

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 6



**TÍTULO: DESARROLLO DE UNA APLICACIÓN DE MAPAS
DE RUTAS DE TRANSPORTE DE LA HABANA SOBRE
GVSIG MOBILE.**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS**

AUTOR:

Yassel Comas Pérez

TUTOR:

Lic. Mario Liosbel Díaz Abreu

CO-TUTOR:

Ing. Odiel Estrada Molina

La Habana, Junio, 2011.
“Año 53 de la Revolución”

*“La mejor forma de predecir el futuro es
implementarlo”*

David Heinemeier Hansson



Dedicatoria

A mis padres Reina y Mario, por ser las personas que más quiero en el mundo. Por brindarme todo su amor, esfuerzo y comprensión. Por apoyarme cuando más lo he necesitado.

A mis hermanos Lenier y Michel por quererme, consentirme y cuidarme tanto. Por darme fuerzas para seguir cada día adelante.



Agradecimientos

*Quiero agradecer a mis padres y hermanos que les debo
todo lo que soy.*

*A Danny por soportarme, apoyarme y estar conmigo
en todo momento.*

A mis sobrinas Laura y Claudia por quererme tanto.

A toda mi familia por darme su apoyo y cariño.

A Anita y Erne por quererme como a un hijo.

A mi tutor por toda su ayuda prestada.

*A todos los profesores de mi proyecto por el tiempo que
me dedicaron.*

A todos mis amigos por su incondicionalidad.

*A todas aquellas personas que de una forma u otra
aportaron su granito de arena en mi vida, aunque
no las mencione siempre estarán en mi corazón.*

Muchas Gracias



Declaración de Autoría

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a 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 _____.

Yassel Comas Pérez
Autor

Mario Liosbel Díaz Abreu
Tutor



Tutor: Lic. Mario Liosbel Díaz Abreu

- Profesión: Licenciado en Ciencia de la Computación
- Categoría docente: Instructor Recién graduado
- Año de graduado: 2008
- Correo electrónico: wariol@uci.cu

Co-Tutor: Ing. Odiel Estrada Molina

- Profesión: Ingeniero en Ciencias Informáticas
- Categoría docente: Instructor Recién graduado
- Año de graduado: 2010
- Correo electrónico: oestrada@uci.cu



La Habana, 21 de Junio del 2011
"Año 53 de la Revolución"

De: Odiel Estrada Molina.
Líder del proyecto SIG-Móviles.

A: Miembros del Tribunal # 24.
Facultad 6.

Por medio del presente documento certifico que la investigación realizada por el estudiante Yassel Comas Pérez y tutorados por el Lic. Mario Liosbel Díaz Abreu e Ing. Odiel Estrada Molina tributa de forma íntegra al proyecto SIG-Móviles. Se realizó un adecuado estudio de la herramienta gvSIG Mobile, permitiéndole agregar nuevas funcionalidades y adecuarlo a las características de la cartografía del Transporte Urbano de La Habana y a la información socio-económico correspondiente.

A continuación se exponen los elementos que amparan la afirmación anterior:

1. Implementación de nuevas funcionalidades en la plataforma gvSIG Mobile como la: localización de paradas, rutas, paradas de intercambio, así como la visualización de la información socio-económica referente a estos requisitos.
2. Se realizó un correcto trabajo colaborativo con el especialista de cartografía del Centro
3. Se realizó correctamente la personalización sobre el Transporte Urbano de La Habana adaptándola a las características socio-económicas de las mismas.

Los resultados de esta investigación fue presentado en:

1. La Jornada Científica Estudiantil 2011 a nivel UCI.

Finalmente, se solicita adjuntar este documento al expediente de Tesis del mencionado estudiante. Para que así conste, firmo la presente, en un ejemplar, en la Universidad de las Ciencias Informáticas a los 30 días del mes de Mayo del año 2011.

Ing. Odiel Estrada Molina
Líder del proyecto



Resumen

El sistema de transporte público de La Habana está compuesto por un gran número de rutas de ómnibus con sus respectivas paradas. Tener un conocimiento actualizado sobre esta información puede resultar engorroso, por lo que se efectuó la siguiente investigación para solucionar dicho problema. En el presente trabajo de diploma se realizó un análisis sobre el estado del arte de los Sistemas de Información Geográfica (SIG) para los dispositivos móviles, se identificó un grupo de funcionalidades a partir del estudio de un conjunto de SIG tanto a nivel nacional como internacional y además se llevo a cabo un análisis de las principales herramientas y tecnologías para seleccionar las más adecuadas en el proceso de construcción de *software*. Se implementó un SIG para dispositivos móviles sobre las rutas de transporte de La Habana utilizando la plataforma GvSIG *Mobile* con vistas a lograr una personalización de la aplicación. En las pruebas realizadas a la aplicación se alcanzaron resultados satisfactorios, garantizando que esta constituye una herramienta de apoyo al proceso de toma de decisiones.

Palabras claves: transporte, ómnibus, ruta, parada, mapa, cartografía, información geográfica, SIG.



Tablas y figuras

Tabla #1: Comparación modelo vector vs ráster.	10
Figura #1: Modelo del Dominio.	30
Tabla #2: Descripción del actor del sistema.	36
Figura #2: Diagrama de Casos de Uso del Sistema.	36
Tabla #3: Descripción textual del caso de uso Localizar_parada.	39
Tabla #4: Descripción textual del caso de uso Localizar_ruta.	42
Tabla #5: Clasificación de las clases del análisis.	44
Figura #3: Diagrama de Clases del Análisis Localizar_parada.	44
Figura #4: Diagrama de Clases del Análisis Localizar_ruta.	44
Figura #5: Diagrama de Colaboración del Análisis Localizar_parada.	45
Figura #6: Diagrama de Colaboración del Análisis Localizar_ruta.	45
Figura #7: Diagrama de Secuencia del Análisis Localizar_parada.	46
Figura #8: Diagrama de Secuencia del Análisis Localizar_ruta.	47
Figura #9: Diagrama de Clases del Diseño Localizar_parada.	49
Figura #10: Diagrama de Clases del Diseño Localizar_ruta.	50
Figura #11: Estructura de los ficheros shapefile.	53
Figura #12: Diagrama de despliegue	54
Figura #13: Diagrama de componentes.	55
Figura #14: Diagrama de componentes. Lógica Geo.	55
Figura #15: Diagrama de componentes. Interfaz.	56
Figura #16: Diagrama de componentes. Acceso a datos.	56
Tabla #6: Secciones a probar en el caso de uso.	58
Tabla #7: Descripción de variables	59
Tabla #8: Matriz de datos de la SC 1 Localizar parada.	59



Tabla de contenido

INTRODUCCIÓN	3
CAPÍTULO 1: Fundamentación Teórica.....	6
1.1 Introducción	6
1.2 Conceptos asociados al dominio del problema	6
1.3 Objeto de estudio.....	7
1.3.1 Descripción general	7
1.3.1.1 Mapas	7
1.3.1.2 Sistemas de Información Geográfica.....	8
1.3.1.3 Telefonía Móvil.....	14
1.3.1.4 SIG para dispositivos móviles	15
1.4 Análisis de soluciones existentes.....	18
1.5 Herramientas y tecnologías a utilizar	19
1.6 Conclusiones	28
CAPÍTULO 2: Presentación de la solución propuesta	29
2.1. Introducción	29
2.2. Entorno donde trabajará el sistema	29
2.2.1. Conceptos y eventos principales del entorno	29
2.2.2. Diagrama de clases del Modelo de Dominio.....	30
2.2.3. Glosario de Términos del Dominio	30
2.3. Requisitos Funcionales.....	31
2.4. Requisitos No Funcionales	34
2.5. Descripción del Sistema Propuesto	36
2.5.1. Descripción de los actores	36
2.5.2. Casos de Uso del Sistema	36
2.6. Conclusiones.....	42
CAPÍTULO 3: Construcción de la solución propuesta.....	43
3.1 Introducción	43
3.2 Modelo de Análisis.....	43
3.3 Modelo de Diseño.....	47



3.4	Principios de diseño.....	51
3.5	Almacenamiento de datos	51
3.6	Implementación	53
3.7	Prueba del sistema propuesto	56
3.8	Conclusiones	59
CONCLUSIONES		60
RECOMENDACIONES		61
REFERENCIAS BIBLIOGRÁFICAS		62
BIBLIOGRAFÍA.....		68
GLOSARIO DE TÉRMINOS		71



INTRODUCCIÓN

La telefonía móvil es una tecnología joven que ha tenido auge en el mundo, se ha convertido en una herramienta esencial para las personas. Martin Cooper, junto a otros individuos, desarrollaron y fabricaron el primer celular en 1973. [1] Su evolución ha permitido que disminuyan tanto en tamaño como en peso y aunque su principal función es la comunicación, se han integrado otros tipos de servicios relacionados con los datos, audio y video, así como una amplia gama de funcionalidades que aumentan diariamente posibilitando que sea un producto de preferencia.

El uso correcto de las nuevas tecnologías ofrece múltiples ventajas como la generación de riquezas y el mejoramiento de la calidad de vida de los individuos. Alcanzar la informatización de todos los sectores sociales es una meta trazada en Cuba para obtener mayores beneficios. Numerosas empresas cubanas no cuentan con la tecnología y las herramientas necesarias para introducirse por completo en el proceso, por lo que estos son los primeros pasos en los que se está trabajando para lograr ese objetivo.

La Universidad de las Ciencias Informáticas (UCI) se ha sumado a la tarea de informatizar el país, contribuye de forma directa con el desarrollo de *software* para su posterior utilización en la rama de la economía u otros sectores de la sociedad. La UCI posee actualmente una estructura organizativa de siete facultades, las cuales cuentan con diferentes centros de producción e investigación implicados en la producción de *software*.

La facultad 6 cuenta con el Centro de Desarrollo Geoinformática y Señales Digitales (GEYSED), se compone de dos departamentos como su nombre lo indica, los cuales a su vez integran diferentes proyectos. El proyecto SIG-Móviles pertenece al departamento Geoinformática y su trabajo está enfocado al desarrollo de Sistemas de Información Geográfica (SIG) para dispositivos móviles.

Como en muchas de las ciudades que albergan un gran número de habitantes, La Habana cuenta con un sistema de transporte público. El mismo se compone de un gran número de rutas, que permiten a las personas desplazarse de un sitio a otro dentro de la ciudad. Dichas rutas están sujetas a cambios o modificaciones (ampliación, desviación, creación y eliminación de rutas) que deben ser conocidos por la población.



Con la intención de divulgar este tipo de información, la confección de mapas constituye una solución parcial a dicho problema ya que el volumen de datos a visualizar es limitado y por ende las consultas realizadas podrían ofrecer información básica. A partir de la situación problemática expuesta anteriormente se identificó el siguiente **problema a resolver**: Al consultar sobre la localización de las rutas de transporte de La Habana se obtienen resultados con un margen de error debido a la imposibilidad de acceder a esta información de forma confiable.

Como **objetivo general** de esta investigación se plantea: desarrollar un Sistema de Información Geográfica sobre GvSIG *Mobile* que permita a los usuarios de dispositivos móviles acceder a los mapas de rutas de transporte de La Habana. El **objeto de estudio** lo conforman los Sistemas de Información Geográfica para dispositivos móviles, delimitando su **campo de acción** a los Sistemas de Información Geográfica basados en GvSIG *Mobile*.

Con el fin de guiar el desarrollo de este proceso se plantea la siguiente **idea a defender**:

Si se desarrolla un Sistema de Información Geográfica utilizando GvSIG *Mobile*, se logrará que a través de dispositivos móviles las personas puedan acceder a información confiable sobre las rutas de transporte de La Habana.

Para cumplir con el objetivo general propuesto y dar solución a la situación problemática planteada, se proponen las tareas siguientes:

Tareas de investigación:

1. Caracterizar tendencias actuales, tecnologías y conceptos más importantes relacionados con los Sistemas de Información Geográfica para definir el estado del arte.
2. Caracterizar las principales herramientas de código abierto y estándares que soporta el GvSIG *Mobile*.
3. Identificar las prácticas para la implementación de un sistema informático sobre GvSIG *Mobile* que cumpla con los estándares establecidos.
4. Analizar y seleccionar la metodología de desarrollo de *software* y herramientas para su utilización en el desarrollo de la aplicación.
5. Implementar la aplicación informática sobre GvSIG *Mobile* para que los usuarios de dispositivos móviles puedan hacer uso de sus funcionalidades.
6. Validar el sistema informático.



Resultado esperado:

Sistema de Información Geográfica de mapas de rutas de transporte de La Habana.

Para tributar al cumplimiento del objetivo general de la presente investigación se propone el uso de los siguientes **métodos científicos**. [2]

Métodos teóricos

Analítico-sintético: Se utilizó en el análisis de las bibliografías relacionadas con los Sistemas de Información Geográfica para dispositivos móviles, posibilitando su comprensión profunda y permitiendo la captura de aquellos elementos que se relacionan con el objeto de estudio.

Histórico-lógico: Se utilizó en el análisis de los Sistemas de Información Geográfica para dispositivos móviles.

Modelación: Se materializó con la elaboración del análisis y diseño de la aplicación, a través de los diversos diagramas obtenidos durante el desarrollo del producto.

Métodos empíricos

Observación: Se utilizó en la identificación del problema científico una vez analizada la situación problemática, además en la fase de pruebas con el fin de comprobar el cumplimiento de los requisitos.

Entrevista: Se utilizó para conocer la opinión de desarrolladores de Sistemas de Información Geográfica de la facultad 6 para acumular una serie de elementos que tributan a la solución del problema.



Capítulo 1: Fundamentación Teórica

CAPÍTULO 1: Fundamentación Teórica

1.1 Introducción

El presente capítulo se compone de las principales definiciones y conceptos que serán de utilidad para lograr un mejor entendimiento de la investigación. Se hace alusión al surgimiento de los SIG así como de los móviles y a la evolución de estas tecnologías hasta la actualidad. Se analizan las funcionalidades de diferentes soluciones, principales tecnologías y herramientas a utilizar durante el desarrollo del sistema.

1.2 Conceptos asociados al dominio del problema

Sistema de Información Geográfica (SIG o GIS, en su acrónimo en inglés): es una integración organizada de *hardware*, *software*, datos geográficos y personal, diseñado 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 unas necesidades concretas de información. [3]

Dispositivos Inalámbricos: son dispositivos que no necesitan estar conectados físicamente mediante cables. Se corresponden con una tecnología mediante la cual todo dispositivo ofrece una conectividad vía radio (ondas). [4]

Dispositivos móviles: son aparatos de pequeño tamaño, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada, diseñados específicamente para una función, pero que pueden llevar a cabo otras funciones más generales. [5]

Teléfono móvil: dispositivo inalámbrico electrónico basado en la tecnología de ondas de radio, que tiene la misma funcionalidad que cualquier teléfono de línea fija. Su principal característica es su portabilidad, ya que la realización de llamadas no es dependiente de ningún terminal fijo y no requiere ningún tipo de cableado para llevar a cabo la conexión a la red telefónica. Aunque su principal función es la comunicación de voz, como el teléfono convencional, su rápido desarrollo ha incorporado funciones



Capítulo 1: Fundamentación Teórica

adicionales como mensajería instantánea (SMS), agenda, juegos, cámara fotográfica, acceso a Internet, reproducción de video e incluso *Global Positioning System* (GPS). [6]

1.3 Objeto de estudio

El objeto de estudio de esta investigación abarca los Sistemas de Información Geográfica para dispositivos móviles. La evolución de los SIG ha permitido tener en el área de aplicación mapas digitales en dispositivos móviles, estos mapas se utilizan para acceder a la información geográfica del servidor. Antes de abordar el objeto de estudio se hace necesario mencionar sus antecedentes, para obtener un conocimiento amplio sobre el tema.

1.3.1 Descripción general

1.3.1.1 Mapas

Un mapa es un documento que tiene que “hablar”, ha de ser capaz de transmitir al usuario la información que la persona que lo realiza ha determinado. Para ello, ha de estar sujeto a las normas y convenciones del lenguaje cartográfico, es lo que se denomina sintaxis cartográfica. [7]

Los mapas en sus inicios eran piezas exóticas y caras, diferentes de los modernos pues las convenciones de representación eran variables y dejaban espacio para los desbordes de la imaginación. Para su creación no se contaba con tecnologías que garantizaran una buena precisión de los mismos. Los navegantes los utilizaban junto con colecciones de notas personales sobre los trayectos. En el siglo XVIII los mapas comenzaron a asemejarse a los que se conocen actualmente, convirtiéndose en aliados incondicionales de los viajeros y herramientas de trabajo para la Geografía. [8]

Los mapas pueden estar en formato duro o digital. Los digitales ofrecen numerosas ventajas sobre los mapas en formato duro ya que son más rentables, la actualización o generación de nuevos mapas es en muchas ocasiones más fácil y la administración de la información es generalmente más sencilla.

La realidad a representar es generalmente volumétrica por lo que hay que cambiar de tres dimensiones a dos. El cambio se realiza mediante la descripción de la tercera dimensión como un atributo, por ejemplo una montaña tiene localización de coordenadas (x,y) que son representadas sobre el mapa y un atributo z, que es la altura.



Capítulo 1: Fundamentación Teórica

Un criterio a tener en cuenta es el paso de una superficie esférica a otra plana, para lo cual se utilizan las proyecciones. Estas se realizan con el objetivo de lograr la menor deformación posible utilizando una red de meridianos y paralelos. Las proyecciones se agrupan en tres sistemas básicos, en dependencia de la forma en que se realiza este proceso: las cilíndricas, cónicas y acimutales o polares. Las primeras utilizan un cilindro tangente a la superficie de la Tierra, en el segundo caso se utiliza un cono tangente o secante. En el último de los casos la proyección iría tangente a un solo punto.

Otro aspecto a tener en cuenta es la escala de un mapa. Representa la relación entre la distancia de dos puntos en un mapa y la distancia real. Las escalas pequeñas son aquellas que son mayores que 1: 200000, las escalas medias están entre la mencionada y 1: 25000 y de gran escala las menores que 1: 25000. La escala en un mapa será la que determine el nivel de información. [8]

Cartografía

La cartografía es la ciencia que se encarga del trazado y el estudio de mapas geográficos. [9] Es una ciencia que existe hace siglos, siempre ha sido de gran utilidad para la ubicación geográfica y espacial del ser humano permitiéndole realizar viajes. Tradicionalmente, el ser humano recurrió a las estrellas y a los elementos celestes para ubicarse, así como también a la matemática, a la geometría y a muchas otras disciplinas para realizar los mapas. [10]

La tecnología desempeña un papel importante en el desarrollo de la cartografía. La utilización de telescopios, satélites, computadoras, así como otros inventos permiten la creación de mapas cada vez más específicos y útiles. Dentro de las clasificaciones de la cartografía se encuentran la general y la temática. La primera se encarga de realizar mapas para el público en general, por ejemplo el mapamundi o el mapa de un país. En la cartografía temática los datos representados se especializan en un tema específico para un determinado público.

1.3.1.2 Sistemas de Información Geográfica

El desarrollo de los SIG se remonta a la década de los años sesenta cuando se creó el *Canadian Geographic Information System* (C.G.I.S.) que tenía como objetivo la gestión de los bosques y superficies marginales de Canadá. El progreso de los SIG está muy ligado al desarrollo de la tecnología informática. Los SIG fueron evolucionando paralelamente al progreso del *hardware* y del *software*.



Capítulo 1: Fundamentación Teórica

Entre las décadas de los años sesenta y setenta comienza el desarrollo de los SIG ráster, no muy precisos en la localización de los objetos por lo que se centran en las propiedades del espacio. Se componen de filas y columnas, donde cada una de las celdas formadas tiene igual tamaño y un único valor.

El modelo SIG vectorial se centra en la precisión de la localización de los objetos geográficos con sus límites bien definidos. La resolución es infinita, es decir, toda imagen vectorial se puede escalar ampliando o reduciendo sin que la visibilidad de la misma se vea afectada, ni en pantalla ni a la hora de imprimir. [11] Para modelar digitalmente las entidades del mundo real se utilizan tres objetos espaciales: el punto, la línea y el polígono.

En los años setenta se desarrolla el SIG vectorial ODYSSEY en el laboratorio de *Harvard*. Una parte considerable de los investigadores de estos laboratorios fueron los responsables del desarrollo y auge en los años ochenta de los SIG como productos industriales. A continuación se muestra una comparación de los modelos ráster y vectorial a partir de sus ventajas y desventajas. [12]

Comparación modelo vector vs ráster		
	Vector	Ráster
Ventajas	<ul style="list-style-type: none">• Buena similitud de las formas.• Cálculo más preciso de áreas, perímetros y longitudes.• Se pueden adaptar bajo bases de datos orientadas a objetos.	<ul style="list-style-type: none">• Estructura más simple.• Operaciones de superposición más sencillas.• La unidad espacial conserva la misma forma y tamaño.• Útil para análisis de grandes extensiones con baja precisión de propiedades espaciales.
Desventajas	<ul style="list-style-type: none">• Estructura más compleja.• Las superposiciones exigen más verificación de errores y pueden ser	<ul style="list-style-type: none">• Alto nivel de error en estimaciones de área, perímetro y longitud.• Gran espacio de almacenamiento a



Capítulo 1: Fundamentación Teórica

	más lentas.	medida que aumenta la resolución. • Transformación de coordenadas menos eficiente.
--	-------------	---

Tabla #1: Comparación modelo vector vs ráster.

Los SIG son herramientas que han alcanzado popularidad entre las personas. Su importancia se puede atribuir a la facilidad de resolver problemas, frecuentes como localizar una edificación o rastrear un vehículo. Esta tecnología es utilizada para el análisis de patrones, relaciones y tendencias en la información, contribuyendo así a la toma de decisiones.

Un SIG funciona como una base de datos con información geográfica que es asociada a objetos gráficos de un mapa digital mediante un identificador común. En la mayoría de los SIG, al seleccionar un objeto del mapa, se pueden conocer sus atributos y al realizar consultas introduciendo atributos específicos es posible conocer la ubicación de dicho objeto en el mapa digital. La información de los SIG es almacenada en capas temáticas que son independientes unas de otras. A partir de la superposición de las capas se obtiene un mapa temático como producto final que contiene la información deseada.

Componentes de un SIG

Para lograr el funcionamiento de un SIG se debe contar con ciertos componentes que son imprescindibles. Ellos son: [13]

Hardware: El *hardware* a emplear varía desde servidores para guardar la información así como computadoras conectadas a la red o individuales para capturar los datos. Para lograr un funcionamiento correcto dentro de una organización se debe utilizar *hardware* específico que solvete las necesidades de la aplicación (velocidad, costo, seguridad, etc.).

Software: Los principales componentes que los SIG proporcionan son los Sistemas de Manejo de Base de Datos (SMBD), la Interfaz Gráfica de Usuarios (GUI en sus siglas en inglés) para facilitar el acceso a las herramientas, instrumentos de captura y manejo de información geográfica y los de soporte de consultas, análisis y visualización de datos geográficos.



Capítulo 1: Fundamentación Teórica

Datos: Son el componente más importante de un SIG. Los datos geográficos y alfanuméricos pueden obtenerse de varias formas, una de ellas es a través de los propios recursos y otra mediante otros proveedores.

Personal: Se hace necesario contar con personal experto que garantice la actualización de la información, de esta forma, el SIG no pierde valor y no se cometen errores que pueden afectar a otros individuos.

Métodos: Para lograr la exitosa implementación de un SIG hay que tener en cuenta un buen diseño y reglas bien definidas.

Algunas de las ventajas que brindan los SIG son:

- La sustitución del mapa impreso en papel al utilizar los mapas digitales.
- La manipulación de elementos de un mapa, relacionando cada objeto con una información más amplia.
- El análisis matemático y salidas gráficas visualizando resultados.
- La generación de muchos mapas con los mismos datos.

El campo de aplicación de los SIG se hace cada vez mayor, llegándose a utilizar en muchas de las circunstancias de la vida cotidiana. La arqueología, la cartografía, la planificación urbana, la evaluación de las redes de servicio y el transporte son algunas de las ramas donde pueden ser útiles estos sistemas.

Formatos de archivos SIG

Algunos de los formatos de archivos que se utilizan para las imágenes vectoriales son:

- *Shapefile* [14]

El formato de representación vectorial *shapefile* fue desarrollado por el *Environmental Systems Research Institute* (ESRI). La localización de los objetos geográficos se almacena en los archivos *shape* con extensión *shp* y los atributos del objeto son almacenados en los archivos *dbf*.



Capítulo 1: Fundamentación Teórica

La cantidad de archivos varía en dependencia de la aplicación con que se generen los *shapefiles*, aunque siempre van a estar presentes los tres primeros archivos que son fundamentales:

- *Shape* (shp): Es el archivo principal y contiene la información geométrica de los elementos de cada capa. Puede contener puntos, líneas o polígonos. Cada vértice lleva implícitas sus coordenadas en un sistema de referencia concreto (que por lo general se especifica en el archivo *project*). Se componen de una cabecera con información general sobre el tipo de *shapefile* y un número variable de registros, que a su vez pueden estar compuestos por varias entidades geométricas independientes.
- *Shape Index* (shx): Consiste en un índice de las entidades geométricas que permite refinar las búsquedas dentro del archivo shp.
- dBase (dbf): Se trata de una tabla de datos en la que se registran los atributos de cada elemento. Es un formato que permite almacenar datos estructurados. En los *shapefiles*, las tablas dBase se emplean para asignar atributos numéricos, de texto o de fecha a los registros contenidos en el archivo principal.
- *Project* (prj): Es un archivo de texto que permite georeferenciar automáticamente los elementos geométricos contenidos en el archivo principal.

En la actualidad la mayoría de las aplicaciones SIG son compatibles con el formato *shapefile* por lo que se ha convertido en un estándar para el intercambio de datos geográficos. Este tipo de archivo permite situar cada elemento del terreno en la ubicación precisa en el SIG.

- *Keyhole Markup Language* (KML): [15]

El formato KML es utilizado para almacenar información geográfica en el formato de Lenguaje de Marcas Extensible (XML). Fue creado por una compañía de *software* llamada *Keyhole Inc.* Los archivos KML contienen una longitud y latitud, además de marcadores de lugar, polígono de formas, imágenes y textos.

Se pueden obtener también otras informaciones como el ángulo de la cámara y la altitud. Los objetos relacionados con texturas en tercera dimensión pueden ser almacenados en este tipo de



Capítulo 1: Fundamentación Teórica

formato. Los archivos KML le brindan la posibilidad a los desarrolladores de crear capas de información personalizadas. El formato KML es un estándar abierto aceptado por el *Open Geospatial Consortium* (OGC) utilizado en numerosos sitios *web* y aplicaciones.

- *Geography Markup Language* (GML): [16]

GML es un lenguaje basado en XML para el almacenamiento y transporte de la información geográfica a través de Internet. Fue desarrollado por OGC con el fin de definir la geometría y propiedades de los objetos. Es una aplicación para la transferencia de información cartográfica por la *web* que les permite a los usuarios decidir cómo se mostrarán los datos.

GML representa la información en formato de texto para la representación de geometrías y objetos geográficos, facilitando la revisión o modificación del mismo. Es independiente de plataformas y aplicaciones propietarias. Debido a que GML se desarrolló bajo una amplia gama de estándares públicos, los datos pueden ser visualizados, editados y transformados por numerosas herramientas gratuitas y privativas.

Entre los formatos de archivos para imágenes ráster se encuentra:

- *Enhanced Compression Wavelet* (ECW): [17]

Fue desarrollado por la empresa *Earth Resource Mapping*, actualmente es propiedad de *Leica Geosystems*. Es un formato de archivos para imágenes de tipo ráster. ECW es un estándar abierto, dispone de un *kit* de desarrollo de compresión y descompresión que los programadores pueden integrar en sus propias aplicaciones.

El formato es muy utilizado para el procesamiento de imágenes digitales debido a las ventajas que ofrece, además preserva la georeferenciación mediante un archivo de cabecera que tiene extensión *ers*. Se considera que este formato es de gran interés para las organizaciones que trabajan con grandes bases de datos de imágenes, las cuales pueden aprovechar sus beneficios.



Capítulo 1: Fundamentación Teórica

1.3.1.3 Telefonía Móvil

Con la creación de los primeros teléfonos, las personas quedaron impresionadas con la nueva forma de comunicación y las facilidades que les ofrecía. Los teléfonos han atravesado diferentes generaciones durante su evolución hasta llegar a los actuales dispositivos móviles.

La primera generación se caracterizó por ser analógica y estrictamente para la voz. La segunda generación fue la digital. La generación 2.5 fue una alternativa que muchos proveedores de servicio de telecomunicaciones utilizaron para introducirse en la próxima generación. La tercera generación tenía como característica la convergencia de la voz y datos con acceso inalámbrico a Internet, aplicaciones multimedia y altas transmisiones de datos.

Los dispositivos móviles pueden ser clasificados en: [6]

Dispositivo Móvil de Datos Limitados (*Limited Data Mobile Device*): es un teléfono móvil clásico que ofrece servicios de datos generalmente limitados a SMS y acceso *Wireless Application Protocol* (WAP).

Dispositivo Móvil de Datos Básicos (*Basic Data Mobile Device*): tiene el menú o navegación basada en íconos y ofrece acceso a correos, listas de direcciones, SMS y en algunos casos a navegadores *web* clásicos.

Dispositivo Móvil de Datos Mejorados (*Enhanced Data Mobile Device*): tiene la navegación tipo *stylus*¹, ofrece las mismas características que la clasificación anterior incluyendo además aplicaciones de *Microsoft Office Mobile* (*Word*, *Excel* y *Power Point*), versiones móviles de aplicaciones corporativas y además incluyen un Sistema Operativo (SO).

La forma en que interactúan los dispositivos móviles con los usuarios está dada por el SO que tenga instalado. Un SO es un *software* que permite la interacción entre el *hardware* y los usuarios. Facilita al usuario o al programador las herramientas e interfaces adecuadas para realizar sus tareas informáticas, abstrayéndole de los complicados procesos necesarios para llevarlas a cabo. [6]

¹ Stylus: Lápiz táctil.



Capítulo 1: Fundamentación Teórica

Entre los SO utilizados en dispositivos móviles se pueden encontrar *Windows Mobile* y *Android*. El primero utiliza algunas convenciones de la interfaz de usuario de *Windows*. Una ventaja que tiene sobre sus competidores es que las aplicaciones pueden ser desarrolladas utilizando los mismos lenguajes y entornos que se emplean con *Windows*. Algunas de las aplicaciones son versiones reducidas de *Outlook*, *Internet Explorer*, *Word*, *Excel* y *Windows Media Player*. *Android* es un SO desarrollado por Google basado en Linux y Java. El sistema busca un modelo estandarizado de programación que simplifique las labores de creación de aplicaciones móviles y normalice las herramientas en el campo de la telefonía móvil.

1.3.1.4 SIG para dispositivos móviles

Un SIG móvil permite al personal capturar, almacenar, actualizar, manipular, analizar y mostrar la información geográfica directamente desde el área de estudio. Tradicionalmente, los procesos de la recolección de datos y edición han sido lentos y propensos a errores. Las ediciones en el área de estudio se realizaban con bocetos y notas en los mapas de papel. Posteriormente las ediciones eran descifradas e introducidas manualmente en la base de datos SIG. Como resultado se obtienen datos que a menudo no son los más actualizados o precisos.

La evolución de los SIG para móviles ha permitido tener en el campo mapas digitales en los dispositivos móviles, proporcionando el acceso desde estos a la información geográfica de la empresa. Esto permite a las organizaciones añadir información en tiempo real a su base de datos y aplicaciones, acelerando el análisis, visualización y toma de decisiones, utilizando hasta la fecha los datos espaciales más precisos.

Los SIG móviles son utilizados para completar las siguientes tareas: [18]

- Asignación de campos: Crear, editar y utilizar mapas SIG en el área de estudio.
- Inventarios de activos: Crear y mantener un inventario de lugares y la información de atributos.
- Mantenimiento de activos: Actualizar la ubicación de activos, la condición y el mantenimiento programado.
- Inspecciones: Mantener los registros digitales y ubicaciones de los activos del área de estudio.



Capítulo 1: Fundamentación Teórica

- Reportes de incidentes: Documentar la ubicación y las circunstancias de los incidentes y acontecimientos de nuevas medidas o la presentación de informes.
- Análisis de SIG y toma de decisiones: Realizar la medición, geoprosesamiento y otros análisis de SIG en el área de estudio.

Componentes básicos de los SIG para dispositivos móviles

Los componentes básicos de los SIG para dispositivos móviles son los siguientes: [19]

Ciente: son los dispositivos clientes utilizados en el campo. El acceso y modificación de los datos de un SIG es posible de dos formas principales:

- Dispositivos *on-line*: Los usuarios pueden hacer peticiones de datos, modificar los mismos y devolver los resultados en tiempo real. Al utilizar este método se elimina la necesidad de almacenar grandes cantidades de datos en el dispositivo para luego poder actualizar la información.
- Dispositivos *off-line*: Los usuarios descargan los datos que necesitan en el dispositivo y utilizan una aplicación SIG instalada en el mismo. Permite visualizar y manipular los datos en el área de estudio. Este método elimina la necesidad de establecer conexiones desde el campo.

Localización: los SIG para dispositivos móviles son generalmente acoplados a Sistemas de Posicionamiento Global y comunicación inalámbrica para facilitar el intercambio entre el servidor y los dispositivos móviles.

La determinación de la ubicación de los dispositivos móviles se realiza en tiempo real. Una de las vías para determinarla es a través del GPS, este se coloca en el teléfono móvil proporcionando una buena precisión y flexibilidad de posicionamiento para la navegación, medición y la captura de datos.

Red: es la red de comunicación inalámbrica para el transporte de la información.

Servidores: este componente de soluciones móviles ofrece datos y aplicaciones para el usuario, por ejemplo, brinda la información de ubicación de los componentes y servicios de procesamiento de información. El tipo de información y servicios varía según el tipo de aplicación que se utiliza.



Capítulo 1: Fundamentación Teórica

Algunos de los beneficios que ofrecen los SIG para dispositivos móviles son: la captura y manejo de datos en el área de estudio, el aumento de la eficiencia de la recolección de datos, la sustitución de los mapas de papel en la recogida de datos, la edición de la información geográfica en el dispositivo, el acceso a la información en cualquier lugar y a cualquier hora y una mejor información sobre el punto para tomar decisiones.

1.3.2 Situación problemática

La Habana a pesar de ser la provincia más pequeña de Cuba, cuenta con la mayor cantidad de habitantes del país. En ella hay un amplio volumen de rutas, las cuales pueden ser definidas como trayectos habituales para ir de un sitio al otro. [20] Las paradas son lugares dentro del recorrido de los autobuses de transporte público en donde se detienen para permitir el ascenso y descenso de los pasajeros, suelen tener placas informativas con el recorrido de las líneas de los ómnibus que por allí circulan. [20]

Tanto las rutas de transporte como las paradas están sujetas a sufrir cambios cada cierto tiempo, por ejemplo el recorrido de una ruta puede ser modificado o eliminado y por tanto hay que crear nuevas paradas. La población de La Habana debe asimilar cierta cantidad de información sobre las rutas para su transporte diario, pero debido a lo planteado anteriormente es difícil obtener un conocimiento amplio sobre las mismas.

La creación de mapas en papel es una alternativa que soluciona parcialmente el problema anterior, pero no es suficiente porque el volumen de datos a visualizar puede ser reducido, es difícil detallar objetos, las modificaciones son en muchas ocasiones complejas y no pueden ser ampliados ni reducidos. Los datos en el mapa no pueden ser fácilmente extraídos y combinados con otros datos espaciales por lo que se debe crear uno nuevo, la publicación e impresión puede ser muy costosa y en ocasiones requiere de mucho tiempo.

La creación de un SIG para computadoras de escritorio sobre las rutas de transporte de La Habana es otra solución en la que se está trabajando en la UCI. A pesar de que permite almacenar una gran cantidad de información, la cual puede ser accedida, modificada o eliminada de forma sencilla, tiene una gran desventaja que es la poca portabilidad.



Capítulo 1: Fundamentación Teórica

1.4 Análisis de soluciones existentes

Actualmente existen múltiples SIG en el mundo que son utilizados para diversos fines, en dependencia del área de aplicación. Estos cuentan con múltiples funcionalidades utilizadas por los usuarios para trabajar con los mismos. Es necesario realizar un análisis de algunos SIG desarrollados a nivel nacional e internacional con el fin de identificar aquellas funcionalidades que pueden ser utilizadas en el sistema a desarrollar. Entre los SIG estudiados se encuentran:

SIGCAM: Es una aplicación SIG para asistir la gestión ambiental en el municipio de Camagüey. Entre las principales funciones definidas para el sistema se pueden encontrar la captura, almacenamiento, recuperación y actualización de las bases de datos geográficas. Está formado por cinco subsistemas en los que se integra una valiosa información gráfica y atributos imprescindibles para asistir la estrategia ambiental de la zona. Estos subsistemas son: Medio Ambiente Físico, Medio Ambiente Socioeconómico, Planificación Ambiental, Percepción y Educación Ambiental y Gestión e Inspección Ambiental. Cuenta con 24 bases de datos con 109 tablas, de ellas 81 gráficas y 28 de atributos. [21]

SIG-ESAC: Es el SIG para la gestión de la estadística de salud de Cuba. Es una personalización de ArcView 3.3. Utiliza para cartografiar los indicadores (demografía, mortalidad, morbilidad y recursos, servicios y cobertura) cuatro menús principales. Emplea ficheros vectoriales en formato *shapefile* para confeccionar la información espacial. Las extensiones añadidas al SIG-ESAC permiten emplear diferentes técnicas epidemiológicas como el análisis de clúster, métodos bayesianos, cálculos de tasas brutas y ajustadas e interpolaciones. [22]

SIGRutas: Es un SIG sobre las rutas de transporte de La Habana dirigido a la comunidad universitaria de la UCI. Entre las funcionalidades que tiene se pueden mencionar la navegación (acercar, alejar, mover, recentrar, etc.), cálculo de áreas, medir distancias, visualización de capas y personalización de rutas. Además, cuenta con un sistema de alerta y permite conocer la localización de direcciones, municipios, paradas y rutas. Brinda a los usuarios la posibilidad de conocer las paradas más cercanas a partir de diferentes criterios introducidos en el sistema. [23]

SITAR: Es el Sistema de Información Territorial de Aragón. Se compone de un conjunto de capas como la de hidrografía, transporte, topografía, calles, centros educativos, suelos, etc. Entre sus funcionalidades se puede identificar la búsqueda de objetos. Esta se realiza a partir de diferentes criterios que son mostrados



Capítulo 1: Fundamentación Teórica

una vez seleccionada la clasificación del objeto a localizar. Otras de las funcionalidades son: ampliar, reducir, mover, obtener información, medir distancias o áreas y dibujo de círculos. Permite además actualizar la visualización del mapa mediante la selección de las capas que el usuario desea que sean mostradas en el mapa. [24]

Al culminar el análisis de dichos SIG se identificó un conjunto de funcionalidades básicas que debe tener el sistema, por ejemplo *zoom* más, *zoom* menos, paneo, recentrar mapa, administración de capas, medir distancias, calcular áreas, mostrar coordenadas y visualizar información. Además, se pueden desarrollar otras funcionalidades que se relacionen específicamente con el objetivo de la investigación como la localización de paradas y rutas. Estas son las funcionalidades propuestas. En el próximo capítulo se identificarán otras que formarán parte del sistema.

1.5 Herramientas y tecnologías a utilizar

Durante la construcción de un *software* se selecciona un conjunto de herramientas y tecnologías a utilizar que son de vital importancia para desarrollar un producto en el tiempo estimado y con la calidad necesaria. Un factor que se debe tener en cuenta para la selección de las herramientas es que sean libres. A continuación se realiza un análisis de las herramientas y tecnologías más utilizadas con el fin de identificar aquellas que ofrecen mayores beneficios durante el desarrollo de la solución.

1.5.1 Plataformas para desarrollar SIG para dispositivos móviles

Existen múltiples plataformas en la actualidad, utilizadas para desarrollar SIG para dispositivos móviles. Algunas son privativas y aunque ofrecen múltiples ventajas que pueden ser aprovechadas por los usuarios, su principal desventaja radica en la necesidad de comprar su licencia para poder utilizarlas. Entre las más utilizadas se pueden destacar:

ArcGIS Mobile: Pertenece a la familia de productos de *software* ArcGIS. Se integra con el ArcGIS Server para el manejo de datos, mapas, aplicaciones y proyectos. La actualización en el servidor de los datos es casi en tiempo real, pues permite trabajar en un entorno conectado o desconectado gracias al caché local de datos con que cuenta [25]. Las herramientas de ArcGIS Mobile no son difíciles de usar y habilitan al personal de campo, no se necesita tener experiencia en SIG para realizar tareas de mapeo, consultas



Capítulo 1: Fundamentación Teórica

espaciales, integración con GPS, edición SIG y acceso inalámbrico a los servicios *web* de ArcGIS Server. [26]

ArcPad: Permite al personal que trabaja sobre el terreno la captura, edición, análisis y despliegue de información geográfica. Puede utilizarse en proyectos existentes tanto de datos vectoriales como ráster o comenzar uno nuevo desde cero. Con ArcPad la toma de datos en el campo puede ser rápida, sencilla y precisa, además ofrece validación y disponibilidad inmediata de la información recogida. [27]

Por otra parte, se pueden encontrar una serie de plataformas que erradican el problema de la licencia al ser libres. Representan una solución factible que pueden ser utilizadas para la construcción de SIG en países del tercer mundo como Cuba. Entre estas se encuentran:

GvSIG Mini: Es una herramienta que permite ser descargada y usada libremente. Consta de dos productos, para dispositivos móviles que permitan aplicaciones Java y para aquellos que tengan Android como sistema operativo. Para la visualización de los mapas utiliza cartografía digital procedente de servicios remotos. [28]

GvSIG Mobile: Es una herramienta libre y de código abierto, desarrollada por la empresa *Prodevelop* e Instituto de Robótica de la Universidad de Valencia. Está dirigida a dispositivos móviles que cuenten con capacidad de programación como teléfonos móviles, Asistentes Digital Personal (PDA por sus siglas en inglés), *tablet PC*, *Internet tablet* y *Ultra Mobile PC* (UMPC). Es tanto un SIG como un Cliente de Infraestructura de Datos para los dispositivos móviles.

Algunas de las características de *GvSIG Mobile* que se pueden destacar son: [29]

- Integra tanto datos locales como remotos a través de estándares OGC.
- Está diseñado para ser fácilmente extensible, permitiendo una mejora continua de la aplicación, así como su uso para desarrollar soluciones a la medida.
- Es *software* libre, con licencia GNU/GPL, lo que permite su libre uso, distribución, estudio y mejora.
- Está disponible en diversos idiomas: español, inglés, francés, alemán e italiano.



Capítulo 1: Fundamentación Teórica

- Está desarrollado con Java y está disponible para plataformas que soporten *Java Micro Edition* CDC 1.1. Actualmente funciona en las plataformas *Windows Mobile* 5.0 y 6.0.

Entre las principales funcionalidades que presenta GvSIG *Mobile* se encuentran el acceso a formatos vectoriales como SHP, GML, KML y *GPS Exchange Format* (GPX). Permite utilizar formatos ráster como ECW, *Joint Photographic Experts Group* (JPEG), *Portable Network Graphics* (PNG) y *Graphics Interchange Format* (GIF). Además se puede acceder a servicios remotos (*Web Map Service* o WMS). Otras de las funcionalidades que se pueden encontrar son la navegación, consulta, selección, búsqueda, edición gráfica mediante la creación de puntos, líneas y polígonos y la edición alfanumérica utilizando formularios personalizados.

En los dispositivos móviles es de vital importancia el diseño de la interfaz gráfica. Para ello hay que tener en cuenta una serie de elementos como el tamaño de la pantalla, el uso intensivo del teclado o pantalla táctil y las limitaciones de memoria. La interfaz de usuario de GvSIG *Mobile* utiliza un sistema de barras de herramientas semitransparentes que permite ocultar los botones dejando más espacio para la visualización del mapa.

Teniendo en cuenta que entre las plataformas estudiadas se encuentran algunas que necesitan utilizar servicios *web* se hace necesario identificar la más conveniente a utilizar. Para solucionar el problema se necesita hacer uso del sistema desde cualquier sitio. Las plataformas que utilizan servicios *web* no son factibles en este caso, pues una vez que el usuario se encuentre en un lugar donde no tenga conexión con el servidor de mapas la aplicación queda inhabilitada. Por otra parte si los mapas estuviesen almacenados localmente se podría acceder a los mismos en cualquier territorio. Otra de las ventajas que ofrecen las aplicaciones de escritorio es su corto tiempo de respuesta. Por lo anteriormente expuesto se selecciona como plataforma GvSIG *Mobile*.

GvSIG *Mobile* es un *software* muy utilizado en el mundo debido a las múltiples facilidades que le ofrece tanto a los usuarios, como a los desarrolladores. Permite a los programadores construir aplicaciones basadas en *software* libre, por lo que pueden ser utilizadas con múltiples fines como la venta, siendo de gran utilidad para el proyecto SIG-Móviles. Por las características de dicho *software* se selecciona entre las herramientas que serán utilizadas para construir el sistema.



Capítulo 1: Fundamentación Teórica

1.5.2 Entorno de Desarrollo Integrado

Un Entorno de Desarrollo Integrado (IDE) es un conjunto de herramientas que ayudan al desarrollo de aplicaciones. La mayoría de los IDE tienen herramientas que permiten escribir y editar el código fuente, consultar los errores a medida que se escribe, compilar el código y examinar las estructuras de clase [30]. Algunos de los IDEs que han ganado popularidad son NetBeans, Eclipse y Qt. El IDE a utilizar es el Eclipse Europa debido a que GvSIG *Mobile* fue desarrollado sobre este y la migración hacia otro puede provocar problemas de compatibilidad, además la estructura de las carpetas en eclipse es diferente a la que tienen los demás IDEs.

Eclipse es independiente de la plataforma de desarrollo, de código abierto, es multilenguaje a través de paquetes de lenguajes, algunos de los soportados son C, C++, Cobol, Python, Perl y PHP por medio de módulos (*plugins* en inglés) y cuenta con una gran comunidad de usuarios que desarrollan su código ofreciendo así nuevas funcionalidades. Eclipse utiliza módulos para proporcionar diferentes funcionalidades, la ventaja que ofrece es que no están incluidos todos, independientemente de que los usuarios las necesiten o no como ocurre con otros IDEs.

1.5.3 Lenguaje de programación

Un lenguaje de programación es aquel elemento dentro de la informática que permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que pueda comunicarse con los dispositivos *hardware* y *software* existentes [31]. Para implementar el sistema se seleccionó el lenguaje de programación Java porque el código de GvSIG *Mobile* fue escrito en este. Java cuenta con una amplia biblioteca de clases predefinidas, las aplicaciones pueden ser ejecutadas en cualquier plataforma que posea su máquina virtual instalada sin necesidad de volver a compilar. Es un lenguaje orientado a objetos y tiene una gestión avanzada de memoria a partir del uso de un recolector de basura. Tiene además una gestión avanzada de errores, pues en tiempo de compilación como de ejecución es capaz de detectarlos y posee una gran variedad de documentación gratuita en la red.

Algunas de las características por las que Java ha ganado popularidad hoy en día son:

- Simple: Fue diseñado para que fuera fácil de leer y escribir.



Capítulo 1: Fundamentación Teórica

- Orientado a objetos: Se basa en varias técnicas, incluyendo herencia, abstracción, polimorfismo y encapsulamiento.
- Robusto: Realiza verificación del código de *byte*, tiene una gestión de excepciones y errores y comprueba punteros y límites de vectores.
- Distribuido: Contiene una serie de clases que son utilizadas en entornos de red que facilitan la creación de aplicaciones distribuidas.
- Interpretado y compilado a la vez: Es compilado porque su código fuente puede ser transformado en código máquina (*bytecodes*) que puede ser ejecutado sin necesitar otro programa. Es interpretado porque en ocasiones pequeños programas son cargados en páginas *web* y para que puedan ser ejecutados hay que utilizar navegadores *web*.
- Multitarea: Al poder sincronizar múltiples hilos de ejecución un usuario puede visualizar una animación, mientras que otro puede realizar cálculos.
- Seguro: Implementa barreras de seguridad en el lenguaje y en el sistema de ejecución en tiempo real.

1.5.4 Metodología de desarrollo de software

En los últimos años ha aumentado la automatización de los procesos empresariales, lo cual significa que sean cada vez más complejas. Se debe ser cauteloso desde un principio en su planificación para evitar gastos que sean más elevados que la solución. Las metodologías de desarrollo de *software* son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de *software* [32].

A partir de lo anteriormente definido se puede decir que las metodologías de desarrollo de *software* no son más que reglas que definen el proceder de forma lógica y detallada, con el objetivo de facilitar el proceso de construcción de un *software*. Guían el proceso de actividades a realizar y al personal implicado en la misma. Identifican también las entradas y salidas que se necesitan y que se obtienen en el desarrollo de un proceso. Las metodologías pueden ser clasificadas en dos grupos: las ágiles y las tradicionales. Estas



Capítulo 1: Fundamentación Teórica

tienen diferencias significativas en la manera en que se desarrolla el *software* y en la organización del equipo de trabajo.

Metodologías ágiles

No generan una amplia documentación, el equipo de trabajo es integrado por pocas personas, existen pocos roles y la planificación es flexible pues el proyecto puede sufrir cambios durante su desarrollo [33]. Cada una de las metodologías ágiles tiene características propias, por ejemplo:

Extreme Programming (XP): Método ágil basado en cuatro principios: simplicidad, comunicación, retroalimentación y valor [34]. Se caracteriza por la programación en parejas donde uno de los programadores escribe códigos y el otro lo prueba. El desarrollo es guiado por pruebas y tiene diseño simple. Está constituido por un grupo pequeño y muy integrado de trabajo. Es ideal para proyectos donde los requisitos sean imprecisos y estén sujetos a posibles cambios. Hace énfasis en el desarrollo del *software* más que en obtener una buena documentación.

Crystal Methodologies: Constituida por un conjunto de metodologías de desarrollo de *software*. Cuenta con pocos roles a asumir por el equipo de desarrollo. Atribuye cierta importancia a los individuos. Aconseja que el tamaño del equipo de desarrollo sea reducido. Cada equipo utiliza políticas diferentes dependiendo del tamaño del equipo, para lo cual se establece una clasificación por colores, por ejemplo *Crystal Clear* (3 a 8 miembros) y *Crystal Orange* (25 a 50 miembros). [35]

SCRUM: Es ideal para proyectos con un rápido cambio de requisitos. El equipo de desarrollo debe estar formado por una cantidad aproximada de siete miembros. Se organiza en iteraciones denominadas *Sprints*, que normalmente tienen entre dos y cuatro semanas de duración. Son reconocidos solamente tres roles: dueño del producto, *scrum* máster y equipo. [36]

Metodologías Tradicionales

En estas metodologías se hace una planificación de todo el trabajo en etapas tempranas del desarrollo del sistema. Cuentan con una amplia variedad de roles y se genera una amplia documentación. Tienen un contrato prefijado con los requisitos que desea el cliente pues las metodologías tradicionales no se adaptan adecuadamente a los cambios como ocurre con las ágiles. Entre estas metodologías se encuentran:



Capítulo 1: Fundamentación Teórica

Iconix: Esta metodología se caracteriza por ser iterativa e incremental, es decir, varias iteraciones ocurren entre el desarrollo del modelo del dominio y la identificación de los casos de uso. La trazabilidad es otra de sus características pues cada paso está referenciado por algún requisito lo que permite seguir la relación entre los diferentes artefactos de *software* producidos. Por último, la metodología ofrece un uso dinámico del Lenguaje Unificado de Modelado (UML) porque utiliza algunos diagramas de este sin exigir la utilización de todos, como en el caso de *Rational Unified Procces* (RUP). [37]

MSF (*Microsoft Solution Framework*): Metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. Se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. Es adaptable, escalable y flexible. Está compuesta por varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto. [38]

RUP: Constituye una de las metodologías estándares más utilizadas para el análisis, implementación y documentación de sistemas orientados a objetos. Define además una serie de actividades que tienen como objetivo convertir las necesidades de los clientes en un *software*. Su ciclo de vida está caracterizado fundamentalmente por tres elementos:

- Dirigido por casos de uso: Los casos de uso (CU) son un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Representan los requisitos funcionales del sistema. Guían los procesos de diseño, implementación y pruebas del sistema. Todos los modelos deben estar sincronizados con el modelo de casos de uso. [39]
- Centrado en la arquitectura: La arquitectura permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo. Indica cómo tiene que ser construido el sistema y ayuda a determinar en qué orden [39]. Describe los elementos que son necesarios para la construcción de la aplicación. Al final de la fase de elaboración se obtiene una aproximación de la arquitectura donde se seleccionan una serie de casos de uso arquitectónicamente significativos.



Capítulo 1: Fundamentación Teórica

- Iterativo e incremental: RUP propone la estrategia de desarrollar cada fase en varias iteraciones. Cada iteración incluirá actividades de cada uno de los flujos de trabajo. En la medida que se realizan iteraciones se obtendrá un crecimiento del producto.

Selección de la metodología

Para seleccionar correctamente la metodología de desarrollo de *software* a utilizar hay que tener en cuenta las características del proyecto y del equipo de desarrollo. No es recomendable utilizar metodologías ágiles porque el proyecto necesita tener documentación detallada, así como los diferentes diagramas que se generan durante el desarrollo del producto para su posterior mantenimiento y evolución. El proyecto SIG-Móviles no tiene un cliente, lo cual influye negativamente para aplicar esta metodología pues la comunicación con el mismo desempeña un papel fundamental.

Las metodologías tradicionales erradican los problemas anteriormente expuestos. Algunas de las desventajas de MSF es que son un modelo de *Microsoft*, lo que implica que se tiene que utilizar herramientas de este tipo limitando de esta forma a los clientes. Utiliza demasiada documentación en sus fases, lo que puede provocar atrasos en el desarrollo de la aplicación y el análisis de riesgo suele llevar mucho tiempo, lo cual frena el avance del proyecto [40]. Entre las desventajas que tiene la utilización de Iconix se destacan: no puede ser usado en proyectos grandes y la información de los requisitos, el diseño y las estimaciones ha de ser rápida y precisa. [41]

RUP es la metodología seleccionada para la construcción del sistema. Una de las ventajas que proporciona es que al ser una metodología estudiada en la UCI, facilitará el manejo de la misma para desarrollar exitosamente el sistema. La manera de desarrollar un *software* es única porque cada uno tiene prioridades, requerimientos y utiliza tecnologías diferentes. Al ser SIG-Móviles un proyecto joven los implicados tal vez no cuentan con vastos conocimientos y experiencia para desarrollar productos de este tipo. La calidad y la entrega en el tiempo estimado son elementos a tener en cuenta durante el desarrollo, por lo que haciendo uso de las guías enfocadas en cada uno de los miembros del equipo de desarrollo se facilita su cumplimiento.

Entre los miembros del proyecto la socialización y entendimiento de la información es muy necesaria para lo cual provee a cada uno, documentos, herramientas y modelos basados en estándares. En algunas



Capítulo 1: Fundamentación Teórica

ocasiones se pueden utilizar herramientas y tecnologías desconocidas para los programadores provocando de esta forma que la probabilidad de riesgos en la aplicación sea mayor. El desarrollo iterativo puede reducir este problema en etapas tempranas.

1.5.5 Lenguaje de modelado

El lenguaje de modelado es la notación generalmente gráfica que usan los métodos para expresar un diseño. El proceso indica los pasos que se deben seguir para llegar a un diseño. Debido a que fue seleccionado RUP como metodología y a su vez esta propone el uso de UML, se decide utilizar durante el desarrollo del producto este lenguaje de modelado.

Los modelos UML se componen de tres clases de bloques de construcción. Estas son: [42]

- Elementos: Abstracciones de objetos reales o ficticias (objetos, acciones, etc.).
- Relaciones: Relacionan los elementos entre sí.
- Diagramas: Colecciones de elementos con sus relaciones.

UML está constituido por un conjunto de diagramas, proporciona un estándar que permite al analista de sistemas generar un prototipo de proyecto de varias facetas, que sean comprensibles para los clientes, los desarrolladores y todos aquellos que estén involucrados en el proceso de construcción. Es necesario contar con todos esos diagramas dado que cada uno se dirige a cada tipo de persona implicada en el sistema. Un modelo UML indica qué es lo que supuestamente hará el sistema, mas no cómo lo hará.

Se utilizará el UML porque este es un estándar internacional y se hace necesaria la socialización de la información entre los clientes, desarrolladores y todos los involucrados en el proceso de desarrollo a partir del entendimiento de los diagramas. Para evitar inconformidades entre desarrolladores y clientes es necesario que estos tengan una buena comunicación, lo cual hace posible la utilización del UML. Los elementos gráficos que se generan pueden ser utilizados como documentación del sistema para futuras revisiones. Antes de comenzar a programar es importante definir cuáles serán las características del sistema para lo cual es posible apoyarse en el UML.



Capítulo 1: Fundamentación Teórica

1.5.6 Herramientas CASE

Las herramientas CASE son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de *software* reduciendo el coste de las mismas en términos de tiempo y de dinero [43]. Proporcionan múltiples facilidades a la hora de documentar el desarrollo del *software*.

Rational Rose Enterprise Edition: Es comercializada por los desarrolladores de UML y líder en el mundo de modelación visual de sistemas, permite especificar, analizar y diseñar el sistema antes de codificarlo [44]. A pesar de ello, posee algunas limitaciones que la hacen débil como la integración solo con herramientas que estén en el mismo grupo de *software* propietario.

Visual Paradigm para UML: La principal ventaja que proporciona es que tiene licencia gratuita y comercial, lo que hace más asequible su utilización para desarrollar aplicaciones bajo herramientas libres. Soporta el ciclo de vida completo del desarrollo de un *software*: análisis y diseño orientado a objetos, construcción, pruebas y despliegue [45]. Ofrece un número considerable de estereotipos que son utilizados en diferentes diagramas que se pueden generar, posibilitando así un mayor entendimiento de los mismos. Por estas razones se decide utilizar la herramienta en su versión 3.4 y a su vez por ser libre.

1.6 Conclusiones

La toma de decisiones relacionadas con el transporte público por parte de los ciudadanos es en muchas ocasiones difícil y engorrosa, se puede concluir que es necesario realizar un SIG para dispositivos móviles relacionado con las rutas de transporte para resolver dicho problema. Al definir los conceptos asociados al problema se sentaron las bases teóricas para lograr un mejor entendimiento del mismo. El análisis de las soluciones existentes incidió directamente en la definición de las funcionalidades a implementar. Finalmente, se concluyó que el sistema será desarrollado utilizando la plataforma GvSIG *Mobile*, como IDE de desarrollo el Eclipse Europa, haciendo uso del lenguaje de programación Java. La metodología de desarrollo de *software* que se utilizará es RUP, con el lenguaje de modelado UML y la herramienta CASE Visual Paradigm.



Capítulo 2: Presentación de la solución propuesta

CAPÍTULO 2: Presentación de la solución propuesta

2.1. Introducción

En el presente capítulo se hace un análisis del entorno donde ha de trabajar el sistema capturando los principales objetos y conceptos mediante el Modelo de Dominio. Se realiza un levantamiento de requisitos funcionales y no funcionales identificando las necesidades y características que ha de tener el sistema. Se analizan los casos de uso del sistema mediante sus descripciones textuales y al actor que los inicializa.

2.2. Entorno donde trabajará el sistema

Los usuarios de dispositivos móviles necesitan un sistema que les permita el acceso a la información georeferenciada sobre las rutas de transporte de La Habana. Partiendo de que los procesos del negocio tienen flujos de información difusos y no están bien definidos se decide realizar una modelación del dominio para tener mayor conocimiento de los mismos. En este se representarán los conceptos más importantes relacionados con el problema para obtener una mejor comprensión de la estructura, dinámica de la organización y problemas actuales en aras de identificar las mejoras potenciales.

2.2.1. Conceptos y eventos principales del entorno

Un **mapa** se compone de la **escala** representativa y la **leyenda** utilizada para lograr que el **cliente** al manipularlo tenga un mejor entendimiento y pueda de esta forma trabajar fácilmente con el mismo. Además, tiene **coordenadas geográficas** compuestas por la latitud y la longitud, así como **información socioeconómica** la cual contiene datos referentes a las **rutas** de transporte de La Habana con cada una de las **paradas** que se distribuyen por todo su territorio.



Capítulo 2: Presentación de la solución propuesta

2.2.2. Diagrama de clases del Modelo de Dominio

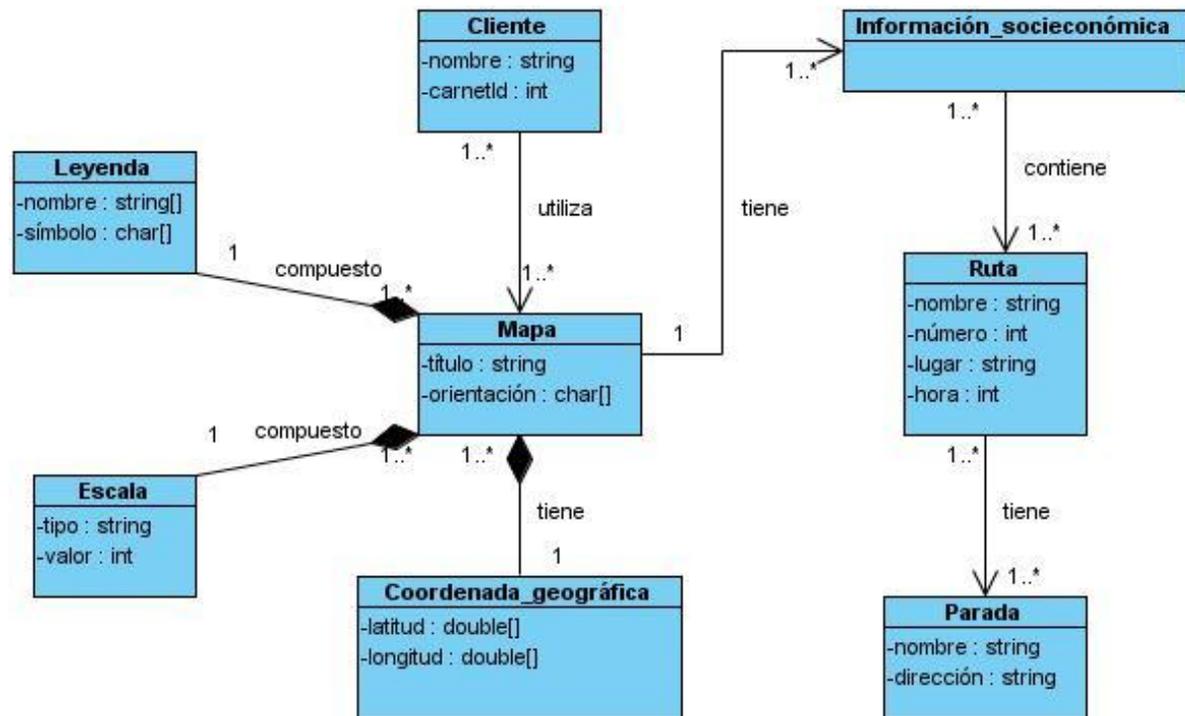


Figura #1: Modelo del Dominio.

2.2.3. Glosario de Términos del Dominio

Cliente: Persona que necesita trabajar o consultar algún tipo de información incluida en un mapa.

Mapa: Es una representación gráfica y métrica de una porción de territorio sobre una superficie bidimensional, generalmente plana, pero que puede ser también esférica como ocurre en el globo terráqueo. El hecho de que el mapa tenga propiedades métricas significa que ha de ser posible medir distancias sobre él y obtener un resultado aproximadamente exacto.

Leyenda: Explicación de los símbolos, colores, tramas y sombreados empleados en un mapa. Los símbolos empleados en los mapas pueden llegar a contener un gran volumen de información, que por su facilidad de lectura permiten una rápida interpretación.



Capítulo 2: Presentación de la solución propuesta

Escala: Relación entre la distancia que separa dos puntos en un mapa y la distancia real de esos dos puntos en la superficie terrestre. Pueden ser expresadas en tres formas distintas, de forma gráfica, con expresiones o palabras y con porciones o fracciones.

Coordenadas geográficas: Líneas imaginarias trazadas sobre la Tierra de forma tal que hacen una cuadrícula utilizada para localizar y medir elementos geográficos. La localización de los elementos es importante para representar el mundo real, para ello se utilizan dos tipos de líneas, las que miden la latitud (paralelos) y las que miden la longitud (meridianos).

Información socioeconómica: Es un conjunto organizado de datos procesados referentes al aspecto social y económico de cualquier lugar de interés.

Ruta: Trayecto habitual para ir de un sitio a otro.

Parada: Las paradas son lugares dentro del recorrido de los autobuses de transporte público en donde estos se detienen para permitir el ascenso y descenso de los pasajeros. Suelen tener placas informativas con el recorrido de las líneas de autobuses que por allí circulan.

2.3. Requisitos Funcionales

Los requisitos se pueden clasificar en funcionales y no funcionales. Deben ser especificados por escrito, posibles de probar o verificar y descritos como una característica del sistema a entregar. Partiendo de las principales necesidades que debe satisfacer el sistema se definen un conjunto de capacidades o condiciones que el mismo ha de cumplir. Estas son:

RF 1 Realizar Navegación: El sistema debe permitir que el usuario pueda navegar por el mapa transformando la visualización del mismo para obtener una mejor representación de los objetos.

RF 1.1 Realizar zoom rectángulo: El sistema debe permitir que el usuario pueda seleccionar un área del mapa para aumentar nivel de detalle de la misma.

RF 1.2 Realizar zoom más: El sistema debe permitir que el usuario pueda aumentar el nivel de detalle del mapa mostrado en pantalla para tener una mejor visualización.



Capítulo 2: Presentación de la solución propuesta

RF 1.3 Realizar *zoom* menos: El sistema debe permitir que el usuario pueda reducir el nivel de detalle del mapa mostrado en pantalla para tener una mejor visualización.

RF 1.4 Realizar *zoom* anterior: El sistema debe permitir que el usuario pueda regresar al estado de visualización del mapa en que se encontraba anteriormente.

RF 1.5 Realizar *zoom* siguiente: El sistema debe permitirle al usuario cambiar al estado de visualización del mapa por el que tenía antes de regresar a un estado anterior.

RF 1.6 Realizar paneo manual: El sistema debe permitir que el usuario pueda mover el mapa para cambiar el área visualizada a partir de la selección y arrastre del mismo en la pantalla.

RF 1.7 Recentrar mapa: El sistema debe permitir que el usuario pueda seleccionar un punto en el mapa para que sea visualizado en el centro del mismo.

RF 1.8 Realizar *zoom* extensión: El sistema debe permitir que el usuario pueda visualizar el mapa completo.

RF 2 Localizar Parada: El sistema debe permitir que el usuario pueda conocer la ubicación en el mapa de una parada seleccionada.

RF 3 Localizar Ruta: El sistema debe permitir que el usuario pueda conocer la ubicación en el mapa de una ruta seleccionada.

RF 4 Visualizar Información: El sistema debe permitir que el usuario pueda consultar la información asociada a la última selección de rutas de ómnibus realizada en el mapa.

RF 5 Encontrar Ruta: El sistema debe permitir que el usuario pueda conocer si existe al menos una ruta de ómnibus que pase por dos paradas seleccionadas y que sean ubicadas en el mapa.

RF 6 Parada de Intercambio: El sistema debe permitir que el usuario pueda conocer aquellas paradas que comparten dos rutas de ómnibus seleccionadas y que sean ubicadas en el mapa.

RF 7 Gestionar proyecto: El sistema debe permitir que el usuario pueda realizar acciones en el proyecto cargado en el sistema.



Capítulo 2: Presentación de la solución propuesta

RF 7.1 Crear nuevo proyecto: El sistema debe permitir que el usuario pueda crear un proyecto nuevo.

RF 7.2 Abrir proyecto: El sistema debe permitir que el usuario pueda abrir un proyecto existente.

RF 7.3 Guardar proyecto: El sistema debe permitir que el usuario pueda guardar los cambios realizados en el proyecto que está cargado.

RF 7.4 Guardar proyecto como: El sistema debe permitir que el usuario pueda guardar el proyecto cargado en una ruta especificada.

RF 7.5 Configurar aplicación: El sistema debe permitir que el usuario pueda cambiar la configuración que tiene la aplicación.

RF 8 Realizar consulta por selección: El sistema debe permitir que el usuario pueda realizar consultas en el mapa a partir de la selección de puntos.

RF 8.1 Mostrar coordenadas: El sistema debe permitir que el usuario pueda conocer las coordenadas que tiene un punto seleccionado en el mapa.

RF 8.2 Medir distancia: El sistema debe permitir que el usuario pueda conocer la distancia que tienen las líneas formadas a partir de la selección de puntos en el mapa.

RF 8.3 Medir área: El sistema debe permitir que el usuario pueda conocer el área que tiene la figura formada a partir de la selección de puntos en el mapa.

RF 9 Administrar capa: El sistema debe permitir que el usuario pueda realizar acciones con las capas cargadas en el sistema.

RF 9.1 Añadir capa vectorial: El sistema debe permitir que el usuario pueda añadir capas con formatos vectoriales al sistema.

RF 9.2 Añadir capa ráster: El sistema debe permitir que el usuario pueda añadir capas con formatos ráster al sistema.

RF 9.3 Mostrar capa: El sistema debe permitir que el usuario pueda visualizar las capas cargadas en el sistema.



Capítulo 2: Presentación de la solución propuesta

RF 9.4 Mostrar información de la capa: El sistema debe permitir que el usuario pueda conocer la información que tiene una capa seleccionada.

RF 9.5 Editar capa: El sistema debe permitir que el usuario pueda seleccionar una capa para editar su información.

RF 9.6 Eliminar capa: El sistema debe permitir que el usuario pueda eliminar una capa cargada en el sistema.

RF 9.7 Subir capa: El sistema debe permitir que el usuario pueda subir la capa de nivel para una mejor visualización del mapa.

RF 9.8 Bajar capa: El sistema debe permitir que el usuario pueda bajar la capa de nivel para una mejor visualización del mapa.

RF 9.9 Realizar *zoom* a capa: El sistema debe permitir que el usuario pueda visualizar una capa seleccionada con sus dimensiones originales.

RF 9.10 Editar simbología de capa: El sistema debe permitir que el usuario pueda realizar cambios en la simbología de una capa seleccionada.

RF 10 Obtener información: El sistema debe permitir que el usuario pueda conocer la información asociada de un objeto seleccionado en el mapa.

RF 11 Limpiar selección: El sistema debe permitir que el usuario pueda eliminar las selecciones realizadas en el mapa.

2.4. Requisitos No Funcionales

Los requisitos no funcionales son aquellas propiedades o cualidades que el producto debe tener. Se pueden clasificar en diferentes categorías según su objetivo, por ejemplo, las restricciones de diseño, usabilidad, soporte y seguridad. A continuación se describen los requisitos no funcionales identificados.



Capítulo 2: Presentación de la solución propuesta

Requisitos de *hardware*

- El dispositivo debe ser PDA, *Smartphone*² o *Pocket PC* para que funcione el sistema correctamente.
- La velocidad de procesamiento mínima debe ser de 320 MHz, pero se recomienda 600 MHz. [46]
- Debe tener una RAM mínima de 64 MB, pero se recomienda una de 96 MB. [46]

Requisitos de *software*

- El dispositivo debe tener la Máquina Virtual de Java Micro *Edition CDC*³ 1.1.
- El sistema operativo que debe tener el dispositivo es *Windows Mobile*.

Requisitos de interfaz

- Diseñado de forma tal que el usuario solo tenga que introducir información indispensable.
- Los textos y mensajes de la aplicación estarán en idioma español.

Requisitos de usabilidad

- El sistema podrá ser utilizado por personas con conocimientos básicos sobre el manejo de móviles.

Requisitos de rendimiento

- El tiempo de respuesta va a depender de la cantidad de información que tiene que procesar la aplicación y serán directamente proporcionales.

Requisitos de soporte

- El sistema recibirá mantenimiento en el período de tiempo determinado por el equipo de desarrollo y los clientes.

² Smartphone: Teléfono inteligente.

³ CDC: Configuración de Dispositivos Conectados.



Capítulo 2: Presentación de la solución propuesta

Requisitos de disponibilidad

- La información debe estar disponible en todo momento, a no ser que ocurran fallos en el dispositivo como en la batería u otros.

2.5. Descripción del Sistema Propuesto

2.5.1. Descripción de los actores

Un actor no es parte del sistema en desarrollo, es un agente externo que interactúa con el mismo en pos de obtener un resultado esperado. A continuación se define el actor que interactuará con el sistema y su descripción.

Actor	Descripción
Usuario	Toda persona que tiene necesidad de conocer información geográfica sobre las rutas de transporte de La Habana mediante dispositivos móviles.

Tabla #2: Descripción del actor del sistema.

2.5.2. Casos de Uso del Sistema

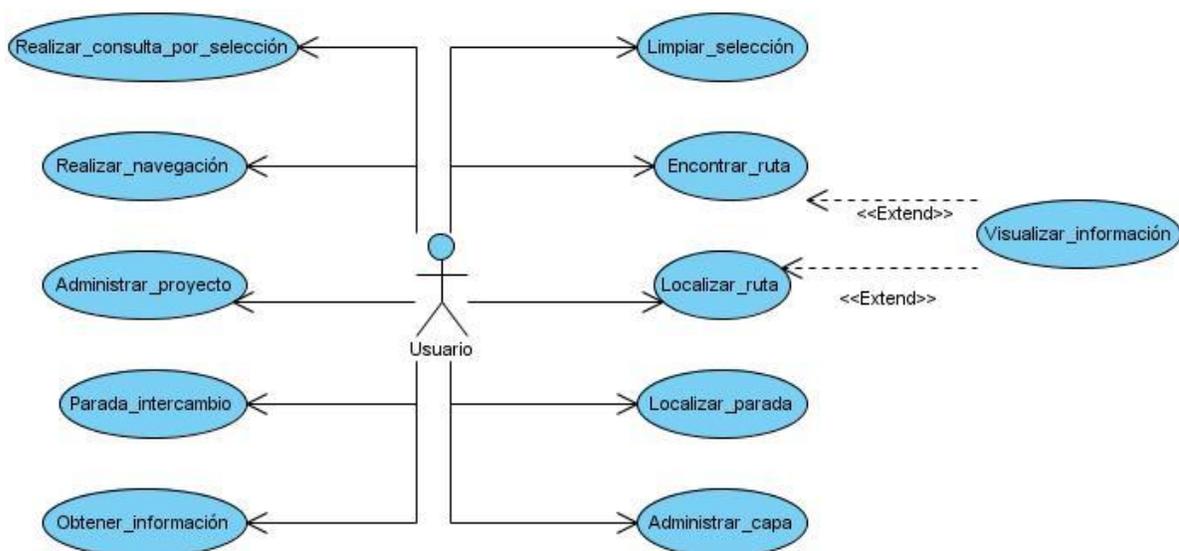


Figura #2: Diagrama de Casos de Uso del Sistema.



Capítulo 2: Presentación de la solución propuesta

En el diagrama anterior se evidencia que el actor Usuario es el encargado de inicializar las interfaces gráficas de las funcionalidades del sistema. A continuación se describen textualmente los casos de uso Localizar_parada y Localizar_ruta. Las otras descripciones de los casos de uso se encuentran en el Anexo I.

CU Localizar_parada

Caso de Uso:	Localizar_parada
Actores:	Usuario
Propósito:	El caso de uso se lleva a cabo con el objetivo de ofrecerle al usuario la posibilidad de conocer la ubicación en el mapa de una parada determinada.
Resumen:	El caso de uso inicia cuando el usuario selecciona la opción Localizar Parada para escoger una parada determinada. El caso de uso finaliza cuando dicha parada es representada en el mapa.
Precondiciones:	La capa de paradas debe estar visible y cargada en el sistema.
Referencias:	RF 2
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando el usuario selecciona el ícono de la opción Localizar Parada en la barra de herramientas de localización. (Ver Interfaz 10)	2. El sistema muestra una nueva interfaz donde le brinda la posibilidad al usuario de seleccionar la dirección de la parada que desea buscar. (Ver



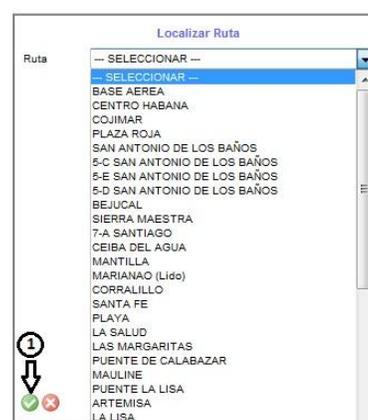
Capítulo 2: Presentación de la solución propuesta

	Interfaz 11)
3. El usuario selecciona la dirección de la parada de ómnibus y presiona el botón de aceptar (1).	4. El sistema valida que la selección sea “---SELECCIONAR---”, en ese caso ir al Flujo Alterno a.
	5. El caso de uso finaliza cuando el sistema muestra la parada con un color amarillo en el mapa. (Ver Interfaz 12)

Prototipo de Interfaz



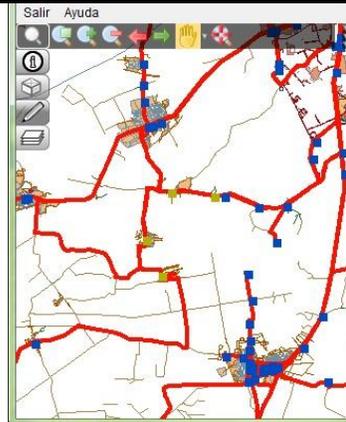
Interfaz 10



Interfaz 11



Capítulo 2: Presentación de la solución propuesta

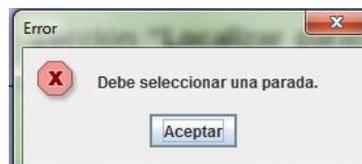


Interfaz 12

Flujo Alternativo

Acción del Actor	Respuesta del Sistema
	4.1 El sistema muestra un mensaje de error indicando que debe seleccionar una parada.
4.2 El usuario selecciona el botón Aceptar.	4.3 El sistema regresa al paso 2 del Flujo Normal de Eventos.

Prototipo de Interfaz



Interfaz 13

Poscondiciones

El sistema muestra la parada seleccionada con color amarillo.

Tabla #3: Descripción textual del caso de uso Localizar_parada.



Capítulo 2: Presentación de la solución propuesta

CU Localizar_ruta

Caso de Uso:	Localizar_ruta	
Actores:	Usuario	
Propósito:	El caso de uso se lleva a cabo con el objetivo de ofrecerle al usuario la posibilidad de conocer la ubicación de una ruta de ómnibus determinada.	
Resumen:	El caso de uso inicia cuando el usuario selecciona la opción Localizar Ruta para escoger una ruta de ómnibus determinada. El caso de uso finaliza cuando dicha ruta es representada en el mapa.	
Precondiciones:	La capa de rutas debe estar visible y cargada en el sistema.	
Referencias:	RF 3	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso inicia cuando el usuario selecciona el ícono de la opción Localizar Ruta en la barra de herramientas de localización. (Ver Interfaz 14)	2. El sistema muestra una nueva interfaz donde le brinda la posibilidad al usuario de seleccionar el nombre de la ruta de ómnibus que desea buscar. (Ver Interfaz 15)	
3. El usuario selecciona el nombre de la ruta de ómnibus y presiona el botón de	4. El sistema valida que la selección sea "--SELECCIONAR--", en ese caso ir al	



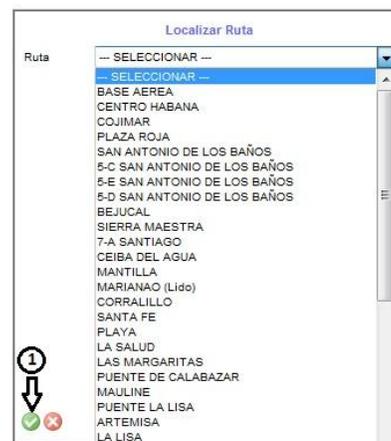
Capítulo 2: Presentación de la solución propuesta

aceptar (1).	Flujo Alternativo.
	5. El caso de uso finaliza cuando el sistema muestra la ruta con un color amarillo en el mapa. (Ver Interfaz 16)

Prototipo de Interfaz



Interfaz 14



Interfaz 15



Interfaz 16



Capítulo 2: Presentación de la solución propuesta

Flujo Alternativo	
Acción del Actor	Respuesta del Sistema
	4.1 El sistema muestra un mensaje de error indicando que debe seleccionar una ruta.
4.2 El usuario selecciona el botón Aceptar.	4.3 El sistema regresa al paso 2 del Flujo Normal de Eventos.
Prototipo de Interfaz	
	
Interfaz 17	
Poscondiciones	El sistema muestra la ruta seleccionada con color amarillo.

Tabla #4: Descripción textual del caso de uso Localizar_ruta.

2.6. Conclusiones

Se concluye en el presente capítulo, que al realizar la modelación del dominio se pudieron identificar aquellos conceptos y definiciones relacionadas con el entorno, permitiendo de esta forma una mejor comprensión del funcionamiento de los procesos del dominio y sus relaciones. Al realizar las actividades del flujo de trabajo requerimientos se crearon las bases para la construcción del sistema a desarrollar, con la identificación de las funcionalidades y restricciones que el sistema debe tener. A partir de las descripciones textuales realizadas a cada uno de los casos de uso recogidos en el diagrama de casos de uso del sistema, se pudo conocer detalladamente la forma en que interactuarán los usuarios y el sistema.



Capítulo 3: Construcción de la solución propuesta

CAPÍTULO 3: Construcción de la solución propuesta.

3.1 Introducción

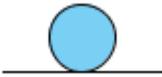
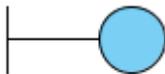
En el presente capítulo se aborda el análisis y diseño de la aplicación y se presentan los artefactos que se generan. Se analiza además la arquitectura del sistema. Para validar que la solución tenga la calidad requerida se ejecutan pruebas con el fin de detectar posibles errores, para ello se realiza el diseño de los casos de prueba.

3.2 Modelo de Análisis

El Modelo de Análisis puede ser considerado una primera aproximación al Modelo de Diseño. En el análisis no se definen características específicas como el lenguaje de programación, plataforma, así como otros relacionados con el sistema, ya que tiene como objetivo entender perfectamente los requisitos del sistema. Para ello se analizan los requisitos funcionales descritos en el capítulo anterior, refinándolos y estructurándolos.

3.2.1 Diagramas de Clases

Es utilizado para visualizar aquellas clases que comprenden el sistema y las relaciones existentes entre ellas, no son más que una abstracción del diseño. Las clases están formadas por los atributos que generalmente forman parte de las clases en el diseño e implementación. Una clase del análisis representa uno de los tres estereotipos en los que RUP propone clasificarlas: [47]

Nombre	Características	Representación
Entidad	Modelan la información de larga vida (persistencia) del sistema.	
Interfaz	Modelan la interacción entre el sistema y los actores. Pueden ser ventanas, formularios, dispositivos como impresoras y sensores, etc.	



Capítulo 3: Construcción de la solución propuesta

Control	Representan la coordinación entre objetos. Se encargan de la lógica del negocio. Se usan para representar el control de un caso de uso concreto. No representan ni interacciones con el actor ni problemas de almacenamiento de información.	
----------------	--	--

Tabla #5: Clasificación de las clases del análisis.

A continuación se muestran los diagramas de clases del análisis de los casos de uso Localizar_parada y Localizar_ruta. Los diagramas correspondientes a los restantes casos de uso se pueden encontrar en el Anexo II.

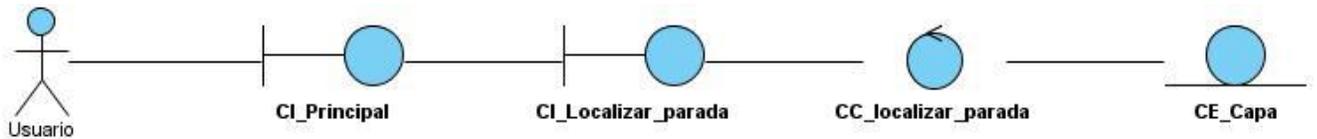


Figura #3: Diagrama de Clases del Análisis Localizar_parada.



Figura #4: Diagrama de Clases del Análisis Localizar_ruta.

3.2.2 Diagramas de Interacción

Un diagrama de interacción modela el flujo de control de una operación, describiendo cómo los objetos interactúan a través de mensajes para realizar las tareas. Existen dos tipos de diagramas de interacción, los de colaboración y los de secuencia.

Diagramas de Colaboración

En este tipo de diagramas se muestran los objetos o clases, las relaciones entre estos y los mensajes. Los mensajes son numerados con el fin de mostrar una secuencia. En las figuras 5 y 6 se pueden apreciar los diagramas de colaboración de los casos de uso Localizar_parada y Localizar_ruta, en el Anexo III se encuentran el resto de los diagramas.



Capítulo 3: Construcción de la solución propuesta

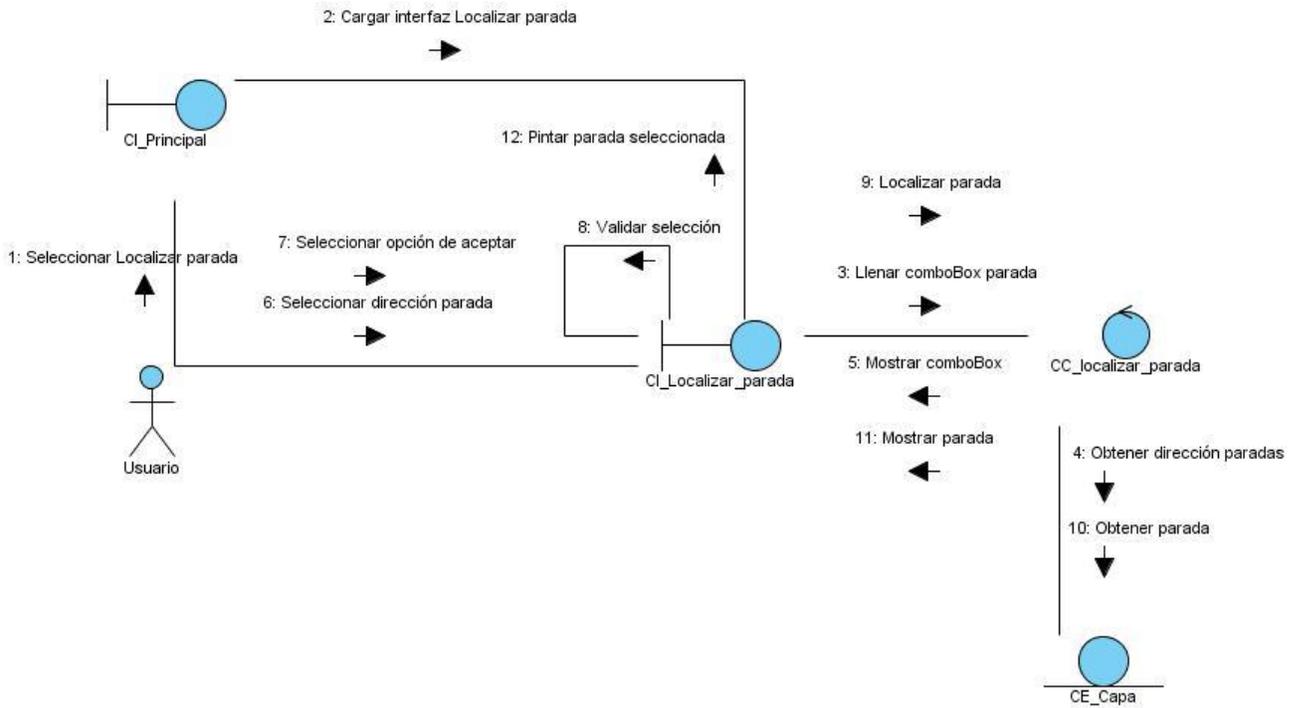


Figura #5: Diagrama de Colaboración del Análisis Localizar_parada.

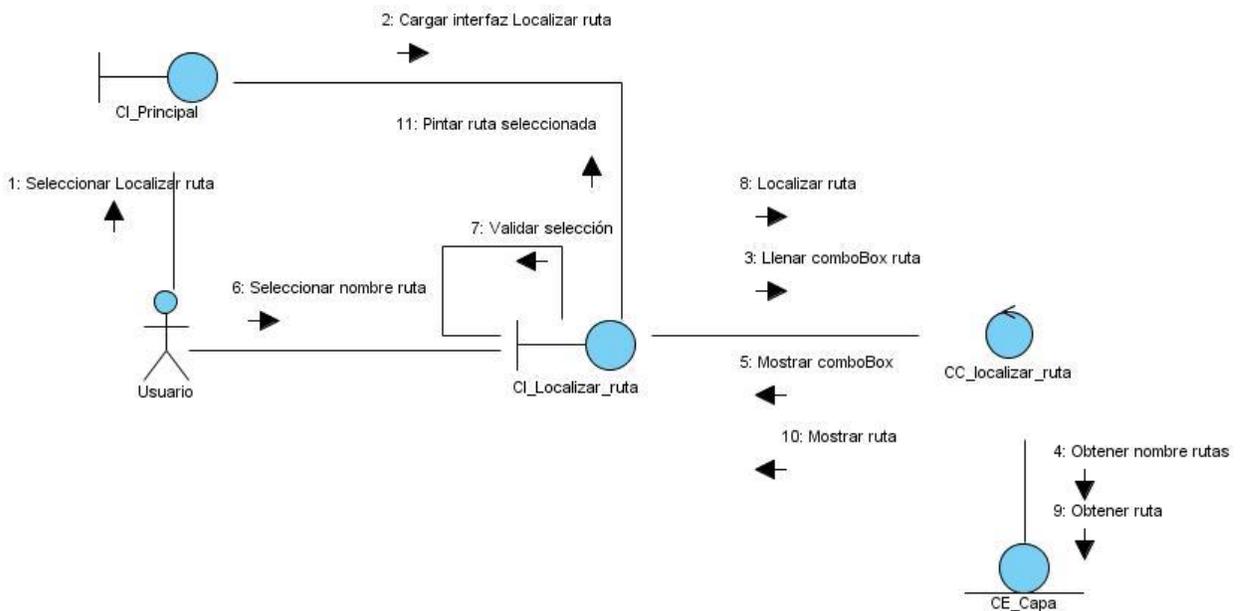


Figura #6: Diagrama de Colaboración del Análisis Localizar_ruta.



Capítulo 3: Construcción de la solución propuesta

Diagramas de Secuencia

Las interacciones son modeladas en un gráfico de dos dimensiones. En el eje X se ubican los objetos que participan en dicha interacción. En el eje Y se modela la línea de vida de los objetos, avanzando hacia abajo. Los mensajes se organizan en el diagrama en orden cronológico. En las figuras 7 y 8 se muestran los diagramas de secuencia de los casos de uso Localizar_parada y Localizar_ruta. El Anexo IV contiene el resto de los diagramas de secuencia.

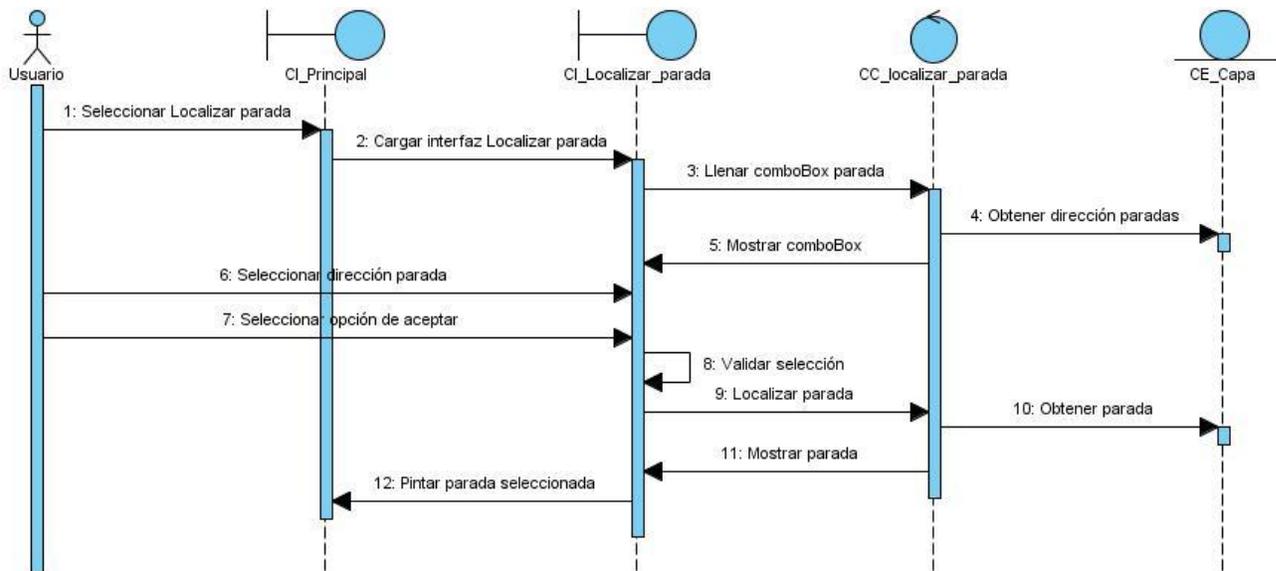


Figura #7: Diagrama de Secuencia del Análisis Localizar_parada.



Capítulo 3: Construcción de la solución propuesta

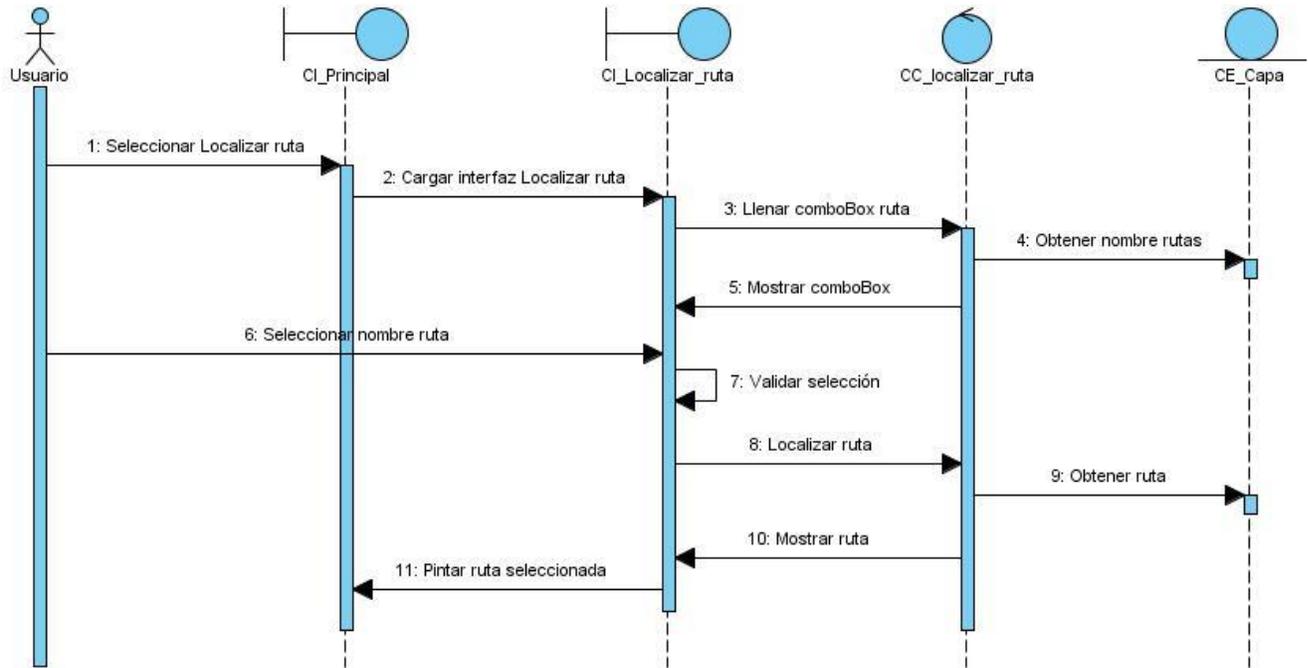


Figura #8: Diagrama de Secuencia del Análisis Localizar_ruta.

3.3 Modelo de Diseño

El diseño tiene como entrada principal el Modelo de Análisis pues proporciona una comprensión detallada de los requisitos. El Modelo de Diseño está muy cercano al de Implementación, aquí el sistema es modelado de forma tal que soporte tanto los requisitos funcionales como los no funcionales.

3.3.1 Arquitectura de Software

Los patrones de arquitectura definen un esquema para estructurar u organizar un *software* mediante subsistemas con sus responsabilidades. Son utilizados para facilitarle a los desarrolladores el diseño de un sistema. Se consideró que para realizar el diseño de la aplicación se utilizó la arquitectura en tres capas, esta consiste en dividir el sistema en varias capas lográndose de esta forma reducir el grado de complejidad. Las aplicaciones contienen códigos para la presentación, procesamiento y almacenamiento de datos, la diferencia se encuentra en la organización del mismo.



Capítulo 3: Construcción de la solución propuesta

La utilización de esta arquitectura permite desarrollar capas en paralelo, un mantenimiento más sencillo al tener que modificar por componentes en lugar de toda la aplicación y permite agregarle nuevos módulos aumentando su flexibilidad. El sistema se divide en las siguientes capas lógicas:

Interfaz: Representa la información proporcionada por el acceso a datos en un formato que permite la interacción entre el usuario y el sistema, se puede evidenciar en las interfaces de usuario.

Lógica Geo: Procesa las solicitudes realizadas por el usuario y se envían las respuestas produciendo generalmente cambios en el sistema. Esta capa sirve como intermediaria entre la interfaz y el acceso a datos.

Acceso a datos: Se encarga de acceder a los ficheros que almacenan los datos persistentes para proveer de información al usuario o a la aplicación misma, o para realizar cambios en la misma.

En la arquitectura se definen niveles que se corresponden con la forma en que las capas son distribuidas de forma física. Como las capas van a residir en el dispositivo móvil entonces la arquitectura de solución es de tres capas y un nivel.

3.3.2 Diagramas de Clases

En este diagrama se representan las clases y relaciones que se involucran en el sistema. Las siguientes figuras muestran los diagramas de clases de diseño de los casos de uso Localizar_parada y Localizar_ruta, en el Anexo V se pueden encontrar los restantes.



Capítulo 3: Construcción de la solución propuesta

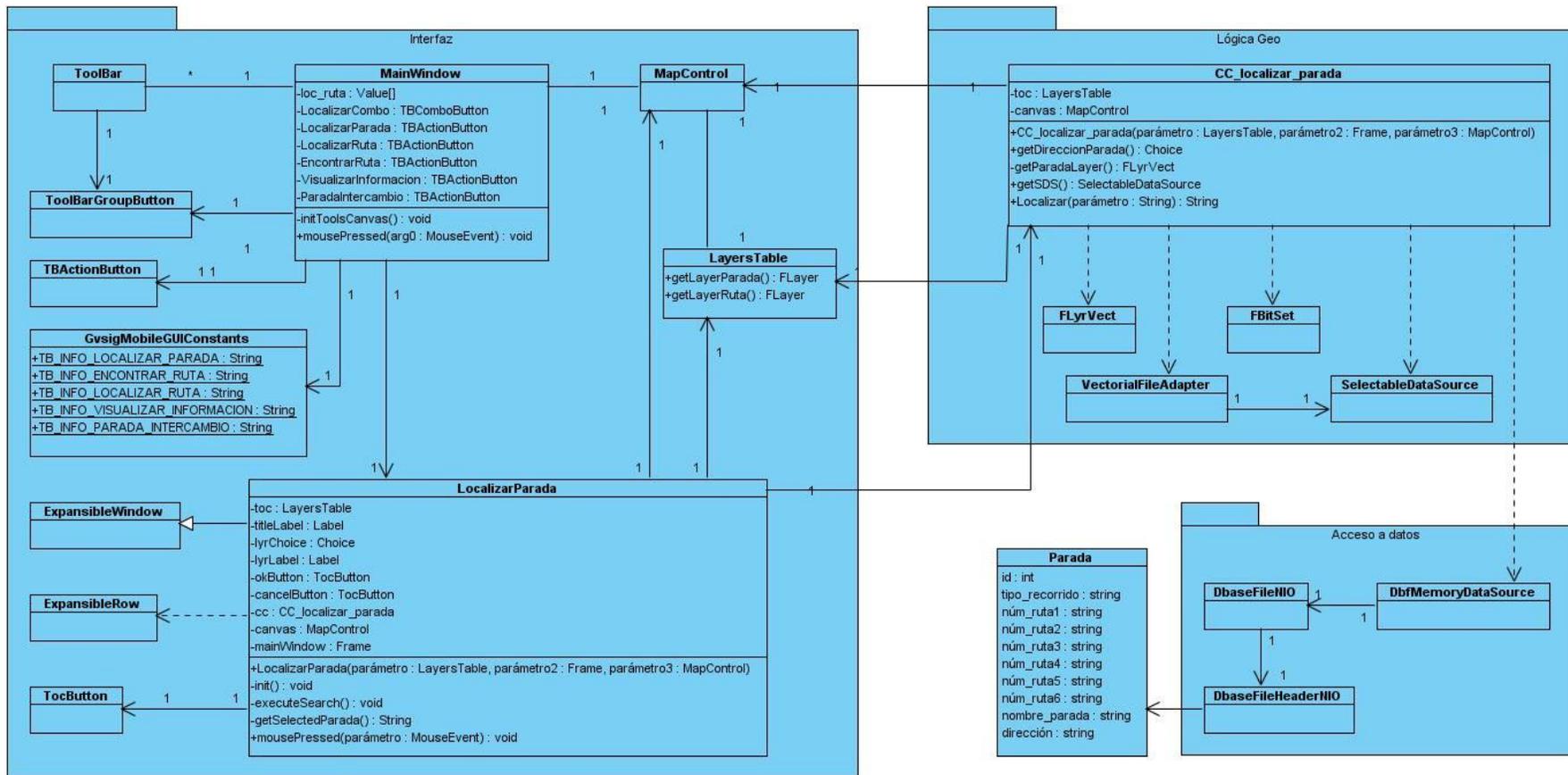


Figura #9: Diagrama de Clases del Diseño Localizar_parada.



Capítulo 3: Construcción de la solución propuesta

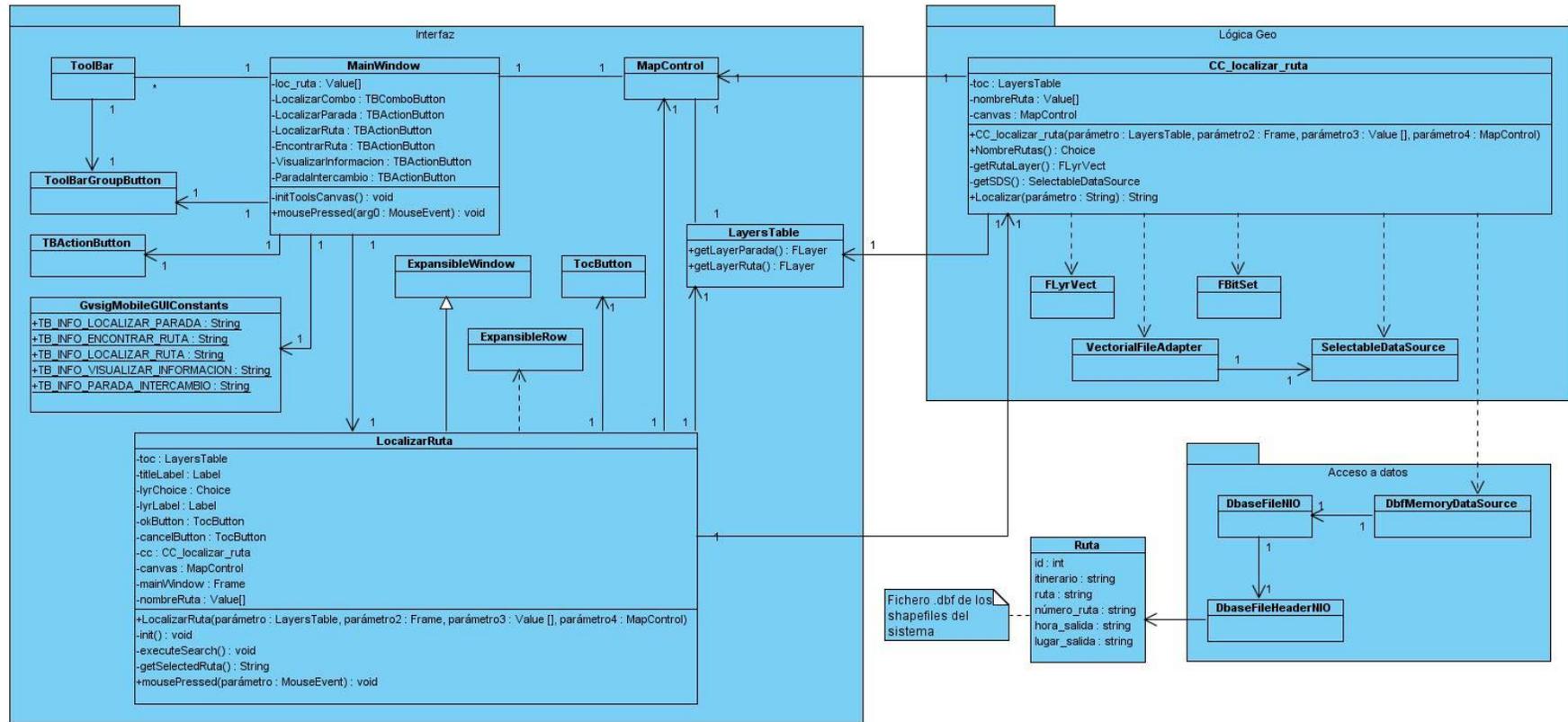


Figura #10: Diagrama de Clases del Diseño Localizar_ruta.



Capítulo 3: Construcción de la solución propuesta

3.4 Principios de diseño

3.4.1 Estándares de la interfaz de la aplicación

Un factor que influye en el éxito o fracaso del sistema lo constituye la calidad de las interfaces de usuario, por lo que se hace necesario prestar una especial atención durante su diseño. Los usuarios deben sentirse cómodos trabajando con la aplicación por lo que se utilizan interfaces amigables pero que a su vez sean efectivas para trabajar con el sistema.

El sistema será utilizado en dispositivos móviles, por lo que hay que tener en cuenta una serie de factores que los caracterizan, por ejemplo el tamaño de la pantalla, la ausencia de ratón y las limitaciones de memoria. Para ello se utilizan barras de herramientas con una cantidad de opciones limitadas para que el grado de complejidad de las interfaces le permita al usuario aprender a utilizar el sistema progresivamente. La información que se presenta debe ser cómoda de leer utilizándose para ello un alto contraste, combinaciones de colores como fondo blanco con texto negro y presentación de poca información.

Para obtener un mayor entendimiento de las funcionalidades se utilizan íconos pequeños que se relacionan con las acciones a realizar. Para aquellas personas que presenten deficiencias visuales como el daltonismo se ofrecerá una alternativa textual para los íconos utilizados. Cuenta con las entradas de texto indispensables para el funcionamiento del sistema. El vocabulario del sistema es claro y simple, semejante al utilizado por los usuarios y la fuente de letra del sistema es la misma para lograr una estandarización. Las interfaces cuentan con un título corto pero a la vez descriptivo.

3.5 Almacenamiento de datos

En la solución se utilizan capas vectoriales de formato *shapefile* para el almacenamiento de los datos persistentes. Este formato permite la representación de un solo tipo de objetos geométricos o *shape* en cada capa, estos pueden ser puntos, polilíneas (sucesiones de puntos unidos) o polígonos (polilíneas cerradas). Los puntos o vértices se encuentran en una posición definida por (X e Y), en ocasiones se puede tener información adicional como la altura (Z) o valores que se obtienen a través de otras mediciones (M) dependiendo del tipo de *shape*.



Capítulo 3: Construcción de la solución propuesta

Un *shapefile* trabaja con dos tipos de variables numéricas el entero de 32 bits y el doble de 64 bits. Los ordenamientos de *bytes* que utiliza son el *LittleIndian*, donde el orden de *bytes* es de derecha a izquierda y el *BigIndian* con orden de izquierda a derecha.

El archivo principal o *shp* está compuesto por una cabecera de archivo y varios registros, donde cada uno se divide en cabecera de registro y contenido de registro. La cabecera tiene una dimensión fija de 100 *bytes* y consta de un código, longitud del archivo, versión, tipo de *shape* y los límites. En los registros la cabecera tiene dimensión de 8 *bytes* para el número y la longitud del registro. El contenido de cada registro tiene una longitud variable pues depende del tipo de *shape*.

El archivo índice o *shx* consta de una cabecera de archivo con la misma estructura y longitud que el del archivo principal. Además, tiene registros de 8 *bytes* que se dividen en campos *Offset* para guardar la posición donde comienza el registro en el archivo principal y el campo tamaño del registro que coincide con la longitud del registro en el archivo principal.

Los archivos *dBase* o *dbf* son utilizados para el almacenamiento de información descriptiva de cada elemento. Cada registro del archivo principal está asociado a un único campo en la tabla, para ello se relacionan el número de registro en el archivo principal y el código de la tabla (*ObjectID*). Los atributos almacenados pueden ser numéricos, de texto o de fecha.



Capítulo 3: Construcción de la solución propuesta

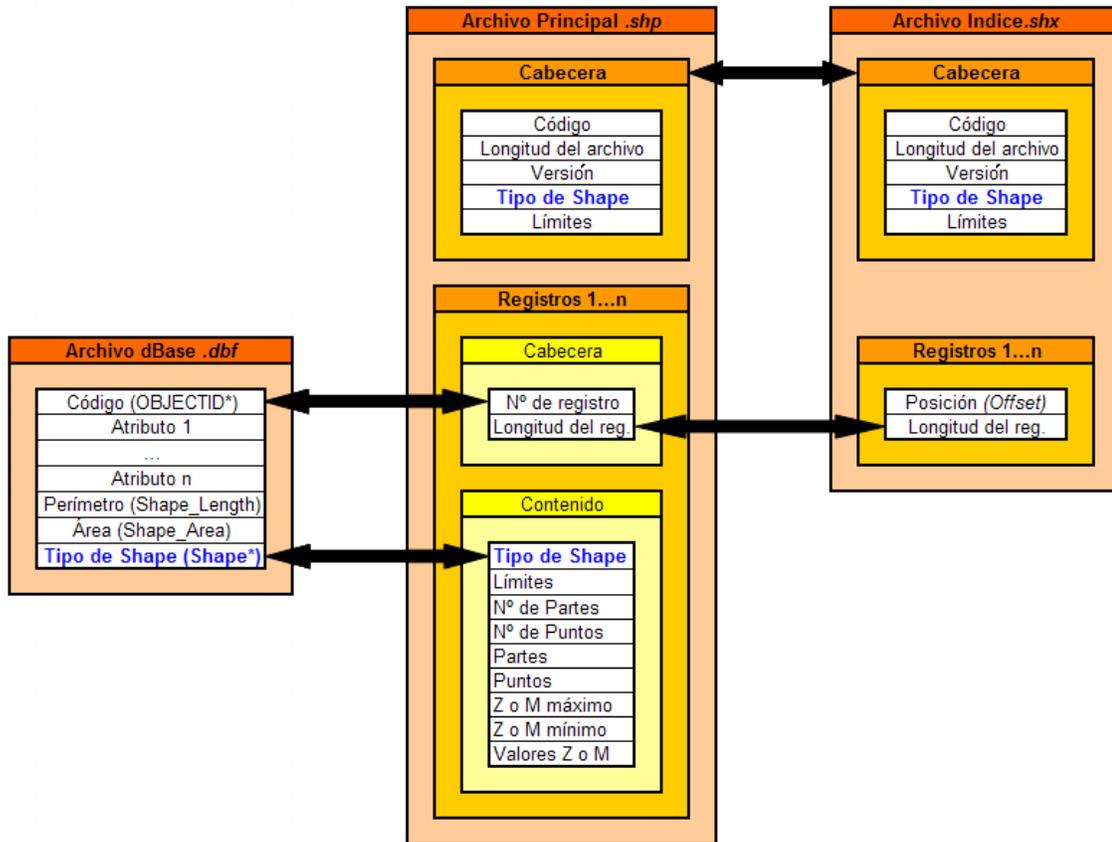


Figura #11: Estructura de los ficheros shapefile.

3.6 Implementación

3.6.1 Diagrama de Despliegue

Los diagramas de despliegue muestran la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Un nodo es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, generalmente tiene algo de memoria y, a menudo, capacidad de procesamiento. Los nodos se utilizan para modelar la topología del *hardware* sobre el que se ejecuta el sistema. Las conexiones entre nodos se modelan como asociaciones. [48]

Estos diagramas se componen también de dispositivos que no tienen capacidad de procesamiento en el nivel de abstracción en que se modela, sino que dan soporte a los nodos y conectores entre los elementos del modelo.



Capítulo 3: Construcción de la solución propuesta



Figura #12: Diagrama de despliegue

El diagrama de despliegue anterior consta de un único nodo que es el dispositivo móvil donde el sistema se ejecutará. La base de datos que utiliza el mismo es interna, es el fichero dbf que pertenece a cada una de las capas utilizadas.

3.6.2 Diagrama de Componente

El Modelo de Implementación tiene como entrada principal el Modelo de Diseño, donde sus elementos son transformados en componentes. Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. [49] Un componente representa un elemento de un modelo, en este caso las clases del Modelo de Diseño. Los componentes pueden ser ficheros de código fuente, ejecutables, tablas, librerías, entre otras. En un diagrama de componentes se modela la vista estática del sistema. Los siguientes diagramas representan los componentes distribuidos por capas lógicas.



Capítulo 3: Construcción de la solución propuesta

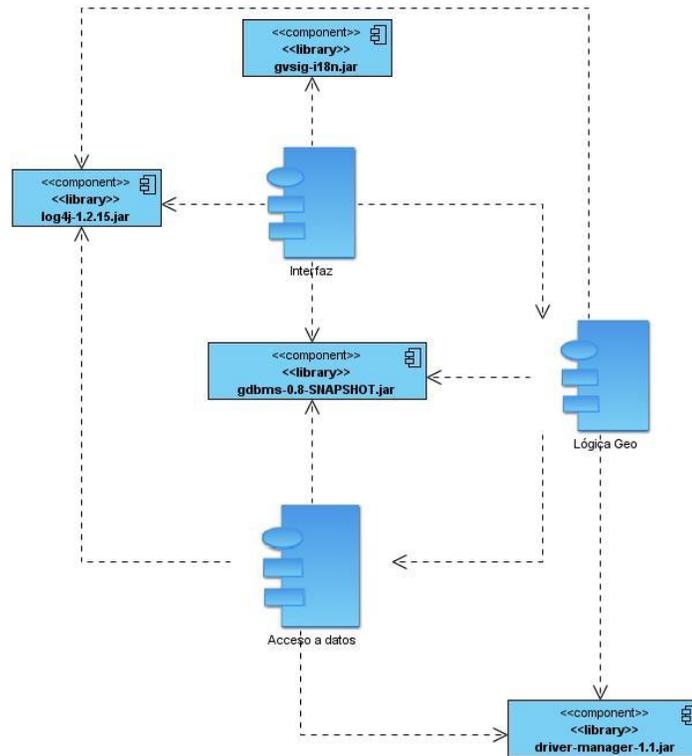


Figura #13: Diagrama de componentes.

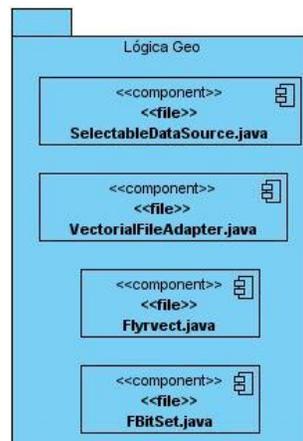


Figura #14: Diagrama de componentes. Lógica Geo.



Capítulo 3: Construcción de la solución propuesta

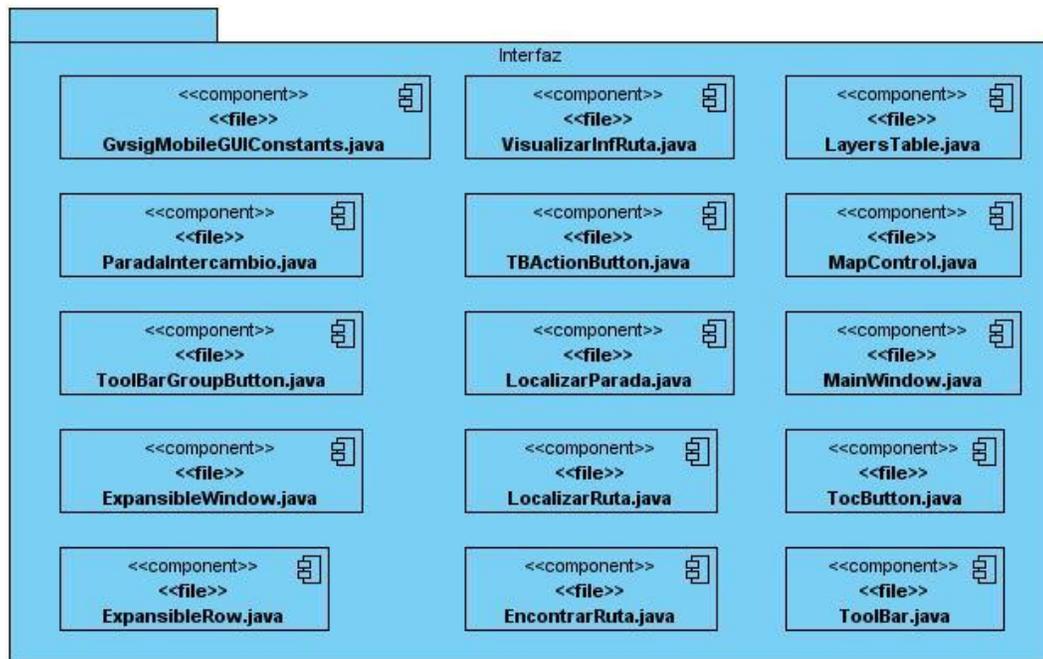


Figura #15: Diagrama de componentes. Interfaz.

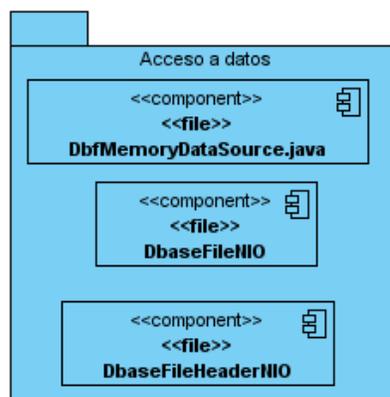


Figura #16: Diagrama de componentes. Acceso a datos.

3.7 Prueba del sistema propuesto

La etapa de prueba durante el desarrollo de un *software* es imprescindible, en esta se realizan una serie de actividades en vista a identificar la mayor cantidad de errores antes de ser entregado al cliente, de esta forma, se está evaluando la calidad del producto. Las pruebas de *software* se realizan con el objetivo de



Capítulo 3: Construcción de la solución propuesta

encontrar errores en el sistema, para ello se ejecuta la aplicación en circunstancias específicas, registrándose, analizándose y evaluándose los resultados obtenidos. Las pruebas deben realizarse durante todo el ciclo de vida de desarrollo del *software*.

El diseño de casos de prueba tiene vital importancia al realizar la estrategia de prueba porque son la base para evaluar un *software*. Tienen un conjunto de entradas y resultados específicos bajo ciertas condiciones, utilizados para validar que el sistema haga lo que está previsto. Antes de diseñar los casos de pruebas se realiza una selección de las técnicas y métodos que se utilizarán una vez que se vayan a aplicar dichas pruebas. Para definir cuáles serán los casos de prueba del sistema el equipo de trabajo se basará en los casos de uso, por lo que por cada caso de uso se realizará un caso de prueba, diseñándose once en total.

Se desea comprobar que las funciones del *software* estén correctas por lo que se utilizará el método de caja negra. En este método las pruebas son realizadas sobre la interfaz de usuario. El comportamiento interno y la estructura son irrelevantes ya que se centra en los requisitos funcionales. El mismo tiene como finalidad encontrar errores tales como: funciones incorrectas o ausentes, errores en la interfaz, errores en la estructura de datos y errores de rendimiento. Las pruebas de caja negra pretenden demostrar que las funcionalidades son operativas, las entradas son aceptadas de forma adecuada y el resultado obtenido es correcto. Para la selección de las entradas hay que identificar aquellas entradas que producen un comportamiento erróneo en el sistema, produciendo de esta forma resultados que revelan defectos.

La técnica seleccionada es la de partición equivalente, esta consiste en dividir el campo de entrada de un programa en clases de los que se pueden derivar casos de prueba. Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. Típicamente, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica. [50] Durante la ejecución de los casos de prueba cada uno de los escenarios de los casos de prueba será probados, se utilizarán los datos previamente definidos. La información que se obtiene durante el proceso de pruebas se recogerá en el expediente de proyecto.

A continuación se muestran las tablas relacionadas con el caso de prueba Localizar parada.



Capítulo 3: Construcción de la solución propuesta

Descripción General

El caso de uso inicia cuando el usuario selecciona la opción Localizar Parada para escoger una parada determinada. El caso de uso finaliza cuando dicha parada es representada en el mapa.

Condiciones de Ejecución

La capa de paradas debe estar visible y cargada en el sistema.

Nombre sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: Localizar parada	EC 1.1: Localizar parada correctamente.	Esta funcionalidad permite al usuario seleccionar una parada para conocer la localización en el mapa dibujándose con un color amarillo.	<ul style="list-style-type: none"> • El usuario selecciona el ícono de la opción Localizar Parada en la barra de herramientas de localización. • El sistema muestra la interfaz con el campo que debe seleccionar el usuario. • El usuario selecciona la parada. • El sistema muestra la ubicación de la parada seleccionada en el mapa.
	EC 1.2: Localizar parada sin seleccionarla.	El sistema no dibuja la parada con color amarillo en el mapa debido a que el usuario no seleccionó ninguna.	<ul style="list-style-type: none"> • El usuario selecciona el ícono de la opción Localizar Parada en la barra de herramientas de localización. • El sistema muestra la interfaz con el campo que debe seleccionar el usuario. • El sistema muestra un mensaje de error indicando que debe seleccionar una parada.

Tabla #6: Secciones a probar en el caso de uso.



Capítulo 3: Construcción de la solución propuesta

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Parada	Lista desplegable	No	Debe seleccionar una parada.

Tabla #7: Descripción de variables

ID del escenario	Escenario	Parada	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Localizar parada correctamente	V/ "La Lisa"	El sistema selecciona con color amarillo la parada en el mapa.	Satisfactorio.
EC 1.2	Localizar parada sin seleccionarla	I/ "--SELECCIONAR--"	El sistema muestra un mensaje de error indicando que debe seleccionar una parada.	Satisfactorio.

Tabla #8: Matriz de datos de la SC 1 Localizar parada.

3.8 Conclusiones

Con la modelación de los diagramas relacionados con el análisis y diseño se crearon las bases para la implementación del sistema. La utilización de la arquitectura tres capas permitió disminuir el grado de complejidad del sistema al dividirlo en capas. En la etapa de implementación se elaboraron los diagramas de componentes y despliegue de la solución los cuales permitieron organizar los componentes y los nodos físicos sobre los cuales funcionará el sistema. Por último, se realizaron las pruebas de caja negra a las interfaces del sistema comprobando que se cumplieron los requisitos especificados al obtenerse resultados satisfactorios.



CONCLUSIONES

Al finalizar la investigación científica y realizar todas las tareas propuestas para dar cumplimiento al objetivo general, se puede arribar a las siguientes conclusiones:

Las funcionalidades del *software* llevan a cabo las tareas correctamente satisfaciendo los requisitos identificados por los clientes. La implementación de las funcionalidades agrupadas en la barra de localización apoyará el proceso de toma de decisiones por parte de los usuarios de dispositivos móviles. Utilizando herramientas libres se desarrolló un producto con esta característica, cumpliendo de esta forma con las políticas de la universidad respecto a no utilizar herramientas privativas. Se obtuvo como resultado un producto escalable al utilizar la arquitectura en capas y que cumple con los requisitos de los clientes. Las pruebas efectuadas a las funcionalidades del sistema garantizaron la calidad de la solución propuesta. Durante el desarrollo del sistema se generaron un conjunto de diagramas que permiten realizar mejoras futuras a la solución.



RECOMENDACIONES

Con el objetivo de mejorar el resultado obtenido se proponen las siguientes recomendaciones:

- ✓ Mejorar la cartografía utilizada en la solución.
- ✓ Crear nuevas funcionalidades en versiones futuras del sistema.



REFERENCIAS BIBLIOGRÁFICAS

1. **Mendiburu, Diego.** La increíble vida de Martin Cooper. Inventó el celular, cambió el mundo y le pagaron un dólar por ello. [En línea] 18 de Enero de 2010. [Citado el: 9 de Noviembre de 2010.] http://docs.google.com/viewer?a=v&q=cache:rVl0AmQ3gKoJ:www.m-x.com.mx/xml/pdf/207/54.pdf+el+celular+%2B+Martin+Cooper&hl=es&gl=cu&pid=bl&srcid=ADGEESiGb1QJWBOnCqejZqsZ151Y3ugh_B-Om0-VXeja9EOZqdyWXeB7hqD7PI8ko4krPwDnWr2etIshAuD6S6GENciY473hNXfKa21WR0YrpZG .
2. **Zayas, Dr. Cs. Carlos Alvarez de.** *METODOLOGIA DE LA INVESTIGACION CIENTIFICA*. 1995.
3. **Yagüez, Ing.Agr. Julio C. Domingo.** Instituto Nacional de Tecnología Agropecuaria. [En línea] 2002. [Citado el: 13 de Noviembre de 2010.] http://www.inta.gov.ar/barrow/info/documentos/SIG/que_es_sig.htm.
4. **Caño, Juan Luís.** Terra. [En línea] 21 de Enero de 2003. [Citado el: 13 de Noviembre de 2010.] periodista español. <http://www.terra.es/tecnologia/articulo/html/tec7905.htm>.
5. **Álvarez, Juan Manuel.** SlideShare. [En línea] [Citado el: 20 de Enero de 2011.] <http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.sli-deshare.net%2FJmaquino%2Fdispositivos-moviles&ei=jA1fTY-NIMKu8QOZnYjeCw&usq=AFQjCNFkaXETdRS7x2OB1nKMQ09f7NjCAQ>.
6. **Alonso, Arturo Baz, y otros.** [En línea] [Citado el: 13 de Noviembre de 2010.] <http://156.35.151.9/~smi/5tm/09trabajos-sistemas/1/Memoria.pdf>.
7. **Bravo, Javier Domínguez.** [En línea] Octubre de 2000. [Citado el: 10 de Noviembre de 2010.] http://www.google.com/cu/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.preval.info%2Fprograma%2Fwp-content%2Fuploads%2F2008%2F09%2Fitc_cartografia_sig.pdf&rct=j&q=Breve%20Introducci%C3%B3n%20a%20la%20Cartograf%C3%ADa%20y%20a%20los%20Sistemas.943..



Referencias Bibliográficas

8. **Páez, Lic. Sergio Luis Alberto.** [En línea] [Citado el: 15 de Noviembre de 2010.] Titular de la Cátedra “Evolución del Pensamiento Geográfico”, Profesorado en Geografía, Instituto Superior “Antonio Ruiz de Montoya. <http://www.scribd.com/doc/2965352/Breve-historia-de-los-Mapas>.
9. Definicion.de. [En línea] 2008. [Citado el: 28 de Noviembre de 2010.] <http://definicion.de/cartografia/>.
10. Definicion ABC. [En línea] 2008. [Citado el: 28 de Noviembre de 2010.] <http://www.definicionabc.com/ciencia/cartografia.php>.
11. **Arranz, Lorena.** [En línea] 17 de Octubre de 2005. [Citado el: 30 de Noviembre de 2010.] <http://recursostic.educacion.es/observatorio/web/es/software/software-general/293-lorena-arranz>.
12. **Franco, Rodolfo.** [En línea] 2 de Agosto de 2010. [Citado el: 28 de Noviembre de 2010.] http://gemini.udistrital.edu.co/comunidad/profesores/rfranco/vector_raster.htm.
13. Geoenseñanza. [En línea] 2006. [Citado el: 12 de Diciembre de 2010.] <http://www.saber.ula.ve/bitstream/123456789/27748/1/articulo9.pdf>.
14. ESBI COMPUTING. [En línea] [Citado el: 26 de Noviembre de 2010.] http://www.esbic.ie/geobusiness/Mobile_gis/Main_Components.htm.
15. extensionfile. [En línea] 2010. [Citado el: 25 de Octubre de 2010.] <http://kml.extensionfile.net/es>.
16. **Romero, Emilio López.** Mapping Interactivo. [En línea] Enero de 2008. [Citado el: 26 de Octubre de 2010.] http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=1461.
17. Alcaldía de Santiago de Cali. [En línea] 2 de Diciembre de 2010. [Citado el: 12 de Diciembre de 2010.] http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=122.
18. **Olloui, Juan Salinas.** [En línea] 25 de Septiembre de 2009. [Citado el: 24 de Noviembre de 2010.] http://www.geogra.uah.es/gisweb/practica-vectorial/Formato_Shapefile.htm.
19. [En línea] Mayo de 2007. [Citado el: 25 de Noviembre de 2010.] <http://www.google.com/cu/url?sa=t&source=web&cd=1&ved=0CB0QFjAA&url=http%3A%2F%2Fwww.esri.com%2Flibrary%2Fbestpractices%2FMobile->



Referencias Bibliográficas

[gis.pdf&rct=i&q=GIS%20Best%20Practices%20Mobile%20GIS&ei=lb76TKviE4mnnQe88fnGCq&usq=AFQjCNH-22xSJhKZTfNQ2sAzTePYH9BYTg&cad=rja.](#)

20. **Cuzco, José Froilán Jiménez.** slideshare. [En línea] [Citado el: 28 de Enero de 2011.] <http://www.slideshare.net/UCACUE/proyecto-de-investigacion-6978494>.
21. Navactiva. [En línea] [Citado el: 28 de Enero de 2011.] http://www.navactiva.com/es/asesoria/definicion-de-rutas-de-transporte_23905.
22. **UCI.** SIG Rutas. *SIG Rutas*. [En línea] [Citado el: 20 de Mayo de 2011.] http://10.34.12.22:5800/sig_rutas/sig/htdocs/index.php.
23. **Núñez1, MSc. Héctor Manuel Fernández.** scielo. *scielo*. [En línea] 2011. [Citado el: 20 de Mayo de 2011.] http://scielo.sld.cu/scielo.php?pid=S1561-30032006000300003&script=sci_arttext.
24. **Primelles Fariñas, MSc Josefa, y otros.** GEOINFO. *GEOINFO*. [En línea] 2004. [Citado el: 20 de Mayo de 2011.] <http://www.iga.cu/Publicaciones/CD1/Temas/Percepcion%20R%20y%20SIG/Josefa%20Primelles/Josefa.htm>.
25. Gobierno de Aragón. *Gobierno de Aragón*. [En línea] [Citado el: 20 de Mayo de 2011.] <http://sitar.aragon.es/visor/#>.
26. **Hofmann, Rob.** [En línea] Octubre de 2008. [Citado el: 21 de Noviembre de 2010.] http://www.esri-chile.com/lauc2008/arch_pdf/proceeding2008/TT10_hofmann_%20ArcGISMobileUnaIntroduccion.pdf.
27. Kosmo. [En línea] [Citado el: 12 de Enero de 2011.] <http://www.opengis.es/index.php>.
28. **Lajara, M. M., Carrasco Marimón, J., & del Rey Pérez, A.** (s.f.). Recuperado el 10 de Noviembre de 2010, de <http://www.sigte.udg.edu/jornadassiglibre2010/uploads/Articles/a41.pdf>.
29. **Nourie, D.** (2005, marzo 24). Oracle. Retrieved Enero 30, 2011, from <http://java.sun.com/developer/technicalArticles/tools/intro.html>



Referencias Bibliográficas

30. [En línea] 30 de Diciembre de 2006. [Citado el: 27 de Noviembre de 2010.] <http://www.um.es/docencia/barzana/IAGP/lagp2.html>.
31. Procalculo Prosis. [En línea] [Citado el: 12 de Enero de 2011.] <http://www.procalculoosis.com/Productos/ESRI-ArcGIS-Online/ArcGIS-Mobile.aspx>.
32. ESRI Chile http://www.esri-chile.com/productos_detalle.php?cod=2322&cat=20&subcat=28.
33. Definición.org. [En línea] [Citado el: 28 de Noviembre de 2010.] <http://www.definicion.org/lenguaje-de-programacion>.
34. **Canós, José H., Letelier, Patricio y Penadés, M^a Carmen.** Metodologías Ágiles en el Desarrollo de Software. [En línea] <http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFiAA&url=http%3A%2F%2Fwww.upv.es%2Fpublicaciones%2Farchivos%2F1069167248521%2Factas.pdf&ei=GphhTdqZOZOD4Qa1p6jcCQ&usg=AFQjCNHCzcum1li2L12xYQ7m5LILs5gS2A>.
35. **Reynoso, Carlos.** [En línea] <http://www.google.com/url?sa=t&source=web&cd=2&ved=0CBsQFjAB&url=http%3A%2F%2Fwww.reynoso.com.ar%2Farchivos%2Fcarlos-reynoso-metodos-agiles-academia.ppt&ei=DqhVTecPwr22B93MqJEN&usg=AFQjCNGsImEnbyE57nYImol8pJiKPKS4Rq>
36. **Enrich, Margarita Fernández.** [En línea] 4 de Abril de 2003. http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFiAA&url=http%3A%2F%2Fwww.dsic.upv.es%2Fasignaturas%2Ffacultad%2Flsi%2Ftrabajos%2F282002.ppt&ei=9apVTbjLC9Ghtwee_4TJDA&usg=AFQjCNF07V1x5vVIqc7I-Vv8yH9uDDp1sQ.
37. [En línea] [Citado el: 28 de Enero de 2011.] http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFiAA&url=http%3A%2F%2Fwww.mastersoft.com.ar%2FMsWeb%2Fotros_archivos%2FNotaScrumPCUsers.pdf&ei=-6hVTaeuKY-CtgeT4rTODA&usg=AFQjCNEEU0CHAG72SomzFHbFTXI-6hhSTg.



Referencias Bibliográficas

38. ICONIX, Características de. [En línea] [Citado el: 28 de Enero de 2011.] <http://eqaula.org/eva/file.php/853/UsodelCONIX.pdf>.
39. **ALBERTO, SERNA BARRERA JUAN y JIMENEZ CRUZ, RICARDO JOEL.** [En línea] [Citado el: 30 de Noviembre de 2010.] http://webcache.googleusercontent.com/search?q=cache:jybBEE_0p5MJ:proyectosport.freeiz.com/archivos_de_categorias/proyecto_Sport_Byke/1PROY1_METODOLOGIA_RUP.docx+rup+centrado+en+a+arquitectura&cd=8&hl=es&ct=clnk&gl=cu&lr=lang_es&client=firefox-a.
40. **Sanchez, María A. Mendoza.** informatizate. [En línea] 7 de Junio de 2004. [Citado el: 28 de Enero de 2011.] http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
41. **Perez, Segundo.** [En línea] 2010. [Citado el: 29 de Enero de 2011.] <http://www.scribd.com/doc/35014145/Comparacion-Metodo-Msf-y-Metodologia-Cascada>.
42. **Figuroa, Roberth G., Solís, Camilo y Cabrera, Armando.** [En línea] [Citado el: 29 de Enero de 2011.] http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fadonisn.et.files.wordpress.com%2F2008%2F06%2Farticulo-metodologia-de-sw-formato.doc&ei=ZsRVTd6VFMqftgfeh_2eDQ&usq=AFQjCNH6ClwoJOFA1MLY_6yuYhpBZUKGFQ.
43. **Orallo, Enrique Hernández.** Google doc. Google doc. [En línea] [Citado el: 23 de Noviembre de 2010.] http://docs.google.com/viewer?a=v&q=cache:xHa5jIFN1ocJ:www.disca.upv.es/enheror/pdf/ActaUML.PDF+Lenguaje+modelado+unificado+ventajas&hl=es&gl=cu&pid=bl&srcid=ADGEESh96QYhwwLharkHouuetxq0tiZ8kQjKgsCjp8qlRBfNimvNbewyAAOpQ_ticFdXJKFghJ8TDTGy7Mk50sE2szyVY9p2h.
44. **Chamba, Julio.** Scribd. [En línea] [Citado el: 25 de Enero de 2011.] <http://www.scribd.com/doc/40173039/Herramientas-Case>.
45. **Blanco, Rubén González y Pérez Tobalina, Sergio.** scribd. [En línea] [Citado el: 25 de Enero de 2011.] <http://www.scribd.com/doc/38823255/LESE-2-Introduccion-a-Rational-Rose>.



Referencias Bibliográficas

46. Free Download Manager. Free Download Manager. [En línea] [Citado el: 29 de Noviembre de 2010.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28PE%29_%5Bcuenta_de_Windows_14730_p/.
47. **Carrera, Mario.** GvSIG. GvSIG. [En línea] 1 de Junio de 2010. [Citado el: 20 de Mayo de 2011.] <http://www.gvsig.org/web/projects/gvsig-mobile/official/piloto-gvsig-mobile-0.3/requerimientos-del-sistema>.
48. **Cáceres, Paloma.** [En línea] 2010. [Citado el: 20 de Mayo de 2011.] http://www.kybele.etsii.urjc.es/docencia/AIR_GIS_M/2010-2011/Material/%5BAIR-1011%5DTema4.pdf.
49. [En línea] [Citado el: 20 de Mayo de 2011.] <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>.
50. Universidad Distrital Francisco Jose de Caldas. [En línea] 2007. [Citado el: 20 de Mayo de 2011.] <http://gemini.udistrital.edu.co/comunidad/grupos/arquisoft/fileadmin/Estudiantes/Pruebas/HTML%20-%20Pruebas%20de%20software/node28.html>.



BIBLIOGRAFÍA

1. **Ing. Agr. Néstor Di Leo.** ZONA EDUCATIVA. [En línea] 10 de Junio de 2002. [Citado el: 10 de Noviembre de 2010.] <http://www.fcagr.unr.edu.ar/mdt/GTS/Zonaedu/pagintrod.htm>.
2. SIG y Medio Ambiente: principios básicos. PRIMERO.
3. [En línea] [Citado el: 24 de Noviembre de 2010.] [http://redgeomatica.rediris.es/cartoprofesores/Moises_y_Teresa/\(Tema_12a\)%20SISTEMAS_INFORMACION_CART.ppt..](http://redgeomatica.rediris.es/cartoprofesores/Moises_y_Teresa/(Tema_12a)%20SISTEMAS_INFORMACION_CART.ppt..)
4. **Bravo, Javier Domínguez.** [En línea] Octubre de 2000. [Citado el: 10 de Noviembre de 2010.] http://www.google.com/cu/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.preval.info%2Fprograma%2Fwp-content%2Fuploads%2F2008%2F09%2Fitc_cartografia_sig.pdf&rct=j&q=Breve%20Introducci%C3%B3n%20a%20la%20Cartograf%C3%ADa%20y%20a%20los%20Sistemas.943.
5. **Álvarez, Juan Manuel.** SlideShare. [En línea] [Citado el: 20 de Enero de 2011.] <http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBUQFjAA&url=http%3A%2F%2Fwww.slideshare.net%2FJmaquino%2Fdispositivos-moviles&ei=jA1fTY-NIMKu8QOZnYjeCw&usq=AFQjCNFkaXETdRS7x2OB1nKMQ09f7NjCAQ>.
6. **Hofmann, Rob.** [En línea] Octubre de 2008. [Citado el: 21 de Noviembre de 2010.] http://www.esri-chile.com/lauc2008/arch_pdf/proceeding2008/TT10_hofmann_%20ArcGISMobileUnaIntroduccion.pdf.
7. Trimble. [En línea] [Citado el: 19 de Noviembre de 2010.] <http://www.elagrimensor.net/trimble.html>.
8. Telefonos-moviles.com. [En línea] 20 de Mayo de 2000. [Citado el: 20 de Noviembre de 2010.] <http://www.telefonos-moviles.com/articles/item.asp?id=6>.
9. Belatrix Software Factory. [En línea] Diciembre de 2008. [Citado el: 20 de Noviembre de 2010.] <http://www.belatrixsf.com/index.php/spdesarrollosmoviles>.



Bibliografía

10. [En línea] 26 de Octubre de 2009. [Citado el: 15 de Noviembre de 2010.] <http://centros.edu.xunta.es/iessantome/files/TECNOLOG%C3%8DAS%20DE%20LA%20INFORMACI%C3%93N%20Y%20LA%20COMUNICACI%C3%93N.pdf>.
11. **Castro, Fidel.** [En línea] [Citado el: 21 de Noviembre de 2010.] <http://www.cuba.cu/gobierno/discursos/1968/esp/f300968e.html>
12. Alcaldía de Santiago de Cali. [En línea] 28 de Septiembre de 2010. [Citado el: 16 de Noviembre de 2010.] http://www.cali.gov.co/publicaciones.php?id=33660#cita_notas_1.
13. **Sanz, Jorge y Montesinos, Miguel.** [En línea] Febrero de 2009. [Citado el: 16 de Noviembre de 2010.] http://www.cuadernos.tpdh.org/file_upload/08_TIG_14_herramientasSIGlibres.pdf.
14. **Barbero, Rosario Escudero y Sánchez Hernández, Blanca.** MAPPING INTERACTIVO. [En línea] Diciembre de 2000. [Citado el: 14 de Noviembre de 2010.] http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=122.
15. **Romero, Emilio López.** MAPPING INTERACTIVO. [En línea] Enero de 2008. [Citado el: 14 de Noviembre de 2010.] http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=1461.
16. File extension database. [En línea] [Citado el: 17 de Noviembre de 2010.] <http://kml.extensionfile.net/es>.
17. **Lajara, Miguel Montesinos y Carrasco Marimón, Javier.** [En línea] [Citado el: 15 de Noviembre de 2010.] http://dugi-doc.udg.edu/bitstream/10256/1110/9/gvSIG_Art.pdf.
18. Pressman, R. S. Ingeniería de Software. Un enfoque práctico.
19. Reynoso, C., & Kiccilof, N. (2004). Estilos y Patrones en la Estrategia de Arquitectura de Microsoft. BUENOS AIRES.
20. Olloqui, J. S. (25 de Septiembre de 2009). Recuperado el 02 de Abril de 2011, de http://www.geogra.uah.es/gisweb/practica-vectorial/Formato_Shapefile.htm.



Bibliografía

21. ESRI. (Julio de 1998). ESRI Shapefile Technical Description. Recuperado el 02 de Abril de 2011, de www.esri.com/library/whitepapers/pdfs/shapefile.pdf.
22. Ontiveros, G. (2000). Formato propietario de ArcView. Recuperado el 02 de Abril de 2011, de http://www.google.com/url?sa=t&source=web&cd=1&ved=0CBYQFjAA&url=http%3A%2F%2Fcatarina.udlap.mx%2Fu_dl_a%2Ftales%2Fdocumentos%2Fflis%2Fgarcia_o_ff%2Fcapitulo4.pdf&ei=732_TcvvKsrEgQe7u6nRBQ&usg=AFQjCNEC21xCGUyvmkdWrxGIFtG4FPnZ6A.



GLOSARIO DE TÉRMINOS

CDC (*Connected Device Configuration*): Es una configuración desarrollada para dispositivos con 2 MB o más de memoria disponible para la plataforma, incluyendo RAM y memoria *flash* o ROM.

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

GPS: Sistema Global de Posicionamiento.

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

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

PDA (*Personal Digital Assistant*): Ordenador de pequeño tamaño cuya principal función era, en principio, mantener una agenda electrónica. No obstante, cada vez más se va confundiendo con los ordenadores de mano y de palma.

SIG: Sistemas de Información Geográfica (GIS por sus siglas en inglés).

SMS (*Short Message Service*): Es un servicio de mensajería por teléfonos celulares. Con este sistema se puede enviar o recibir mensajes entre celulares y, luego, a través de internet.

WMS (*Web Map Services*): Servicios de Mapas en Web. Especificación emitida por OGC.

W3C: *World Wide Web Consortium*.

XML (*Extensible Markup Language*): Desarrollado por el W3C. Su objetivo es conseguir páginas *Web* más semánticas. XML separa la estructura del contenido y permite el desarrollo de vocabularios modulares. Se trata de un formato abierto.