

Universidad de las Ciencias Informáticas

Facultad 4



**Título: Modelación de Servicios Web para el
procesamiento y la representación de información
geoespacial.**

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

Autoras: Dinia Zayas Romero
Gerxis Sam Alcántara

Tutor: Ing. Joaquín Quintas Santiago

Junio de 2007

“Año 49 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Facultad # 4 de la Universidad de las Ciencias Informáticas para que haga el uso que estime pertinente con este trabajo.

Para que así conste firmamos la presente a los _____ días del mes de _____ del _____.

Gerxis Sam Alcántara

Dinia Zayas Romero

Ing. Joaquín Quintas Santiago



*“Cuando se es joven, se crea. Cuando se es inteligente, se produce.
No se adapta, se innova: la medianía copia; la originalidad se atreve.”*

José Martí

AGRADECIMIENTOS

A mis padres, por el cariño y el amor que me han brindado toda la vida, por ser mis principales educadores, por ser el motivo de inspiración de llevar todos estos sueños adelante y poder siempre contar con ellos.

A mis familias, por brindarme amor y apoyo incondicionalmente.

A todos los que invirtieron parte de sus días en nuestra formación profesional a lo largo de estos cinco años.

A nuestros compañeros de aula, por haber tenido la posibilidad de compartir estos años con ellos y haber conocido personas maravillosas que realmente se merecen todo mi apoyo.

A nuestro tutor, por la consagración y apoyo para que este trabajo resultara lo mejor posible.

A todas las personas que forman parte del proyecto del MINFAR que de una forma u otra nos brindaron su ayuda para que este trabajo saliera adelante.

A nuestra revolución, que nos educa y nos prepara ante la vida, con el propósito de convertirnos en personas mejores cada día.

Gerxis Sam

A Nuestro Comandante en Jefe Fidel, por idear este sueño que se convierte en realidad.

A mis padres, por guiarme con su ejemplo y ayudarme a encontrar el camino, por haber confiado en mí en todos los momentos.

A mis abuelos, por brindarme siempre su apoyo.

A mis amigos por aguantarme en mis malos momentos y reír conmigo en los buenos.

A todos los que, de una forma u otra, han dedicado una parte de su valioso tiempo para que llegáramos hasta aquí y nos han ayudado en la elaboración de este Trabajo de Diploma.

A los que confiaron, y a los que no.

Dinia Zayas Romero.

DEDICATORIA

A mis padres, quienes me han brindado amor incondicional y siempre me han impulsado a superarme profesionalmente ofreciéndome aliento constante para lograrlo.

A mi hermano para que siga estudiando y superándose para que en el futuro sea mucho mejor que yo en todos los aspectos de su vida.

Al resto de mis familiares, que me han apoyado siempre ayudadme a conseguir mis anhelos.

A nuestro Comandante en Jefe Fidel Castro Ruz por la confianza que depositó en nuestro proyecto y en nosotros como parte importante para que este sueño se hiciera realidad.

Gerxis Sam

A mis padres, Edelys y Alberto,
A Ale y a Dayna,
A mi abuelita Zoraida,
Y a todos los que, como ellos, forman parte importante de mi vida.

Dinia Zayas

RESUMEN

Debido a la necesidad de almacenar y manipular la información geográfica y realizar diversos análisis a partir de su representación para la toma de decisiones, surgieron los Sistemas de Información Geográfica (SIG). Un SIG almacena información sobre el mundo como una colección de modelos temáticos (capas) que se relacionan por la geografía. Es importante aclarar que un SIG no es un sistema automático de toma de decisiones, sino que es una herramienta para consultar y analizar datos como soporte a este proceso. A raíz de esto muchas empresas han desarrollado sus propias soluciones de software para facilitar el manejo de la información geoespacial, cada una de ellas con características específicas que las hacen más o menos convenientes para quienes las utilizan.

Actualmente en Cuba se está llevando a cabo un proceso de migración a software libre que afecta en gran medida la utilización de las aplicaciones que gestionan la información geográfica, pues las que usan nuestros especialistas, además de pertenecer al grupo de los software propietarios, basan su funcionamiento en un solo tipo de plataforma. Con este Trabajo de Diploma se pretende realizar la modelación de un conjunto de servicios web para el procesamiento y la representación de información geoespacial. Los servicios web representan una nueva aproximación de una arquitectura orientada a servicios, que permite la construcción de aplicaciones multiplataforma mediante la localización y la coordinación de los servicios disponibles en la red, con ventajas de personalización, redistribución, interoperabilidad y reutilización de mejoras de desarrollo que ofrecen estas aplicaciones.

PALABRAS CLAVE

Sistemas de Información Geográfica, SIG, Servicios Web, información geoespacial.

ÍNDICE

AGRADECIMIENTOS	II
DEDICATORIA.....	III
RESUMEN.....	IV
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1. Introducción	4
1.1. Sistemas de Información Geográficos	4
1.2. Software Libre.....	4
2. Sistemas de Información Geográfica en el mundo.....	5
2.1. Los SIG y el Software Libre	5
3. Sistemas de Información Geográfica en Cuba.....	7
4. Sistemas de Información Geográfica en la Universidad de Ciencias Informáticas (UCI).	8
5. Conceptos básicos	8
6. Metodología a utilizar	10
6.1. Lenguaje Unificado de Modelado (UML)	10
6.2. Proceso Unificado del Rational (RUP).....	10
7. Tecnologías actuales.....	12
7.1. Servicios Web	12
7.2. Arquitectura Orientada a Servicios.	16
8. Herramientas a utilizar.....	17
8.1. Visual Paradigm - UML	17
8.2. Macromedia Dreamweaver.....	17
8.3. MapServer	18

8.4.	Lenguajes de programación: PHP	18
8.5.	Lenguajes de programación: JavaScript.....	19
8.6.	Sistema Gestor de Bases de Datos: PostGIS/PostgreSQL	20
9.	Conclusiones	21
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA		22
1.	Introducción	22
2.	Propuesta de sistema	22
3.	Modelo de Dominio	23
3.1.	Definición de las entidades y conceptos principales.....	23
3.2.	Representación del modelo de dominio.....	24
4.	Especificación de los requisitos del Software	24
4.1.	Requerimientos Funcionales	24
4.2.	Requerimientos no funcionales.....	25
5.	Definición de los casos de uso	27
5.1.	Definición de los actores.....	27
5.2.	Descripción de los Casos de Uso	28
5.3.	Diagrama de Casos de Uso del Sistema.	33
5.4.	Descripción detallada de los Casos de Uso.	34
6.	Conclusiones	60
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....		61
1.	Introducción	61
2.	Modelo de análisis	61
2.1.	Diagramas de Clases de Análisis	61
3.	Modelo de Diseño.	68

3.1. Patrones de diseño	68
3.2. Diagramas de clases Web	69
3.3. Diagramas de interacción	76
3.4. Descripción de las clases.	90
4. Diseño de la Base de Datos	91
4.1. Diagramas de Clases persistentes.	91
4.2. Modelo de datos.	92
4.3. Descripción de las tablas.	92
5. Principios de diseño.	93
6. Mecanismos de diseño.	93
7. Seguridad.....	94
8. Interfaz.	95
9. Tratamiento de errores	95
10. Ayuda.....	95
11. Conclusiones	95
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS	96
1. Introducción	96
2. Diagrama de Componentes.....	96
3. Diagrama de Despliegue.	99
4. Conclusiones.....	100
CONCLUSIONES GENERALES.....	101
RECOMENDACIONES.....	102
BIBLIOGRAFÍA.....	103
GLOSARIO.....	106

INTRODUCCIÓN

Uno de los aspectos fundamentales para el proceso de toma de decisiones de una Empresa u Organismo es contar con una herramienta que les permita a sus directivos tener a su disposición en cualquier momento la mayor cantidad de información posible, así como realizar su representación de una manera sencilla y esclarecedora para lograr una acertada apreciación de una situación dada, en menor tiempo.

En la Organización se hace necesaria la representación de la información de forma geoespacial, la cual actualmente se realiza de forma manual en muchos de los casos o utilizando pequeñas aplicaciones de escritorio con las que se trabaja de forma local, lo que representa un problema, pues la utilización de estas herramientas trae consigo numerosos inconvenientes como la repetición innecesaria de datos, demoras en el flujo de información y dificultades en la distribución de la misma. Además, las aplicaciones que se utilizan, no cumplen con todos los requisitos que les son de interés a la Organización, en muchos casos sus funcionalidades se quedan muy por debajo de las necesidades reales; tampoco son multiplataforma, lo que limita su utilización al Sistema Operativo Windows. Esto, unido a que dichas aplicaciones son propietarias resulta un inconveniente, pues actualmente la Organización se encuentra llevando a cabo un proceso de migración a Software Libre.

Lo planteado hasta ahora conduce al siguiente **problema**: ¿Cómo lograr proveer a toda una infraestructura de desarrolladores de software o a otra aplicación, de una herramienta que permita la gestión, procesamiento y representación geoespacial de la información en línea en la Organización?

Con este Trabajo de Diploma se pretende diseñar un **Servicio Web** que brinde una serie de funcionalidades propias de los SIG que permitan a los desarrolladores de software crear aplicaciones que se encarguen del procesamiento y la representación de la información geoespacial, seleccionando cuál de las funcionalidades de las que se les brinda desean utilizar. Además, se espera que le permita al Organismo acceder a la información geoespacial con la que cuentan las distintas entidades a lo largo del país. De ahí la importancia del desarrollo de un sistema de este tipo, pues no existen antecedentes de ningún otro en nuestro país que aporte funcionalidades similares.

Analizando los aspectos señalados anteriormente, para el desarrollo de este trabajo se han definido como **objeto de estudio** los procesos de gestión y procesamiento de la información para su representación geoespacial en las empresas cubanas y, estos mismos procesos aplicados a la Institución representan su **campo de acción**.

Apoiados en lo expuesto anteriormente se llega a la siguiente **hipótesis**: si la Organización contara con un modelo de un software basado en la utilización de Servicios Web para el procesamiento y representación geoespacial de la información en línea, entonces se lograría implementar una herramienta que garantice mejorar el acceso a la información geoespacial y, al mismo tiempo, sirva de apoyo en el proceso de toma de decisiones.

Para lograr esto se ha trazado como **objetivo general** la modelación de un producto de software multiplataforma que permita la gestión, el procesamiento y la representación de la información de forma geoespacial; y como **objetivos específicos**:

- realizar el proceso de análisis y diseño de los servicios,
- diseñar Servicios Web accesibles a otras aplicaciones que necesiten procesar y representar información geográfica, independientemente de la plataforma que utilicen.

Tareas de la Investigación:

- Estudio de las principales prácticas aceptadas de la representación geoespacial de la información en Cuba y el Mundo.
- Estudio de los procesos gestión y representación de la información geoespacial realizados hasta el momento en la organización.
- Selección de la metodología de Análisis y Diseño de sistemas informáticos, que facilite y garantice la creación con calidad de la aplicación.
- Selección de las herramientas para llevar a cabo el proyecto y la elección de la plataforma en la que se desarrollará la aplicación.

Este documento está estructurado en cuatro capítulos:

Capítulo 1. Fundamentación Teórica: En este capítulo se realiza un análisis del estado del arte a nivel internacional, nacional y en la Universidad de las Ciencias Informáticas acerca de los SIG. 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. También se realiza una propuesta de sistema a través de la definición de los casos de uso que guiarán todo el proceso de desarrollo.

Capítulo 3. Análisis y Diseño del sistema: se exponen los diagramas de Clases del Análisis y Diseño, los diagramas de secuencia, y de clases persistentes, así como el Modelo de Datos y la descripción de forma general de las clases utilizadas.

Capítulo 4. Implementación: se presenta el modelo de implementación, o sea, los diagramas de componentes y el diagrama de despliegue.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1. Introducción

1.1. Sistemas de Información Geográficos

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 hayan convertido en un factor importante para el apoyo a la toma de decisiones. Ante esta necesidad, los Sistemas de Información Geográfica (SIG) han surgido como poderosas herramientas para la manipulación y análisis de grandes volúmenes de datos estadísticos, espaciales y temporales, que son necesarios para generar productos de información, ya sean mapas o informes, para la toma de decisiones.

Un SIG se puede definir como una tecnología de manejo de información geográfica formada por equipos electrónicos programados adecuadamente que permiten manejar una serie de datos espaciales y realizar análisis complejos con éstos, siguiendo los criterios impuestos por el equipo científico. [1]

Es precisamente la información geográfica el elemento que diferencia un SIG de cualquier otro tipo de sistema de información. Mientras otros sistemas contienen datos alfanuméricos, la base de datos de un SIG contiene, además, la delimitación espacial de cada uno de los objetos geográficos. Por ello, el SIG tiene que trabajar con ambas partes de la información a la vez: su forma definida en plano (cartografía) y sus atributos temáticos asociados (base de datos), constituyendo con todo ello una sola base de datos geográfica. [1]

1.2. 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 basado en cuatro libertades básicas:

- Libertad para ejecutar el programa para cualquier propósito.
- Libertad para estudiar cómo funciona el programa y adaptarlo a cualquier necesidad.

- Libertad para redistribuir copias y compartirlas con la comunidad.
- Libertad para mejorar el programa y compartir dichas mejoras con el público de manera que la comunidad se pueda beneficiar de ellas.

Stallman creó la Fundación por el Software Libre (FSF) y se definió el concepto de “copyleft” en contraposición al de “copyright” lo que dio lugar a la licencia GNU General Public License (GPL), que asegura las libertades de los programas distribuidos con dicha licencia. Son muchos los programas que se han creado bajo esta filosofía, incluyendo algunos SIG, y lo que anteriormente comenzó como una serie de proyectos anclados en el entorno universitario ya ha trascendido esos ámbitos para llegar incluso a las instituciones públicas y gubernamentales. [2]

2. Sistemas de Información Geográfica en el mundo

2.1. Los SIG y el Software Libre

La revolución que ha supuesto la filosofía del software libre ha llegado también a los SIG y hoy en día existen ya varios proyectos serios que pretenden ser una alternativa viable a programas comerciales como ArcView, Geomedia o IDRISI.

Se puede considerar Sistemas de Información Geográfica de sobremesa como aquellas aplicaciones SIG orientadas al trabajo normal de visualización, análisis, edición y salidas gráficas de información geográfica. Por otro lado estarían los servidores de datos geográficos como Mapserver, Geoserver o las Geodatabases como PostGIS y MySQL Spatial. [2] A algunas de estas herramientas nos referiremos más adelante.

Como una de estas aplicaciones de sobremesa podemos mencionar a **GRASS GIS**. GRASS (*Geographic Resources Analysis Support System*) fue inicialmente concebido y desarrollado por el laboratorio de investigación del cuerpo de ingenieros del ejército de los Estados Unidos (USA-CERL) para la gestión del territorio y la gestión medioambiental. GRASS comenzó a difundirse en ámbitos educativos y de instituciones públicas y se desarrollaron numerosas aplicaciones alrededor de dicho sistema, hasta que en 1999 pasó a tener licencia del tipo GNU GPL. Este SIG es uno de los que más desarrollo ha tenido, por lo

Capítulo 1: Fundamentación Teórica

que el número de herramientas y utilidades que presenta es muy elevado, pero tiene como inconveniente que está diseñado para entornos UNIX; esto, unido a su complejidad de uso y a su interfaz poco amigable, ha frenado su expansión para el público en general.[2]

Otro SIG que se ha difundido es **Jump**, que fue uno de los primeros SIG gratuitos y por lo tanto ha servido de base a otros desarrollos, tanto públicos como desarrollados por empresas de programación. Se trata de un SIG modular programado en Java y que basa su funcionalidad en módulos (*plugins*). Es posible conectarse a servidores de cartografía WMS y existen plugins para numerosos de formatos tanto de archivo como de servidores. Una de las cosas más interesantes son las herramientas de edición de que se disponen para modificar datos vectoriales, así como herramientas básicas de geoprocésamiento (zonas de influencia, uniones, etc.). Actualmente han aparecido versiones internacionalizadas y varias páginas que albergan proyectos relativos a Jump, tanto para la creación de nuevos plugins como proyectos que basándose en Jump pretenden generar nuevos programas con funcionalidades más específicas. Cabe señalar como una debilidad de Jump el hecho de que se echan en falta algunas funcionalidades básicas como por ejemplo la impresión de cartografía, cuadrículas, etc., además de que existe cierta descoordinación en la generación de versiones.[2]

SAGA (*System for Automated Geoscientific Analyses*) es un SIG que está programado en C++. Está más orientado al tratamiento de datos matriciales que a la información vectorial. Tiene sus orígenes en el Instituto Geográfico de Göttingen (Alemania). SAGA también se basa en una filosofía modular para la ampliación de funcionalidades y formatos soportados. Tiene como puntos fuertes el poseer una interfaz gráfica realmente madura y conseguida y la posibilidad de programación de módulos por parte del usuario. Cabe señalar que este SIG está desarrollado para Windows, no está basado en estándares del Open Geospatial Consortium (OGC), ni soporta datos en servidores. [2]

El principal problema de estas aplicaciones es que al contrario de las aplicaciones comerciales, en estos programas es posible que se echen en falta algunas funciones básicas que si nos brindan las comerciales. Pero queda claro que teniendo en cuenta la situación de bloqueo económico que se vive en nuestro país, es necesario prescindir de muchos productos de software a nivel mundial por la prohibición de ventas de licencias a Cuba.

3. Sistemas de Información Geográfica 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 la aparición de los Sistemas de Información Geográfica. Las expectativas creadas sobre SIG están también presentes en Cuba con sus correspondientes limitaciones y paradojas.

A pesar de que existen SIG creados y desarrollados por empresas españolas, holandesas, brasileñas, etc., que podrían adquirirse con sus licencias, una gran parte de los especialistas cubanos prefieren los SIG desarrollados por compañías de EE.UU. [3] Es conveniente aclarar que todos los SIG diseñados y creados en EEUU, adquiridos en Cuba por distintas vías con la desventaja de que estos no tienen licencia de utilización de usuario debido a que las compañías estadounidenses no están autorizadas a vender este software a los cubanos.

Ante la imposibilidad de obtener SIG comerciales, en Cuba se crea el Sistema de Información Geográfica de Cuba (SIGC), pero ante la existencia de potentes Sistemas de Información Geográfica el primer SIG cubano quedó obsoleto. En la década de los noventa llegó a Cuba otro SIG de formato raster, cuando se inició el auge de las computadoras personales. El IDRISI, en su primera versión para D.O.S, fue utilizado por algunos pocos especialistas del Instituto de Ecología y Sistemática (Academia de Ciencias de Cuba). Ya en el año 2005 este SIG era poco empleado en nuestro país. [3] En esa misma época la Facultad de Geografía de la Universidad de La Habana recibió la donación de un SIG llamado ILWIS, desarrollado en Holanda. En el presente, el ILWIS es utilizado por unos pocos especialistas cubanos que han recibido adiestramiento en el ITC (Institute for Aerospace Survey and Earth Sciences) de Holanda y que incluso tienen sus licencias de operación. [3]

Cuba recibió además otros SIG como el ATLAS GIS, proveniente de México a mediados de los noventa. La versión 3.01 del Atlas GIS para Windows (AGISW) también fue una donación de científicos foráneos amigos aunque se utilizó realmente poco pues ya a mediados de los noventa se introdujo y se diseminó rápidamente entre los especialistas el MapInfo, que no es un SIG potente, pero es muy amigable y relativamente fácil de manejar.[3]

En la década del noventa los cubanos deseaban tener su propio Sistema de Información Geográfica; de esta forma surgió el TELEMAT, el único SIG diseñado, creado y producido en Cuba por especialistas del antiguo Instituto Cubano de Hidrografía, una institución que hoy forma parte de GEOCUBA. En su momento fue muy utilizado, fue un logro de los técnicos cubanos, pero con el inconveniente de que requería una llave de usuario para su operación y además era poco compatible con otros SIG, por lo que ya no se trabaja con él. Actualmente los cubanos ya utilizan el ArcGIS 8.1, que al parecer ha tomado lo mejor del AtlasGIS y del ArcView para lograr un mejor y potente Sistema de Información Geográfica. [3]

4. Sistemas de Información Geográfica en la Universidad de Ciencias Informáticas (UCI).

Como hemos podido ver, el desarrollo de los SIG a escala mundial es cada vez mayor, y la Universidad de la Ciencias informáticas no ha dejado pasar por alto este hecho. 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 tres los Trabajos de Diploma que sobre este tema se han presentado: *Sistema de Información Geográfica sobre WEB de la UCI*, *Sistema de Información Geográfica de la UCI basado en tecnología OpenSource*, y *Sistema de Información Geográfico de la UCI. Gráficos Estadísticos y Mapas Temáticos*. Actualmente se encuentra está en funcionamiento una aplicación web que permite realizar localizaciones y brinda la información que haya disponible sobre el lugar localizado por el usuario.

5. Conceptos básicos

En un Sistema de Información Geográfico la información puede ser almacenada en formato raster o vectorial. El **modelo de SIG raster** o de retícula se centra en las propiedades del espacio más que en la precisión de la localización. Divide el espacio en celdas regulares donde cada una de ellas representa un único valor. Mientras mayores sean las dimensiones de las celdas (resolución) menor es la precisión o detalle en la representación del espacio geográfico. Una imagen rasterizada, también llamada bitmap, es una estructura o fichero de datos que representan generalmente una rejilla rectangular de pixeles o puntos

Capítulo 1: Fundamentación Teórica

de color, denominada raster, en un monitor de ordenador, papel u otro dispositivo de representación. El color de cada pixel está definido individualmente.

Los modelos lógicos menos complejos son los basados en el modelo conceptual raster, en buena medida porque la geo-referenciación y la topología son implícitas a la posición (columna y fila) del píxel en la malla. La separación entre datos cartográficos y datos temáticos no existe, pues cada capa representa un único tema y cada celda contiene un único dato numérico. La malla de píxeles puede ser regular o también irregular.

En el caso del **modelo de SIG vectorial**, el interés de las representaciones se centra en la precisión de localización de los elementos sobre el espacio. Para modelar digitalmente las entidades del mundo real se utilizan tres objetos espaciales: el punto, la línea y el polígono. En el modelo vectorial los objetos son representados por los puntos y líneas que definen sus límites. La posición de cada objeto se define por su ubicación en un mapa espacial que se organiza por un sistema de referencia coordinado.

En cuanto a los **datos** que se manejan existen dos tipos: los **datos gráficos** y los **datos alfanuméricos**. Los datos gráficos son descripciones digitales de las entidades del plano. Suelen incluir las coordenadas, reglas y símbolos que definen los elementos cartográficos en un mapa. El SIG utiliza esos datos para generar un mapa o representación gráfica en una pantalla de ordenador o bien sobre papel. Para la representación de datos gráficos se utilizan tres tipos básicos de entidades:

- **Nodos o puntos:** Es un objeto sin dimensiones que representa una unión topológica o un punto terminal y que especifica una localización geométrica; en cualquier caso, se trata de la entidad básica para representar entidades con posición pero sin dimensión (al menos a la escala escogida). En el formato vectorial se les denomina puntos.
- **Líneas:**(o arcos). Son objetos de una dimensión definidos por un nodo inicio y un nodo fin.
- **Polígonos:** (o áreas). Son objetos limitados y continuos de dos dimensiones.

Los datos alfanuméricos son descripciones de las características de las entidades gráficas. Generalmente son almacenados en formatos convencionales para este tipo de información, si bien se están comenzando

a utilizar junto con los sistemas de información geográfica sistemas de gestión documental, que gestionan estos datos como imágenes gráficas en formato raster.

6. Metodología a utilizar

Durante el desarrollo de este proyecto estaremos utilizando la metodología de desarrollo RUP, que se basa en el UML, debido a las facilidades que nos brinda y teniendo en cuenta además que la Entidad con la que se está desarrollando el proyecto se encuentra más familiarizada con esta metodología de desarrollo.

6.1. Lenguaje Unificado de Modelado (UML)

UML es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje para especificar y no un método o un proceso, utilizado para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para soportar una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar.

6.2. Proceso Unificado del Rational (RUP)

El 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. RUP es en realidad un refinamiento realizado por Rational Software del más genérico Proceso Unificado.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso.

Capítulo 1: Fundamentación Teórica

Son estas tres características las que distinguen precisamente a RUP de otras metodologías de desarrollo:

Proceso dirigido por Casos de Uso

Los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Se define un Caso de Uso como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Los Casos de Uso representan los requisitos funcionales del sistema. En RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo. Los Casos de Uso no sólo inician el proceso de desarrollo sino que proporcionan un hilo conductor, permitiendo establecer trazabilidad entre los artefactos que son generados en las diferentes actividades del proceso de desarrollo.

Proceso centrado en la arquitectura

La arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo. La arquitectura involucra los aspectos estáticos y dinámicos más significativos del sistema, está relacionada con la toma de decisiones que indican cómo tiene que ser construido el sistema y ayuda a determinar en qué orden. Además la definición de la arquitectura debe tomar en consideración elementos de calidad del sistema, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible durante todo el proceso de desarrollo. La arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados. Muchas de estas restricciones constituyen requisitos no funcionales del sistema.

Proceso iterativo e incremental

El equilibrio correcto entre los Casos de Uso y la arquitectura es algo muy parecido al equilibrio de la forma y la función en el desarrollo del producto, lo cual se consigue con el tiempo. Para esto, la estrategia

que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos. Permitiendo que el equilibrio entre Casos de Uso y arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto. Se pasa por los flujos fundamentales (Requisitos, Análisis, Diseño, Implementación y Pruebas), también existe una planificación de la iteración, un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores.

Otros aspectos a tener en cuenta a la hora de elegir esta metodología de desarrollo son, en primer lugar, el hecho de que RUP es explícito en la definición de artefactos y su trazabilidad, contempla en relación causal de los artefactos creados desde los requerimientos hasta la implementación y las pruebas lo que facilita el seguimiento y el control del proyecto; como segundo aspecto importante podemos señalar que RUP identifica claramente a los profesionales (trabajadores) involucrados en el desarrollo del software y sus responsabilidades en cada una de las actividades, indicando qué trabajador es responsable de qué artefacto en cada actividad.

7. Tecnologías actuales

7.1. Servicios Web

Los servicios web son componentes de software que permiten a los usuarios usar aplicaciones de negocio que comparten datos con otros programas modulares, utilizando la Internet. Son aplicaciones independientes de la plataforma que pueden ser fácilmente publicadas, localizadas o invocadas a través de protocolos Web estándar, como XML, WSDL, SOAP o UDDI. Los Servicios Web realizan funciones que pueden ser de todo tipo: desde simples peticiones hasta complejos procesos de negocio. Una vez puesto en marcha, otras aplicaciones pueden localizarlo e invocar el servicio. 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. [4]

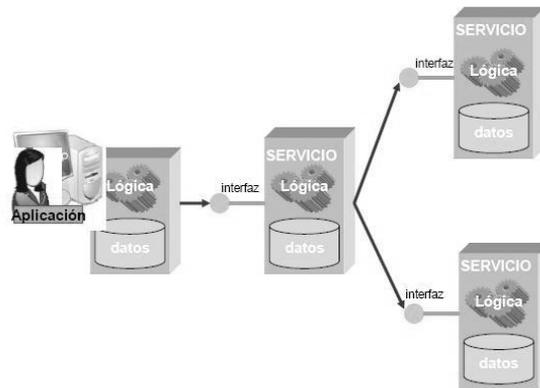


Figura 1: Sistema basado en servicios.

En la figura se representa un sistema sencillo basado en servicios. Una aplicación además de implementar sus propios componentes de negocio y datos, también puede reutilizar la funcionalidad de servicios existentes en la red Empresarial.

Un servicio debe ser una aplicación completamente autónoma e independiente. A pesar de esto, expone una interfaz de llamado basada en mensajes, capaz de ser accedida a través de la red. Generalmente los servicios incluyen tanto la lógica de negocio como los datos relevantes a la solución del problema para el cual fueron diseñados. [4]

La comunicación hacia y desde el servicio, se realiza utilizando mensajes que deben contener o referenciar toda la información necesaria para entenderlo. [4]

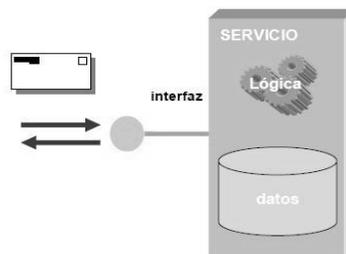


Figura 2: Visión interna de los servicios.

Ventajas de los servicios Web.

- 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.
- 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 estándar.

Protocolos que utilizan los servicios web

XML: eXtensible Markup Language o lenguaje de anotación extensible. XML no es sólo un lenguaje, es un conjunto de reglas que sirven para definir etiquetas semánticas para organizar un documento. Todo lenguaje que se exprese de una forma determinada puede ser XML. No ha nacido sólo 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. 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.

SOAP: es un protocolo basado en XML para el intercambio de mensajes sobre redes de computadoras, generalmente usando HTTP. Existen varios modelos de mensajes SOAP, pero el más común es el RPC, en donde un nodo de la red (el cliente) envía un mensaje de solicitud a otro nodo (el servidor) y este último responde al mensaje del cliente. Estos mensajes son independientes del sistema operativo y pueden transportarse en protocolos como SMTP, MIME y HTTP.

SOAP 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

Capítulo 1: Fundamentación Teórica

en raras ocasiones cooperan entre sí están ofreciendo su apoyo a este protocolo. Algunas de las mayores Compañías que soportan SOAP son Microsoft, IBM, SUN, Microsystems, SAP y Ariba.

WSDL: El lenguaje de descripción de servicios Web (WSDL, Web Service Description Language) es un dialecto basado en XML sobre el esquema que describe un servicio Web. Un documento WSDL proporciona la información necesaria al cliente para interactuar con el servicio Web. WSDL 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.

Los documentos WSDL definen los servicios como colecciones de puntos finales de red o puertos. En WSDL, la definición abstracta de puntos finales y de mensajes se separa de la instalación concreta de red o de los enlaces del formato de datos. Esto permite la reutilización de definiciones abstractas: mensajes, que son descripciones abstractas de los datos que se están intercambiando y tipos de puertos, que son colecciones abstractas de operaciones.

Describe cuatro piezas principales:

- Información sobre la interfaz exportada.
- Tipo de datos de los mensajes intercambiados (peticiones, respuestas, errores).
- Información de brindar la definición de cómo transportar los mensajes en la comunicación.
- Dirección para localizar los servidores.

UDDI: (Descripción Universal, Descubrimiento e Integración) es un registro público diseñado para almacenar de forma estructurada información sobre empresas y los servicios que estas ofrecen. A través de UDDI, se puede publicar y descubrir información de una empresa y de sus servicios. Se pueden utilizar sistemas taxonómicos estándar para clasificar estos datos y poder encontrarlos posteriormente en función de la categorización. Lo más importante es que UDDI contiene información sobre las interfaces técnicas de los servicios de una empresa. A través de un conjunto de llamadas a basadas en SOAP, se puede interactuar con UDDI tanto en tiempo de diseño como de ejecución para descubrir datos técnicos de los servicios que permitan invocarlos y utilizarlos.

7.2. 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. Al contrario de las arquitecturas orientadas a objetos, las SOAs están formadas por servicios de aplicación débilmente acoplados y altamente interoperables. 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. 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, presentamos la estructura básica de funcionamiento de un SOA tradicional.

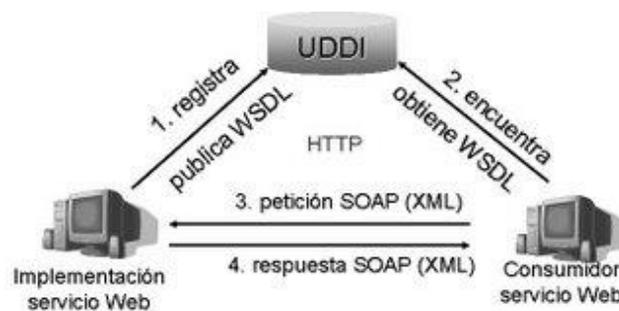


Figura 3: Arquitectura SOA tradicional.

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.

8. 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 espacial de código abierto para cubrir prácticamente cualquier necesidad.

8.1. Visual Paradigm - UML

Visual Paradigm para UML es una de las herramientas UML CASE del mercado, considerada como muy completa y fácil de usar, con soporte multiplataforma y que proporciona excelentes facilidades de interoperabilidad con otras aplicaciones. Fue creada para el ciclo vital completo del desarrollo del software que lo automatiza y acelera, permitiendo la captura de requisitos, análisis, diseño e implementación.

Visual Paradigm-uml también proporciona características 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 crear la definición de base de datos a partir del esquema de clases. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al instante, creando de manera simple toda la documentación. Está diseñada para usuarios interesados en sistemas de software de gran escala con el uso del acercamiento orientado a objetos, además apoya los estándares más recientes de las notaciones de Java y de 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 y es muy fácil de instalar y de utilizar.

8.2. Macromedia Dreamweaver

Macromedia Dreamweaver es un editor de páginas web, creado por Macromedia (actualmente Adobe Systems). Es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, por su soporte de los estándares del World Wide Web Consortium. Dreamweaver permite al usuario utilizar la mayoría de los navegadores. Con la llegada de la versión MX, Macromedia

incorporó herramientas de creación de contenido dinámico en Dreamweaver. En lo fundamental de las herramientas HTML también permite la conexión a Bases de Datos como MySQL y Microsoft Access, para filtrar y mostrar el contenido utilizando tecnología de script como, por ejemplo, ASP (Active Server Pages), ASP.NET, ColdFusion, JSP (JavaServer Pages) o PHP sin necesidad de tener experiencia previa en programación, además posee una arquitectura extensible.

8.3. MapServer

MapServer es un entorno de desarrollo en código abierto para la creación de aplicaciones SIG en Internet/Intranet con el fin de consultar y analizar información geográfica a través de la red mediante la tecnología Internet Map Server (IMS). Tiene como una de sus características principales que corre bajo plataformas Linux/Apache y Windows 95 o superior. Se considera que MapServer es el producto de software libre más maduro y popular para el desarrollo de servidores de información geográfica en la web. Cuando se utiliza en combinación con PostGIS/PostgreSQL se alcanzan resultados significativamente superiores.

8.4. Lenguajes de programación: PHP

El PHP es un lenguaje de script incrustado dentro del HTML. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. [5] Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre su soporte pueden mencionarse InterBase, mSQL, MySQL, Oracle, Informix, PostgreSQL, entre otras.

¿Por qué utilizar PHP?:

PHP no soporta directamente punteros de forma que no existen los problemas de depuración provocados por estos y permite hacer grandes cosas con pocas líneas de código. Su característica más importante es que es un lenguaje multiplataforma, funciona en todas las plataformas que soporten Apache. Es además

software libre, se puede obtener en la Web y su código está disponible bajo la licencia GPL. Este es uno de los aspectos fundamentales para nuestra elección de este lenguaje.

Posee además cuatro grandes características: [6]

- **Velocidad:** No solo la velocidad de ejecución, sino además no crear demoras en la máquina. Por esta razón no debe requerir demasiados recursos de sistema. PHP se integra muy bien junto a otro software, especialmente bajo ambientes Unix, cuando se configura como módulo de Apache, está listo para ser utilizado.
- **Estabilidad:** PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- **Seguridad:** El sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo .ini
- **Simplicidad:** Les permite a los programadores generar código productivamente en el menor tiempo posible.

8.5. Lenguajes de programación: JavaScript

JavaScript es un lenguaje de programación utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página Web. Con JavaScript podemos crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

JavaScript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras complejas de datos, etc.

8.6. Sistema Gestor de Bases de Datos: PostGIS/PostgreSQL

PostGIS es un módulo que añade soporte de objetos geográficos a la base de datos objeto-relacional PostgreSQL para su utilización en SIG. Se publica bajo la GNU GPL y puede ser usado en las plataformas Windows, GNU/Linux y Mac OS X. Actualmente hay numerosos productos informáticos que pueden utilizar PostGIS como apoyo para su base de datos, entre ellos está MapServer a quien sirve de fuente de datos mediante interfaces que trae definidas. Además permite usar todos los objetos que aparecen en las especificaciones OpenGIS (puntos, líneas, polígonos, multilíneas, multipuntos y colecciones geométricas).

PostGIS 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.

PostgreSQL es un proyecto de código abierto que está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. [7] Ofrece muchas ventajas respecto a otros sistemas de base de datos: es extensible, el código está disponible para todos sin costo lo que permite que puede ser extendido por el usuario en múltiples formas; por ejemplo, agregando nuevos tipos de datos, funciones, operadores, métodos de indexación, funciones de agregación y lenguajes procedurales; es multiplataforma y está diseñado para ambientes de alto volumen, pues usa una estrategia de almacenamiento llamada MVCC para conseguir una mejor respuesta, y además posee herramientas gráficas de diseño y de administración de Bases de Datos (ej. pgAdmin y Data Architect); debido a la licencia libre, PostgreSQL puede ser usado, modificado y distribuido libre de cargos para cualquier propósito, sea privado, comercial o académico.

9. Conclusiones

En el presente capítulo se han tratado temas relacionados con los SIG y de cómo se han desarrollado sistemas de este tipo bajo Software Libre, aunque sus funcionalidades no están desarrolladas al mismo nivel que los SIG comerciales. Se brinda además una panorámica de la utilización de los SIG en Cuba, donde cabe destacar que históricamente se han empleado los productos de software comerciales a pesar de las limitaciones que traen consigo por la necesidad de pagar licencias.

Se realiza un análisis de la metodología de desarrollo que se utilizará, los lenguajes de programación y todas las herramientas que se tendrán en cuenta para la elaboración de los servicios que estamos proponiendo. Se aborda el tema de los servicios web y los principales protocolos en que se basan, así como la arquitectura que utilizan. Todos estos aspectos son de gran importancia para el desarrollo y comprensión del sistema que se presenta.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

1. Introducción

En este capítulo realizaremos un análisis de los objetivos estratégicos de la organización y analizaremos los procesos del negocio que soportan a dichos objetivos. Estudiaremos el entorno donde se implantará el futuro sistema mediante la modelación del dominio y haremos una propuesta de sistema después de haber identificado los requerimientos que éste debe cumplir.

2. Propuesta de sistema

Como se ha visto ya en capítulos anteriores, nos hemos planteado como objeto de estudio la gestión, procesamiento y representación de la información geoespacial en las grandes empresas; y nuestro campo de acción se enmarca en estos mismos procesos específicamente dentro de la Organización en la que estamos insertados.

Uno de los objetivos generales de la Organización es la preparación del país para la defensa y la lucha armada ante una posible agresión extranjera. Teniendo en cuenta la importancia estratégica de lo mencionado anteriormente, nuestro trabajo se concentrará en una propuesta de sistema basado en **Servicios Web** que posibilite un mayor control y brinde la información geoespacial necesaria, a quienes corresponda, de los recursos materiales, financieros y humanos, y que sirva además de herramienta de apoyo en el proceso de toma de decisiones. Esta serie de funcionalidades que estaremos modelando 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 cartográfica más actualizada posible utilizando la Web, con altos niveles de disponibilidad e inmediatez.

El sistema tratará, como se ha dicho ya, con información de tipo geoespacial, es decir: se manejará principalmente el trabajo con mapas geográficos mediante la creación, modificación y realizando acciones de manera general sobre las diferentes capas que los conforman. La propuesta de sistema que modelaremos más adelante está constituida por una serie de Servicios Web que ofrecen funcionalidades que permitirán el desarrollo de aplicaciones para el manejo de este tipo de información de forma centralizada y que logre erradicar el duplicado innecesario de ésta en la Organización pues, los sistemas que existen hoy trabajan localmente y se emplea mucho tiempo y esfuerzo para distribuir dicha información a todo el país y corriendo el riesgo de que no sea la más correcta.

3. Modelo de Dominio

3.1. Definición de las entidades y conceptos principales

Concepto	Descripción
Desarrollador	Profesional de la informática que forma parte de un equipo para el desarrollo y mantenimiento de Aplicaciones de Información Geográfica.
Aplicación IG	Es una Aplicación de Información Geográfica que se encarga de gestionar este tipo de información.
Funcionalidades	Acciones que es capaz de realizar una Aplicación de Información Geográfica para gestionar este tipo de información.
Mapa	Conjunto de capas que representan de forma gráfica y métrica una porción de territorio sobre una superficie bidimensional.
Capa	Entidades gráficas organizadas en planos de visualización que contienen la información de los puntos en la pantalla (o píxeles) para su representación. Es la base de un sistema de información geográfica.
Documentación	Toda la documentación (manuales, tutoriales, etc.) que brinda información sobre el uso de una herramienta de software.
Especialista	Persona especializada en el trabajo con la cartografía que utiliza aplicaciones de información geográfica para el manejo de esta.
Departamento Automatización	Departamento que se encarga de automatizar los procesos de la organización.
Organización	Representa a las dependencias de la Institución, tales como regiones, sectores, unidades militares, sectores militares, etc.

3.2. Representación del modelo de dominio

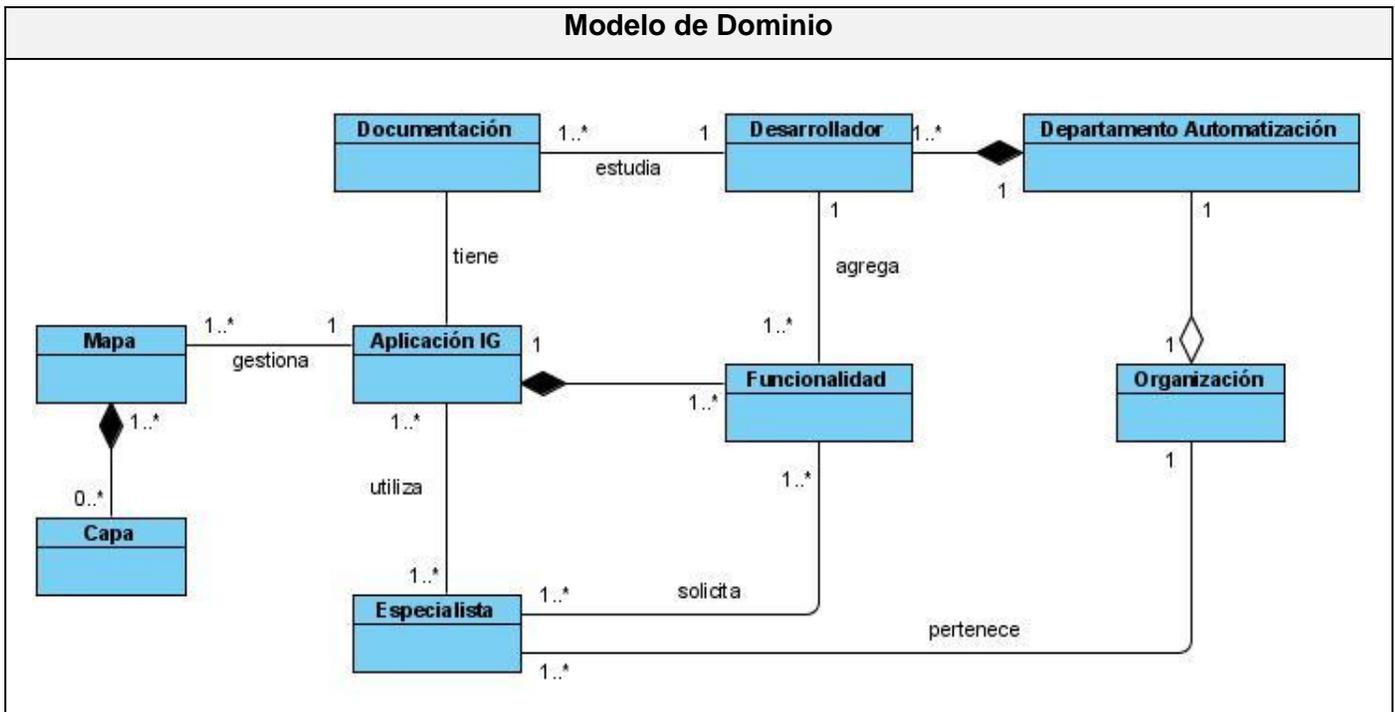


Figura 4: Modelo de Dominio

4. Especificación de los requisitos del Software

Dependencias y relaciones con otro Software: Nuestro sistema se relacionará con el servicio para el control de la Seguridad y la Auditoria, implantado ya en la organización y con el sistema de Control de la Información Geoespacial En Línea (CIGEL) que está actualmente en desarrollo.

4.1. Requerimientos Funcionales

RF1: Gestionar mapas

- 2.1 Crear un mapa nuevo
- 2.2 Eliminar un mapa existente

RF2: Gestionar capas

- 2.1 Crear una nueva capa
- 2.2 Modificar una capa
- 2.2 Eliminar una capa

RF3: Solicitar la publicación de una capa o conjunto de ellas.

RF4: Publicar capas.

RF5: Listar mapas disponibles.

RF6: Listar capas disponibles.

6.1 Buscar capas públicas.

6.2 Buscar capas personales.

RF7: Configurar Mapas.

RF8: Configurar Capas.

RF9: Ampliar mapa.

9.1 Ampliar un mapa por escala.

9.2 Ampliar un mapa por área.

9.3 Alejar un mapa.

RF10: Ampliar capa.

10.1 Ampliar un mapa por escala.

10.2 Ampliar un mapa por área.

10.3 Alejar un mapa.

RF11: Mover Mapa

RF12: Mover Capa

RF13: Abrir un mapa

RF14: Abrir una capa

RF15: Informar del estado de una capa publicas al CIGEL

RF16: Solicitar estado de Capa publicas al CIGEL

RF15: Listar capas de un Mapa

RF17: Obtener propiedades

17.1 Obtener propiedades de un mapa

17.2 Obtener propiedades de una capa

4.2. Requerimientos no funcionales

Apariencia o interfaz externa:

- Diseño sencillo, con pocas entradas, donde no sea necesario mucho entrenamiento para utilizar el sistema.

Capítulo 2: Características del Sistema

- Empleo de los colores: gris, blanco y azul principalmente, que son los definidos en los estándares del proyecto.

Usabilidad:

- El sistema podrá ser usado por cualquier desarrollador que desee utilizar las funcionalidades para la construcción de nuevas aplicaciones.
- El software tendrá siempre la posibilidad de ayuda disponible para cualquier tipo de usuario, lo que le permitirá un avance considerable en la explotación de la aplicación en todas sus funcionalidades.

Rendimiento:

- El tiempo de respuesta está dado por la cantidad de información a procesar, entre mayor cantidad de información mayor será el tiempo de procesamiento.
- Al igual que el tiempo de respuesta, la velocidad de procesamiento de la información, la actualización y la recuperación dependerá de la cantidad de información que tenga que procesar la aplicación.

Soporte: Se requiere un servidor de bases de datos con las siguientes características:

- Soporte para grandes volúmenes de datos geográficos y velocidad de procesamiento.
- Tiempo de respuesta rápido en accesos concurrentes.
- Versión de PHP 5.01.

Portabilidad:

- Necesidad de que el sistema sea multiplataforma.

Seguridad:

- Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que esté activo.
- Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- Verificación sobre acciones irreversibles (eliminaciones).

Disponibilidad:

- El sistema debe estar disponible las 24 horas del día todos los días de la semana.

Confiabilidad:

- La herramienta de implementación a utilizar debe tener soporte para recuperación ante fallos y errores.

Funcionalidad:

Capítulo 2: Características del Sistema

- Sencillez a la hora de la navegación de la aplicación, minimizando la cantidad de páginas para ejecutar todas las funciones posibles (preferentemente que estén relacionadas).

Implantación

- Entregar toda la documentación asociada al proyecto.
- Organizar el adiestramiento de los usuarios.

Software: En secciones anteriores se ha mencionado que la construcción de nuestra aplicación funcionará bajo los conceptos de arquitectura cliente/servidor. Por tanto el servidor del usuario final debe tener como requerimientos mínimos de software:

- Una computadora personal con plataforma de los sistemas operativos Linux o Windows Advanced Server 2000 o superior.
- Apache 2.0 o superior como servidor Web, con módulo PHP 5 disponible y debe estar configurado con la extensión pgsql incluida.
- PostgreSQL (PostGis) como Sistema Gestor de Base de Datos.

Hardware: Partiendo del mismo supuesto que los requerimientos de software, nuestro modelo ideal (cliente/servidor), para los requerimientos mínimos de hardware, el usuario final debe tener un servidor con las siguientes características:

Una máquina Servidora:

- 1Gb. de RAM o superior.
- 120 Gb. de disco duro o superior, también se requiere más de un microprocesador.

Una computadora que sirva de cliente:

- Pentium a 800 MHz. de velocidad de procesamiento o superior.
- Tarjeta de red de 100Mbps

Políticos-culturales: Tenemos que tener en cuenta la política a seguir en la institución y la cultura general de nuestro país puesto que este producto esta dedicado específicamente a las Fuerzas Armadas Revolucionarias, esto no implica que este producto no se pueda distribuir en otras organizaciones u organismos.

Política de la Entidad: Solo tienen acceso a la información de dicho sistema el personal autorizado.

5. Definición de los casos de uso.

5.1. Definición de los actores.

Capítulo 2: Características del Sistema

Actores	Justificación
Sistema Externo	Es la aplicación o sistema que se conecta para consumir nuestro servicio.
Administrador	Es la persona encargada de administrar todas las actividades que se realizan en la aplicación.
CIGEL	Es un sistema externo que se encarga del control de la información geográfica en línea con el cual va a interactuar nuestro sistema para el intercambio de información, mediante servicio web.

5.2. Descripción de los Casos de Uso

CU1	Publicar Capas
Actor	Administrador
Descripción	El caso de Uso comienza el usuario registrado solicita publicar y termina cuando el administrador publica la capa.
Referencia	RF4

CU2	Configurar Capas
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea personalizar o cambiar el estilo de la capa.
Referencia	RF8

CU3	Configurar Mapa
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea personalizar o cambiar el estilo del mapa.

Capítulo 2: Características del Sistema

Referencia	RF7
-------------------	-----

CU4	Gestionar Mapas
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea crear, modificar o eliminar un mapa.
Referencia	RF1

CU5	Gestionar Capas
Actor	Sistema externo
Descripción	El caso de Uso comienza cuando el actor desea crear, modificar o eliminar una capa.
Referencia	RF2

CU6	Controlar Mapas
Actor	Sistema Externo
Descripción	El caso de Uso comienza el actor desea realizar alguna acción sobre la capa como: abrir, mover, ampliar y listar los mapas.
Referencia	RF5, 9, 11, 13

CU7	Controlar Capa
Actor	Sistema Externo
Descripción	El caso de Uso comienza el actor desea realizar alguna acción sobre la capa como: abrir, mover, ampliar y listar las capas.

Capítulo 2: Características del Sistema

Referencia	RF6, 10, 12, 14
-------------------	-----------------

CU8	Obtener Propiedades
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea modificar las propiedades de la capa o del mapa.
Referencia	RF17

CU9	Solicitar Información del CIGEL
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea conocer donde están publicadas determinadas capas.
Referencia	RF16

CU10	Solicitar Publicar Capa
Actor	Sistema Externo
Descripción	El caso de Uso comienza cuando el actor desea publicar alguna capa y envía una solicitud al administrador.
Referencia	RF3

Capítulo 2: Características del Sistema

CU11	Enviar Notificación
Actor	
Descripción	El caso de Uso comienza cuando lo invoca los casos de usos Solicitar Publicar o Publicar Capa.
Referencia	RF3, 4

CU12	Informar al CIGEL
Actor	Administrador
Descripción	El caso de Uso comienza cuando el actor desea informar los cambios ocurrido sobre una capa pública, cuando se publica una capa o se deja de publicar una capa.
Referencia	RF15

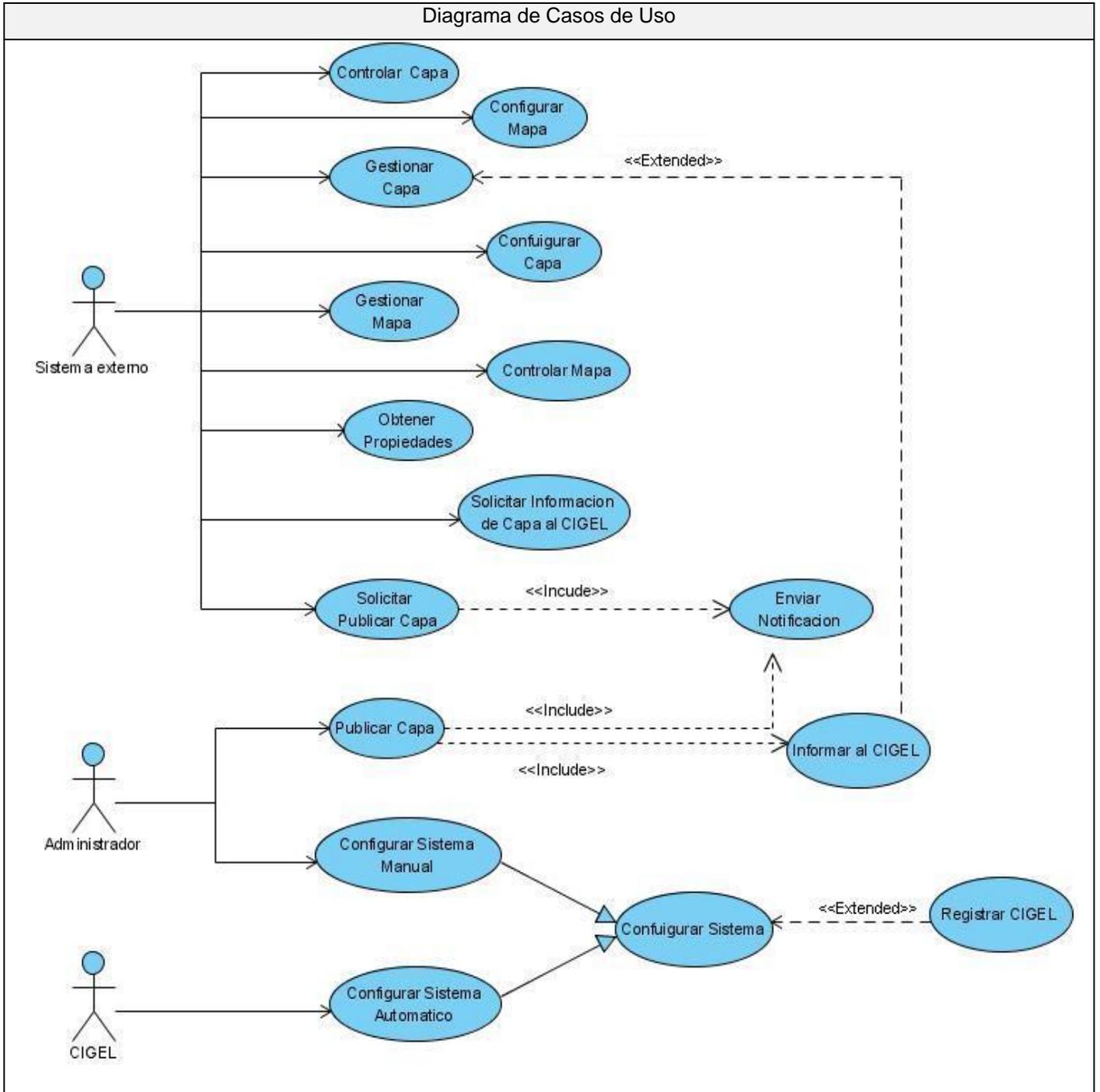
CU13	Configurar Sistema Automáticamente
Actor	Sistema CIGEL (inicia)
Descripción	El caso de uso comienza cuando el sistema CIGEL envía un mensaje informando que la dirección del servidor de configuración ha cambiado y se le asigna un nuevo servidor a nuestro sistema.
Referencia	RF:15, 16

Capítulo 2: Características del Sistema

CU14	Configurar Sistema Manualmente.
Actor	Administrador
Descripción	El caso de uso comienza cuando el administrador decide configurar el sistema para así definir los directorios donde cada usuario va a archivar sus ficheros y también para que el administrador defina, en caso de que exista conexión al CIGEL, cuál va a ser la dirección de este.
Referencia	RF: 15, 16

CU15	Registrar CIGEL
Actor	
Descripción	Este caso de uso comienza cuando se crea una nueva conexión con un CIGEL y el sistema le envía un listado de toda la cartografía que tiene disponible, el nombre y la dirección.
Referencia	RF: 15, 16

5.3. Diagrama de Casos de Uso del Sistema.



5.4. Descripción detallada de los Casos de Uso.

Caso de uso	Configurar Sistema Automáticamente	
Actores	Sistema CIGEL (inicia)	
Propósito	Establecer los parámetros de conexión al sistema CIGEL y definir un directorio para el almacenamiento de los datos de los usuarios.	
Resumen	Este caso de uso comienza cuando el Sistema CIGEL envía un mensaje informando que la dirección del servidor CIGEL ha cambiado para que el sistema establezca una nueva conexión al nuevo servidor.	
Precondiciones	Debe existir una conexión a CIGEL.	
Poscondiciones		
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF:15, 16	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
1. El sistema CIGEL envía un mensaje con una notificación informando que ha ocurrido un error en la conexión y con la nueva dirección.	2. Recibe el mensaje.	

Capítulo 2: Características del Sistema

	<p>3. Guarda la configuración.</p> <p>4. Invoca al CU Registrar</p>
--	---

Caso de uso	Configurar Capa
Actores	Sistema Externo
Propósito	Personalizar o cambiar los elementos de una capa.
Resumen	Este caso de uso comienza cuando el actor desea cambiar algún elemento de una capa. Termina cuando se actualiza la información en la Base de Datos y devuelve al usuario el resultado de la operación realizada.
Precondiciones	El usuario debe estar autenticado.
Poscondiciones	Se almacena en BD toda la información de la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	
Requisitos especiales	RF 8
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema
<p>1. El actor envía las siguientes operaciones:</p> <ul style="list-style-type: none"> • Estilo • Editable 	<p>2. El sistema realiza las operaciones enviadas:</p> <ul style="list-style-type: none"> • Estilo, ver sesión “Estilo” • Editable, ver sección “Editable”
Sección: Estilo	
<p>1. Envía la opción estilo pasándole todos los elementos de la apariencia de una capa que desee cambiar.</p>	<p>2. El sistema realiza la operación estilo, que es cambiarle la apariencia a los elementos de una capa, como por ejemplo</p>

Capítulo 2: Características del Sistema

5. Recibe la respuesta de la operación.	<p>color y tamaño.</p> <p>3. Actualiza la información.</p> <p>4. Devuelve el resultado de la operación realizada.</p>
Sección: Editable	
1. Envía la operación editar una capa.	<p>2. El sistema realiza la operación editar, que es darle permiso de escritura a la capa.</p> <p>3. Actualiza la información.</p> <p>4. Devuelve un el resultado de la operación realizada.</p>
5. Recibe la respuesta de la operación.	
Cursos alternos	
Si Ocorre un Error en el estilo o en el editable se envía un mensaje de error.	

Caso de uso	Configurar Mapa	
Actores	Sistema Externo	
Propósito	Personalizar o cambiar los elementos de un mapa.	
Resumen	En este caso de uso comienza cuando el actor desea cambiar algún elemento de un mapa. Termina cuando se actualiza la información en la Base de Datos y devuelve al usuario el resultado de la operación realizada.	
Precondiciones	El usuario debe estar autenticado.	
Poscondiciones	Se almacena en BD toda la información de la capa.	
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF7	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
1.El actor envía las siguientes operaciones: <ul style="list-style-type: none"> Zona geográfica 		

Capítulo 2: Características del Sistema

<ul style="list-style-type: none"> • Unidad de Mapa • Activar Capa • Desactivar Capa • Ordenar Capas 	<p>2. El sistema realiza las operaciones enviadas:</p> <ul style="list-style-type: none"> • Zona geográfica, ver sesión “Zona geográfica” • Unidad de Mapa, ver sesión “Unidad de Mapa” • Activar Capa, ver sesión “Activar Capa” • Desactivar Capa, ver sección “Desactivar Capa” • Ordenar Capa, ver sesión “Ordenar Capas”
Sección: Zona geográfica	
<p>1. El actor envía la operación de cambiar zona geográfica pasándole una nueva zona geográfica.</p> <p>4. Recibe la respuesta del sistema.</p>	<p>2. Realiza la operación de cambiar zona geográfica al mapa y a las capas que este tiene asociada.</p> <p>3. Envía la respuesta de la operación.</p>
Sección: Unidad de Mapa	
<p>1. El actor envía la operación de cambiar la unidad en la que se encuentra el mapa pasándole una nueva unidad.</p> <p>5. Recibe el resultado del sistema.</p>	<p>2. El sistema realiza la operación de cambiar la unidad en la que se encuentra el mapa y a las capas que este tiene asociadas.</p> <p>3. Actualiza la información.</p> <p>4. Devuelve el resultado de la operación realizada.</p>
Sección: Activar Capa	
<p>1. El actor envía la operación de activar una capa pasándole el id_capa a la que desea activar.</p>	<p>2. El sistema realiza la operación de activar la capa, mostrando los elementos que contiene esta capa en el mapa.</p>

Capítulo 2: Características del Sistema

4. Recibe el resultado del sistema.	3. Devuelve el resultado de la operación realizada.
Sección: Desactivar Capa	
1. El actor envía la operación de desactivar una capa pasándole el id_capa de la capa a la que desea desactivar.	2. El sistema realiza la operación de desactivar la capa, ocultando los elementos que contiene esta capa en el mapa.
4. Recibe el resultado del sistema.	3. Devuelve el resultado de la operación realizada.
Sección: Ordenar Capa	
1. El actor envía la operación de ordenar capas pasándole un listado de id_capa por el cual desea hacer el nuevo orden de capas.	2. El sistema realiza la operación de ordenar las capas por el orden que el usuario le pasó.
5. Recibe el resultado del sistema.	3. Actualiza la información 4. Devuelve el resultado de la operación realizada.
Cursos alternos	
Si Ocurre un Error en el proceso de Zona geográfica, Unidad de Mapa, Activar Capa, Desactivar Capa y Ordenar Capa se envía un mensaje de error.	

Caso de uso	Controlar Capa
Actores	Administrador
Propósito	Realizar algún cambio en una capa.
Resumen	En este caso de uso comienza cuando el actor desea realizar alguna acción sobre una capa. Termina cuando actualiza la información.
Precondiciones	El usuario tiene que estar autenticado.

Capítulo 2: Características del Sistema

Poscondiciones	Se actualiza la información sobre la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	
Requisitos especiales	RF 6, 8, 10, 14
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema
<p>1. El actor envía un mensaje solicitando:</p> <ul style="list-style-type: none"> • Listar Capa • Abrir Capa • Ampliar Capa • Mover Capa • Listar Capas Públicas 	<p>2. El sistema realiza las operaciones correspondientes:</p> <ul style="list-style-type: none"> • Listar Capa, ver sesión “Listar Capa” • Ampliar Capa, ver sesión “Ampliar Capa” • Ampliar Capa, ver sesión “Ampliar Capa” • Mover Capa, ver sesión “Mover Capa” • Listar Capas Públicas, ver sesión “Listar Capas Públicas”
Sección: Listar Capas	
<p>1. El actor envía la operación un listado de las capas con a las que tiene acceso.</p> <p>4. Recibe el listado de las capas a la que tiene acceso.</p>	<p>2. El sistema realiza una búsqueda en el servidor local y se comunica mediante servicio Web a los servidores CIGEL.</p> <p>3. Devuelve un listado de capas accesibles para el usuario.</p>

Sección: Abrir Capa	
1. El actor envía la operación de abrir una capa pasándole (id_capa) que desea abrir.	2. Realiza una búsqueda de las capas disponibles. 3. Abre la capa.
Sección: Ampliar Capa	
1. El actor envía la operación de ampliar una capa por escala pasándole un id_capa, y las escalas o por área pasándole un id_capa con las coordenadas (X, Y) respectivamente para el área. 4. Recibe el resultado correspondiente a la operación que fue realizada.	2. El sistema realiza la operación de ampliar por escala, ampliando la capa a la escala que se le paso o realiza ampliar capa por área, ampliando el mapa por lo parámetros que le pasaron. 3. Devuelve el resultado de la operación realizada ya sea por escala o por área.
Sección: Mover Capa	
1. El actor envía la operación de mover una capa pasándole un id_capa y las nuevas coordenadas para donde serán desplazados los elementos de la capa.	2. El sistema realiza la operación de mover una capa, moviendo los elementos de la capa hacia las coordenadas que le envió el actor. 3. Devuelve el resultado de la operación realizada.

Capítulo 2: Características del Sistema

4. Recibe el resultado del sistema.	
Sección: Listar Capa Publicas	
1. El actor envía la operación un listado de las capas públicas con a las que tiene acceso.	2. El sistema realiza una búsqueda en el servidor de todas las capas públicas que tiene y crea un listado. 3. Devuelve un listado de capas públicas.
4. Recibe el listado de las capas públicas.	
Cursos alternos	
Si Ocurre un Error en el proceso de listar, abrir, ampliar y mover una capa, se envía un mensaje de error.	

Caso de uso	Controlar Mapa	
Actores	Sistema Externo	
Propósito	Modificar o eliminar un usuario.	
Resumen	En este caso de uso comienza cuando el actor desea realizar alguna acción sobre una capa. Termina cuando el actor obtiene dicha información.	
Precondiciones		
Poscondiciones		
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF 5, 7, 9, 11, 13	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
1. El actor envía las opciones de:		

Capítulo 2: Características del Sistema

<ul style="list-style-type: none"> • Listar Mapa • Abrir Mapa • Ampliar Mapa • Mover Mapa • Obtener Capa de Mapa 	<p>2. El sistema realiza las operaciones correspondientes:</p> <ul style="list-style-type: none"> • Listar Capa, ver sección “Listar Mapa” • Ampliar Capa, ver sección “Ampliar Capa” • Ampliar Capa, ver sección “Ampliar Mapa” • Mover Capa, ver sección “Mover Mapa” • Obtener Capa de Mapa, ver sección “Obtener Capa de Mapa”
Sección: Listar Mapa	
<p>1. El actor solicita un listado de los mapas a las que tiene acceso.</p> <p>4. Recibe el listado de los mapas a los que tienes acceso.</p>	<p>2. El sistema realiza una búsqueda en el servidor local y se comunica mediante servicio Web a los servidores CIGEL.</p> <p>3. Devuelve un listado de mapas accesibles para el usuario.</p>
Sección: Abrir Mapa	
<p>1. El actor envía la operación de abrir un mapa pasándole el id_mapa del mapa que desea abrir.</p> <p>5. Recibe el resultado de la operación</p>	<p>2. Realiza una búsqueda de los mapas disponibles.</p> <p>3. Abre el mapa.</p> <p>4. Devuelve el resultado de la operación.</p>

realizada.	
Sección: Ampliar Mapa	
<p>1. El actor envía la operación de ampliar un mapa por escala pasándole el id_mapa y las escalas, o por el área pasándole el id_mapa y las coordenadas(X,Y) correspondiente al área.</p> <p>4. Recibe el resultado de la operación.</p>	<p>2. El sistema realiza la operación de ampliar por escala, ampliando el mapa a la escala que se le envió o realiza ampliar mapa por área, ampliando el mapa por lo parámetros que le mandó el actor.</p> <p>3. Devuelve el resultado de la operación realizada.</p>
Sección: Mover Mapa	
<p>1. El actor envía la operación de mover un mapa pasándole el id_mapa, y las nuevas coordenadas de la posición donde desea que el mapa esté.</p> <p>4. Recibe el resultado de la operación.</p>	<p>2. El sistema realiza la operación de mover un mapa, moviendo el mapa hacia las coordenadas que le pasaron.</p> <p>3. Devuelve los resultados de la operación realizada.</p>
Sección: Obtener Capa de Mapa	
<p>1. El actor envía la opción obtener las capas de un mapa pasándole el id_mapa.</p>	<p>2. El sistema realiza la operación buscar todas las capas que tiene ese mapa.</p>

Capítulo 2: Características del Sistema

4. Recibe el resultado de la operación.	3. Devuelve los resultados de la operación realizada.
Cursos alternos	
Si Ocurre un Error en el proceso de Listar Mapa, Abrir Mapa, Ampliar Mapa, Mover Mapa e Obtener Capa de Mapa de un mapa se envía un mensaje de error.	

Caso de uso	Enviar Notificación
Actores	
Propósito	Enviar una informar sobre la realización de una acción que puede ser a través de una solicitud o una notificación.
Resumen	En este caso de uso comienza cuando uno de los casos de uso Solicitar Publicar Capa o Publicar Capa lo ejecutan.
Precondiciones	Solicitar publicar una capa o publicar o no la capa.
Poscondiciones	Se actualiza la información sobre la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	Solicitar Publicar Capa, Publicar Capa
Requisitos especiales	RF 3, 4
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema
1. El actor se autentica.	2. Verifica que usuario esta autenticado. <ul style="list-style-type: none"> • Si es un Usuario Registrado, ver sección “Enviar notificación 1” • Si es el Administrador, ver sección “Enviar notificación 2”

Capítulo 2: Características del Sistema

Sección: Enviar notificación 1	
3. Recibe el mensaje.	<ol style="list-style-type: none"> Envía la solicitud al administrador una función del sistema. Devuelve la confirmación de la operación realizada.
Sección: Enviar notificación 2	
	<ol style="list-style-type: none"> Envía un mensaje de notificación informado al sistema externo, que fue publicada la capa mediante un mecanismo de alarmas del sistema.
Cursos alternos	
Si Ocorre un Error en el proceso de enviar la solicitud o la notificación se envía un mensaje de error.	

Caso de uso	Gestionar Capa	
Actores	Sistema Externo	
Propósito	Permitir Crear, Modificar o eliminar una capa.	
Resumen	En este caso de uso comienza cuando el actor desea crear, modificar o eliminar una capa. Termina cuando actualiza la información de la base de dato y envía el resultado al actor.	
Precondiciones	Tiene que ser un usuario registrado del sistema.	
Poscondiciones	Actualizar la BD	
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF2	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
1. El actor envía un mensaje solicitando:		

Capítulo 2: Características del Sistema

<ul style="list-style-type: none"> • Crear Capa • Modificar Capa • Eliminar Capa 	<p>2. El sistema realiza las operaciones según el mensaje:</p> <ul style="list-style-type: none"> • Crear Capa, ver sección “Crear Capa” • Modificar Capa, ver sección “Modificar Capa” • Eliminar Capa, ver sección “Eliminar Capa”
Sección: Crear Capa	
<p>1. Envía en el mensaje el nombre_mapa, el id_capa y la descripción de la capa que desea crear.</p> <p>5. Recibe el resultado de la operación realizada.</p>	<p>2. El sistema verifica que no existe esa capa y la crea.</p> <p>3. Actualiza la BD.</p> <p>4 El sistema envía el resultado de la operación realizada.</p>
Sección: Modificar Capa	
<p>1. El actor envía en el mensaje el id_capa, nombre_capa y nuevos datos de la capa que desea modificar.</p> <p>5. Recibe el resultado de la operación realizada.</p>	<p>2. El sistema verifica que existe la capa, y que esta no es pública y procede a realizar la operación de modificar los datos.</p> <p>3. Actualiza la BD.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
Sección: Eliminar Capa	
<p>1. El actor envía en el mensaje el id_capa, nombre_capa a la que desea eliminar.</p>	<p>2. El sistema realiza la operación de buscar dicha capa y verificar que no es pública, si existe la elimina.</p>

Capítulo 2: Características del Sistema

5. Recibe el resultado de la operación realizada.	<p>3. Actualiza la BD.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
Cursos alternos	
<p>Sección Crear Capa: Línea 2: Existe una capa con ese nombre el sistema devuelve un error notificando al actor.</p> <p>Sección Modificar Capa: Línea 2: Si la capa es pública ejecuta el CU Informar a CIGEL y continúa el flujo normal de los eventos.</p> <p>Sección Eliminar Capa: Línea 2: Si la capa es pública ejecuta el CU Informar a CIGEL y continúa el flujo normal de los eventos..</p>	

Caso de uso	Gestionar Mapa
Actores	Sistema Externo
Propósito	Permitir Crear, Modificar o eliminar un mapa.
Resumen	En este caso de uso comienza cuando el actor desea crear un nuevo, modificar o eliminar un mapa. Termina cuando actualiza la información de la base de datos y envía el resultado al actor.
Precondiciones	Tiene que ser un usuario registrado del sistema.
Poscondiciones	Actualiza la información.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	
Requisitos especiales	RF1
Curso normal de eventos para el caso de uso	

Acción del actor	Respuesta del sistema
<p>1. El actor envía las siguientes operaciones:</p> <ul style="list-style-type: none"> • Crear Mapa • Modificar Mapa • Eliminar Mapa 	<p>2. El sistema realiza las operaciones según haya sido el envió:</p> <ul style="list-style-type: none"> • Crear Mapa, ver sesión “Crear Mapa” • Modificar Mapa, ver sesión “Modificar Mapa” • Eliminar Mapa, ver sesión “Eliminar Mapa”
Sección: Crear Mapa	
<p>1. El actor envía la operación crear una mapa pasándole el nombre_mapa, id_mapa y los nuevos datos del mapa que desea crear.</p> <p>5. Recibe el resultado de la operación realizada.</p>	<p>2. El sistema realiza la operación de buscar si existe este mapa y si no existe crear el nuevo mapa con los datos que le pasaron.</p> <p>3. Guarda los datos en la base de dato.</p> <p>4. El sistema envía el resultado de la operación realizada.</p>
Sección: Modifica Mapa	
<p>1. Envía la operación de modificar un mapa pasándole el id_mapa, nombre_mapa y los nuevos datos del mapa que desea modificar.</p>	<p>2.El sistema realiza la operación de buscar si existe dicho mapa y si existe cambia los datos por los que le pasaron</p>

Capítulo 2: Características del Sistema

5. Recibe el resultado de la operación realizada.	4. El sistema envía el resultado de la operación realizada.
Sección: Eliminar Mapa	
1. Envía la operación de eliminar pasándole el id_mapa y nombre_mapa del mapa que desea eliminar. 4. Recibe el resultado de la operación realizada.	2. El sistema realiza la operación de buscar si existe dicho mapa y lo elimina. 3. El sistema envía el resultado de la operación realizada.
Cursos alternos	
Si Ocurre un Error en el proceso de crear, modificar o eliminar se envía un mensaje de error.	

Capítulo 2: Características del Sistema

Caso de uso	Informar Estado de Capas al CIGEL	
Actores	Administrador	
Propósito	Informar al CIGEL los cambios en una capa publica	
Resumen	En este caso de uso comienza cuando el actor desea informar al CIGEL los cambios que se realicen sobre una capa y cuando esta publica o no. Termina cuando se informa al CIGEL.	
Precondiciones	Tiene que haberse cubicado una capa o realizado cualquier cambio sobre ella.	
Poscondiciones		
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados	Publicar Capa	
Requisitos especiales	RF 15	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
<p>1. El actor envía los nuevos cambios en la localización, descripción, nombre_capa, cuando la capa ha sido publicada o ha dejarlo de serlo.</p> <p>4. Recibe mensaje de confirmación.</p>	<p>2. El sistema envía la información de la capa a otro sistema externo (CIGEL) mediante un Servicio Web. Lo que le envía específicamente es (localización, descripción, nombre_capa)</p> <p>3. El sistema envía un mensaje al actor("La operación fue realizada con éxito")</p>	
Cursos alternos		
Si Ocurre un Error en el proceso de información al CIGEL se envía un mensaje de error.		

Capítulo 2: Características del Sistema

Caso de uso	Obtener Propiedades	
Actores	Sistema Registrado	
Propósito	Obtener alguna propiedad de algún mapa o capa.	
Resumen	En este caso de uso comienza cuando el actor desea obtener información de una capa o de un mapa. Termina cuando obtiene dicha información y devuelve el resultado al actor.	
Precondiciones	El usuario debe estar autenticado.	
Poscondiciones		
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF 17	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
<p>1. El actor envía la opciones siguientes:</p> <ul style="list-style-type: none"> • Obtener propiedad de Mapa • Obtener Propiedad de Capa 	<p>2. El sistema realiza la operación enviada por el actor:</p> <ul style="list-style-type: none"> • Obtener propiedad de Mapa, ver sesión “Obtener propiedad de Mapa” • Obtener propiedad de Capa, ver sesión “Obtener propiedad de Capa” 	
Sección: Obtener Propiedad de Mapa		
<p>1. Envía un mensaje solicitando obtener las propiedades de un mapa pasándole el id mapa.</p>	<p>2. El sistema realiza la operación de obtener propiedad del mapa, que no es más que mostrar el tamaño del mapa y los permisos que tiene para determinado usuario.</p> <p>3. Devuelve el resultado de la operación realizada.</p>	

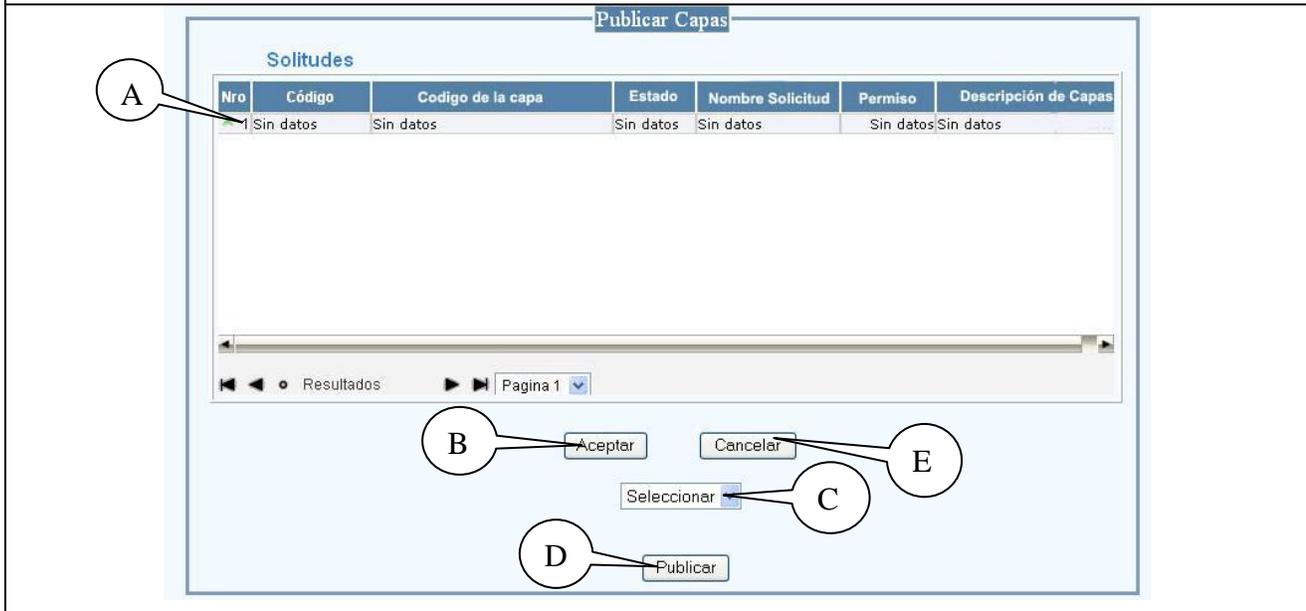
Capítulo 2: Características del Sistema

4. Recibe el resultado de la operación realizada por el sistema.	
Sección: Obtener Propiedad de Capa	
1. Envía la operación de obtener propiedad de un capa pasándole el id capa. 4. Recibe el resultado de la operación realizada por al sistema.	4. El sistema realiza la operación de obtener propiedad de la capa, que no es más que mostrar el tamaño que tiene dicha capa y los permisos que tiene para determinado usuario. 5. Devuelve el resultado de la operación realizada.
Cursos alternos	
Si Ocurre un Error en el proceso de Obtener propiedad de Mapa, Obtener Propiedad de Capa se envía un mensaje de error.	

Caso de uso	Publicar Capa
Actores	Administrador
Propósito	Publicar una capa
Resumen	Este caso de uso comienza cuando el actor acepta o no la solicitud de publicar capa. Termina cuando se envía una notificación al actor informándole lo ocurrido.
Precondiciones	El usuario tiene que haber hecho la solicitud.
Poscondiciones	Informar al solicitante si fue o no publicada la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	Enviar Notificación, Solicitar Publicar Capa
Requisitos especiales	RF 4
Curso normal de eventos para el caso de uso	
Pantalla Operaciones Administrativas	



Pantalla Publicar Capas



Acción del actor	Respuesta del sistema
<p>1. El actor recibe la solicitud de publicar una capa (A).</p> <p>2. Acciona el botón Aceptar (B).</p> <p>4. Selecciona el Id de la capa en la lista desplegable (C) y acciona el botón Publicar (D).</p>	<p>3. Cambia el estado de la solicitud a Aceptada y habilita la lista desplegable (C) y el botón Publicar (D).</p> <p>5. El sistema realiza la operación de publicar la capa. Que es darle acceso de lectura y escritura a</p>

Capítulo 2: Características del Sistema

9. Recibe el mensaje.	<p>determinados usuarios.</p> <p>6. Ejecuta caso de uso Enviar Notificación</p> <p>7. Ejecuta el CU Informar al CIGEL</p> <p>8. El sistema envía un mensaje al actor("La operación fue realizada con éxito")</p>
Cursos alternos	
<p>Línea 2: Acciona el botón Cancelar (E) la solicitud de publicar una capa, se cambia el estado de la solicitud ha Denegado y se envía un mensaje informativo informándole que no fue aceptada su solicitud.</p>	

Caso de uso	Registrar CIGEL
Actores	
Propósito	Que el CIGEL con el que tiene una conexión nuestro sistema conozca la dirección, nombre y tenga un listado de las capas que están disponibles para otros sistemas.
Resumen	Este caso de uso comienza cuando se realiza un cambio en la conexión a un servidor CIGEL. El sistema le envía la nueva configuración y un listado de la cartografía disponible en el sistema.
Precondiciones	Debe existir una conexión con un CIGEL.
Poscondiciones	
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	
Requisitos especiales	RF15, 16
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema

Capítulo 2: Características del Sistema

	<ol style="list-style-type: none"> 1. Recibe una notificación de que se ha modificado la conexión al CIGEL. 2. Envía a la nueva dirección del CIGEL su nombre y su dirección, así como un listado de las capas que son públicas.
Cursos alternos	

Caso de uso	Solicitar Información de Capa al CIGEL	
Actores	Sistema Externo	
Propósito	Solicitar información de capas al CIGEL.	
Resumen	En este caso de uso comienza cuando el actor desea solicitar las capas que tiene acceso. Termina cuando obtiene esta información y devuelve el resultado al actor.	
Precondiciones	El usuario debe estar logeado.	
Poscondiciones	Actualiza la información.	
Tipo	Real y expandido	
Responsabilidades		
Cu relacionados		
Requisitos especiales	RF 16	
Curso normal de eventos para el caso de uso		
Acción del actor	Respuesta del sistema	
1. Solicita información de capa pasándole el id_capa y el nombre_capa de la capa, la información que puede solicitar es descripción, nombre_capa y localización de la capa de la cual solicita información.	<ol style="list-style-type: none"> 2. El sistema se relaciona con un sistema externo (CIGEL) mediante Servicios Web y solicita la información. 	

Capítulo 2: Características del Sistema

5. Recibe el resultado de la operación.	3. El sistema actualiza los datos. 4. Devuelve el resultado de la operación realizada.
Cursos alternos	
Si Ocorre un Error en el proceso de la solicitud de información al CIGEL se envía un mensaje de error.	

Caso de uso	Solicitud de Publicar Capas
Actores	Sistema Externo
Propósito	Publicar una capa.
Resumen	En este caso de uso comienza cuando el actor desea publicar una capa. Termina cuando envía la solicitud al administrador y devuelve el resultado al actor.
Precondiciones	El usuario debe estar logeado.
Poscondiciones	El administrador debe aceptar la solicitud para poder publicar la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	Enviar Notificación, Publicar Capa
Requisitos especiales	RF 3
Curso normal de eventos para el caso de uso	
Acción del actor	Respuesta del sistema
1. El actor crea y envía la solicitud, lo parámetros que tiene la solicitud son: id _ solicitud, nombre_solicitud, id_capa, descripción, estado. 5. El actor recibe el resultado de la operación.	2. El sistema realiza la operación de enviar la solicitud al administrador. 3. Ejecuta el Caso de Uso “Enviar Notificación”. 4. El sistema devuelve el resultado de la operación realizada.

Capítulo 2: Características del Sistema

Cursos alternos
Si Ocorre un Error en el proceso de crear la solicitud se envía un mensaje de error.

Caso de uso	Configurar Sistema Manualmente
Actores	Administrador
Propósito	Establecer los parámetros de conexión al sistema CIGEL y definir un directorio para el almacenamiento de los datos de los usuarios.
Resumen	Este caso de uso comienza cuando el administrador del sistema selecciona la opción de configurar el sistema. El administrador deberá especificar el directorio donde tendrá almacenada la información de los usuarios y, en caso de que se vaya a establecer una conexión a un servidor CIGEL, deberá especificar la dirección de este.
Precondiciones	El usuario debe estar autenticado.
Poscondiciones	Se almacena en BD toda la información de la capa.
Tipo	Real y expandido
Responsabilidades	
Cu relacionados	
Requisitos especiales	RF15, 16

Interfaces:

Pantalla Operaciones de Administración



Pantalla Configurar Sistema

The image shows a 'Configurar Sistema' dialog box with two sections: 'Crear Configuración' and 'Modificar Configuración'. Each section contains the following fields and controls:

- Crear Configuración:**
 - Nombre FGIS: [Text Field] (Callout B)
 - Dirección FGIS: [Text Field] (Callout C)
 - Guardar: [Text Field] (Callout D) | Examinar... [Button] (Callout A)
 - Conexión CIGEL: [Radio Button] (Callout D)
 - Nombre CIGEL: [Text Field] (Callout E)
 - Dirección CIGEL: [Text Field] (Callout F)
- Modificar Configuración:**
 - Nombre FGIS: [Text Field] (Callout J)
 - Dirección FGIS: [Text Field] (Callout I)
 - Guardar: [Text Field] (Callout L) | Examinar... [Button] (Callout K)
 - Conexión CIGEL: [Radio Button] (Callout L)
 - Nombre CIGEL: [Text Field] (Callout H)
 - Dirección CIGEL: [Text Field] (Callout H)

At the bottom of the dialog are 'Aceptar' and 'Cancelar' buttons (Callout G).

Curso normal de eventos para el caso de uso

Acción del actor	Respuesta del sistema
------------------	-----------------------

<p>1. El administrador selecciona la opción Configurar Sistema.</p> <p>3. Selecciona:</p> <ul style="list-style-type: none"> • Crear Configuración • Modificar Configuración 	<p>2. Muestra una interfaz para que el administrador escoja la acción que desea realizar.</p> <p>4. Si el Administrador selecciona:</p> <ul style="list-style-type: none"> • Crear Configuración, ver sección “Nueva Configuración” • Modificar Configuración, ver sección “Modificar Configuración”
Sección Nueva Configuración	
<p>1. Especifica el directorio donde se va a guardar la información de los usuarios (A), el nombre (B) y la dirección del sistema (C).</p> <p>2. Selecciona la opción conectar a CIGEL (D).</p> <p>4. Introduce el nombre (E) y la dirección solicitados (F).</p> <p>5. Acepta la configuración establecida (G).</p>	<p>3. Solicita el nombre y la dirección del servidor CIGEL al que se va a conectar.</p> <p>6. Valida la dirección del servidor.</p> <p>7. Valida los datos del directorio.</p> <p>8. Guarda la configuración.</p> <p>9. Invoca al CU Registrar CIGEL</p>
Sección Modificar Configuración	
	<p>1. El sistema busca la configuración existente.</p> <p>2. Muestra el nombre del sistema, la dirección del sistema,</p>

Capítulo 2: Características del Sistema

<p>3. Modifica la dirección del CIGEL (H) o la dirección del sistema (I), el nombre del sistema (J) o el nombre del CIGEL (K).</p> <p>4. Acepta los cambios realizados</p>	<p>la dirección del directorio donde se guarda la información de los usuarios, la dirección del CIGEL al que está conectado.</p> <p>5. Verifica la validez de los datos modificados.</p> <p>6. Guarda la configuración.</p> <p>7. Invoca al CU Registrar CIGEL</p>
Cursos alternos	
<p>Sección Nueva Configuración:</p> <p>Línea 2. No selecciona la opción conectarse al servidor CIGEL va a la línea 7 y el CU termina en la Línea 8.</p> <p>Línea 6: Detecta un error en la dirección del servidor y lo notifica.</p> <p>Línea 7: Detecta un error en el directorio, el nombre o la dirección del sistema y lo notifica.</p> <p>Sección Modificar Configuración:</p> <p>Línea 2: En caso que modifique solamente el directorio (L) se ejecuta la Línea 4 y el caso de uso termina en la Línea 6.</p> <p>Línea 4: En caso de que los datos no sean válidos, se muestra un mensaje de error.</p>	

6. Conclusiones

En este capítulo se comenzó a profundizar en el desarrollo de la propuesta de solución, obteniéndose una lista de los requisitos con los que debe cumplir el sistema. La elaboración del modelo del dominio o modelo conceptual ha permitido un mejor estudio del entorno en el que se desarrollará el sistema; y el diagrama de casos de usos del sistema; muestra de manera gráfica y se explican de manera detallada las funcionalidades que brindará el mismo. Con estos elementos se dará paso 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 del sistema que se está 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). Se obtendrá también un Modelo de Datos que servirá para la creación de la Base de Datos que utilizará el sistema.

2. Modelo de análisis

En el capítulo anterior se elaboró un Modelo Conceptual que describe y ayuda a comprender los conceptos significativos del dominio 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 punto de partida para realizar el diseño del sistema. En el modelo de análisis no se toman 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, pues su objetivo principal es comprender exactamente los requisitos del software y no precisar cómo se realizará su implementación.

Se considera que este modelo es la primera aproximación al Modelo de Diseño y se describe en un lenguaje comprensible para los desarrolladores; se compone además 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 del análisis es un artefacto en el que se representan los conceptos del dominio de un problema y que representa las cosas del mundo real, no de la implementación automatizada. Está compuesto por clases del análisis y sus relaciones.

Las clases del análisis están centradas en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Dichas clases tienen atributos y existen relaciones de asociación, agregación / composición, generalización / especialización y tipos asociativos entre ellas. Estas clases se clasifican en:

Capítulo 3: Análisis y Diseño del Sistema

- *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.

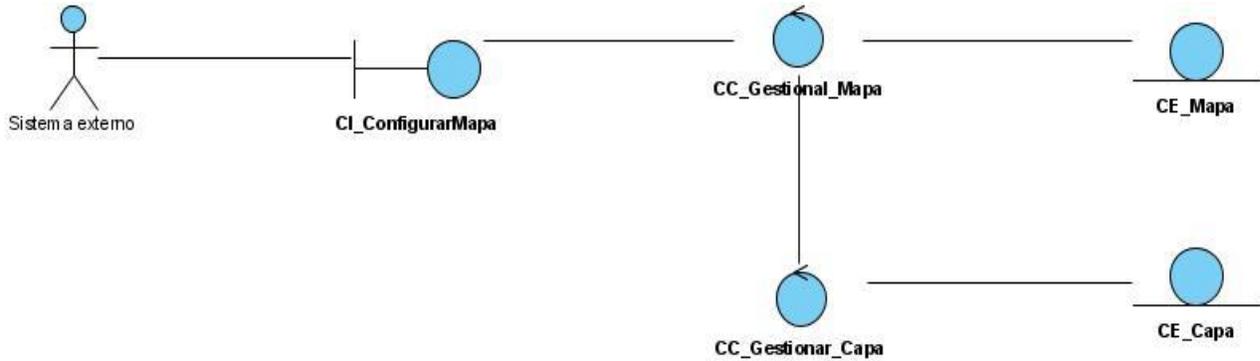


Figura #1 Diagrama de Análisis del Caso de Uso Configurar Mapa

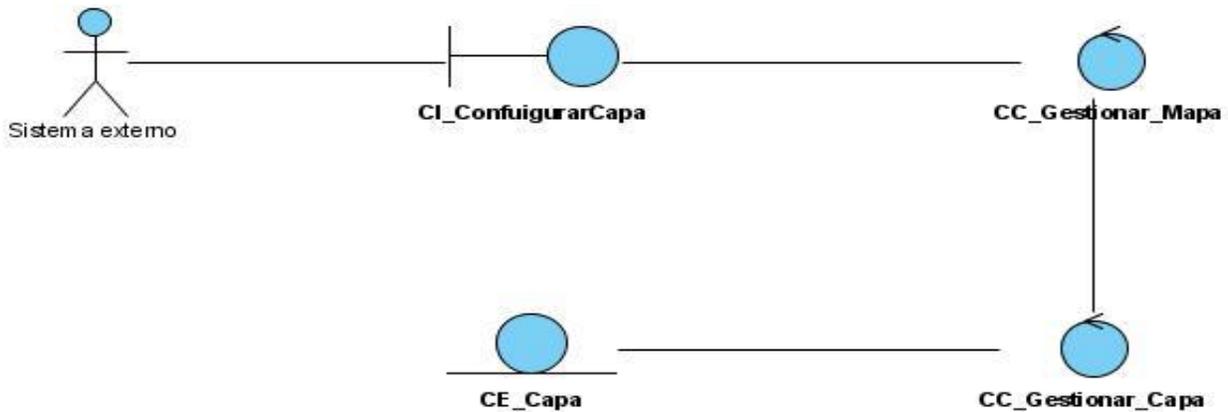


Figura #2 Diagrama de Análisis del Caso de Uso Configurar Capa

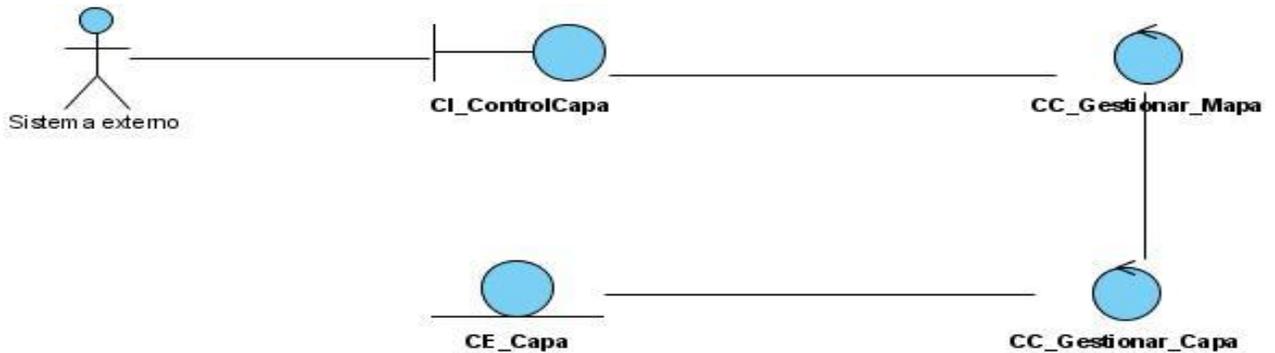


Figura #3 Diagrama de Análisis del Caso de Uso Control de Capa

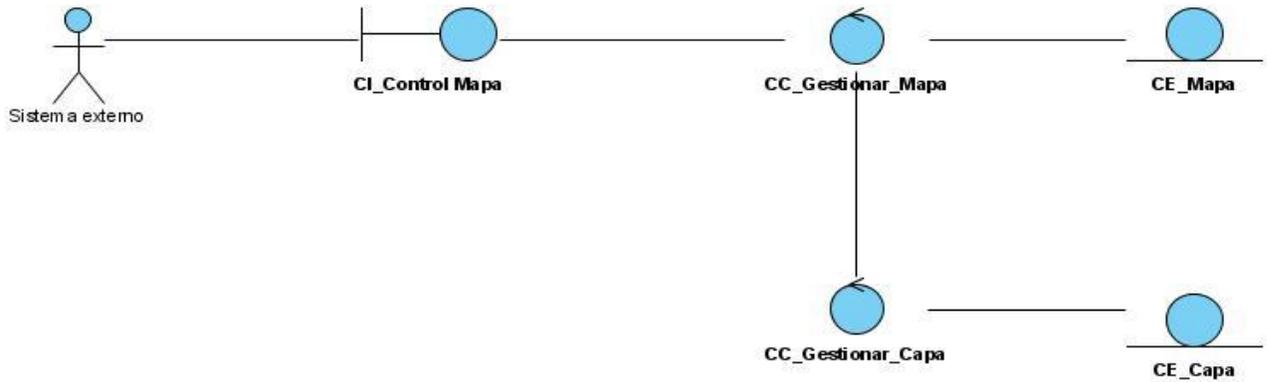


Figura #4 Diagrama de Análisis del Caso de Uso Control de Mapa

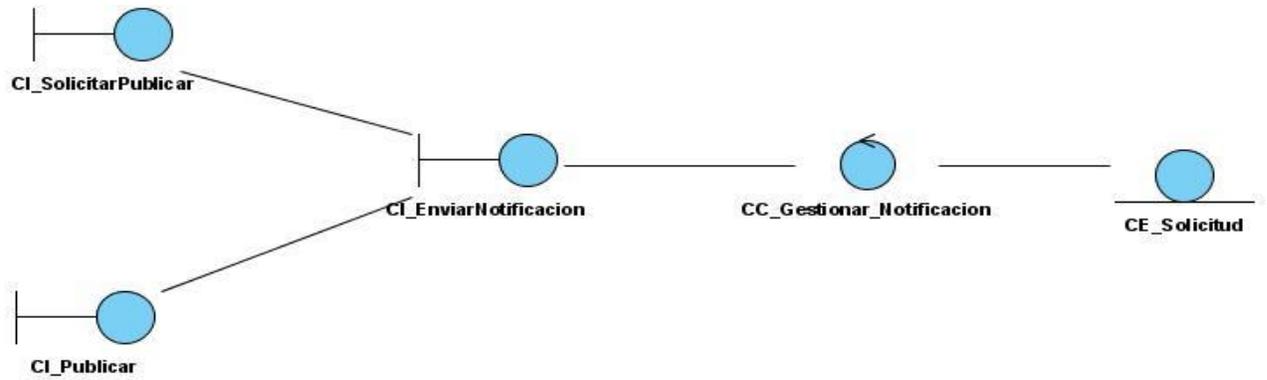


Figura #5 Diagrama de Análisis del Caso de Uso Enviar Notificación

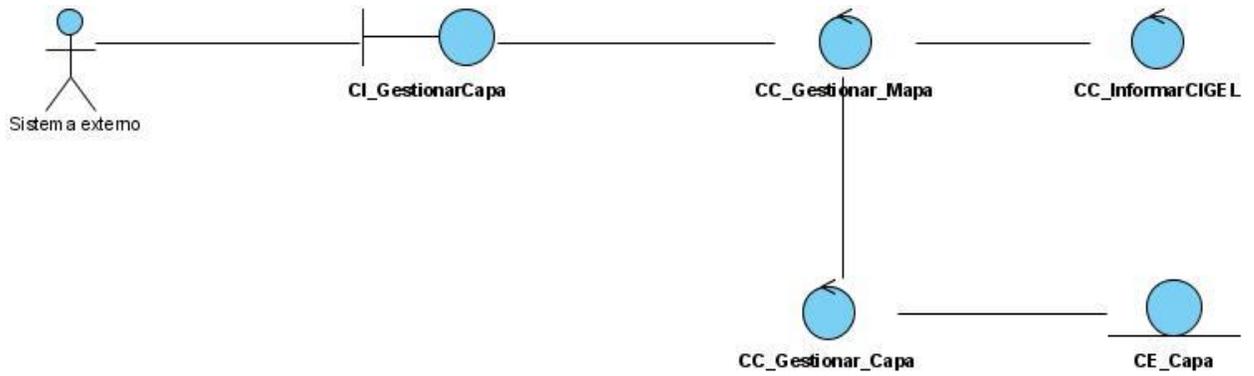


Figura #6 Diagrama de Análisis del Caso de Uso Gestionar Capa

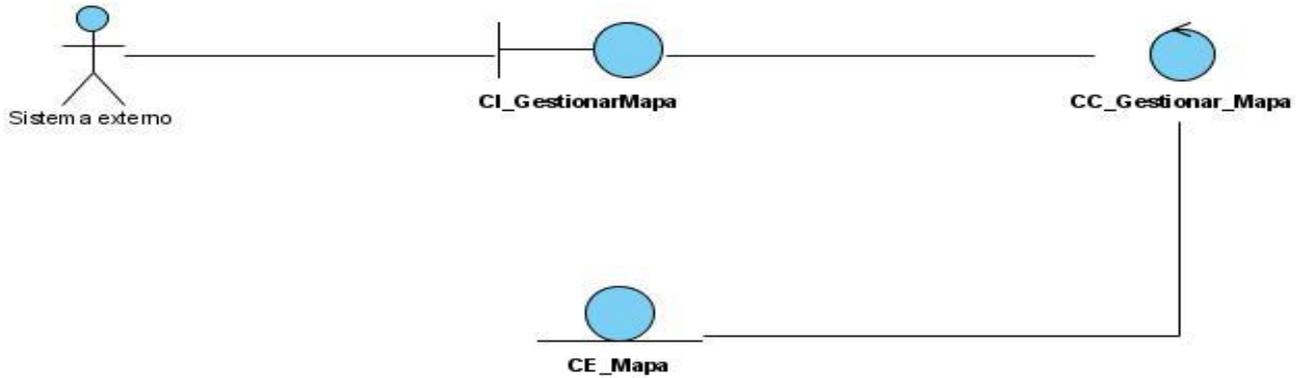


Figura #7 Diagrama de Análisis del Caso de Uso Gestionar Mapa

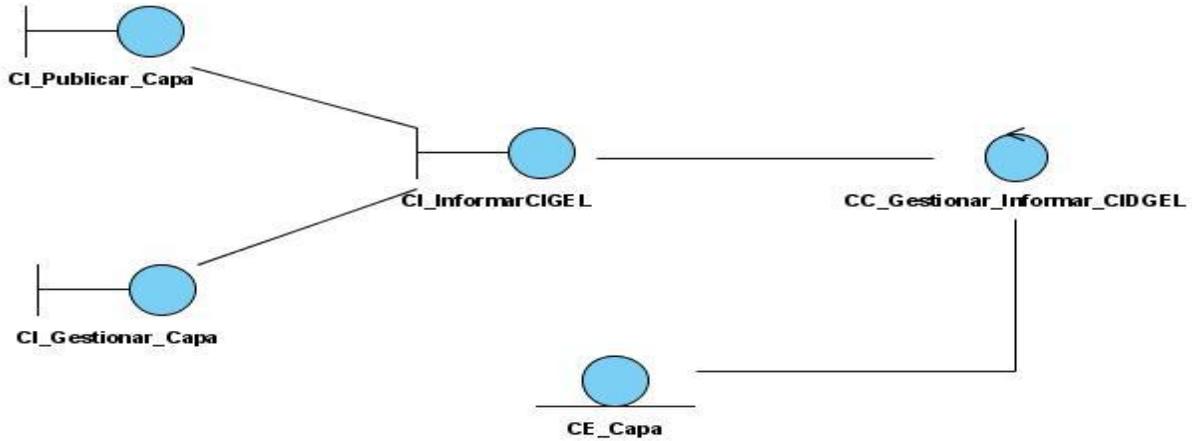


Figura #8 Diagrama de Análisis del Caso de Uso Informar Estado de Capa al CIGEL

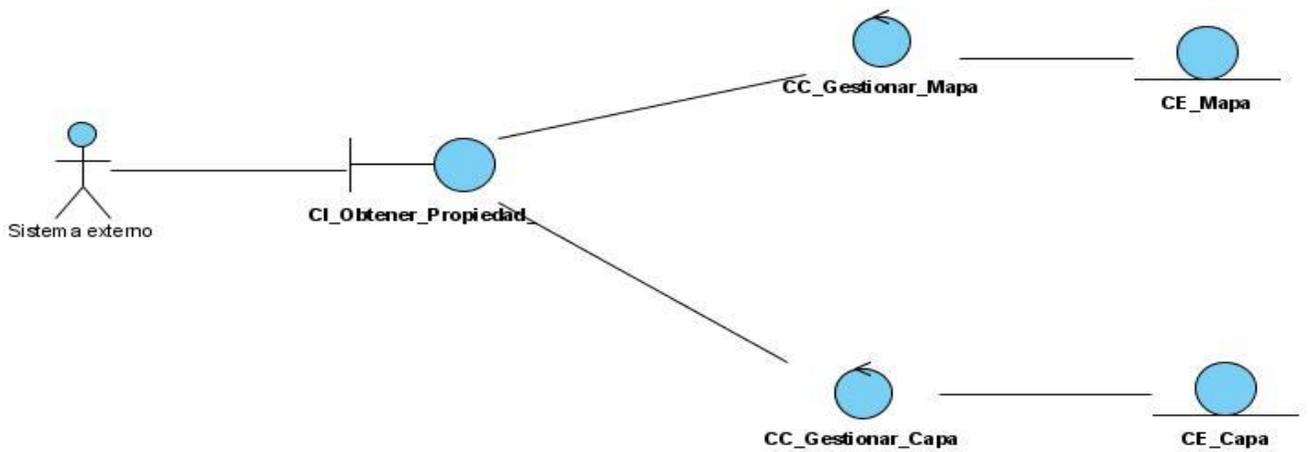


Figura #9 Diagrama de Análisis del Caso de Uso Obtener Propiedad

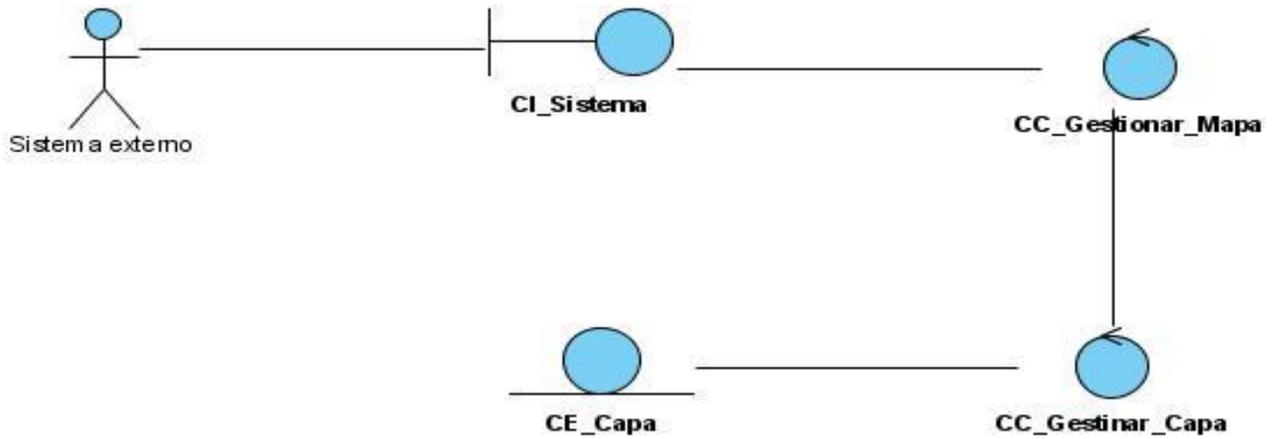


Figura #10 Diagrama de Análisis del Caso de Uso Obtener Propiedad Capa

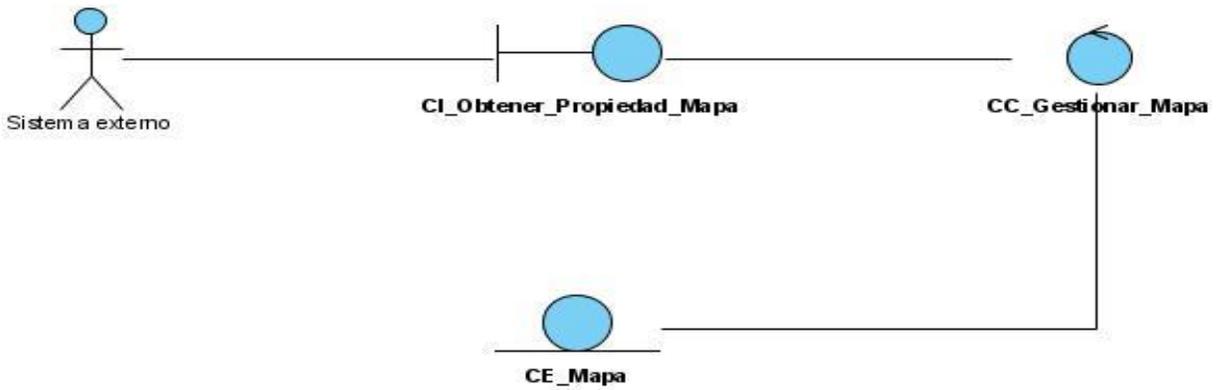


Figura #11 Diagrama de Análisis del Caso de Uso Obtener Propiedad Mapa

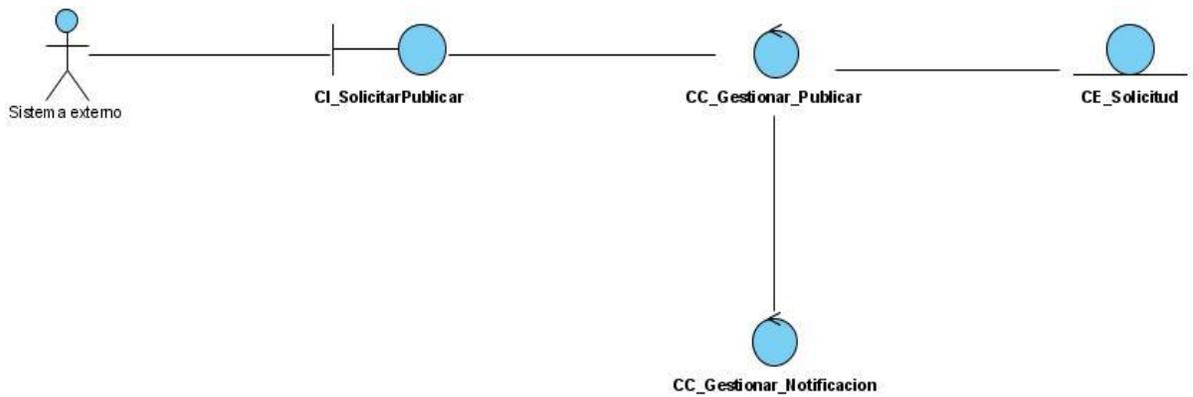


Figura #12 Diagrama de Análisis del Caso de Uso Solicitar Publicar Capa

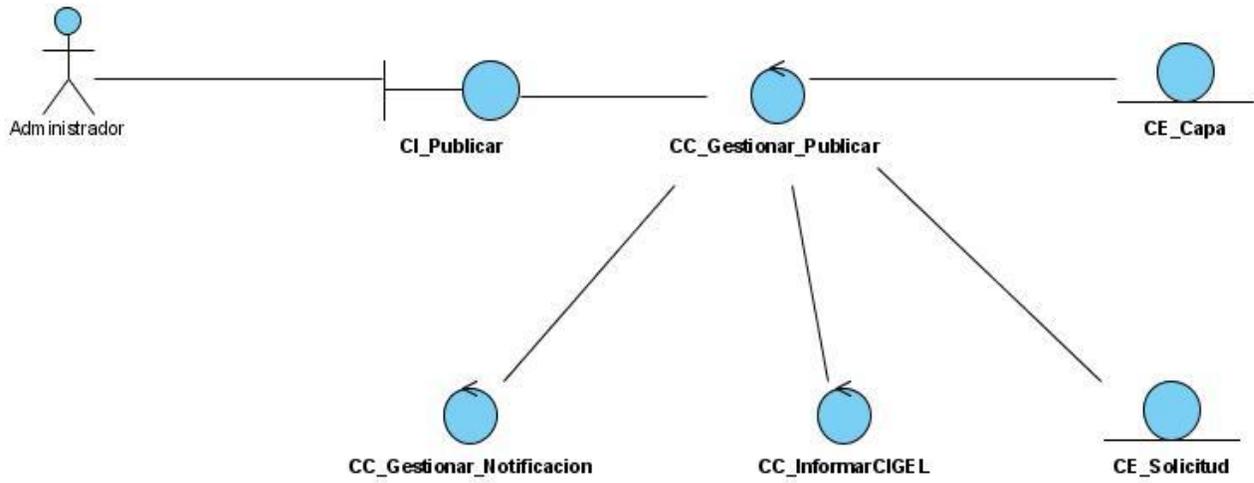


Figura #13 Diagrama de Análisis del Caso de Uso Publicar Capa

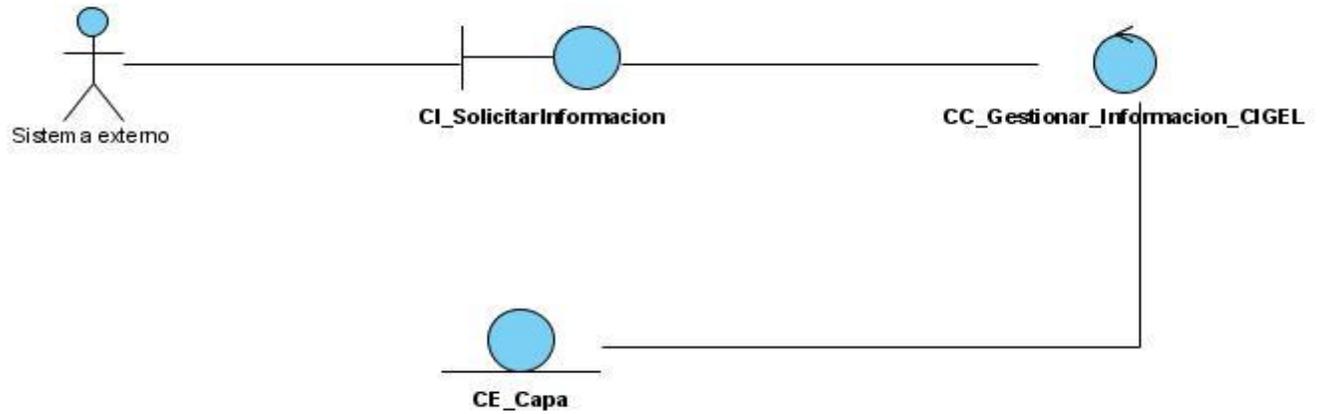


Figura #14 Diagrama de Análisis del Caso de Uso Solicitar Información al CIGEL

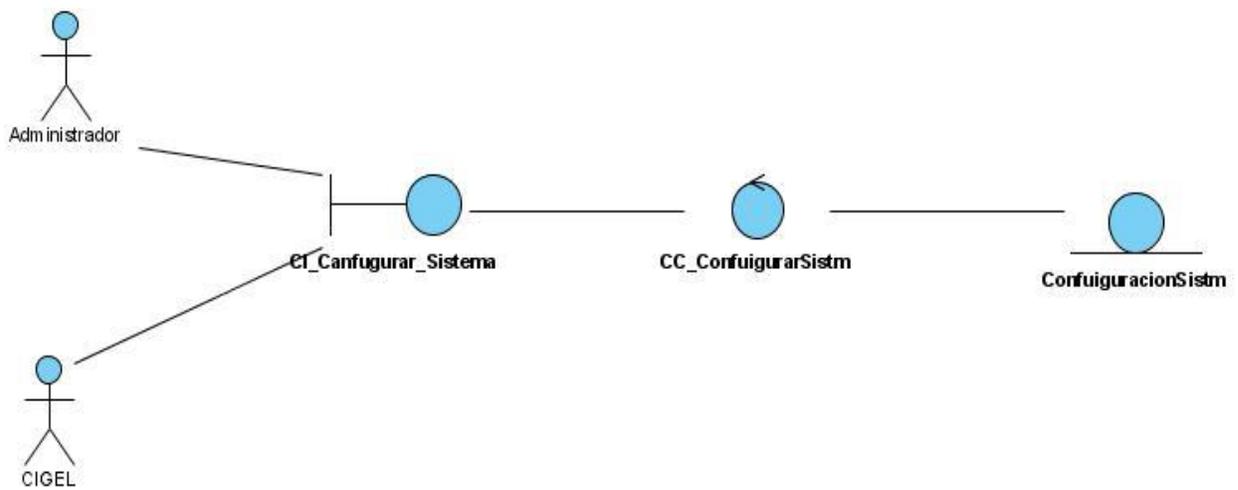


Figura # 15 Diagrama de Análisis del Caso de Uso Configurar Sistema

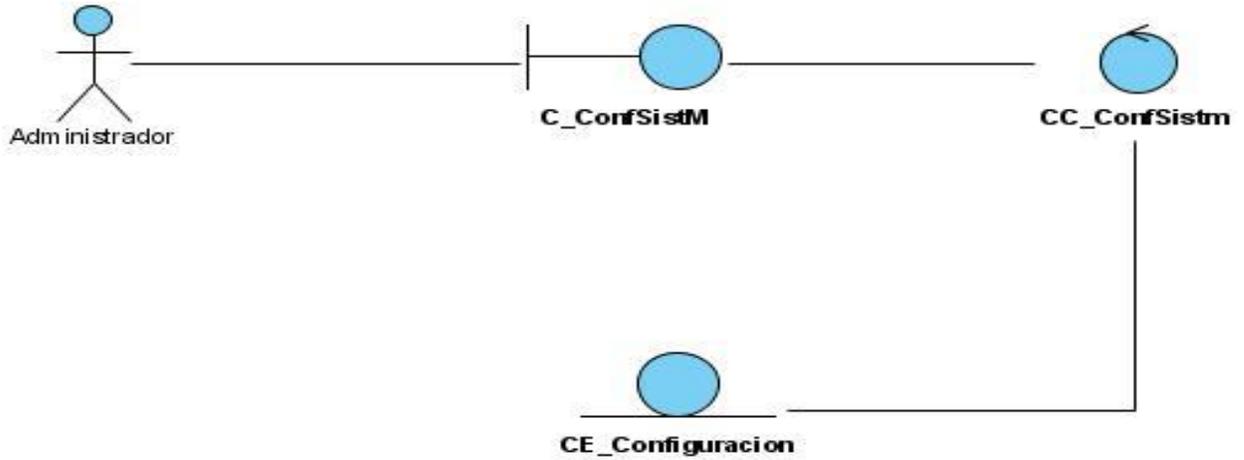


Figura # 16 Diagrama de Análisis del Caso de Uso Configurar Sistema Manual

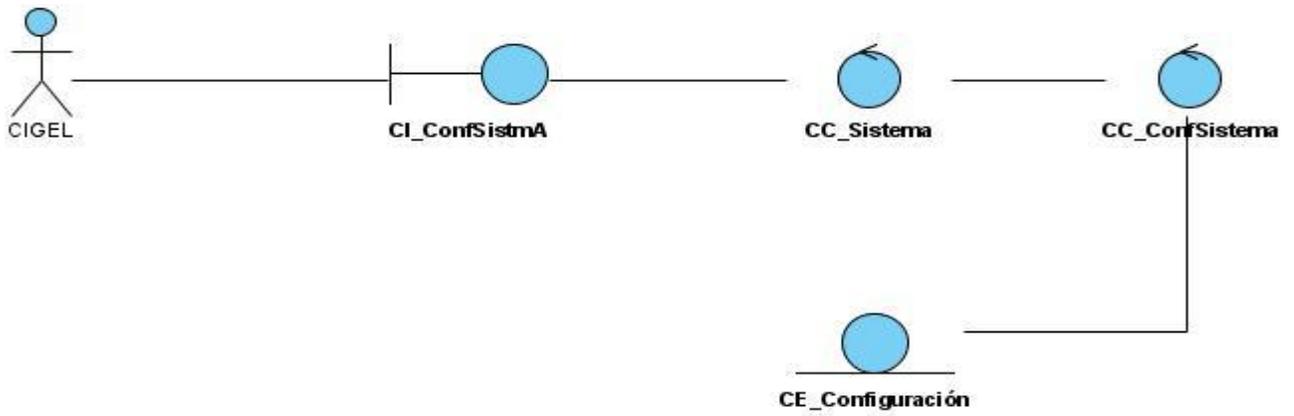


Figura # 17 Diagrama de Análisis del Caso de Uso Configurar Sistema Automático

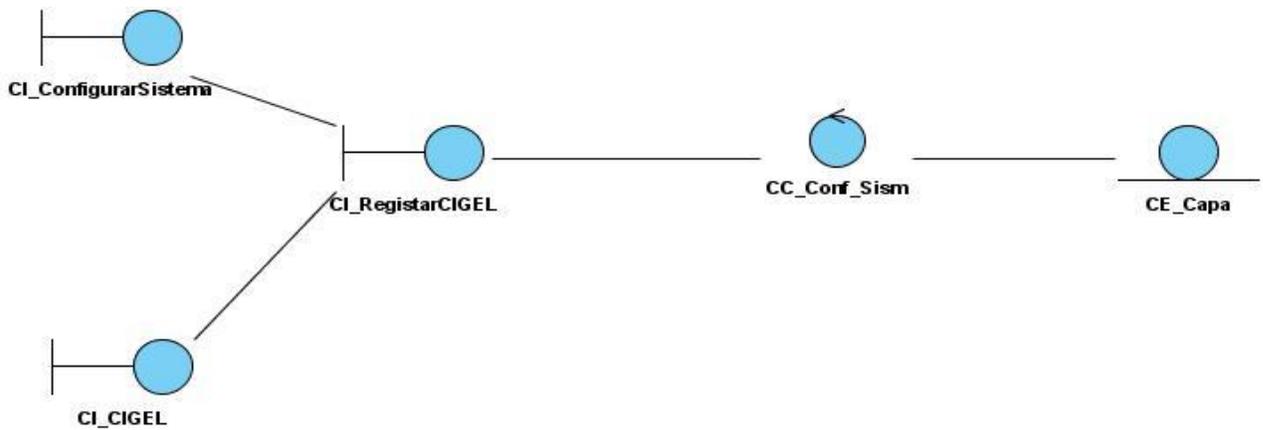


Figura # 18 Diagrama de Análisis del Caso de Uso Registrar CIGEL

3. Modelo de Diseño.

En el diseño modelamos el sistema y encontramos su forma, incluyendo la arquitectura, para que soporte todos los requisitos funcionales y no funcionales. El Modelo de Diseño es un modelo de objetos que describe la realización física de los casos de uso. 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. 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. Patrones de diseño

Los patrones de diseño que empleamos son los patrones **GRASP**: General Responsibility Assignment Software Patterns (patrones generales de software para asignar responsabilidades)

- Experto

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

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

Beneficios:

- Se conserva el encapsulamiento.
- El comportamiento se distribuye entre las clases que cuentan con la información requerida.
- Clases mas 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 clases de crear instancias de otra, ya se agregando, utilizando, conteniendo, registrando o creando objetos de la otra clase.

Beneficios:

- Brinda soporte a un **bajo acoplamiento**
- 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

- Mejoran la claridad y la facilidad con que se entiende el diseño.
 - Se simplifican el mantenimiento y las mejoras en funcionalidad.
 - A menudo se genera un bajo acoplamiento.
 - 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

Beneficio:

- No se afectan por cambios de otros componentes.
 - Fáciles de entender por separado.
 - 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

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

3.2. Diagramas de clases Web

A diferencia del resto de los diagramas de clases, en el que se construye para las aplicaciones web son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. Se opta en este caso por modelar los artefactos del sistema, es decir: modelar las páginas, los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de estas, una vez que estén en el navegador del cliente para obtener un nivel correcto de abstracción y detalle que nos permita obtener un resultado final.

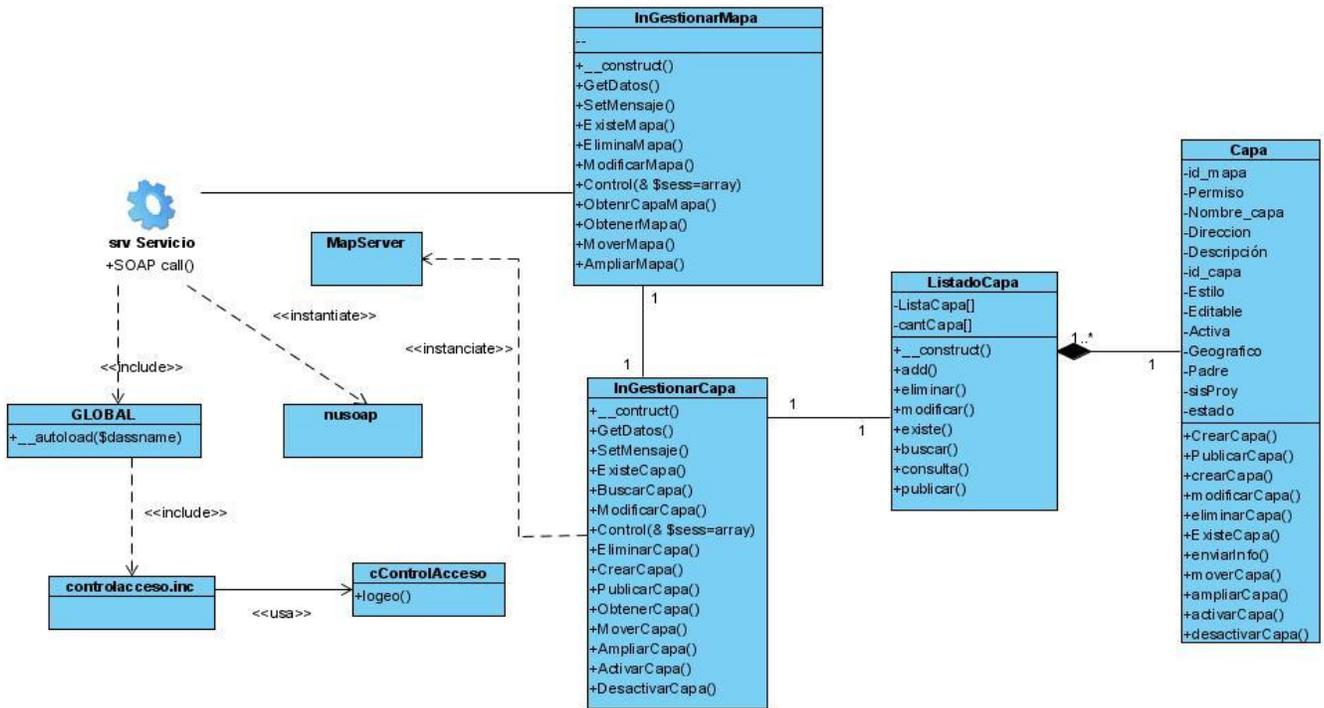


Figura # 19 Diagrama del Diseño del Caso de Uso Configurar Capa

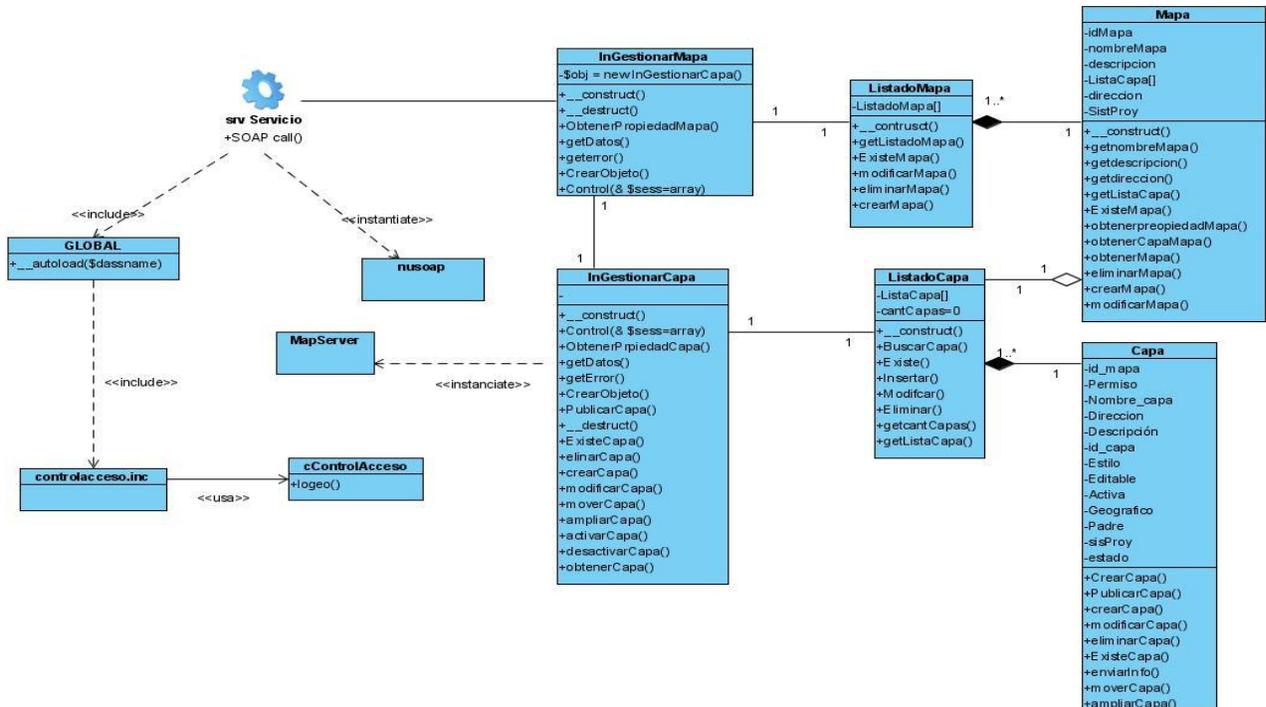


Figura # 20 Diagrama del Diseño del Caso de Uso Configurar Mapa

Capítulo 3: Análisis y Diseño del Sistema

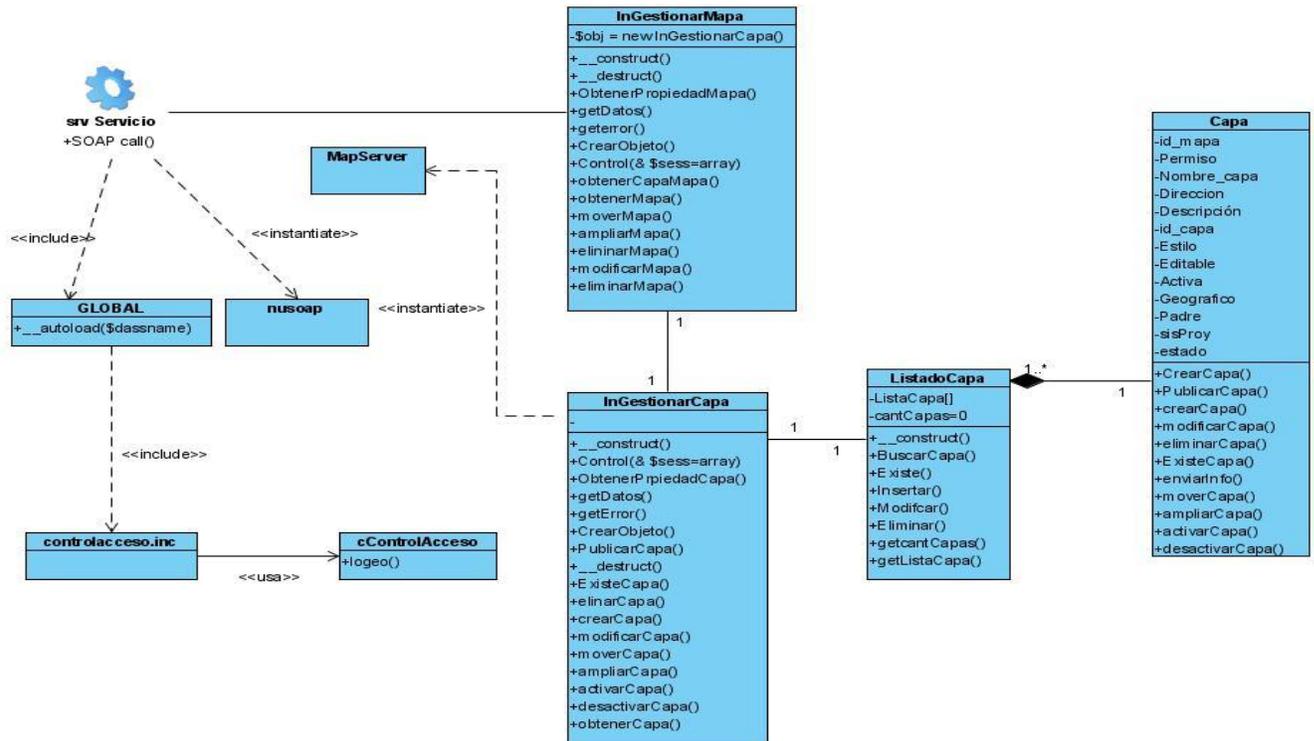


Figura #21 Diagrama del Diseño del Caso de Uso Control de Capa

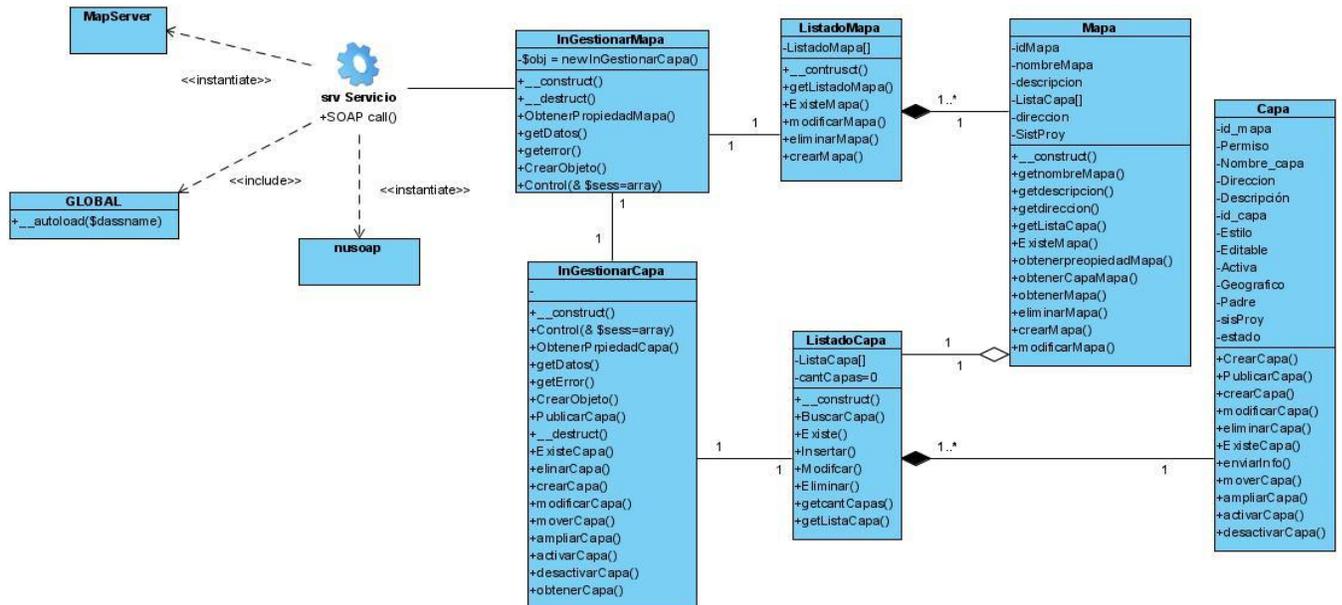


Figura #22 Diagrama del Diseño del Caso de Uso Control de Mapa

Capítulo 3: Análisis y Diseño del Sistema

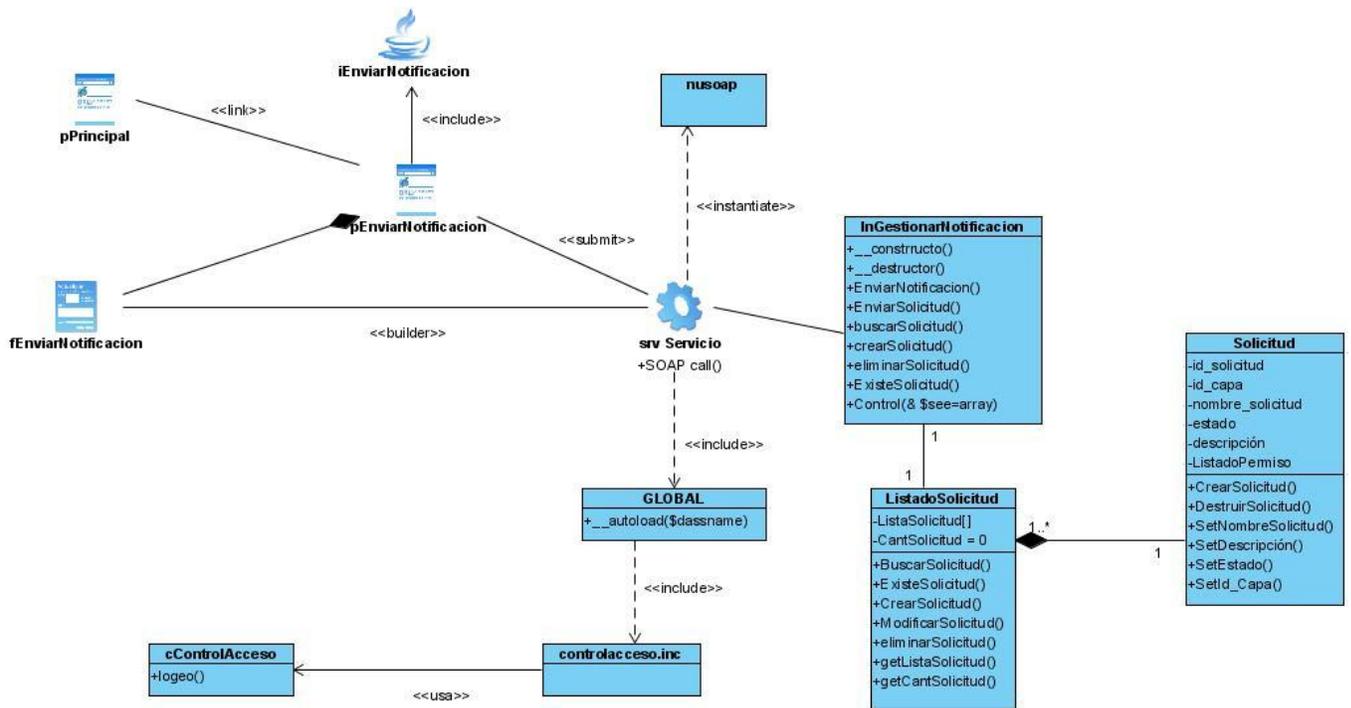


Figura #23 Diagrama del Diseño del Caso de Uso Enviar Notificación

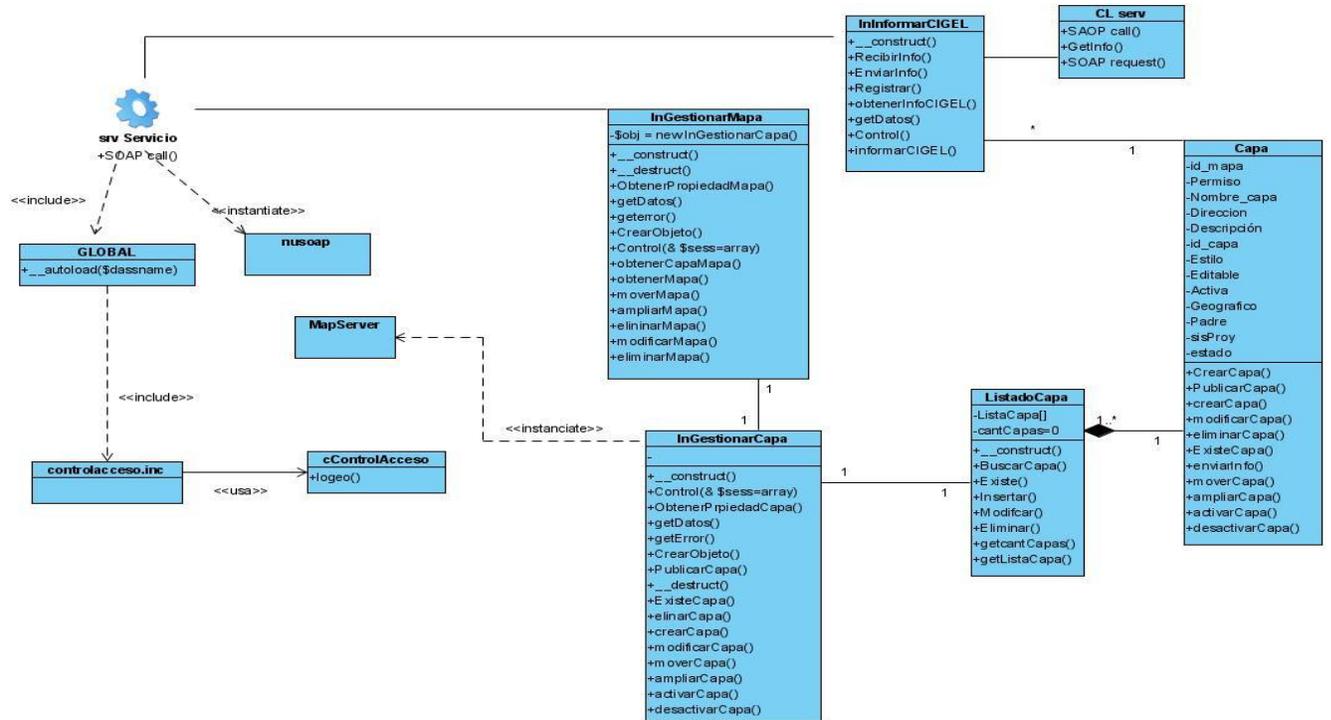


Figura #24 Diagrama de Diseño del Caso de Uso Gestionar Capa

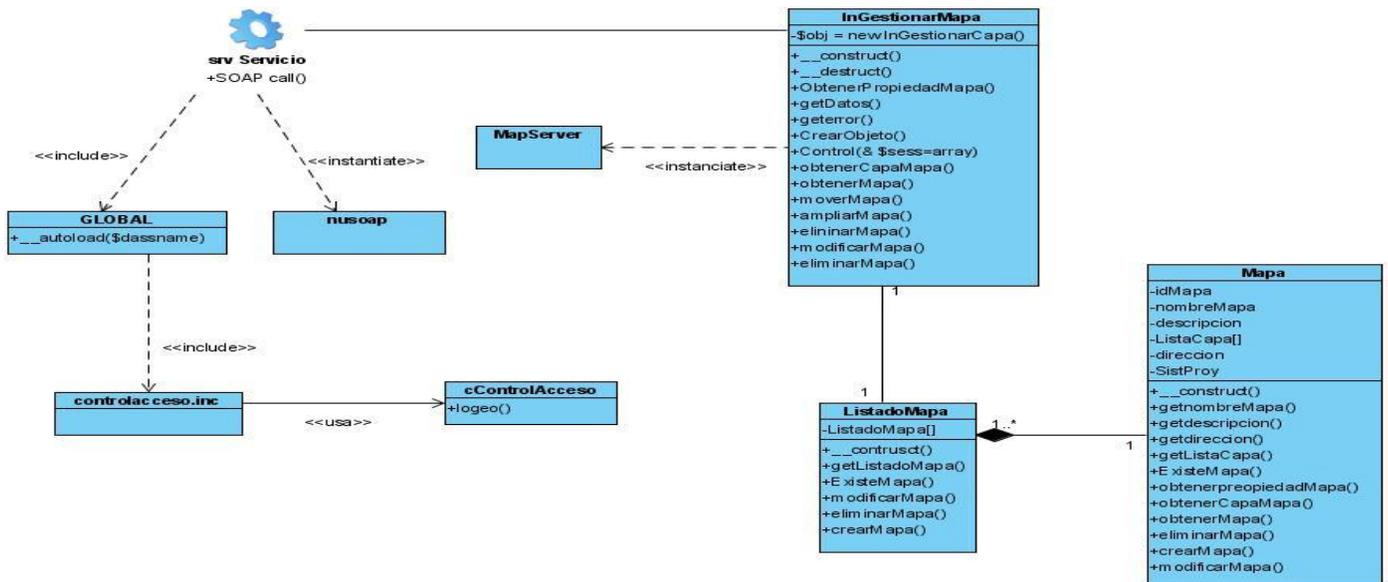


Figura #25 Diagrama del Diseño del Gestionar Mapa

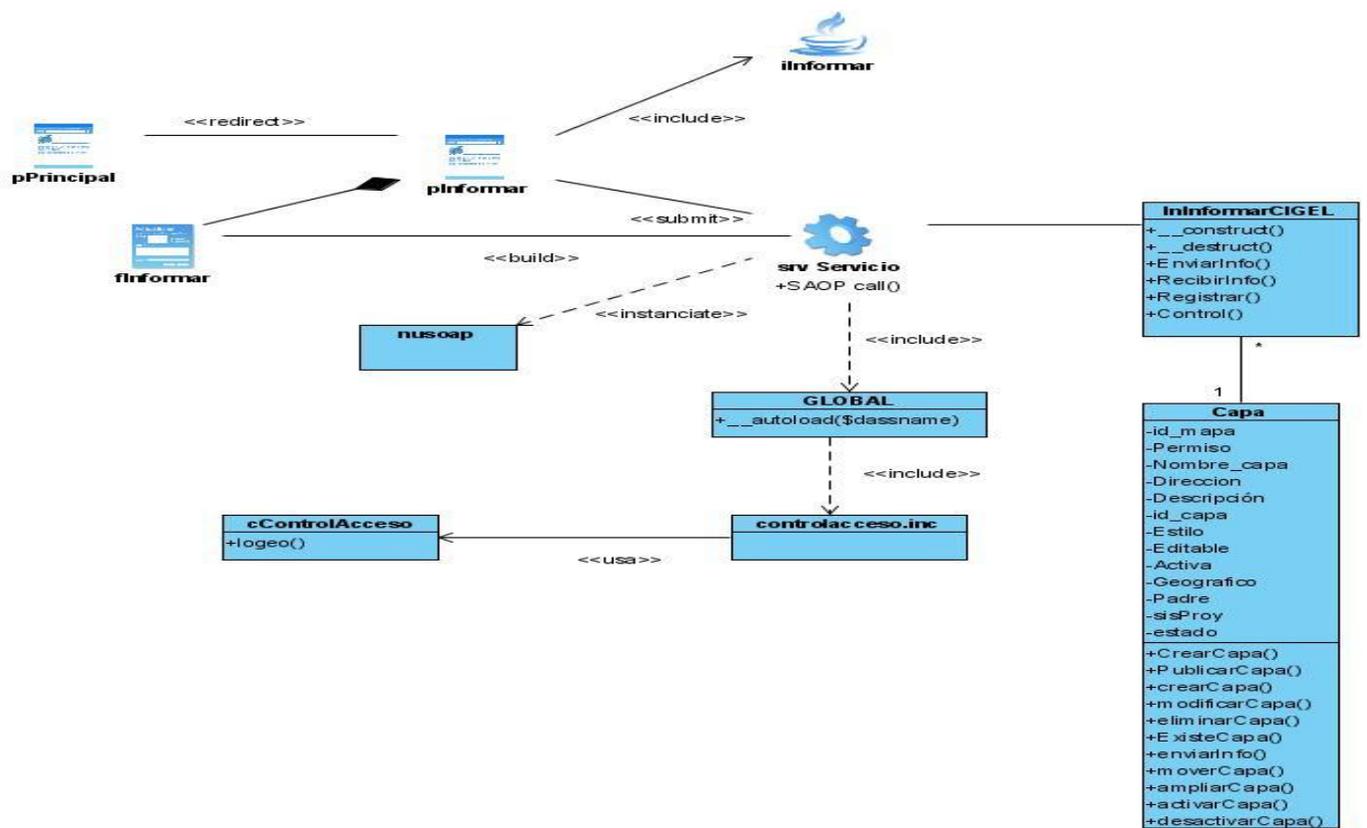


Figura #26 Diagrama del Diseño del Caso de Uso Informar al CIGEL

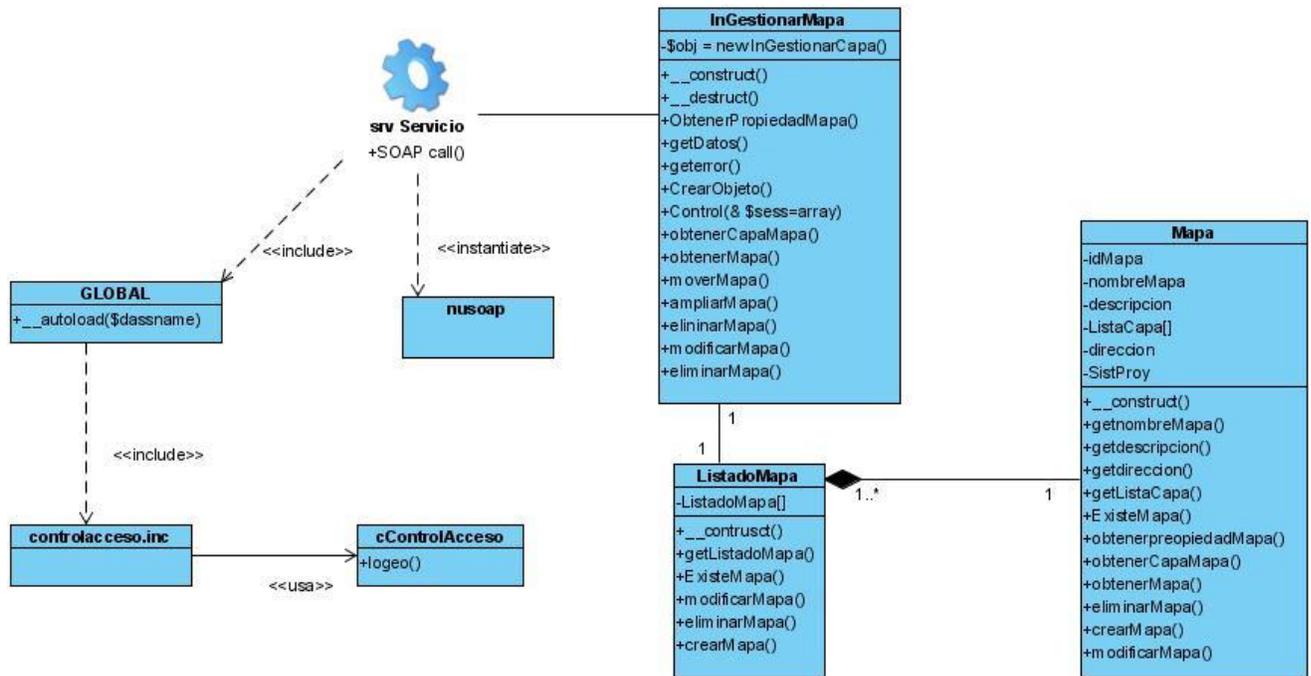


Figura #27 Diagrama del Diseño del Caso de Uso Obtener Propiedad Mapa

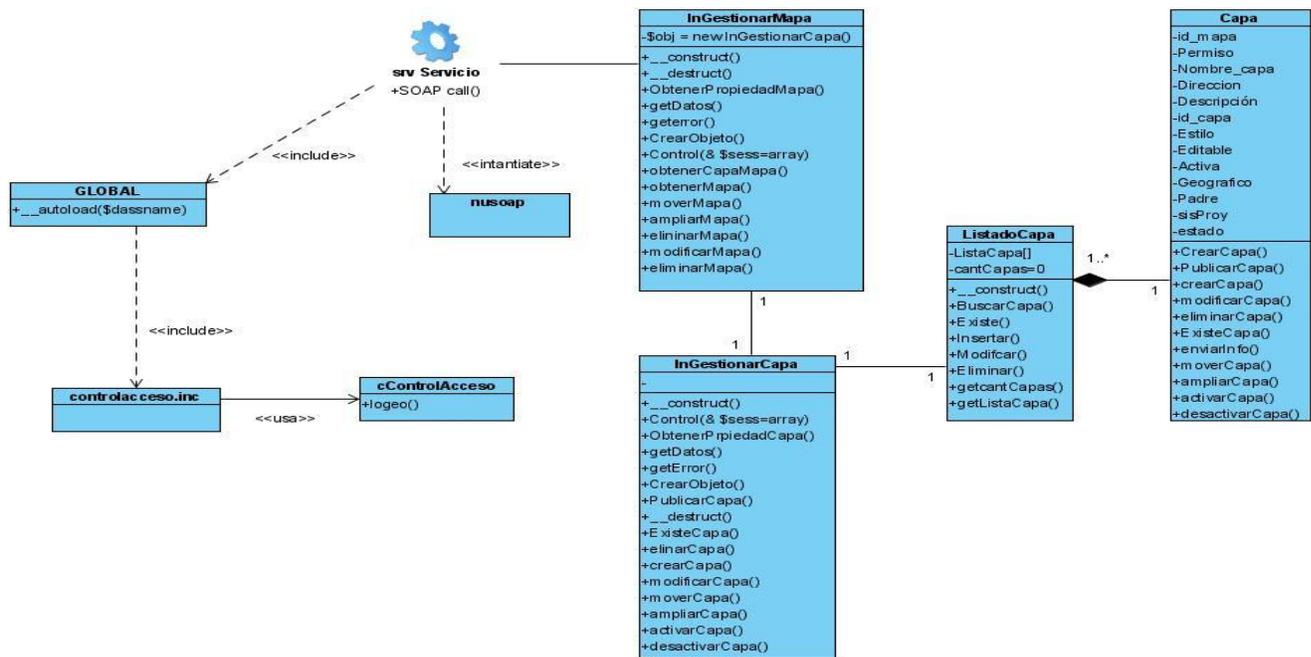


Figura #28 Diagrama del Diseño del Caso de Uso Obtener Propiedad Capa

Capítulo 3: Análisis y Diseño del Sistema

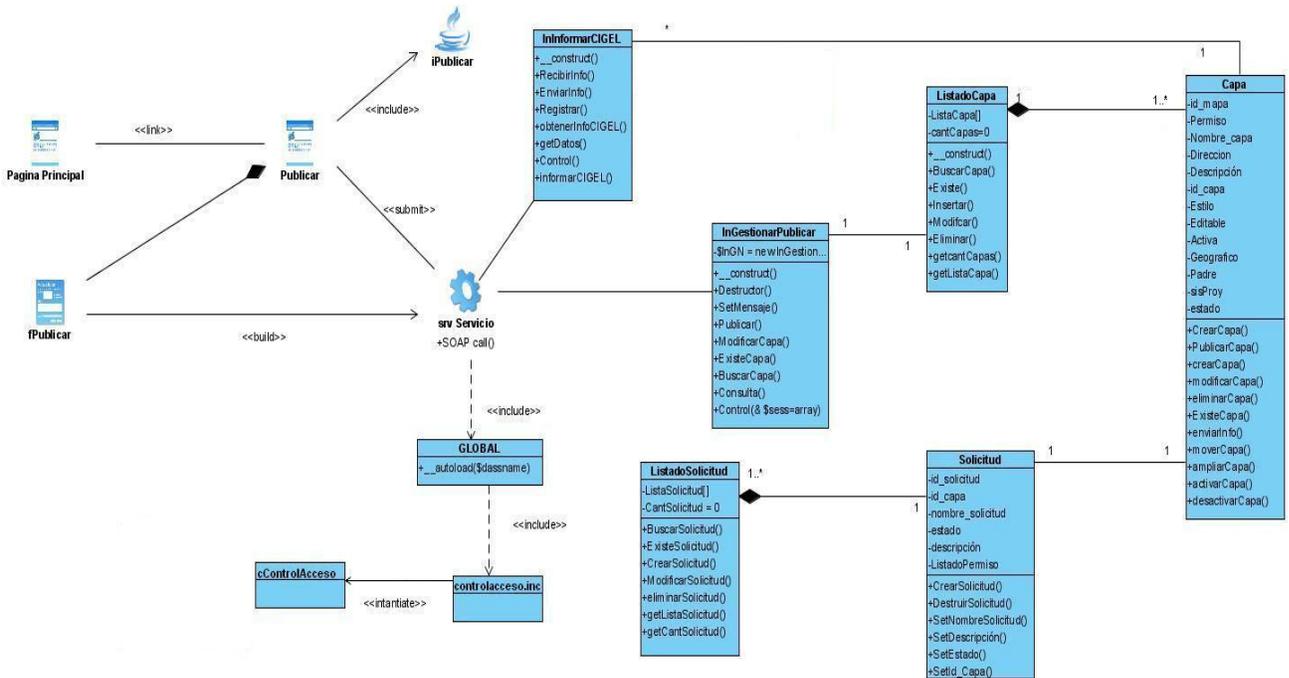


Figura #29 Diagrama de Diseño del Caso de Uso Publicar Capa

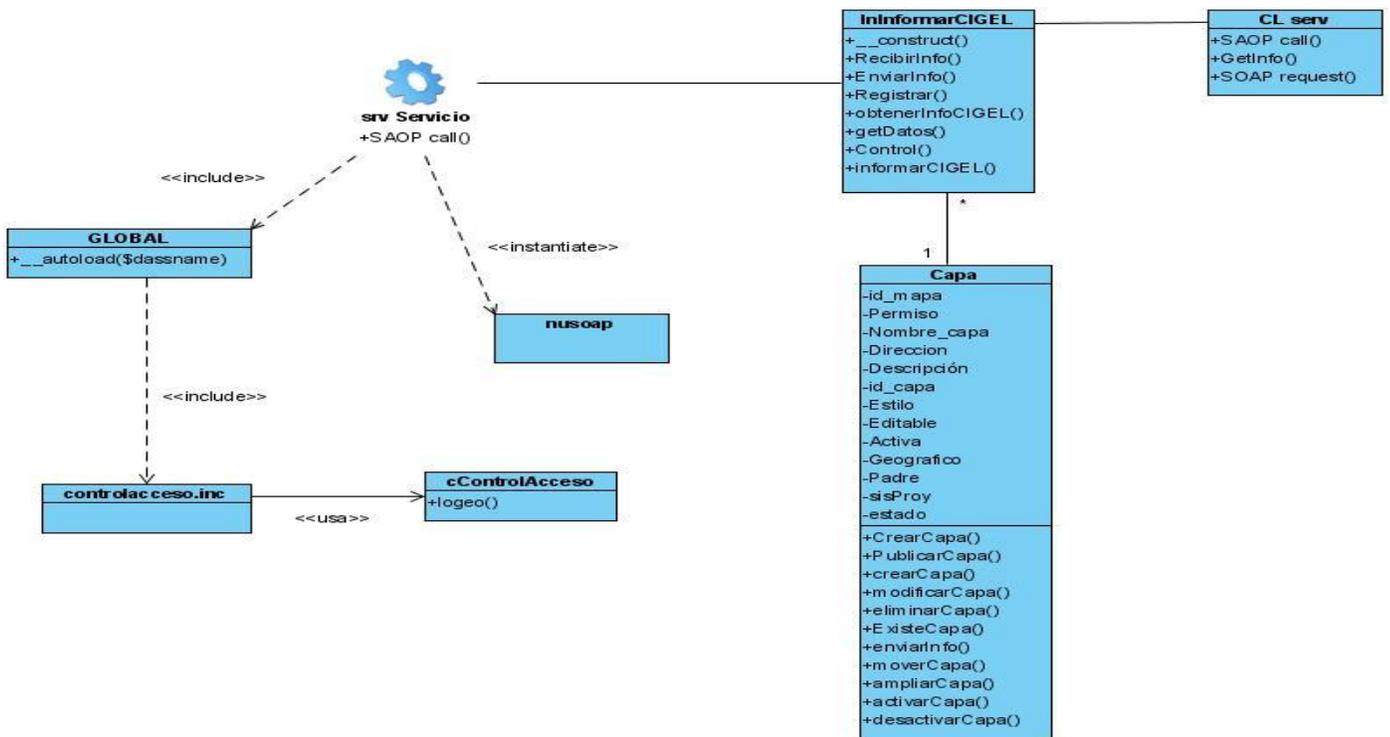


Figura #30 Diagrama del Diseño del Caso de Uso Solicitar Información al CIGEL

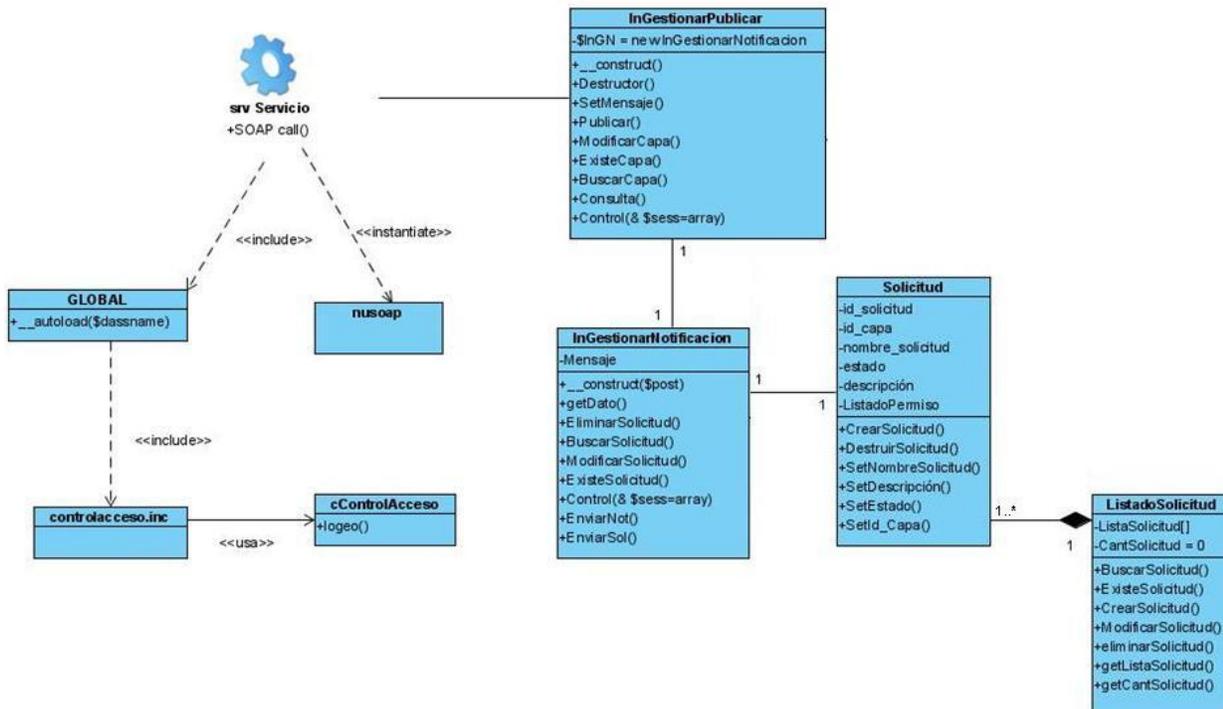


Figura #31 Diagrama del Diseño del Caso de Uso Solicitar Publicar Capa

3.3. Diagramas de interacción

Un diagrama de interacción muestra una interacción, que consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Estos diagramas se dividen en dos tipos de diagramas de UML: los diagramas de secuencia y los diagramas de colaboración. Para modelar los aspectos dinámicos de este sistema se utilizaron diagramas de secuencia por cada caso de uso, que son los que destacan la ordenación temporal de los mensajes, lo que permite formar una idea más acertada de la forma en que ocurren los sucesos.

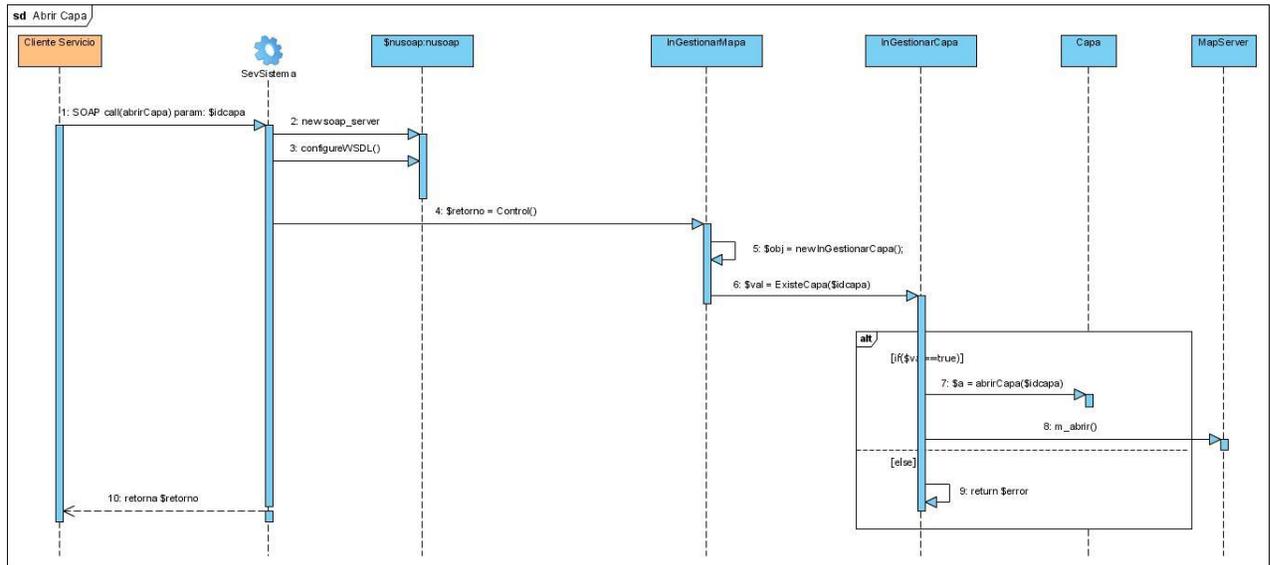


Figura # 32 Diagrama de Secuencia del Caso de Uso Abrir Capa

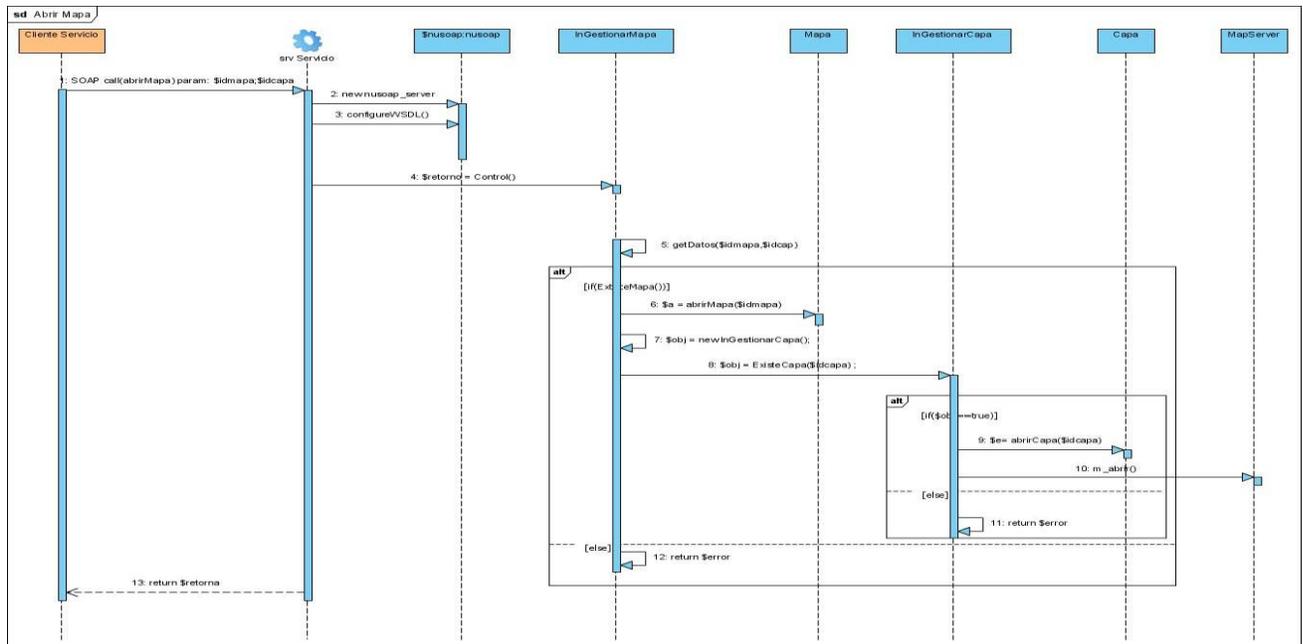


Figura # 33 Diagrama de Secuencia del Caso de Uso Abrir Mapa

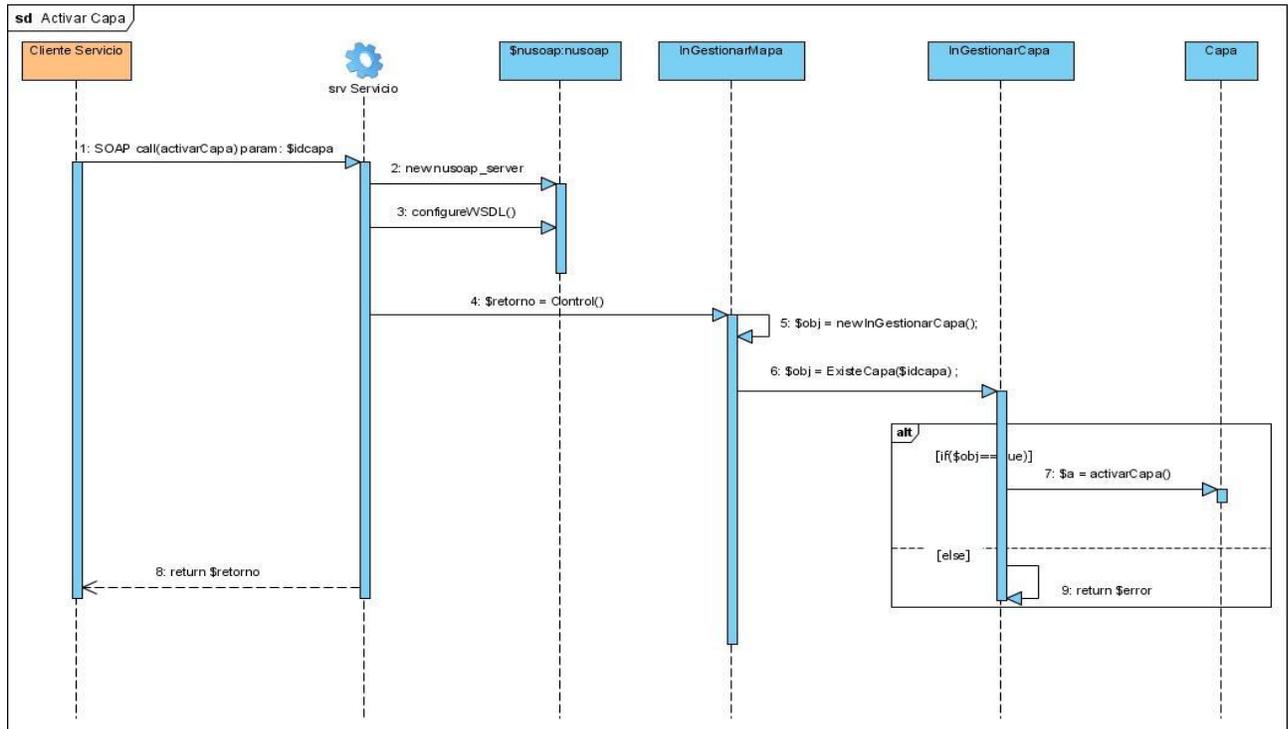


Figura # 34 Diagrama de Secuencia del Caso de Uso Activar Capa

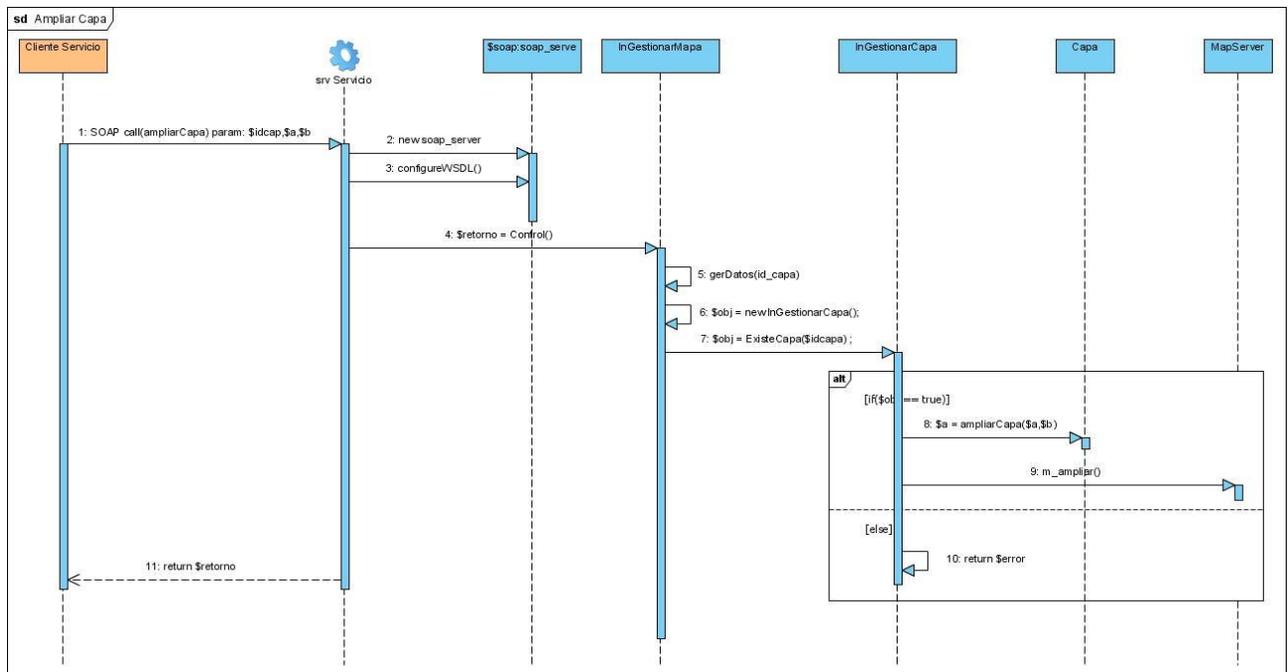


Figura # 35 Diagrama de Secuencia del Caso de Uso Ampliar Capa

Capítulo 3: Análisis y Diseño del Sistema

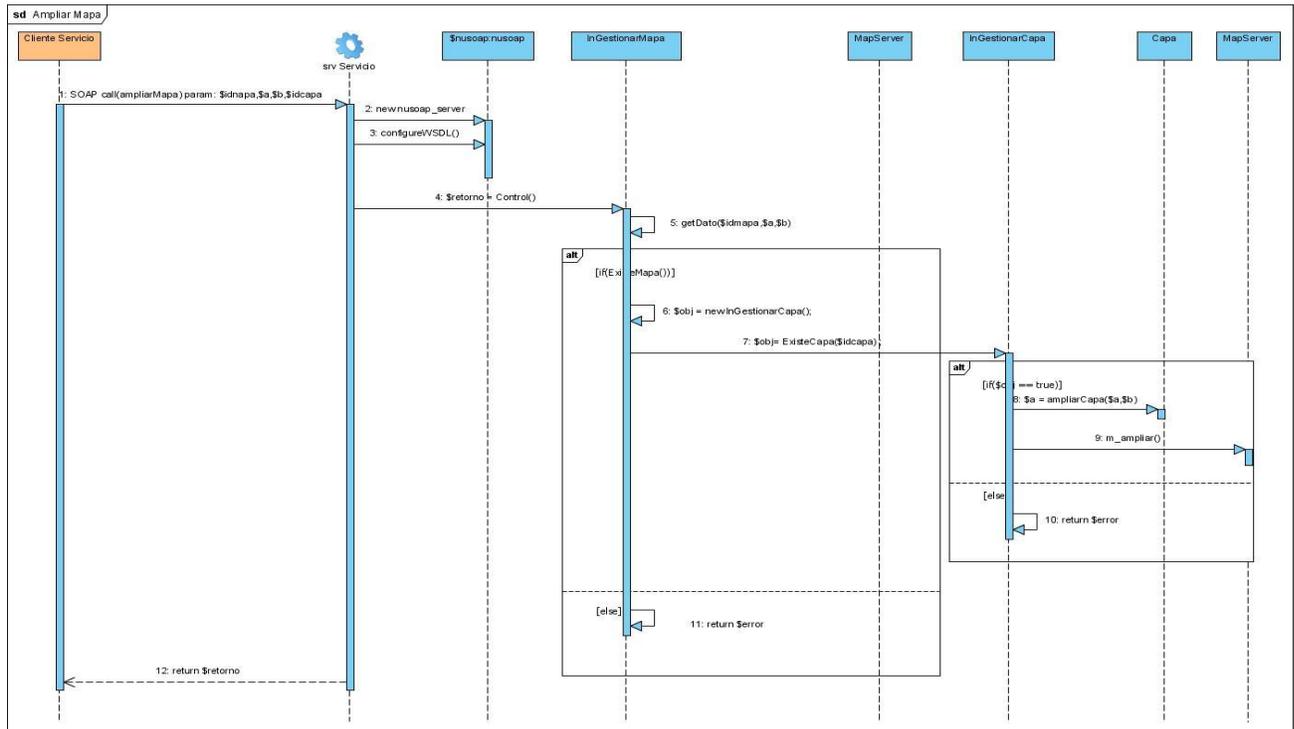


Figura # 36 Diagrama de Secuencia del Caso de Uso Ampliar Mapa

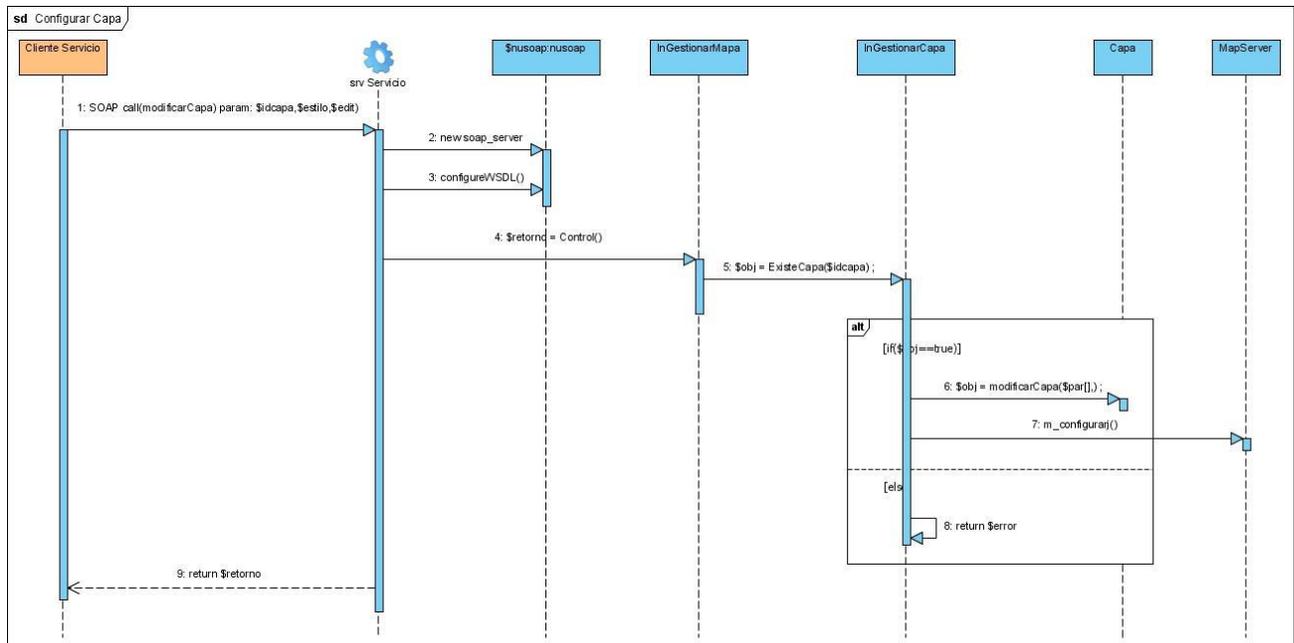


Figura # 37 Diagrama de Secuencia del Caso de Uso Configurar Capa

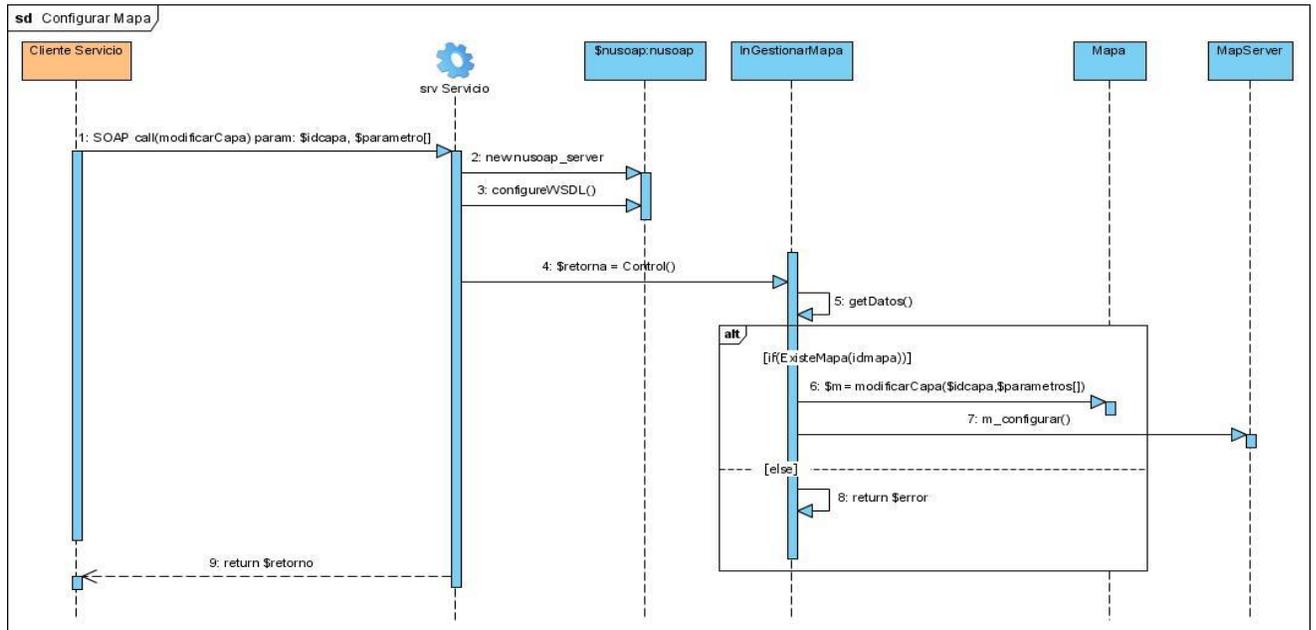


Figura # 38 Diagrama de Secuencia del Caso de Uso Configurar Mapa

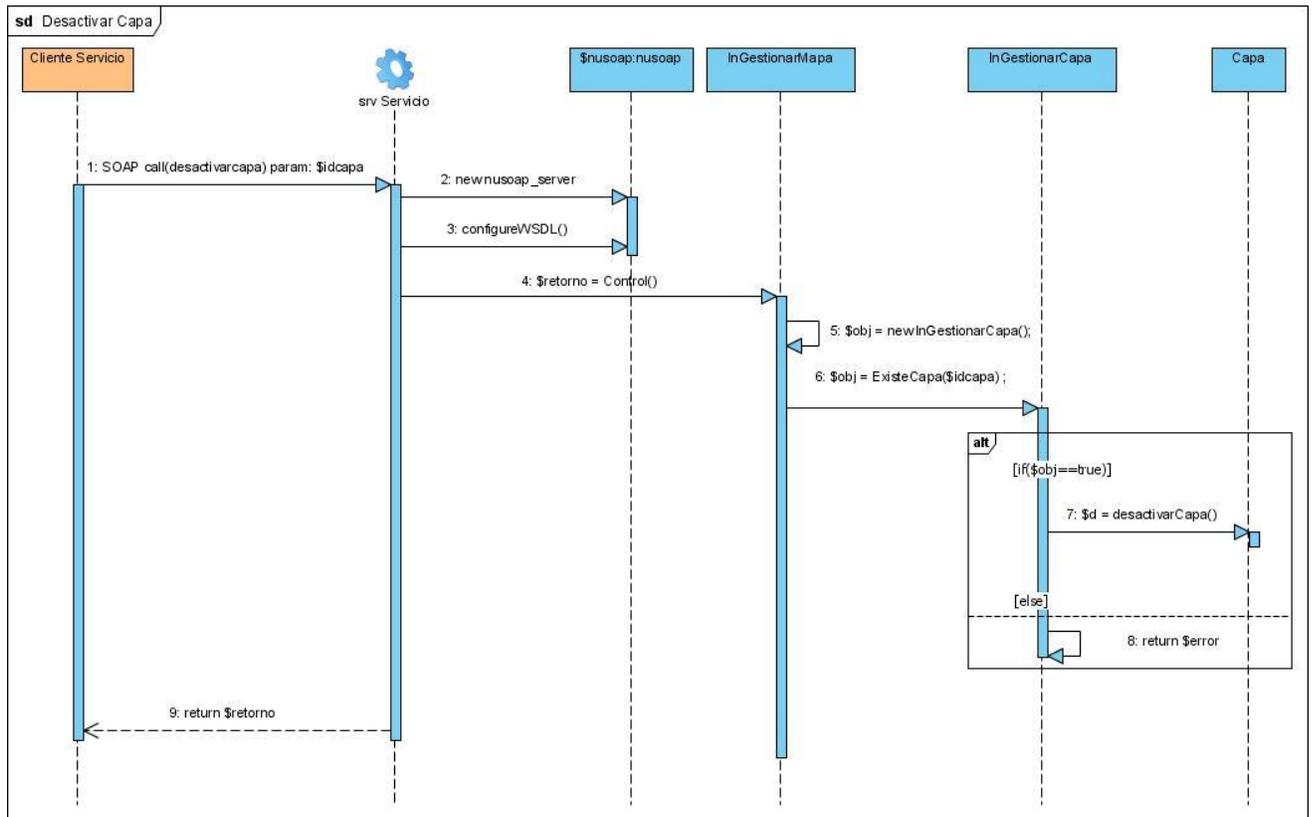


Figura # 39 Diagrama de Secuencia del Caso de Uso Desactivar Capa

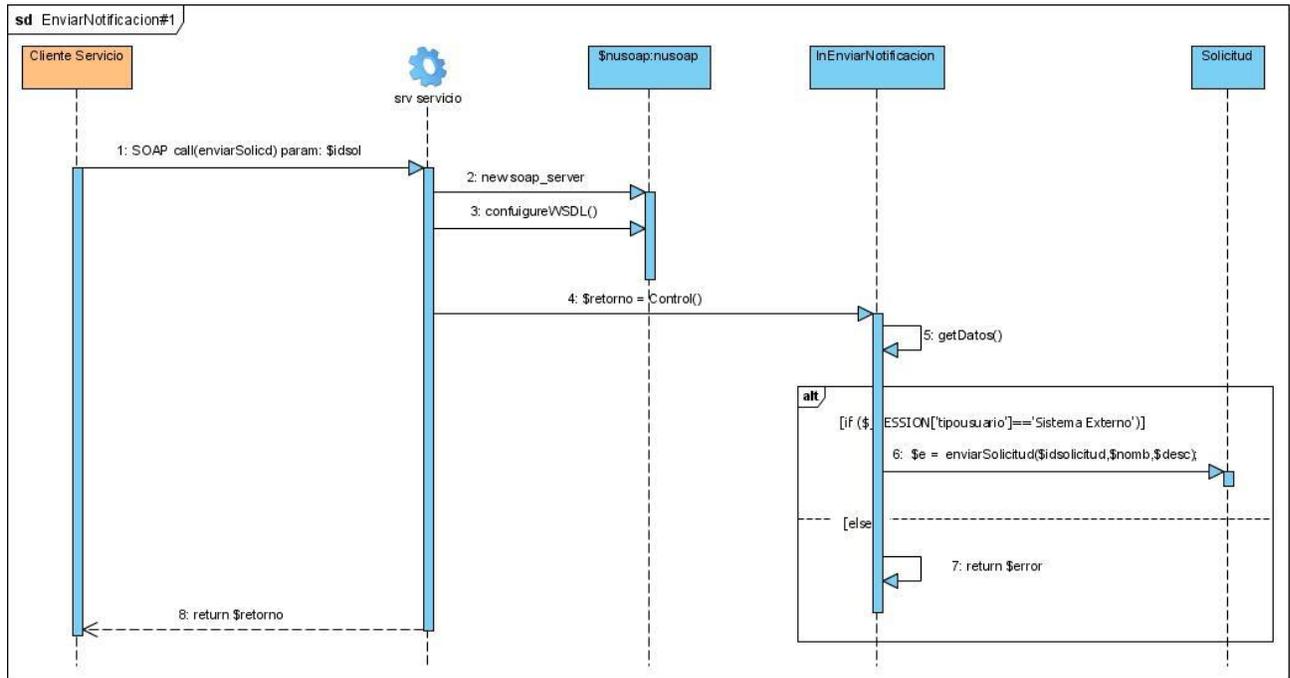


Figura # 40 Diagrama de Secuencia del Caso de Uso Enviar Notificación 1

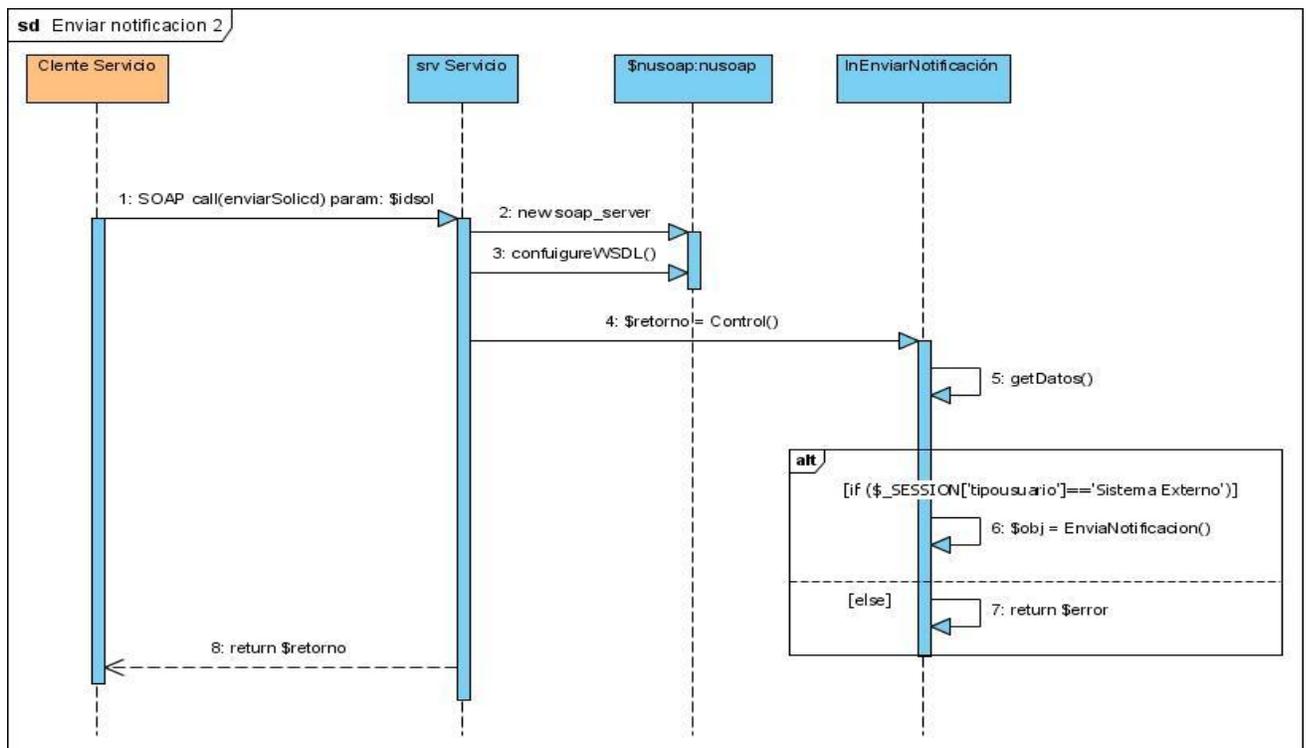


Figura # 41 Diagrama de Secuencia del Caso de Uso Enviar Notificación 2

Capítulo 3: Análisis y Diseño del Sistema

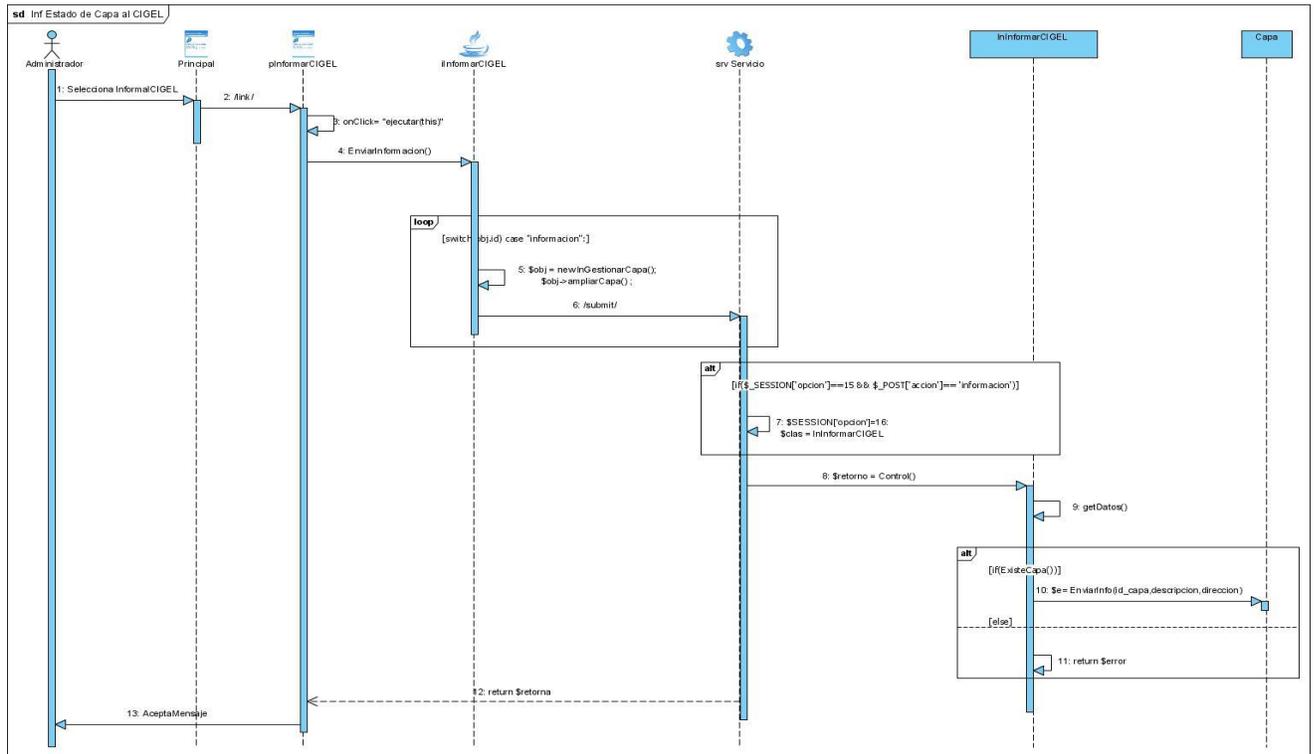


Figura # 42 Diagrama de Secuencia del Caso de Uso Informar Estado de Capa al CIGEL

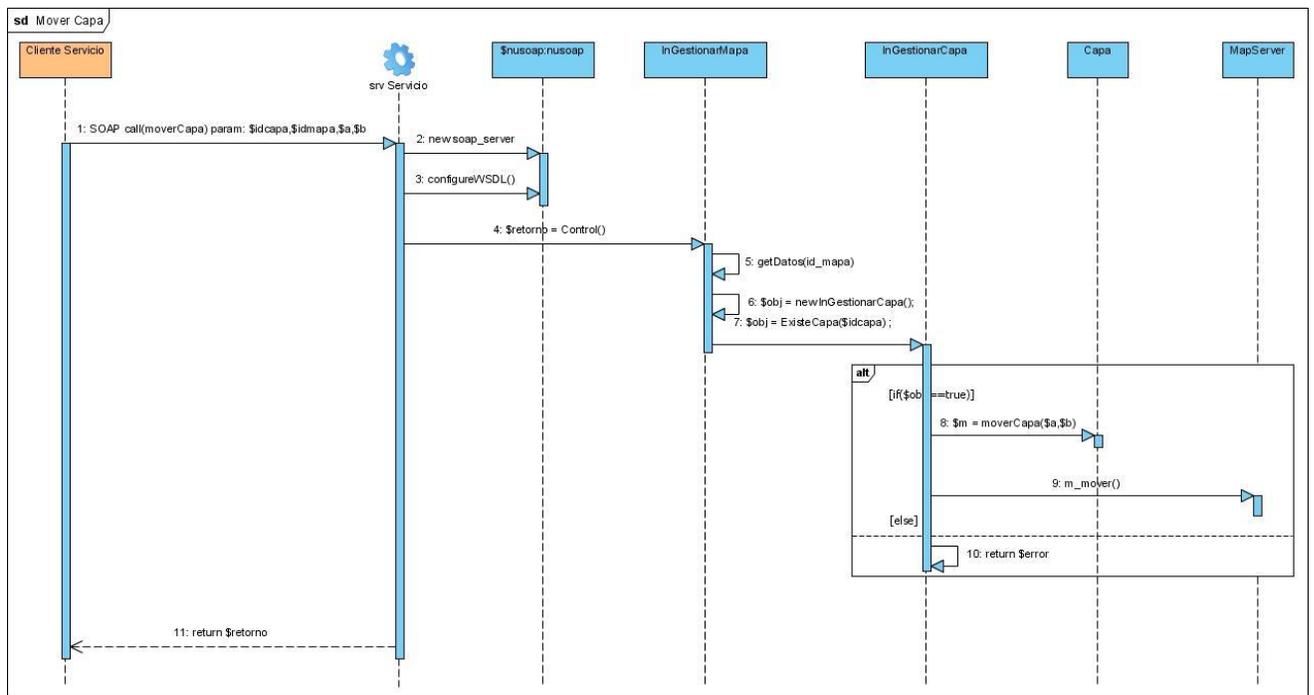


Figura # 43 Diagrama de Secuencia del Caso de Uso Mover Capa

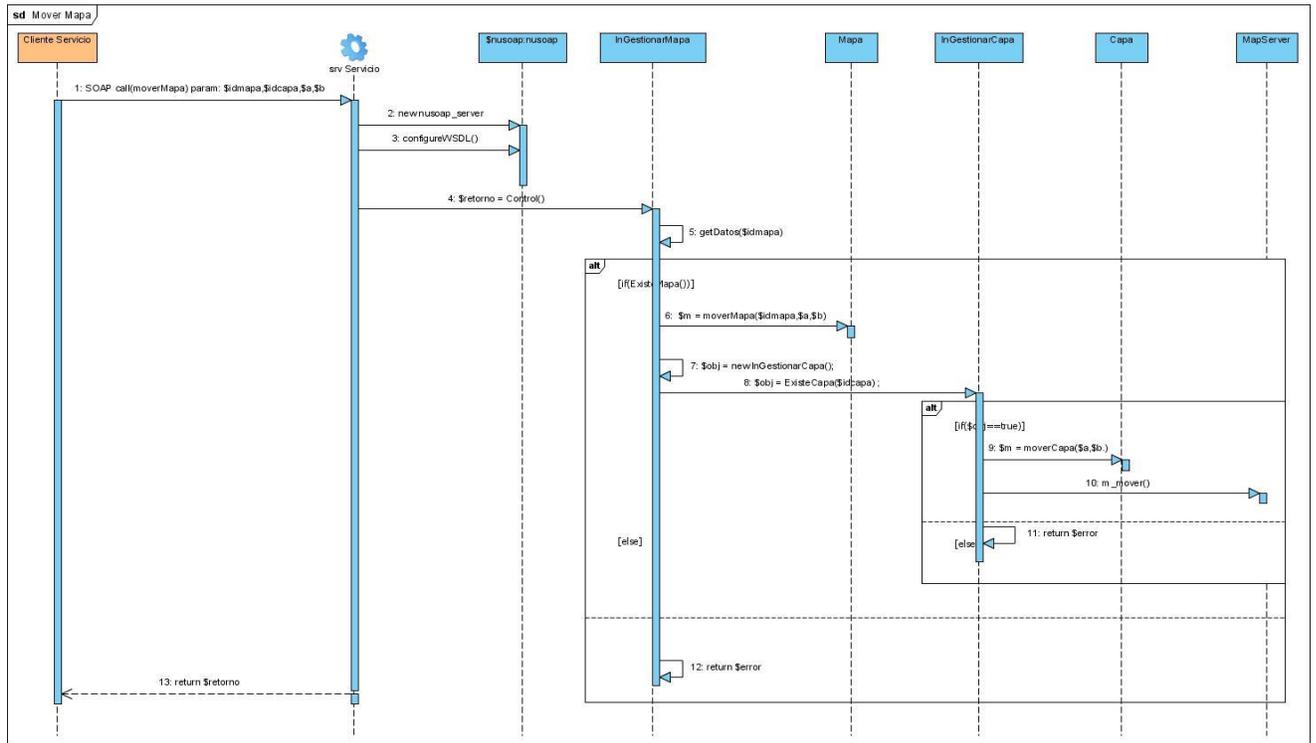


Figura # 44 Diagrama de Secuencia del Caso de Uso Mover Mapa

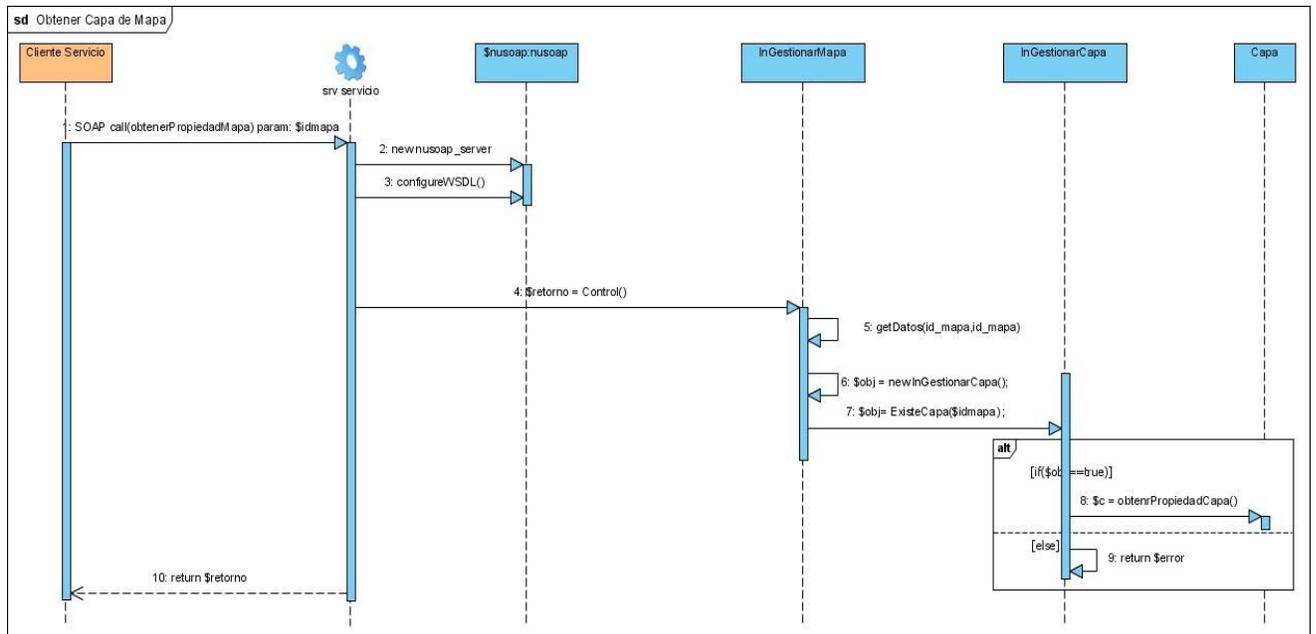


Figura # 45 Diagrama de Secuencia del Caso de Uso Obtener Capa de Mapa

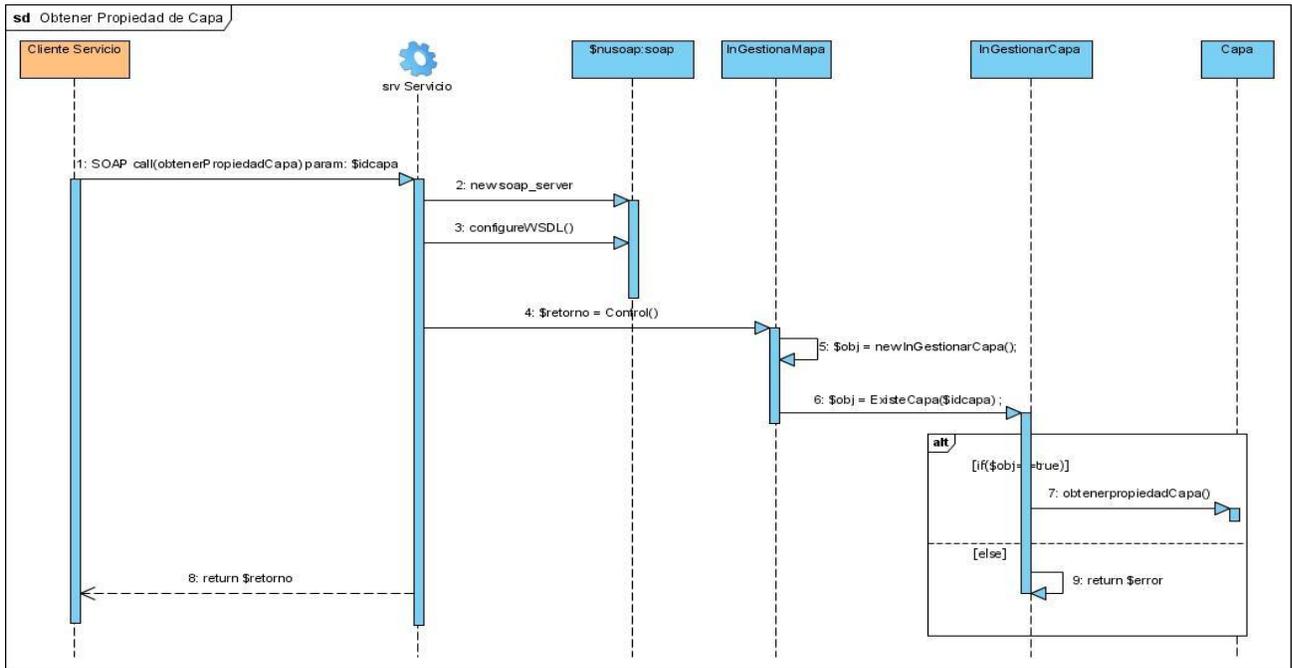


Figura # 46 Diagrama de Secuencia del Caso de Uso Obtener Propiedad de Capa

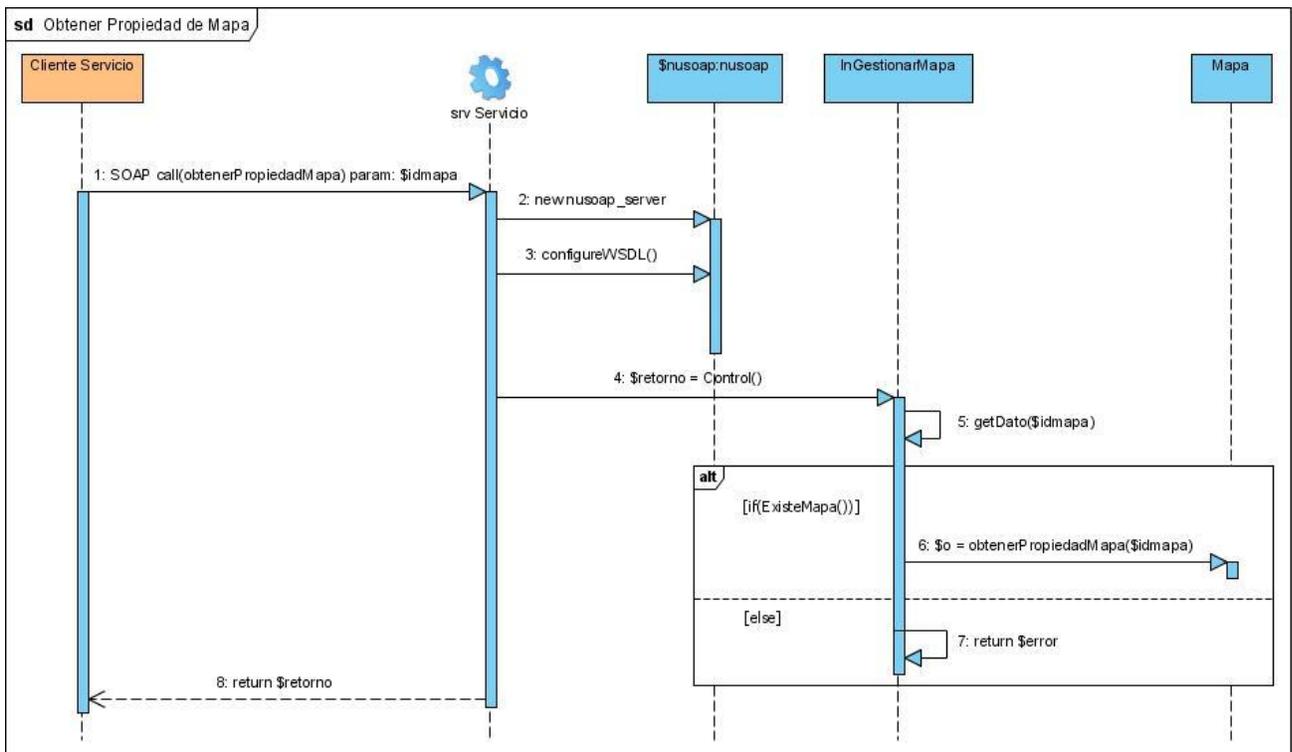


Figura # 47 Diagrama de Secuencia del Caso de Uso Obtener Propiedad de Mapa

Capítulo 3: Análisis y Diseño del Sistema

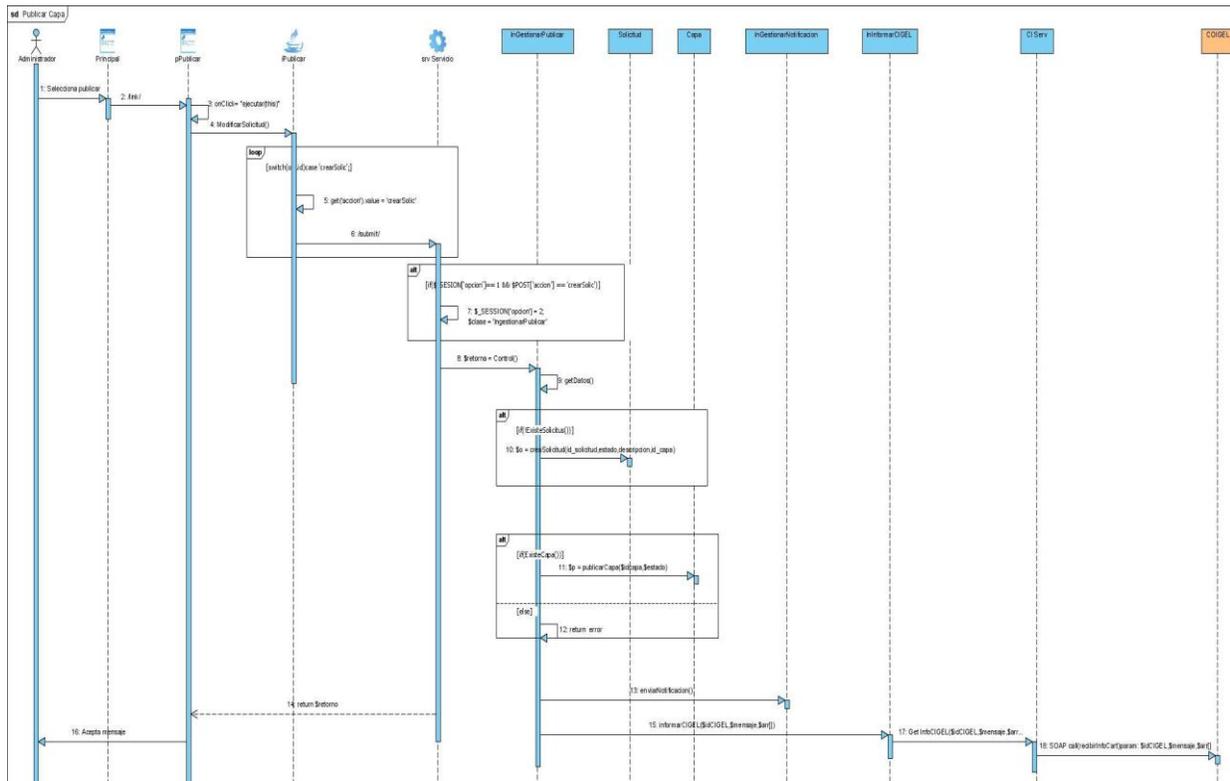


Figura # 48 Diagrama de Secuencia del Caso de Uso Publicar Capa

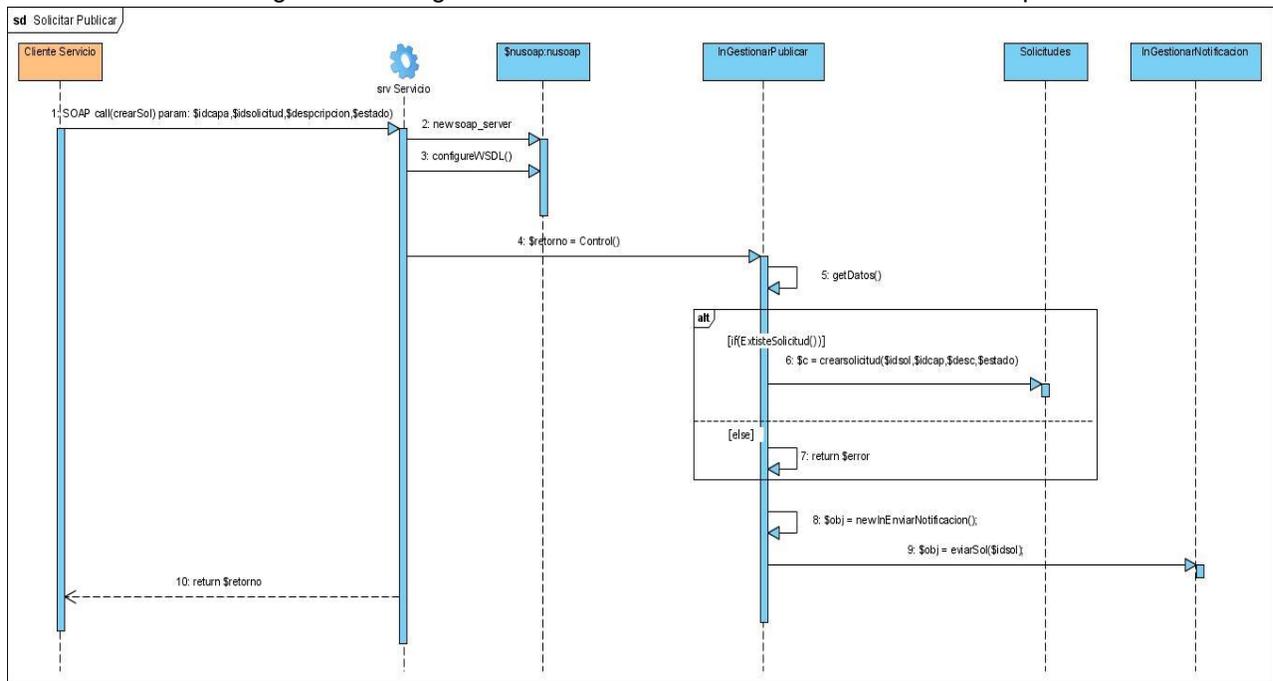


Figura # 49 Diagrama de Secuencia del Caso de Uso Solicitar Publicar Capa

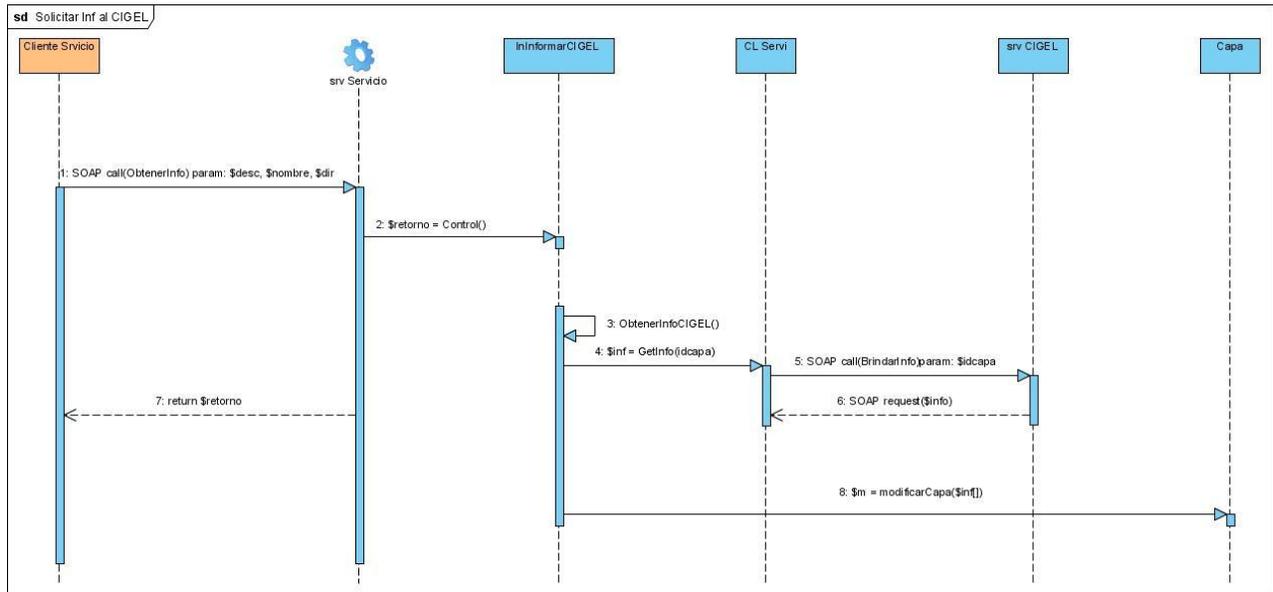


Figura # 50 Diagrama de Secuencia del Caso de Uso Solicitar Información del CIGEL

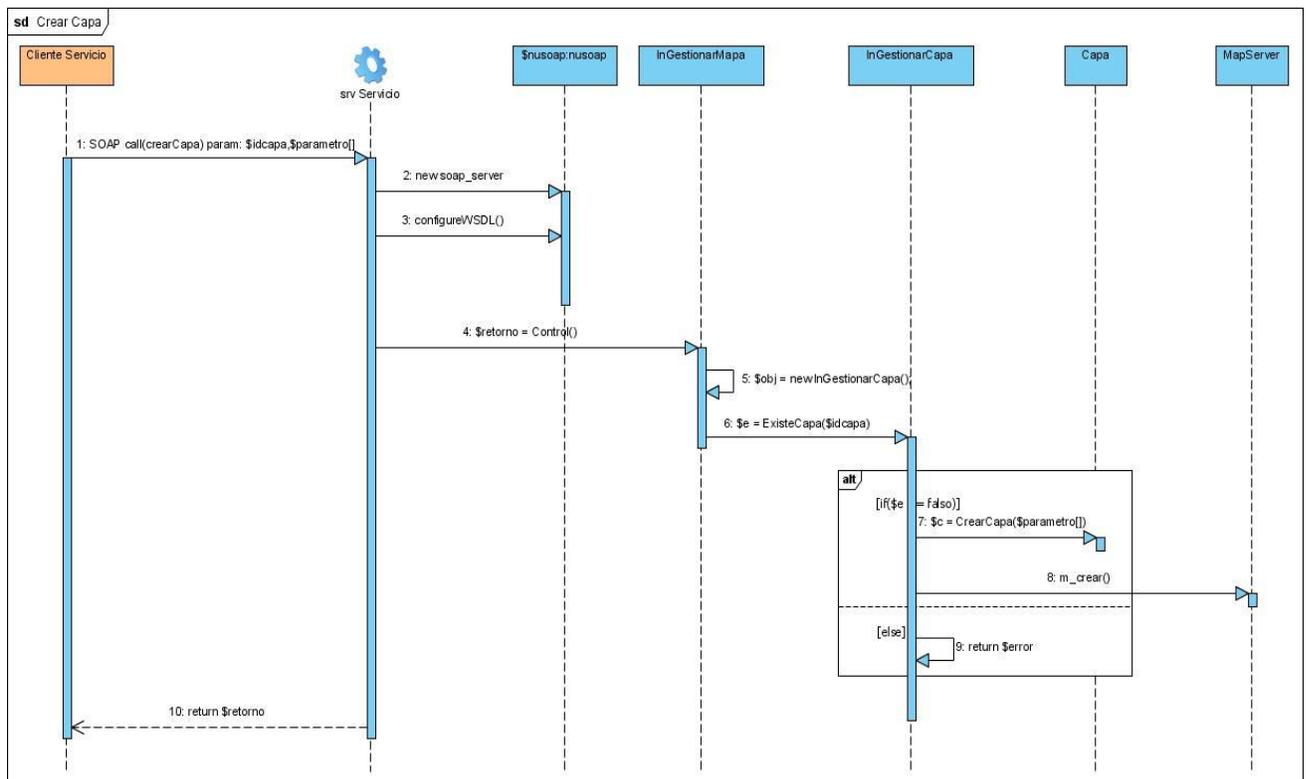


Figura # 51 Diagrama de Secuencia del Caso de Uso Crear Capa

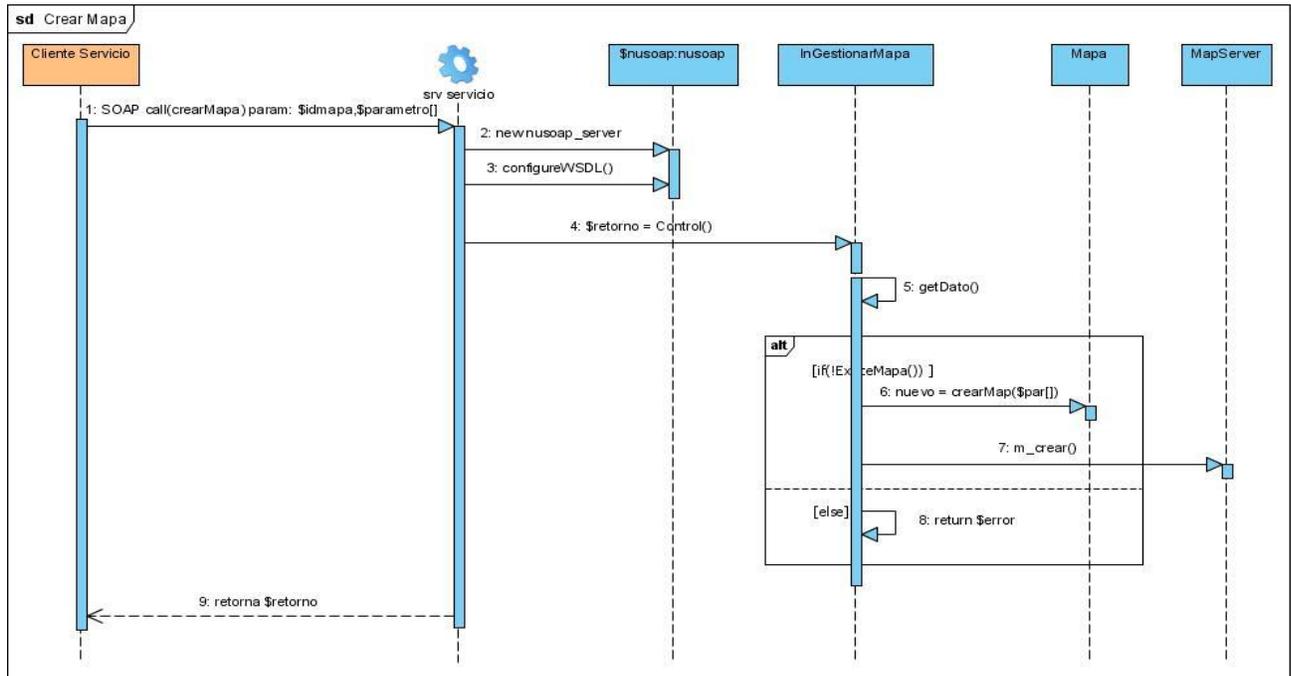


Figura #52 Diagrama de Secuencia del Caso de Uso Crear Mapa

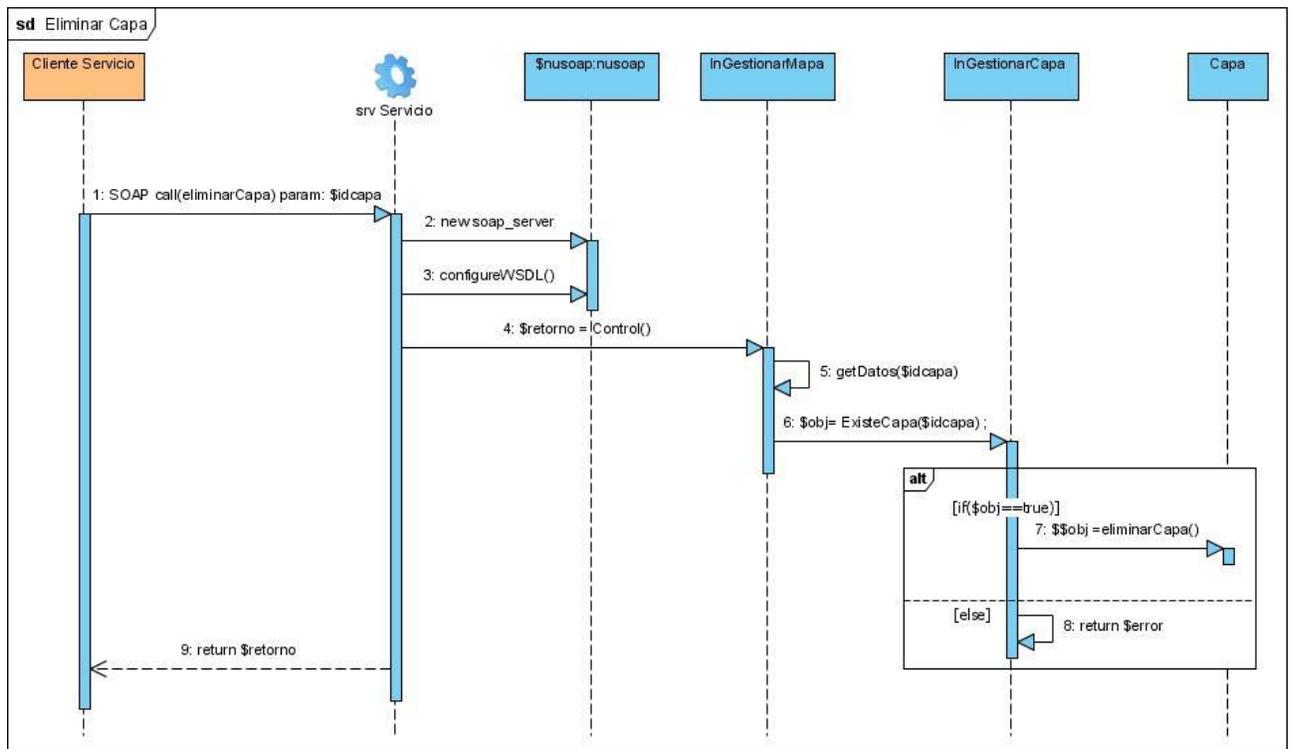


Figura # 53 Diagrama de Secuencia del Caso de Uso Eliminar Capa

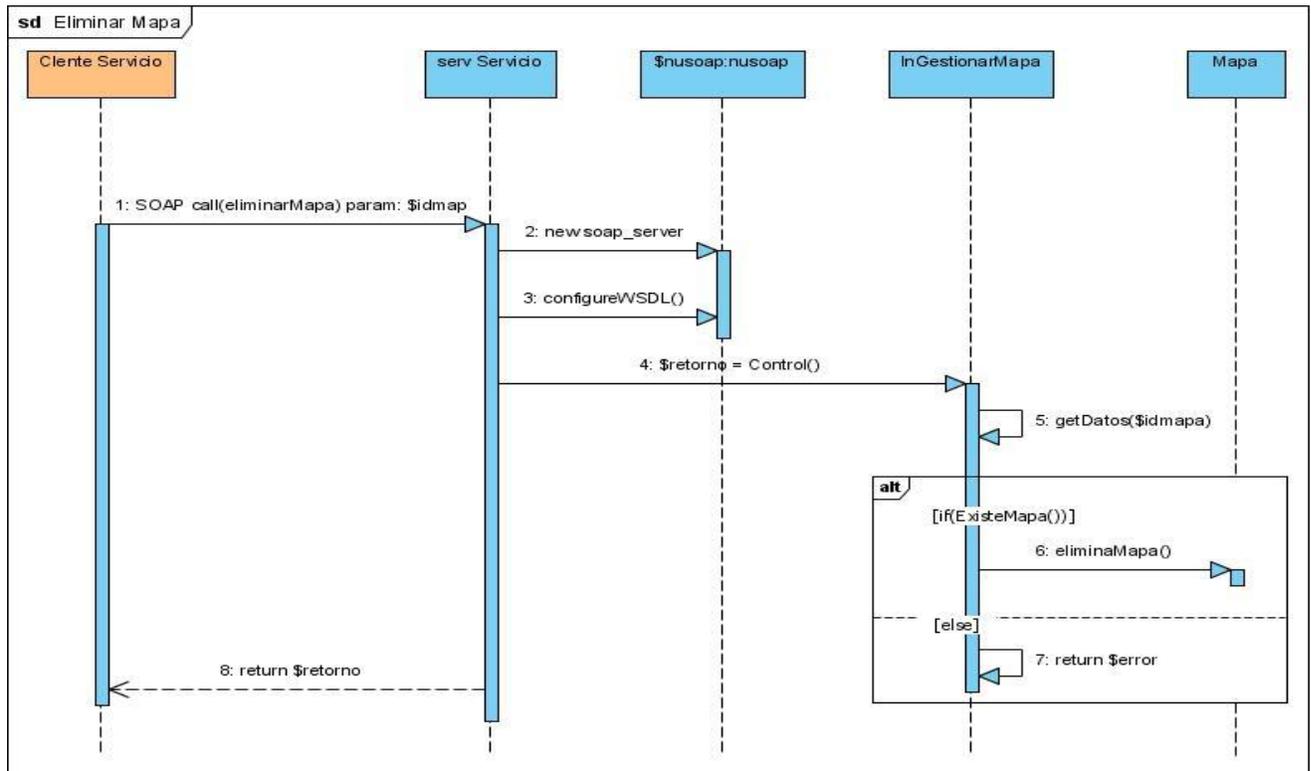


Figura # 54 Diagrama de Secuencia del Caso de Uso Eliminar Mapa

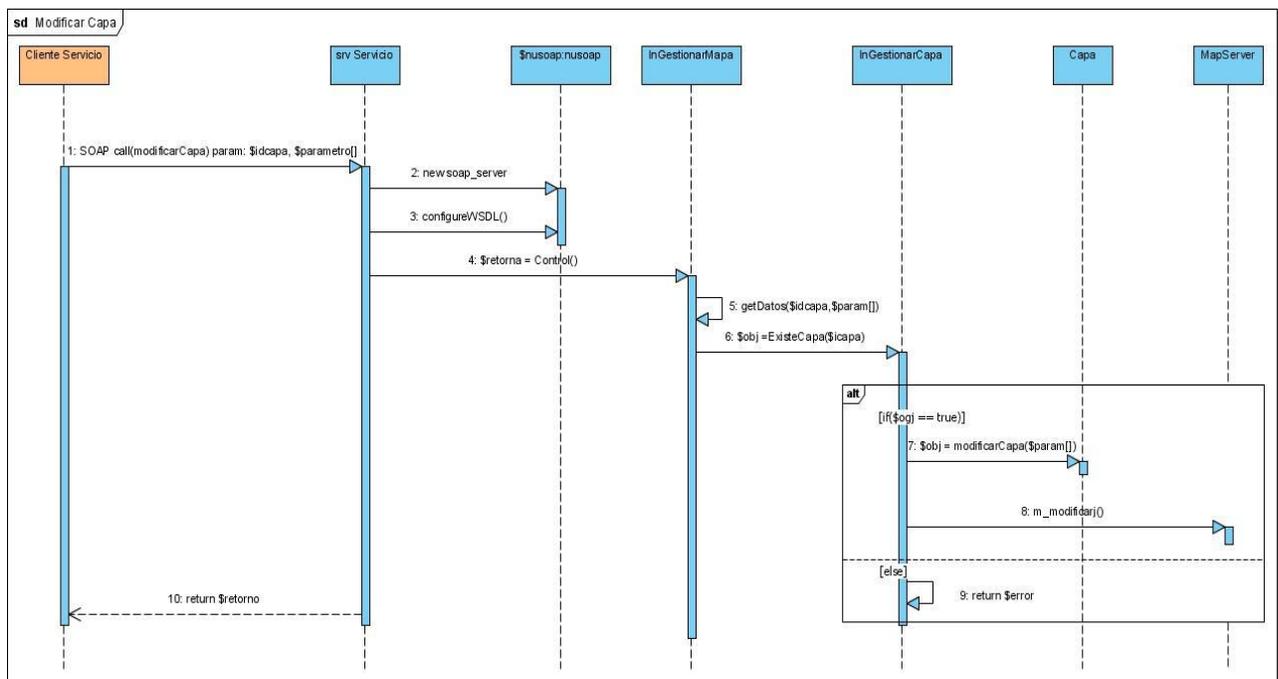


Figura # 55 Diagrama de Secuencia del Caso de Uso Modificar Capa

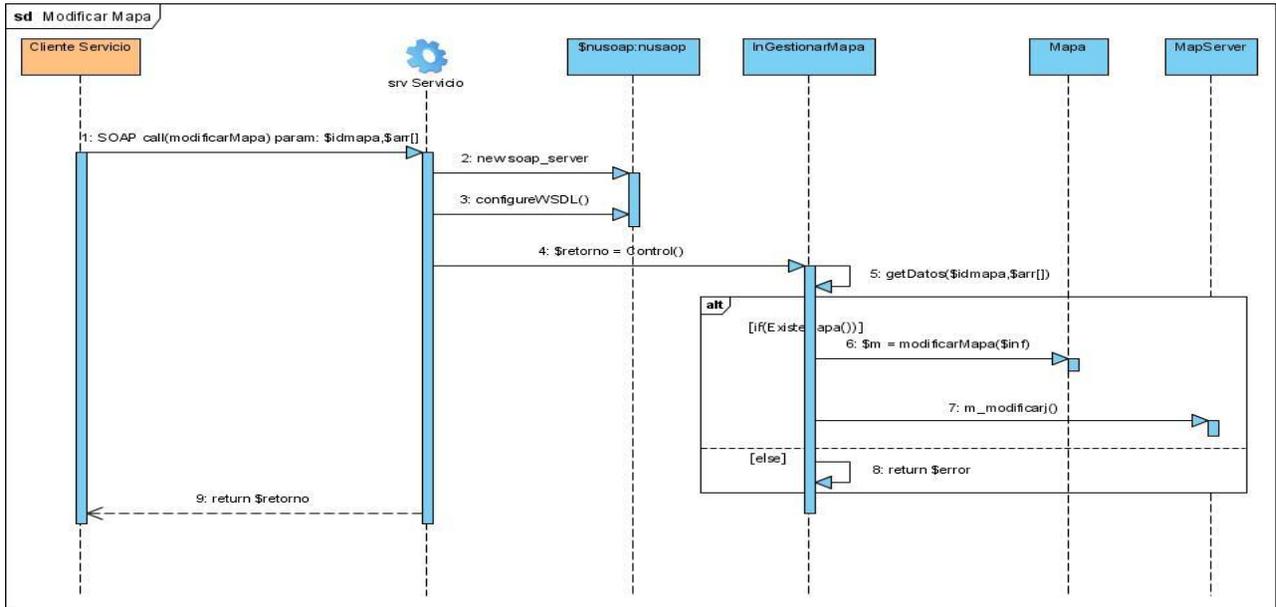


Figura # 56 Diagrama de Secuencia del Caso de Uso Modificar Mapa

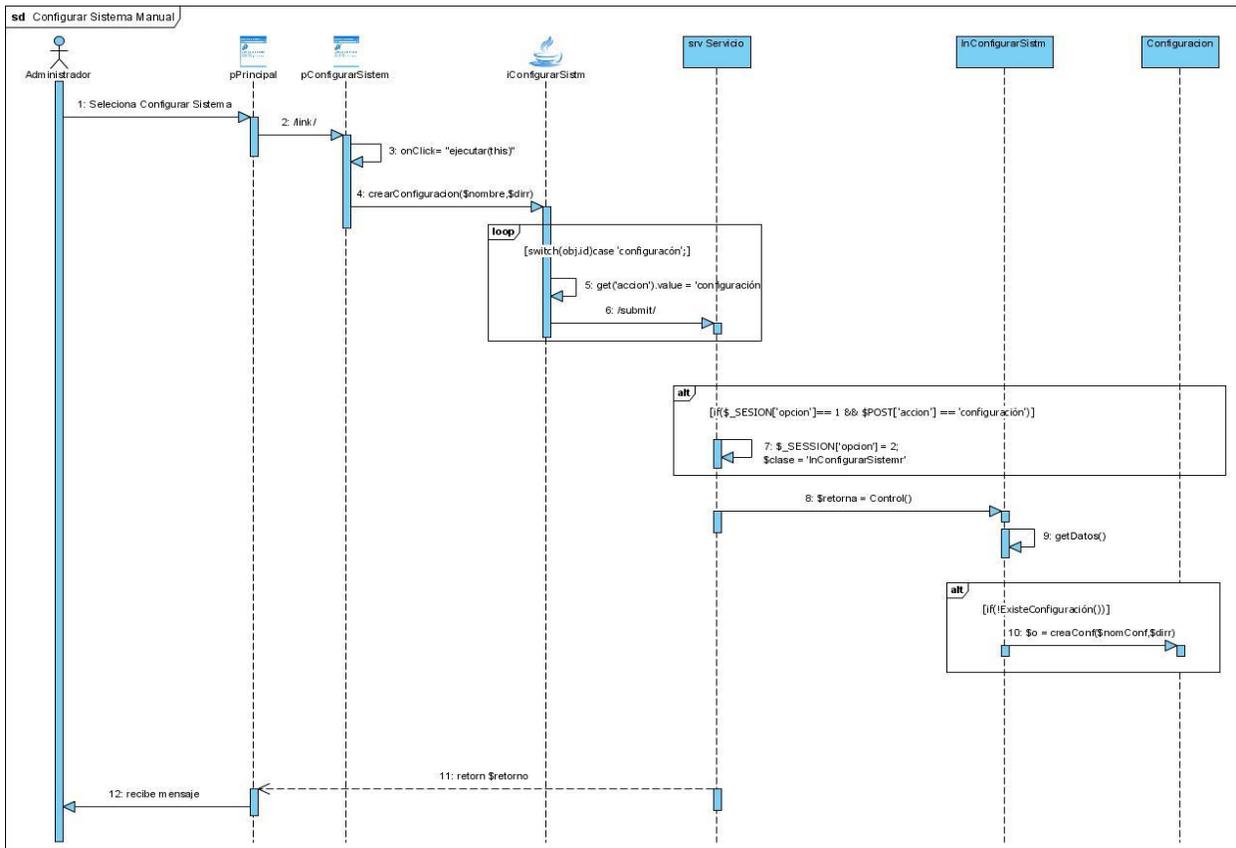


Figura # 57 Diagrama de Secuencia del Caso de Uso Configurar Sistema Manual

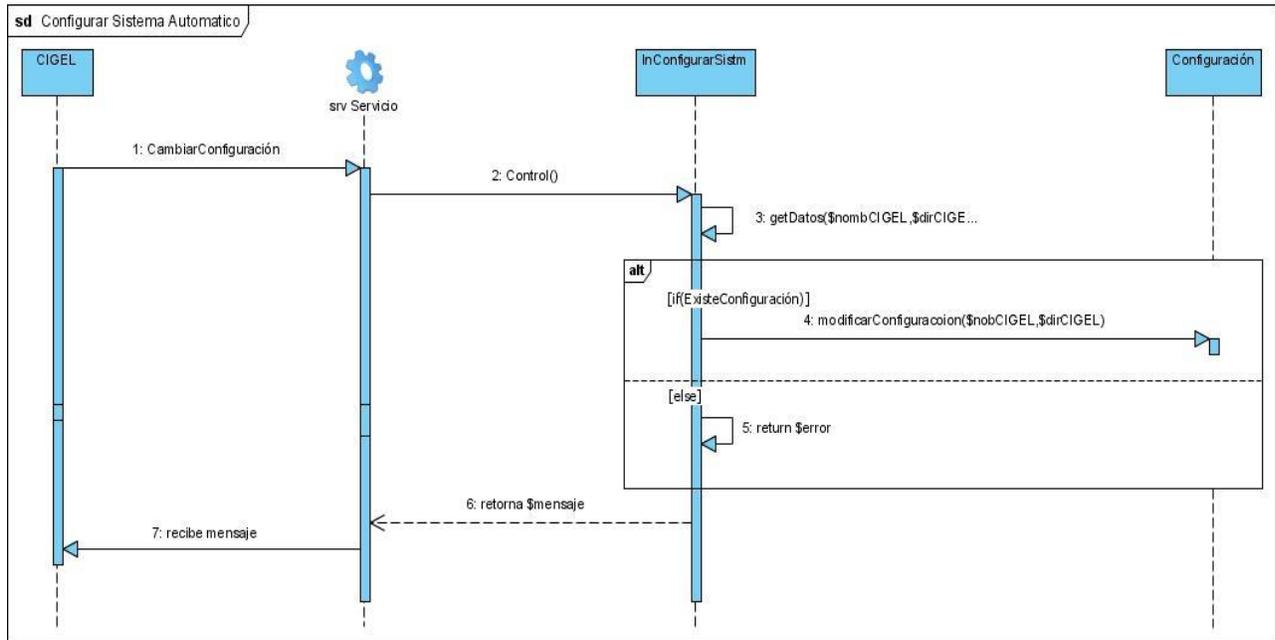


Figura # 58 Diagrama de Secuencia del Caso de Uso Configurar Sistema Automático

3.4. Descripción de las clases.

Clases	Descripción
[Plantillas]	Son las clases que contienen en su estructura todo el contenido estático de una página Web. Estas serán empleadas para generar las páginas clientes.
[Lógica del negocio]	Son las clases con prefijo In y encapsulan en sus métodos todas las operaciones específicas para cada caso de uso del sistema.
[Global]	Fichero con código PHP que contiene instancias y funciones que serán empleadas por todas las páginas servidoras.
[Entidades]	Son las clases existe una para cada entidad de la base de datos. Implementan las operaciones básicas personalizadas que se realizan sobre las entidades del sistema. (insertar, modificar, eliminar, y consultar)
[NuSOAP]	Es un kit de herramientas (Toolkit) para desarrollar Web Services bajo el lenguaje PHP, licenciada bajo LGPL. Está compuesto por una serie de clases que nos harán mucho más fácil el desarrollo de Web Services. NuSOAP está basado en SOAP 1.1,

	WSDL 1.1 y HTTP 1.0/1.1
[CL Serv]	Esta clase es la parte cliente del servicio Web, que se comunica directamente con la parte servidora y permite el acceso a los servicios disponibles. La clase Cl Serv implementa los métodos fundamentales de controlar capa, controlar mapa, gestionar mapa, gestionar capa, informar al CIGEL, solicitar información al CIGEL, solicitar publicar capa o publicar capa.
[srv Servicio]	Es la clase base del servicio, que implementa todas las funcionalidades del lado del servidor. Publica todas las funciones disponibles como servicio y brinda la posibilidad al usuario de saber cuáles de las funciones están disponibles y sus características mediante la lectura de un fichero generado en XML (WSDL).

4. Diseño de la Base de Datos

Para diseñar la base de datos del sistema, utilizamos el diagrama de clases persistentes y el modelo de datos. Las clases que modelamos en el diagrama de clases persistentes son aquellas que almacenan y obtienen datos durante los procesos de la aplicación.

4.1. Diagramas de Clases persistentes.



Figura # 58 Diagrama de Clases Persistentes

4.2. Modelo de datos.

cap_Capa		
+id_capa	integer(10)	Nullable = f...
id_mapa	integer(10)	Nullable = true
Permiso	integer(10)	Nullable = true
Nom_Capa	integer(10)	Nullable = true
Dirección	integer(10)	Nullable = true
Estilo	integer(10)	Nullable = true
Editable	integer(10)	Nullable = true
Activa	integer(10)	Nullable = true
geografica	integer(10)	Nullable = true
Sistem_Proyec	integer(10)	Nullable = true
Descripción	integer(10)	Nullable = true
Estado	integer(10)	Nullable = true

Figura # 59 Modelo de Datos

4.3. Descripción de las tablas.

Nombre: cap_capa		
Descripción: Esta tabla almacena datos del capa		
Atributo	Tipo	Descripción
Id_capa	serial	Este campo es Identificador de la tabla, viene de dat_capa
Nom_capa	varchar	Es el nombre de la tabla dat_mapa, también es identificador de la tabla.
Dirección_capa	varchar	Es la dirección donde esta ubicado el mapa.
Id_capa	serial	Otro identificador de la tabla, fecha de inicio del curso
Descripción_capa	varchar	Es la descripción que tienes la capa.
sisteProy	varchar	Es el sistema de proyección que con el que se mostrara el mapa

padre	varchar	Es la conexión del CIGEL al que solicita información.
geografica	varchar	Es donde se guardan todo los datos geográficos del mapa.(línea, polilínea, polígono, etc)
activa	integer	Es para mostrar o no la capa en el mapa(si es 1 esta activa y si es 0 esta desactiva)
Estado_capa	varchar	El estado en que se encuentra la capa.
Permiso	varchar	Los usuarios que tienen permiso a la capa.

5. Principios de diseño.

El diseño del sistema es la parte del mismo con que el usuario interactúa directamente; por lo que consideramos que es una parte fundamental dentro de nuestro proceso de desarrollo.

De modo general nuestros usuarios solo están interesados en cómo se le muestra la interfaz de la aplicación y en cómo usarla, o sea que entre los mismos en el uso de las nuevas tecnologías prevalece un nivel medio; aunque en cuanto a la experiencia y conocimiento del proceso que se automatiza la capacidad de los estos es muy amplia y su interés en la aplicación informática la evaluamos de positiva; por lo que nos planteamos las siguientes características para el diseño de nuestro sistema:

- Los elementos mostrados en la pantalla no serán numerosos y en caso de repetirse alguno en otras interfaces, estos serán mostrados en la misma posición en todas para familiarizarlo con el sistema.
- Las páginas que muestran información, lo harán en el mismo orden y de manera similar.
- Cada elemento se diseñará siguiendo un patrón de tamaño, formas y colores (no serán ni fuertes ni brillantes).

6. Mecanismos de diseño.

Con el propósito de hacer el diseño un sistema más objetivo, claro y eficiente para la futura implementación proponemos la utilización de los mecanismos de diseño para la seguridad y la

persistencia en el diseño una aplicación de administración y servicios web para el manejo de la información geoespacial; por lo que con la utilización del Proceso Unificado de Modelado y mecanismos de diseño hemos realizado un conjunto de diagramas (de clases y secuencia) que agrupan aspectos comunes.

7. Seguridad

El diseño de nuestro sistema de servicios web para representar la información geoespacial como parte del ERP de las FAR cuenta con una serie de subsistemas, dentro del mismo encontramos el de Seguridad; aprovechando la interfaz cControlacceso que el subsistema nos aporta para acceder al servicio Web que este brinda para la autenticación de los usuarios y apoyándose en su método público logueo proponemos como solución para la implementación de la seguridad el uso de un servicio Web encargado del control de los accesos, autenticación y registro de los eventos que ocurren en el mismo.

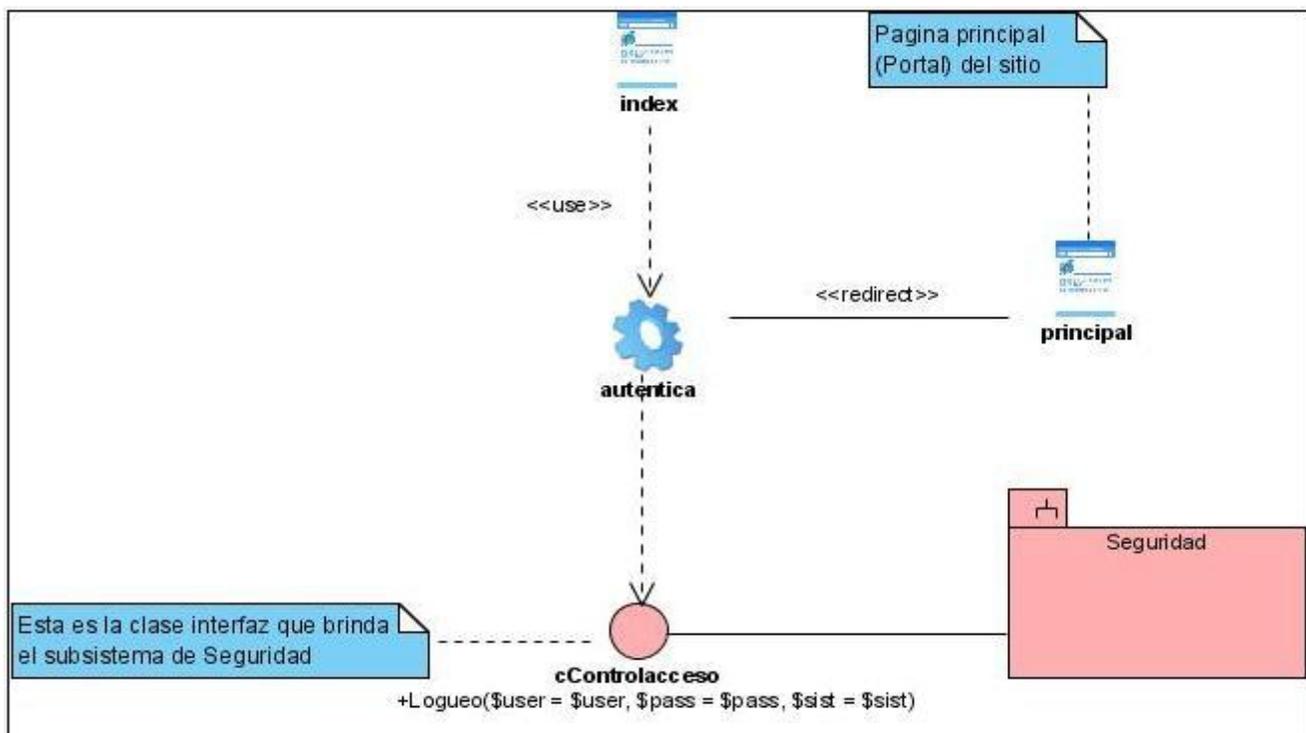


Figura # 60 Mecanismo de Diseño de la seguridad

8. Interfaz.

Proponemos que las interfaces sean amigables, sencillas, agradables y fáciles de usar. Contará con un banner de presentación en la cabecera no muy grande que identificará a la aplicación. Se trabajará con las familias de fuentes: Verdana, Arial, Helvetica, Sans-serif; el tamaño de la fuente no debe diferir mucho de 16 px y los colores los trabajaremos sobre tonalidades claras basados fundamentalmente en blanco y azul.

9. Tratamiento de errores

De manera general los errores serán generados por funciones JavaScript para evitar la ejecución de la página procesos cuando existan errores. Este es el caso de los formularios de inserción/actualización, donde se validara la entrada correcta de los datos, además se incorporan las clases .js que se encargan de la validación de la entrada de los datos en los formularios. En el caso de las eliminaciones se muestra mensaje de confirmación, de manera que no se realice una operación no deseada.

10. Ayuda

La aplicación cuenta con un material complementario que se pone a disposición de los usuarios para el correcto manejo de la misma. Un manual avanzado de uso para el usuario, el mismo permite el dominio eficiente de cada una de las funcionalidades que brinda la aplicación y así un correcto aprovechamiento de sus potencialidades.

11. Conclusiones

Con el desarrollo de este capítulo se ha logrado obtener un diseño del sistema que sirve de base para su Implementación.

En este capítulo se ha llevado a cabo la descripción de las clases y demás elementos necesarios para la implementación. Se obtuvo el diagrama de clases del sistema. Se definieron las clases persistentes y a partir de esto, se construyó el modelo de datos. Se expusieron los servicios que se brindarían y la apariencia que proponíamos que tenga las interfaces de administración para el diseño de la interfaz, y se explicó cómo está estructurada la aplicación físicamente, mediante los modelos de despliegue y de componentes.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS

1. Introducción

Este capítulo se centra principalmente en la elaboración del Modelo de Implementación propuesto conjuntamente con el Diagrama de Despliegue de la aplicación. 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 modelo de despliegue.

2. Diagrama de Componentes.

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. Se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del *software*, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Los elementos de modelado dentro de un diagrama de componentes serán componentes y paquetes.

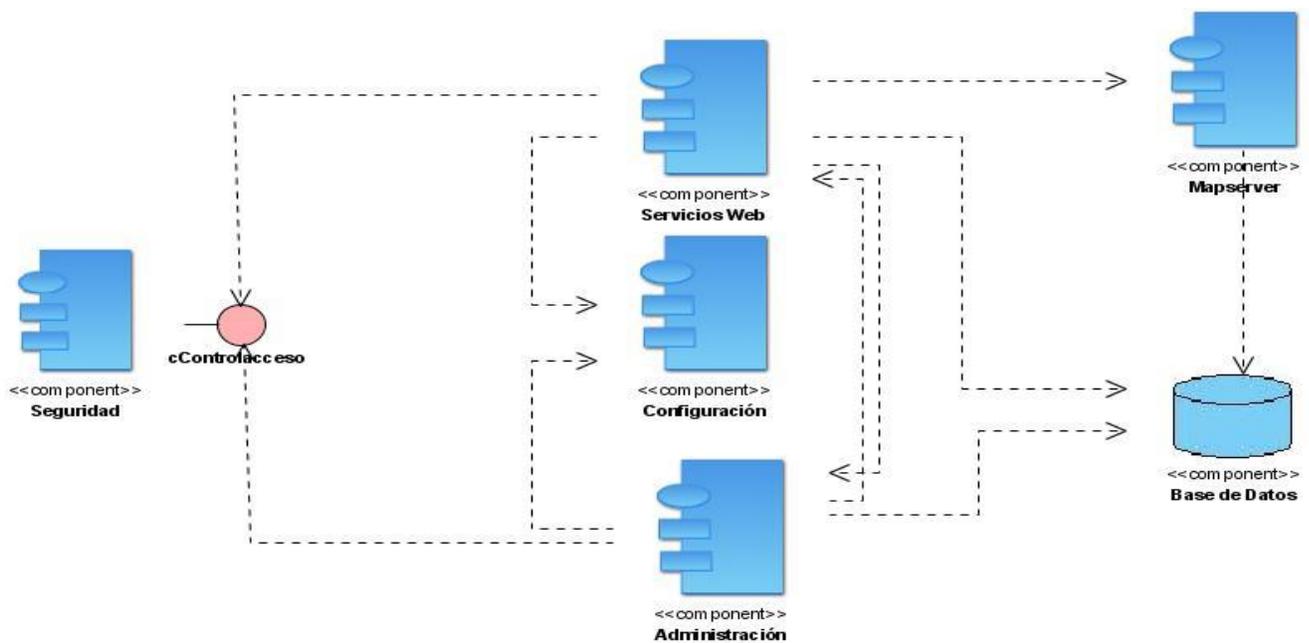


Figura # 58 Diagrama de Componentes Físico



Figura # 59 Diagrama de Componentes paquete de Administración



Figura # 60 Diagrama de Componentes paquete de MapServer

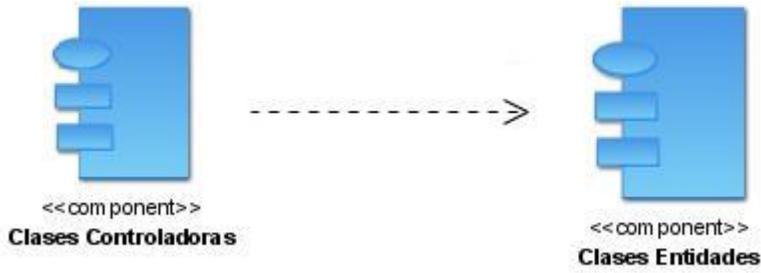


Figura # 61 Diagrama de Componentes paquete de Servicios Web



Figura #62 Paquete de Servicios Web

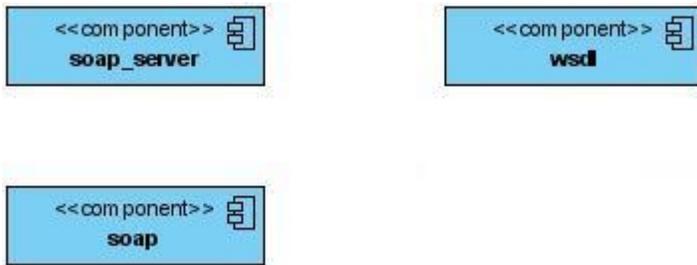


Figura # 63 Diagrama de Componentes paquete de NuSoap

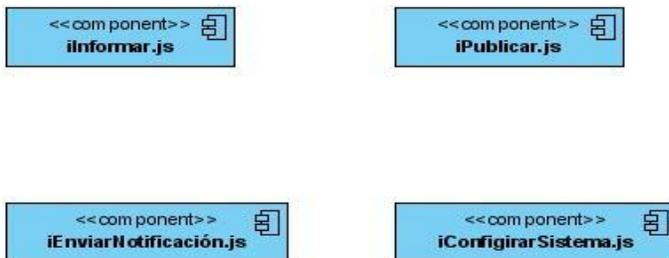


Figura # 65 Diagrama de Componentes paquete JS

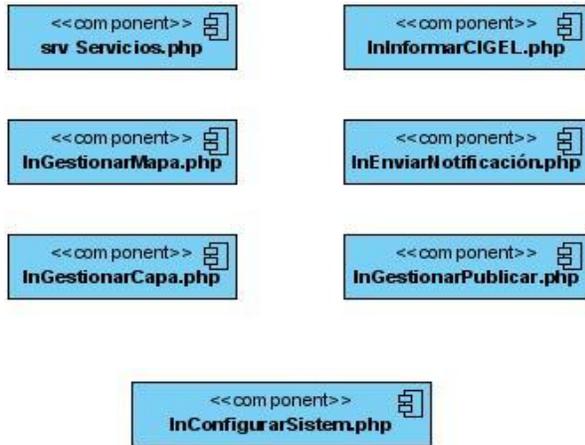


Figura 66: Diagrama de componentes paquete Clases Controladoras.

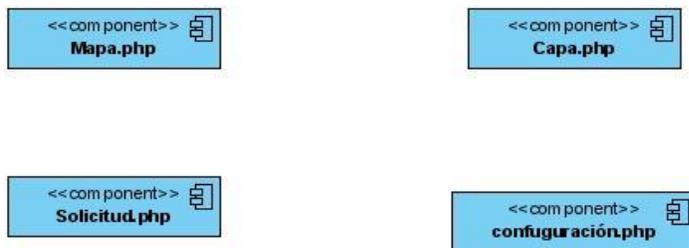


Figura 67: Diagrama de Componentes Paquete Clases Entidades.

3. Diagrama de Despliegue.

El modelo de despliegue describe la distribución física del sistema, muestra como están distribuidos los componentes de software entre los distintos nodos de cómputo y permite además comprender la correspondencia entre la arquitectura software y la arquitectura hardware.

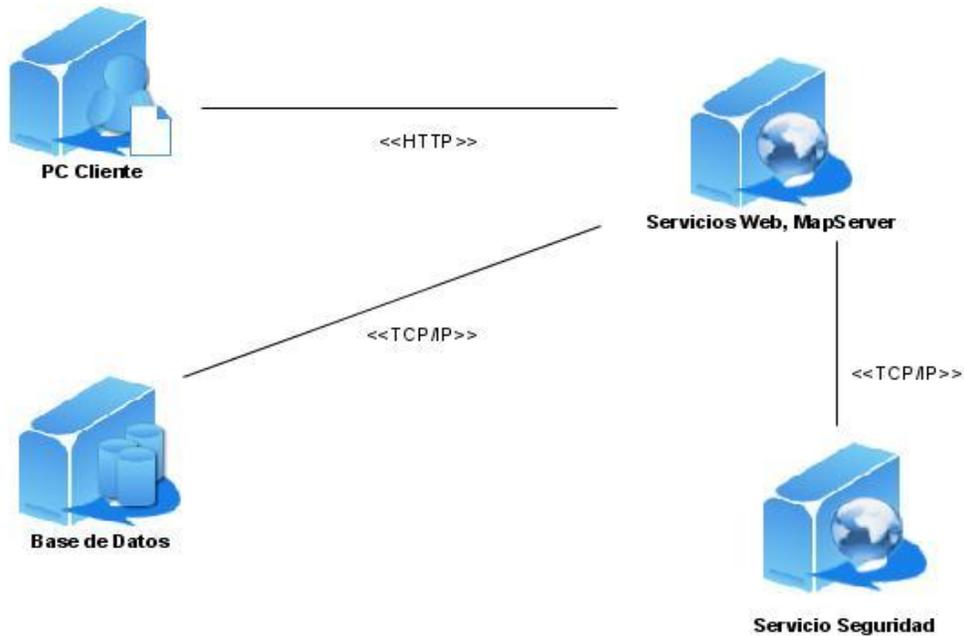


Figura # 68 Diagrama de Despliegue

4. Conclusiones.

En este capítulo mostramos los diagramas de componentes y el diagrama de despliegue del flujo de trabajo de implementación, que con ellos se tiene una mejor visión para su posterior implementación. Con la culminación de este capítulo dejamos plasmado todos los pasos que el desarrollador debe seguir para una buena implementación de la aplicación y los servicios Web que modelamos.

CONCLUSIONES GENERALES

Luego del estudio realizado y del correspondiente diseño del sistema se logró la modelación del sistema de servicios web para controlar de manera centralizada la representación de la información de forma geoespacial. Además se hace de una manera accesible a todos los sistemas que están y estarán en funcionamiento dentro de la institución, ya que se garantiza interoperabilidad entre las diferentes aplicaciones de gestión de información. Todo el sistema se representó mediante tecnología RUP y se modelaron todos los diagramas y representaciones necesarias.

El desarrollo de esta aplicación constituye un aporte práctico muy importante, debido a que es una novedad tecnológica que marca un proceso de avance en el desarrollo de software multiplataforma e interoperable dentro de la entidad.

De aquí la conclusión de que el trabajo ha dado cumplimiento a los objetivos propuestos satisfactoriamente y se hacen además las recomendaciones para un trabajo futuro que mejore la calidad de nuestro proyecto.

RECOMENDACIONES

- Extender el uso de las aplicaciones basadas en los servicios Web para otras funcionalidades dentro de la entidad.
- Fomentar el uso de aplicaciones distribuidas multiplataforma para la gestión de la información en el ambiente empresarial.
- Profundizar en las ventajas del uso de los servicios Web como alternativas en software de control y gestión de la información de forma geoespacial.
- Recomendamos la terminación del módulo de administración propuesto para mejorar la seguridad y uso extensible de la aplicación.
- Añadir las funcionalidades para lograr mejores resultados y mantener a la organización a la vanguardia de la tecnología en cuanto a Sistemas de Información Geográfica.
- Recomendamos la implementación de este sistema para comenzar a explotarlo lo antes posible en la organización.

BIBLIOGRAFÍA

BIBLIOGRAFÍA CITADA

[1] CORDÓN, Nuria. *Los Sistemas de Información Geográfica se hacen populares en la empresa*. Computerworld, 2007 [consultado en: mayo 2007] Disponible en: <http://www.idg.es/computerworld/articulo.asp?id=300536762>

[2] JIMENEZ, José A.; AGUILERA, Ma. Jesús; MEROÑO, José E. *ALTERNATIVAS DE SOFTWARE LIBRE A LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA COMERCIALES*. [consultado en: Mayo 2007] Disponible en: <http://www.cartesia.org/geodoc/ingegraf2005/gis10.pdf>

[3] BATISTA, José Luis. *APLICACIÓN DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA EN CUBA*. [consultado en: enero 2007] Disponible en: http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=1051

[4] GUINEA, Alejandro; JORRÍN, Sergio. *Arquitectura SOA para la integración entre software libre y software propietario en entornos mixtos*. [consultado en: abril 2007] Disponible en: <http://www.sigte.udg.es/JornadasSIGLibre/comun/1pdf/13.pdf>

[5] VAN DER HENST, Christian. *¿Qué es el PHP?* [consultado en: mayo 2007] Disponible en: <http://www.maestrosdelweb.com/editorial/phpintro/>

[6] ALFARO, Ricardo. *¿Por qué elegir PHP?* [consultado en: mayo 2007] Disponible en: <http://www.mmug.cl/articulos.php?id=283>

[7] WORSLEY, John; DRAKE, Joshua. *PostgreSQL Práctico*. [consultado en: mayo 2007] Disponible en: <http://www.sobl.org/traduccion/practical-postgres/node12.html>

BIBLIOGRAFÍA CONSULTADA

- ✓ Jacobson, Ivar; Booch, Grady; Rumbaugh James. *El proceso unificado de software*. Volumen I. La Habana, 2004. Editorial Feliz Varela
- ✓ Jacobson, Ivar; Booch, Grady; Rumbaugh James. *El proceso unificado de software*. Volumen II. La Habana, 2004. Editorial Félix Varela

- ✓ Pressman, Rogers. *Ingeniería del software*. Parte 1. La Habana, 2005. Editorial Félix Varela.
- ✓ Pressman, Rogers. *Ingeniería del software*. Parte 2. La Habana, 2005. Editorial Félix Varela.
- ✓ Carmona, Alvaro de J., Monsalve, Jhon Jairo. [consultado en: enero 2007] *Sistemas de Información Geográficos*. Disponible en: <http://www.monografias.com/trabajos/gis/gis.shtml>
- ✓ Vrsalovic, Vinko. *USA POSTGRESQL Y POSTGIS PARA DESARROLLAR GIS (SIG)* [consultado en: diciembre 2006] Disponible en: http://www.postgresql.cl/entrevistas/v_vrsalovic.htm
- ✓ [Álvarez] - Álvarez, Sofía, Hernández Anaisa. Metodología para el desarrollo de aplicaciones con tecnología Orientada a Objetos utilizando notación UML. La Habana, 2000
- ✓ [ASP] - ¿Qué es eso de asp? <http://www.pobladores.com/territorios/informatica/asp>. (17/01/2002)
- [Cocomoll] - Boehm, B. et al 2000 "Software Cost Estimation with COCOMO II"
- ✓ [Dealer] - Dealer World, la revista de referencia para el canal. <http://www.idg.es/dealer> (15/01/2002)
- ✓ [e-Contact] - Las Mesas de Ayuda. Revista e-Contact. Julio-Agosto. 2002. <http://www.imt.com.mx/revista/numero19>. (15/01/2002)
- ✓ [Farell] - Tourniaire, Françoise y Farell, Richard. *The Art of Software Support*. Prentice Hall. 1998.
- ✓ [Febles-a] - MSc. Febles Estrada, Aylin. Presentación en Power Point. Clase de Soporte de Software. Asignatura ADS. 5to Año. Curso 2001-2002.
- ✓ [Febles-b] - MSc. Febles Estrada, Aylin. "Case Corporativo para el proceso de control de cambios" Tesis presentada en opción al título de Master en Informática Aplicada, Ciudad de la Habana, 2001.
- ✓ [Hernández] - Hernández, Yanko e Banderas, Ideal. "Case para la planificación y control de configuración de software", Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico "José Antonio Echeverría", Ciudad de la Habana, 2001.
- ✓ [Herrera] - Herrera J., Lizka Johany. "*Ingeniería de Requerimientos – Ingeniería de Software*", <http://www.monografias.com/trabajos6/resof> (04/01/2002)
- ✓ [Huidobro] - Huidobro Moya, José Manuel. *Redes y Servicios de Telecomunicaciones*. Segunda Edición.
- ✓ [Internet] - Colectivo de autores. *Secretos de Internet*. Editorial Academia, La Habana 2001.

- ✓ [Jones] - JONES, Capers. Programming Language Table Release 8.2, March 1996.
<http://www.spr.com/library> (17/01/2002)
- ✓ [JS-a] - Guía de JavaScript
<http://developer.netscape.com/docs/manuals/communicator/jsguide4>. (17/01/2002)
- ✓ [JS-b] - Manual de JavaScript <http://www.redestb.es/soporte/aula/jScript>. (17/01/2002)
- ✓ [JS-c] - Sitio de Macromedia <http://www.macromedia.cl/products>. (17/01/2002)
- ✓ [Matos] - Matos, Rosa María. Introducción al trabajo con Base de Datos. Asignatura de Sistemas de Gestión de Base de Datos.
- ✓ [Morales] - Morales Moliner, Alex. Conceptos básicos de ASP
http://www.programacion.com/asp/tutoriales/asp_basics. (17/01/2002)
- ✓ [MySQL-a] - Manual de referencia de MySQL <http://www.mysql.com>. (17/01/2002)
- ✓ [MySQL-b] - Tutorial de MySQL <http://www.aplushosting.com/spanish/tutorials>. (17/01/2002)
- ✓ [PHP] - Introducción a php. <http://www.ciberteca.net/webmaster/php>. (17/01/2002)
- ✓ [Pressman] - Pressman, Roger S. Ingeniería de Software, un enfoque práctico. Cuarta Edición
- ✓ [Rational] - "Lo nuevo de Rational Rose 2000" . Rational Corporation, 2000.
<http://www.abists.com.mf/Fabs/Rational/notasTK>. (17/01/2002)
- ✓ [Rodríguez] - Rodríguez, Daniel y Bravo, Joaquín. Tutorial de HTML
<http://html.programacion.net>. (17/01/2002)
- ✓ [RUP] - Rational Unified Process, Rational Software Corporation, "Rational Unified Process", Version 2001A.04.00, Copyright 1987-2001.
- ✓ [Salas] - Salas, Juan Francisco. La satisfacción del cliente: una estrategia empresarial. Febrero 2001. <http://www.cinterac.com.ar>. (15/01/2002)
- ✓ [Sánchez] - Sánchez Cavazos, Yolanda del Carmen. Satisfacción del cliente e inteligencia corporativa. Universidad de Monterrey. <http://www.mex-i-co.com/scip>. (15/01/2002)
- ✓ [Satisfacción] - Satisfacción del cliente. <http://www.ceocant.es/documentosvarios>. (15/01/2002)
- ✓ [SQL-a] - Manual de SQL <http://www.lobocom.es/~claudio>. (17/01/2002)
- ✓ [SQL-b] - Introducción a SQL <http://www.lafacu.com/apuntes/informatica/sql>. (17/01/2002)
- ✓ [SQL-c] - SQL <http://www.arsys.es/soporte/programacion>. (17/01/2002)
- ✓ [SQL-d] - Manual de SQL <http://walter.freesevers.com>. (17/01/2002)
- ✓ [UML-a] - Tutorial de UML <http://www.dcc.uchile.cl/~psalinas/uml>. (17/01/2002)

GLOSARIO

SIG: sistema de hardware, software y procedimientos diseñados para soportar la captura, administración, manipulación, análisis, modelamiento y graficación de datos u objetos referenciados espacialmente, para resolver problemas complejos de planeación y administración. Es un sistema de computador capaz de mantener y usar datos con localizaciones exactas en una superficie terrestre. Es una herramienta de análisis de información.

WMS (Web Map Services): El servicio Web Map Service (WMS) definido por el OGC (Open Geospatial Consortium) produce mapas de datos espaciales referidos de forma dinámica a partir de información geográfica.

WFS (Web Features Services): Servicios de rasgos (características) en Web. Especificación emitida por OGC.

WCS (Web Coverages Services): Servicios de Coberturas en Web. Especificación emitida por OGC.

GML (Geographic Markup Language): Lenguaje estándar de marcado basado en XML para el intercambio de información geográfica entre sistemas.

(SVG) Scalable Vector Graphics: Lenguaje de marcado basado en XML para la representación de gráficos vectoriales en páginas Web

OGC (Open GIS Consortium). Consorcio 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.

Internet: Red de computadoras alrededor de todo el mundo que comparten información unas con otras por medio de páginas o sitios.

UML: Lenguaje Unificado de Modelado. Es el lenguaje de modelado de sistema de software más conocido en la actualidad.

PHP (Personal Home Page) Es un lenguaje interpretado de alto nivel impregnado en páginas HTML y ejecutado en el servidor.

SGBD Sistemas de gestión de bases de datos.

Datos Alfanuméricos: Son descripciones de las características de las entidades gráficas. Generalmente son almacenados en formatos convencionales para este tipo de información.

Georreferenciación: Es el posicionamiento en el que se define la localización de un objeto espacial en un sistema de coordenadas.

Cartografía: Es una disciplina que integra ciencia, técnica y arte, que trata de la representación de la Tierra sobre un mapa o representación cartográfica.

DCOM: (Distributed Component Object Model) es una tecnología propietaria de Windows para desarrollar componentes software distribuidos sobre varios ordenadores y que se comunican entre sí.

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