminar un mapa existente.

RF2: Gestionar capa.

- 2.1. Crear una capa nueva.
- 2.2. Obtener lista de capas disponibles.
- 2.3. Obtener una capa existente.
- 2.4. Modificar una capa existente.
- 2.5. Adicionar objetos a una capa existente.
- 2.6. Eliminar una capa existente.

RF3: Gestionar objeto.

- 3.1. Crear un objeto nuevo.
- 3.2. Modificar un objeto existente.
- 3.3. Obtener un objeto dado un punto expresado en coordenadas geográficas.

3.4. Obtener un objeto dado un punto expresado en píxel.

3.5. Listar objetos de una capa.

3.6. Eliminar un objeto existente.

3.7. Obtener la distancia entre dos puntos.

3.8. Obtener información de un punto.

4.2 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en dichas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Se consideran los atributos del sistema.

Usabilidad:

- Los servicios web podrán ser utilizados por personas que tengan un conocimiento básico en el manejo de las computadoras y de programación.

Portabilidad:

- Los servicios web son multiplataforma.

Confiabilidad:

- Debe mantenerse la consistencia de los datos en correspondencia con la realidad.

Software:

Como el conjunto de servicios web es multiplataforma se requiere en el servidor web:

- Tener instalado cualquier sistema operativo.
- Servidor Web Apache2.x.

Para el servidor de bases de datos, se requiere:

- Tener instalado cualquier sistema operativo, PostgreSQL 8.1.

En la computadora de los usuarios se requiere:

- Tener instalado cualquier sistema operativo.
- Navegador web.

Hardware:

Requerimientos mínimos, la plataforma puede estar instalada en una computadora:

- 256 Mb de RAM, 15 GB de disco duro.

Para el servidor de bases de datos, se requiere como mínimo:

- 64 MB de RAM, 100 MB de disco duro.

Legales :

- La plataforma de servicios web debe ajustarse y regirse por los Lineamientos de Arquitectura establecidos en la Universidad de las Ciencias Informáticas.

5. Modelo del sistema

5.1 Definición de los actores

Actores	Justificación
Sistema Externo	Es la aplicación o sistema que se conecta para consumir los servicios web.

5.2 Definición de los casos de uso

5.2.1 Descripción de los casos de uso

CU1	Gestionar mapa
Actor	Sistema Externo
Descripción	El caso de uso comienza cuando el actor desea obtener la lista de mapas disponibles, crear uno nuevo, modificarlo, obtener un mapa, adicionarle capas y/o eliminar uno ya existente.
Referencia	RF 1.1, RF 1.2, RF 1.3, RF 1.4, RF 1.5, RF 1.6

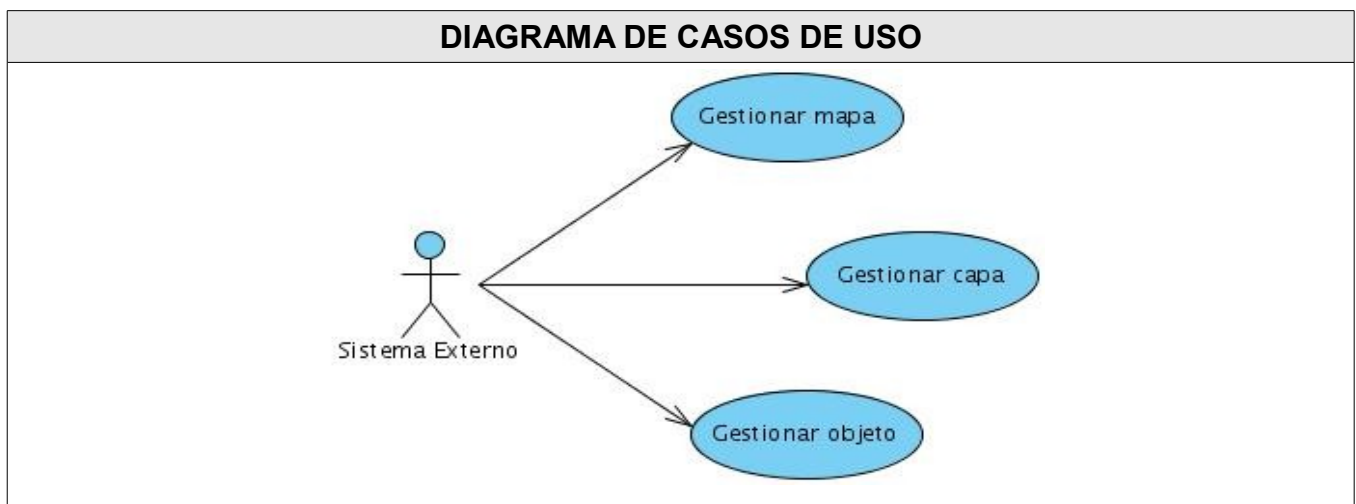
CU2	Gestionar capa
Actor	Sistema Externo
Descripción	El caso de uso comienza cuando el actor desea obtener la lista de capas disponibles, crear una capa nueva, modificarla, obtener una capa, adicionarle objetos y/o eliminar una ya existente.

Referencia	RF 2.1, RF 2.2, RF 2.3, RF 2.4, RF 2.5, RF 2.6
-------------------	--

CU3	Gestionar objeto
Actor	Sistema Externo
Descripción	El caso de uso comienza cuando el actor desea crear un nuevo objeto, modificar o eliminar uno ya existente, conocer la distancia entre dos puntos seleccionados, pedir información sobre un punto en el mapa, listar objetos de una capa u obtenerlo dado un punto expresado en coordenadas geográficas o en píxel.
Referencia	RF 3.1, RF 3.2, RF 3.3, RF 3.4, RF 3.5, RF 3.6, RF 3.7, RF 3.8

5.2.2 Diagrama de casos de uso del sistema

El Diagrama de casos de uso del sistema describe las interacciones del mismo con su entorno, identificando los actores, que representan los diferentes roles desempeñados por los usuarios del sistema, y los casos de uso, que corresponden a la funcionalidad que el sistema ofrece a sus usuarios, expresada en los requisitos funcionales.



5.2.3 Descripción detallada de los casos de uso

Caso de uso	Gestionar mapa	
Actores	Sistema Externo (inicia)	
Propósito	Ofrecer los servicios web destinados a gestionar mapa.	
Resumen	Este caso de uso comienza cuando el actor desea obtener la lista de mapas disponibles, crear uno nuevo, modificarlo, adicionarle capas, obtener y/o eliminar uno ya existente. Termina cuando se actualiza la información en la base de datos o en el fichero de configuración del mapa correspondiente y envía el resultado al actor.	
Precondiciones	Se realice la solicitud de algún servicio.	
Poscondiciones	<p>La sección Crear mapa al terminar actualiza la información en la base de datos y crea el fichero de configuración del mapa correspondiente.</p> <p>Las secciones Modificar mapa, Adicionar capa y Eliminar mapa al terminar actualizan la información en la base de datos y en el fichero de configuración del mapa correspondiente.</p> <p>Las secciones Listar mapas disponibles y Obtener mapa muestran el resultado al actor y no actualizan la base de datos ni el fichero de configuración del mapa correspondiente.</p>	
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Referencias	RF 1.1, RF 1.2, RF 1.3, RF 1.4, RF 1.5, RF 1.6	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
<p>1. El actor realiza las siguientes operaciones:</p> <ul style="list-style-type: none"> • Crear mapa • Modificar mapa • Obtener lista de mapas 		

<p>disponibles</p> <ul style="list-style-type: none"> • Obtener mapa • Adicionar capa • Eliminar mapa 	<p>2. El sistema realiza las operaciones según haya sido el envío:</p> <ul style="list-style-type: none"> • Crear mapa, ver sección “Crear mapa” • Modificar mapa, ver sección “Modificar mapa” • Obtener lista de mapas disponibles, ver sección “Listar mapas disponibles” • Obtener mapa, ver sección “Obtener mapa” • Adicionar capa, ver sección ”Adicionar capa” • Eliminar mapa, ver sección “Eliminar mapa”
<p>Sección: Crear mapa</p>	
<p>1. El actor realiza la operación crear mapa, pasándole el nombre, el extent (dos coordenadas que especifican que parte del mapa se va a mostrar), el formato de salida, la leyenda, la forma en que se verá el mapa después de una consulta, la referencia, la escala, el ancho y el alto de la imagen que va a generar el servicio obtener mapa, la especificación del directorio donde se guardarán las imágenes generadas por el servidor de mapas y el alias en</p>	

<p>el servidor web para acceder a las mismas.</p> <p>6. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe este mapa y si no existe crea el nuevo mapa con los datos que le pasaron.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema crea un fichero texto con la especificación del mapa en el formato que requiere MapServer.</p> <p>5. El sistema envía el resultado de la operación realizada.</p>
<p>Sección: Modificar mapa</p>	
<p>1. El actor realiza la operación modificar mapa, pasándole el nombre del mismo y al menos un valor que desee modificar.</p> <p>6. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicho mapa y si existe cambia los datos a modificar.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema actualiza el fichero de configuración correspondiente al mapa.</p> <p>5. El sistema envía el resultado de la operación realizada.</p>
<p>Sección: Listar mapas disponibles</p>	
<p>1. El actor realiza la operación listar mapas disponibles.</p>	<p>2. El sistema realiza la operación de verificar todos los mapas disponibles, si existen muestra los</p>

<p>4. El actor recibe el resultado de la operación realizada.</p>	<p>nombres de los mismos. 3. El sistema envía el resultado de la operación realizada.</p>
Sección: Obtener mapa	
<p>1. El actor realiza la operación obtener mapa, pasándole el nombre del mapa. 4. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicho mapa, si existe devuelve la url de la imagen de dicho mapa. 3. El sistema envía el resultado de la operación realizada.</p>
Sección: Adicionar capa	
<p>1. El actor realiza la operación adicionar capa, pasándole el nombre del mapa y el nombre de la capa que desea adicionar. 6. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicho mapa, si existe adiciona la capa. 3. El sistema actualiza la base de datos. 4. El sistema actualiza el fichero de configuración correspondiente al mapa. 5. El sistema envía el resultado de la operación realizada.</p>
Sección: Eliminar mapa	
<p>1. El actor realiza la operación eliminar mapa, pasándole el nombre del que desea eliminar.</p>	<p>2. El sistema realiza la operación de verificar si</p>

<p>6. El actor recibe el resultado de la operación realizada.</p>	<p>existe dicho mapa, sí existe lo elimina.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema elimina el fichero de configuración correspondiente al mapa.</p> <p>5. El sistema envía el resultado de la operación realizada.</p>
<p>Cursos alternos</p>	
<p>Si ocurre un error en el proceso de crear, modificar, obtener listado de mapas disponibles, obtener mapa, adicionar capas o eliminar un mapa existente se envía el mensaje de error correspondiente.</p>	

Caso de uso	Gestionar capa
Actores	Sistema Externo (inicia)
Propósito	Ofrecer los servicios web destinados a gestionar capa.
Resumen	Este caso de uso comienza cuando el actor desea obtener la lista de capas disponibles, crear una nueva, modificarla, obtener una capa, adicionarle objetos y/o eliminar una ya existente. Termina cuando se actualiza la información en la base de datos o en el fichero de configuración del mapa correspondiente y envía el resultado al actor.
Precondiciones	Se realice la solicitud de algún servicio.
Poscondiciones	<p>Las secciones Crear capa y Adicionar objeto al terminar actualizan la información en la base de datos.</p> <p>Las secciones Modificar capa y Eliminar capa al terminar actualizan la información en la base de datos y en el fichero de configuración de los mapas correspondientes.</p> <p>Las secciones Listar capas disponibles y Obtener capa muestran el resultado al actor y no actualizan la base de datos ni el fichero</p>

	de configuración de los mapas que tengan la capa en cuestión.	
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Referencias	RF 2.1, RF 2.2, RF 2.3, RF 2.4, RF 2.5, RF 2.6	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
<p>1. El actor realiza las siguientes operaciones:</p> <ul style="list-style-type: none"> • Crear capa • Modificar capa • Obtener lista de capas disponibles • Obtener capa • Adicionar objeto • Eliminar capa 	<p>2. El sistema realiza las operaciones según haya sido el envío:</p> <ul style="list-style-type: none"> • Crear capa, ver sección “Crear capa” • Modificar capa, ver sección “Modificar capa” • Obtener lista de capas, ver sección “Listar capas disponibles” • Obtener capa, ver sección “Obtener capa” • Adicionar objetos, ver sección “Adicionar objeto” • Eliminar capa, ver sección “Eliminar capa” 	
Sección: Crear capa		
<p>1. El actor realiza la operación crear capa, pasándole el nombre, el tipo de todos los objetos que conforman la capa, classitem (el</p>		

<p>parámetro por el cuál se mostrará información adicional de los objetos) y class (define la forma en que se mostrará la capa).</p> <p>5. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe esta capa y sí no existe, crea la nueva capa con los datos que le pasaron.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
Sección: Modificar capa	
<p>1. El actor realiza la operación modificar capa, pasándole el nombre de la capa y al menos un valor que desee modificar.</p> <p>6. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema verifica que existe la capa y sí existe cambia los datos a modificar.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema actualiza el fichero de configuración de todos los mapas que incluyan la capa que se está modificando.</p> <p>5. El sistema envía el resultado de la operación realizada.</p>
Sección: Listar capas disponibles	
<p>1. El actor realiza la operación listar capas disponibles.</p>	<p>2. El sistema realiza la operación de verificar todas las capas disponibles, sí existen las muestra.</p>

<p>4. El actor recibe el resultado de la operación realizada.</p>	<p>3. El sistema envía el resultado de la operación realizada.</p>
<p>Sección: Obtener capa</p>	
<p>1. El actor realiza la operación obtener capa, pasándole el nombre de la capa.</p> <p>4. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicha capa, sí existe devuelve la url de la imagen que representa a la misma.</p> <p>3. El sistema envía el resultado de la operación realizada.</p>
<p>Sección: Adicionar Objeto</p>	
<p>1. El actor realiza la operación adicionar objeto, pasándole el nombre de la capa y la geometría del objeto en formato WKT.</p> <p>5. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe la capa, sí existe verifica que exista el objeto, sino existe el objeto lo adiciona.</p> <p>3. El sistema añade el objeto a la capa.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
<p>Sección: Eliminar capa</p>	
<p>1. El actor realiza la operación eliminar capa, pasándole el nombre de la capa que desea eliminar.</p>	<p>2. El sistema realiza la operación de verificar si existe dicha capa, sí existe la elimina.</p> <p>3. El sistema actualiza los mapas que utilizan la capa</p>

<p>5. El actor recibe el resultado de la operación realizada.</p>	<p>eliminada, en la base de datos y en el fichero de configuración de los mismos.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
<p>Cursos alternos</p> <p>Si ocurre un error en el proceso de crear, modificar, obtener listado de capas disponibles, obtener capa, adicionar objetos o eliminar una ya existente se envía el mensaje de error correspondiente.</p>	

Caso de uso	Gestionar objeto
Actores	Sistema Externo (inicia)
Propósito	Ofrecer los servicios web destinados a gestionar objeto.
Resumen	El caso de uso comienza cuando el actor desea crear un nuevo objeto, modificar o eliminar uno ya existente, conocer la distancia entre dos puntos seleccionados, pedir información sobre un punto en el mapa, listar objetos de una capa u obtenerlo dado un punto expresado en coordenadas geográficas o en píxel. Termina cuando se actualiza la información en la base de datos y envía el resultado al actor.
Precondiciones	Se realice la solicitud de algún servicio.
Poscondiciones	Las secciones Crear objeto, Modificar objeto y Eliminar objeto al terminar actualizan la información en la base de datos. Las secciones Obtener objeto dado píxel, Obtener objeto dado coordenadas geográficas, Obtener distancia, Obtener información y Listar objetos muestran el resultado al usuario y no se actualiza en la base de datos.
Tipo	Real y expandido
Responsabilidades	

Cu relacionados	
Referencias	RF 3.1, RF 3.2, RF 3.3, RF 3.4, RF 3.5, RF 3.6, RF 3.7, RF 3.8
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema
<p>1. El actor realiza las siguientes operaciones:</p> <ul style="list-style-type: none"> • Crear objeto • Modificar objeto • Obtener objeto dado píxel • Obtener objeto dado coordenadas geográficas • Obtener distancia entre dos puntos • Obtener información de un punto • Listar objetos por capa • Eliminar objeto 	<p>2. El sistema realiza las operaciones según haya sido el envío:</p> <ul style="list-style-type: none"> • Crear objeto, ver sección “Crear objeto” • Modificar objeto, ver sección “Modificar objeto” • Obtener objeto dado píxel, ver sección “Obtener objeto dado píxel” • Obtener objeto dado coordenadas geográficas, ver sección “Obtener objeto dado coordenadas ” • Obtener distancia entre dos puntos, ver sección “Obtener distancia” • Obtener información de un punto, ver

	<p>sección “Obtener información”</p> <ul style="list-style-type: none"> • Listar objetos por capas, ver sección “Listar objetos” • Eliminar objeto, ver sección “Eliminar objeto”
Sección: Crear objeto	
<p>1. El actor realiza la operación crear objeto, pasándole la geometría en formato WKT, el nombre y la descripción del mismo.</p> <p>4. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe este objeto y sino existe:</p> <ul style="list-style-type: none"> • El sistema crea un nuevo objeto. • El sistema actualiza la base de datos. <p>3. El sistema envía el resultado de la operación realizada.</p>
Sección: Modificar objeto	
<p>1. El actor realiza la operación modificar objeto, pasándole el identificador (la geometría del mismo en formato WKT) y al menos un valor que desee modificar.</p> <p>5. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema verifica que existe ese objeto y si existe cambia los datos a modificar.</p> <p>3. El sistema actualiza la base de datos.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
Sección: Obtener objeto dado píxel	
<p>1. El actor realiza la operación obtener objeto dado píxel, pasándole las</p>	

<p>coordenadas del punto especificada en píxel y el nombre del mapa.</p> <p>4. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicho objeto, si existe obtiene la geometría que representa al objeto.</p> <p>3. El sistema envía el resultado de la operación realizada.</p>
Sección: Obtener objeto dado coordenadas	
<p>1. El actor realiza la operación obtener objeto, pasándole las coordenadas geográficas de un punto.</p> <p>4. El actor recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de verificar si existe dicho objeto, si existe obtiene la geometría que representa al mismo.</p> <p>3. El sistema envía el resultado de la operación realizada.</p>
Sección: Obtener distancia	
<p>1. El actor realiza la operación obtener la distancia entre dos puntos especificados por él.</p> <p>5. El actor recibe el resultado de la operación.</p>	<p>2. El sistema verifica que los puntos sean válidos.</p> <p>3. El sistema realiza la operación de calcular la distancia entre ellos.</p> <p>4. El sistema devuelve el resultado de la operación realizada.</p>
Sección: Obtener información	
<p>1. El actor realiza la operación obtener la información de un punto seleccionado</p>	

<p>por él, pasándole el nombre del mapa, las coordenadas del punto en píxel y un valor booleano que especifica el nivel de detalle.</p> <p>6. El actor recibe el resultado de la operación.</p>	<ol style="list-style-type: none"> 2. El sistema verifica que sea un punto válido. 3. El sistema verifica si el mapa existe o no. 4. El sistema verifica que el punto pertenezca al mapa. 5. El sistema comprueba el nivel de detalle especificado, si es true devuelve toda la información asociada al punto. En caso de ser false sólo devuelve el nombre y la descripción.
Sección: Listar objetos	
<ol style="list-style-type: none"> 1. El actor realiza la operación listar objetos, pasándole el nombre de la capa. 5. El actor recibe el resultado de la operación realizada. 	<ol style="list-style-type: none"> 2. El sistema verifica que exista la capa. 3. El sistema obtiene todos los objetos (id, nombre y descripción) que pertenecen a dicha capa 4. El sistema envía el resultado de la operación realizada.
Sección: Eliminar objeto	
<ol style="list-style-type: none"> 1. El actor realiza la operación eliminar objeto, pasándole el identificador (la geometría del mismo en formato WKT) del objeto que desea eliminar. 	<ol style="list-style-type: none"> 2. El sistema realiza la operación de verificar si existe ese objeto, si existe lo elimina. 3. El sistema actualiza la base de datos.

5. Recibe el resultado de la operación realizada.	4. El sistema envía el resultado de la operación realizada.
Cursos alternos	
Si ocurre un error en el proceso de crear, modificar, obtener o eliminar un objeto se envía el mensaje de error correspondiente.	

6. Conclusiones

En este capítulo se comenzó a profundizar en el desarrollo de la propuesta de solución, obteniéndose:

1. El modelo del dominio o modelo conceptual, el cual ha permitido un mejor estudio del entorno en el que se desarrollará el sistema.
2. Una lista de los requisitos funcionales y no funcionales con los que debe cumplir el software.
3. El actor del sistema, los casos de uso del sistema y su relación con dichos actores.
4. El Diagrama de casos de uso del sistema, dónde se muestra de manera gráfica las funcionalidades que brindará el mismo, que se corresponden con el conjunto de servicios web a desarrollar.
5. Una descripción detallada de los casos de uso seleccionados.

Teniendo en cuenta los artefactos modelados, se puede pasar a la próxima fase de este proyecto, tomando como guía los casos de uso que se han definido y que estarán presentes durante todo su desarrollo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

1. Introducción

Con el presente capítulo se pretende obtener el Modelo de análisis y diseño de los servicios web que se están desarrollando. Este modelo establece la realización de los casos de uso en clases, pasando desde una representación en términos de análisis (donde no se incluyen aspectos de la implementación) hacia una de diseño (que sí incluye una orientación hacia el entorno de implementación). El análisis consiste en obtener una visión del sistema que se preocupa de ver QUÉ hace, de modo que sólo se interesa por los requisitos funcionales. Por otro lado, el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva CÓMO cumple el sistema sus objetivos. Se obtendrá además un Modelo de datos que servirá para la creación de la base de datos que utilizará la plataforma de servicios web.

2. Modelo de análisis

En el capítulo anterior se elaboró un Modelo de dominio que describe y ayuda a comprender los conceptos significativos al entorno del problema; no obstante, para lograr un mayor entendimiento de este, se hace necesario realizar el Modelo de análisis que servirá a su vez como una primera aproximación al Modelo de diseño.

En el modelo de análisis no se toma en cuenta el lenguaje de programación que se va a utilizar en la construcción, ni otros aspectos como la plataforma o los componentes reutilizables de otros sistemas. Su objetivo principal es comprender exactamente los requisitos del software y no precisar cómo se realizará su implementación. Se compone por clases del análisis y sus objetos organizados en paquetes que colaboran entre sí.

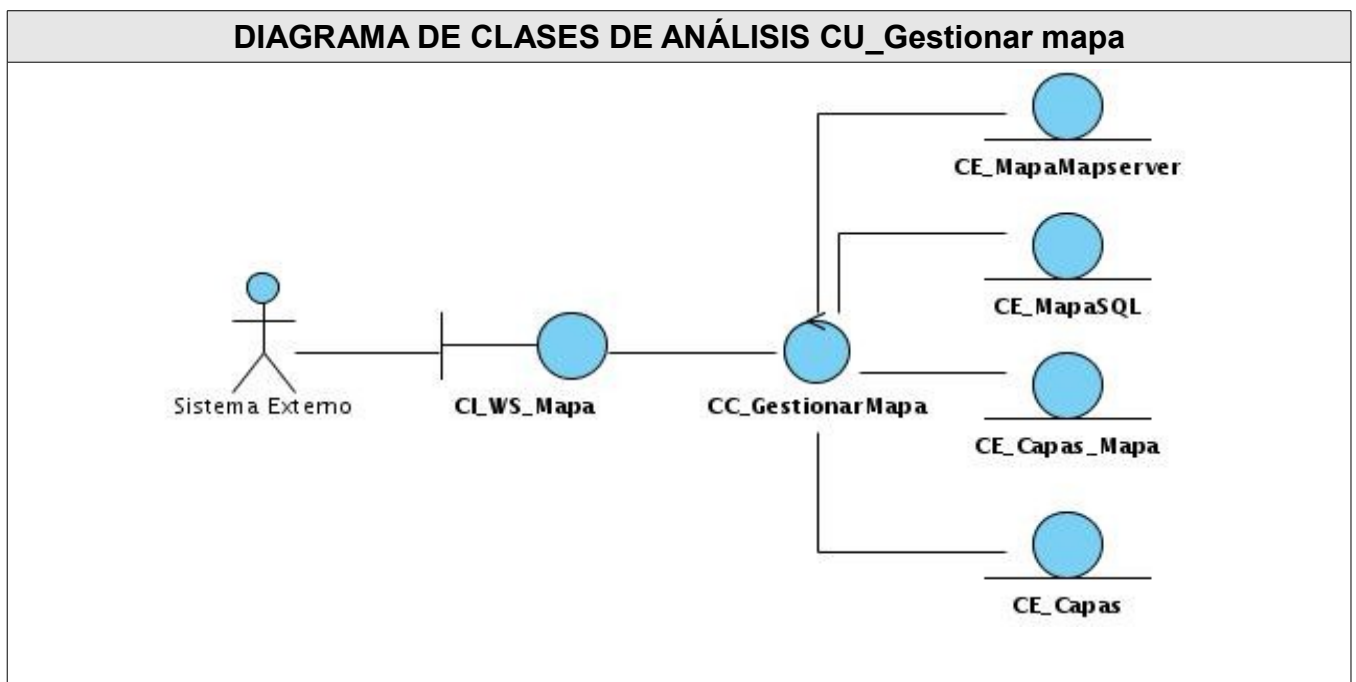
2.1 Diagramas de clases de análisis

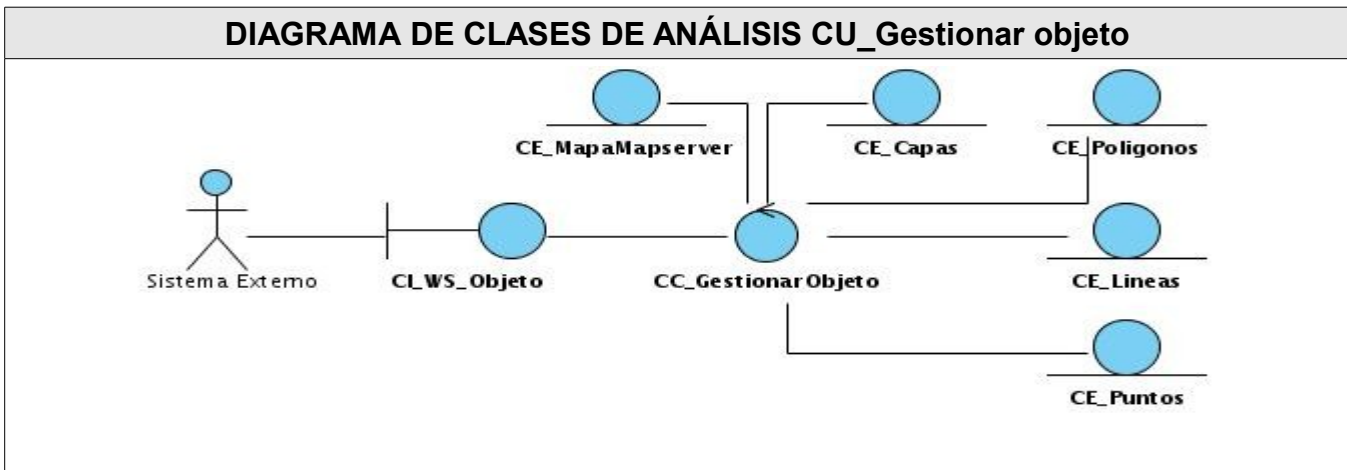
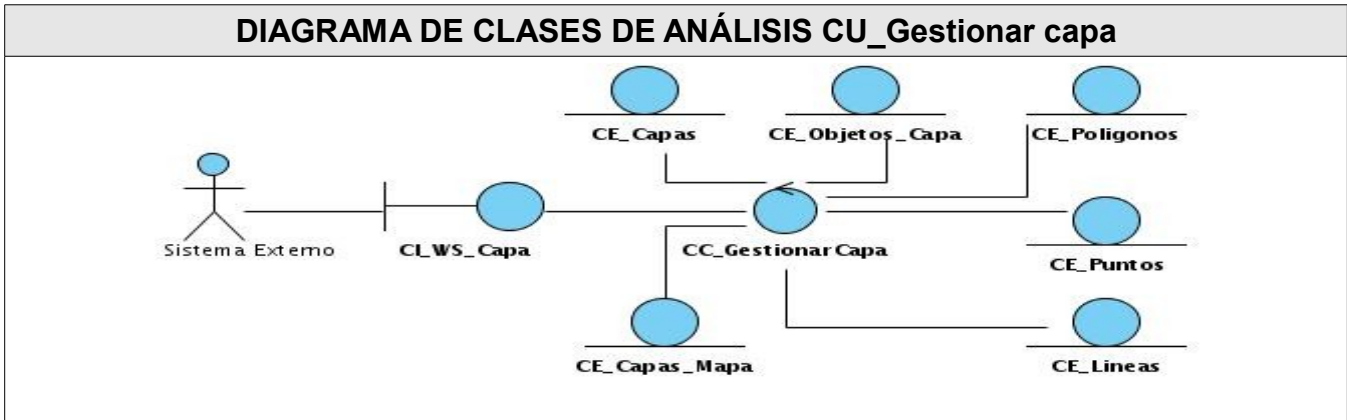
El diagrama de clases de análisis es un artefacto en el que se representan los conceptos del dominio del problema y las cosas del mundo real. Está compuesto por clases del análisis y sus relaciones. Se clasifican en:

Entidad: Modelan información que posee larga vida y que es a menudo persistente.

Interfaz: Modelan la interacción entre el sistema y sus actores.

Control: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.





3. Modelo de diseño

El Modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso. Se centra en cómo los requisitos funcionales y no funcionales tienen impacto en el sistema. Crea una entrada apropiada y un punto de partida para la implementación. A diferencia del Modelo de análisis, que es genérico frente al diseño (puede ser aplicado para varios diseños), el Modelo de diseño es específico para una implementación y depende del lenguaje de programación.

En el diseño modelamos el sistema y encontramos su forma (incluida la arquitectura) para que soporte todos los requisitos, incluyendo los no funcionales y las restricciones que se le suponen. La esencia del diseño es la elaboración de los diagramas de interacción, que describen cómo los objetos se comunican entre ellos para dar cumplimiento a los requerimientos que fueron definidos y permiten la realización de los diagramas de clases del diseño, que contienen las clases que se pueden implementar en un software.

3.1 Arquitectura del sistema

Con el objetivo de comprender y organizar mejor el sistema se hace necesario definir una arquitectura que sea robusta y estable. La arquitectura que se empleará para el desarrollo de esta aplicación responde a la Arquitectura en Capas.

El estilo en capas se define como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior. [Garlan, Shaw 1994] Las capas son agrupaciones lógicas de los componentes de software que constituyen una aplicación o negocio. Estas ayudan a diferenciar entre la clase de tareas desarrolladas por los componentes y hacen fácil el diseño de la reutilización.

Ventajas del estilo en capas:

1. Soporta un diseño basado en niveles de abstracción crecientes.
2. Permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
3. Admite optimizaciones y refinamientos.
4. Proporciona amplia reutilización. Al igual que los tipos de datos abstractos.
5. Se pueden utilizar diferentes implementaciones o versiones de una misma capa en la medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto brinda la posibilidad de definir interfaces de capa estándar, a partir de las cuales se pueden construir extensiones o prestaciones específicas.

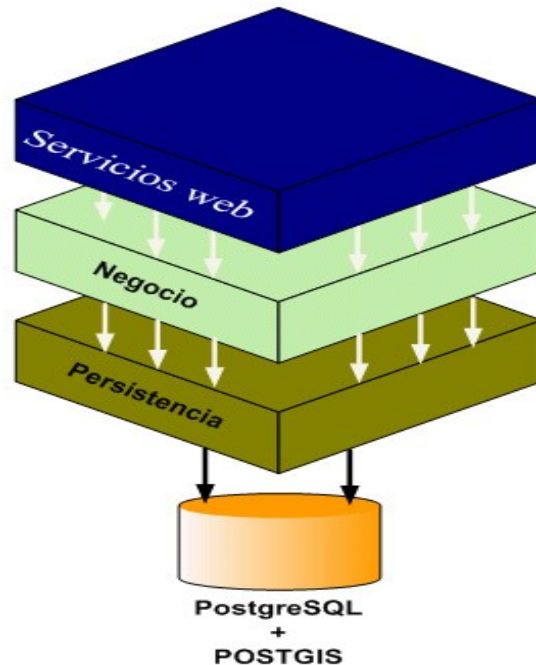


Figura # 3.1 Propuesta de capas que forman parte de la arquitectura del sistema

Como se observa en la figura anterior, se presenta una división entre capas, en la cual existe una definición muy clara de las responsabilidades a nivel de software de cada una de ellas, permitiendo el mantenimiento del código y la incorporación de nuevas funcionalidades, sin necesidad de que para esto sea necesario realizar modificaciones en todo el programa. Obteniendo mejores resultados, tanto de rendimiento como de mantenimiento del mismo.

El uso de esta arquitectura en capas permite mejorar la encapsulación, pues brinda más flexibilidad y escalabilidad. Además, mejora la seguridad. Proporcionando una buena organización y estructuración entre los distintos niveles de abstracción.

3.2 Patrones de diseño

Los patrones de diseño que empleamos son los patrones **GRASP**: General Responsibility Assignment Software Patterns (patrones de software para la asignación general

de responsabilidades).

- Experto

Problema: ¿Cuál es el principio fundamental en virtud del cual se asignan las responsabilidades en el diseño?

Solución: Asignar una responsabilidad al experto en información, la clase que cuenta con la información necesaria para cualquier responsabilidad.

Beneficios:

1. Se conserva el encapsulamiento.
2. El comportamiento se distribuye entre las clases que cuentan con la información requerida.
3. Clases más fáciles de mantener y de comprender.

- Creador

Problema: ¿Quién debería ser responsable de crear una nueva instancia de alguna clase?

Solución: Asignarle la responsabilidad a una clase de crear instancias de otra, ya sea agregando, utilizando, conteniendo, registrando o creando objetos de la otra clase.

Beneficios:

1. Brinda soporte a un bajo acoplamiento.
2. Mejores oportunidades de reutilización.

- Alta Cohesión

Problema: ¿Cómo mantener la complejidad dentro de límites manejables?

Solución: Asignar una responsabilidad de modo que la cohesión siga siendo alta.

Beneficios:

1. Mejoran la claridad y la facilidad con que se entiende el diseño.
2. Se simplifica el mantenimiento y las mejoras en funcionalidad.
3. A menudo se genera un bajo acoplamiento.
4. Mayor capacidad de reutilización.

- Bajo Acoplamiento

Problema: ¿Cómo dar soporte a una dependencia escasa y a un aumento de la reutilización?

Solución: Asignar responsabilidades a las clases de manera que la comunicación entre ellas sea la menor posible.

Beneficios:

1. No se afectan por cambios de otros componentes.
2. Fáciles de entender por separado.
3. Fáciles de reutilizar.

- Controlador

Problema: ¿Quién debería encargarse de atender un evento del sistema?

Solución: Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente una de las siguientes opciones:

El "sistema" global (controlador de fachada).

La empresa u organización global (controlador de fachada).

Beneficios:

1. Mayor potencial de los componentes reutilizables.
2. Reflexionar sobre el estado del caso de uso.

3.3 Diagramas de clases del diseño

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Muestran un conjunto de interfaces, colaboraciones y sus relaciones. Gráficamente, es una colección de nodos y arcos.

A continuación se muestran los diagramas de clases del diseño para cada caso de uso.

DIAGRAMA DE CLASES DEL DISEÑO CU_Gestionar mapa

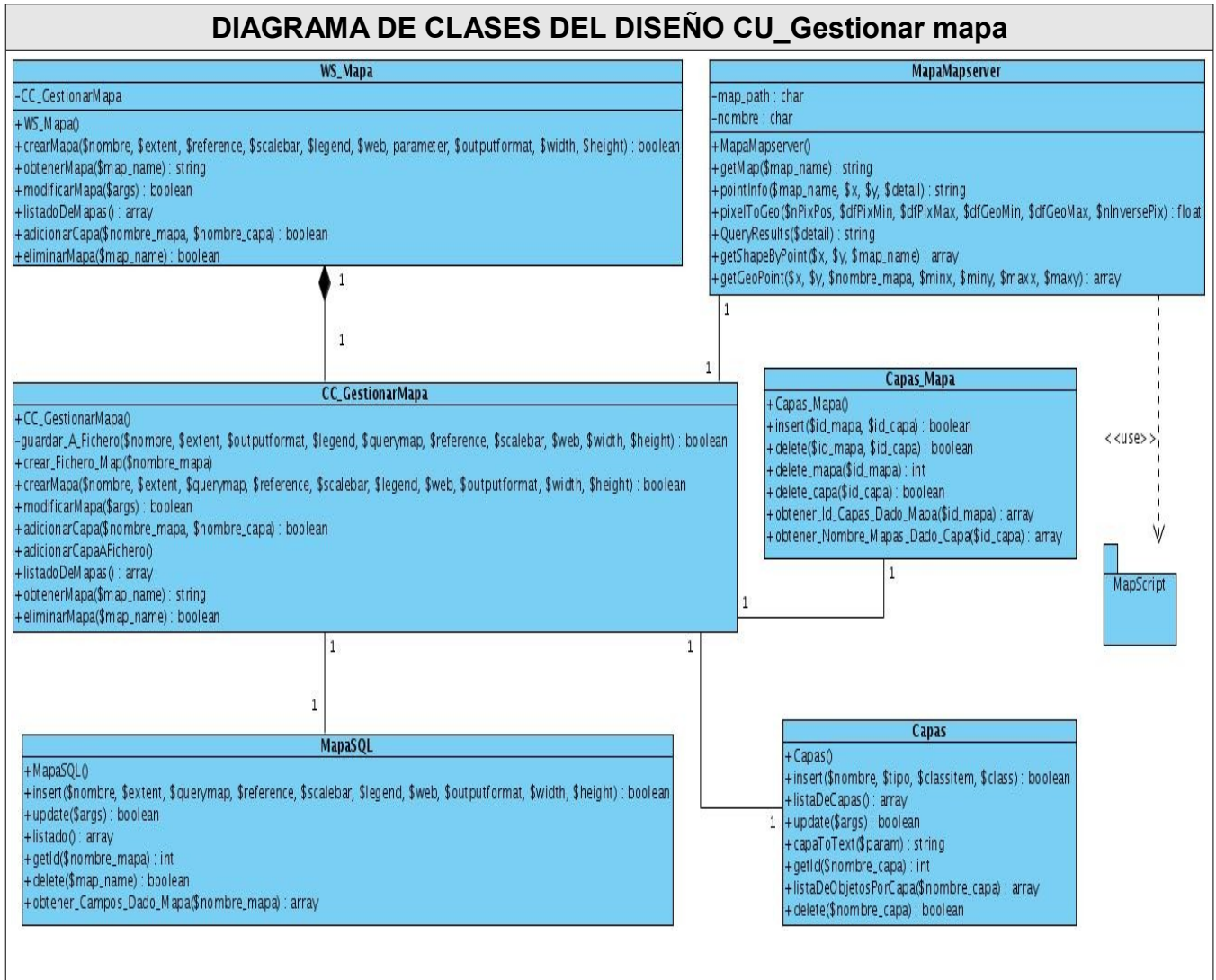


DIAGRAMA DE CLASES DEL DISEÑO CU_Gestionar capa

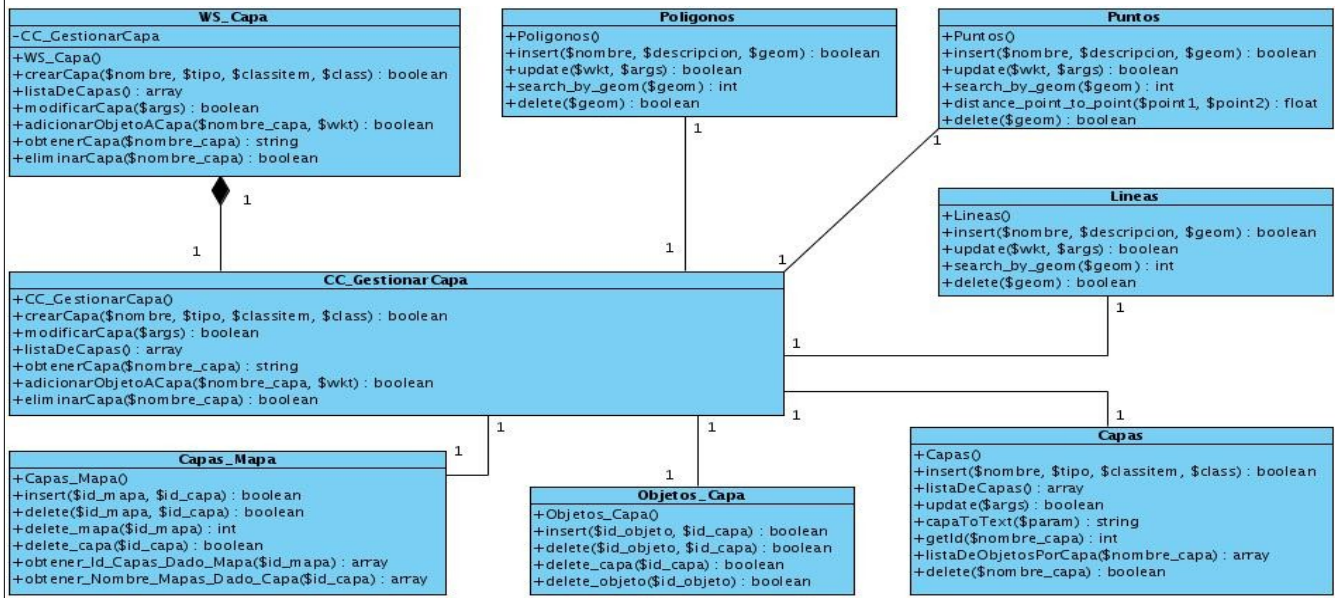
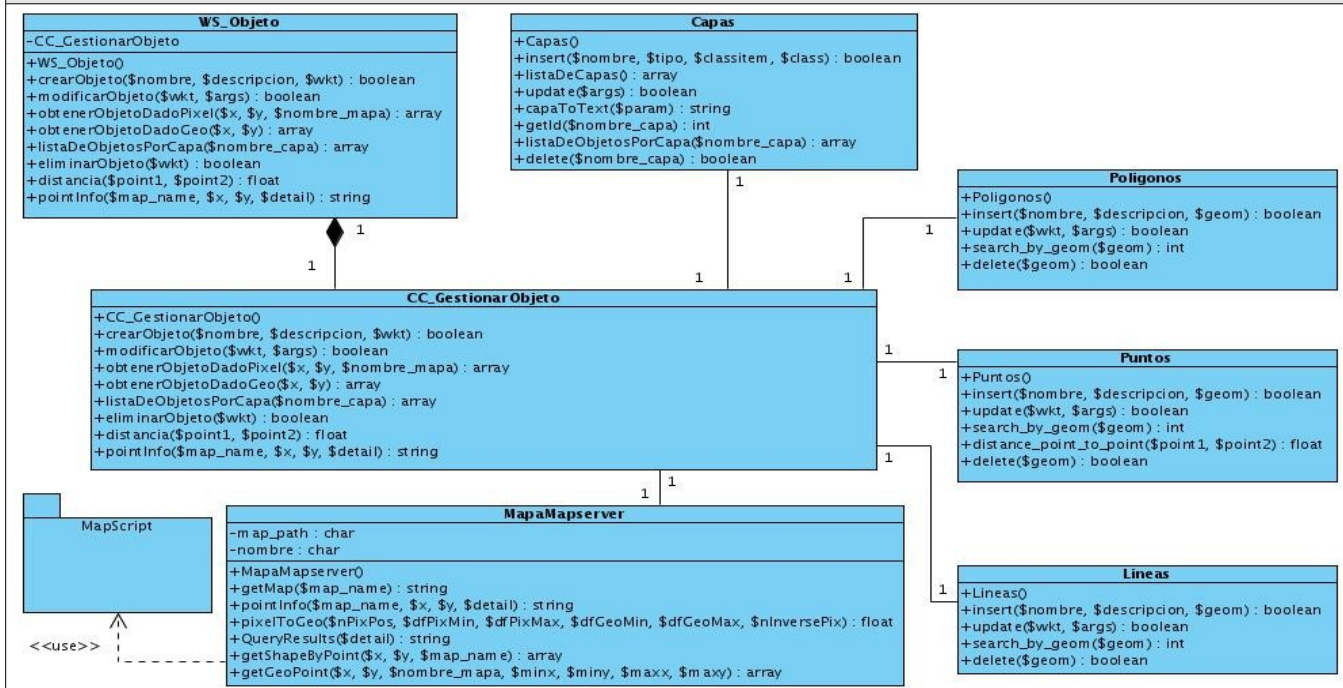
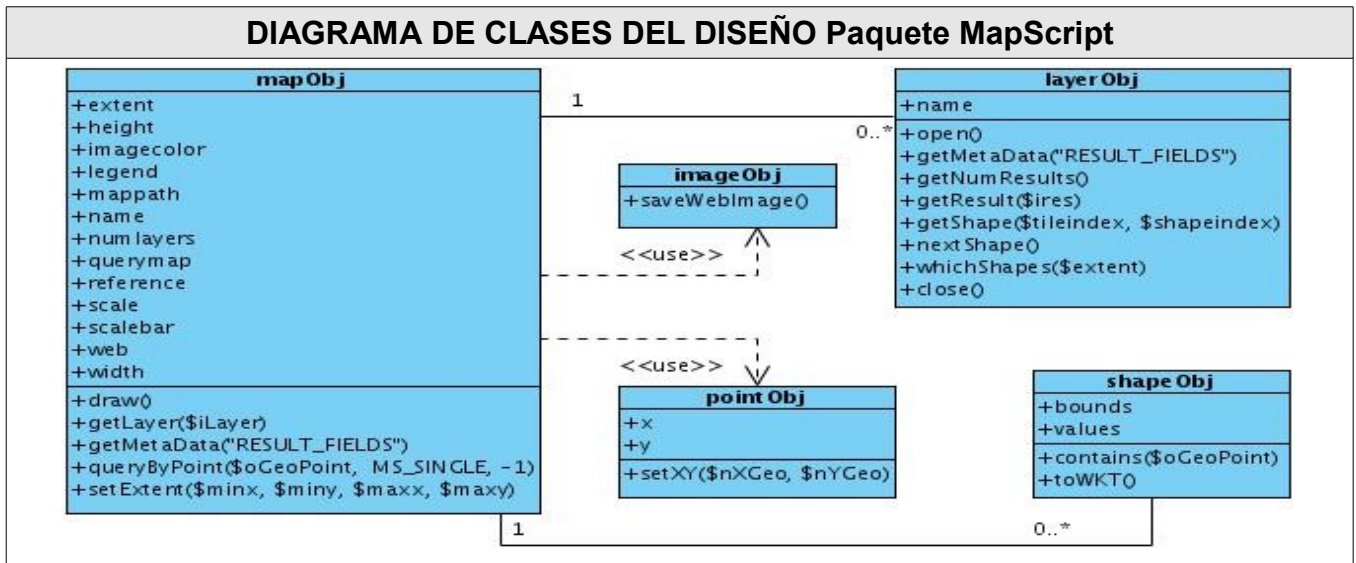


DIAGRAMA DE CLASES DEL DISEÑO CU_Gestionar objeto





3.4 Descripción de las clases del diseño

Clases	Descripción
[ws Servicio web]	Son las clases bases con prefijo WS . Publican todas las funciones disponibles como servicio web y brindan la posibilidad de saber cuáles son las características de dichas funciones mediante un fichero generado en XML (WSDL).
[CC Controladora]	Son las clases con prefijo CC y encapsulan en sus métodos todas las operaciones específicas para cada caso de uso del sistema.
[Entidades]	Existe una clase de acceso a datos para cada entidad de la base de datos. Implementan las operaciones básicas que se realizan sobre las clases persistentes del sistema. (insertar, eliminar, obtener, actualizar, entre otras).
MapScript	Es un módulo para PHP que permite acceder a la API de MapServer. Contiene funciones y clases como mapObj, pointObj, shapeObj, layerObj, entre otras, que estarán disponible dentro del entorno de desarrollo de la plataforma de servicios web. Está representado como un paquete.

Nombre: CC_GestionarMapa	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	CC_GestionarMapa()
Descripción:	Método constructor de la clase.
Nombre:	guardar_A_Fichero(\$nombre, \$extent, \$web, \$reference, \$querymap, \$legend, \$scalebar, \$outputformat, \$width, \$height)
Descripción:	Método que guarda en un fichero .map (fichero de configuración del mapa) los atributos del mismo, excepto el atributo Layer.
Nombre:	crear_Fichero_Map(\$nombre_mapa)
Descripción:	Método para crear el fichero .map (fichero de configuración del mapa), esta vez incluye la capa modificada o eliminada.
Nombre:	crearMapa(\$nombre, \$extent, \$web, \$reference, \$querymap, \$legend, \$scalebar, \$outputformat, \$width, \$height)
Descripción:	Método que nos permite crear un nuevo mapa.
Nombre:	modificarMapa(\$args)
Descripción:	Método que nos permite modificar un mapa existente.
Nombre:	adicionarCapa(\$nombre_mapa,\$nombre_capa)
Descripción:	Método que nos permite adicionarle una capa a un mapa existente.
Nombre:	adicionarCapaAFichero()
Descripción:	Método que nos permite adicionarle una capa al fichero de configuración del mapa.
Nombre:	listadoDeMapas()
Descripción:	Método que nos permite obtener un listado de todos los mapas existentes.
Nombre:	obtenerMapa(\$map_name)
Descripción:	Método que nos permite obtener un mapa existente.
Nombre:	eliminarMapa(\$map_name)
Descripción:	Método que nos permite eliminar un mapa existente.

Nombre: CC_GestionarCapa	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	CC_GestionarCapa()
Descripción:	Método constructor de la clase.
Nombre:	crearCapa(\$nombre,\$tipo,\$classitem,\$class)
Descripción:	Método que nos permite crear una nueva capa.
Nombre:	modificarCapa(\$args)
Descripción:	Método que nos permite modificar una capa existente.
Nombre:	adicionarObjetoACapa(\$nombre_capa, \$wkt)
Descripción:	Método que nos permite adicionarle un objeto a una capa existente.
Nombre:	listadoDeCapas()
Descripción:	Método que nos permite obtener un listado de todas los capas existentes.
Nombre:	obtenerCapa(\$nombre_capa)
Descripción:	Método que nos permite obtener una capa existente.
Nombre:	eliminarCapa(\$nombre_capa)
Descripción:	Método que nos permite eliminar una capa existente.

Nombre: CC_GestionarObjeto	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	CC_GestionarObjeto()
Descripción:	Método constructor de la clase.
Nombre:	crearObjeto(\$nombre,\$descripcion,\$wkt)
Descripción:	Método que nos permite crear un nuevo objeto.
Nombre:	modificarObjeto(\$wkt,\$args)
Descripción:	Método que nos permite modificar un objeto existente.
Nombre:	obtenerObjetoDadoPixel(\$x,\$y,\$nombre_mapa)

Descripción:	Método que nos permite obtener un objeto dado un punto expresado en píxel.
Nombre:	obtenerObjetoDadoGeo(\$x,\$y)
Descripción:	Método que nos permite obtener un objeto dado un punto expresado en coordenadas geográficas.
Nombre:	listaDeObjetosPorCapa(\$nombre_capa)
Descripción:	Método que nos devuelve el listado de objetos pertenecientes a la capa seleccionada.
Nombre:	distancia(\$point1,\$point2)
Descripción:	Método que nos permite obtener la distancia existente entre dos puntos seleccionados.
Nombre:	pointInfo(\$map_name,\$x, \$y,\$detail)
Descripción:	Método que nos permite obtener la información de un punto especificado.
Nombre:	eliminarObjeto(\$wkt)
Descripción:	Método que nos permite eliminar un objeto existente.

3.5 Diagramas de interacción

Muestran las interacciones entre objetos mediante la transferencia de mensajes entre objetos o subsistemas. Se dividen en dos tipos de diagramas de UML: los diagramas de secuencia (muestran las interacciones entre objetos, ordenadas en secuencia temporal) y los diagramas de colaboración (muestran las interacciones entre objetos organizadas entorno a los objetos y los enlaces entre ellos).

Para modelar los aspectos dinámicos de este sistema se utilizaron diagramas de secuencia por cada escenario de los casos de uso. (Ver Anexo II)

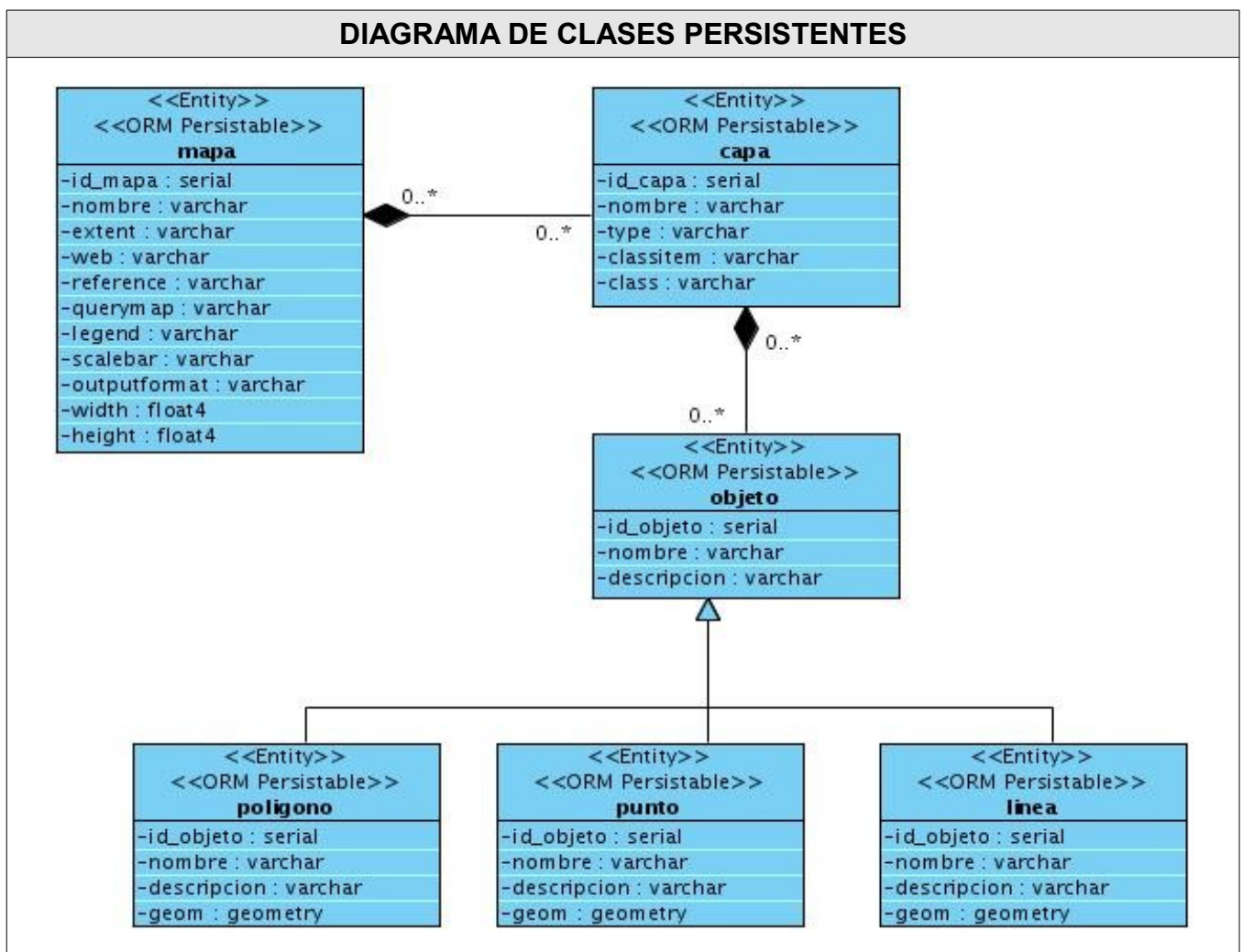
4. Diseño de la base de datos

Para el desarrollo de la plataforma de servicios web se hace necesario el almacenamiento y recuperación de información. A los efectos se utilizó para diseñar la base de datos el diagrama de clases persistentes y el modelo de datos.

4.1 Diagrama de clases persistentes

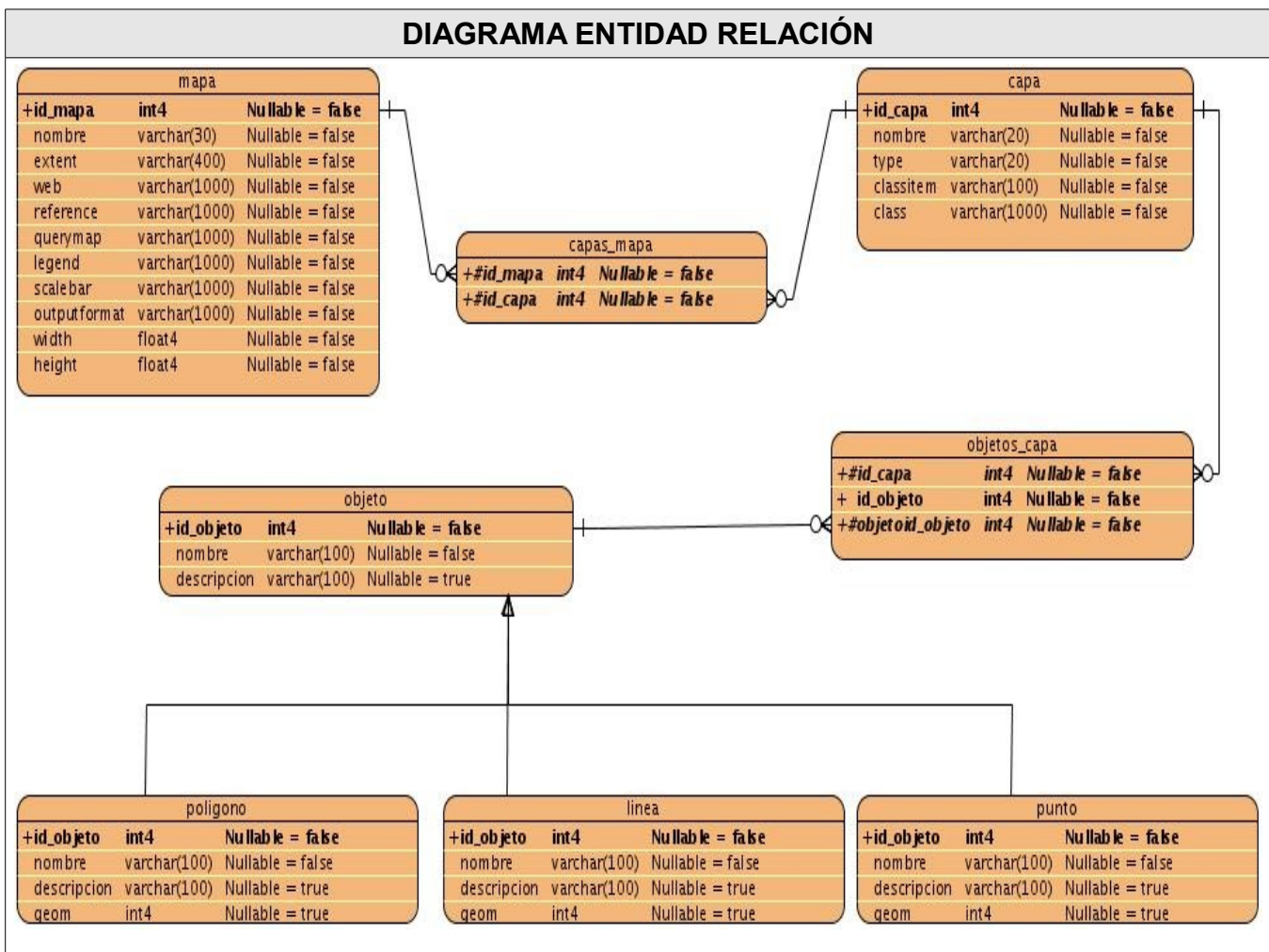
La persistencia es la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. A continuación se presenta el diagrama de clases persistentes del sistema. Las clases que modelamos en este diagrama son aquellas que almacenan y obtienen datos durante los procesos de la aplicación.

A continuación se muestra el diagrama de clases persistentes:



4.2 Modelo de datos

A partir del diagrama de clases persistentes se obtiene el Modelo de datos, aplicando para esto las restricciones de las bases de datos relacionales. Este modelo es usado para describir la representación física de la información persistente manejada por la plataforma de servicios web.



4.3 Descripción de las tablas de la base de datos

(Ver Anexo III)

5. Conclusiones

Con el desarrollo de este capítulo se ha logrado:

1. Obtener los modelos de análisis y diseño del sistema, que sirven de base para su implementación.
2. Llevar a cabo la descripción de las clases del diseño.
3. Definir las clases persistentes y a partir de esto, se construyó el modelo de datos.
4. Describir las tablas que almacenan la información que persiste a lo largo del tiempo en la base de datos del sistema.

Por tanto se decide comenzar la implementación de la plataforma hasta ahora modelada.

CAPÍTULO 4: IMPLEMENTACIÓN

1. Introducción

Este capítulo se centra principalmente en la elaboración del Modelo de implementación propuesto. Conjuntamente se representa el Modelo de despliegue de la aplicación para una mejor descripción de la solución. En la implementación se toma como punto de partida el resultado obtenido en el diseño y describe cómo los elementos del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos en el diagrama de despliegue.

2. Modelo de despliegue

Es un modelo de objeto que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre nodos de cómputo. Se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. Permite además comprender la correspondencia entre la arquitectura software y la arquitectura hardware.

A continuación se muestra el diagrama de despliegue correspondiente al sistema de esta investigación.

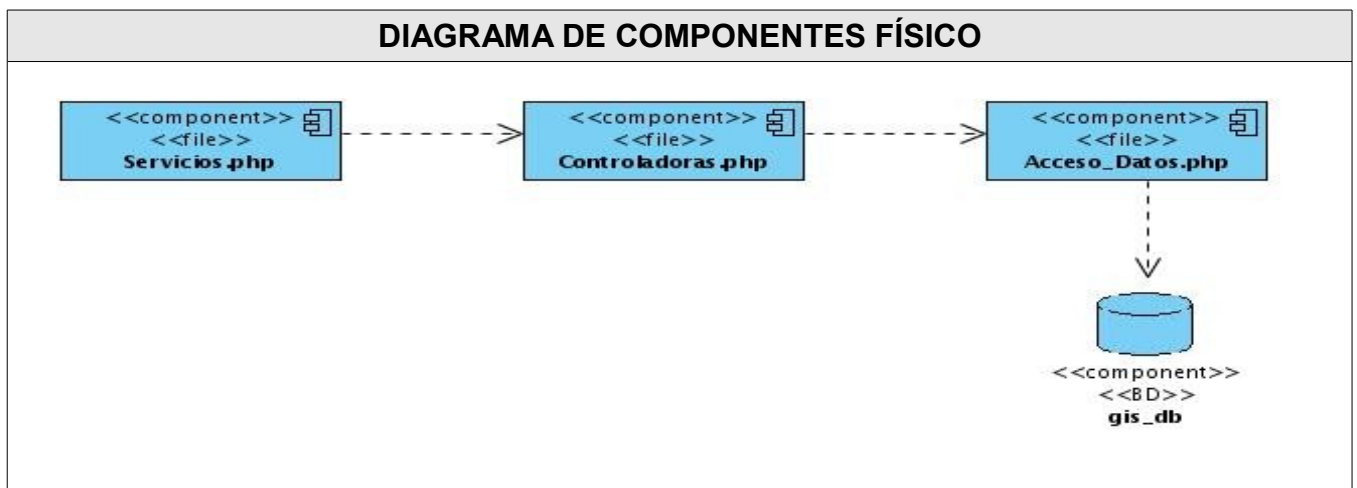


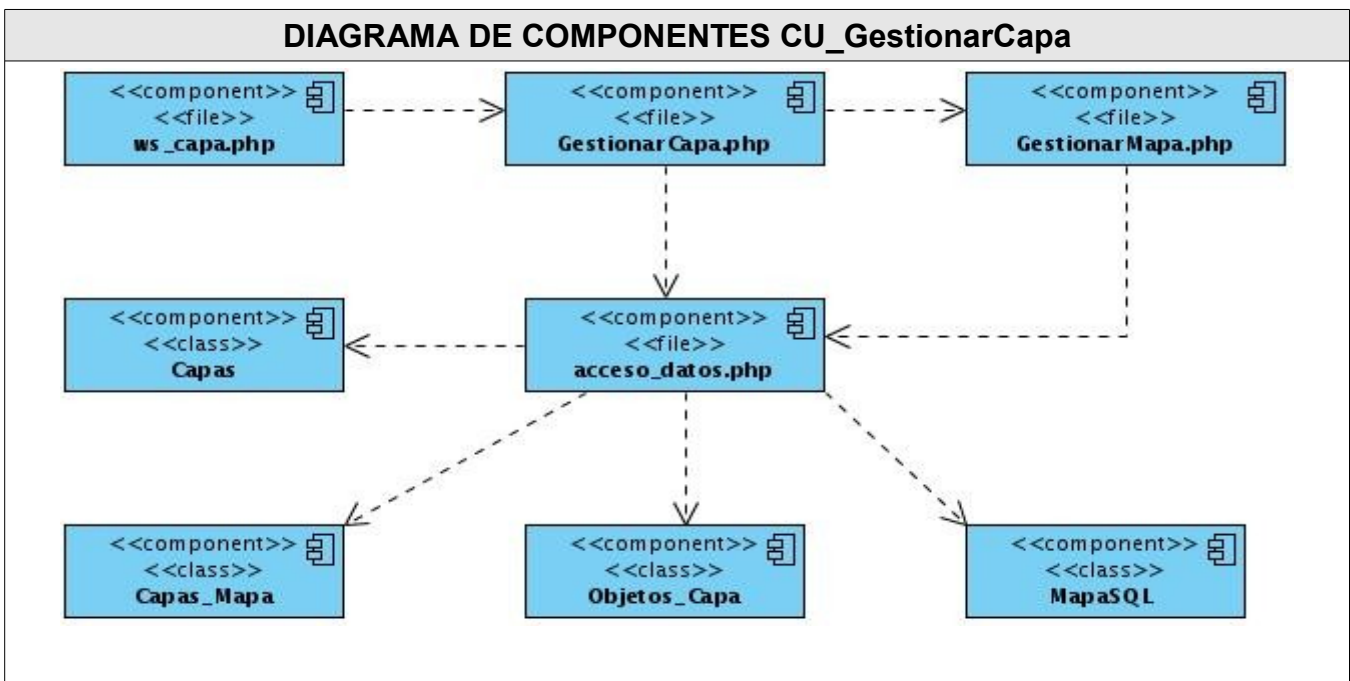
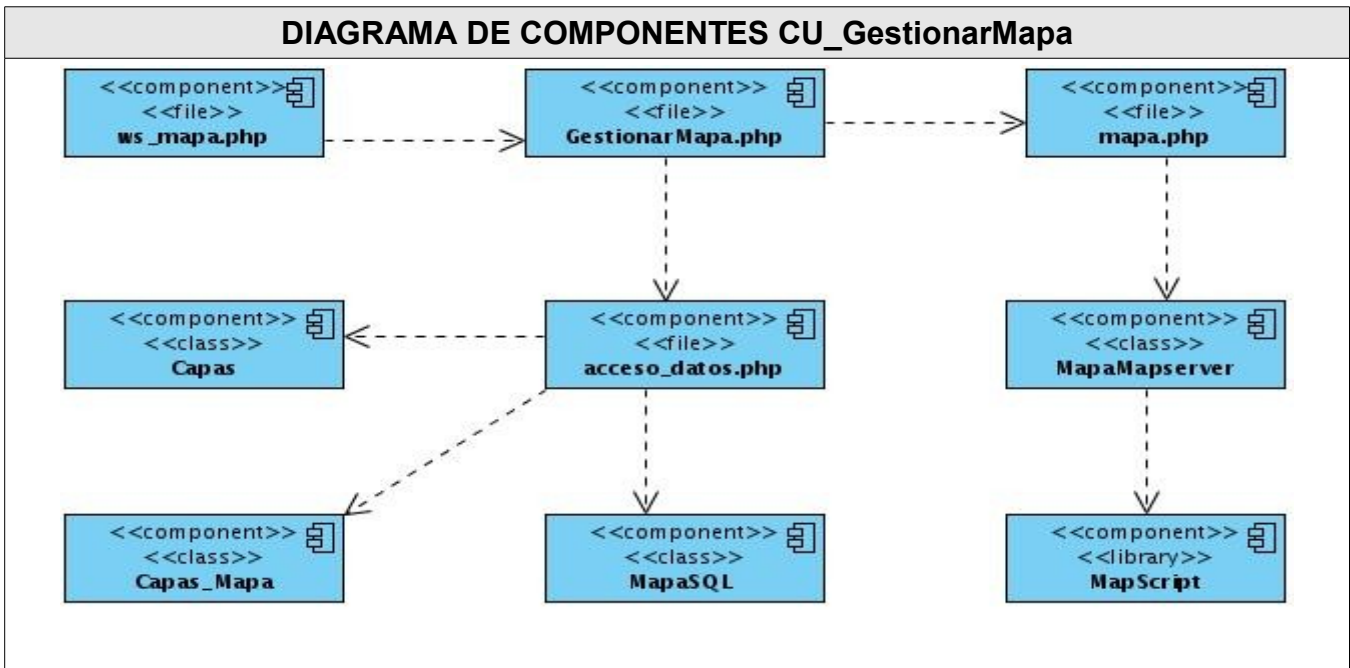
En este diagrama el nodo PC_Cliente, representa al Sistema Externo que se conecta para consumir los servicios web, al nodo Servidor Web Apache (php, php_mapscript), utilizando el protocolo HTTP. Dicho nodo representa dónde están publicados los servicios web y se comunica con el Servidor de Base de Datos (PostgreSQL/ PostGIS), a través del protocolo TCP/IP, para realizar consultas y actualizaciones de la información que manipula el sistema con la base de datos.

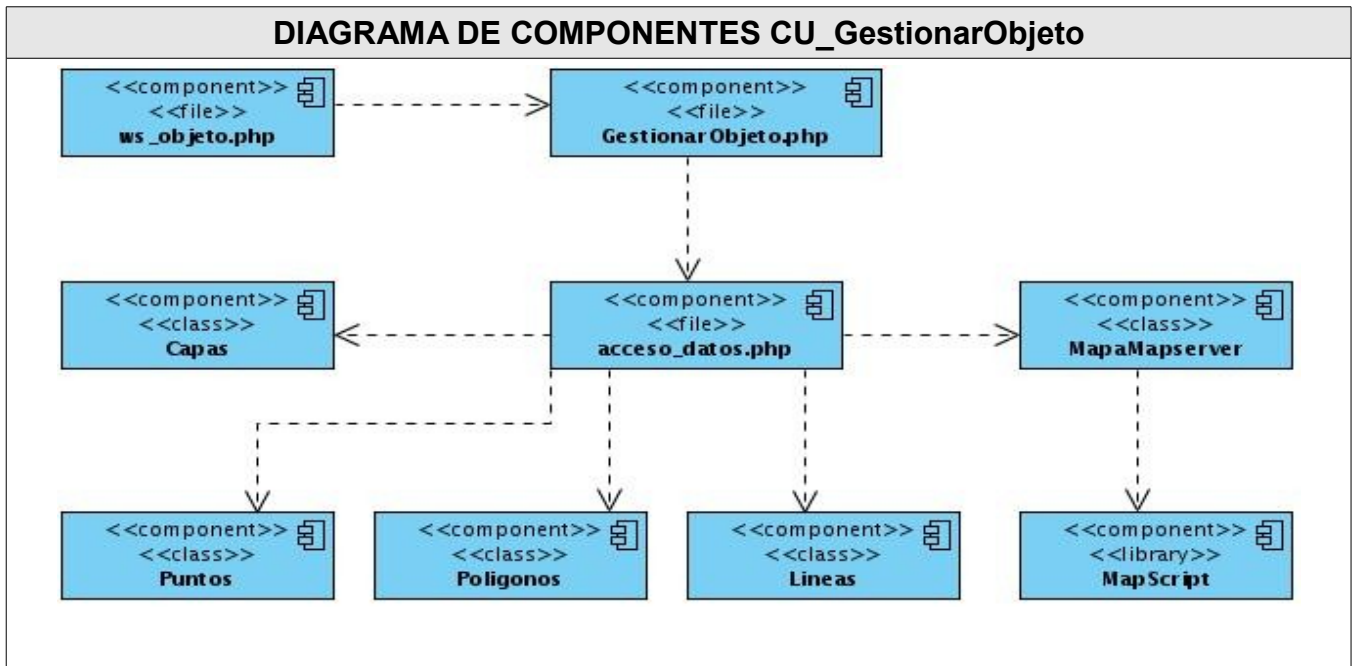
3. Modelo de implementación

Este modelo es la entrada principal de las etapas de prueba que siguen a la implementación. Dentro de este modelo se incluyen los Diagramas de componentes, los cuales se representan como un grafo de componentes software unidos por relaciones de dependencia (compilación, ejecución, traza, usa, entre otros) y muestran un conjunto de elementos del modelo de implementación, tales como: componentes, subsistemas de implementación y sus relaciones.

A continuación se muestra el diagrama de componentes físico y un diagrama de componentes por cada caso de uso.







4. Conclusiones

En este capítulo se expuso la estructura de la aplicación obteniéndose:

1. El diagrama de despliegue del sistema, el cual nos permitió definir las relaciones existentes entre el software y el hardware donde se implantará, así como las relaciones entre las otras máquinas que interactuarán con él.
2. El Modelo de implementación, representándose los diagrama de componentes de cada uno de los casos de uso de la aplicación guiados por las definiciones realizadas durante el diseño.

De esta forma cualquier desarrollador podrá conocer cómo está implementada internamente esta plataforma de servicios web.

CONCLUSIONES GENERALES

Como resultado del trabajo realizado, se llega a las siguientes conclusiones:

1. El objetivo general planteado fue cumplido satisfactoriamente, así como las tareas descritas para llevar a cabo dicho objetivo.
2. La plataforma de servicios web está construida en su totalidad por herramientas libres, reduciendo el costo de desarrollo.
3. El sistema cuenta con una documentación detallada de lo implementado hasta el momento, por lo que constituye una base para futuros desarrolladores.
4. Se modelaron todos los diagramas y representaciones necesarias.
5. La utilización de la metodología RUP avala que el desarrollo del software quede estructurado por fases, con la calidad requerida, además de reducir el tiempo de desarrollo.
6. Se lograron definir, diseñar e implementar todos los servicios web requeridos inicialmente para el procesamiento y gestión de información geoespacial.
7. El desarrollo de esta plataforma constituye un aporte práctico muy importante, debido a que es una novedad tecnológica que marca un paso de avance en el desarrollo de software multiplataforma e interoperable en el ámbito de los sistemas de información geográfica.

RECOMENDACIONES

Teniendo en cuenta las conclusiones del trabajo se recomienda:

1. La realización de las pruebas a la plataforma de servicios web para validar el sistema implementado.
2. La utilización del presente trabajo como bibliografía para posibles investigaciones referentes al tema desarrollado en el mismo.
3. Publicar los servicios web implementados en la UDDI de la Universidad de las Ciencias Informáticas, para permitir el acceso a la información espacial de dicha universidad a todos los sistemas que están y estarán en funcionamiento dentro de la institución.
4. Analizar su posible incorporación en la Infraestructura de Datos Espaciales de la República de Cuba (IDERC).
5. Expandir el uso de aplicaciones que estén estructuradas y basadas en el funcionamiento de servicios web.

REFERENCIAS BIBLIOGRÁFICAS

1. [CCIDEP 2008] CCIDEP Comité Coordinador de la Infraestructura de Datos Espaciales. *Servicios WMS Activos*. 2008. [consultado en: febrero 2008] Disponible en: <<http://www.ccidep.gob.pe>>
2. [Delgado 2005] Delgado Fernández, Tatiana. *Infraestructuras de datos espaciales para países de bajo desarrollo tecnológico. Implementación en Cuba*. Tesis de Doctorado. Instituto Técnico Militar “José Martí”. 2005.
3. [Convencion 2003] Convención Nacional de Geografía Memoria digital. *Estándares Globales en infraestructuras de datos espaciales*. 2003. [consultado en: febrero 2008] Disponible en: <<http://mapserver.inegi.gob.mx/geografia/espanol/cartcat/convencion/menu/5112.pdf>>
4. [Delgado 2004] Delgado Fernández, Tatiana. *IDERC Portal Geoespacial Nacional*. 2004. [consultado en: febrero 2008] Disponible en: <<http://www.iderc.co.cu/>>
5. [OPTIC 2007] OPTIC Oficina Presidencial de Tecnologías de la Información y Comunicación. *Normas y estándares de interoperabilidad*. 2007 [consultado en: febrero 2008] Disponible en: <http://www.optic.gob.do/biblioteca_virtual/Norma_Interoperabilidad.pdf>
6. [CAS 2001] CAS Corporación Autónoma Regional de Santander. *Servicios Web*. 2001. [consultado en: febrero 2008] Disponible en: <<http://sig.cas.gov.co/services/web>>
7. [Terés 2007] Terés Martínez, R. *Publicación y acceso a servicios web en una red peer to peer DHT*. 2007. [consultado en: febrero 2008] Disponible en: <<https://upcommons.upc.edu/pfc/bitstream/2099.1/4471/1/martinez.pdf>>

8. [E,A 2006] Espinoza M, Alvarez P. *El ciclo de vida de un servicio Web compuesto: virtudes y carencias de las soluciones actuales*. 2006. [consultado en: febrero 2008] Disponible en:<<http://webdiis.unizar.es/~mepinoz/ReporteComposici%F3n.pdf>>
9. [Barron 2007] Barron Ramírez, M. *SOA - Introduccion*. 2007.[consultado en: febrero 2008] Disponible en:<<http://comunidadnetmonterrey.org/blogs/mbarron>>
- 10.[Barco 2006] Barco Antonio. *Arquitectura Orientada a Servicios (SOA)*. 2006. [consultado en: febrero 2008] Disponible en: <<http://arquitecturaorientadaaservicios.blogspot.com>>
- 11.[Mendoza 2004] Mendoza, M.A. *Metodologías de desarrollo de software*. 2004. [consultado en: febrero 2008] Disponible en:<http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07_062004.html>
- 12.[Alvarez, Cisneros 2006] Alvarez I; Cisneros F. *Mapserver: Herramienta para el desarrollo de aplicaciones Web de información georreferenciada*. 2006. [consultado en: febrero 2008] Disponible en:<http://www.informaticahabana.com/evento_virtual/files/MUL006.ppt>
- 13.[Herrera, Lopez, Barreras 2006] Herrera JP; Lopez OI; Barreras F. *OPENGIS aplicado a la mitigación de desastres naturales*. 2006. [consultado en: febrero 2008] Disponible en:<<http://www.cibersociedad.net/congres2006/gts/comunicacio.php?id=1032&llengua=es>>
- 14.[VAN DER HENST 2001] VAN DER HENST, Christian. *¿Qué es el PHP?*. 2001. [consultado en: febrero 2008] Disponible en:<<http://www.maestrosdelweb.com/editorial/phpintro/>>

- 15.[Jimenez 2006] Jimenez A.R. *¿Por qué elegir PHP?*. 2006. [consultado en: febrero 2008] Disponible en:<<http://www.mmug.cl/articulos.php?id=283>>
- 16.[Worsley, Drake 2001] Worsley John; Drake Joshua. *PostgreSQL Práctico*. 2001. [consultado en: febrero 2008] Disponible en: <<http://www.sobl.org/traduccion/practical-postgres/node12.html>>
- 17.[Martín 2006] Martín Martín, M. *Manual PostGIS*. 2006. [consultado en: febrero 2008] Disponible en:<<http://postgis.refractory.net/documentation/postgis-spanish.pdf>>
- 18.[Palacín 2007] Palacín Lorente David. *Representación y edición de un plan de vuelo sobre un modelo digital de elevaciones en entorno .NET*. Tesis de final de carrera. Universidad Politécnica de Catalunya. Septiembre 2007. [consultado en: marzo 2008] Disponible en: <<http://upcommons.upc.edu/pfc/bitstream/2099.1/4472/1/lorente.pdf>>
- 19.[Escobar, Hunter, Bishop, Zerger 1999] Escobar F; Hunter G; Bishop I; Zerger A. *Introducción a los SIG*. 1999. [consultado en: marzo 2008] Disponible en: <<http://www.geogra.uah.es/gisweb/1modulosespanyol/IntroduccionSIG/GISModule/GISTheory.doc>>
- 20.[Eduteka 2005] Fundación Gabriel Pedrahita Uribe. *LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA (SIG) EN LA EDUCACIÓN ESCOLAR DEL SIGLO XXI*. Junio 2005. [consultado en: marzo 2008] Disponible en:<<http://www.eduteka.org/SIG1.php>>
- 21.[Garlan, Shaw 1994] Garlan David; Shaw Mary. *An introduction to Software Architecture*. 1994. [consultado en: marzo 2008] Disponible en: <http://www.cs.cmu.edu/afs/cs/project/vit/ftp/pdf/intro_softarch.pdf>

BIBLIOGRAFÍA CONSULTADA

1. Jacobson, Ivar; Booch, Grady; Rumbaugh James. *El proceso unificado de software*. Volumen I. La Habana, 2004. Editorial Félix Varela
2. Jacobson, Ivar; Booch, Grady; Rumbaugh James. *El proceso unificado de software*. Volumen II. La Habana, 2004. Editorial Félix Varela
3. Larman, Craig. 1999. *UML Y Patrones, Introducción al análisis y diseño orientado a objeto* Tomo(I, II). Ciudad de la Habana, Cuba. Editorial Felix Varela
4. Pressman, Roger S. *Ingeniería del software, un enfoque práctico*. Parte 1. La Habana, 2005. Editorial Félix Varela
5. Pressman, Roger S. *Ingeniería del software, un enfoque práctico*. Parte 2. La Habana, 2005. Editorial Félix Varela
6. Rational Unified Process, Rational Software Corporation, *Rational Unified Process*, Version 2001A.04.00, Copyright 1987-2001
7. Delgado Fernández, Tatiana. *Iniciativa cubana para la creación de una Infraestructura Nacional de Datos Espaciales: Marco institucional, Estudio de Factibilidad y Servicio de Catálogos*. Ponencia presentada a la 5ta Conferencia de GSDI. Cartagena de Indias, Colombia. Mayo 2001
8. Delgado Fernández, Tatiana. *Avances en la Implementación de la Infraestructura de Datos Espaciales de la República de Cuba. Aspectos tecnológicos*. III Congreso Internacional GEOMATICA 2002, La Habana, Cuba. Marzo 2002
9. Delgado Fernández, Tatiana. *La Infraestructura de Datos Espaciales de la República de Cuba (IDERC): Perspectivas de negocio, de información, computacional y de*

ingeniería. IV Congreso Internacional GEOMATICA. La Habana, Cuba. Mayo 2004

10. Estupiñán Gómez, J.F. *PLANIFICACIÓN Y COLECCIÓN DE INFORMACIÓN GEOGRÁFICA MEDIANTE GEOSENSORES EN AMBIENTE GRID*. Tesis de maestría en ciencias de la información y las comunicaciones. 2007
11. González V, Oñate F. *Infraestructura de datos espaciales (ide) para el estudio y análisis ambiental: una experiencia en el sur del Ecuador*. 2007. [consultado en: febrero 2008] Disponible en: http://www.fronate.pro.ec/fronate/wp-content/media/2008/01/ide_ambiental_ecuador.PDF
12. Jimenez, José A.; Aguilera, Ma. Jesús; Meroño, José E. *ALTERNATIVAS DE SOFTWARE LIBRE A LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA COMERCIALES*. [consultado en: febrero 2008] Disponible en: <http://www.cartesia.org/geodoc/ingegraf2005/gis10.pdf>
13. OGC 2002. *Sitio oficial de Open Geospatial Consortium*. [consultado en: febrero 2008] Disponible en: <http://www.opengeospatial.org/>
14. OGC 2002. *OpenGIS® Web Map Service (WMS) Implementation Specification*, OGC 03-109r1, Versión 1.3.0. Enero 2004. [consultado en: febrero 2008] Disponible en: http://portal.opengeospatial.org/files/?artifact_id=4756
15. OGC 2002. *OpenGIS® Catalogue Service Implementation Specification*, OGC 07-006r1, Version 2.0.2. Febrero 2007. [consultado en: febrero 2008] Disponible en: http://portal.opengeospatial.org/files/?artifact_id=20555
16. OGC 2002. *OpenGIS® Web Coverage Service (WCS) Implementation Specification*, OGC 06-083r8, Versión 1.1.0. Octubre 2006. [consultado en: febrero 2008] Disponible en: https://portal.opengeospatial.org/files/?artifact_id=18153

17. OGC 2002. *OpenGIS® Web Feature Service (WFS) Implementation Specification*, OGC 04-094, Version 1.1.0. Mayo 2005. [consultado en: febrero 2008] Disponible en:<http://portal.opengeospatial.org/files/?artifact_id=8339>
18. Consorcio World Wide Web(W3C). *¿Qué es el Consorcio World Wide Web(W3C)?* [consultado en: febrero 2008] Disponible en:<<http://www.w3c.es/Consortio/>>
19. Consorcio World Wide Web(W3C). *Guía Breve de Servicios Web*. 2005. [consultado en: febrero 2008] Disponible en: <<http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>>
20. Wikipedia. *La Enciclopedia Libre*. 2006. [consultado en: febrero 2008] Disponible en:<http://es.wikipedia.org/wiki/Servicio_Web#Ventajas_de_los_servicios_Web>
21. Benjamín González C. *WSDL para la documentación de Servicios Web*. [consultado en: febrero 2008] Disponible en:<<http://www.desarrolloweb.com/articulos/1581.php>>
22. Benjamín González, C. *SOAP (Simple Object Access Protocol)*. [consultado en: febrero 2008] Disponible en:<<http://www.desarrolloweb.com/articulos/1557.php>>
23. Benjamín González, C. *XML: el lenguaje de los Servicios Web*. [consultado en: febrero 2008] Disponible en: <<http://www.desarrolloweb.com/articulos/1574.php>>
24. Benjamín González, C. *UDDI (Universal Description Discovery and Integration)*. [consultado en: febrero 2008] Disponible en:<<http://www.desarrolloweb.com/articulos/1589.php>>
25. Alvarez, M. A. *Qué es PHP*. [consultado en: febrero 2008] Disponible en:<<http://www.desarrolloweb.com/articulos/392.php>>
26. Vizcaíno A; García F.O; Caballero I. *Una herramienta CASE para ADOO: Visual Paradigm*. [consultado en: febrero 2008] Disponible en:<<http://alarcos.inf->

cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_VP.pdf>

27. Universidad de Minnesota. *Sitio Oficial de Mapserver*. [consultado en: febrero 2008]
Disponible en: <<http://mapserver.gis.umn.edu/>>
28. Penroz Díaz, A. A. *Graphical User Interface (GUI) para el programa servidor de mapas MapServer 4.6.1*. Chile, 2005. [consultado en: febrero 2008] Disponible en:
<http://berlin.dis.ufro.cl/titulo/GUI_mapserver.pdf>
29. Jaramillo González, V.H. *MapServer y su aplicación a SIG*. Universidad Técnica Particular de Loja. Ecuador, 2005. [consultado en: febrero 2008] Disponible en:
<http://sig.utpl.edu.ec/sigutpl/biblioteca/manuales/curso_mapserver.PDF>
30. Vrsalovic, Vinko. *USA POSTGRESQL Y POSTGIS PARA DESARROLLAR GIS (SIG)*. [consultado en: febrero 2008] Disponible en:
<http://www.postgresql.cl/entrevistas/v_vrsalovic.htm>
31. Sitio Oficial de PostGIS. *PostGIS*. [consultado en: febrero 2008] Disponible en:
<<http://postgis.refractions.net/>>
32. Corro Acosta, Héctor Javier. *Mapa virtual de la República Mexicana*. Tesis de maestría en ciencias. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional de México. Marzo 2007. [consultado en: marzo 2008] Disponible en:
<<http://www.cs.cinvestav.mx/Estudiantes/TesisGraduados/2007/tesisHectorJavierA.pdf>>
33. Vrsalovic, Vinko. *PHP/Mapscript By Example*. 2005. [consultado en: abril 2008]
Disponible en: <<http://mapserver.gis.umn.edu/docs/howto/phpmapscript-byexample>>
34. McKenna, Jeff. *PHP/Mapscript Class Reference*. 2005. [consultado en: abril 2008]
Disponible en: <<http://mapserver.gis.umn.edu/docs/reference/phpmapscript-class/classes>>
35. Aguayo González, Rafael. *Diccionario de Términos SIG*. 1994 [consultado en: abril

- 2008] Disponible en:
<http://www.fcnym.unlp.edu.ar/catedras/geofoto/geo_html/informacion/pdf/diccionario_sig.pdf>
36. Palau García, Juan. *INTRODUCCIÓN AL SISTEMA DE INFORMACIÓN GEOGRÁFICO*. Septiembre 2005. [consultado en: abril 2008]. Disponible en:
<<http://www.protecproyectos.com/Documents/sig.pdf>>
37. Directorio de Servicios Web UCI. *Lineamientos de Arquitectura*. Mayo 2007. [consultado en: marzo 2008] Disponible en: <<http://uddi.uci.cu/files/arquitectura.2007.5.9.pdf>>

ANEXO I ESTÁNDARES ISO TC 211

19101 Información geográfica – Modelo de Referencia

Alcance: El modelo de referencia describe el entorno para el que tiene lugar la estandarización de la información geográfica, los principios básicos que sigue y la arquitectura de la misma. Define y relaciona todos los conceptos y componentes necesarios para dicha estandarización. Está estructurado dentro de los estándares de la tecnología de la información y es independiente de cualquier tecnología, metodología y aplicación.

19102 Información Geográfica – Generalidades

Alcance: Brinda una visión general de la familia de estándares ISO/TC 211.

19103 Información geográfica – Lenguaje de esquema conceptual

Alcance: La adopción de un lenguaje de esquema conceptual (CSL) para ser usado en el desarrollo del esquema conceptual en el campo de la información geográfica.

19104 Información geográfica - Terminología

Alcance: Un conjunto armonizado de todos los términos específicos que relacionan la familia ISO/TC 211 de estándares.

19105 (15046-5) Información geográfica – Conformidad y prueba.

Alcance: El esquema, los conceptos y los métodos y criterios de prueba a ser cumplidos para reclamar conformidad con la familia de estándares ISO/TC 211.

19106 (15046-6) Información geográfica - Perfiles

Alcance: Definición de líneas directivas para definir perfiles dentro de la familia de estándares ISO/TC 211.

19107 (15046-7) Información geográfica – Esquema espacial

Alcance: La definición del esquema conceptual define las características de los tipos de objetos.

19108 (15046-8) Información geográfica – Esquema temporal

Alcance: La definición del esquema conceptual define las características temporales de los

tipos de objetos.

19109 (15046-9) Información geográfica – Reglas para Esquemas de Aplicación

Alcance: Definición de las reglas para definir un esquema de aplicación, incluyendo los principios para clasificación de objetos geográficos y sus relaciones para un esquema de aplicación.

19110 (15046-10) Información geográfica – Metodología de catalogación de elementos

Alcance: Definición de la metodología para la creación de catálogos de objetos geográficos, atributos y relaciones, y la determinación de la factibilidad de establecer un catálogo multilingüe internacional y su administración.

19111 (15046-11) Información geográfica – Referencia espacial por coordenadas

Alcance: Definición del esquema conceptual y líneas guías para describir los sistemas de referencias geodésicos. Incluye las referencias a los sistemas de referencia internacional seleccionados.

19112 (15046-12) Información geográfica – Referencia espacial por identificadores geográficos

Alcance: Definición del esquema conceptual y guías para describir sistemas de referencia espacial indirectamente (no por coordenadas).

19113 (15046-13) Información geográfica – Principios de calidad

Alcance: Definición del esquema para la calidad aplicable a los datos geográficos.

19114 (15046-14) Información geográfica – Procedimientos de evaluación de calidad

Alcance: Desarrollo de guías para los métodos de especificar/evaluar la calidad de los datos.

19115 (15046-15) Información geográfica - Metadatos

Alcance: Definición de esquema requerido para describir información geográfica y servicios.

19116 (15046-16) Información geográfica – Servicios de posicionamiento

Alcance: Definición de un protocolo estándar de interface para sistemas de posicionamiento.

19117 (15046-17) Información geográfica - Representación

Alcance: Definición de un esquema que describa la representación de la información geográfica en una forma entendible por los humanos incluyendo la metodología para

describir símbolos. Este trabajo no incluye la estandarización de la simbología cartográfica.

19118 (15046-18) Información geográfica – Codificación

Alcance: Selección de reglas de codificación compatibles con el esquema conceptual que aplica la información geográfica y definición del intercambio entre el lenguaje de esquema conceptual y las reglas de codificación.

19119 (15046-19) Información geográfica - Servicios

Alcance: Identificación y definición de las interfaces de servicios usadas para la información geográfica y definición de las relaciones para el modelo Open System Environment (OSE).

19120 (15854) Información geográfica – Estándares funcionales

Alcance: Desarrollar una taxonomía de estándares funcionales reconocidos en el campo de la información geográfica/ Geomática desarrollado en otro cuerpo de estandarización internacional o multinacional. Identificar los componentes de estos estándares reconocidos e identificar elementos que pueden ser armonizados entre estos y los estándares de ISO/TC 211. Brindar asistencia con el desarrollo de perfiles, cuando los estándares básicos de ISO/TC 211 estén disponibles, y se correspondan con estos estándares funcionales reconocidos. El desarrollo de los perfiles no está incluido en este alcance.

19120 Enmienda 1 Información geográfica – Estándares funcionales – Enmienda 1

Alcance: Identificar áreas donde el desarrollo de estándares básicos de ISO/TC 211 pueda estar influenciado o guiado por la experiencia de comunidades de estándares funcionales. Un estándar funcional es identificado como un estándar de información geográfica en uso activo dentro de la comunidad internacional.

19121 (16569) Información geográfica – Imaginería y datos matriciales

Alcance: Orientar la manera en que TC 211 debe manejar la imagería y datos raster en el campo de la información geográfica/Geomática. Identificar aquellos aspectos que han sido estandarizados en otros comités ISO y organizaciones externas que influyen o soportan el establecimiento de estándares de datos matriciales y raster. Identificar los componentes de estos estándares para que puedan ser armonizados con TC 211.

19122 (16822) Información geográfica/Geomatics – Calificación y Certificación del Personal

Alcance: Describir un sistema para la calificación y certificación del personal, por un cuerpo central independiente, en el campo de las Ciencias de la Información Geográfica/Geomática. Definir las fronteras entre Ciencia de información geográfica/Geomática y otras disciplinas y profesiones relacionadas. Especificar las tecnologías y tareas pertenecientes a la Ciencia de Información Geográfica/Geomática. Establecer un conjunto de habilidades y niveles de competencia para los equipos de profesionales, tecnólogos y trabajadores de campo. Investigar la relación entre esta iniciativa y otros procesos similares de certificación ejecutados por asociaciones profesionales existentes. Desarrollar un plan para la acreditación de instituciones y programas candidatos para la certificación de individuos y para la colaboración con otros cuerpos profesionales.

19123 (17753) Información geográfica – Esquema para geometría y funciones de coberturas

Alcance: Definición de un esquema conceptual estándar para describir las características de las coberturas (capas).

19124 (17754) Información geográfica – Imaginería y componentes de datos raster

Alcance: Estandarizar conceptos para la descripción y representación de datos matriciales e imágenes. Identificar aspectos que necesitan ser expandidos para este tipo de datos.

19125-1: Información geográfica – Acceso simple a elementos - Parte 1: Arquitectura común y 19125-2: Información geográfica – Acceso simple a elementos - Parte 2: Opción SQL

Alcance: Brindar una especificación de implementación en conformidad con el ambiente SQL. Especificar un esquema SQL que soporte almacenamiento, recuperación, búsqueda y actualización de colecciones de elementos geoespaciales. Establecer una arquitectura para la implementación de tablas de elementos.

19125-3 Información geográfica – Acceso simple a elementos - Part 3: opción COM/OLE

Alcance: Brindar una especificación de implementación en conformidad con COM/OLE. Especificar un esquema COM que soporte almacenamiento, recuperación, búsqueda y actualización de colecciones de elementos geoespaciales. Establecer una arquitectura para la implementación de tablas de elementos.

19126 Información geográfica - Perfil - Diccionario de Datos FACC

Alcance: Es un perfil basado en reglas y métodos definidos en ISO CD 19110 -Información geográfica – Metodología de catalogación de elementos. Define un Diccionario de Datos que incluye la definición de elementos y atributos, que pueden ser usados por la comunidad internacional.

19127 Información geográfica – Parámetros y códigos geodésicos

Alcance: Desarrollar una especificación técnica sobre parámetros y códigos geodésicos.

19128 Información geográfica – Interface de Servidor de Mapas Web

Alcance: Describir un Servidor Web de Mapas.

19129 Información geográfica – Datos de cobertura, rejillas e imágenes

Alcance: Estandarizar conceptos para la descripción y representación de datos de cobertura, rejillas e imágenes en el contexto de la familia ISO 19100.

19130 Información geográfica – Sensores y modelos de datos para datos matriciales y de imágenes

Alcance: Especificará un modelo de sensor que describa las propiedades físicas y geométricas de cada sensor de fotogrametría, teledetección y otros que producen datos de imágenes.

Definirá un modelo conceptual de datos que especifique, para cada tipo de sensor, el requerimiento de contenido mínimo y las relaciones entre los componentes de los contenidos para el dato crudo que ha sido medido por el sensor y brindado en un instrumento con sistema de coordenadas para hacer posible la geolocalización y el análisis de los datos.

19131 Información geográfica – Especificaciones de productos de datos

Alcance: Proveer requerimientos para especificación de productos de datos. Estas incluirán esquemas de aplicación, sistemas de referencia espacial y temporal, calidad y procesos de captura y mantenimiento de datos.

19132 Información geográfica – Estándares posibles para servicios basados en ubicaciones (LBS)

Alcance: Investigar la necesidad de los siguientes estándares de LBS:

- Formato para expresiones de ubicación (incluyendo orientación).
- Coordenadas.
- Direcciones.
- Rutas

- Expresiones de orientación (ángulo, dirección, ángulo offset).
- Formatos para la expresión de rutas.
- Secuencias de segmentos.
- Instrucciones de giros.
- Formatos y reglas para expresiones de comandos de navegación.
- Formatos para la expresión de selección por clientes de formas de comandos.
- Formats for the expression of choice by clients of forms of commands; potentially
- Formatos para la expresión de condiciones de tráfico.
- Formatos para la transferencia entre clientes y servidores de solicitudes y respuestas para cada uno de las aplicaciones anteriores.

19133 Información geográfica – Servicios de monitoreo y navegación basados en ubicaciones.

Alcance: Este estándar especificará los servicios basados en web como soporte a clientes en diferentes escenarios (móviles), como por ejemplo:

- Rutas de navegación entre dos destinos.
- Rutas como condiciones a través de la ruta, o cambios a rutas adyacentes alternativas.
- Cómo mantener una base de datos de historia en soporte de sus aplicaciones, incluyendo condiciones a lo largo de las rutas potenciales tales como el Monitoreo de tráfico.

19134 Información geográfica – Servicios Multimodales basados en ubicación para rutas y navegación

Alcance: Este estándar propuesto especificará la búsqueda encontrar rutas o navegación entre dos destinos usando dos o más modos de transportación; y calculando un conjunto de “decisiones de navegación” o comandos de seguimiento de rutas que ejecutarán la ruta en una simple red o en redes multimodales.

19135 Información geográfica – Procedimientos para el registro de items de Información Geográfica

Alcance: El desarrollo de un estándar que especifique procedimientos para ser seguidos en preparación, mantenimiento, y publicación de un registro o registros de único, no ambiguos y permanentes identificadores asignados a los items de información geográfica.

19136 Información geográfica – Lenguaje geográfico de etiquetas (GML)

Alcance: El Lenguaje geográfico de etiquetas (GML) es un XML en conformidad con ISO 19118 para el transporte y almacenamiento de información geográfica modelada de acuerdo con las propiedades espaciales y no-espaciales de elementos geográficos. Esta especificación define el esquema XML, sus mecanismos y convenciones que:

- Provean una base abierta para la definición de esquemas de aplicación geoespacial.
- Permitir perfiles que soporten subconjuntos de GML.
- Soportar la descripción de esquemas de aplicación geoespacial para dominios especializados y comunidades de información.
- Soportar el almacenamiento y transportación de esquemas de aplicación y datos.
- Incrementar la habilidad de organizaciones para compartir la información y esquemas de aplicación en GML.

19137 Información geográfica – Perfiles generalmente usados del esquema espacial y de otros esquemas similares importantes

Alcance: Desarrollar un conjunto de perfiles de esquema espacial para brindar un mínimo conjunto de elementos geométricos necesarios para una creación eficiente de esquemas de aplicación.

19138 Información geográfica – Medidas de calidad de datos

Alcance: Define un conjunto de medidas para los subelementos de calidad de los datos identificados en ISO 19113 información geográfica – Principios de calidad. SE establecerá un registro de medidas de calidad de datos, incluyendo para cada medida un código y un identificador. Estas medidas serán aplicables en la evaluación de la calidad de conjuntos de datos geográficos y en la evaluación de su adaptabilidad al propósito previsto.

19139 Información geográfica - Metadatos – Especificaciones de implementación

Alcance: Define un modelo de implementación UML que está basado en el modelo abstracto de 19115. Estos modelos serán usados en conjunción con un esquema XML que será definido para describir conjuntos de datos geográficos digitales. Esta implementación ilustra cómo brindar información acerca de la identificación, calidad, referencia espacial y la distribución de datos geográficos digitales.

19140 Información geográfica – Metadatos – Especificación de implementación

Alcance: Enmiendas técnicas a ISO 191** para lograra armonización entre sus estándares. Incluirá aspectos de consistencia, referencias cruzadas, terminología, modelo de datos y presentación.

ANEXO II DIAGRAMAS DE SECUENCIA

DIAGRAMA DE SECUENCIA escenario Crear mapa

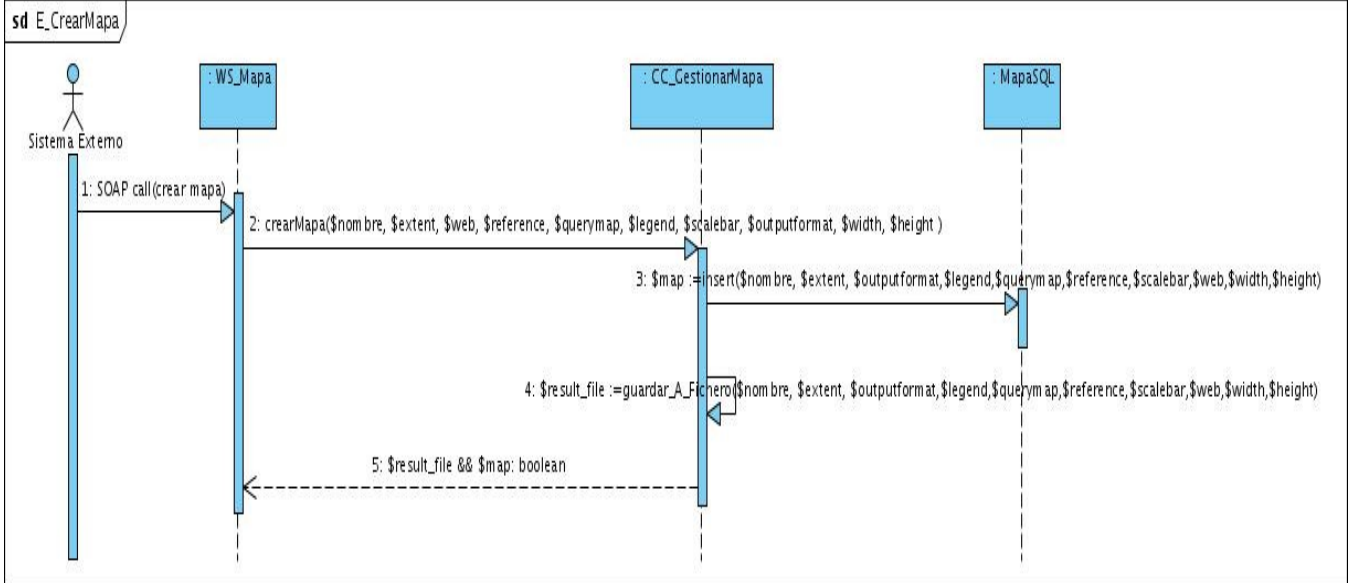
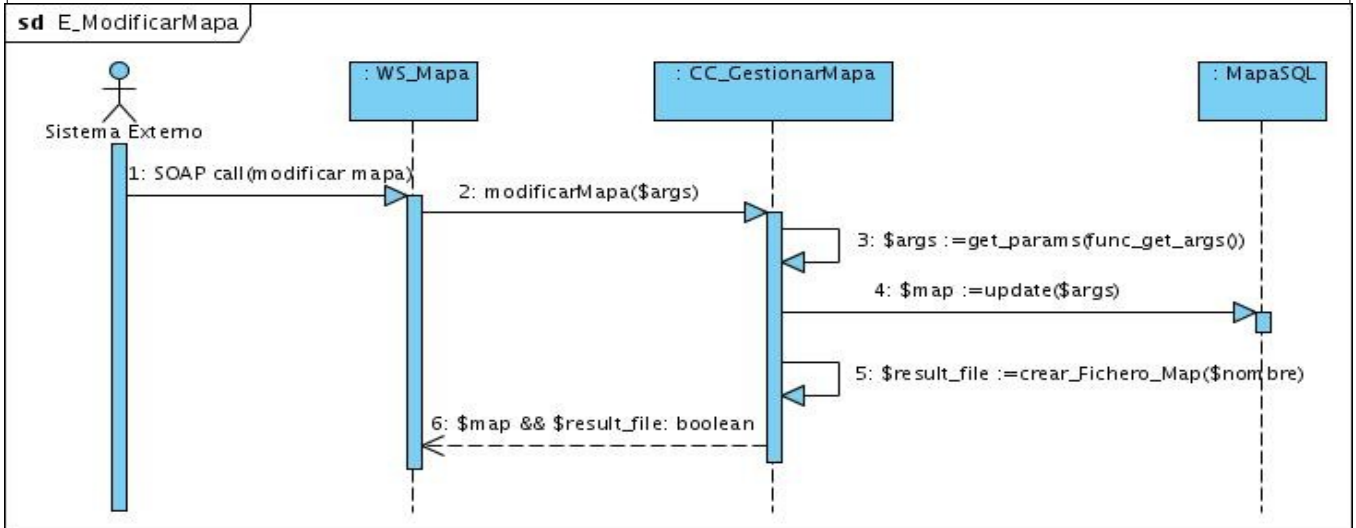
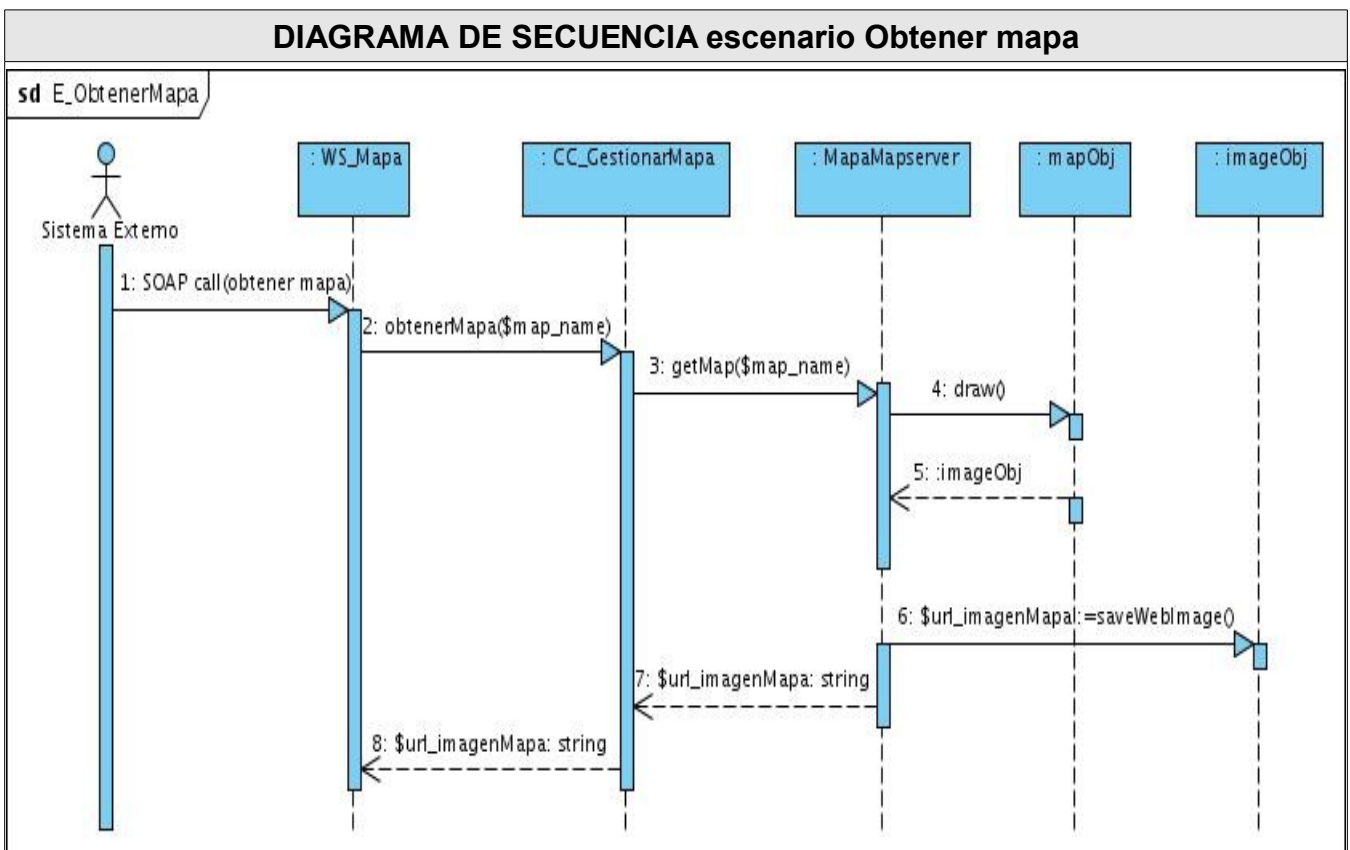
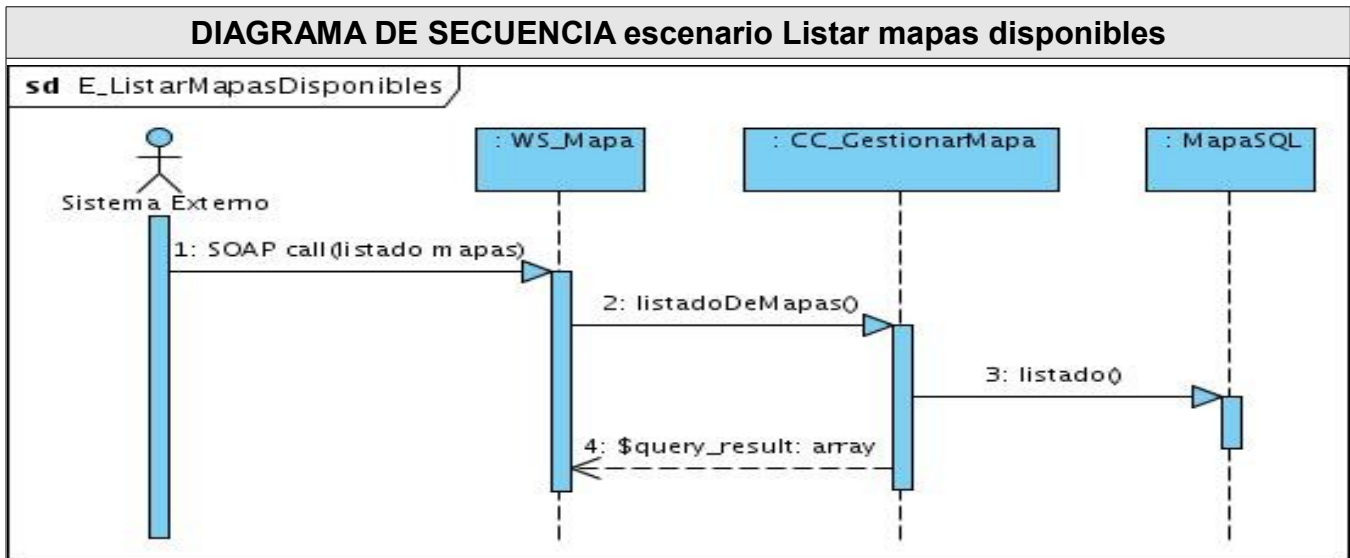


DIAGRAMA DE SECUENCIA escenario Modificar mapa





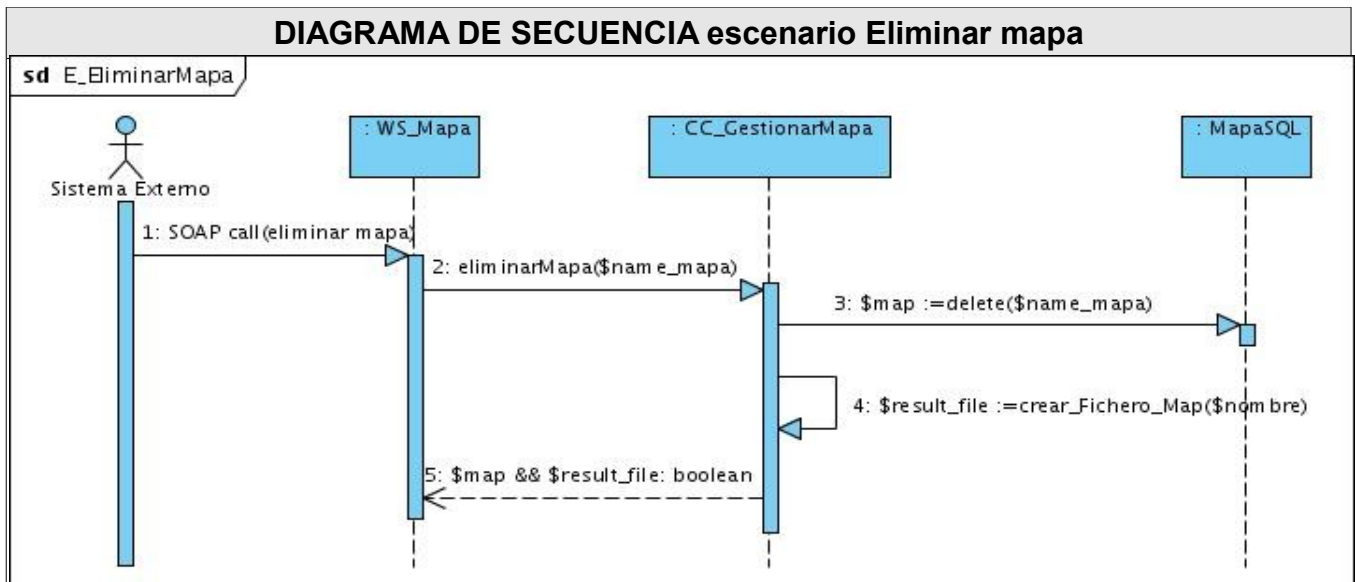
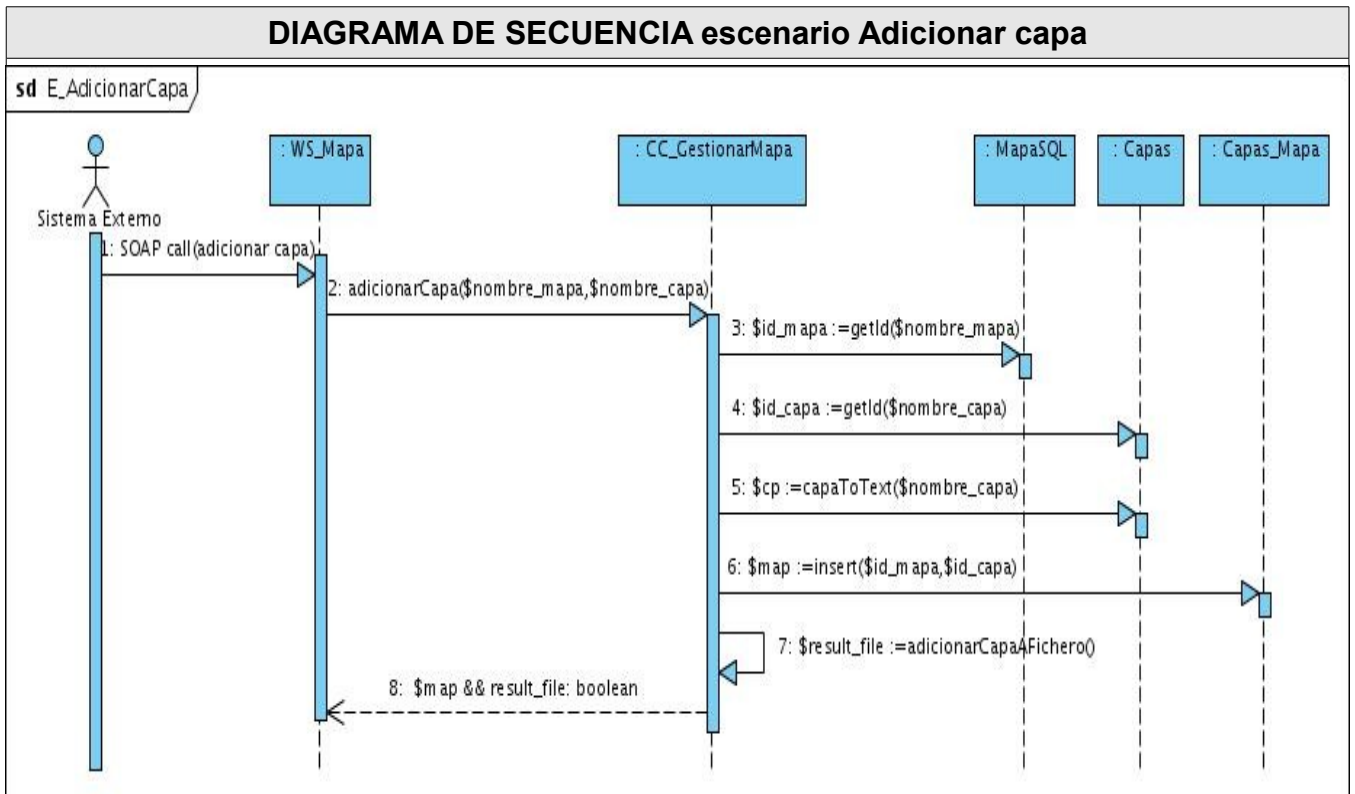


DIAGRAMA DE SECUENCIA escenario Crear capa

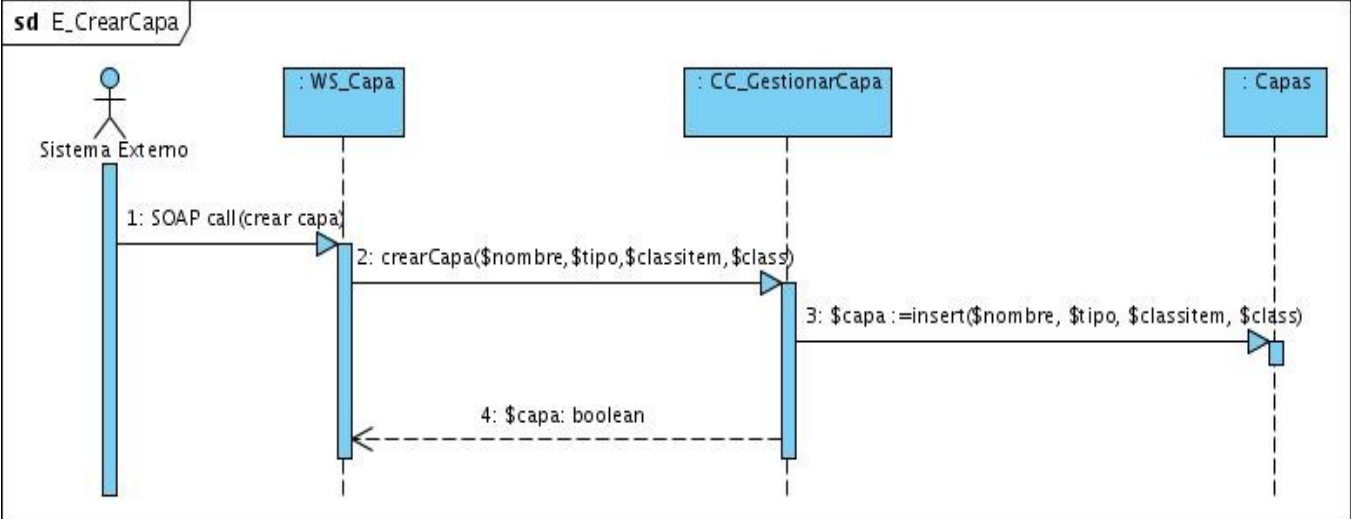
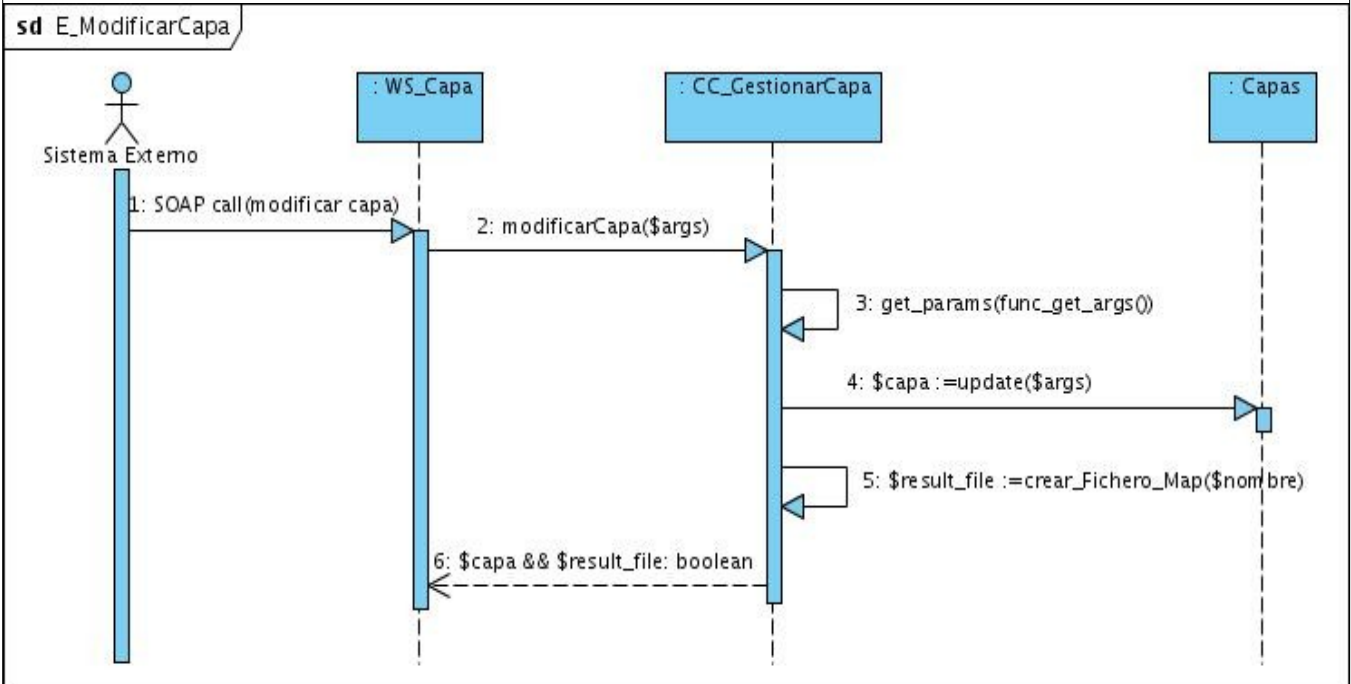


DIAGRAMA DE SECUENCIA escenario Modificar capa



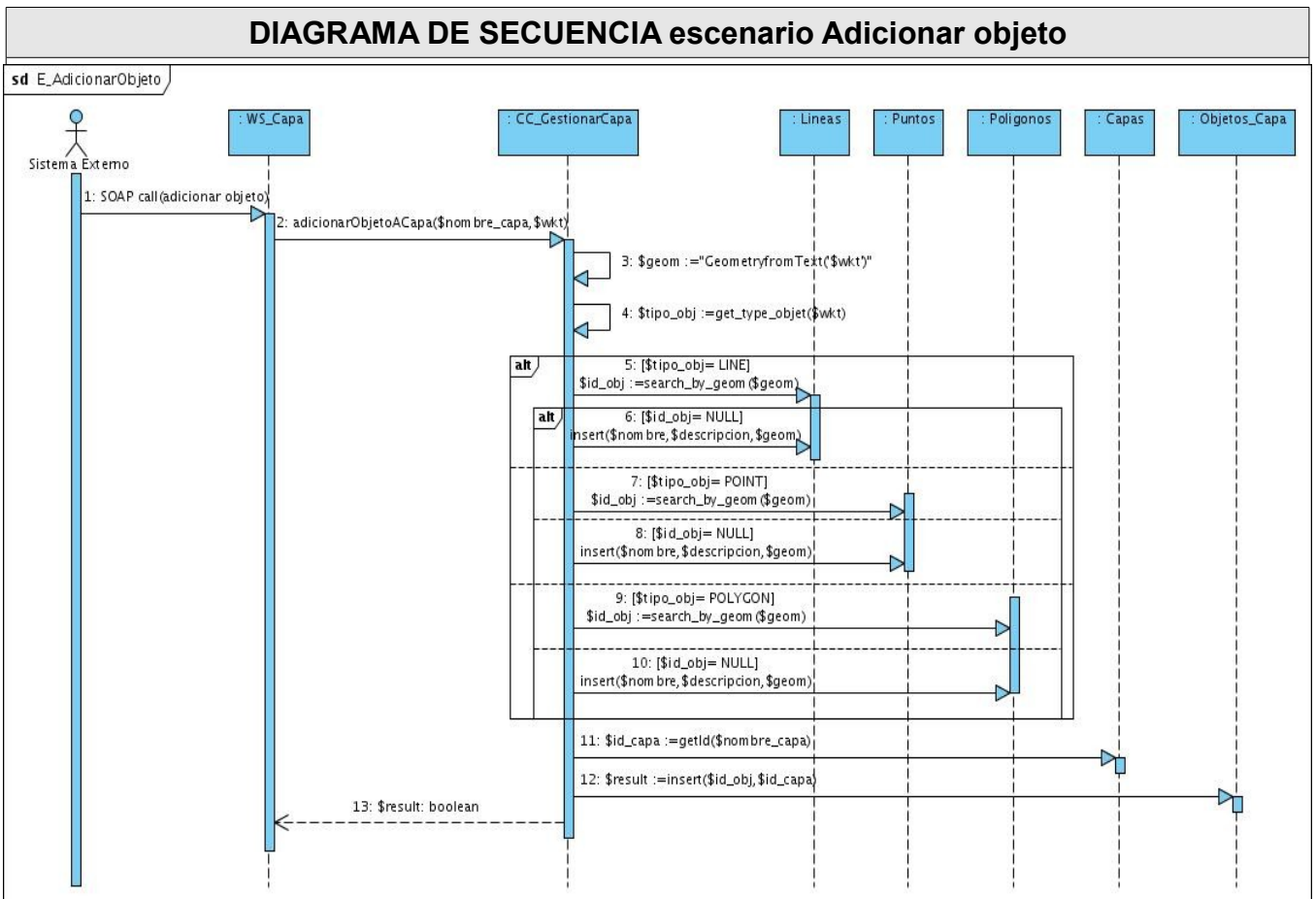
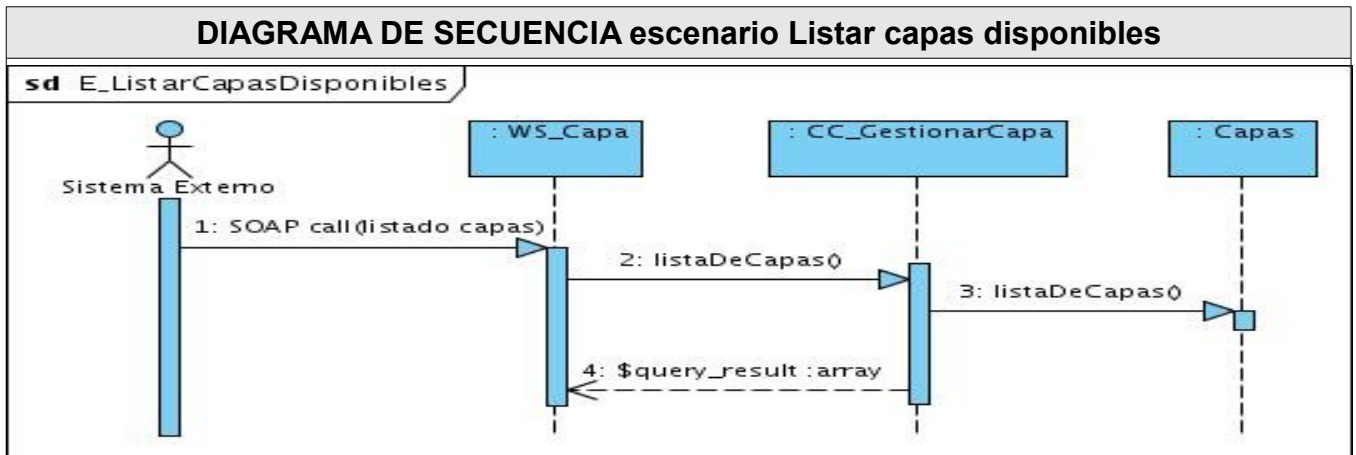


DIAGRAMA DE SECUENCIA escenario Obtener capa

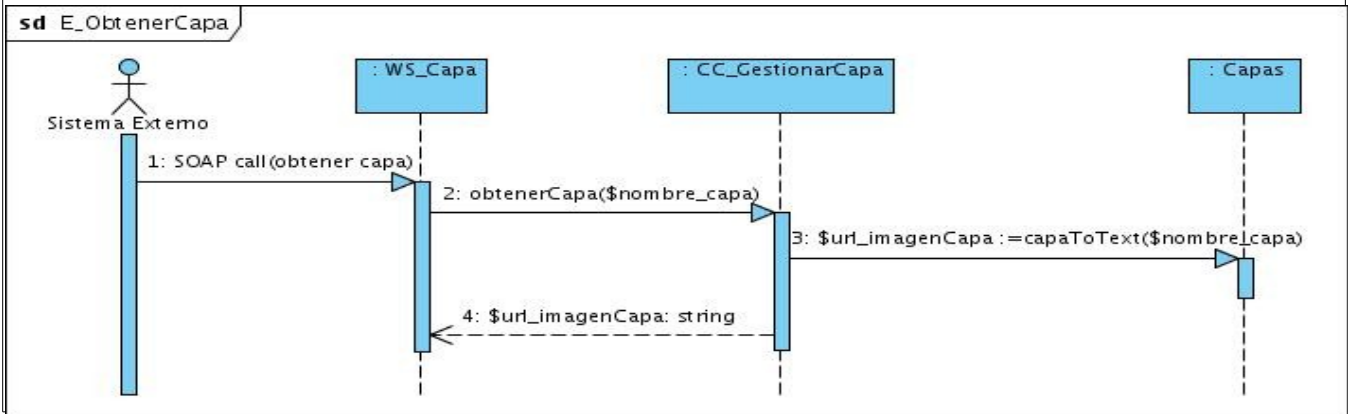


DIAGRAMA DE SECUENCIA escenario Eliminar capa

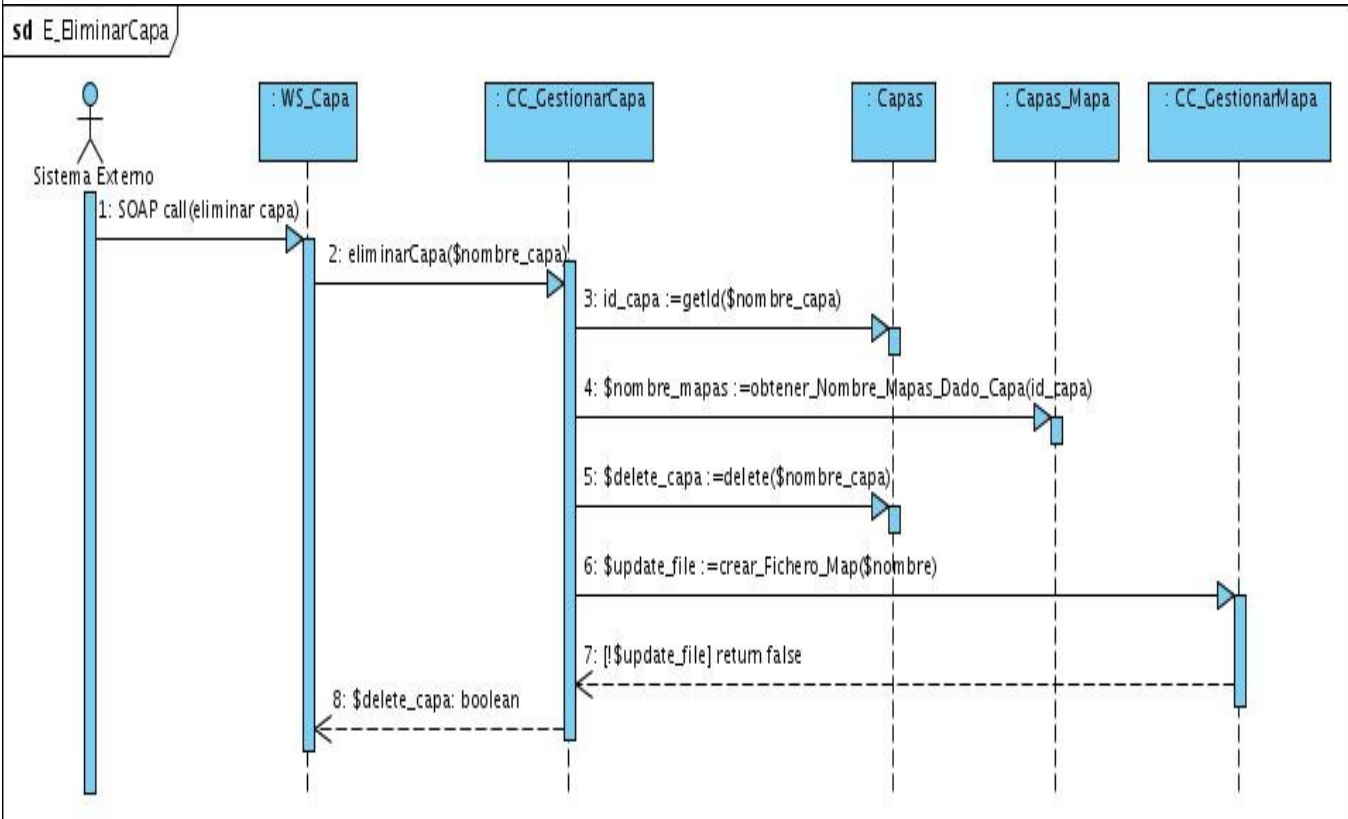


DIAGRAMA DE SECUENCIA escenario Crear objeto

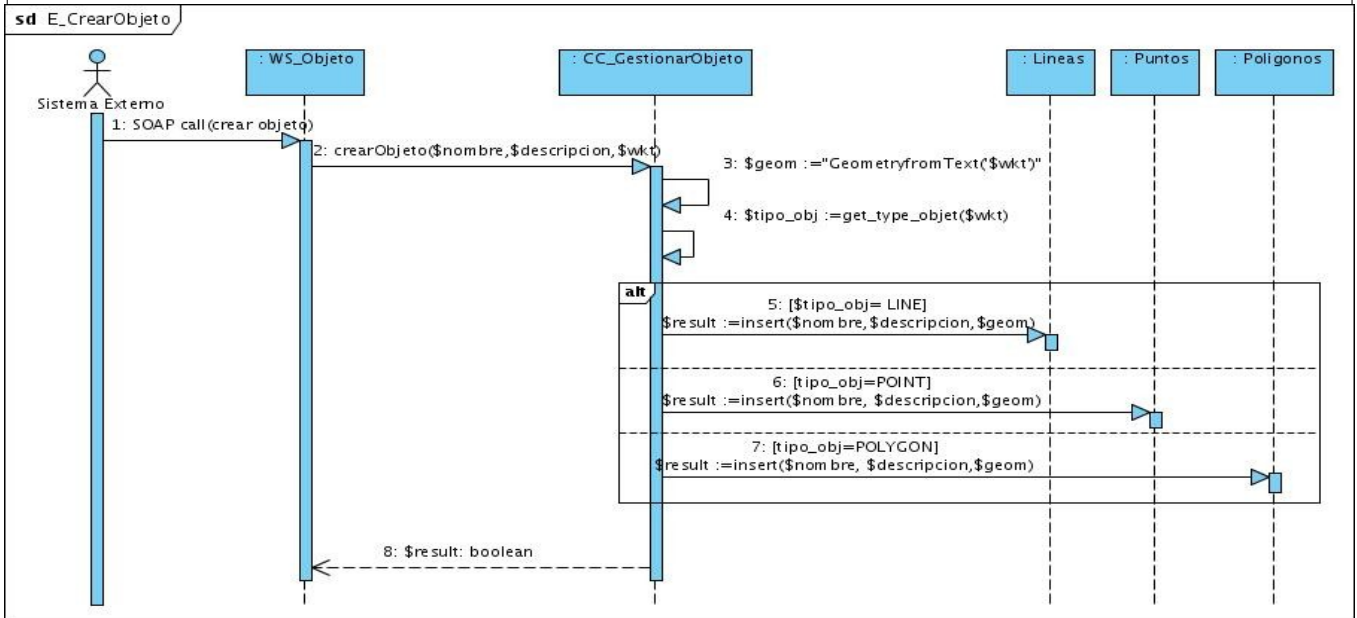
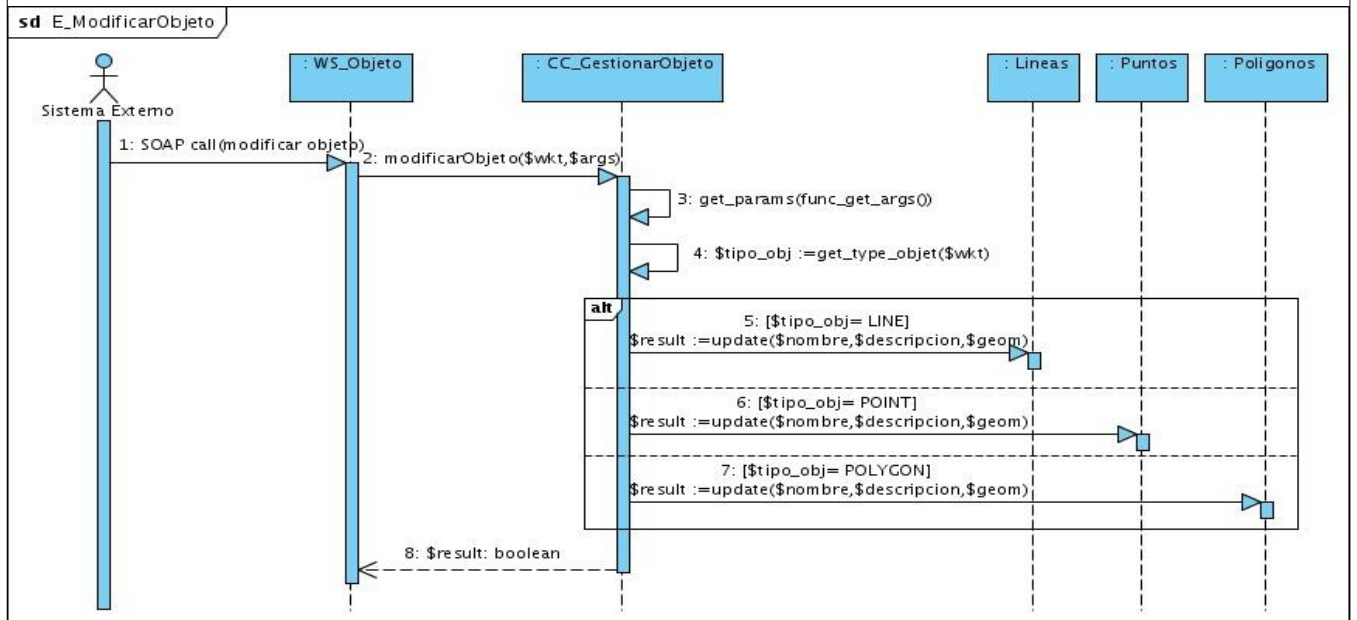
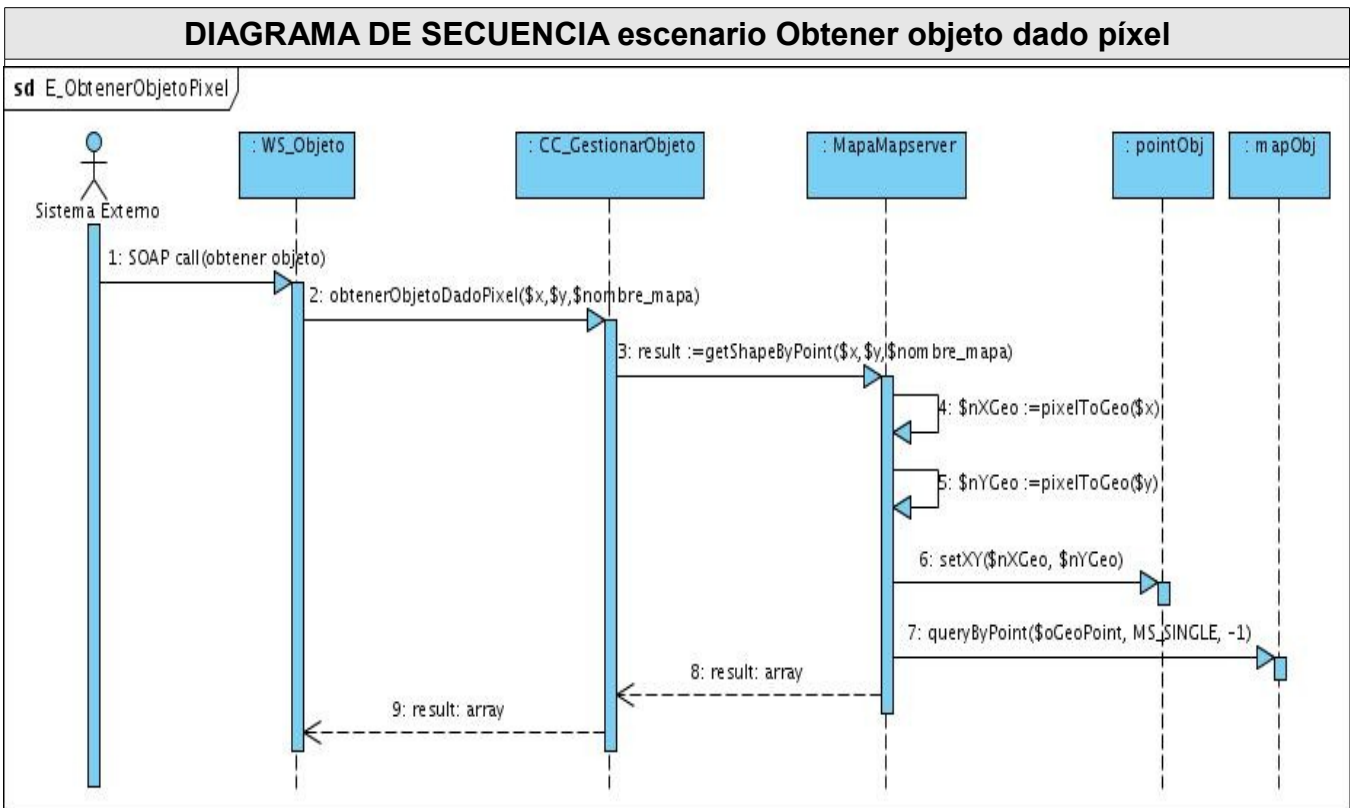
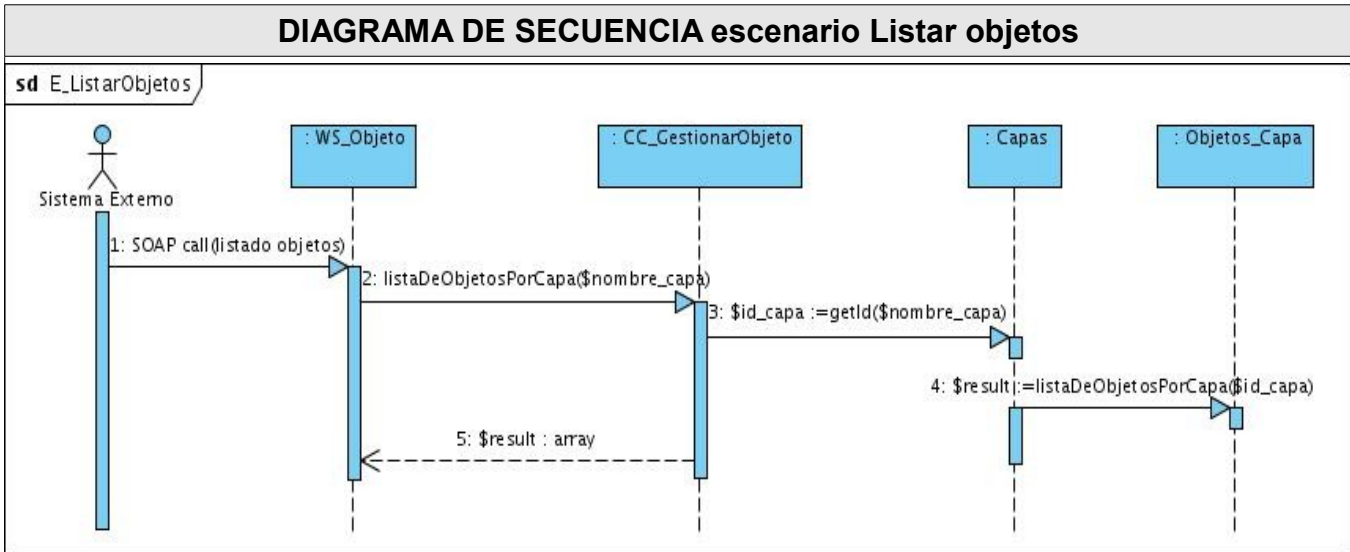
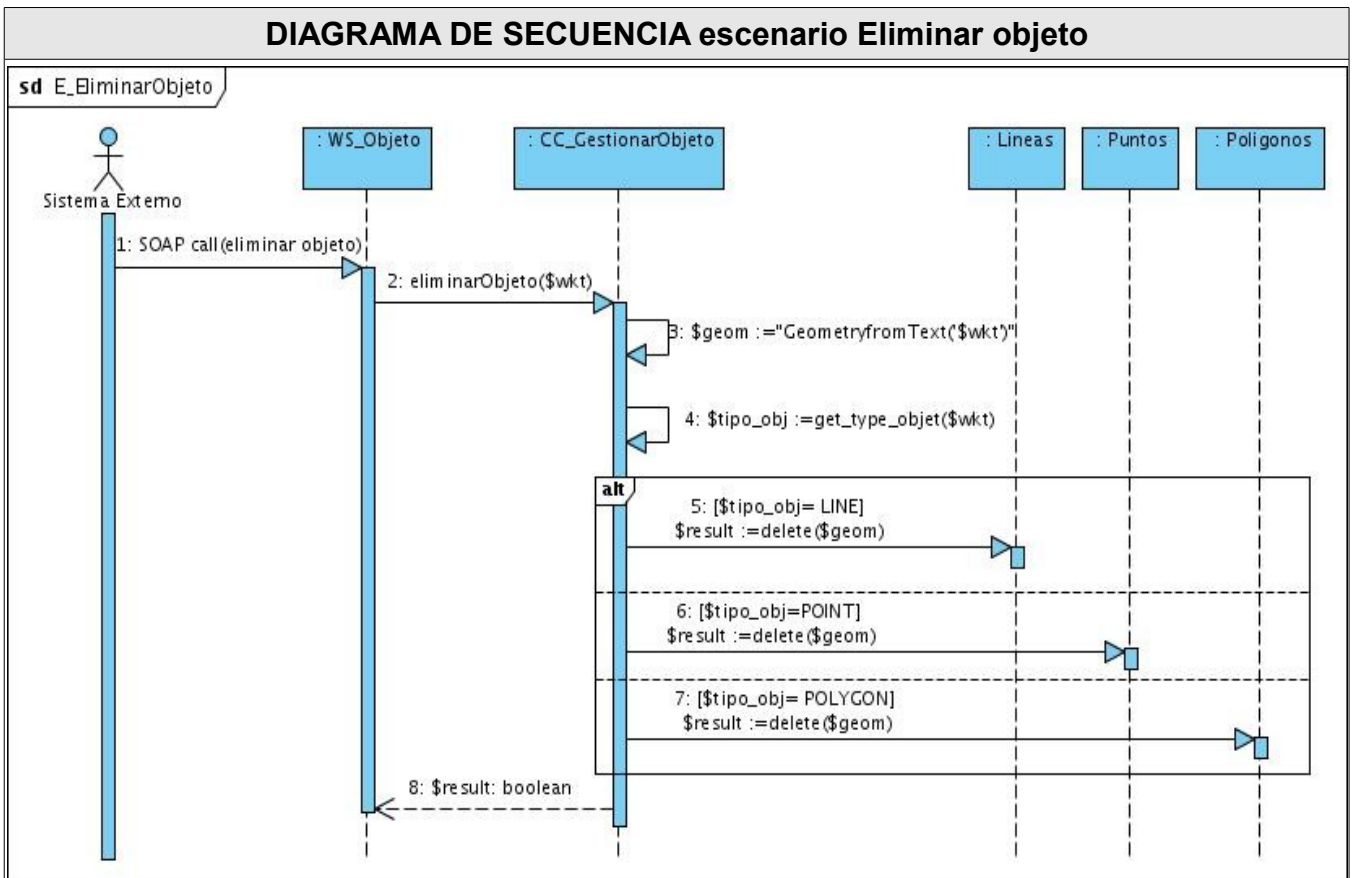
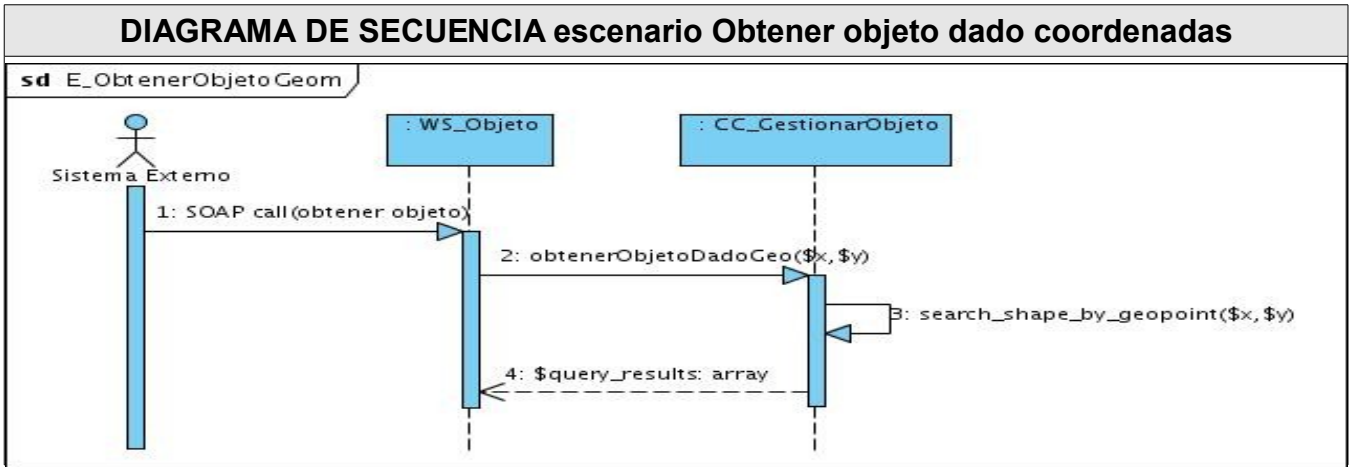
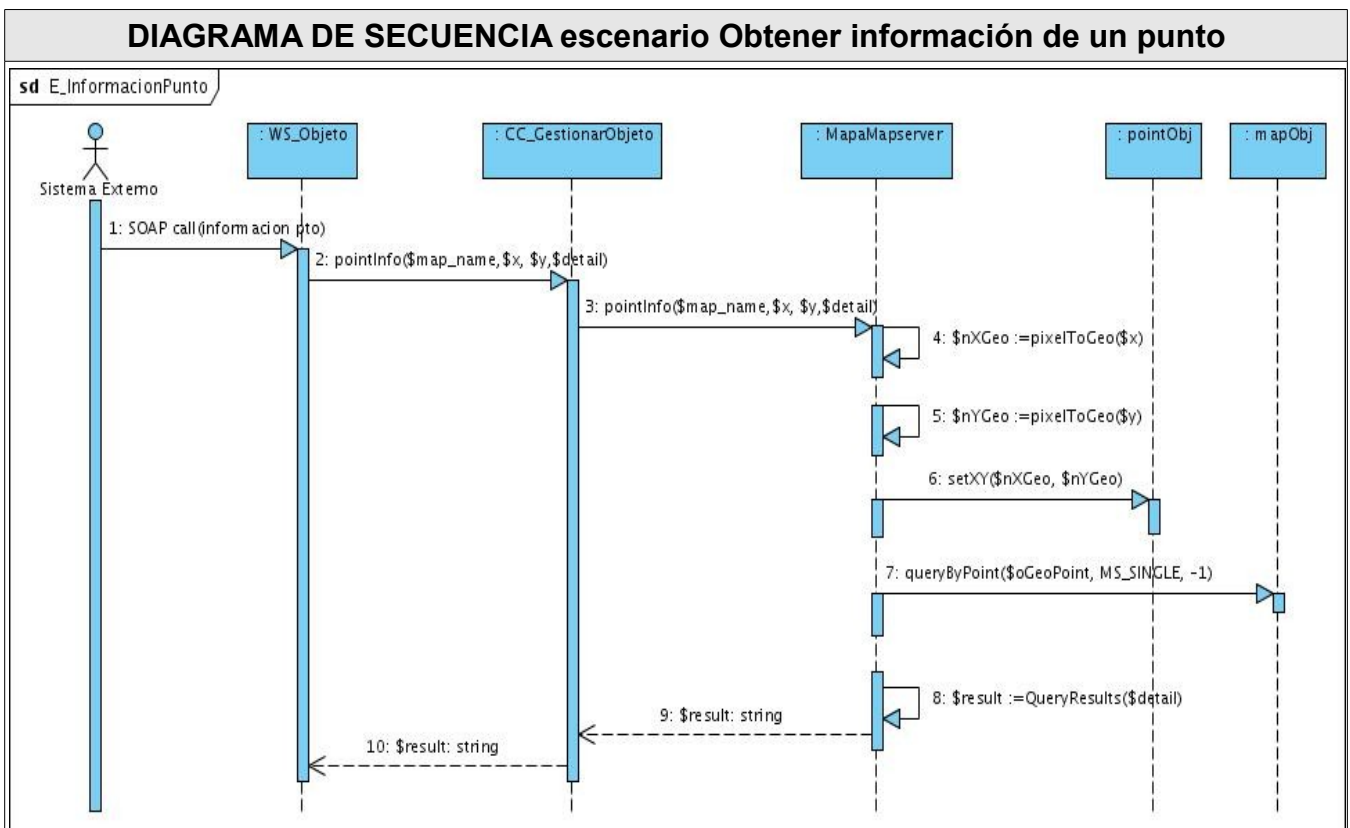
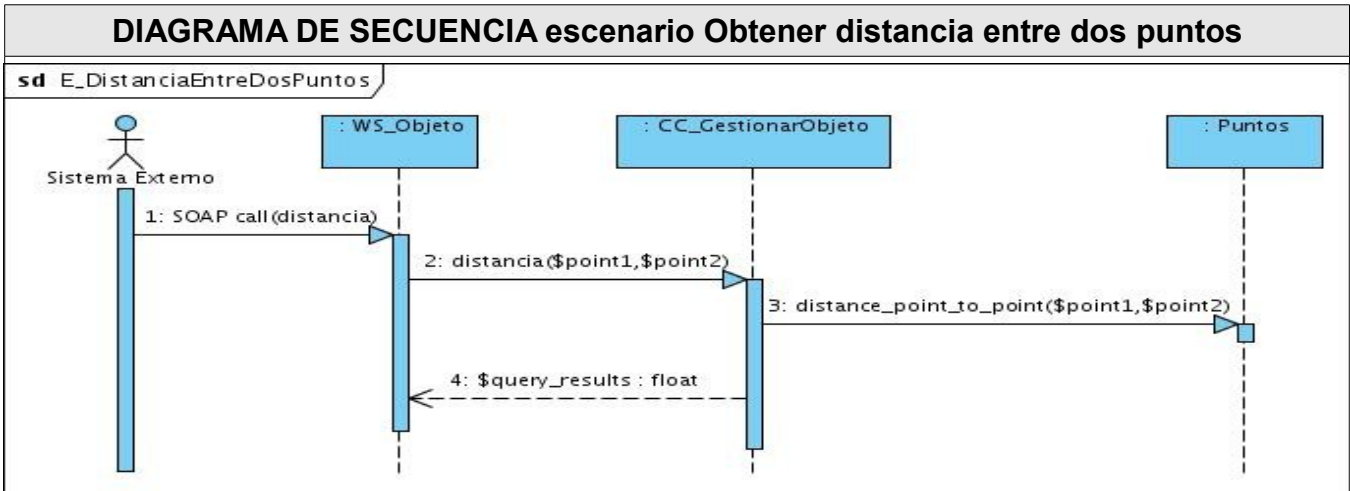


DIAGRAMA DE SECUENCIA escenario Modificar objeto









ANEXO III DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATOS

Nombre: mapa		
Descripción: Registro de los mapas		
Atributo	Tipo	Descripción
id_mapa	serial	Identificador del mapa
nombre	varchar(30)	Nombre del mapa
extent	varchar(400)	Dos coordenadas que especifican que parte del mapa se va a visualizar
web	varchar(1000)	La especificación del directorio donde se guardarán las imágenes generadas por el servidor de mapas y el alias en el servidor web para acceder a las mismas.
reference	varchar(1000)	La referencia del mapa
querymap	varchar(1000)	La forma en que se verá el mapa después de una consulta
legend	varchar(1000)	La leyenda del mapa
scalebar	varchar(1000)	La forma en que se va a mostrar la escala
outputformat	varchar(1000)	El formato de salida del mapa
width	float4	El ancho de la imagen que va a generar el servicio obtener mapa
heigth	float4	El alto de la imagen que va a generar el servicio obtener mapa

Nombre: capa		
Descripción: Registro de las capas		
Atributo	Tipo	Descripción
id_capa	serial	Identificador de la capa
nombre	varchar(20)	Nombre de la capa
type	varchar(20)	Tipo de todos los objetos que conforman la capa

classitem	varchar(100)	Muestra información adicional de los objetos
class	varchar(1000)	Define la forma en que se mostrará la capa

Nombre: objeto		
Descripción: Registro de los objetos, tabla padre		
Atributo	Tipo	Descripción
id_objeto	serial	Identificador del objeto
nombre	varchar(100)	Nombre del objeto
descripcion	varchar(100)	Descripción del objeto

Nombre: linea		
Descripción: Registro del tipo de objeto línea, hereda de la tabla objeto		
Atributo	Tipo	Descripción
id_objeto	serial	Identificador del objeto línea
nombre	varchar(100)	Nombre del objeto línea
descripcion	varchar(100)	Descripción del objeto línea
geom	geometry	Geometría del objeto línea

Nombre: punto		
Descripción: Registro del tipo de objeto punto, hereda de la tabla objeto		
Atributo	Tipo	Descripción
id_objeto	serial	Identificador del objeto punto
nombre	varchar(100)	Nombre del objeto punto
descripcion	varchar(100)	Descripción del objeto punto
geom	geometry	Geometría del objeto punto

Nombre: poligono		
Descripción: Registro del tipo de objeto polígono, hereda de la tabla objeto		

Atributo	Tipo	Descripción
id_objeto	serial	Identificador del objeto polígono
nombre	varchar(100)	Nombre del objeto polígono
descripcion	varchar(100)	Descripción del objeto polígono
geom	geometry	Geometría del objeto polígono

GLOSARIO DE TÉRMINOS Y ACRÓNIMOS

Geomática: Es la ciencia y tecnología que trata de la recopilación, análisis, interpretación, distribución, uso y almacenamiento de la información geográfica.

GIS: (Geographic Information Systems, Sistemas de Información Geográfica), son sistemas informáticos que se usan para la entrada, manejo (almacenamiento y recuperación de información), manipulación, análisis y representación de datos geográficos.

Herramientas CASE: (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador), son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software, reduciendo el coste de las mismas en términos de tiempo y de dinero.

HTTP: (Hiper-Text Transfert Protocol, Protocolo de Transferencia de Hipertexto), es usado para solicitar y recibir contenidos tales como ficheros, imágenes, flujos binarios, etc.

Información geoespacial: Es la información que describe objetos del mundo real en términos de su posición y geometría con respecto a un sistema de coordenadas, sus atributos y sus relaciones espaciales con otros objetos.

Interoperabilidad: Condición mediante la cual sistemas heterogéneos pueden intercambiar procesos o datos.

ISO/TC 211: El Comité Técnico 211 de la Organización Internacional de Estandarización es el encargado de desarrollar estándares globales en el campo de la información geográfica y la

Geomática.

Licencia GPL: GNU Public License, creada por la Free Software Foundation y orientada principalmente a los términos de distribución, modificación y uso de software, su propósito es declarar que el software cubierto por esta licencia es software libre.

MapServer: Entorno de desarrollo en código abierto para la creación de aplicaciones SIG en Internet/Intranet con el fin de visualizar, consultar y analizar información geográfica a través de la red mediante la tecnología Internet Map Server (IMS).

OGC: (Open Geospatial Consortium) consorcio de especificaciones encargado de definir los estándares a seguir por los SIG. Es un consorcio internacional formado por 256 empresas, organismos estatales y universidades, que participan en un proceso para el desarrollo de especificaciones de interfaces disponibles para el público en general.

OpenGIS: Consorcio sin ánimo de lucro formado por organizaciones públicas y privadas creado en 1994 y cuyo fin es la definición de estándares abiertos e interoperables dentro de los Sistemas de Información Geográfica. 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.

OpenSource: Cualidad de algunos softwares de incluir el código fuente en la distribución del programa. En general se usa para referirse al software libre.

PostgreSQL: Es un motor de base de datos, permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.

PostGIS (Geographic Objects For PostgreSQL, Objetos Geográficos Para PostgreSQL), es una extensión al sistema de base de datos objeto-relacional PostgreSQL para su utilización en SIG.

RUP: (Rational Unified Process, Proceso Unificado de Rational) es un proceso de desarrollo de software y junto con el lenguaje unificado de modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

SOAP: (Simple Object Access Protocol, Protocolo Simple de Acceso a Datos) es un protocolo de mensajes basado en XML entre computadoras, que especifica el formato de mensaje que accede e invoca a los objetos. El mismo es un protocolo simple de comunicación para enviar mensajes del tipo RPC, con independencia del lenguaje de programación y del Sistema Operativo.

TCP/IP: (Transmission Control Protocol/Internet Protocol), protocolo de control de transmisión más comúnmente usado sobre Internet. IP(Internet Protocol), es un protocolo de capa de red que determina hacia dónde son encaminados los paquetes, en función de su dirección de destino. TCP (Transmission Control Protocol), es un protocolo de capa de transporte que se asegura de que los paquetes lleguen correctamente a su destino.

UDDI: (Universal Description, Discovery and Integration, Especificaciones Universales de Integración, Descripción y Descubrimiento), definen una forma para publicar y descubrir información sobre servicios Web. Consta de varios documentos y un XML. Permite comprobar qué servicios web están disponibles.

UML: (Unified Modeling Language, Lenguaje Unificado de Modelo) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad.

WCS: (Web Coverages Services, Servicios de Coberturas en Web) especificación para la interface o protocolo de comunicación entre un cliente y un servidor, en el que se transfiere información de rasgos geográficos (componente espacial y descriptivo) en formato raster a través de la web. Especificación emitida por OGC.

WFS: (Web Features Services, Servicios de Rasgos (características) en Web) especificación para la interface o protocolo de comunicación entre un cliente y un servidor, en el que se transfiere información vectorial de rasgos geográficos (componente espacial y descriptivo) a través de la web. Especificación emitida por OGC.

WMS: (Web Map Services, Servicios Web de Mapas) especificación para la interface o protocolo de comunicación entre un cliente y un servidor, en el que se transfieren mapas (imágenes) generados dinámicamente a partir de datos espaciales, a través de la web. Especificación emitida por OGC.

WKT: (Well-known text) es un estándar de representación de objetos geométricos vectoriales en un mapa. Este formato está regulado por el Open Geospatial Consortium (OGC).

WSDL: (Web Service Description Language, Lenguaje de Descripción de Servicios Web) es básicamente un lenguaje de definición de interfaces, basado en el estándar XML para servicios basados en redes.

XML: (eXtensible Markup Language, lenguaje de anotación extensible) es un conjunto de reglas que sirven para definir etiquetas semánticas para organizar un documento.