

**Universidad de las Ciencias Informáticas  
Facultad 3**



**Componente de posicionamiento global y rutas del Sistema  
de Gestión Integral de Aduanas**

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

**Autor:** Yanerkys Cabrera Martínez

**Tutores:** Ing. Yisel Rodríguez Pérez

Ing. Andy Fernandez Garabote

**12 de junio del 2013**



"EL TIEMPO Y LA REFLEXIÓN

VAN MODIFICANDO

PAULATINAMENTE NUESTRA

VISIÓN, HASTA QUE POR

ÚLTIMO, LLEGAMOS A

COMPRENDER".

## DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo al Centro de la Informatización de la Gestión de Entidades (CEIGE) de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Yanerkys Cabrera Martínez

---

Yisel Rodríguez Pérez

---

Andy Fernandez Garabote

## DATOS DE CONTACTO

**Autora: Yanerkys Cabrera Martínez**

**Universidad de las Ciencias Informáticas, La Habana, Cuba**

**Email: [ycabrera@estudiantes.uci.cu](mailto:ycabrera@estudiantes.uci.cu)**

**Tutores:**

**Ing. Yisel Rodríguez Pérez**

**Universidad de las Ciencias Informáticas, La Habana, Cuba**

**Email: [yrperez@uci.cu](mailto:yrperez@uci.cu)**

**Ing. Andy Fernandez Garabote**

**Universidad de las Ciencias Informáticas, La Habana, Cuba**

**Email: [agarabote@uci.cu](mailto:agarabote@uci.cu)**



## AGRADECIMIENTOS

**A mis padres** quienes son mi razón de ser, por haber sido mis guías, apoyo e inspiración; a mi mamá que siempre estuvo a mi lado, por sufrir mis problemas y esforzarse cada día para que yo lograra mis sueños; a mi papá que siempre está apoyándome en todas mis decisiones y celebrando cada uno de mis logros como si fueran propios. Por ser los mejores padres del mundo, los adoro.

**A mi abuelita** que tanto quiero y que siempre está pendiente de mí, que indudablemente es la guía e inspiración de toda la familia.

**A mis tías** por ser mis cómplices y ayudarme en todo lo que he necesitado. **A mis tíos** que tanto quiero y **a mis primos** que forman parte de cada una de mis travesuras. **A mi padrastro** por siempre estar pendiente de mis cosas. En general **a toda mi familia**, de la cual estoy orgullosa de formar parte, la cual adoro y que como dice Pimpinela “se pelean por un mes, pero cuando las cosas van mal a tu lado siempre están”.

**A mi novio** que en este tiempo juntos he aprendido a quererlo por su apoyo incondicional y los lindos momentos vividos.

**A mis amigas de toda la vida** Yisel y Evelyn que realmente las quiero mucho y siempre les deseo lo mejor en sus vidas.

**A mi familia de Matanzas** que nunca podré olvidar, y que me dieron tanto apoyo cuando más lo necesité.

Gracias **a mis compañeros de aula** por todo este tiempo que hemos pasado juntos y por todas las experiencias que hemos compartido. A Yanet, Tania y Yaneisi por ser mis mejores amigas desde primer año. A Nolvis, Analie, Gisel, Helen que la vida realmente me ha enseñado a quererlas. **A mi primo** Lisvany y a Marbe muchas gracias por siempre apoyarme en los momentos más difíciles. A todos mis hermanitos de aula, de verdad nunca los voy a olvidar y siempre van a ocupar un lugar especial en mi mente.

**A mis tutores** Andy, Yisel y Céspedes que realmente no los cambiaría por nadie, por estar siempre pendientes de mí, ayudándome y defendiéndome durante todo el camino.

**Gracias al tribunal y oponente** por tratarme con profesionalidad y justicia, por brindarme su experiencia y aconsejarme en cada uno de los cortes.

DEDICATORIA

*A mis padres con todo mi amor,  
Por haberse dedicado a mi formación,  
Por confiar plenamente en mí,  
Y apoyarme en todo momento, para realizar este  
gran sueño...*

*Yanerkys Cabrera Martínez.*

## **RESUMEN**

Las Tecnologías de la Información y las Comunicaciones son cada vez más utilizadas para la creación de herramientas de apoyo al proceso de toma de decisiones, producto a la gran cantidad de información existente en los principales sectores de la sociedad y la economía, siendo difícil gestionar la misma; entre estas herramientas se encuentran los Sistemas de Información Geográfica (SIG). El presente trabajo desarrolla un SIG mediante el uso de herramientas de software libre. El mismo facilita el proceso de ubicación geográfica de lugares y rutas sin la necesidad de consumir servicios de georeferencias que se encuentran públicos en el mundo.

La implementación de este SIG surge con el objetivo de ser utilizado como herramienta de apoyo a la toma de decisiones en cualquier sistema que necesite de sus funcionalidades y que cumpla con dos precondiciones importantes: estar desarrollado sobre el marco de trabajo Symfony versión 1.x y contener información necesaria para el proceso de ubicación geográfica de los lugares. Además permite la ubicación de lugares, rutas, así como la posibilidad de observar la descripción de los lugares ubicados en un mapa global. También permite controlar el mapa, moverlo en la dirección que desee y acercarlo o alejarlo a través del mouse o del teclado como más conveniente resulte. Estas funcionalidades contribuyen a una mejor organización de la información relacionada con datos geográficos, así como al ahorro de tiempo y menores probabilidades de desacierto en cuanto a la ubicación geográfica.

## **PALABRAS CLAVE**

Sistemas de información geográfica, toma de decisiones, ubicación geográfica.

## ÍNDICE

INTRODUCCIÓN .....	1
1. CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1.    Introducción .....	5
1.2.    Conceptos asociados al dominio del problema .....	5
1.3.    Sistemas de Información Geográfica .....	6
1.4.    Análisis de otras soluciones existentes.....	10
1.4.1.  Ámbito internacional.....	10
1.4.2.  Ámbito nacional.....	12
1.5.    Selección de las tecnologías a utilizar para la implementación .....	14
1.6.    Entorno de trabajo a utilizar .....	17
1.7.    Herramientas de desarrollo.....	18
1.8.    Modelo de desarrollo de software .....	19
1.9.    Conclusiones parciales .....	21
2. CAPÍTULO 2: ANÁLISIS Y DISEÑO DE LA SOLUCIÓN.....	22
2.1.    Introducción.....	22
2.2.    Modelo de dominio o conceptual .....	22
2.3.    Requisitos funcionales .....	23
2.4.    Requisitos no funcionales .....	25
2.5.    Validación de los requisitos de software.....	26
2.6.    Arquitectura de software .....	27
2.7.    Patrones arquitectónicos .....	28
2.8.    Diagrama de clases del diseño con estereotipos web.....	30
2.9.    Diagrama de secuencia .....	32
2.10.   Métricas para la evaluación del diseño .....	33
2.11.   Conclusiones parciales.....	34
3. CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA.....	35
3.1.    Introducción.....	35
3.2.    Precondiciones para utilizar el componente.....	35
3.3.    Funcionamiento del componente desarrollado.....	36



3.4.	El componente orientado a la arquitectura del GINA.....	36
3.5.	Diagrama de componentes.....	37
3.6.	Diagrama de despliegue.....	39
3.7.	Interfaz de usuario.....	39
3.8.	Pruebas de software.....	42
3.9.	Resultados de las pruebas.....	46
3.10.	Conclusiones parciales.....	48
CONCLUSIONES.....		49
RECOMENDACIONES.....		50
GLOSARIO.....		51
REFERENCIA BIBLIOGRÁFICA.....		53
ANEXOS.....		56
	Anexo 1: Descripción de requisitos.....	56
	Anexo 2: Modelo Conceptual.....	62

## ÍNDICE DE ILUSTRACIONES

Ilustración 1. Sistema de Información Geográfica [35]. .....	7
Ilustración 2. Diferencias entre Modelo raster y Modelo vectorial. ....	9
Ilustración 3. Portada de SIGUCI. ....	13
Ilustración 4. Ciclo de vida de proyectos del CEIGE [12]. ....	20
Ilustración 5. Modelo de Dominio. ....	22
Ilustración 6. Arquitectura del componente a desarrollar. ....	27
Ilustración 7. El patrón Modelo Vista Controlador [18]. ....	29
Ilustración 8. Diagrama de clases del diseño para requisito funcional “Cargar mapa del mundo”. ....	31
Ilustración 9. Diagrama de clases del diseño para el requisito funcional "Localizar lugares". ....	31
Ilustración 10. Diagrama de secuencias del requisito funcional “Cargar mapa del mundo”. ....	32
Ilustración 11. Diagrama de secuencias del requisito funcional “Localizar lugares” ....	33
Ilustración 12. El componente sfGinaMaps orientado a la arquitectura del GINA. ....	37
Ilustración 13. Diagrama de componentes. ....	38
Ilustración 14. Diagrama de componentes físicos de la aplicación. ....	38
Ilustración 15. Diagrama de despliegue. ....	39
Ilustración 16. Pantalla principal. ....	40
Ilustración 17. RF Ubicar lugares y RF Mostrar descripción de lugares. ....	40
Ilustración 18. RF Mostrar rutas ascendentes, RF Leyenda y RF Mapa de referencia. ....	41
Ilustración 19. Dibujar líneas y Dibujar polígonos. ....	42
Ilustración 20. Resultados de las pruebas. ....	47
Ilustración 21. Resultados de las iteraciones. ....	48

## ÍNDICE DE TABLAS

Tabla 1. Requisitos funcionales. ....	24
Tabla 2. Tamaño operacional de clase (TOC).....	34
Tabla 3. Categoría que ocupa la clase mapas. ....	34
Tabla 4. Resumen de los diseños de casos de prueba. ....	46
Tabla 5. Resultados de la primera iteración. ....	47
Tabla 6. Descripción textual del requisito Cargar mapa del mundo. ....	56
Tabla 7. Descripción textual del requisito Mover el mapa.....	57
Tabla 8. Descripción textual del requisito Acercar el mapa. ....	58
Tabla 9. Descripción textual del requisito Alejar el mapa. ....	58
Tabla 10. Descripción textual del requisito Mostrar mapa de referencia.....	58
Tabla 11. Descripción textual del requisito Ver todo el mapa. ....	59
Tabla 12. Descripción textual del requisito Realizar acción anterior. ....	59
Tabla 13. Descripción textual del requisito Realizar acción siguiente.....	60
Tabla 14. Descripción textual del requisito Ubicar lugares. ....	60
Tabla 15. Descripción textual del requisito Mostrar escala del mapa. ....	60
Tabla 16. Descripción textual del requisito Mostrar rutas ascendentes. ....	61
Tabla 17. Descripción textual del requisito Mostrar leyenda.....	61
Tabla 18. Descripción textual del requisito Mostrar descripción de lugares.....	62
Tabla 19. Descripción textual del requisito Mostrar información de los botones utilizados. ....	62

## INTRODUCCIÓN

La ubicación geográfica es un tema muy valorado en la actualidad, debido a la importancia que tiene la relación existente entre los sucesos de la vida y el reconocimiento del lugar donde se ocasionan. Cada día aumentan los recursos para conocer la localización de un sitio en base a sus características geográficas. En la actualidad existen varios tipos de mapas los cuales tienen el objetivo de facilitar conocimientos exactos acerca del tema que tratan. Los más importantes son los mapas físicos, que muestran el relieve de una zona o de diferentes regiones y los mapas políticos, que tienen el objetivo de mostrar la división política del mundo.

Para las personas que están relacionadas con las Tecnologías de la Información y las Comunicaciones (TIC) existen, además de una variedad inigualable de mapas digitales, los Sistemas de Información Geográfica (SIG). Los mismos son herramientas de apoyo a la toma de decisiones y algunos autores, como el profesor Bosque Sendra los define como:

La integración organizada de hardware, software y datos geográficos diseñados para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada; con el fin de resolver problemas complejos de planificación y gestión [3].

Cuba no está ajena al tema de la informatización por lo que se encuentra inmersa en este proceso en todos los campos de la sociedad que le sea posible: salud, educación, ciencia, cultura, entre otros [1].

En el año 2004 una de las entidades que comprendió la necesidad de estos servicios fue la Aduana General de la República (AGR), ya que maneja grandes cúmulos de información que son de gran relevancia para el país. Por esta razón, en estos momentos está atravesando el proceso de informatización en conjunto con la Universidad de las Ciencias Informáticas (UCI), desarrollando el Sistema de Gestión Integral de Aduanas (GINA).

Para los trabajadores de la AGR el proceso de localización de lugares es muy complejo. Tienen que realizar el mismo en mapas no digitales, por lo que se podrían cometer errores de ubicación. Es prácticamente imposible que los trabajadores, teniendo conocimientos básicos de geografía, conozcan la ubicación exacta de más de 971 puertos y 9461 aeropuertos.

Otro problema que presentan los especialistas aduaneros es a la hora de localizar las rutas en un mapa no digital, ya que es más complicado analizar la ruta exacta de una travesía cuando sus recorridos son a lugares que se encuentran relativamente cerca, pues estos mapas tienen una vista muy general, por eso

son necesarias las vistas a diferentes escalas, pues se ubicarían los lugares y se analizarían las rutas con una mayor precisión.

También se necesita conocer los lugares aledaños a cada una de las escalas del viaje y si están cerca de localidades peligrosas se debería tomar un mayor número de precauciones, pues podrían sufrir ataques de piratería entre otras peligrosas situaciones que afectarían a la transportación. Realizar este proceso sin su debida digitalización, traería consigo un mayor empleo del tiempo en esta actividad, así como posibilidades de desacierto en muchos de los casos a partir de la complejidad de visualizar grandes cúmulos de información en formatos no digitales.

El Sistema de Gestión Integral de Aduanas no cuenta con un componente que gestione la información necesaria para facilitar la creación de rutas y la realización del proceso de ubicación geográfica, sin la necesidad de consumir servicios de georeferencias que se encuentran públicos en el mundo; imposibilitándose el acceso a los mismos debido a que su empleo es a través de una conexión a internet, lo que implica tener un enlace de altas prestaciones. Además estos servicios se encuentran en servidores externos al país, trayendo consigo que la información presente en la AGR se vea expuesta a problemas de confidencialidad, integridad y disponibilidad.

Debido al gran significado que ha tenido la representación y análisis de la información geográfica en el desarrollo informático y atendiendo a la situación problemática antes expuesta, se plantea el siguiente **problema de investigación** ¿Cómo localizar lugares y rutas a nivel global en el Sistema de Gestión Integral de Aduanas para facilitar una mayor percepción geográfica?

Se define como **objeto de estudio** Sistemas de Información Geográfica y como **campo de acción** un mecanismo de representación de coordenadas en un mapa digital.

Se trazó como **objetivo general** de la investigación: Desarrollar un componente en el Sistema de Gestión Integral de Aduanas que permita la localización de lugares y rutas específicas a nivel global, con el fin de facilitar la información necesaria para el seguimiento, evaluación y toma de decisiones por parte de los directivos de la aduana.

Para dar solución al objetivo general fue necesario definir los siguientes **objetivos específicos**:

1. Elaborar el marco teórico relativo a la investigación para obtener un estado del arte que permita tener un conocimiento amplio del tema.

2. Analizar el componente de posicionamiento global para el GINA.
3. Implementar la solución modelada de manera que cumpla con las necesidades del cliente.
4. Validar la solución mediante pruebas de caja negra utilizando el método partición equivalente.

En la presente investigación científica se emplearon los siguientes métodos teóricos y empíricos:

### **Métodos Teóricos:**

**Histórico – lógico:** Este método es empleado para investigar la existencia de alguna plataforma que implemente un SIG con características similares a la solución que se propone en el presente trabajo.

**Analítico – sintético:** Este método se utiliza en la investigación para analizar la bibliografía relacionada con los SIG, permitiendo la extracción de los elementos más importantes. El volumen de información estudiado se sintetizó en una nueva información concreta, resumiendo en ella las características generales y sus relaciones.

**Modelación:** Se emplea para mostrar los diferentes diagramas y componentes que se construyen como resultado del proceso de desarrollo de software.

Con la realización de este trabajo de diploma se obtendrá como **posible resultado** un componente que se integre al GINA y que permita localizar y visualizar la posición de lugares y rutas para todos los módulos del Sistema de Gestión Integral de Aduanas que lo requieran, facilitando el proceso de toma de decisiones a los trabajadores de la AGR.

El presente trabajo cuenta con tres capítulos. A continuación se hace una breve descripción de cada uno de ellos:

### **Capítulo 1: Fundamentación teórica**

En este capítulo se definirán los principales conceptos que permitirán un mejor entendimiento del tema, también se abordarán las características principales del objeto de estudio. Se hará referencia al estado del arte del tema tratado desde el ámbito nacional e internacional, enumerando y argumentando las soluciones parciales que existen y demostrando por qué no es posible el uso de las mismas. Además se realizará una descripción de las herramientas y tecnologías empleadas en el desarrollo de la solución así como el modelo de desarrollo empleado.

## **Capítulo 2: Análisis y diseño de la solución**

En este capítulo se definirán los requisitos funcionales y no funcionales que debe poseer el componente, los mismos serán validados a través de la técnica revisión formal de requerimientos. También se describirá la arquitectura base del componente, así como los diagramas de secuencia y de clases del diseño con estereotipos web. El diseño será validado utilizando la métrica tamaño operacional de clases (TOC).

## **Capítulo 3: Implementación y pruebas del sistema**

En el último capítulo de esta investigación, se definirán con exactitud cuáles son las precondiciones necesarias para el buen funcionamiento del componente. Después de haber concluido la implementación del mismo, se comprobará el estado funcional del componente implementado, a través de pruebas de caja negra, utilizando la técnica de particiones equivalentes. El objetivo principal de este capítulo es demostrar que la solución implementada es satisfactoria y que brinda una solución al objetivo general por el cual surgió la presente investigación.

## 1. CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### 1.1. Introducción

Los Sistemas de Información Geográfica son herramientas de apoyo a la toma de decisiones, tienen el objetivo de capturar, almacenar, analizar, transformar y presentar toda la información geográfica, con el fin de proporcionar respuestas a posibles interrogantes que podrían presentar los usuarios de los mismos. Existen varios tipos de SIG, en el desarrollo de este trabajo se estudiarán a mayor profundidad los SIG web, los cuales permiten visualizar datos, acceder a funcionalidades de análisis y consultar servidores SIG a través de una red de computadoras.

### 1.2. Conceptos asociados al dominio del problema

A continuación se enuncian varias definiciones que contribuirán a la comprensión del presente trabajo.

**Cartografía:** La cartografía es la ciencia que se encarga del estudio y la elaboración de los mapas geográficos, territoriales y de diferentes dimensiones lineales. Además se puede definir como el conjunto de mapas producidos por una institución, relativos a un determinado territorio. Es la técnica geográfica que estudia la secuencia de etapas y procesos ejecutados para la visualización de un espacio geográfico mediante la producción de mapas, cartas, planos o croquis [22].

**Tecnología de la Información y las Comunicaciones:** Se denominan Tecnologías de la Información y las Comunicaciones al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética [34].

**Información Geográfica:** Se denomina Información Geográfica (IG) a aquellos datos espaciales georeferenciados requeridos como parte de las operaciones científicas, administrativas o legales. Dichos datos espaciales suelen llevar una información alfanumérica asociada [23].

**Mapa:** El concepto de mapa proviene del término latino mappa. Se trata de una representación gráfica y métrica de una porción de territorio sobre una superficie bidimensional, que por lo general suele ser plana, aunque también puede ser esférica como en el caso de los globos terráqueos [29].



### 1.3. Sistemas de Información Geográfica

Los **Sistemas de Información Geográfica**, son herramientas de apoyo a la toma de decisiones, que facilitan el análisis de la información geográfica. Existen varias definiciones, dentro de las que se encuentran:

- Un SIG se define como un conjunto de métodos, herramientas y datos que están diseñados para actuar coordinada y lógicamente para capturar, almacenar, analizar, transformar y presentar toda la información geográfica y de sus atributos con el fin de satisfacer múltiples propósitos. Los SIG son una nueva tecnología que permite gestionar y analizar la información espacial y que surgió como resultado de la necesidad de disponer rápidamente de información para resolver problemas y contestar a preguntas de modo inmediato [2].
- Una integración organizada de hardware, software y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y gestión. También puede definirse como un modelo de una parte de la realidad referido a un sistema de coordenadas terrestre y construido para satisfacer necesidades concretas de información [3].
- Un sistema de Información concebido para trabajar con datos referenciados en el espacio geográfico. Tiene capacidades específicas para manejar datos espaciales y analizarlos mediante un conjunto de operaciones [30].

Después de analizar los conceptos anteriores se puede concluir que para el desarrollo de este trabajo un SIG es una herramienta de apoyo a la toma de decisiones que está diseñado para capturar, almacenar, analizar, transformar y presentar toda la información geográfica, con el fin de dar respuestas a posibles preguntas formuladas por los usuarios que utilicen este componente como son ¿Dónde se encuentra este lugar?, ¿Qué ocurriría si se toma esta ruta?, ¿Qué problemas traería esta escala del viaje?

La siguiente ilustración muestra los componentes de un SIG:



Ilustración 1. Sistema de Información Geográfica [35].

### Equipos (Hardware)

Es donde opera el SIG. Estos programas se pueden ejecutar en un amplio rango de equipos, desde servidores hasta computadores personales usados en red o trabajando en modo "desconectado" [2].

### Programas (Software)

Los programas de SIG proveen las funciones y las herramientas necesarias para almacenar, analizar y desplegar la información geográfica. Los principales componentes de los programas son [35]:

- Herramientas para la entrada y manipulación de la información geográfica.
- Un sistema de administración de base de datos.
- Herramientas que permitan búsquedas geográficas, análisis y visualización.
- Interfaz gráfica para que el usuario acceda fácilmente a las herramientas.

### Datos:

Probablemente la parte más importante de un sistema de información geográfico son sus datos, estos pueden ser geográficos y tabulares, son adquiridos por la persona que implementa el sistema de información, así como por terceros que ya los tienen disponibles. El sistema de información geográfico integra los datos espaciales con otros recursos de datos y puede incluso utilizar los manejadores de base

de datos más comunes para manejar la información geográfica [35].

### **Recurso humano:**

La tecnología de los SIG está limitada si no se cuenta con el personal que opera, desarrolla y administra el sistema y que establece planes para aplicarlo en problemas del mundo real [2].

### **Procedimientos:**

Un SIG operará acorde con un plan bien diseñado y con unas reglas claras del negocio [2].

### **Tipos de SIG:**

Los SIG pueden ser divididos en 4 grupos de programas informáticos según su dominio en el mercado internacional [4]:

- **SIG de escritorio:** Utilizados para crear, editar, administrar, analizar y visualizar los datos geográficos, son clasificados en tres categorías según su funcionalidad:

Visor SIG: Son software que permiten desplegar información geográfica a través de una interfaz que funciona como visor, en la cual se pueden agregar varias capas de información.

Editor SIG: Es aquel SIG orientado principalmente al tratamiento previo de la información geográfica para su posterior análisis. Antes de introducir datos a un SIG es necesario prepararlos para su uso en este tipo de sistemas.

SIG de análisis: Disponen de funcionalidades de análisis espacial y modelización cartográfica de procesos.

- **Componentes de desarrollo SIG:** Paquete de herramientas que permiten la programación de funciones de visualización y consulta de información geográfica.
- **SIG web:** Permiten la visualización de datos, así como el acceso a funcionalidades de análisis y consulta de servidores SIG a través de una red de computadoras.
- **SIG móviles:** Usados para la recogida de datos a través de dispositivos móviles. Con la adopción generalizada por parte de estos, de dispositivos de localización *Global Positioning System* (GPS) integrados, el software SIG permite utilizarlos para la captura y manejo de datos en campo.

Existen dos formas de almacenar la información que maneja un SIG:

**Modelo raster:** Se centra, más que en la precisión de la localización, en las propiedades que posee el espacio de trabajo. Para lograr estas precisiones divide este espacio en celdas, donde cada una de ellas representa una información específica con un único valor. Si las resoluciones de las celdas son mayores, menores serán las precisiones de representación del espacio geográfico. Este modelo es utilizado en estudios medioambientales ya que no se requiere una elevada precisión [26].

**Modelo vectorial:** Este modelo se centra en la precisión de la localización o en los detalles de la información geográfica que se desee representar. Permitirá graficar un objeto a través de puntos, líneas y polígonos. Los cálculos de áreas, perímetros y longitudes son más precisos que en el Modelo raster [26]. En la siguiente ilustración se puede observar de una forma más precisa las diferencias entre el Modelo raster y el Modelo vectorial.



**Ilustración 2. Diferencias entre Modelo raster y Modelo vectorial [45].**

El mundo real es representado por el Modelo raster utilizando dos capas: el tipo de suelo y las alturas de terreno; como se puede observar este modelo se centra en las propiedades que posee el espacio de trabajo.

El mundo real es representado por el Modelo vectorial utilizando tres capas: polígonos, líneas y puntos; como se puede observar este modelo se centra en la precisión de la localización.

## 1.4. Análisis de otras soluciones existentes

En el mundo existe una gran cantidad de aplicaciones que tienen el objetivo de facilitar la ubicación geográfica de forma digital. A continuación se realiza una descripción de algunos SIG, los cuales son utilizados por usuarios con diferentes niveles de preparación. Se analizarán las principales características de cada uno de ellos, así como las limitaciones que tiene su uso para la AGR.

### 1.4.1. Ámbito internacional

**Google Maps** es el nombre de un servicio gratuito de Google. Es un servidor de aplicaciones de mapas en la web. Desde el 6 de octubre del 2005, Google Maps es parte del Google Local. Algunas de sus principales características son [27]:

Los mapas de la API (Interfaz de Programación de Aplicaciones) de Google se basan en tecnología JavaScript lo que constituye una gran ventaja ya que de esta manera no se necesita instalar software alguno y es compatible con prácticamente todos los navegadores, Firefox, Mozilla, Opera, Internet Explorer, Netscape y otros.

Utiliza la librería Mapstraction la cual permite desarrollar mapas en JavaScript para múltiples proveedores con un mismo código. Se puede agregar texto HTML (Lenguaje de Marcado de Hipertexto) y XML (Lenguaje de Marcas Extensible) en una pequeña ventana de información.

Este servicio de Google es muy utilizado en el mundo debido a las variadas y útiles funcionalidades que brinda como son [27]:

- Ofrece imágenes de mapas desplazables, así como fotos satelitales del mundo e incluso la ruta entre diferentes ubicaciones.
- También ofrece la capacidad de hacer acercamientos o alejamientos para mostrar el mapa.
- El usuario puede controlar el mapa con el mouse o las teclas de dirección para moverse a la ubicación que se desee.
- Para permitir un movimiento más rápido, las teclas "+" y "-" pueden ser usadas para controlar el nivel de *zoom*.
- Los usuarios pueden ingresar una dirección, una intersección o un área en general para buscar en el mapa.

*Google Maps* cuenta además con varias herramientas de apoyo como es *Google Street View* la cual

muestra imágenes a pie de calle, *Google Ride Finder* se basa en la localización permanente de taxis y limusinas mediante el uso de GPS, *Google Moon* hizo público el uso de las imágenes de la NASA (Administración Nacional de la Aeronáutica y del Espacio) acerca de la geografía lunar y *Google Mars* el cual ofrece imágenes de satélite de Marte así como imágenes infrarrojas y del relieve de este planeta.

**Yahoo Maps** es un SIG muy similar a Google Maps, tanto en interfaz como en posibilidades de uso algunas de sus principales características son:

Puede agregar texto HTML en una pequeña ventana de información. Las fotos aéreas proporcionan una menor nitidez que los servicios de Google Maps y al igual que este utiliza la librería Mapstraction.

Las herramientas de visualización de mapas y herramientas para desarrolladores incluyen una interfaz de programación de aplicaciones (API) que permite integrar fácilmente mapas interactivos en la web y aplicaciones de escritorio, pudiendo elegir las tecnologías a utilizar entre las que se encuentran Flash o Ajax (JavaScript asíncrono y XML) [36].

También ofrece un API de Geocodificación denominado *Yahoo Place Finder* que permite transformar las direcciones físicas en localizaciones codificadas (latitud y longitud).

Este SIG cuenta con varias funcionalidades como son [36]:

- Capacidad de hacer acercamientos o alejamientos para mostrar el mapa.
- El usuario puede controlar el mapa con el mouse o las teclas de dirección para moverse a la ubicación que se desee.
- Para permitir un movimiento más rápido, las teclas "+" y "-" pueden ser usadas para controlar el nivel de *zoom*.
- Además de las clásicas opciones como alternar entre vista satélite o de mapa, realización de rutas o marcado de puntos, se incluye un interesante buscador específico de negocios y una utilísima vista del tráfico en tiempo real.

**Bing Maps** es un servicio de mapas web, que forma parte del motor de búsqueda Bing de Microsoft y que funciona con *Bing Maps for Enterprise* [28]:

Dentro de las funcionalidades de este SIG se encuentran [28]:

- Ofrece la capacidad de hacer acercamientos o alejamientos para mostrar el mapa.
- El usuario puede controlar el mapa con el mouse o las teclas de dirección para moverse a la ubicación que se desee.

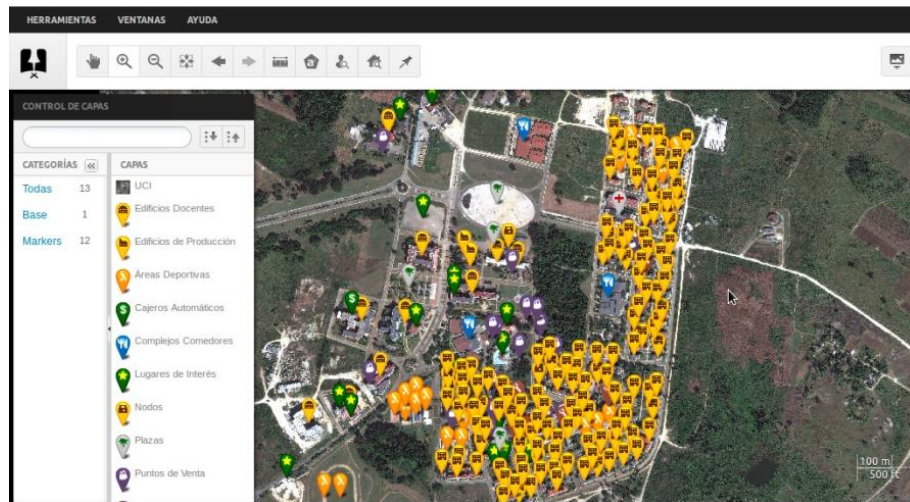
- Para permitir un movimiento más rápido, las teclas "+" y "-" pueden ser usadas para controlar el nivel de *zoom* el cual logra aproximarse y mantener una buena calidad de la imagen.
- Combina los mapas dibujados con las imágenes planas del satélite.
- Ofrece como extra una visualización distinta, denominada "*Vista de pájaro*", la cual consiste en fotos aéreas tomadas de 4 ángulos diferentes. Estas imágenes (*Pictometry*) son más detalladas que las vistas aéreas de edificios anteriores. Las señales, anuncios, peatones y otros objetos son claramente visibles en muchas *Vistas de pájaro*. Sustituye la perspectiva cenital por imágenes reales con una leve inclinación que ayudan a captar mejor la perspectiva. Además, muestra partes de las ciudades en tres dimensiones, con algunos de sus edificios más característicos.

Uno de los requisitos del Sistema de Gestión Integral de Aduanas es precisamente que no necesite consumir datos de servidores externos al país, por lo que el mismo no puede hacer uso de estos servicios de georeferencias Google Maps, Yahoo Maps y Bing Maps debido a que el acceso a los mismos es a través de una conexión a internet, que implica tener un enlace de altas prestaciones y que además se presentan en servidores externos al país, trayendo consigo que la información presente en la AGR se vea expuesta a problemas de confidencialidad, integridad y disponibilidad.

#### 1.4.2. **Ámbito nacional**

Cuba no está ajena a los servicios que brindan facilidades para la ubicación geográfica, por lo que también cuenta con Sistemas de Información Geográfica que permiten la localización exacta de lugares.

Uno de estos servicios es el SIG de la Universidad de Ciencias Informáticas conocido como **SIGUCI** el cual fue desarrollado en la UCI con el objetivo de facilitarle a la comunidad universitaria una mejor ubicación dentro de la misma [38]. En la siguiente ilustración se presenta la portada principal de SIGUCI.



**Ilustración 3. Portada de SIGUCI.**

Esta aplicación está desarrollada con la plataforma GeneSig que permite el desarrollo de sistemas de este tipo. Algunas de sus principales funcionalidades son [38]:

- Medir distancias, ubicar lugares, acercar, alejar o mover el mapa, para estas opciones se puede utilizar el *scroll* o a través de mouse como más cómodo resulte al usuario.

SIGUCI no brinda la posibilidad de enmarcar todas sus funcionalidades en el mundo, sino que solo se centra en el área de la UCI, por lo que no cumple objetivo su utilización para la AGR ya que no cuenta con los requerimientos necesarios.

Otro SIG desarrollado en la UCI es el **SIG\_VERANO** el cual tiene el objetivo de permitir la representación geoespacial de la información, asociada a la disponibilidad de los distintos locales que son utilizados para el recreo y disfrute de la familia cubana en cada una de las provincias durante la etapa veraniega [37].

Una de las principales deficiencias que tiene este sistema es que no permite ubicar puntos geográficos dada una latitud y una longitud, además solo enmarca sus límites en Cuba; este no cumple con las funcionalidades que exige el usuario por lo tanto no es posible su uso en la AGR.

También existe el Sistema para la representación geográfica de objetivos hidrográficos de precipitaciones, **SIG-Hidro**, una aplicación desarrollada sobre la plataforma GeneSig con el objetivo de permitir la representación geoespacial de la información asociada a los objetivos hidrográficos de precipitaciones [43]. Este sistema no cumple con las características necesarias para el cliente, entre otras razones porque utiliza el Modelo raster para la representación de la información, este es el idóneo para representar



objetivos hidrográficos, pero no es recomendable en la representación de puntos geográficos.

Después de haber realizado un análisis crítico de las herramientas existentes en el mundo y en Cuba que se relacionan con el tema, se afirma que no existe ninguna que pueda ser utilizada por las exigencias de la AGR para la ubicación geográfica de lugares de interés. Las soluciones nacionales no permiten realizar ubicaciones en un mapa global y las soluciones internacionales se encuentran públicas en internet y es necesario no consumir servicios de este. Por las razones argumentadas anteriormente se puede concluir que es imprescindible la creación de un componente que permita la localización de lugares y rutas específicas a nivel global, con el fin de facilitar la información necesaria, para el seguimiento, evaluación y toma de decisiones, por parte de los directivos de la aduana.

### **1.5. Selección de las tecnologías a utilizar para la implementación**

En este epígrafe se realizará un estudio detallado de todas las posibles herramientas que se pueden utilizar en el desarrollo del componente, siempre y cuando sean específicas para el trabajo con mapas digitales, llegando a la conclusión de cuáles son las más idóneas. Debido a la importancia que tiene para el país la migración hacia Plataformas y Sistemas de Código Abierto (Software Libre) se seleccionarán tecnologías que cumplan con estas condiciones. Existen otras tecnologías que serán utilizadas debido a que su uso está definido en el proyecto GINA.

#### **Servidores de mapas de código abierto**

Dentro de los servidores de mapas de código abierto más reconocidos en el mundo están MapGuide Open Source, Deegree, Geoserver y MapServer.

MapGuide Open Source es una plataforma basada en la web que permite a los usuarios desarrollar y desplegar aplicaciones web y servicios de cartografía geoespacial web. MapGuide incluye una base de datos XML (Lenguaje de Marcas Extensible) para la gestión de contenidos, y soporta los formatos más populares de archivos geoespaciales y bases de datos. MapGuide puede implementarse en Linux o Windows. Ofrece soporte a una gran cantidad de lenguajes tales como: PHP, .NET, Java, JavaScript [5].

Deegree es un software de código abierto para las infraestructuras de datos espaciales y de la web geoespacial. Incluye componentes para la administración de datos geoespaciales, incluyendo el acceso a

datos, la visualización, el descubrimiento y la seguridad. Los estándares abiertos son el corazón de deegree. El software se basa en los estándares del Open Geospatial Consortium (OGC) y el Comité Técnico ISO 211 [6].

GeoServer es un servidor implementado con Software Libre que permite publicar y editar datos espaciales, aunque está desarrollado en Java puede ser bastante rápido cuando se configura correctamente.

Principales ventajas:

- Facilidad de manejo, la configuración se realiza a través de una aplicación web que evita la edición de complejos ficheros de configuración.
- Proporciona una web para administración, lo que facilita la configuración.

MapServer es un entorno de desarrollo en código abierto para la creación de aplicaciones SIG con el fin de visualizar, consultar y analizar información geográfica. Fue desarrollado por la Universidad de Minnesota a mediados de los años 1990.

Para el desarrollo del trabajo se seleccionó este ya que posee las características necesarias para desarrollar la aplicación y brinda varias facilidades para el desarrollo de la misma. Dentro de las principales ventajas están:

- Soporte multiplataforma.
- Velocidad de Acceso a datos.
- Generación automática de los elementos de un mapa (barra de escala, mapa de referencia y leyenda).
- Dispone de un poderoso sistema cartográfico, proporcionando datos bajo vectores dinámicos con alta calidad.
- Soporta formatos vectoriales: ESRI, shapefiles y PostGIS.
- Formatos raster soportados: JPG, PNG y GIF.
- MapServer es generalmente mejor con Web Map Service (WMS), mientras que GeoServer trabaja mejor con Web Features Service (WFS).

Además brinda soporte a los lenguajes de scripting y ambientes de desarrollo más populares como son: PHP, Python, Perl, Ruby, Java y .NET [7].

## **Servidores web**

Apache es el servidor web hecho por excelencia: su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Este fue el seleccionado para el desarrollo del componente, ya que además de estar definido su uso por el proyecto [31] presenta un grupo de ventajas que se mencionan a continuación [8]:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código abierto.
- Apache trabaja con gran cantidad de lenguajes de programación, como son Perl, PHP y otros lenguajes de script.
- Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.

## **Lenguajes de programación de código abierto**

PHP es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. El uso de este está definido en el proyecto productivo GINA [31] debido a la diversidad de ventajas que presenta como son:

- Es un lenguaje multiplataforma [9].
- Es un lenguaje de código abierto [24].
- PHP es completamente expandible. Está compuesto por un sistema principal, un conjunto de módulos y una variedad de extensiones de código [24].
- PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz.
- Facilita la conexión con un gran número de bases de datos como PostgreSQL, MySQL, Oracle, Informix, entre otros [9].

## **Tecnología SIG del lado del cliente**

OpenLayers es una librería JavaScript de código abierto que tiene el objetivo de mostrar mapas interactivos en los navegadores web [10].

Ext JS es una librería JavaScript que ofrece un extraordinario conjunto de componentes para incluir dentro de una aplicación web como rejillas, árboles de datos, menús y paneles [10].

Para el desarrollo del componente se seleccionó GeoExt con el objetivo de mantener el formato de las interfaces visuales definidas por el proyecto GINA. Algunas características son:

- Permite crear, de forma fácil, aplicaciones para la visualización, edición y diseño de datos geoespaciales.
- Combina los controles geoespaciales de OpenLayers con los componentes de interfaz de usuario de Ext JS en un *framework* que permite construir aplicaciones SIG de estilo similar a las de escritorio, pero en un navegador [10].

### **Sistema gestor de base de datos**

PostgreSQL es un potente gestor de bases de datos de código abierto que tiene las siguientes características [39]:

- Perteneciente al ámbito del software libre.
- Es robusto, cumple con los estándares SQL y es escalable.
- Cuenta con versiones para una amplia gama de sistemas operativos, entre ellos: Linux, Windows, Mac OS X, Solaris, BSD y Tru64.

PostGIS es un módulo utilizado en el desarrollo de este trabajo ya que añade soporte de objetos geográficos a la base de datos objeto-relacional PostgreSQL, convirtiéndola en una base de datos espacial y pudiendo ser utilizada para almacenar y consultar datos espaciales (puntos, líneas y polígonos).

Algunas de sus principales características son [11]:

- Base de datos espaciales de alto rendimiento y robusta construida en PostgreSQL.
- Alta fiabilidad.
- Proporciona representaciones espaciales de los tipos geométricos (puntos, líneas y polígonos).
- Compatible con varios servidores de código abierto: GeoServer, MapServer, Mapnik, Deegree.

### **1.6. Entorno de trabajo a utilizar**

En el desarrollo de proyectos informáticos, el entorno de trabajo o marco de trabajo es una estructura de soporte que se define para que el proyecto pueda ser organizado y desarrollado sobre el mismo, son diseñados con la intención de facilitar y agilizar el desarrollo de software [12].

**Symfony** es un completo *framework* diseñado para optimizar el desarrollo de las aplicaciones web, gracias a sus características este es el utilizado en el proyecto GINA [31] por lo que también será utilizado en el desarrollo del presente trabajo. Algunas de sus características son [12]:

- Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web.
- Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja.
- Automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.
- Está desarrollado completamente con PHP 5.
- Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel.
- Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft.
- Se puede ejecutar tanto en plataformas \*nix (Unix, Linux, etc.) como en plataformas Windows.

## 1.7. Herramientas de desarrollo

Las herramientas de desarrollo de software permiten a los desarrolladores la creación de aplicaciones para sistemas concretos. Las más comunes incluyen técnicas de soporte para la detección de errores de programación, la generación automática de código, entre otras características que facilitan y agilizan el desarrollo de soluciones de software. Frecuentemente incluyen también códigos de ejemplo, notas técnicas y documentación de soporte [13]. Las herramientas utilizadas para el desarrollo del componente son:

**Netbeans** es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero se utiliza para desarrollar en varios lenguajes de programación [14].

Características:

- Netbeans es un Entorno de Desarrollo Integrado (IDE) de código abierto y gratuito.
- Es multiplataforma, haciendo posible su uso en diversos sistemas operativos.
- Brinda las herramientas necesarias para el desarrollo en lenguajes como Java, XML, HTML, PHP y

JavaScript.

- Presenta aplicaciones para la detección y tratamiento de errores.
- Tiene una amplia comunidad de desarrollo y soporte.

**Visual Paradigm** es una herramienta CASE (Ingeniería de Software Asistida por Computadora) profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El lenguaje unificado de modelado (UML) ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación [15].

Esta herramienta es utilizada en el desarrollo del presente trabajo pues permite dibujar todos los tipos de diagramas de clases, generar código desde diagramas y generar documentación, también proporciona abundantes tutoriales de UML y demostraciones interactivas de UML.

Características:

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio.
- Ingeniería inversa.
- Modelo a código, diagrama a código.
- Diagramas de flujo de datos.
- Soporte a los sistemas de Mapeo Objeto-Relacional (ORM).
- Generación de bases de datos.
- Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Distribución automática de diagramas.

## **1.8. Modelo de desarrollo de software**

Para el buen funcionamiento del Centro de Informatización de la Gestión de Entidades (CEIGE) y de los proyectos que pertenecen al mismo se ha establecido el siguiente modelo de desarrollo en el cual se especifican las actividades de cada una de las fases del ciclo de vida de los proyectos del centro teniendo en cuenta los procesos de CMMI (Integración de Modelos de Madurez de Capacidades) del nivel 2 para la UCI. También se detallan los artefactos a generar en cada momento independientemente de las

herramientas o métodos que se utilicen para ello. Debido a la buena organización que brinda este modelo para el desarrollo de un proyecto y por estar establecida su utilización en el proyecto GINA, será utilizado en el desarrollo del presente trabajo. A continuación se muestra el ciclo de vida de los proyectos del CEIGE el cual tiene en cuenta las actividades de cada una de las fases [16].



**Ilustración 4.** Ciclo de vida de proyectos del CEIGE [12].

### **Descripción de las fases del ciclo de vida de los proyectos**

Inicio o Estudio preliminar: Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto a un alto nivel, la evaluación de la factibilidad del proyecto y el registro de este. En esta fase se realiza un estudio inicial de la organización cliente, que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo, para de esta manera decidir si se ejecuta o no el proyecto.

Los objetivos de la fase son [12]:

- Asegurar la factibilidad del proyecto.
- Establecer un plan para la ejecución del proyecto.

Hitos:

- Plan de desarrollo de software.
- Acta de inicio del proyecto firmada.

Desarrollo: En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se refinan los requisitos, se elaboran la arquitectura y el diseño, se implementa y por último se libera el producto.

El objetivo de esta fase es [12]:

- Obtener un sistema que satisfaga las necesidades de los clientes y usuarios finales.

Hito:

- Producto liberado por entidad certificadora de calidad.

En esta fase se ejecutan las disciplinas Modelado de negocio, Requisitos, Análisis y diseño, Implementación, Pruebas internas y Pruebas de liberación.

## **1.9. Conclusiones parciales**

En este capítulo se realizó un estudio amplio de los Sistemas de Información Geográfica con el objetivo de contribuir a la comprensión del presente trabajo.

También se analizaron las soluciones existentes en el ámbito nacional e internacional, arribándose a la conclusión de que ninguna cumple con todos los requerimientos necesarios para ser utilizadas por la AGR, por lo que se hace necesaria la implementación de un componente que cumpla con los objetivos trazados.

Se analizaron las tecnologías más adecuadas a utilizar en el desarrollo del componente, seleccionando como servidor de mapas MapServer y como lenguaje de programación por parte del cliente la librería GeoExt, ambas tecnologías son específicas en el trabajo con datos geográficos.



## 2. CAPÍTULO 2: ANÁLISIS Y DISEÑO DE LA SOLUCIÓN

### 2.1. Introducción

Para cumplir con el objetivo general de la presente investigación es necesario definir los requisitos funcionales que se implementarán y realizar una validación de los mismos; así como los requisitos no funcionales, los cuales contribuirán al buen funcionamiento del componente. La arquitectura de software establece la estructura, funcionamiento e interacción entre las partes del software; la misma está compuesta por los artefactos más significativos, permitiendo establecer un esquema de cómo va a quedar constituido el mismo. Todos estos elementos serán explicados detalladamente en los epígrafes del presente capítulo.

### 2.2. Modelo de dominio o conceptual

Puede utilizarse para capturar y expresar el entendimiento obtenido en un área bajo análisis como paso previo al diseño de un sistema. El modelo de dominio es utilizado por el analista como un medio para comprender el sector de negocios al cual el sistema estará dirigido. Este proporciona una perspectiva conceptual a través de [21]:

- Objetos o clases conceptuales.
- Asociaciones entre las clases conceptuales.
- Atributos de las clases conceptuales.

A continuación se muestra el modelo conceptual del sistema, para mayor información ver anexo 2 correspondiente al tema.

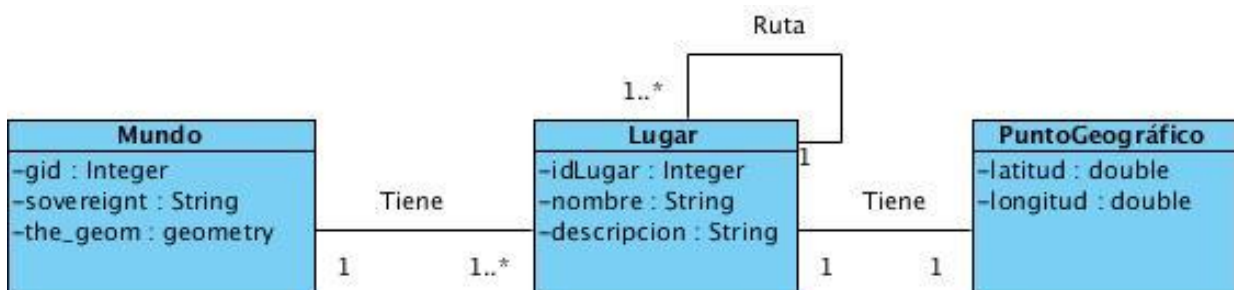


Ilustración 5. Modelo de Dominio.

### 2.3. Requisitos funcionales

Existen varias técnicas para capturar los requisitos, una de las utilizadas fue la introspección que consiste en ponerse en el lugar del cliente e imaginar cómo quisiera que quedara el sistema, luego sobre la base de estas suposiciones se le recomendó un grupo de funcionalidades que pudieran resolver su problema. También se realizó un estudio amplio de algunas herramientas existentes que fueron brindando una idea de cómo lograr las especificaciones del cliente. Llegando como conclusión a los siguientes requisitos funcionales y no funcionales:

Nº	Funcionalidad	Descripción	Complejidad
R1	Cargar el mapa del mundo.	El sistema debe cargar un mapa del mundo para permitir realizar las ubicaciones sobre el mismo.	Alto
R2	Acercar el mapa.	El sistema debe permitir acercar el mapa lo que permite disminuir la escala de visualización.	Media
R3	Alejar el mapa.	El sistema debe permitir alejar el mapa lo que permite aumentar la escala de visualización.	Media
R4	Mover el mapa.	Con este requisito se quiere que el usuario pueda mover el mapa utilizando el mouse para obtener la vista que se desee.	Media
R5	Mostrar mapa de referencia.	Permitir que el usuario pueda seleccionar los lugares que desea observar a través de un mapa de referencia, según la escala de visualización que haya seleccionado.	Baja
R6	Ver todo el mapa.	El sistema debe permitir visualizar el	Media

		mapa según la escala inicial de la aplicación.	
R7	Realizar acción anterior.	El sistema debe permitir visualizar el mapa anterior al que se visualiza en la aplicación.	Media
R8	Realizar acción siguiente.	El sistema debe permitir que una vez que haya seleccionado la acción anterior vuelva al estado en que se encontraba.	Media
R9	Localizar lugares.	El sistema debe permitir la localización de puntos geográficos, dada las coordenadas del mismo.	Alta
R10	Mostrar escala del mapa.	Mostrar la escala de visualización, la cual varía en dependencia del nivel de zoom.	Media
R11	Mostrar rutas ascendentes.	El sistema debe ir señalando los puntos que pertenecen a una ruta con un número de forma ascendente.	Media
R12	Mostrar leyenda.	El sistema debe mostrar una leyenda de todos los símbolos que se utilizan en el mapa.	Alta
R13	Mostrar información de lugares.	Al pasar el mouse por encima de un lugar señalado en el mapa el sistema debe mostrar un cuadro de texto con la información referente al lugar.	Alta
R14	Mostrar información de los botones utilizados.	Al pasar el mouse por encima de los iconos, muestra el objetivo del mismo.	Media

**Tabla 1. Requisitos funcionales.**

## 2.4. Requisitos no funcionales

### 1. Usabilidad:

El componente implementado podrá ser utilizado por personas con conocimientos mínimos de la informática y las funcionalidades del mismo estarán orientadas a iconos que muestren un reconocimiento más fácil por parte del usuario.

### 2. Eficiencia:

El tiempo de respuesta estará dado por la cantidad de información a procesar, al igual que la velocidad de procesamiento de los datos.

### 3. Interfaz de usuario:

El sistema debe tener un diseño gráfico sencillo, con una interfaz intuitiva y amigable para el usuario.

### 4. Restricciones de diseño:

El producto de software debe estar diseñado sobre una arquitectura:  
Modelo – Vista – Controlador.

### 5. Seguridad :

La información manejada por el sistema será protegida de accesos no autorizados, garantizando la confidencialidad de la información.

### 6. Requisitos de software:

El componente desarrollado debe cumplir con las siguientes características:

#### PC cliente:

- Un navegador web Mozilla Firefox 15.x o superior, Internet Explorer 8 o superior.
- Sistema Operativo GNU/Linux o Windows.

#### Servidores:

- Sistema Operativo GNU/Linux.
- Servidor web Apache 2.0 o superior.
- PostgreSQL 9.0 como sistema gestor de base de datos, utilizando el plugin PostGIS versión 2.0.3 para el trabajo con los datos geoespaciales.
- MapServer con extensión PHP 5 y MapScript.
- OpenLayers versión 2.10 o superior.

- GeoExt versión 0.7 y Ext JS 3.0
- Symfony versión 1.x.

## 7. Requerimientos de hardware:

### PC cliente:

- Las PC tienen que tener como mínimo 1 Giga Byte de memoria RAM.
- Como mínimo un procesador core 2 duo a 1.7 GHz de frecuencia.

### Servidor:

- Memoria RAM de 2 GB.
- Espacio en disco disponible de 10 GB.

## 8. Portabilidad:

El sistema será multiplataforma, funciona en los sistemas operativos GNU/Linux y Windows.

## 2.5. Validación de los requisitos de software

La validación de requisitos tiene como misión demostrar que la definición de estos, realmente representa lo que el usuario necesita. Además tiene el propósito de conocer errores y prevenirlos a tiempo antes de comprometer recursos en las fases futuras del desarrollo de software. Los parámetros a validar en los requisitos son [44]:

- Consistencia: No debe haber contradicciones entre unos requisitos y otros.
- Realismo: Se pueden implementar con la tecnología actual.
- Verificabilidad: Para reducir la posibilidad de discusiones entre el cliente y el contratista los requerimientos del sistema deben redactarse de tal forma que sean verificables.
- Completitud: El documento de requerimientos debe incluir requerimientos que definan todas las funciones y restricciones propuestas por el usuario del sistema.
- Validez: Un usuario puede pensar que necesita un sistema para llevar a cabo ciertas funciones. Sin embargo el razonamiento y el análisis pueden identificar que se requieren funciones adicionales o diferentes.

Existen un conjunto de técnicas que se utilizan para llevar a cabo la validación de los requisitos, una de estas son las revisiones de requerimientos; las mismas son un proceso manual que involucra a personas tanto de la organización del cliente como de la del contratista. En una revisión formal de requerimientos el

equipo de desarrollo debe conducir al cliente a través de los requerimientos del sistema, explicándole las implicaciones de cada uno de ellos [32].

La técnica revisión formal de requerimientos fue utilizada en la presente investigación con el objetivo de leer y corregir la documentación, incluidos los documentos de apoyo como son especificación y descripción de requisitos para validar la correcta interpretación de la información transmitida.

## 2.6. Arquitectura de software

La Arquitectura de Software es la organización fundamental de un sistema, encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución [13]. Es considerada el nivel más alto en el diseño de la arquitectura de un sistema puesto que establece la estructura, funcionamiento e interacción entre las partes del software. La misma está compuesta por los artefactos más significativos, permitiendo establecer un esquema de cómo quedaría constituido el software. Sobre la arquitectura inicial se van agregando nuevos artefactos propiciado así una arquitectura más robusta, describiéndose en la misma los principales aspectos tomados en cuenta para la construcción del software de forma progresiva.

A continuación se muestra una figura que ilustra los componentes fundamentales para la implementación del SIG y constituye la arquitectura base del mismo.

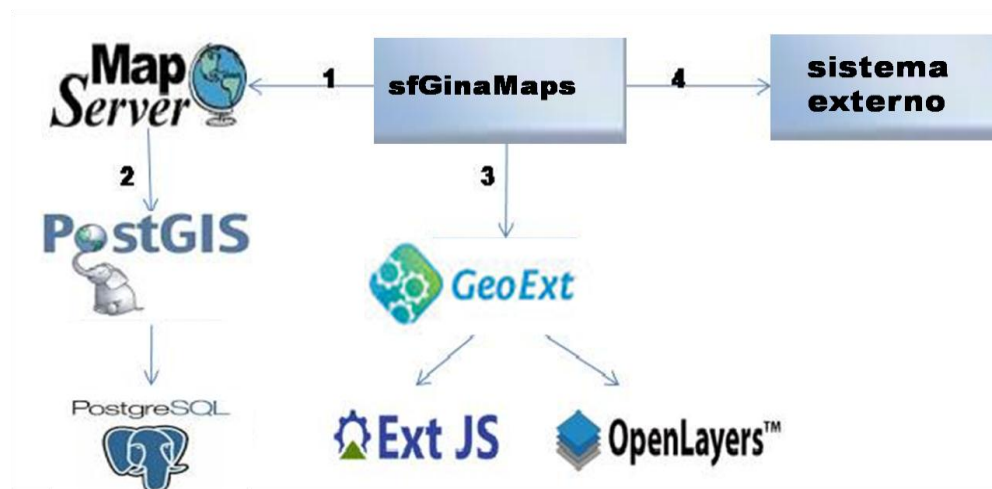


Ilustración 6. Arquitectura del componente a desarrollar.

1-El componente a desarrollar utiliza el servidor de mapas MapServer para visualizar, consultar y analizar la información geográfica. El archivo mapfile (en el caso del componente sería el archivo mapas.map) es el archivo de configuración que MapServer usa para convertir datos geoespaciales en imágenes o datos vectoriales; su propósito principal es definir las capas que puede generar, cómo obtener los datos necesarios y cómo dibujarlas, estas imágenes generadas pueden ser almacenadas en un directorio local o presentadas en un navegador web como en este caso.

2-El archivo mapas.map contiene una capa la cual realiza la conexión con la base de datos del componente, en este caso utilizando el sistema gestor de base de datos postgresQL y realizando una conexión de tipo PostGIS que es el encargado de permitir el trabajo con datos geográficos, obteniendo los mismos a través de consultas SQL.

3-El componente a desarrollar utiliza GeoExt como tecnología SIG del lado del cliente ya que esta mantiene el formato de las interfaces visuales definidas por el proyecto GINA, pues utiliza las librerías JavaScript de código abierto Ext JS y OpenLayers.

4-El componente a desarrollar podrá ser integrado con cualquier sistema externo que necesite ubicar lugares geográficamente. Para poder utilizar el mismo el sistema tiene que estar desarrollado sobre el *framework* Symfony y garantizar las precondiciones del epígrafe 3.2.

La arquitectura de software es un conjunto de patrones que proporcionan un marco de referencia necesario para guiar la construcción de un software, permitiendo a los programadores, analistas y todo el conjunto de desarrolladores del software compartir una misma línea de trabajo y cubrir todos los objetivos y restricciones de la aplicación [33].

## **2.7. Patrones arquitectónicos**

Un patrón es un modelo que se puede seguir para realizar algo, estos surgen de la experiencia de seres humanos de tratar de lograr ciertos objetivos. Los mismos capturan la experiencia existente y probada para promover buenas prácticas.

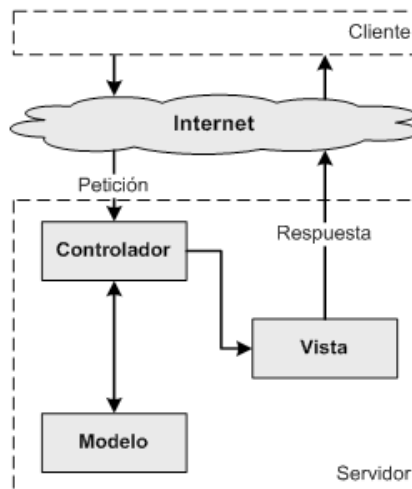
Las principales características de los patrones son [17]:

- Tiene un nombre.
- Tiene un contexto o problema a resolver.
- Tiene una solución.

- Tiene unas consecuencias al utilizarlo.

El *framework* Symfony brinda varias facilidades a los desarrolladores, con el objetivo de facilitar el trabajo y lograr una mejor organización, se basa en el patrón arquitectónico Modelo Vista Controlador, el cual está formado por tres niveles como se muestra en la ilustración 7:

- El Modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La Vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El Controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.



**Ilustración 7. El patrón Modelo Vista Controlador [18].**

Esta arquitectura separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como en un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes del tipo de gestor de bases de datos utilizado por la aplicación [18].

Este patrón cumple con las principales características de los mismos:



**Nombre:**

Modelo Vista Controlador.

**Contexto o Problema a resolver:**

Es muy frecuente que se solicite cambio a interfaz. Los cambios a interfaz deberían ser fáciles y efectuados en tiempo de ejecución. El cambio de interfaz no debería de tener consecuencias para el núcleo del código de la aplicación.

**Solución:**

El sistema se divide en tres partes: procesamiento, entradas y salidas.

Modelo: encapsula los datos y la funcionalidad de la aplicación.

Vista – despliega la información contenida en el modelo (pueden existir varias vistas).

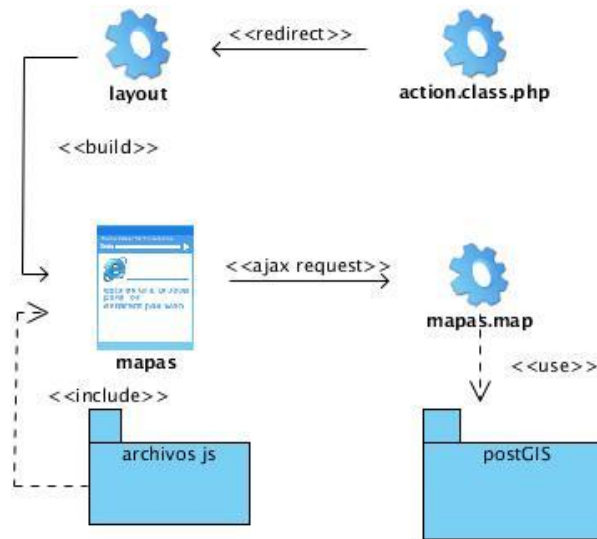
Controlador – está asociado a cada vista, recibe entradas que traduce en invocaciones de métodos del Modelo o de la Vista. El usuario interactúa con el sistema solamente vía controladores.

## 2.8. Diagrama de clases del diseño con estereotipos web

Una clase de diseño y sus objetos participan en varias realizaciones de requisitos funcionales. También puede suceder que algunas operaciones, atributos y asociaciones sobre una clase específica sean solo relevantes para una sola realización de un requisito funcional. Para manejar todo esto se utilizan los diagramas de clases conectados a una realización de un requisito funcional, mostrando así sus clases participantes, subsistemas y sus relaciones.

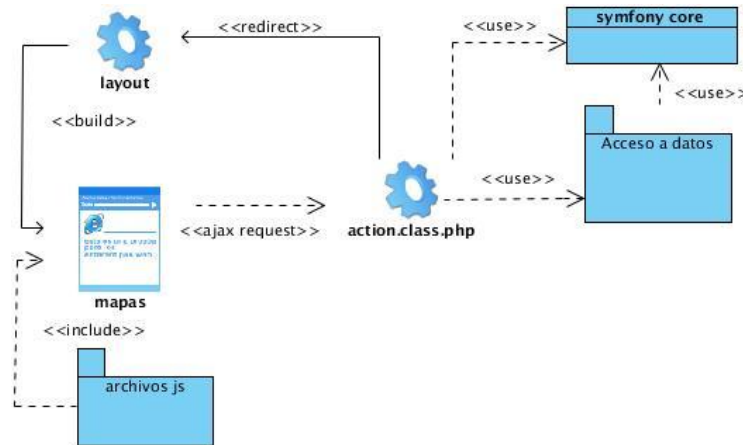
La utilización del *framework* Symfony y el uso de las librerías Ext JS permiten desarrollar un diseño con una estructura similar para varios requisitos funcionales. La página servidora `action.class.php` se encargará de las peticiones que realiza el controlador frontal de Symfony, luego estas se redireccionan a la clase servidora `layout` que es la encargada de crear la plantilla de la página (en el presente caso por ser la plantilla una página en blanco no se presenta en el diagrama), luego se construye la página cliente que importa todas las interfaces de usuario del módulo, estas interfaces están definidas en el fichero de configuración `view.yml` y se encuentran representadas dentro del paquete `archivos.js`.

En la siguiente ilustración se muestra el diagrama de clases del diseño con estereotipos web para el requisito funcional Cargar mapa del mundo. En este caso la `CP_Mapas` realiza un servicio `ajax request` a la clase servidora `mapas.map` para obtener los datos geográficos. La clase servidora `mapas.map` utiliza el plugin `postGIS` para convertir los datos geográficos en la imagen que se muestra en el navegador web.



**Ilustración 8. Diagrama de clases del diseño para requisito funcional “Cargar mapa del mundo”.**

En la ilustración 9 se muestra el diagrama de clases del diseño con estereotipos web para el requisito funcional Localizar lugares. En este caso la clase action.class.php, mediante las clases del paquete de Acceso a Datos y el Núcleo de Symfony (estos paquetes corresponden al sistema externo que utilice el componente), es capaz de obtener la información necesaria para localizar los lugares.



**Ilustración 9. Diagrama de clases del diseño para el requisito funcional "Localizar lugares".**

## 2.9. Diagrama de secuencia

Los diagramas de secuencia muestran el flujo exacto que se lleva a cabo en el desarrollo de un requisito funcional. Estos tipos de diagramas son realizados para definir las acciones que se pueden realizar en la aplicación.

En la ilustración 10 se muestra el diagrama de secuencia para el requisito funcional Cargar mapa del mundo. El mismo comienza cuando el usuario selecciona la opción Ver Mapa, esta interfaz principal creará la interfaz mapas la cual cargará la capa base mundo a través del archivo mapas.map, el cual se encargará de obtener los datos geográficos a través de la base de datos *World*, la misma retornará estos datos al archivo que realizó la consulta y de esta manera se creará el mapa mundo con su capa base; por último se mostrará el mismo al usuario.

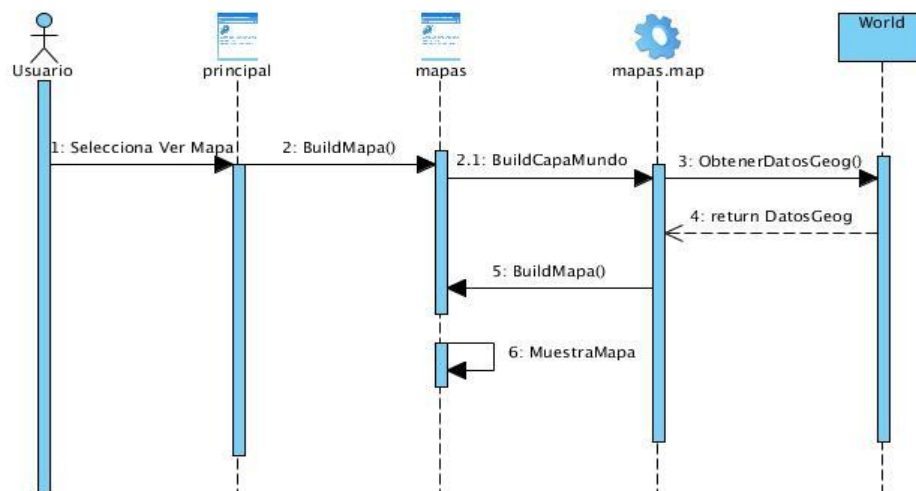


Ilustración 10. Diagrama de secuencias del requisito funcional “Cargar mapa del mundo”.

En la ilustración 11 se muestra el diagrama de secuencia para el requisito funcional localizar lugares. El mismo comienza cuando el usuario selecciona la opción localizar lugares mediante la interfaz, la cual se encargará de enviar un listado con los id de los lugares a ubicar a la clase controladora *action.class.php*, esta obtiene la latitud y longitud de la clase entidad *Lugares*, la cual se encarga de devolver estos datos a la clase controladora, la misma retorna un json a la clase interfaz con los datos necesarios para la posterior ubicación y por último serían ubicados dada su latitud y longitud los lugares deseados.

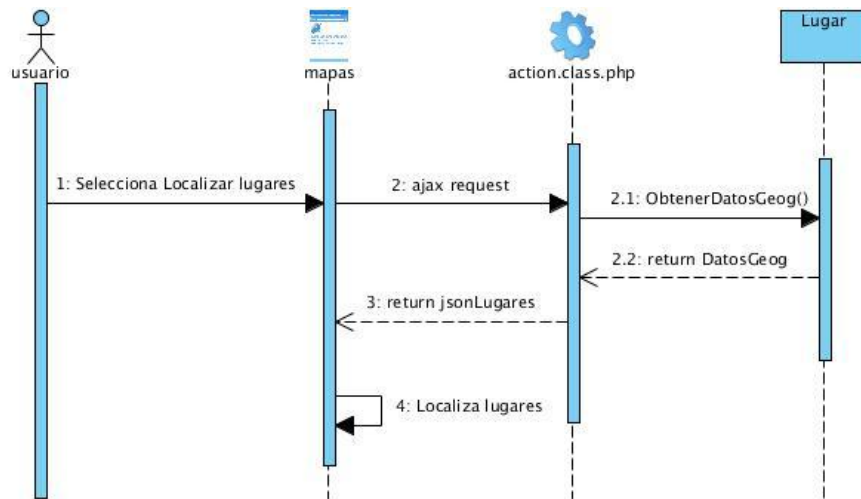


Ilustración 11. Diagrama de secuencias del requisito funcional “Localizar lugares”.

## 2.10. Métricas para la evaluación del diseño

Para realizar la validación a las clases del diseño se seleccionó como métrica Tamaño operacional de clases (TOC) [41].

La aplicación de la métrica TOC define los siguientes atributos de calidad:

- Responsabilidad: Consiste en la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta.
- Complejidad de implementación: Consiste en el grado de dificultad que tiene implementar un diseño de clases determinado.
- Reutilización: Consiste en el grado de reutilización presente en una clase o estructura de clase, dentro de un diseño de software.

El TOC está dado por el número de métodos asignados a una clase.

Atributo de calidad	Modo en que lo afecta
Responsabilidad	Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
Complejidad de implementación	Un aumento del TOC implica un aumento de la

	complejidad de implementación de la clase.
Reutilización	Un aumento del TOC implica una disminución del grado de reutilización de la clase.

**Tabla 2. Tamaño operacional de clase (TOC).**

La clase mapas es la encargada de implementar el componente sfGinaMaps, apoyándose en las librerías necesarias, esta cuenta con 12 métodos. En cuanto al tamaño operacional de clase la misma tiene una responsabilidad y complejidad de implementación baja debido a la gran cantidad de métodos con que cuenta, mientras que la reutilización de la clase es alta. En la siguiente tabla se indican los resultados obtenidos al aplicar la métrica TOC.

<b>Atributo de calidad</b>	<b>Categoría</b>	<b>Criterio</b>
Responsabilidad	Baja	<=Promedio
Complejidad de implementación	Baja	<=Promedio
Reutilización	Alta	<=Promedio

**Tabla 3. Categoría que ocupa la clase mapas en correspondencia con la evaluación de la métrica (TOC).**

## **2.11. Conclusiones parciales**

En este capítulo se especificaron todos los requisitos funcionales y no funcionales del componente a desarrollar, logrando un acuerdo entre los clientes y el equipo de desarrollo.

Al aplicar la técnica de validación: revisión formal de requerimientos se garantizó que los mismos realmente resuelven el problema de la presente investigación.

Se definió la arquitectura del componente, la misma establece la estructura, funcionamiento e interacción entre las distintas partes del mismo, lo que facilita el marco de referencia necesario para guiar la construcción de sfGinaMaps.

Se realizaron los diagramas de secuencia y diagramas de clases del diseño con estereotipos web los cuales facilitan la implementación de los requisitos funcionales.

### 3. CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA

#### 3.1. Introducción

El componente a implementar podrá ser utilizado por todo sistema externo que necesite ubicar lugares y rutas en un mapa global, siempre y cuando cumpla con una serie de precondiciones que serán explicadas en el presente capítulo. El único instrumento adecuado para determinar el estado de la calidad de un producto software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos [19]. Para medir la calidad del componente desarrollado se realizarán pruebas de caja negra, a través de los diseños de caso de prueba garantizados. También se analizarán los resultados obtenidos en las pruebas a través de las dos iteraciones realizadas.

#### 3.2. Precondiciones para utilizar el componente

Para el buen funcionamiento del componente desarrollado es necesario garantizar una serie de precondiciones, ya que sin estas el mismo no cumpliría con el objetivo principal.

A continuación serán mencionadas y explicadas detalladamente, las precondiciones que tiene que cumplir el sistema externo que desee adaptar el plugin desarrollado a sus necesidades específicas:

- Debe estar implementado con el marco de trabajo Symfony versión 1.x.
- Debe tener información de los lugares que se deseen ubicar en el mapa como es un identificador, latitud, longitud y descripción.

Las precondiciones específicas del componente desarrollado se explican a continuación, estas son enviadas como parámetros a través del botón que brinda acceso al componente:

- nombreObj: Nombre de los objetos que se desean ubicar, para que pueda ser actualizado el nombre de la funcionalidad en el visual, así como la leyenda.
- lugares: Una lista que posee el identificador de todos los objetos que se desean ubicar en el mapa.
- url: Una url que contiene la dirección de la función, desde la que se obtienen los datos geográficos (latitud y longitud) para la posterior ubicación de los lugares así como su descripción.

### 3.3. Funcionamiento del componente desarrollado

Las precondiciones mencionadas anteriormente se relacionan de la siguiente manera:

- Cuando se desea realizar la ubicación geográfica de algún lugar, el usuario presiona el botón seleccionado para el acceso al componente.
- Al mismo tiempo garantiza los parámetros anteriormente mencionados (nombreObj, lugares, url).
- Cuando se presiona el botón que indica la funcionalidad “ubicar lugares”, se envían los parámetros necesarios para la ubicación a la dirección especificada, utilizando la función AjaxRequest().
- Esta función devuelve un arreglo que incluye latitud, longitud, descripción y el identificador de todos los objetos que se desean ubicar.
- Por último se ubican los lugares deseados en el mapa con los datos garantizados.

Todo sistema externo que necesite ubicar lugares en un mapa global puede satisfacer sus necesidades utilizando el componente desarrollado, siempre y cuando cumpla con las precondiciones explicadas en el epígrafe 3.2. El acceso al mismo se garantiza a través de un botón que crea una instancia de la clase mapas y garantice las precondiciones específicas del componente definidas anteriormente.

### 3.4. El componente orientado a la arquitectura del GINA

El Sistema de Gestión Integral de Aduanas necesita un componente para la ubicación de puertos y aeropuertos en un mapa global, para satisfacer las necesidades del mismo es necesario garantizar la integración del componente desarrollado al proyecto GINA.

La ilustración 12 muestra cómo se integra el componente sfGinaMaps al Sistema de Gestión Integral de Aduanas. La carpeta que contiene el componente lleva por nombre sfGinaMapPlugin y consta de 4 carpetas internas config, lib, modules y web:

- En la carpeta “lib” se incluyen las librerías específicas para el desarrollo del componente.
- En la carpeta “modules” se incluye el módulo maps el cual incluye los archivos de configuración de Symfony.
- En la carpeta “web” se incluyen todos los archivos necesarios para la creación de las interfaces visuales.

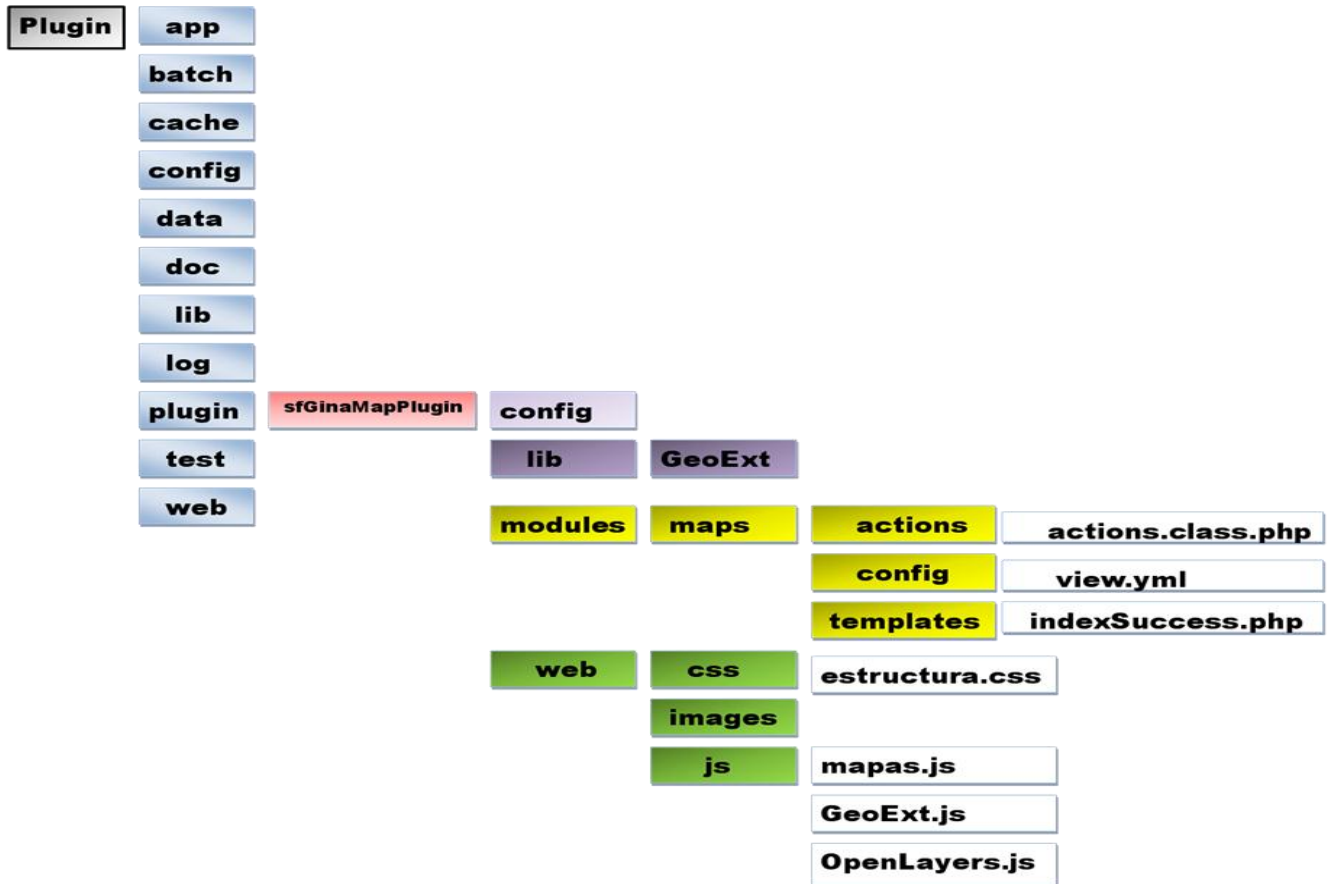
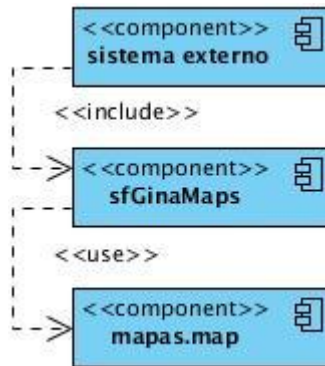


Ilustración 12. El componente sfGinaMaps orientado a la arquitectura del GINA.

### 3.5. Diagrama de componentes

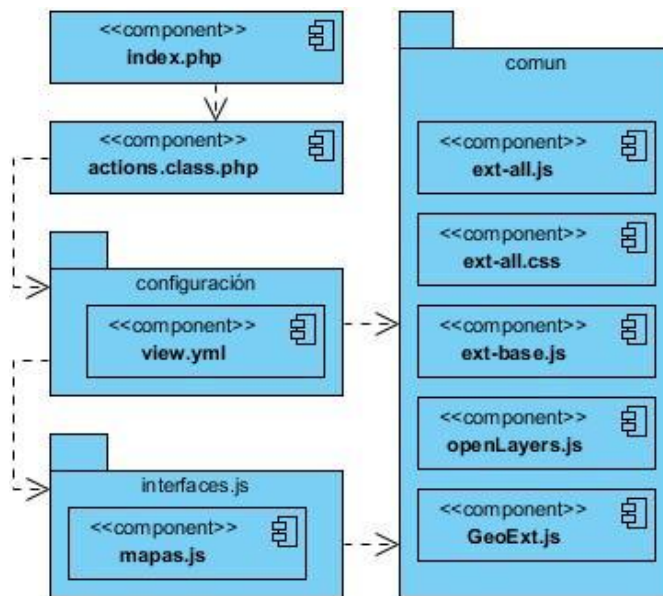
El sistema externo que necesite ubicar lugares en un mapa global debe incluir el componente desarrollado, (aclarar que el sistema externo debe cumplir con las precondiciones que se explican en el epígrafe 3.2), este componente sfGinaMaps utiliza el archivo de configuración de MapServer mapas.map, para convertir datos geoespaciales en imágenes. Lo explicado anteriormente se evidencia en la siguiente ilustración.





**Ilustración 13. Diagrama de componentes.**

En la ilustración 14 se muestra el diagrama de componentes físicos de la aplicación, el cual describe los elementos físicos del componente desarrollado (sfGinaMaps) y sus relaciones. El sistema tiene un único acceso mediante el controlador frontal de Symfony, este envía las peticiones a la clase controladora `action.class.php` y la misma a su vez a través de la configuración del sistema interactúa con las interfaces visuales, las cuales utilizan la librería GeoExt.



**Ilustración 14. Diagrama de componentes físicos de la aplicación.**

### 3.6. Diagrama de despliegue

En la ilustración 15 se muestra el diagrama de despliegue correspondiente a la aplicación. En el mismo las computadoras son representadas por nodos con sus respectivos estereotipos, los cuales permiten precisar la naturaleza del equipo. En una unidad de procesamiento se encuentra el servidor de aplicaciones, el cual a través del protocolo HTTP o HTTPS realiza una conexión al servidor MapServer; este a través del servicio TCP/IP se conecta al sistema gestor de base de datos (SGBD) PostgreSQL.

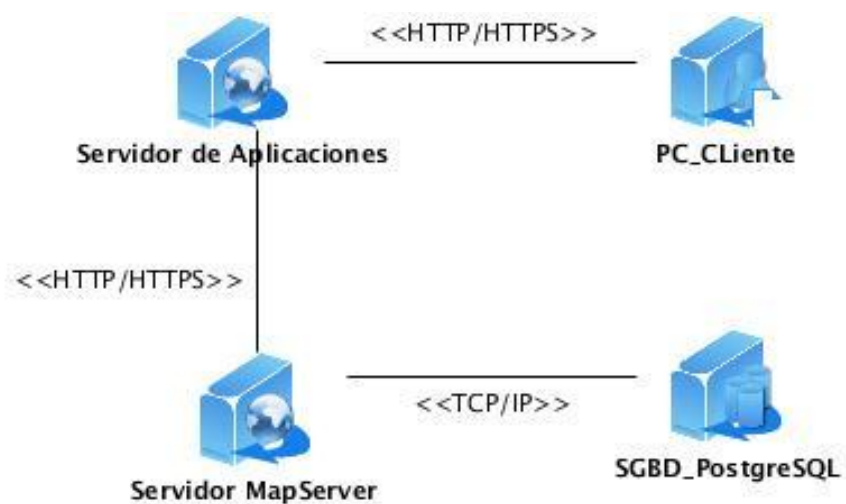
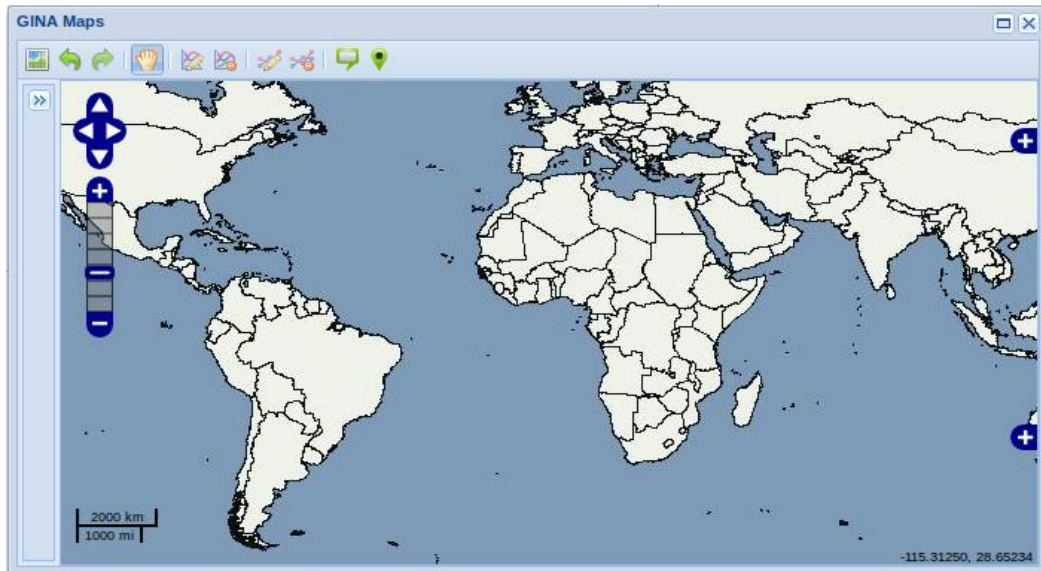


Ilustración 15. Diagrama de despliegue.

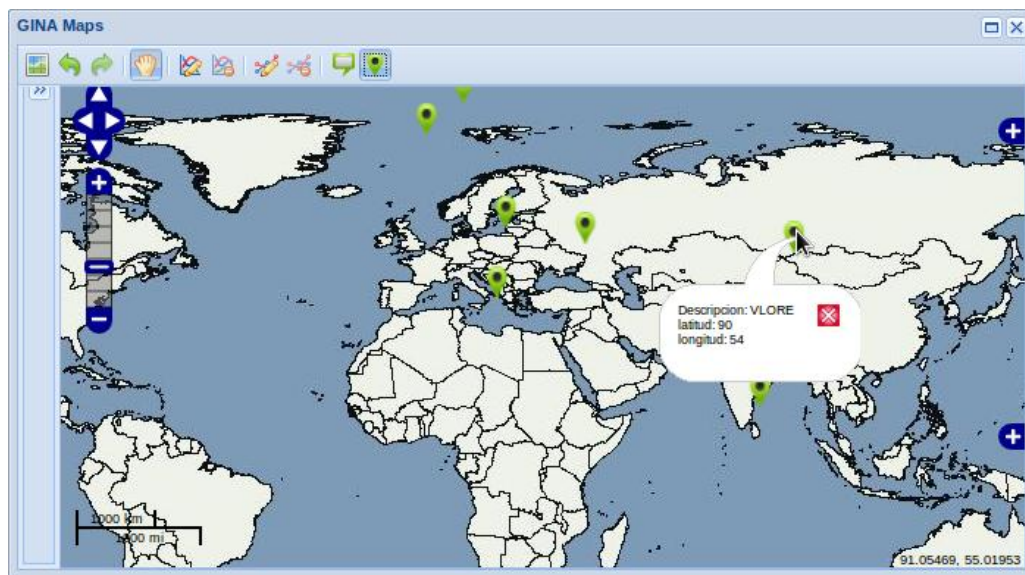
### 3.7. Interfaz de usuario

El sistema cuenta con una pantalla principal, la misma se muestra en la ilustración 16. Esta interfaz presenta el mapa del mundo con varios controles que permiten la navegación sobre el mismo.



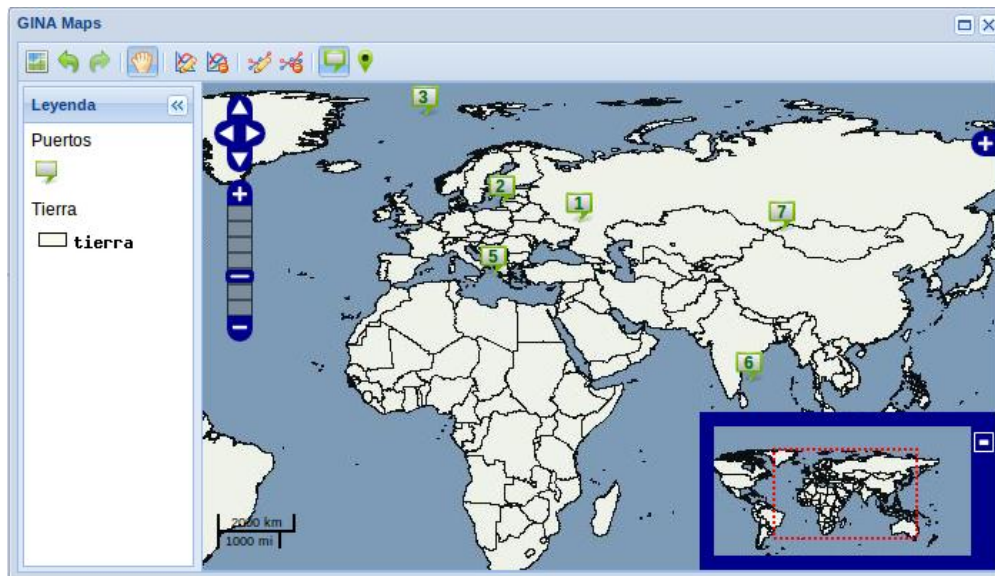
**Ilustración 16. Pantalla principal.**

En la ilustración 17 se muestra la interfaz de usuario para los requisitos funcionales Ubicar lugares y Mostrar descripción de lugares.



**Ilustración 17. RF Ubicar lugares y RF Mostrar descripción de lugares.**

En la ilustración 18 se muestra la interfaz de usuario para los requisitos funcionales Mostrar rutas ascendentes, Mostrar leyenda y Mostrar mapa de referencia.



**Ilustración 18. RF Mostrar rutas ascendentes, RF Mostrar leyenda y RF Mostrar mapa de referencia.**

En la ilustración 19 se muestra la interfaz de usuario para dos requisitos funcionales adicionales que fueron implementados Dibujar líneas y Dibujar polígonos. Estos son de gran ayuda para analizar una determinada ruta ubicada en el mapa o un área específica que puede ser señalada utilizando los polígonos.

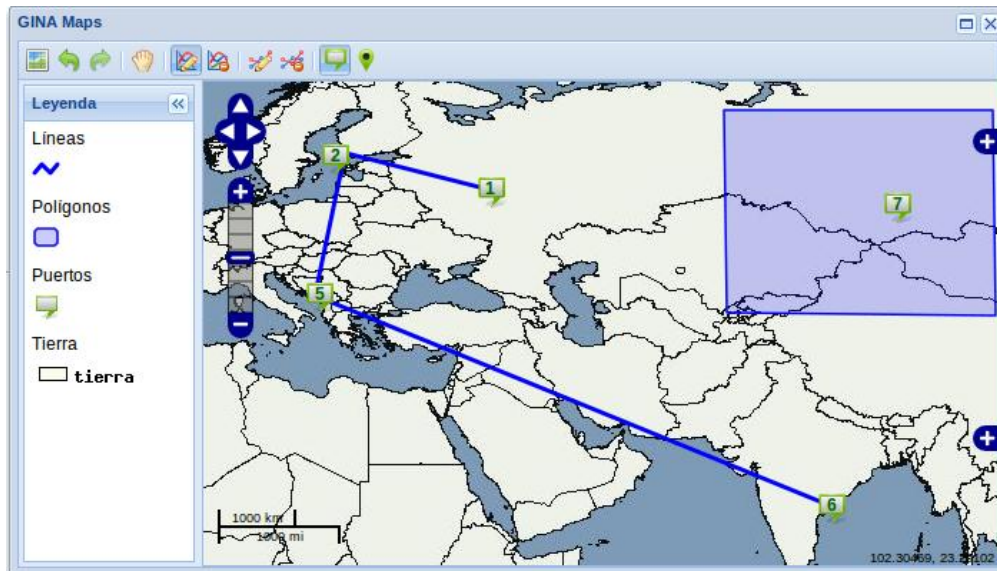


Ilustración 19. Dibujar líneas y Dibujar polígonos.

### 3.8. Pruebas de software

En el proceso desarrollo de software, una de las fases que tiene gran importancia es el proceso de pruebas, ya que es el único instrumento adecuado para determinar el estado de la calidad de un producto software. Entre las más utilizadas, están las pruebas de caja negra [20].

#### Pruebas de caja negra

Las mismas se centran en lo que se espera de un módulo. Por ello son denominadas pruebas funcionales; el probador se limita a suministrar datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro, estas fueron seleccionadas para aplicar en el presente trabajo. Las pruebas de caja negra pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que el resultado producido es correcto. Permite obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa [40]. Las pruebas de este tipo tienen como objetivo encontrar errores de las siguientes categorías: funciones incorrectas o ausentes; errores de interfaz, en estructuras de datos o en acceso a bases de datos externas; errores de rendimiento, de inicialización y de terminación.

Dentro de las pruebas de caja negra la partición equivalente es uno de los métodos más efectivos pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico [42]. A continuación, se muestra un resumen de los diseños de casos de prueba (DCP) confeccionados para evaluar las funcionalidades del componente utilizando el método particiones equivalentes.

<b>Casos de prueba</b>	<b>Escenario</b>	<b>Respuesta del sistema</b>	<b>Flujo central</b>
CP para el requisito funcional Cargar el mapa del mundo.	EC 1 Mostrar mapa.	El sistema muestra el mapa del mundo con varios controles que permiten la navegación por el mismo.	El usuario presiona el botón que da acceso al componente sfGinaMaps.
CP para el requisito funcional Ver todo el mapa.	EC 2 Ver todo el mapa.	El sistema muestra el mapa en correspondencia con la escala del mapa inicial.	El usuario selecciona en el menú principal la opción Ver Mapa Inicial.
CP para el requisito funcional Mostrar información de los botones utilizados.	EC 3 Mostrar información de los botones utilizados.	Al pasar el mouse por encima de un botón el sistema muestra una descripción de este.	El usuario desliza el mouse por encima del botón que desea conocer la descripción.
CP para el requisito funcional Realizar acción anterior.	EC 4 Realizar acción anterior.	El sistema muestra el mapa anterior al visualizado en la pantalla.	El usuario selecciona en el menú principal la opción Anterior que se indica con una flecha señalando a la izquierda.
CP para el requisito funcional Realizar acción siguiente.	EC 5 Realizar acción siguiente.	Una vez que haya seleccionado la acción anterior, el sistema vuelve	El usuario selecciona en el menú principal la opción Siguiente que se

		al estado en que se encontraba el mapa.	indica con una flecha señalando a la derecha.
CP para el requisito funcional Acercar el mapa.	EC 6 Acercar el mapa.	El sistema acerca el mapa, disminuyendo la escala de visualización.	El usuario presiona doble clic sobre el mapa y de esta manera disminuye la escala de visualización del mismo.
CP para el requisito funcional Alejar el mapa.	EC 7 Alejar el mapa.	El sistema aleja el mapa, aumentando la escala de visualización.	El usuario selecciona en la región superior izquierda el botón que indica esta operación y de esta manera aumenta la escala de visualización.
CP para el requisito funcional Mover el mapa.	EC 8 Mover el mapa.	El sistema mueve el mapa hacia la dirección indicada.	El usuario selecciona en la región superior izquierda el botón que indica esta operación en dependencia con la dirección hacia la que desea mover el mapa.
CP para el requisito funcional Localizar lugares.	EC 9 Localizar lugares.	El sistema ubica en el mapa los lugares garantizados por el usuario, señalando con un marcador la posición de estos.	El usuario selecciona en el menú principal la opción Localizar Lugares.
CP para el requisito funcional Mostrar rutas ascendentes.	EC 10 Mostrar rutas ascendentes.	El sistema ubica en el mapa los lugares garantizados por el usuario y señala con un marcador	El usuario selecciona en el menú principal la opción Mostrar Rutas Ascendentes.

		la posición de estos, así como un número que garantiza el orden en que se ubicaron los lugares.	
CP para el requisito funcional Mostrar información de lugares.	EC 11 Mostrar información de lugares.	Al pasar el mouse por encima de un lugar ubicado en el mapa el sistema muestra una descripción de este.	El usuario desliza el mouse por encima del lugar que desea conocer la descripción.
CP para el requisito funcional Mostrar mapa de referencia.	EC 12 Mostrar mapa de referencia.	El sistema muestra un mapa de referencia para que el cliente tenga una mejor navegación por el mapa.	El usuario da clic en la pestaña que se encuentra en la parte inferior derecha del mapa y se despliega el mapa de referencia.
CP para el requisito funcional Mostrar leyenda.	EC 13 Mostrar leyenda.	El sistema muestra una leyenda para tener una mayor información de las representaciones realizadas en el mapa.	El usuario despliega el panel situado a la izquierda, dando clic en el botón que indica esta operación, desplegándose la leyenda del mismo.
CP para el requisito funcional Dibujar líneas.	EC 14 Dibujar líneas.	El sistema muestra las líneas dibujadas por el usuario.	1- El usuario tiene que localizar los lugares o las rutas. 2- Selecciona en el menú principal la opción Dibujar Líneas. 3- Marca en el mapa donde comienza la línea con un clic y doble clic



			para indicar la posición en que termina la misma.
CP para el requisito funcional adicional Dibujar polígonos.	EC 15 Dibujar polígonos.	El sistema muestra los polígonos dibujados por el usuario.	1- El usuario tiene que localizar los lugares o las rutas. 2- Selecciona en el menú principal la opción Dibujar Polígonos. 3- Marca en el mapa donde comienza el polígono con un clic y doble clic para indicar la posición en que termina el mismo.

**Tabla 4. Resumen de los diseños de casos de prueba.**

### **3.9. Resultados de las pruebas**

En la ilustración 20 se muestran los resultados de la primera iteración de las pruebas de caja negra realizadas al sistema. Las mismas fueron ejecutadas por analistas del Departamento de Soluciones para la Aduana y apoyadas en los diseños de casos de prueba descritos anteriormente. Se presentaron 15 escenarios de prueba, en estos fueron encontradas 4 no conformidades, clasificadas como significativas, lo cual representó un 74% de aceptación del sistema.

### Resultados de las pruebas

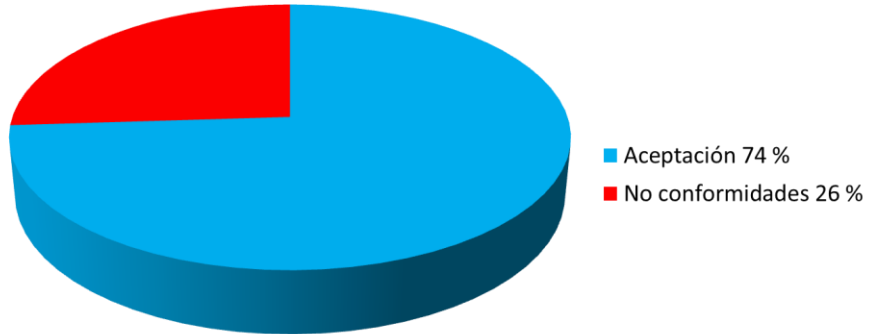


Ilustración 20. Resultados de las pruebas.

En la siguiente tabla se muestra el resultado de esta iteración:

Funcionalidades	No conformidades	Error
Localizar lugares.	1	Cuando se desea ver la descripción del ícono del menú Localizar puertos este no cumple con el estándar de interfaz de usuario.
Mostrar leyenda.	1	Muestra un título indefinido al lado de cada capa vectorial.
Mostrar información de botones utilizados.	2	Cuando se desea ver la descripción del ícono del menú Navegar este no cumple con el estándar de interfaz de usuario. En el menú existen errores ortográficos (líneas y polígonos).

Tabla 5. Resultados de la primera iteración.

Para mayor información acerca de los resultados obtenidos en la primera iteración de las pruebas aplicadas consultar el documento NC- GINA MAPS.

En la segunda iteración las no conformidades habían sido resueltas y no se encontró ninguna nueva, obteniéndose un 100% de aceptación, de esta manera quedó demostrada la calidad del componente desarrollado. La ilustración 21 muestra los resultados obtenido en ambas iteraciones.

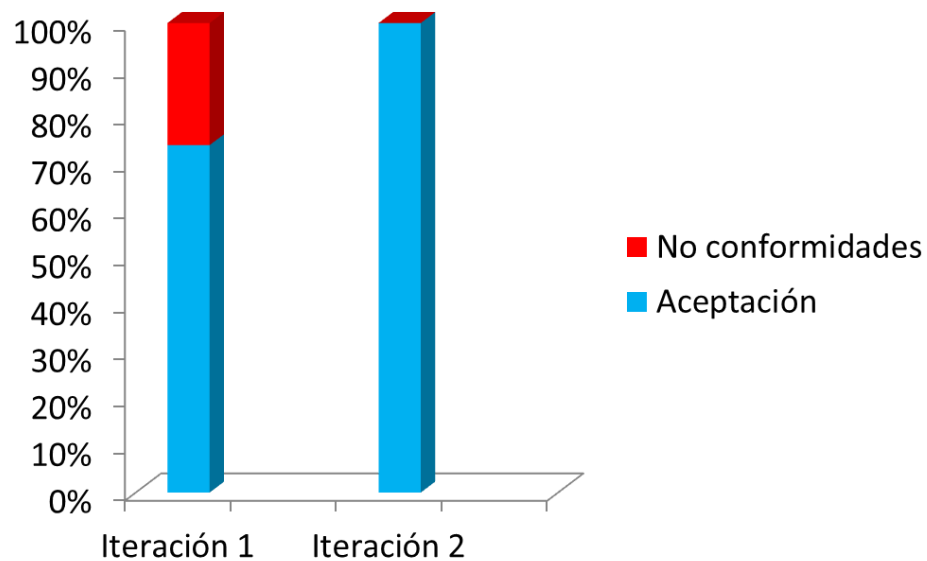


Ilustración 21. Resultados de las iteraciones.

### 3.10. Conclusiones parciales

En el presente capítulo fueron explicadas las condiciones necesarias para utilizar el componente implementado, lo que facilita la utilización del mismo por sistemas externos que requieran de sus funcionalidades.

Se realizó la integración de sfGinaMaps al proyecto GINA, de esta manera se satisfacen las necesidades de ubicación geográfica en un mapa global, que tienen los trabajadores de la AGR.

Se realizó la validación de la solución utilizando las pruebas de caja negra las mismas permitieron corroborar el correcto y completo funcionamiento del componente desarrollado, logrando de esta manera un producto totalmente funcional y que cumple con todos los objetivos propuestos inicialmente.

## CONCLUSIONES

Después de realizar un estudio de los Sistemas de Información Geográfica, haciendo énfasis en las soluciones internacionales y las desarrolladas en la universidad, se puede decir que las mismas no pueden ser utilizadas para darle solución al problema propuesto en el marco de la presente investigación, pues no permiten ubicar lugares y rutas sin la necesidad de consumir servicios de georeferencia que se encuentran públicos en el mundo.

La utilización del sistema gestor de base de datos PostgreSQL y del plugin PostGIS genera como resultado una herramienta eficaz para el trabajo con datos geográficos, constituyendo una opción viable para la implementación de SIG, pues al ser herramienta libre no es necesario pagar costosas licencias.

En la fase de diseño fueron generados los diagramas establecidos por el modelo de desarrollo utilizado, los cuales constituyen un lenguaje común para los desarrolladores facilitando la posterior implementación del sistema.

Con el uso de la librería de JavaScript GeoExt se implementaron interfaces amigables e intuitivas a la vista del usuario. Las mismas se encuentran acorde con el estándar de interfaz de usuario del proyecto GINA.

Al realizar las pruebas de caja negra al sistema, se comprobó el correcto funcionamiento de las diferentes funcionalidades, obteniéndose un 100% de aceptación.

## **RECOMENDACIONES**

Después de haber dado cumplimiento a los objetivos de la investigación y teniendo en cuenta las experiencias obtenidas en la misma se recomienda:

Implementar el requisito funcional: Calcular distancia entre dos lugares, lo cual aportaría una mayor información para la toma de decisiones.

También es recomendable que el componente sea revisado por calidad interna del Centro de Informatización para la Gestión de Entidades (CEIGE), para que este tenga una mayor aceptación por parte de los sistemas externos que necesiten de sus funcionalidades.

## GLOSARIO

**Aduana:** La aduana es la oficina pública y/o fiscal que, a menudo bajo las órdenes de un estado o gobierno político, se establece en costas y fronteras con el propósito de registrar, administrar y regular el tráfico internacional de mercancías y productos que ingresan y egresan de un país.

**AJAX:** Asíncrono JavaScript y XML. Es una técnica de desarrollo web para crear aplicaciones web interactivas.

**API:** Se trata del conjunto de llamadas a ciertas bibliotecas que ofrecen acceso a ciertos servicios. Uno de los principales propósitos de una API consiste en proporcionar un conjunto de funciones de uso general. De esta forma, los programadores se benefician de las ventajas de la API haciendo uso de su funcionalidad, evitándose el trabajo de programar todo desde el principio.

**Framework:** Un *framework*, es una estructura de soporte definida mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

**La Aduana General de la República:** Es la encargada de regular el control aduanero aplicable a la entrada, el tránsito, el cabotaje, el trasbordo, el depósito y la salida del territorio nacional de mercancías, viajeros y sus equipajes, bienes y valores sujetos a regulaciones especiales, incluidas la flora y la fauna protegidas, así como los medios en que se transporten, además forma parte de la Administración del Estado y se subordina al Consejo de Ministros.

**Librería Mapstraction:** Permite desarrollar mapas en JavaScript para múltiples proveedores con un mismo código.

**Linux:** Es una implementación de libre distribución UNIX para computadoras personales (PC), servidores, y estaciones de trabajo.

**Software libre:** El software libre es la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

**Toma de decisiones:** Es el proceso mediante el cual se realiza una elección entre las opciones o formas para resolver diferentes situaciones de la vida en diferentes contextos. Consiste, básicamente, en elegir una opción entre las disponibles, a los efectos de resolver un problema actual o potencial.

**.NET:** Es un *framework* de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

## REFERENCIA BIBLIOGRÁFICA

- [1] Lic. Gómez Ferrera, Miriam Cecilia y Lic. Fuentes Domínguez Joel, «La Tecnología informática en la sociedad cubana». [Online]. Disponible: <http://www.pedagogiaprofesional.rimed.cu/especial%20/Joe.htm>. [Consultado el: 24 enero 2013].
- [2] «GIS». [Online]. Disponible: <http://www.vegavaldavia.com/nueva%20carpeta/cursos/gis.htm>. [Consultado el: 24 enero 2013].
- [3] 2. SENDRA, BOSQUE, Sistemas de Información Geográfica, 1992.<sup>a</sup> ed. Madrid : Rialp.
- [4] 6. Botella Plana, A., Munoz Bollas, A., Olivella Gonzalez, R., Olmedillas Hernandez, J.C. AND Rodriguez Lloret, « Introducción a los sistemas de información geográfica y geo telemática ». .
- [5] «MapGuide Project Home | MapGuide Open Source». [Online]. Disponible: <http://mapguide.osgeo.org/>. [Consultado el: 12 abril 2013].
- [6] «Home | deegree». [Online]. Disponible: <http://www.deegree.org/>. [Consultado el: 12 abril 2013].
- [7] Mitchel T «Map Server open source web mapping», 2008
- [8] «Una Introducción a Apache». [Online]. Disponible: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro](http://linux.ciberaula.com/articulo/linux_apache_intro). [Consultado el: 24 enero 2013].
- [9] «El lenguaje de programación PHP». [Online]. Disponible: <http://programandoideas.com/el-lenguaje-de-programacion-php/>. [Consultado el: 24 enero 2013].
- [10] «Por qué utilizar Openlayers y GeoExt - MappingGIS». [Online]. Disponible: <http://mappinggis.com/2012/11/30/por-que-utilizar-openlayers-y-GeoExt/>. [Consultado el: 24 enero 2013].
- [11] «OpenGeo : PostGIS». [Online]. Disponible: <http://opengeo.org/technology/postGIS/>. [Consultado el: 12-abr-2013].
- [12] Potencier, Fabien y Zaninotto Francois «Symfony, guía definitiva.» Publicación: 2008.
- [13] Reynoso Billy « Introducción a la Arquitectura de Software» Universidad de Buenos Aires
- [14] «NetBeans.org». [Online]. Disponible: [https://netbeans.org/index\\_es.html](https://netbeans.org/index_es.html). [Consultado el: 12 abril 2013].
- [15] «UML CASE tool for software development». [Online]. Disponible: <http://www.visual-paradigm.com/product/vpum/>. [Consultado el: 24 enero 2013].
- [16] Centro de la Informatización de la Gestión de Entidades «CEIGE Modelo de desarrollo».
- [17] «Patrones Grasp y Anti-Patrones: un Enfoque Orientado a Objetos desde Lógica de Programación ». 2011 - biblioteca.ucp.edu.co [Consultado el: 24 enero 2013]



- [18] González YD, Romero YF «Patrón Modelo-Vista-Controlador». - Revista Telem@tica, 2012 - [revistatelematica.cujae.edu.cu](http://revistatelematica.cujae.edu.cu) [Consultado el: 24 enero 2013]
- [19] «Pruebas de software». [Online]. Disponible: <http://pruebasdesoftware.com/laspruebasdesoftware.htm>
- [20] Mañas, J.A. Prueba de Programas. 1994 [Consultado el: 03 de mayo de 2013]. Disponible en: [www.lab.dit.upm.es/~lprg/material/apuntes/pruebas/testing.htm](http://www.lab.dit.upm.es/~lprg/material/apuntes/pruebas/testing.htm)
- [21] «Modelo de dominio». [Online]. Disponible: [http://www.ecured.cu/index.php/Modelo\\_de\\_dominio](http://www.ecured.cu/index.php/Modelo_de_dominio)
- [22] «Definición abc». [Online]. Disponible: <http://www.definicionabc.com/ciencia/cartografia.php>  
[Consultado el: 24-ene-2013]
- [23] «Definición abc». [Online]. Disponible: [Consultado el: 24 enero 2013]  
[http://enciclopedia\\_universal.esacademic.com/22695/Informaci%C3%B3n\\_geogr%C3%A1fica](http://enciclopedia_universal.esacademic.com/22695/Informaci%C3%B3n_geogr%C3%A1fica)
- [24] « ¿Por qué elegir PHP?». [Online]. Disponible:  
[http://www.programacion.com/articulo/por\\_que\\_elegir\\_php\\_143](http://www.programacion.com/articulo/por_que_elegir_php_143)
- [25] Martínez, José Carlos « PostGIS 2. Análisis Espacial Avanzado»
- [26] Olaya, Victor «Sistemas de Información Geográfica Volumen 1»
- [27] Google Maps. In: [online]. [Consultado el 3 enero 2013]. Disponible from:  
<http://www.slideshare.net/guestda62e9/google-maps-3393126>.
- [28] Home - Bing Maps Account Center. In: [online]. [Consultado el: 3 enero 2013]. Disponible from:  
<http://www.bingmapsportal.com/>.
- [29] Urrutia Martínez, J. Curso de Cartografía y Orientación 2006, [Consultado el: 15 de noviembre 2012].
- [30] Elemental.Cowen, D. «Lectura en el Centro Nacional de Análisis e Información Geográfica». 1989, Consultado el: 02 diciembre 2012]. Disponible en:  
[www.vegavaldavia.com/nueva%20carpeta/cursos/gis.htm](http://www.vegavaldavia.com/nueva%20carpeta/cursos/gis.htm).
- [31] Arquitectura de Software del proyecto GINA. «CIG-ADU-N-GS-i3401\_Arquitectura de software v2.0.doc». [Consultado el: 12 diciembre 2012].
- [32] « Ingeniería de software séptima edición» In: [online]. [Consultado el: 12 enero 2013]. Disponible from:  
<http://books.google.com/cu/books?id=gQWd49zSut4C&pg=PA145&lpg=PA145&dq=revisi3n+formal+de+requisitos+de+sw&source=bl&ots=s55ayqrzv9&sig=v7rDKbrC0qzlgKzyLaDbUVdoTmw&hl=es-419&sa=X&ei=qaKnUc-eMYTYtQawpoDoCg&ved=0CCgQ6AEwAA#v=onepage&q=revisi3n%20formal%20de%20requisitos%20de%20sw&f=false>

- [33] Arquitectura de software In: [online]. [Consultado el: 12 enero 2013]. Disponible from:  
[http://www.ecured.cu/index.php/Arquitectura\\_de\\_software](http://www.ecured.cu/index.php/Arquitectura_de_software)
- [34] J Rosario «La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual» In: [online]. [Consultado el: 18 enero 2013].
- [35] J Rosario «Componentes y funciones de los SIG» In: [online]. [Consultado el: 18 enero 2013].  
<http://www.rlc.fao.org/es/prioridades/transfron/sig/intro/compo.htm>
- [36] Yahoo Maps Web Services. In: [online]. [Consultado el: 10 diciembre 2012]. Disponible from:  
<http://developer.yahoo.com/maps/>
- [37] de la Cruz Rigueiro, K. SIG\_Verano: Sistema de Información Geográfica para la Oficina Nacional de la Unión de Jóvenes Comunistas. 2011, vol. 4, no. 12 [Consultado el: 15 noviembre 2012]. Disponible en:  
<http://publicaciones.uci.cu/index.php/SC/article/view/581>.
- [38] Sánchez Gutierrez, A. AND Pantoja Zaldivar, Y. Plataforma de Servicios de Mapas UCI (PLASMA v1.0) GeoNews [Type of Work]. 2011, vol. 13, [Consultado el: 12 diciembre 2012]. Disponible en:  
<http://revistageonews.wordpress.com/2011/10/18/plataforma-de-servicios-de-mapas-uci-plasma-v1-0/>.
- [39] Sobre PostgreSQL In: [online]. [Consultado el: 10 diciembre 2012]. Disponible from:  
[http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql)
- [40] Urióstegui Ocampo, Antonino Rafael; 2010 « Manual de Ingeniería de Software I», Zapopan, Jalisco. [Consultado el: 10 marzo 2013].
- [41] Boizan Romero, Denis; 2012. «Sistema para Gestionar la Actividad Científica del Departamento de Informática de la Universidad de Guantánamo».
- [42] Grupo Alarcos, Escuela Superior de Informática (UCLM) «Pruebas del Software» [Consultado el: 15 marzo 2013].
- [43] León Companioni, Alain. SIG\_Hidro: Sistema para la representación geográfica de objetivos hidrográficos de precipitaciones. 2012, [Consultado el: 15 noviembre 2012]. Disponible en:  
<http://publicaciones.uci.cu/index.php/SC/article/view/842>.
- [44] «Técnicas de validación de requisitos» Marco de Desarrollo de la Junta de Andalucía[Consultado el: 15 marzo 2013]. Disponible en:  
<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/419>
- [45] «Geografía Aplicada: Modelos Geográficos» [Consultado el: 20 abril 2013]. Disponible en:  
<http://geomodelos.blogspot.com/>

## ANEXOS

A continuación se especifican los dos anexos utilizados y se ilustran ejemplos de cada uno de ellos. Los mismos son:

- Descripción de requisitos de software.
- Modelo Conceptual.

### Anexo 1: Descripción de requisitos

Para mayor información consultar el documento CIG-ADU-N-GINA-NOT-i2601 Descripción de requisitos.

#### Descripción textual del requisito Cargar mapa del mundo.

<b>Precondiciones</b>	El usuario debe estar autenticado. El sistema externo debe mostrar un enlace al componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Cargar mapa del mundo</b>	
1	Recibe los datos geográficos necesarios del sistema gestor de base de datos PostgreSQL, específicamente de su extensión PostGIS.
2	Estos datos vectoriales son utilizados para la formación de las capas
3	Estas capas (layers) son cargadas en una ventana.
4	Se observa esta ventana con un mapa del mundo dibujado.
5	Concluye el requisito.
<b>Pos-condiciones</b>	
•	Se observa una ventana con un mapa del mundo dibujado.
<b>Requisitos especiales</b>	Tener instalado el SGBD PostgreSQL y el plugin PostGIS para el trabajo con los datos geográficos.

Tabla 6. Descripción textual del requisito Cargar mapa del mundo.

#### Descripción textual del requisito Mover el mapa.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
-----------------------	---

<b>Flujo de eventos</b>	
<b>Flujo básico Mover el mapa</b>	
1.	El usuario da clic en uno de los botones: norte, sur, este, oeste.
2.	El sistema mueve el mapa hacia la dirección indicada por el actor.
3.	El sistema procesa la información y devuelve el área de representación corregida en dependencia de la dirección seleccionada.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1	El sistema muestra el mapa hacia la dirección que seleccionó el usuario.
<b>Flujos alternativos</b>	
<b>Flujo alternativo Nº 2.a Mover el mapa a través del mouse</b>	
1	El usuario mantiene el clic izquierdo presionado mientras mueve el mapa hacia la dirección deseada.
2	El sistema procesa la información y devuelve el área de representación corregida en dependencia de la dirección seleccionada.
3	Concluye el requisito.
<b>Pos-condiciones</b>	
1	El sistema muestra el mapa hacia la dirección que seleccionó el usuario.

Tabla 7.Descripción textual del requisito Mover el mapa.

### Descripción textual del requisito Acercar el mapa.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Acercar el mapa</b>	
1.	El usuario da clic en el botón que indica esta operación (+).
2.	El sistema realiza la operación de zoom+ seleccionada por el actor, donde acerca la imagen del mapa a la vista del usuario.
3.	El sistema disminuye la escala de visualización.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa con el nivel de zoom seleccionado por el usuario.
<b>Flujos alternativos</b>	
<b>Flujo alternativo Nº 2.a Acercar el mapa a través del mouse</b>	
1.	El usuario presiona doble clic izquierdo.
2.	El sistema procesa la información y devuelve el área de representación corregida en dependencia del zoom deseado.
3.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa con el nivel de zoom seleccionado por el usuario.

Tabla 8. Descripción textual del requisito Acercar el mapa.

**Descripción textual del requisito Alejar el mapa.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Alejar el mapa</b>	
1.	El usuario da clic en el botón que indica esta operación (-).
2.	El sistema realiza la operación de zoom- seleccionada por el actor, donde aleja la imagen del mapa a la vista del usuario.
3.	El sistema aumenta la escala de visualización.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa con el nivel de zoom seleccionado por el usuario.

Tabla 9. Descripción textual del requisito Alejar el mapa.

**Descripción textual del requisito Mostrar mapa de referencia.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar mapa de referencia</b>	
1.	El usuario da clic en el botón que indica esta operación.
2.	El sistema procesa la información a partir de la acción realizada por el usuario y muestra en la región inferior derecha el mapa de referencia.
3.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa global y en la esquina inferior derecha el mapa de referencia

Tabla 10. Descripción textual del requisito Mostrar mapa de referencia.

**Descripción textual del requisito Ver todo el mapa.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Ver todo el mapa</b>	

1.	El usuario da clic en el botón que indica esta operación.
2.	El sistema procesa la información a partir de la acción realizada por el usuario y actualiza la visualización del mapa.
3.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa con el nivel de zoom inicial de la aplicación

Tabla 11. Descripción textual del requisito Ver todo el mapa.

### Descripción textual del requisito Realizar acción anterior.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Alejar el mapa</b>	
1.	El usuario da clic en el botón que indica esta operación (-).
2.	El sistema comprueba que tenga almacenado al menos una visualización anterior, en caso de no tener ninguna o de llegar a la última esta funcionalidad se encontrará desactivada.
3.	El sistema visualiza el mapa anterior.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa anterior.

Tabla 12. Descripción textual del requisito Realizar acción anterior.

### Descripción textual del requisito Realizar acción siguiente.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Realizar acción siguiente</b>	
1.	El usuario da clic en el botón que indica esta operación para que una vez que haya seleccionado la opción anterior vuelva al estado en el que se encontraba.
2.	El sistema comprueba que tenga almacenado al menos una visualización siguiente, en caso de no tener ninguna o de llegar a la última esta funcionalidad se encontrará desactivada.
3.	El sistema visualiza el mapa siguiente.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa siguiente.

Tabla 13. Descripción textual del requisito Realizar acción siguiente.

**Descripción textual del requisito Ubicar lugares.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Ubicar lugares</b>	
1.	El usuario presiona el botón que indica la operación ubicar lugares
2.	El sistema muestra los lugares que el usuario desea, ubicados en el mapa.
3.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa con todos los lugares ubicados en este.

Tabla 14. Descripción textual del requisito Ubicar lugares.

**Descripción textual del requisito Mostrar escala del mapa.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar escala del mapa</b>	
1.	El sistema muestra en la región inferior izquierda la escala de visualización, la cual varía en dependencia del nivel de visualización del mapa.
2.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa global y en la parte inferior del mapa la escala del mismo.

Tabla 15. Descripción textual del requisito Mostrar escala del mapa.

**Descripción textual del requisito Mostrar rutas ascendentes.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar rutas ascendentes</b>	
1.	El usuario da clic en el botón que indica esta operación.
2.	El sistema procesa la información a partir de la acción realizada por el usuario y

	ubica en el mapa todos los lugares que integran la ruta.
3.	El sistema señala los puntos que pertenecen a la ruta con un número de forma ascendente, en dependencia del orden en que fueron garantizados por el usuario. Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa global, con la ruta ubicada, la misma muestra los lugares identificados por un número de forma ascendente.

Tabla 16. Descripción textual del requisito Mostrar rutas ascendentes.

### Descripción textual del requisito Mostrar leyenda.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar leyenda</b>	
1.	El usuario da clic en el botón que indica esta operación.
2.	El sistema procesa la información a partir de la acción realizada por el usuario y muestra en la región izquierda la leyenda del mapa
3.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra el mapa global y en la región izquierda la leyenda del mismo.

Tabla 17. Descripción textual del requisito Mostrar leyenda.

### Descripción textual del requisito Mostrar descripción de lugares.

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente, además los lugares están ubicados en el mismo
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar información de lugares.</b>	
1.	Si el usuario desea ver la descripción de un lugar ubicado en el mapa, desliza el mouse encima del mismo.
2.	El sistema debe mostrar un cuadro de texto con la información referente al lugar.
3.	Este cuadro desaparecerá en cuanto el usuario deslice el mouse fuera de la marca del lugar.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra un cuadro de texto con la información del lugar previamente ubicado.



Tabla 18. Descripción textual del requisito Mostrar descripción de lugares.

**Descripción textual del requisito Mostrar información de los botones utilizados.**

<b>Precondiciones</b>	El mapa del mundo está cargado en una ventana y en un menú se muestran los botones con las diferentes funcionalidades del componente.
<b>Flujo de eventos</b>	
<b>Flujo básico Mostrar información de lugares.</b>	
1.	El usuario si desea ver la descripción referente al botón deseado desliza el mouse encima del mismo.
2.	El sistema debe mostrar un cuadro de texto con la descripción del botón.
3.	Este cuadro desaparecerá en cuanto el usuario deslice el mouse fuera del botón.
4.	Concluye el requisito.
<b>Pos-condiciones</b>	
1.	El sistema muestra un cuadro de texto con la descripción del botón deseado.

Tabla 19. Descripción textual del requisito Mostrar información de los botones utilizados.

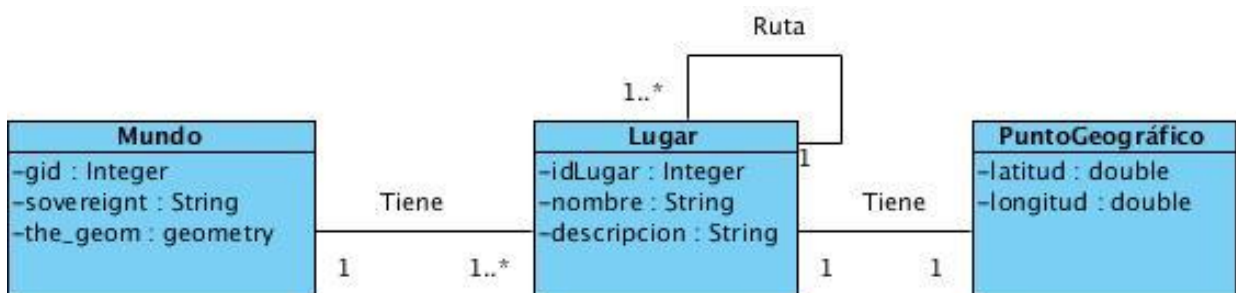
**Anexo 2: Modelo Conceptual**

En el modelo conceptual del presente trabajo se especifican los términos que fueron utilizados en la investigación e implementación y que podrían resultar de difícil entendimiento para el usuario. A continuación se explican las relaciones existentes entre cada clase conceptual y se muestra una descripción de las mismas.

La clase Mundo tiene uno o muchos lugares.

Los lugares pertenecen a una única ruta, mientras que una ruta tiene 1 o muchos lugares.

Los lugares tienen asociado un punto geográfico, mientras que cada punto geográfico pertenece a un solo lugar.



### Descripción de la clase conceptual Mundo:

Descripción		Es utilizada para la creación del mapa global.				
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
					Clases válidas	Clases no válidas
gid	Identificador del país.	Integer	No	Sí	Caracteres numéricos[0..9][números positivos]	Caracteres alfabéticos, incluido caracteres especiales[A..Z, _, -, /, a..z]
SOVEREIGNT	Muestra el nombre del país.	String	No	No	Caracteres [A...Z, a...z]	Caracteres especiales [_, -, /, i, ç, '] Caracteres numéricos del 0..9.
The_geom	Muestra la ubicación geométrica que tiene ese país en el mapa.	geometry	No	Si	Caracteres [A...Z, a...z] numéricos[0..9][números positivos]	Caracteres especiales [_, -, /, i, ç, ']

### Descripción de la clase conceptual PuntoGeográfico:

Descripción		Son utilizados para la ubicación de lugares en el mapa.				
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
					Clases válidas	Clases no válidas
latitud	La posición vertical en el mapa.	Double	No	No	Caracteres numéricos[0..9][números positivos y negativos]	Caracteres alfabéticos, incluido caracteres especiales[A..Z, _, -, /, a..z]

<b>longitud</b>	La posición horizontal en el mapa.	Double	No	No	Caracteres numéricos[0..9][números positivos y negativos]	Caracteres alfabéticos, incluido caracteres especiales[A..Z, _, -, /,a..z]
-----------------	------------------------------------	--------	----	----	---	--

### Descripción de la clase conceptual Lugar:

Descripción						
Contiene la información necesaria para su ubicación geográfica.						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
					Clases válidas	Clases no válidas
<b>idLugar</b>	Identificador del lugar.	Integer	No	Si	Caracteres numéricos[0..9][números positivos]	Caracteres alfabéticos, incluido caracteres especiales[A..Z, _, -, /,a..z]
<b>nombre</b>	Nombre del lugar.	String	No	No	Caracteres [A...Z, a...z]	Caracteres especiales [_, -, /, i, ç, ´] Caracteres numéricos del 0..9.
<b>descripción</b>	Descripción del lugar.	String	No	No	Caracteres [A...Z, a...z]	Caracteres especiales [_, -, /, i, ç, ´] Caracteres numéricos del 0..9.

