



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 6

Desarrollo de un Sistema de Información Geográfica para la Universidad de las Ciencias Informáticas basado en la Plataforma GvSIG Mobile.

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

Autor:

Ruben Vázquez Manso

Tutor:

Lic. Mario Liosbel Díaz Abreu

Co-Tutora:

Lic. Surelys Veunes Pérez

Ciudad de la Habana, Junio 2011

Año 53 de la Revolución

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al proyecto SIG Móviles y al Centro GEYSED de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los _____ días del mes de _____ del año 2011.

Ruben Vázquez Manso

Lic. Mario Liosbel Díaz Abreu

AVAL DEL PROYECTO

La Habana, 30 de Mayo de 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 Ruben Vázquez Manso tutorado por el Lic. Mario Liosbel Díaz Abreu y la Lic. Surelys Veunes Pérez tributa de forma integral 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 de la Universidad de las Ciencias Informáticas y a la información socio-económico correspondiente.

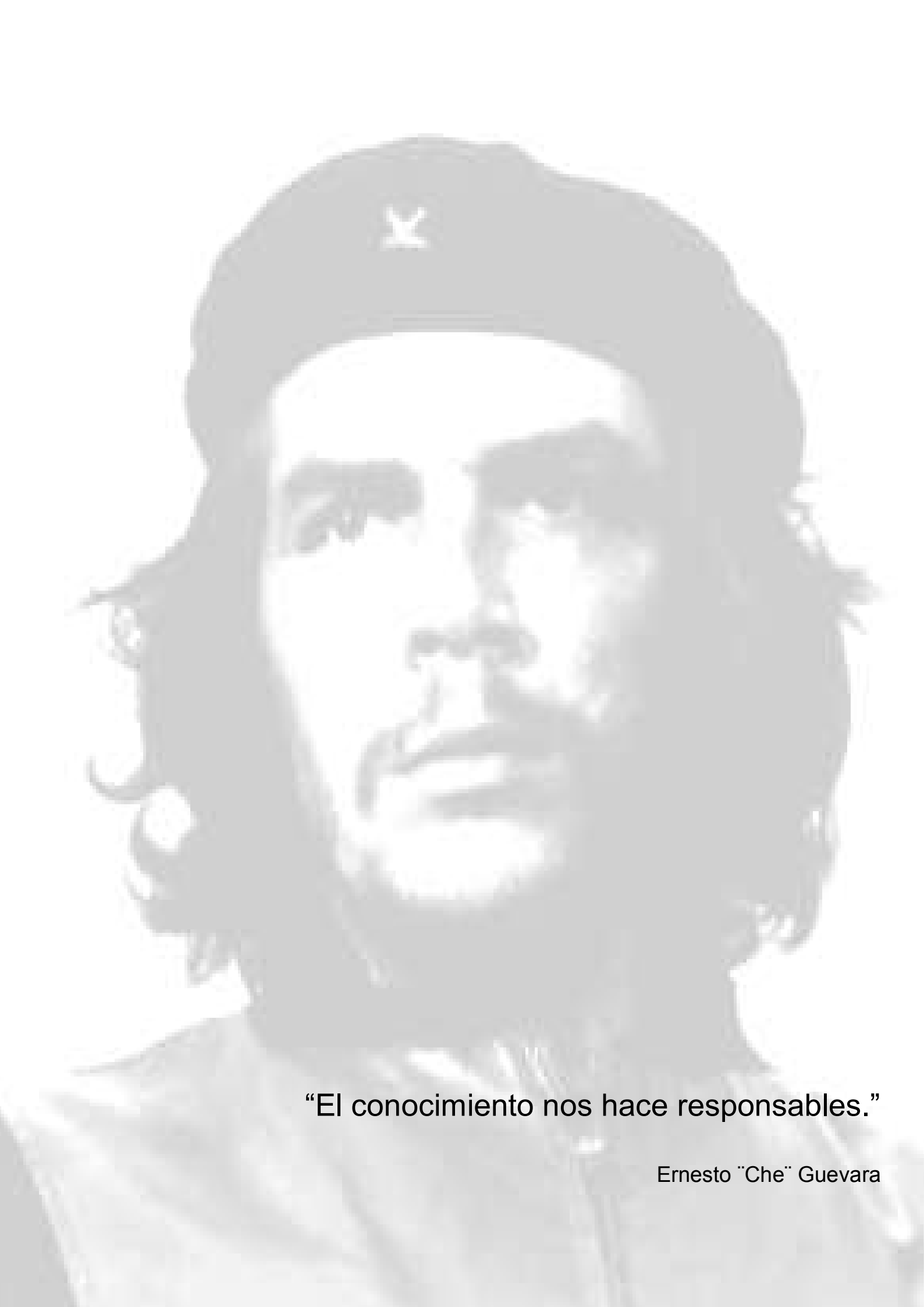
A continuación se exponen los elementos que amparan la afirmación anterior:

1. Implementación de nuevas funcionalidades en la plataforma gvSIG Mobile como la: cálculo de perímetro, distancia, y área; localización de edificios y personas.
2. Se realizó un correcto trabajo colaborativo con el especialista de cartografía del Centro
3. Se realizó correctamente la personalización sobre la Universidad de las Ciencias Informáticas adaptándola a las características socio-económicas de la misma.

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



“El conocimiento nos hace responsables.”

Ernesto “Che” Guevara

AGRADECIMIENTOS

A mis tutores por la ayuda que me brindaron en todo momento.

A los miembros del tribunal y a mi oponente por las críticas constructivas que me realizaron durante el desarrollo del trabajo.

A mis padres y amigos por brindarme el apoyo moral en los momentos más difíciles.

DEDICATORIA

A mi familia, en especial a mis padres por apoyarme durante todos estos años.

A mi novia por su apoyo y ayuda en todo momento.

RESUMEN

En la actualidad los Sistemas de Información Geográfica (SIG) han tomado un gran auge y su aplicación se ha llevado hasta diversas esferas de de la economía y la sociedad. Su aporte a la toma de decisiones constituye un elemento que hace incrementar su uso cada vez con mayor fuerza. El objetivo de este trabajo es desarrollar un SIG para dispositivos móviles que permita el acceso a la información geográfica de la Universidad de las Ciencias Informáticas (UCI), desde cualquier lugar de esta, sin necesidad de utilizar para ello un ordenador. Esta aplicación permite la visualización y el análisis de la información geográfica y socioeconómica de la UCI a partir de las capas que contiene el sistema y de los servicios web que brinda el repositorio de la Universidad. De este modo es posible apoyar la toma de decisiones de la comunidad universitaria.

PALABRAS CLAVE

Sistemas de Información Geográfica, GvSIG

SUMMARY

Today the Geographic Information Systems (GIS) have taken a major boom and its use has spread to various areas of the economy and society. Its contribution to the decision-making is one element that makes increasing it's use with increasing force. The aim of this work is to develop a mobile GIS that allows access to geographical information of the University of Information Sciences (UCI), from any place of it, without using a computer for it. This application allows the visualization and analysis of geographic and socioeconomic information from the ICU from the layers containing the system and from the web services provided by the repository of the University. This makes it possible to support the decision making of the university community.

KEY WORDS

Geographic Information Systems, GvSIG

Índice de Figuras

Figura # 1: Modelo de Clases del Dominio.	36
Figura # 2: Diagrama de Casos de Uso del Sistema.	43
Figura # 3: Diagrama de Clases del análisis CU_Calcular_Superficie.	52
Figura # 4: Diagrama de Clases del análisis CU_Localizar_Edificio.	52
Figura # 5: Diagrama de Colaboración CU_Calcular_Superficie.	53
Figura # 6: Diagrama de Colaboración CU_Localizar_Edificio.	53
Figura # 7: Diagrama de clases del diseño CU_Calcular_Superficie.	54
Figura # 8: Diagrama de clases del diseño CU_Localizar_Edificio.	55
Figura # 9: Interacción entre archivos que forman el shapefile.	57
Figura # 10: Diagrama de Despliegue.	59
Figura # 11: Diagrama de Componentes.	60

Índice de Tablas

Tabla # 1:	Requisito Funcional Cargar Aplicación.....	37
Tabla # 2:	Requisito Funcional Configurar aplicación.	37
Tabla # 3:	Requisito Funcional Gestionar Proyecto.	38
Tabla # 4:	Requisito Funcional Gestión de capas.	39
Tabla # 5:	Requisito Funcional Redimensionar mapa.....	39
Tabla # 6:	Requisito Funcional Calcular Superficie.....	39
Tabla # 7:	Requisito Funcional Tratar Información.....	40
Tabla # 8:	Requisito Funcional Localizar Edificio.....	40
Tabla # 9:	Requisito Funcional Localizar Persona.	41
Tabla # 10:	Descripción de los Actores.	42
Tabla # 11:	Descripción Textual del CU_Calcular_Superficie.....	46
Tabla # 12:	Descripción Textual del CU_Localizar_Edificio.....	51
Tabla # 13:	Secciones a probar en el CU_Calcular_Superficie.....	63
Tabla # 14:	Matriz de de Caso de Prueba para el CU_Calcular_Superficie.....	63
Tabla # 15:	Matriz de Datos para el CU_Calcular_Superficie.....	64
Tabla # 16:	Secciones a probar en el CU_Localizar_Edificio.....	67
Tabla # 17:	Descripción de variables del CU_Localizar_Edificio.....	68
Tabla # 18:	Matriz de de Caso de Prueba para el CU_Localizar_Edificio.....	69

Índice

DECLARACIÓN DE AUTORÍA.....	II
AVAL DEL PROYECTO.....	III
AGRADECIMIENTOS.....	V
DEDICATORIA	VI
RESUMEN.....	VII
PALABRAS CLAVE	VII
SUMMARY	VII
KEY WORDS.....	VII
Índice de Figuras	VIII
Índice de Tablas	IX
Índice	X
INTRODUCCIÓN.....	14
Capítulo 1: Fundamentación Teórica.....	17
Introducción.....	17
1.1. Conceptos asociados a los sistemas de información geográfica.	17
1.2. Situación problemática.....	19
1.3. Sistemas de información geográfica.	19
1.3.1. ¿Qué es un sistema de información geográfica?	19
1.3.2. Antecedentes históricos.....	20
1.3.3. Antecedentes del uso y desarrollo de los sistemas de información geográfica	

en Cuba.....	22
1.4. Análisis de las soluciones existentes.....	23
1.5. Herramientas y tecnologías para desarrollar sistemas de información geográfica. 25	
1.5.1. Metodología de desarrollo de software.....	26
1.5.1.1. Definición.....	26
1.5.1.2. Metodologías ágiles o livianas.....	26
1.5.1.3. Metodologías tradicionales o pesadas.....	28
1.5.2. Plataformas para el desarrollo de sistemas de información geográfica para dispositivos móviles.....	30
1.5.3. Lenguajes.....	32
1.5.3.1. Lenguajes de modelado.....	32
1.5.3.2. Lenguaje de programación.....	32
1.5.4. Herramientas de modelado.....	33
1.5.5. Entorno de desarrollo.....	33
1.5.6. Servicios web (WS):.....	33
1.6. Conclusiones parciales.....	34
Capítulo 2: Propuesta de solución.....	35
Introducción.....	35
2.1. Entorno donde trabajará el sistema.....	35
2.1.1. Conceptos y eventos principales.....	35
2.1.2. Diagrama de clases del modelo de dominio.....	36

2.1.3.	Glosario de términos del dominio.....	36
2.2.	Requisitos funcionales.....	37
2.3.	Requisitos no funcionales.....	41
2.4.	Descripción del sistema propuesto.....	42
2.4.1.	Descripción de los actores.....	42
2.4.2.	Diagrama de casos de uso del sistema.....	43
2.4.3.	Descripción textual de los casos de uso.....	43
2.5.	Conclusiones parciales.....	51
Capítulo 3:	Análisis y diseño de la solución.....	52
Introducción.....		52
3.1.	Modelo de análisis.....	52
3.1.1.	Diagramas de clases del análisis.....	52
3.1.2.	Diagramas de interacción.....	53
3.1.2.1.	Diagramas de colaboración.....	53
3.2.	Modelo de diseño.....	53
3.2.1.	Diagramas de clases de diseño.....	54
3.3.	Arquitectura.....	55
3.4.	Modelo de datos.....	56
3.4.1.	Formato shapefile:.....	56
3.4.2.	Servicios web utilizados:.....	57
3.5.	Conclusiones parciales.....	58

Capítulo 4: Implementación y prueba	59
4.1. Implementación	59
4.1.1. Modelo de despliegue	59
4.1.2. Diagrama de componentes	60
4.2. Plan de pruebas del sistema	60
4.2.1. Objetivo	60
4.2.2. Nivel de prueba	60
4.2.3. Método utilizado	61
4.2.4. Técnica utilizada	61
4.2.5. Diseño de casos de prueba	61
4.2.5.1. CU_Calcular_Superficie	61
4.2.5.2. CU_Localizar_Edificio	65
4.3. Conclusiones parciales	70
Conclusiones generales	71
Recomendaciones	72
Glosario de Términos	79

INTRODUCCIÓN

La Universidad de las Ciencias Informáticas (UCI) que constituye no sólo un centro docente sino también productivo, desarrolla en sus facultades diferentes líneas de producción e investigación. Las temáticas relacionadas con los sistemas para el manejo de información geográfica son investigadas y desarrolladas por uno de los departamentos del Centro de Geoinformática y Señales Digitales (GEYSED) que pertenece a la facultad 6 de esta institución. Como parte del trabajo realizado para estos fines se han desarrollado diferentes aplicaciones con variados objetivos. El producto más completo que se ha llevado a cabo atendiendo a la complejidad de las operaciones que realiza y a la utilización que a nivel mundial se hace de este tipo de sistemas en la actualidad es GeneSIG, plataforma base para crear Sistemas de Información Geográfica (SIG), primera y única de su tipo en el país. La UCI como principal creadora del sistema realizó la primera personalización de este, dando como resultado el Sistema de Información Geográfica de la Universidad de las Ciencias Informáticas (SIGUCI).

A nivel mundial existen numerosas soluciones que realizan funciones similares, entre ellas sobresale el caso de ArcGIS, conjunto de productos de *software* en el campo de los SIG, producido y comercializado por el Instituto de Investigaciones de Sistemas Ambientales (*Environmental Systems Research Institute*, ESRI). Bajo el nombre genérico ArcGIS, se agrupan varias aplicaciones para la captura, edición, análisis, tratamiento, diseño, publicación e impresión de información geográfica,(1). Sin embargo, sería muy costoso el pago de la licencia para su uso.

Hoy en día la telefonía celular ha alcanzado un gran desarrollo y casi a diario aparecen nuevas tecnologías que hacen que estos pequeños dispositivos ganen en capacidad y rapidez permitiendo que se ejecuten en ellos aplicaciones cada vez más complejas. Estas capacidades han sido aprovechadas para integrar las posibilidades que brindan estos dispositivos con las de los SIG y actualmente ya existen algunas soluciones SIG móviles. Por tal razón y atendiendo al hecho de que GeneSIG no es accesible desde dispositivos móviles, en enero de 2010, a partir de la reorganización de las estructuras productivas de la universidad en centros de desarrollo asociados a las facultades, surge un proyecto conjunto entre los centros GEYSED y el Centro de Desarrollo de Sistemas y Servicios Telemáticos (SITEL) de la facultad 2, con el objetivo de crear soluciones SIG para dispositivos móviles.

Por tales motivos y atendiendo a que el SIGUCI no es accesible desde dispositivos móviles, se plantea el siguiente **problema a resolver**: La inaccesibilidad desde dispositivos móviles a la información geográfica de la UCI aumenta el esfuerzo en la toma de decisiones por parte de la comunidad universitaria. Este se enmarca en el **objeto de estudio** sistemas de información

geográfica. El **Objetivo general** de la investigación es desarrollar una aplicación SIGUCI que permita el acceso a la información geográfica de la UCI desde dispositivos móviles para disminuir el esfuerzo en la toma de decisiones.

Objetivos específicos:

- Elaborar el marco teórico-conceptual de la investigación.
- Implementar la aplicación.
- Aplicar pruebas funcionales al sistema.

El campo de acción se encuentra enmarcado en: los sistemas de información geográfica de escritorio para dispositivos móviles. Para el desarrollo de la presente investigación se plantea como **Idea a defender**: si se desarrolla una aplicación que garantice el acceso a la información geográfica de la UCI desde dispositivos móviles se disminuirá el esfuerzo en la toma de decisiones por parte de la comunidad universitaria.

Para lograr los objetivos trazados se han planteado las siguientes **tareas**:

- Fundamentar tendencias actuales, tecnologías, herramientas y conceptos más importantes relacionados con los SIG.
- Identificar las principales herramientas y estándares que se utilizan para la creación de SIG de escritorio para dispositivos móviles.
- Definir requisitos funcionales y no funcionales que debe cumplir la aplicación.
- Identificar los casos de uso del sistema.
- Describir los casos de uso del sistema.
- Elaborar el diagrama de casos de uso del sistema.
- Confeccionar los diagramas de clases del diseño para cada caso de uso.
- Programar los casos de uso diseñados
- Diseñar los casos de prueba.
- Probar las funcionalidades de la aplicación.

Para desarrollar la investigación se utilizarán **métodos científicos** teóricos: el analítico-sintético para analizar las teorías y documentos relacionados con los SIG, el análisis histórico-lógico para aprovechar los resultados arrojados por el análisis de los antecedentes en la elaboración de la nueva solución.

Resultados esperados.

Con el desarrollo de este trabajo se espera obtener un sistema de información geográfica que funcione en dispositivos móviles y que permita a los usuarios el acceso permanente a los datos geográficos de la UCI.

Capítulo 1: Fundamentación Teórica.

Introducción.

En este capítulo se abordan los conceptos generales, herramientas, metodologías, tendencias y tecnologías relacionadas con el proceso de desarrollo de los SIG, para seleccionar las más adecuadas para el desarrollo del sistema propuesto. Además se realiza un análisis de algunos de los sistemas existentes en la actualidad.

1.1. Conceptos asociados a los sistemas de información geográfica.

❖ Datos geográficos.

Según ESRI, se le llama datos geográficos a “la información que describe la ubicación y los atributos de los objetos incluyendo sus formas y las representaciones. Es la combinación de datos espaciales y sus atributos”,(2).

Por lo que se concluye que son todos los datos presentes en un mapa.

❖ Datos espaciales.

“La información sobre los lugares, elementos geográficos y las relaciones entre ellos, normalmente se almacena como coordenadas y topologías”, (2), esta es a juicio de ESRI la definición más apropiada para datos espaciales.

Según un artículo publicado por la Universidad de Michigan, los datos espaciales son las características que representan un lugar en el mapa. Las coordenadas de un punto, el sistema de proyección, los atributos de línea y polígono, constituyen ejemplos de estos, (3).

Los datos espaciales también se pueden encontrar en un mapa al igual que los datos geográficos, con la única diferencia que datos espaciales son aquellos que muestran alguna característica espacial de un objeto, es decir, ubicación en el espacio, proyección, etc.

❖ Capas.

División o estrato de la realidad geográfica en una zona determinada, es la forma de representar visualmente conjuntos de datos geográficos en el entorno de un mapa digital. Una capa es el equivalente a un elemento de la leyenda de un mapa de papel,(2). Las capas de datos

geográficos pueden ser de dos tipos de formato:

- **Formato de datos raster.**

Define el espacio como una matriz de celdas de igual tamaño, se disponen en filas y columnas. Cada celda contiene un valor de atributo y una coordenada de ubicación. Los grupos de celdas que contienen el mismo valor representan el mismo tipo de accidente geográfico, (2).

Son mapas de bits que cubren un área de interés. Cada *pixel*, la unidad más pequeña de información, muestra un atributo único,(4).

La información en los formatos de datos raster está almacenada directamente en los *pixel* de las capas, por lo que la información que brinda es mucho más detallada.

- **Formato de datos vectoriales.**

Las características geográficas como puntos, líneas y polígonos se representan por coordenadas. Cada entidad (punto) está constituida por un solo par de coordenadas, mientras que las funciones de línea y polígono se simbolizan como listas ordenadas de vértices, (2).

Los formatos de datos vectoriales tienen magnitud y dirección. Son una forma de codificar líneas e información de área en forma de unidades de datos que expresan magnitud, dirección y conectividad. Un archivo vectorial por lo general tiene un conjunto definido de elementos (puntos, líneas, texto, elipses, formas, curvas, elementos complejos agrupados) que se pueden utilizar, (5).

La unidad de medida más pequeña en este tipo de datos son los puntos, los cuales tienen asociados un par de coordenadas; para la representación de líneas se utilizan listas de vértices.

- ❖ **Referenciación geoespacial.**

Según la Universidad de Columbia, la referenciación geoespacial “es el proceso de alineación de datos espaciales a un archivo de imagen como un mapa histórico, imagen de satélite o fotografías aéreas”,(6).

A criterio de la Agencia de Protección Ambiental de los Estados Unidos (EPA) la referenciación geoespacial “describe el proceso de localización de una entidad en el mundo real mediante coordenadas”,(7).

Según ESRI la referenciación geoespacial es “la alineación de los datos geográficos a un sistema de coordenadas conocido para que se puedan visualizar, consultar y analizar. Puede implicar un desplazamiento, rotación y en algunos casos deformación de los datos”, (2).

La referenciación geoespacial es el proceso mediante el cual se le asignan datos espaciales a un objeto del mapa permitiendo luego analizarlos y llegar a soluciones lo más exactas posible.

❖ Mapa.

Según ESRI un mapa no es más que “una representación gráfica de las relaciones espaciales de las entidades dentro de un área (...) Contiene una o más capas de datos geográficos, que figura en tramas de datos, y varios elementos de soporte del mapa, como una barra de escala”, (2).

Por lo que se puede llegar a la conclusión de que un mapa es la representación gráfica de datos espaciales y geográficos de un área determinada.

❖ Cartografía.

La cartografía es la ciencia que se encarga de expresar gráficamente a través de los mapas las características naturales y sociales de la tierra, (2).

1.2. Situación problemática.

Dado el auge que hoy en día han tenido los SIG y por la importancia que tienen estos en la toma de decisiones, la UCI en colaboración con el centro UCID y con Geocuba comenzó el desarrollo de este tipo de aplicaciones. De esta unión surgió la plataforma soberana para la creación de SIG web GeneSIG.

La UCI como principal desarrollador de dicha plataforma tuvo la oportunidad de contar con la primera personalización de esta. Actualmente la única forma de acceder al SIGUCI es mediante una computadora personal (*Personal Computer*, PC), lo que propicia que al no disponer en todo momento de un ordenador, no sea posible consultar la aplicación, provocando que la comunidad universitaria en ocasiones realice un trabajo excesivo a la hora de desplazarse de un lugar a otro o que en ocasiones no conozca con exactitud el lugar a donde se dirige.

Por esta razón, actualmente es necesaria la creación de un SIGUCI que funcione en dispositivos móviles y que brinde la posibilidad a la comunidad de acceder a la cartografía de la universidad en cualquier momento.

1.3. Sistemas de información geográfica.

1.3.1. ¿Qué es un sistema de información geográfica?

Es difícil emitir un significado exacto de lo que es un SIG ya que la mayoría de los autores tiene su propio concepto basado en diferentes criterios, algunos definen el término basándose meramente

en el elemento *software*, de esta manera se pueden encontrar definiciones como:

Software utilizado para automatizar, analizar y representar datos gráficos referenciados geográficamente y organizados según un modelo topológico, (11).

Sistema computarizado que provee los siguientes cuatro conjuntos de operaciones para tratar datos referenciados geográficamente: 1) entrada de datos; 2) uso de los datos (almacenamiento y recuperación); 3) manipulación y análisis; 4) salida,(12).

Estas definiciones, aunque no del todo mal, dejan un amplio margen para que algunos sistemas sean considerados como SIG sin realmente serlo, por lo que otros autores han introducido además en el significado, las funcionalidades básicas que deben cumplir. Así es posible encontrar algunas definiciones como:

“Conjunto de herramientas para reunir, introducir, almacenar, recuperar, transformar y cartografiar datos espaciales sobre el mundo real para un conjunto particular de objetivos”,(13).

“Sistema para capturar, almacenar, validar, integrar, manipular, analizar y representar datos referenciados sobre la Tierra”,(14).

Aunque los SIG realicen las funciones mencionadas con anterioridad estos conceptos se alejan de una definición exacta, pues no contienen el elemento *hardware* que también forma parte de ellos, y sin este los SIG fueran incapaces de realizar las funciones para los que fueron creados, por lo que muchos estudiosos del tema lo incluyen en algunos conceptos, así se encuentran definiciones como:

“Un sistema de *hardware*, *software* y procedimientos elaborados, para facilitar la obtención, gestión, manipulación, análisis, modelado, representación y salida de datos espacialmente referenciados para resolver problemas complejos de planificación y gestión”,(15).

“Conjunto integrado de medios y métodos informáticos, capaz de recoger, verificar, intercambiar, almacenar, gestionar, actualizar, manipular, recuperar, transformar, analizar y mostrar datos referenciados geográficamente a la Tierra”,(16).

Estas últimas definiciones son las que reflejan de forma más completa el concepto de SIG pues abordan todos los aspectos que conforman un sistema de este tipo.

1.3.2. Antecedentes históricos.

Los SIG tienen sus antecedentes en el surgimiento de la humanidad evidenciándose en las pinturas rupestres que realizaron los primeros habitantes, aunque algunos autores no lo

consideran así, la gran mayoría converge a este criterio. Si bien no se puede realizar una comparación de los primeros utilizados por los hombres con los actuales, sí se puede decir que se ha utilizado desde tiempos remotos. En el año 1854 el Dr. John Snow, cartografió los casos de cólera en un distrito de Londres, lo que le permitió localizar con gran precisión un pozo de agua contaminado como fuente del brote. A principios del siglo XX se comienzan a recolectar imágenes aéreas, en el año 1903, con cámaras fotográficas ajustadas a las palomas y en 1905 se utiliza por primera vez un avión para fotografías aéreas,(17).

El primer SIG reconocido fue el llamado Sistema de Información Geográfica de Canadá (*Canadian Geographic Information System*, CGIS) su desarrollo se inició en 1964 en el Departamento Federal de Silvicultura y Desarrollo Rural por Roger Tomlinson y comenzó a brindar servicio desde 1967.

Aunque el término en sí es bastante reciente, algunos especialistas en el tema definen el desarrollo de los SIG en cuatro etapas:

Primera etapa entre 1966 y 1970: Se plantean los primeros intentos de desarrollo de dichos sistemas. En esta etapa surgen algunos como:

- Sistema de Información de Recursos Naturales y uso de la Tierra (*Land Use and Natural Resources Information System*, LUNR), New York, 1967.
- Sistema de Gestión de Información de Tierra de Minnesota (*Minnesota Land Management Information System*, MLMIS), Minnesota, 1969.

Segunda etapa entre 1970 y 1980: Está relacionada con el impulso tecnológico de los años setenta. En esta etapa surgen programas como el POLYVRT que introduce la tipología de objetos cartográficos. Los SIG que más sobresalen en esta etapa son:

- Sistema de Información de Superposición de Polígonos (*Polygon Information Overlay System*, PIOS), en 1971.

Tercera etapa entre 1980 y 1997: Está enmarcada en el desarrollo industrial de los SIG y su adquisición por un gran número de empresas para la gestión de información territorial. Fue el comienzo de la difusión de estos sistemas. En esta etapa surge el primer gestor verdadero de información geográfica, el *software* Odyssey en el que se incluye la digitalización semiautomática de datos espaciales, la gestión de bases de datos y la elaboración interactiva de los datos. Durante estos años la empresa ESRI desarrolla ARC/INFO para el trabajo con datos vectoriales, además comienzan a desarrollarse programas como IDRISI y ERDAS para el trabajo con

formatos de datos *raster*.

Cuarta etapa a partir de 1997: Los SIG logran un gran desarrollo y expansión debido fundamentalmente al avance alcanzado por las computadoras, los lenguajes de programación, el tratamiento de gráficos, el acceso masivo a los programas, datos y al abaratamiento de la tecnología. También en esta etapa comienzan a surgir los primeros SIG distribuidos bajo licencias libres, (18),(19).

1.3.3. Antecedentes del uso y desarrollo de los sistemas de información geográfica en Cuba.

La utilización de los SIG en Cuba se remonta al año 1987 cuando se comenzó a usar el Paquete de Análisis de Mapas (*MAP ANALYSIS PACKAGE*) en el Instituto de Geografía Tropical de Cuba para digitalizar el mapa de riesgos de inundaciones. Este sistema fue desarrollado para Computadoras Personales IBM por la Universidad de *Ohio* a partir de la versión *PC-MAP* de la Universidad de *Harvard*, basado en los trabajos de Dana Tomlin. Para realizar este trabajo y ante la imposibilidad de obtener SIG comerciales, los especialistas cubanos tuvieron que implementar el módulo Digicapt el cual tenía como propósito actualizar el nuevo Atlas Nacional de Cuba tomándolo como fuente inicial de datos. A este sistema se le integraron otros 14 módulos como el *PC-Sace* que permitía introducir cartodiagramas y gráficos dentro de los mapas, variante desarrollada por expertos cubanos antes del *ArcView* de la Compañía ESRI, así surgió el primer SIG cubano que muchos llamaron SIGC, el cual tenía una compleja estructura.

En la década de los noventa con el auge de las computadoras personales llega a Cuba otro SIG, el IDRISI que no tuvo tanta aceptación como el anterior ya que la entrada de datos resultaba compleja, ante la carencia de tabletas digitalizadoras en las instituciones fue poco accesible por los especialistas que intentaron utilizarlo, en esto influyó también la introducción de la versión 2.0 de Tosca (módulo de digitalización) en 1993.

Es esos tiempos la Universidad de La Habana recibía la donación del SIG LWIS desarrollado por el Instituto de Estudios Aeroespaciales y Ciencias de la Tierra (*Institute for Aerospace Survey and Earth Sciences, ITC*) de Holanda. Este tenía la capacidad de trabajar con imágenes y mapas vectoriales, así como datos tabulados de atributos. La captación de imágenes posibilitó el monitoreo de algunas investigaciones sobre todo en regiones donde no existían suficientes datos de observación del terreno. La principal dificultad del sistema radicaba en que las versiones para MS-DOS utilizaban dos monitores para trabajar, además no eran compatibles con otros SIG

comerciales.

El ATLAS GIS desarrollado por *Strategic Mapping* de California, Estados Unidos, fue el primer SIG de formato digital utilizado en Cuba, llegó a mediados de los noventa desde México. Se utilizó durante varios años en el Instituto de Geografía Tropical y en otros organismos para elaborar mapas temáticos. Este sistema cuenta con un módulo de digitalización que aún se utiliza para llevar los archivos digitalizados a otros sistemas como *AGISW*, *MapInfo* y *ArcView*. La versión 3.10 se utilizó muy poco en Cuba ya que también a mediados de los años noventa se había introducido el *MapInfo*, que aunque no era potente era muy fácil de manejar.

En la década de los noventa surgió TeleMap, el único SIG diseñado, creado y producido en Cuba por los especialistas del Instituto Cubano de Hidrografía. Era un módulo de análisis que integraba un *software* para las geociencias y se le denominó Telemap/GIS. Al principio fue muy utilizado pues con él se lograba convertir formatos *raster* a vectorial y viceversa, además de permitir la unión exacta de las imágenes lo cual fue un logro para los cubanos. Este SIG tenía como inconveniente que necesitaba una llave de usuario para operar y que no era compatible con otros SIG.

Otro SIG ampliamente utilizado en Cuba ha sido el *ArcView*. Este es un sistema vectorial que permite incorporar imágenes y trabajar sobre ellas. Los módulos de análisis del *ArcView* constituyen herramientas que permiten utilizarlo en cualquier rama de las ciencias naturales y medioambientales.

Ya a comienzos del nuevo milenio se introdujo en Cuba el LatinoGIS por parte de la empresa Cesigma S. A., compañía cubano-española, especializada en soluciones integrales geográficas y medio ambientales, junto al grupo español CADIC, S.A. Este SIG se basa en tecnología .NET lo que potencia sus funcionalidades,(20).

A partir del año 2008 especialistas de Geocuba, las FAR y estudiantes y profesores de la UCI comienzan a trabajar en lo que sería la primera plataforma soberana SIG web cubana basada en *software* libre llamada en un principio LiberGIS la cual se rebautizó luego con el nombre de GeneSIG.

1.4. Análisis de las soluciones existentes.

Con el objetivo de identificar las principales funcionalidades con que debe contar la aplicación y las posibles herramientas a utilizar durante el proceso de desarrollo, se realiza un análisis de las soluciones existentes en distintas áreas del conocimiento a nivel nacional e internacional.

Sistema de Información Geográfica de la Universidad de Alicante (SIGUA): Es un sistema que ha sido diseñado haciendo uso de programas y librerías de código abierto. Como gestor de bases de datos se utiliza PostgreSQL con su extensión PostGIS, como servidor de mapas se utiliza MapServer. El acceso a la información Geográfica y las tareas de mantenimiento de la aplicación se realizan mediante una interfaz web utilizando como lenguaje PHP y un servidor Apache. Esta aplicación hace uso de los servicios web para resolver diferentes demandas de información. Cuenta con un programa de escritorio que se encarga de administrar los diferentes servicios que se les brindan a los usuarios dependiendo de sus privilegios. Las funcionalidades que brinda dicha aplicación son: mostrar cartografía, mapa de referencia, navegación en mapa dinámico, información de un objeto, vista rápida de un objeto, control de plantas, exportar a PDF, localización de personas y estancias, leyenda, informes especializados, etc. Aunque esta aplicación está desarrollada para la web es accesible desde dispositivos móviles,(21).

Sistema de Información Geográfica de la Consejería de Cultura y Turismo de la Junta de Extremadura (SIGATEX): Este sistema se realizó con el objetivo de difundir los datos sobre los recursos e infraestructuras turísticas de Extremadura. Está desarrollado siguiendo la política de software libre adoptado por la Junta de este lugar. Actualmente el sistema cuenta con más de sesenta capas temáticas y más de 20000 registros. Como gestor de base de datos se utiliza PostgreSQL con su extensión PostGIS. Para la representación y actualización de la información geográfica utiliza el cliente GvSIG integrando además el acceso desde GvSIG Mobile y un cliente ligero basado en Open Layers, de esta forma el usuario puede generar rutas turísticas tanto en la web como en su móvil,(22).

Sistema inteligente de guiado turístico por móvil (Movípolis): Ofrece un servicio avanzado de información y guía turística a todos los visitantes de una localidad o entorno con patrimonio histórico-artístico. Proporciona servicios avanzados de información como visita guiada dirigida, información multimedia, consejos para aprovechamiento de la estancia, información relevante sobre eventos y circunstancias especiales, direcciones y orientación, alojamiento, transportes y comunicaciones, restaurantes, bares, tiendas, datos básicos como horarios, cajeros, aseos, etc., (23), (24).

Sistema de Información Geográfica para la gestión de la estadística de salud de Cuba (SIG-ESAC): Este sistema fue desarrollado con el objetivo de facilitar la gestión de las estadísticas de salud desde los municipios hasta nivel nacional. Está desarrollado utilizando la herramienta SIG ArcView y para su visualización se utiliza ArcExplore. Este sistema permite mediante cuatro

menús cartografiar y realizar análisis de diferentes indicadores de salud como: morbilidad, mortalidad, demográficos, recursos y servicios. La base de datos de este sistema está disponible en los formatos ASCII (.TXT) y DBASE (.DBF), utilizados por la mayoría de los SIG y gestores de bases de datos comerciales. La base cartográfica fue editada por GEOCUBA usando el formato Shapefile (*.shp). Cuenta con las funcionalidades básicas de navegación y brinda la posibilidad de aplicar un grupo de técnicas epidemiológicas, tales como análisis de *clúster*, métodos bayesianos, cálculos de tasas brutas y ajustadas e interpolaciones,(25).

Sistema de Información Geográfica de la Universidad de las Ciencias Informáticas (SIGUCI): Se encuentra desarrollado sobre la plataforma GeneSIG. Entre sus funciones están la de proporcionar acceso a la información geográfica de la UCI, permitir que los usuarios realicen análisis y consultas de los datos asociados a esta e integrar parte de la información socioeconómica de la UCI con su información geográfica. Brinda un visor web que permite el acceso a los datos y servicios como la visualización de consultas realizadas por los usuarios a la base de datos con las referencias geográficas de los objetos identificados. Ofrece un servicio de catálogos de mapa para el resguardo de los procesos de edición personalizada y un grupo de herramientas para la administración de los datos y la configuración del sistema. Además cuenta con módulos de Análisis, Selección, Navegación, Consulta, Visualización, Servicios, Exportar PDF, Edición y Estructura,(26), (27). Actualmente este sistema no es accesible desde dispositivos móviles, por lo que se hace necesario contar con un ordenador para su utilización.

Luego de analizar algunas de las soluciones existentes se ha llegado a la conclusión que la aplicación debe proporcionar un visor que permita mostrar la cartografía, permitir gestionar proyectos y capas, proporcionar herramientas de navegación, análisis y consulta de información. Para la consulta de la información es necesaria la utilización de servicios web, los cuales se brindan mediante el UDDI de la UCI.

1.5. Herramientas y tecnologías para desarrollar sistemas de información geográfica.

Durante el desarrollo de cualquier aplicación informática se hace necesaria la utilización de herramientas y tecnologías que faciliten el proceso. Seguidamente se tratarán las más utilizadas en el mundo de los SIG y se definirán cuáles son las más convenientes para el desarrollo de la aplicación propuesta.

1.5.1. Metodología de desarrollo de software.

1.5.1.1. Definición.

Para llevar a cabo el proceso de desarrollo de un *software* determinado, se debe utilizar un conjunto de conceptos, herramientas, y tecnologías que influirán en la calidad del producto final y en el cumplimiento de los requisitos solicitados por el cliente. Para realizar las tareas que se definen en el proceso, se debe seguir un conjunto de pasos con el fin de llevarlo a cabo de la forma más eficiente posible tanto para el equipo de desarrollo como para el cliente, a este conjunto de pasos se le denomina metodología de desarrollo de *software*. A continuación se especifican algunas definiciones:

“Las metodologías de desarrollo de *software* son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de *software*”,(28).

Las metodologías de desarrollo de *software* definen quién está haciendo qué, cuándo y cómo para alcanzar un determinado objetivo,(29).

Basándose en estas dos posibles definiciones, se puede llegar a la conclusión de que las metodologías de desarrollo de *software* definen las tareas que debe realizar cada miembro del equipo de desarrollo, cuándo las realizará y cómo, para lograr así un *software* que cumpla con las condiciones acordadas.

En la actualidad existen muchas metodologías de desarrollo de *software* con características diferentes. Cumpliendo sus requisitos durante el proceso de desarrollo, se puede lograr un producto de calidad, todo depende de realizar un análisis previo para definir cuál se ajusta mejor a las necesidades de nuestro producto. Estas metodologías se clasifican en dos tipos: las llamadas metodologías ágiles y las convencionales.

1.5.1.2. Metodologías ágiles o livianas.

Las metodologías ágiles realizan un cambio significativo en las características de las metodologías tradicionales, están guiadas principalmente por las siguientes características:

- Están basadas en heurísticas provenientes de prácticas de producción de código anteriores.
- Están preparadas especialmente para aceptar cambios durante el proyecto sin afectar lo antes realizado.
- Son elegidas internamente por parte del equipo de desarrollo.

- Son procesos pocos controlados.
- Los contratos en el caso de que existan son bastante flexibles.
- El cliente es parte del equipo de desarrollo formando parte activa de este.
- Se utilizan pequeños grupos de trabajo (menos de diez integrantes).
- Durante el desarrollo se generan pocos artefactos.
- Existen pocos roles en el equipo de desarrollo.
- Se hace menos énfasis en la arquitectura durante el desarrollo.

Estas eliminan gran parte de la documentación generada, es decir, se orientan al código, basándose en la idea de que la parte importante de la documentación es el código fuente, (30).

De este tipo de metodología las más utilizadas son:

Extreme programming (XP): Esta metodología de desarrollo se centra principalmente en las relaciones interpersonales como clave del éxito en el desarrollo, promueve el trabajo en equipo, brindándole una importancia primordial al aprendizaje de los desarrolladores, propiciando así un buen clima de trabajo. También es muy importante que la comunicación entre cliente y equipo de desarrollo fluya sin contratiempo, lo que permite una retroalimentación continua. XP además simplifica las soluciones. Esta metodología es adecuada para proyectos de alto riesgo técnico en los cuales los requisitos son imprecisos, (31).

SCRUM: Esta metodología se ha utilizado durante los últimos años con grandes éxitos. Está desarrollada principalmente para proyectos en los que existen cambios de requisitos de forma rápida. Se caracteriza principalmente por realizar iteraciones con una duración de treinta días y cuyo resultado se le presenta al cliente. La segunda característica importante son las reuniones que se realizan diariamente en el seno del equipo de desarrollo con el objetivo de coordinar e integrar, (31).

Dynamic systems development method (DSDM): Se basa en un proceso iterativo e incremental donde el equipo de desarrollo y el cliente trabajan de la mano. En esta metodología se proponen cinco fases: viabilidad, negocio, modelado funcional, diseño y construcción, y finalmente implementación, donde solo las tres últimas son iterativas. Además exige una retroalimentación en todas ellas, (31).

Crystal methodologies: Este es un conjunto de metodologías de desarrollo que se centran principalmente en la fortaleza del equipo de trabajo y la reducción al máximo de los artefactos generados durante el proceso de desarrollo de *software*. Estas metodologías crean políticas de

trabajo dependiendo del tamaño del equipo de desarrollo. Aquí se considera el desarrollo de *software* como un juego corporativo de invención y comunicación que se ve limitado por los recursos que se utilizan, (31).

No se cree conveniente la utilización de estas metodologías debido a que el proyecto a realizar es una adaptación para dispositivos móviles de un sistema ya desarrollado y no se cuenta con un cliente, el cual juega un papel decisivo en este tipo de metodologías. Además se pretende contar con la mayor cantidad posible de documentación para el desarrollo de futuras versiones.

1.5.1.3. Metodologías tradicionales o pesadas.

Estas metodologías son especialmente criticadas debido a la gran cantidad de documentación que generan. Se basan principalmente en las siguientes características:

- Fundamentadas en normas provenientes de estándares seguidos por el entorno de desarrollo
- Cierta resistencia a los cambios
- Impuestas externamente
- Proceso mucho más controlado, con numerosas políticas/normas
- Existe un contrato prefijado
- El cliente interactúa con el equipo de desarrollo mediante reuniones
- Grupos grandes y posiblemente distribuidos
- Abundantes artefactos y roles
- La arquitectura del *software* es esencial y se expresa mediante modelos, (31).

Estas metodologías generan una gran cantidad de documentación. En ellas se hace especial énfasis en la planificación total de todo el trabajo y una vez que está listo se comienza el proceso de desarrollo. También se definen rigurosamente roles, actividades, artefactos, herramientas, notaciones para modelado y documentación detallada. A diferencia de las ágiles no están preparadas para recibir cambios, por lo que no son métodos adecuados cuando se trabaja en un ambiente donde los requisitos varían con facilidad, (32).

Entre las más utilizadas están:

Microsoft solutions framework (MSF): Esta metodología brinda a los desarrolladores prácticas probadas para la planificación, construcción y despliegue de una gran variedad de soluciones tecnológicas, combinando aspectos del diseño y el desarrollo de aplicaciones, así como la

creación y el despliegue de las infraestructuras del ciclo de vida de dichas soluciones. Esta metodología ayuda a cumplir los compromisos, la velocidad y reducir al mínimo los riesgos de una forma equilibrada. En esta metodología se dejan en segundo plano las elecciones tecnológicas, concentrando los mayores esfuerzos en los modelos de proceso y de equipos. Las herramientas que se utilizan para desarrollar todos sus componentes son propiedad de *Microsoft*, algo que encarece su utilización, (33).

Proceso unificado de desarrollo de software (RUP): es más que un simple proceso; es un marco de trabajo genérico que puede adaptarse a una gran variedad de sistemas para diferentes áreas, tipos de organización, niveles de aptitud y tamaños de proyectos. Está basado en componentes por lo que el sistema en desarrollo está formado por componentes interconectados por medio de interfaces. Las tres características fundamentales en que se basa RUP son, (34):

Iterativo e incremental: Son pequeños proyectos que van incrementando funcionalidades a la aplicación y su desarrollo es considerado como una iteración, dando como resultado una versión estable de la aplicación sin tenerla aún terminada. Esto permite:

- Que se reduzcan los riesgos de no obtener el producto en el tiempo previsto y con la calidad requerida.
- Atacar problemas con requisitos incompletos.
- Que la versión final del producto sea lo más robusta posible, (35).

Dirigido por los casos de uso: Un caso de uso es un fragmento de funcionalidad que facilita al usuario un resultado importante. Estos aunque guían el proceso no se desarrollan aisladamente, sino que van a la par de la arquitectura del sistema, es decir, que los casos de uso guían la arquitectura y esta a su vez influye en la selección de estos. Esto posibilita que:

- Se brinda a los actores los servicios del sistema y se les proporciona un resultado.
- Se proveen los requisitos funcionales del sistema.
- Se describen las funcionalidades.
- Ofrece facilidad en la detección y solución de cambios en los requisitos de un caso de uso, así como las clases y componentes que afectan, (36).

Centrado en la arquitectura: La arquitectura es la encargada de proporcionar una imagen completa del sistema antes que se comience su desarrollo, esto permite que se tomen todas las decisiones técnicas antes de comenzar el mismo. Por lo tanto la arquitectura:

- Define la forma del sistema
- Los casos de uso describen la funcionalidad del sistema.
- Describe mediante vistas que se incorporan los casos de uso más relevantes, (36).

La utilización de RUP no requiere de pago alguno y a diferencia de MSF se encuentran algunas herramientas libres para generar los componentes necesarios durante el proceso de desarrollo. Aunque esta metodología en muchos casos genera excesiva documentación, en este caso como ya se mencionaba, constituye una ventaja pues se pretende documentar detalladamente las principales funcionalidades con que contará el sistema para realizar futuras versiones. Por estas razones y por contar con mayor experiencia en el trabajo con dicha metodología se selecciona la misma para ser utilizada en el presente trabajo.

1.5.2. Plataformas para el desarrollo de sistemas de información geográfica para dispositivos móviles.

Como se planteó anteriormente los SIG hoy en día son una importante ayuda en la toma de muchas decisiones en numerosas ramas, tanto de la vida económica, política o social en cualquier lugar del mundo y mientras más exactos sean los datos con los que se cuenta, mejor serán las decisiones. Esta razón unida al desarrollo acelerado de las telecomunicaciones y el aumento y mejora de las capacidades de los dispositivos móviles, permitió que las plataformas para desarrollar SIG más distribuidas del mundo cuenten con extensiones para dichos equipos. En este mercado sobresalen aplicaciones como:

Mobile SIG: es una familia de aplicaciones que fueron creadas por ESRI y que permiten la captura, almacenamiento, actualización, manipulación, análisis y representación de la información geográfica directamente desde el terreno. Es la extensión de un SIG de la oficina al campo y es una importante herramienta para lograr un SIG empresarial completo. *Mobile SIG* incluye: *ArcGIS Mobile*, *ArcPad*, y *Mobile ArcGIS Desktop Systems*, (37).

IntelliWhere OnDemand: Creada por *GeoMedia*, esta herramienta permite el traslado de cartografías y datos activos desde cualquier sistema corporativo hasta el terreno, permitiendo además trabajar en modo conectado, pudiendo así actualizar los datos en los servidores de

manera casi instantánea. Los usuarios de *IntelliWhere* pueden consultar, visualizar y señalar mapas, diagramas e información activa y cargar después estas actualizaciones en el sistema de información corporativa. Aunque está hecha especialmente para gobiernos, transportes, empresas de servicios, ejército y medio ambiente, esta herramienta está ampliamente distribuida en todo el mundo y es utilizada por un gran número de personas, (38).

Estas herramientas brindan la posibilidad de crear SIG con la mayoría de las funcionalidades que se pretenden implementar, pero agregarle las que son específicas del nuevo sistema resulta complicado y costoso, tanto en recursos como en tiempo, ya que son soluciones privadas, lo que trae consigo que para su utilización sea necesario adquirir una licencia que por su alto costo no está al alcance de todos.

También en el mundo SIG libre existen algunas plataformas para la creación de SIG, entre ellas la que más adelante ha alcanzado atendiendo a la cantidad de funciones que realiza, es GvSIG, el cual cuenta con dos plataformas para el desarrollo de aplicaciones para dispositivos móviles:

GvSIG Mini: esta herramienta es un visor de servicios de mapas, aunque cuenta con algunas funcionalidades locales pero muy básicas. Actualmente GvSIG mini cuenta con dos tipos de distribuciones, una para Android y otra para dispositivos que soporten aplicaciones Java. Esta aplicación es accesible desde distintos tipos de dispositivos pero debido a que la mayoría de sus funcionalidades son dependientes de un servidor de mapas no se considera conveniente su utilización, (39).

GvSIG Mobile: nace con la idea de convertirse en el primer cliente genérico SIG /IDE libre y de código abierto y no como un simple visor de datos, heredando toda la tecnología y experiencia de GvSIG y redefiniendo su diseño hacia una amplia variedad de dispositivos móviles, se convierte en una herramienta para la gestión de datos geográficos. En la actualidad GvSIG *Mobile* permite trabajar con *PDA*s, *Smartphones* y *Tablet PCs* gracias a que está desarrollado en Java. Esta aplicación permite que los usuarios dispongan de información confiable y actualizada en el campo, además de permitir editarla y así lograr la mayor exactitud en los cambios que se le realicen, (40), (41).

Luego de analizar algunas de las herramientas existentes en la actualidad para la creación de SIG, se decidió utilizar GvSIG *Mobile*, pues brinda un grupo de funcionalidades ya implementadas que facilitan la labor de desarrollo. Además, como se distribuye utilizando una licencia libre no requiere de pago alguno y se tiene acceso al código fuente, lo que permite agregar las funcionalidades personalizadas del SIGUCI *Mobile* sin la necesidad de la participación de terceros

ni de pago adicional.

1.5.3. Lenguajes.

1.5.3.1. Lenguajes de modelado.

Notación de Modelado de Procesos de Negocio (*Business Process Modeling Notation, BPMN*): tiene como principal objetivo facilitar el proceso de modelado del negocio reduciendo las brechas entre los técnicos y el proceso en sí. Este lenguaje solamente es compatible con los conceptos utilizados en el proceso de modelado del negocio, es decir, con él no se pueden modelar ni estrategias ni estructuras organizativas. Este lenguaje resume aspectos de otros como UML, IDEF, ebXML, *RosettaNet* e *Ivem* obteniendo de cada uno las mejores ideas y resumiéndolas en un solo lenguaje,(42).

Lenguaje unificado de modelado (UML): Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. No es un lenguaje de programación, es un lenguaje de modelado de propósito general, (43).

Luego de este análisis se decide la utilización de UML ya que este es un lenguaje de propósito general muy utilizado actualmente en la industria del software a diferencia de BPMN que aunque muy utilizado está desarrollado específicamente para modelar procesos de negocio. Además UML es el lenguaje que define RUP para la creación de todos los diagramas que se generan y es del que más conocimiento tiene el equipo de desarrollo.

1.5.3.2. Lenguaje de programación.

Java: Fue concebido para utilizarse en dispositivos electrodomésticos. Es un lenguaje orientado a objetos que brinda API, clases y librerías que se entregan como estándar. La principal facilidad de Java está en poderse ejecutar en casi cualquier dispositivo independientemente del *hardware* que utilice y esto es posible gracias a la Máquina Virtual de Java (*JVM*). La plataforma de Java se distribuye de tres formas, cada una especializada para una función. La Java 2 Micro Edition (*J2ME*) es la utilizada en dispositivos pequeños, tales como relojes o celulares. Hoy en día la mayoría de los celulares se fabrican con soportes para este lenguaje, lo que lo convierte en uno de los mejores para realizar aplicaciones para este tipo de dispositivos, (44). Por estas razones y debido a que la plataforma GvSIG está desarrollada con Java se decide escoger para implementar

las nuevas funcionalidades.

1.5.4. Herramientas de modelado.

Rational Rose (RR): Permite la generación de códigos en Ada, ANSI, C++, CORBA, Java, Visual C++ y Visual Basic. Proporciona un lenguaje común de modelado que facilita la rápida creación de un producto. Posibilita además realizar ingeniería inversa para Java 1.5 y brinda soporte para Java Beans 2.0. Contiene las herramientas necesarias para el modelado y visualización de aplicaciones Web. También proporciona la facilidad de generar documentación a partir de un modelo dado, (47). El principal problema de su uso es la licencia que se debe pagar, la cual en ocasiones tiene un alto precio.

Visual Paradigm for UML (VP-UML): Permite el diseño de sistemas utilizando diagramas UML que ayudan durante el proceso de desarrollo. Con esta herramienta el equipo de desarrollo puede realizar la captura de requisitos, la planificación de programas y controles, así como análisis y diseño de sistemas. Permite el modelado de los diagramas que se generan durante el ciclo de vida de RUP. Esta herramienta está distribuida bajo licencias libres, lo que hace más accesible su utilización y permite que esté al alcance de todos, (48).

Luego del análisis de estas dos herramientas se decide utilizar Visual Paradigm ya que su utilización es libre y no requiere de pago. Permite modelar todos los diagramas que se generan durante el ciclo de vida de RUP y está desarrollada específicamente para trabajar con UML. Además, el equipo de desarrollo ya cuenta con experiencia en la utilización de esta herramienta.

1.5.5. Entorno de desarrollo.

Eclipse Europa: Esta herramienta es una de las tantas existentes para desarrollar aplicaciones utilizando el lenguaje Java. Contiene una Arquitectura basada en *plugin* o extensiones lo que la convierte en fácilmente adaptable. Eclipse se distribuye bajo una licencia propia de los proyectos de la Fundación Eclipse, la Licencia Pública de Eclipse (EPL) ,(49). Atendiendo a que la plataforma que se seleccionó para el desarrollo lo utiliza como entorno de trabajo, se decide continuar su uso en la implementación de las nuevas funcionalidades pues resultaría muy engorroso y poco factible convertir el sistema de carpetas de un entorno a otro, además la licencia bajo la que se distribuye no implica que se deba pagar para su utilización.

1.5.6. Servicios web (WS):

No son más que un conjunto de aplicaciones que tienen la capacidad de interactuar en la red

intercambiando datos entre sí con el objetivo de ofrecer servicios. Los proveedores brindan estos servicios como procedimientos remotos y los usuarios los utilizan llamándolos a través de la red. Los WS proporcionan a los usuarios un mecanismo para comunicar las aplicaciones y así presentar la información contenida en estas de una forma dinámica, además aumenta la interoperabilidad y la extensibilidad de las aplicaciones que los utilizan, (46). En el desarrollo de la aplicación SIGUCI Mobile esta tecnología resolvería problemas de memoria en los casos de uso que requieran mucha información como la localización territorial, ya que esta se brinda actualmente mediante un grupo de servicios web que tiene implementados la universidad.

1.6. Conclusiones parciales.

Este capítulo abordó el desarrollo y evolución que han tenido los SIG desde su surgimiento hasta la actualidad y la utilización que han tenido en nuestro país, con lo que se logró una mejor comprensión de estos sistemas. Se analizaron un grupo de aplicaciones existentes que ayudaron a definir algunas funcionalidades que debía brindar el sistema, lo que permitió contar con una mejor base para analizar las herramientas y tecnologías actuales. Basándose en el análisis anterior se escogió la plataforma GvSIG *Mobile* como herramienta base para el desarrollo y la metodología RUP para documentar este proceso, pues se consideró que son las que mejor dan respuesta a las necesidades existentes. Teniendo ya definidas las herramientas y tecnologías con que se trabajará y conociendo las funcionalidades que debe presentar la aplicación se decide pasar a la propuesta de solución del sistema.

Capítulo 2: Propuesta de solución.

Introducción.

En este capítulo se realiza una representación del entorno donde funcionará la aplicación, modelando el dominio para lograr un mejor entendimiento del mismo. También se definen los requisitos que permiten reflejar las características con las que el sistema debe cumplir. Además se presenta el diagrama de casos de uso del sistema para un mejor entendimiento de las relaciones existentes entre los actores y los casos de uso del sistema.

2.1. Entorno donde trabajará el sistema.

2.1.1. Conceptos y eventos principales.

La **UCI** es la institución encargada de proveer toda la **información socioeconómica** necesaria para el correcto funcionamiento de la aplicación.

La **información socioeconómica** es muy variada y se brinda de distintas formas, mediante las **capas**, o mediante servicios web que se ofrecen en el repositorio de servicios web de la universidad (**UDDI**).

El **UDDI** brinda **información socioeconómica** que no está contenida en las **capas** y que es necesaria para el correcto funcionamiento de la aplicación.

Los **mapas** a su vez están formados por **capas** y por servicios web del **UDDI**, la cantidad de **capas** podrá ser ilimitada, la aplicación tendrá acceso a los servicios web que sean necesarios para su correcto funcionamiento.

El **Usuario** desde un dispositivo móvil puede visualizar los **mapas** y realizar modificaciones o analizar la información contenida en estos.

2.1.2. Diagrama de clases del modelo de dominio.

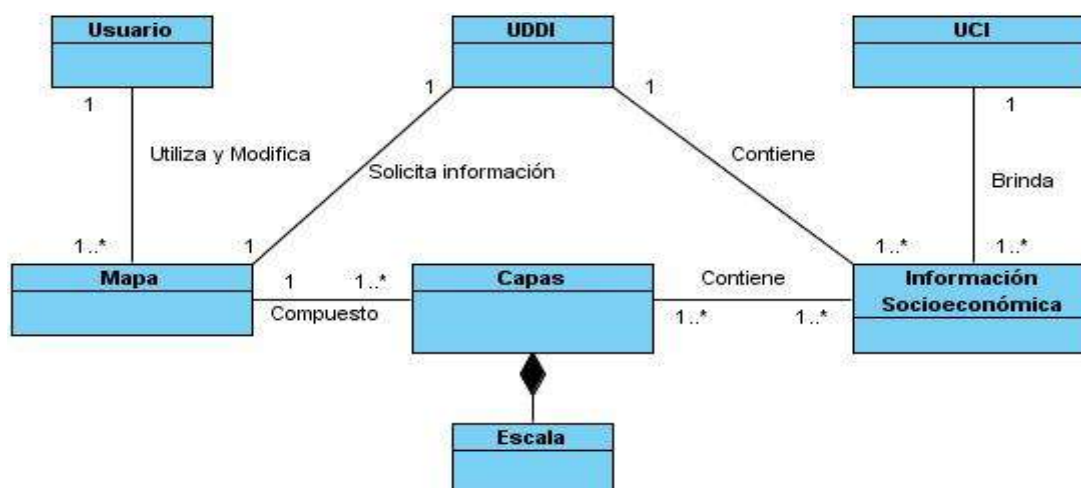


Figura # 1: *Modelo de Clases del Dominio.*

2.1.3. Glosario de términos del dominio.

Usuario.

Es el propietario del dispositivo móvil donde se encuentra instalada la aplicación.

UCI.

Es la institución encargada de proporcionar la información socioeconómica necesaria.

Mapa.

Contiene las capas y la información con que interactúa el usuario.

Capas.

Son el equivalente en un entorno digital a la leyenda en un mapa de papel, es decir, que cada capa agrupa elementos similares, por ejemplo los edificios pertenecen a una capa, los viales a otra y así sucesivamente. Las capas además tienen la información socioeconómica relacionada con los objetos que contiene.

Escala.

No es más que la relación que existe entre la distancia que separa a dos puntos en un mapa y la distancia real de estos en la superficie. Cuanto mayor sea la escala mayor será la exactitud de los datos. Por lo general los mapas con escala pequeña son para grandes extensiones y menos detallados que los de escala grande. Las escalas se pueden expresar de tres formas distintas en

los mapas: con escala gráfica, con expresiones o palabras, y con porciones o fracciones.

Información socioeconómica.

Conjunto de datos procesados y organizados que contienen información referente al aspecto social y económico de un lugar específico.

UDDI

Es el repositorio de servicios web de la UCI.

2.2. Requisitos funcionales.

Cargar Aplicación.

Nombre	RF 1 Cargar Aplicación
Descripción	Con este requisito se pretende que el sistema sea capaz de cargar todos los componentes necesarios para su correcto funcionamiento.
Seguimiento	CU_Cargar_Aplicación.
Prioridad	crítico

Tabla # 1: Requisito Funcional Cargar Aplicación.

Configurar aplicación.

Nombre	RF 2 Configurar aplicación
Descripción	Con este requisito se pretende que el usuario pueda configurar diferentes características del sistema como el idioma, el tamaño de los iconos, el tipo de referenciación que utiliza el sistema (SRS), la ruta, el fondo, las extensiones, y definir un proyecto por defecto.
Seguimiento	CU_Configurar_Aplicación.
Prioridad	crítico

Tabla # 2: Requisito Funcional Configurar aplicación.

Gestión de Proyecto.

Nombre	RF 3 Gestión de Proyecto
Asociados	RF 3.1 Nuevo proyecto. RF 3.4 Guardar proyecto como. RF 3.2 Abrir proyecto. RF 3.5 Salir. RF 3.3 Guardar proyecto.
Descripción	Con este requisito se pretende que el usuario pueda gestionar proyectos en la aplicación. Este caso de uso brinda la posibilidad de crear un nuevo proyecto, abrir un proyecto existente, guardar un proyecto nuevo o salvar los cambios en uno ya existente y además brinda la posibilidad de acceder a las propiedades del proyecto cargado.
Seguimiento	CU_Gestionar_Proyecto.
Prioridad	crítico

Tabla # 3: Requisito Funcional Gestionar Proyecto.

Gestión de Capas.

Nombre	RF 4 Gestión de capas
Asociados	RF 4.1 Añadir capa. RF 4.6 Bajar Nivel. RF 4.2 Eliminar Capa. RF 4.7 Poner Capa en Edición. RF 4.3 Simbología de la Capa. RF 4.8 Poner Capa Visible. RF 4.4 Agregar Atributo. RF 4.9 Capa Seleccionable. RF 4.5 Subir Nivel.
Descripción	Con este requisito se pretende que el usuario pueda gestionar las capas que conforman un proyecto. Este caso de uso brinda la posibilidad de cargar y eliminar una capa, acceder a la simbología, agregarle atributos, subir o bajar el nivel de visualización de la capa, Poner Capa en Edición, Poner Capa Visible, Capa Seleccionable.
Seguimiento	CU_Gestionar_Capa.

Prioridad	crítico
------------------	---------

Tabla # 4: Requisito Funcional Gestión de capas.

Redimensionar mapa.

Nombre	RF 5 Redimensionar mapa
Asociados	RF 5.1 Realizar zoom rectángulo. RF 5.5 Realizar zoom siguiente. RF 5.2 Realizar zoom +. RF 5.6 Realizar Paneo RF 5.3 Realizar zoom -. RF 5.7 Realizar zoom extenso. RF 5.4 Realizar zoom previo. RF 5.8 Recentrar.
Descripción	Este caso de uso se lleva a cabo con el objetivo de cambiar la forma de visualización del mapa en la pantalla.
Seguimiento	CU_Redimensionar_Mapa.
Prioridad	crítico

Tabla # 5: Requisito Funcional Redimensionar mapa.

Calcular Superficie.

Nombre	RF 6 Calcular Superficie
Asociados	RF 6.1 Calcular Área. RF 6.3 Calcular distancia. RF 6.2 Calcular Perímetro.
Descripción	Con este requerimiento se pretende que el usuario pueda realizar diferentes cálculos sobre el mapa tales como calcular área y perímetro y saber la distancia entre una sucesión de puntos.
Seguimiento	CU_Calcular_Superficie.
Prioridad	crítico

Tabla # 6: Requisito Funcional Calcular Superficie.

Tratar Información.

Nombre	RF 7 Tratar Información.
Asociados	RF 7.1 Modificar Información. RF 7.2 Consultar Información.
Descripción	Este caso de uso tiene como objetivo que el usuario pueda tratar la información de un objeto en el mapa, pudiendo consultarla o modificarla.
Seguimiento	CU_Tratar_Información.
Prioridad	crítico

Tabla # 7: Requisito Funcional Tratar Información.

Localizar Edificio.

Nombre	RF 8 Localizar Edificio.
Descripción	Con este requisito se pretende que el usuario pueda obtener, pasando el número de un edificio, información asociada a este, como el número de la manzana en la que se encuentra, el total de apartamentos, el total de pasos de escalera, y el total de residentes que en él viven. Además el sistema localizará el edificio en el mapa dibujándolo en otro color.
Seguimiento	CU_Localizar_Edificio.
Prioridad	crítico

Tabla # 8: Requisito Funcional Localizar Edificio.

Localizar Persona.

Nombre	RF 9 Localizar Persona.
Descripción	Con este requisito se pretende que el usuario pueda obtener, pasando el número de Solapín de una persona, la información asociada a ella como el nombre, los apellidos, el número de edificio, apartamento y teléfono del lugar donde reside en la universidad y el área a la que pertenece. Además se mostrará en el mapa el lugar de residencia de

	esta.
Seguimiento	CU_Localizar_Persona.
Prioridad	crítico

Tabla # 9: Requisito Funcional Localizar Persona.

2.3. Requisitos no funcionales.

Usabilidad.

- El sistema podrá ser utilizado por todos los usuarios que tengan en su poder un dispositivo móvil que cuente con los requisitos de *hardware* y *software* para soportarlo.

Portabilidad.

- El sistema debe poderse ejecutar en distintas plataformas de hardware como:
 - ✓ Asistentes de datos personales (PDA).
 - ✓ Internet *Tablet*.
 - ✓ *Tablet* PC.
 - ✓ PC ultraportatil.

Fiabilidad y seguridad.

- La información manejada por el sistema será de uso personal del usuario siendo este quien de acceso o no a terceras personas.
- La UCI brindará la información que crea conveniente para cada tipo de usuario.

Eficiencia.

- El tiempo de respuesta, la velocidad de procesamiento de la información, la actualización y la recuperación estarán dados por la cantidad de información a procesar, la utilización de la red en aquellas funcionalidades que así lo requieran y las capacidades de cada dispositivo, pudiendo ser esta última la de mayor peso.

Soporte.

- La aplicación recibirá mantenimiento en el período de tiempo determinado por el equipo de desarrollo.
- La cartografía será actualizada por la UCI en el período de tiempo estimado.

Software.

- Sistemas operativos Windows Mobile 5.0/6.0, Windows CE 2003, Windows XP,(50).
- Máquina virtual de Java Micro Edition CDC 1.1(J9 o *PhoneMe*), (51).
- Soporte para el protocolo de acceso a Internet WAP.

Hardware.

- **Procesador:** 32 Bit.
- **Velocidad de procesamiento:** Mínimo: 320MHz, Recomendable: 600MHz.
- **RAM:** Mínimo: 64 MB, Recomendable: 96 MB RAM,(50).
- **Conectividad:** sí.

2.4. Descripción del sistema propuesto.

2.4.1. Descripción de los actores.

Actor	Descripción
Usuario	Es el propietario del dispositivo móvil donde se encuentra instalado el sistema y el que interactúa con él.
UDDI	Es el directorio de servicios web de la UCI.

Tabla # 10: Descripción de los Actores.

2.4.2. Diagrama de casos de uso del sistema.

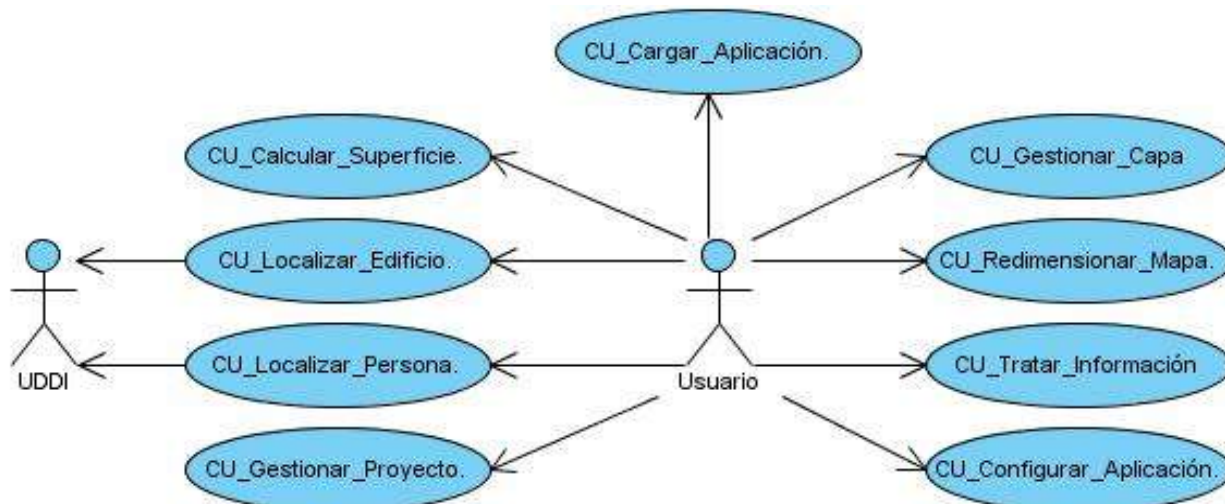
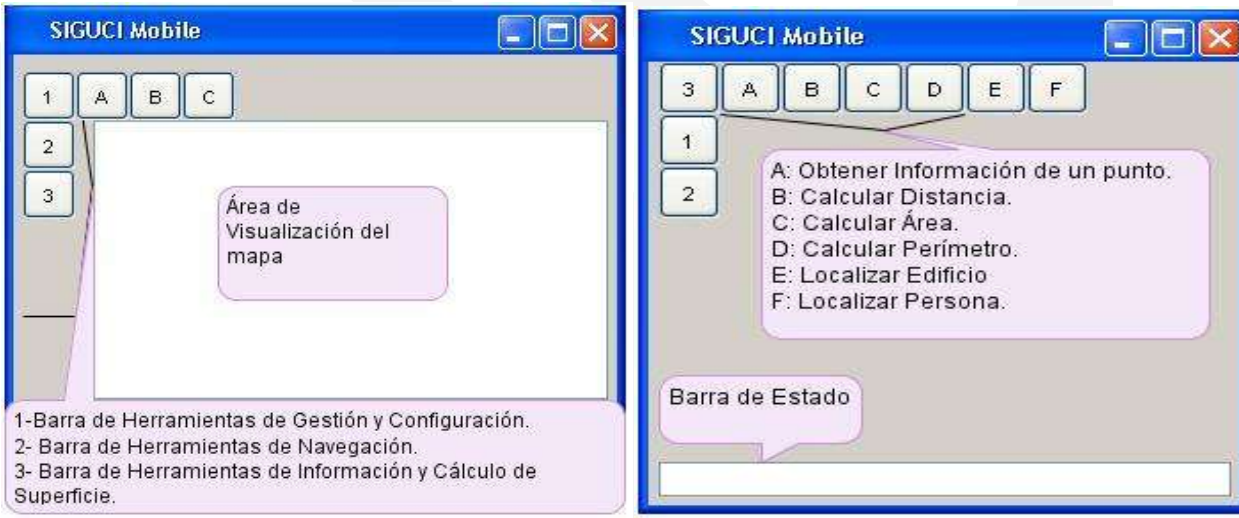


Figura # 2: *Diagrama de Casos de Uso del Sistema.*

2.4.3. Descripción textual de los casos de uso.

CU_Calcular_Superficie.

Caso de Uso	CU_Calcular_Superficie.
Actores:	Usuario
Propósito	Con este requerimiento se pretende que el usuario pueda realizar diferentes cálculos sobre el mapa, tales como calcular área y perímetro y saber la distancia entre una sucesión de puntos.
Resumen:	El caso de uso se inicia cuando el usuario selecciona la Barra de "Herramientas de Información y Cálculo de Superficie", en esta selecciona alguna de las opciones de este caso de uso y marca los puntos sobre el mapa. El caso de uso termina cuando el usuario desmarca la opción seleccionada.
Precondiciones	El CU_Cargar_Aplicación tiene que haber sido inicializado.
Referencias	RF 6, RF 6.1, RF 6.2, RF 6.3

Prioridad	crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
<p>1. El usuario selecciona la Barra de Herramientas de Información y Cálculo de Superficie (3).</p>	<p>2. El sistema actualiza la interfaz principal mostrando las herramientas contenidas en dicha barra.</p>	
<p>3. El usuario selecciona una de las opciones que brinda este caso de uso:</p> <ul style="list-style-type: none"> • Calcular Distancia (B). • Calcular Área (C). • Calcular Perímetro (D). 	<p>4. El sistema realiza la operación según la opción seleccionada por el usuario:</p> <ul style="list-style-type: none"> • Si la opción es “Calcular Distancia” ver sección “Calcular Distancia”. • Si la opción es “Calcular Área” ver sección “Calcular Área”. • Si la opción es “Calcular Perímetro” ver sección “Calcular Perímetro”. 	
Prototipo de Interfaz		
 <p>1-Barra de Herramientas de Gestión y Configuración. 2- Barra de Herramientas de Navegación. 3- Barra de Herramientas de Información y Cálculo de Superficie.</p> <p>Área de Visualización del mapa</p> <p>Barra de Estado</p> <p>A: Obtener Información de un punto. B: Calcular Distancia. C: Calcular Área. D: Calcular Perímetro. E: Localizar Edificio. F: Localizar Persona.</p>		

Sección "Calcular Perímetro".	
Acción del Actor	Respuesta del Sistema
	5. El sistema muestra la Barra de Estado en la parte inferior de la pantalla sin valor alguno.
6. El usuario marca puntos en el mapa.	7. El sistema verifica si existen más de dos puntos y cierra automáticamente el polígono, si no va al flujo alterno F.
	8. El sistema realiza el cálculo del perímetro.
	9. El sistema muestra el resultado del cálculo en la Barra de Estado y vuelve a la acción 6 de este flujo. El CU finaliza cuando el usuario desmarca el botón seleccionado.
Sección "Calcular Distancia".	
Acción del Actor	Respuesta del Sistema
	5. El sistema muestra la Barra de Estado en la parte inferior de la pantalla sin valor alguno.
6. El usuario marca puntos en el mapa.	7. El sistema realiza el cálculo de la distancia entre los puntos.
	8. El sistema muestra el resultado del cálculo en la Barra de Estado y vuelve a la acción 6 de este flujo. El CU finaliza cuando el usuario desmarca el botón seleccionado.
Sección "Calcular Área".	
Acción del Actor	Respuesta del Sistema
	5. El sistema muestra la Barra de Estado en la parte inferior de la pantalla sin valor alguno.

6. El usuario marca puntos en el mapa.	7. El sistema verifica si existen más de dos puntos y cierra automáticamente el polígono, si no va al flujo alternativo F.
	8. El sistema realiza el cálculo del área.
	9. El sistema muestra el resultado del cálculo en la Barra de Estado y vuelve a la acción 6 de este flujo. El CU finaliza cuando el usuario desmarca el botón seleccionado.
Flujo Alternativo F	
Acción del Actor	Respuesta del Sistema
	7. El sistema da como resultado cero en la Barra de Estado.
	8. Va a la acción 6 del flujo de eventos.
Poscondiciones	El sistema muestra el resultado del cálculo del perímetro en la Barra de Estado.

Tabla # 11: Descripción Textual del CU_Calcular_Superficie.

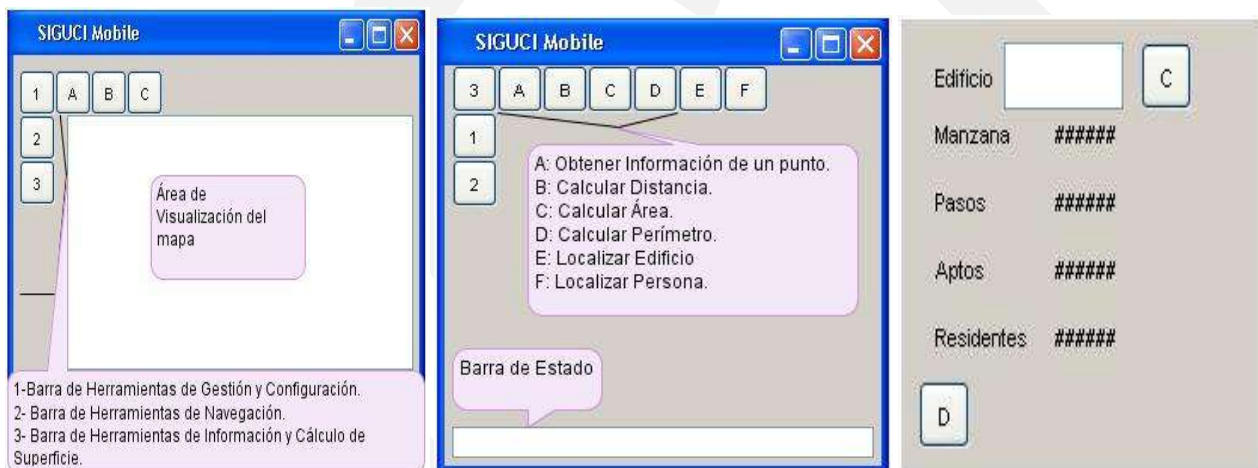
CU_Localizar_Edificio.

Caso de Uso	CU_Localizar_Edificio.
Actores	Usuario
Propósito	Con este requisito se pretende que el usuario pueda obtener, pasando el número de un edificio, información asociada a este, como el número de la manzana en la que se encuentra, el total de apartamentos, el total de pasos de escalera, y el total de residentes que en él viven. Además el sistema localizará el edificio en el mapa dibujándolo en otro color.
Resumen	El caso de uso se inicia cuando el usuario selecciona la Barra de Herramientas de Información y Cálculo de Superficie, en esta barra selecciona la opción de Localizar Edificio, el sistema muestra una interfaz de

	búsqueda donde el usuario introduce los datos de un edificio y tiene como salida características como el número del edificio, el número de la manzana donde se encuentra, el total de pasos de escalera que tiene, el total de apartamentos y el total de residentes que viven en él.
Precondiciones	El CU_Cargar_Aplicación tiene que haber sido inicializado.
Referencias	RF 8
Prioridad	crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la Barra de Herramientas de Información y Cálculo de Superficie (3) .	2. El sistema actualiza la interfaz principal mostrando las herramientas contenidas en dicha barra.
3. El usuario selecciona la opción de Localizar Edificio (E) .	4. El sistema muestra una interfaz solicitando el número del edificio.
5. El usuario introduce el número del edificio y da clic en el botón buscar (C) .	6. El sistema verifica que el valor que se introdujo es un número. En caso de no ser así, va al flujo alternativo G .
	7. El sistema trata de conectarse al servicio web que proporciona los datos, si no va al flujo alternativo H .
	8. Si se establece la conexión muestra el número del edificio, el número de la manzana, el total de pasos de escalera, el total de apartamentos y el total de residentes; si los datos no se encuentran va al flujo alternativo I .

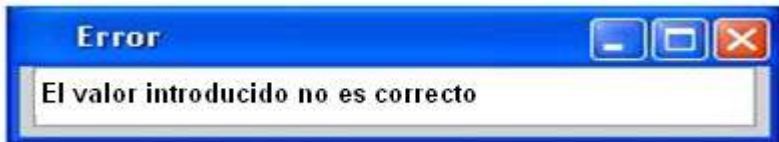

<p>9. El usuario consulta los datos deseados y da clic en el botón de cerrar (D).</p>	<p>10. El sistema verifica que la capa de edificios esté cargada, si no va al flujo alterno J.</p>
	<p>11. El sistema busca una coincidencia entre el número del edificio y los objetos que existen en el mapa. Si logra establecer una coincidencia muestra el edificio seleccionado de otro color en el mapa, si no va al flujo alterno K.</p>
	<p>12. Muestra la interfaz Principal. Finaliza el CU.</p>


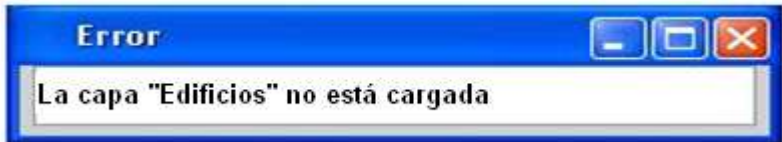
Prototipo de Interfaz



Flujo Alterno G

Acción del Actor	Respuesta del Sistema
	<p>6. El sistema muestra un mensaje informando que el valor introducido no es correcto.</p>
<p>7. El usuario lee el mensaje y da clic en el botón cerrar.</p>	<p>8. Va a la acción 3 del flujo normal de eventos.</p>

<i>Prototipo de Interfaz</i>	
	
Flujo Alternativo H	
Acción del Actor	Respuesta del Sistema
	7. El sistema muestra un mensaje informando que no se pudo establecer la conexión y pide que lo intente de nuevo.
8. El usuario lee el mensaje y da clic en el botón cerrar.	9. Va a la acción 3 del flujo normal de eventos.
<i>Prototipo de Interfaz</i>	
	
Flujo Alternativo I	
Acción del Actor	Respuesta del Sistema
	8. El sistema muestra un mensaje de "No disponible" en los datos que no se encontraron.
	9. Va a la acción 3 del flujo normal de eventos.

Prototipo de Interfaz	
	
Flujo Alternativo J	
Acción del Actor	Respuesta del Sistema
	10. El sistema muestra un mensaje que informa que "La Capa de edificios no está cargada".
	11. Muestra la interfaz principal y finaliza el CU.
Prototipo de Interfaz	
	
Flujo Alternativo K	
Acción del Actor	Respuesta del Sistema
	8. El sistema muestra un mensaje que informa que "No se encontró ninguna coincidencia".
	9. Muestra la interfaz principal y finaliza el CU.


<i>Prototipo de Interfaz</i>	
	
Poscondiciones	El sistema muestra una ventana informando el número del edificio, el número de la manzana en la que se encuentra, el total de pasos de escaleras, el total de apartamentos y el total de residentes. Además marca el edificio en el mapa de otro color.

Tabla # 12: *Descripción Textual del CU_Localizar_Edificio.*

Las restantes descripciones textuales de los casos de uso se podrán encontrar en los anexos (Tablas de la 19 a la 25).

2.5. Conclusiones parciales.

En este capítulo se definieron los conceptos y eventos más importantes y se realizó el diagrama de clases del dominio para un mejor entendimiento del entorno. Se describieron los requisitos funcionales y no funcionales, así como actores y casos de uso y se realizó el diagrama de casos de uso del sistema. Además se diseñó el prototipo de interfaz de usuario de cada una de las funcionalidades. Luego de realizar todas las tareas y haber generado todos los diagramas y documentos del flujo de trabajo de la Captura de Requisitos, se pasa al flujo de Análisis y Diseño de la solución.

Capítulo 3: Análisis y diseño de la solución.

Introducción.

En este capítulo se abordan las principales actividades dentro del flujo de trabajo de Análisis y Diseño de la metodología RUP. En él se tratan los modelos de análisis y de diseño y se realiza una descripción de la arquitectura utilizada.

3.1. Modelo de análisis.

En el modelo de análisis se ofrece una especificación de requisitos más precisa que en el flujo anterior, ya que se describen estos utilizando el lenguaje de los desarrolladores y se estructuran de un modo que faciliten su comprensión, por lo tanto se pueden introducir conceptos técnicos lo que permite razonar sobre el funcionamiento interno que tendrá el *software*, (52).

3.1.1. Diagramas de clases del análisis.

Las clases del análisis representan abstracciones de una o varias clases y/o subsistemas del diseño y siempre encajan dentro de uno de los tres estereotipos definidos (Interfaz, Controladoras, Entidades). Están centradas fundamentalmente en los requisitos funcionales, dejando los no funcionales para otro momento. Las relaciones entre ellas son más conceptuales que las relaciones existentes en el diseño y la implementación, (53)

DCA (CU_Calcular_Superficie).

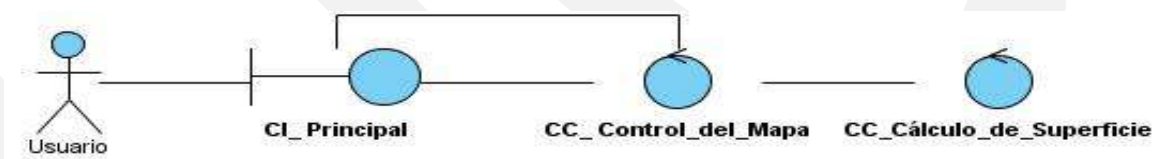


Figura # 3: Diagrama de Clases del análisis CU_Calcular_Superficie.

DCA (CU_Localizar_Edificio).

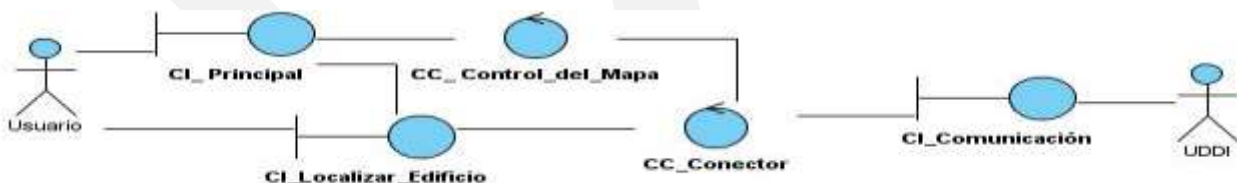


Figura # 4: Diagrama de Clases del análisis CU_Localizar_Edificio.

Los restantes diagramas de clases del análisis se podrán encontrar en los anexos (figuras de la 12 a la 18).

3.1.2. Diagramas de interacción.

Los diagramas de interacción muestran cómo se comunican los objetos durante el ciclo de vida del caso de uso. Existen dos tipos de estos diagramas: los de colaboración y los de secuencia, (54).

3.1.2.1. Diagramas de colaboración.

DColab (CU_Calcular_Superficie).

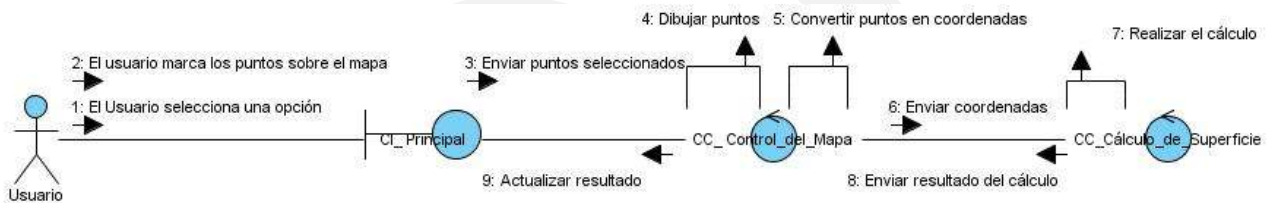


Figura # 5: Diagrama de Colaboración CU_Calcular_Superficie.

DColab (CU_Localizar_Edificio).



Figura # 6: Diagrama de Colaboración CU_Localizar_Edificio.

Los restantes diagramas de colaboración se podrán encontrar en los anexos (figuras de la 19 a la 25).

3.2. Modelo de diseño.

El modelo de diseño crea una representación o modelo del software que a diferencia del modelo de análisis brinda detalles más profundos acerca de los datos, la arquitectura, las interfaces y los componentes que son necesarios para la implementación del software. Es aquí donde se le comienza a dar importancia a los requisitos no funcionales y se comienza a tener en cuenta el

lenguaje de programación con que será desarrollado el sistema, (55).

3.2.1. Diagramas de clases de diseño.

Los diagramas de clases del diseño, a diferencia de los de análisis, representan clases y relaciones más reales. En estos diagramas se tienen en cuenta el lenguaje y los requisitos no funcionales. Por la complejidad del diseño de la plataforma GvSIG *Mobile* solamente se representarán en los diagramas las clases que intervienen directamente en la funcionalidad que se lleva a cabo, (56).

DCD (CU_Calcular_Superficie).

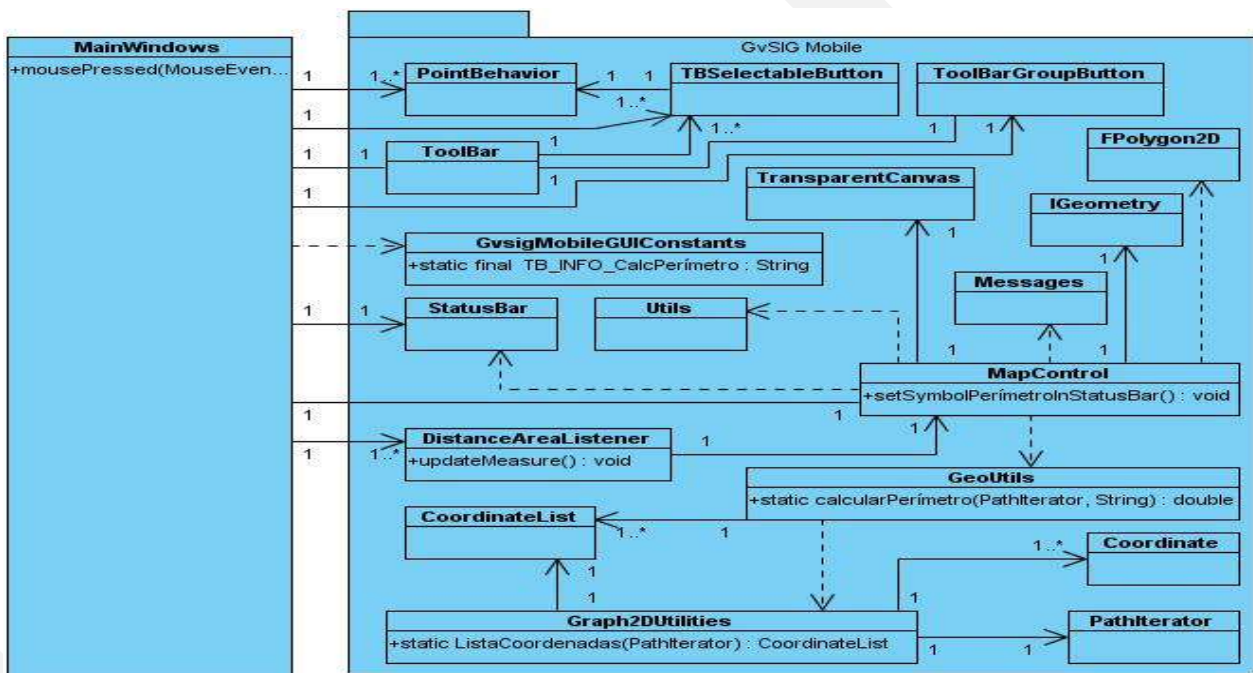


Figura # 7: Diagrama de clases del diseño CU_Calcular_Superficie.

DCD (CU_Localizar_Edificio).

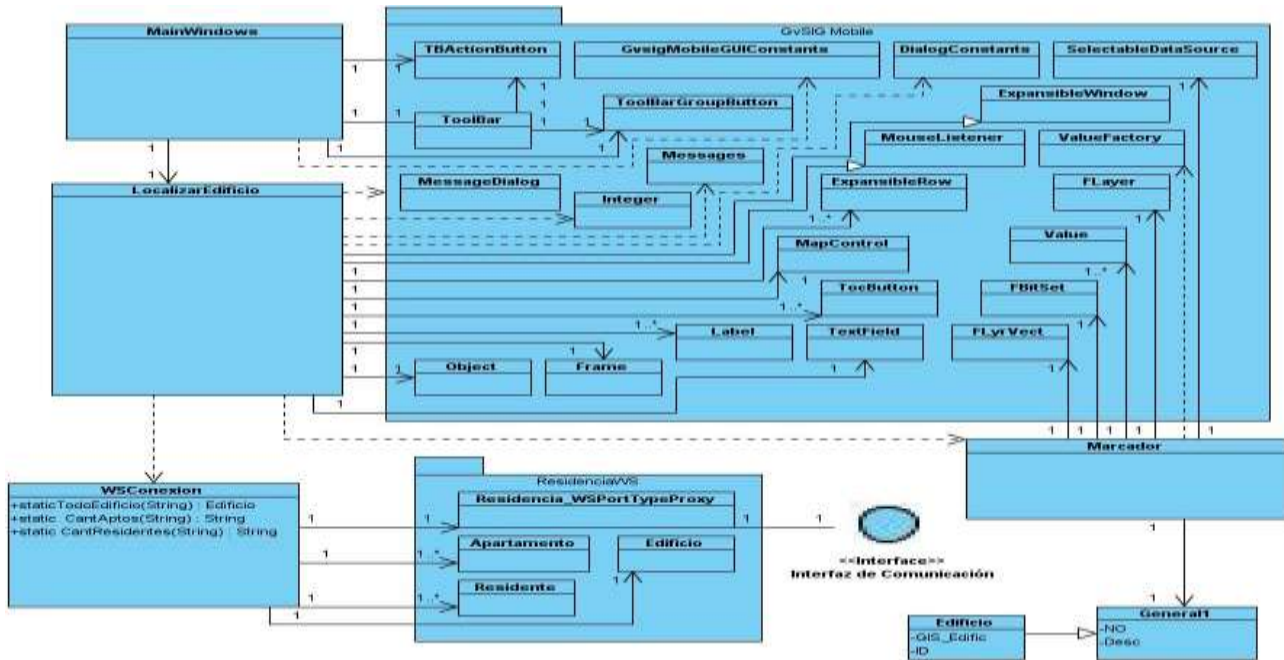


Figura # 8: Diagrama de clases del diseño CU_Localizar_Edificio.

Los restantes diagramas de clases del diseño se podrán encontrar en los anexos (Figuras de la 26 a la 32).

3.3. Arquitectura.

En la arquitectura se da forma al sistema, ya que este se analiza desde varios puntos de vista y se definen los artefactos más importantes del sistema a los que se le agregarán los demás componentes sin realizarle muchos cambios, (57).

Para la realización de la aplicación se decidió utilizar la arquitectura en capas, específicamente la de tres niveles. La misma está dividida en tres capas lógicas, las cuales tiene una función específica y bien definida, estas capas son:

Interfaz: En esta capa se definen todos los aspectos que tienen que ver con las interfaces, tanto los componentes con que contarán, como los eventos y componentes útiles.

Lógica: Esta capa contiene toda la lógica del negocio, es decir, que contiene las funcionalidades que realiza la aplicación. Proporciona además un puente entre la capa Interfaz y la capa de Acceso a Datos.

Acceso a Datos: En esta capa están representados todos los datos con que interactúa el sistema.

3.4. Modelo de datos.

La información con que interactúa la aplicación se encuentra principalmente en una serie de capas en formato *Shapefile*, aunque también se utilizan una serie de servicios web que proporcionan información no contenida en estas capas.

3.4.1. Formato shapefile:

Este formato consiste en una serie de archivos que representan vectorialmente datos espaciales. Existen tres tipos de archivos que son obligatorios para el formato *Shapefile*:

-**Shape (.shp):** Este es el archivo principal dentro del formato *Shapefile*. En él se almacenan características geométricas de los elementos que existen en las capas mediante puntos, polilíneas o polígonos. Toda la información que se encuentra en este archivo posee una posición geográfica definida mediante coordenadas XY, aunque en algunos casos se puede encontrar también información como la altura (Z) o valores de mediciones. Este tipo de archivo no permite el almacenamiento de información topológica lo que lo hace más sencillo para la edición, es una ventaja en cuanto a almacenamiento y velocidad de acceso. La siguiente figura muestra cómo está compuesto un archivo .shp.

- **ShapeIndex (.shx):** Este archivo como su nombre lo indica contiene índices de las entidades geométricas que permite refinar las búsquedas dentro de los archivos .shp. La cabecera de este archivo es idéntica a la del archivo principal (.shp). Luego de la cabecera se encuentran los registros en el mismo orden en que se encuentran en el archivo principal y contienen las posiciones de los registros respectivos en el .shp y la longitud.

- **dBase (dbf):** Este archivo es una tabla de datos en la que se registran atributos alfanuméricos o de fecha a los elementos que se contienen en el .shp. Estos registros se asocian con una única entrada en la tabla y ambos archivos se vinculan con un número de registro en el archivo principal y el código de la tabla, (58).

Los archivos que componen el *Shapefile* solo deben variar su extensión. Además incluyen un código común que relaciona los registros y la información asociada en los archivos:

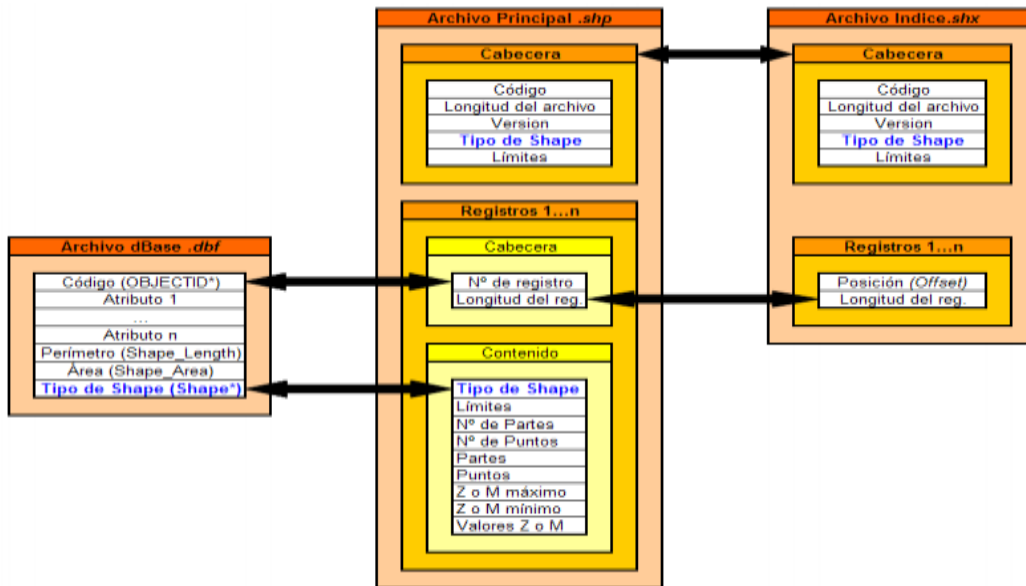


Figura # 9: Interacción entre archivos que forman el shapefile.

Las capas que componen la cartografía de la universidad son: AreasDeportivas, Contornos, Edificios, Manzanas, Separadores y Viales.

En la mayoría de las capas se encuentran polígonos y en las capas Separadores y Viales podemos encontrar polilíneas.

3.4.2. Servicios web utilizados:

La UCI mediante el Descripción Universal de Descubrimiento e Integración (*Universal Description Discovery and Integration*, UDDI) presta un grupo de servicios web (*Web Service*, WS) que brindan la información necesaria para el correcto funcionamiento de las funcionalidades Localizar Edificio y Persona. Estos son:

Servicio web del Sistema de Gestión Académica (AkademosWS):

Brinda información importante sobre los estudiantes de la universidad. Este servicio es utilizado para brindar información de si una persona es o no estudiante y en caso de que lo sea conocer el nombre, los apellidos y la facultad a la que pertenece.

Servicios web del Sistema de Capital Humano (AssetsWS) y del Registro Personal (PersonalWS):

Estos servicios brindan información acerca de los trabajadores de la universidad, es necesaria la utilización de ambos ya que cada cual brinda la información dependiendo de la categoría de los

trabajadores, es decir, que cada sistema se especializa en la información de diferentes categorías de trabajadores. Estos servicios brindan el nombre, los apellidos y el área a la que pertenece una persona, además AssetsWS brinda el número de teléfono de la misma.

Servicio web del Registro de Residentes (ResidenciaWS):

Este servicio brinda información de los residentes de la universidad, es el encargado de definir si una persona es o no residente y de brindar los números de edificio, de apartamento y de teléfono en caso de que lo sea. Además brinda todos los datos de un edificio en específico.

3.5. Conclusiones parciales.

En este capítulo como parte del modelo de análisis se desarrollaron los diagramas de clases y de secuencia para cada caso de uso del sistema y como parte del diseño se elaboraron los diagramas de clases del diseño. También se realizó una breve descripción de los servicios web que se utilizan y del funcionamiento del formato de datos *Shapefile* para así tener una mejor comprensión de estos. Además se efectuó la descripción de la arquitectura utilizada en el desarrollo de la aplicación. Luego de realizar estas tareas y de generar los artefactos de este flujo de trabajo, se decide pasar a los flujos de trabajo de Implementación y Prueba.

Capítulo 4: Implementación y prueba.

Introducción.

En este capítulo se muestran los principales artefactos generados durante el flujo de trabajo de implementación y prueba. Se describen los modelos de despliegue y de componentes y se realiza un plan de pruebas para la solución.

4.1. Implementación.

El principal objetivo de este flujo de trabajo es desarrollar la arquitectura y el sistema como un todo. Es por eso que en este flujo se planifican las integraciones necesarias para cada iteración, se distribuye el sistema asignando componentes ejecutables a los nodos del diagrama de despliegue, se implementan las clases y los subsistemas y se prueban los componentes individualmente enlazándolos en uno o más ejecutables,(59).

4.1.1. Modelo de despliegue.

En el modelo de despliegue se muestra cuál será la disposición física de los elementos de procesamiento (nodos) necesarios para la ejecución satisfactoria del sistema y la interacción entre estos. Además se modelan también los dispositivos que no tienen capacidad de procesamiento pero que brindan soporte a los nodos,(60).

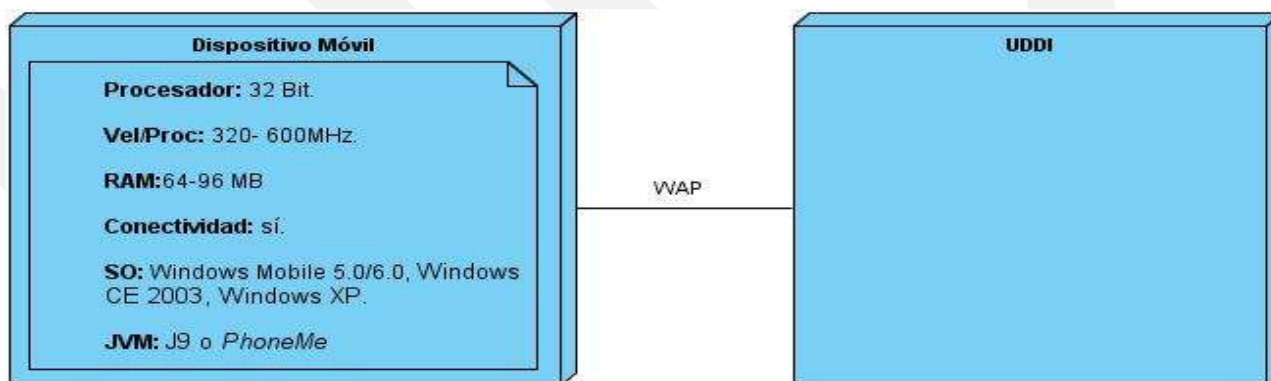


Figura # 10: Diagrama de Despliegue.

En el dispositivo móvil se encontrará corriendo la aplicación SIGUCI Mobile basada en la plataforma GvSIG Mobile la cual se conecta al UDDI mediante el protocolo WAP para solicitar los datos que son necesarios y no se encuentran contenidos en el mapa por las limitaciones de

memoria.

4.1.2. Diagrama de componentes.

Estos diagramas representan los componentes que forman el sistema y las dependencias que existen entre ellos y son utilizados para modelar la arquitectura del sistema. Un componente puede ser una biblioteca, un archivo, módulo, o paquete, (60).

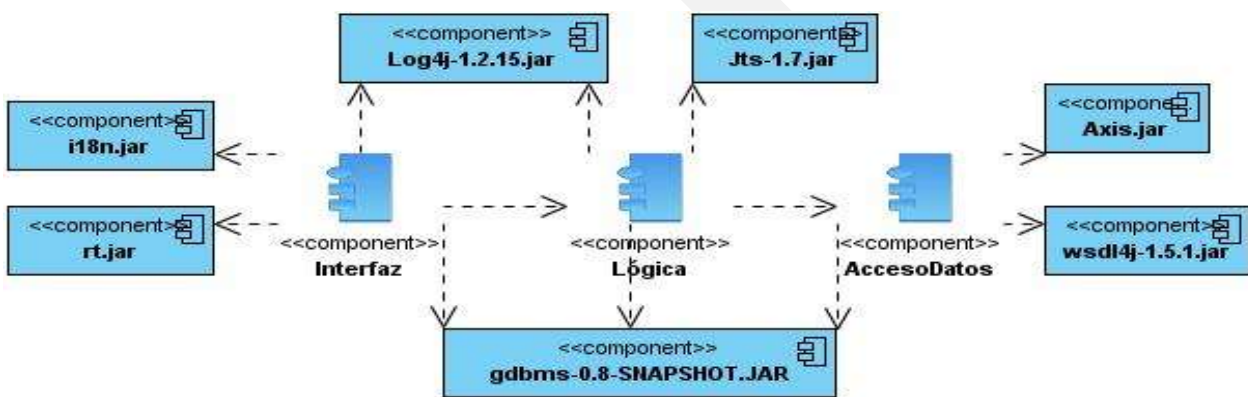


Figura # 11: *Diagrama de Componentes.*

4.2. Plan de pruebas del sistema.

Durante este flujo de trabajo se verifica el resultado de la implementación del sistema, probando las construcciones generadas en cada iteración y el resultado final del sistema. Es por eso que durante este flujo se planifican, diseñan e implementan las pruebas que serán necesarias para cada iteración del sistema, además se aplican y su resultado se maneja sistemáticamente, para devolver las construcciones que den resultados negativos a los flujos de diseño o implementación, (61).

4.2.1. Objetivo.

El principal objetivo de aplicar pruebas es tener la certeza de que el sistema cumple de manera satisfactoria todas las funcionalidades que se definieron.

4.2.2. Nivel de prueba.

El nivel definido para aplicar las pruebas es el de sistema, ya que aquí se puede verificar el correcto funcionamiento de la aplicación sin la necesidad de aplicar pruebas en los demás. Este nivel tiene como inconveniente que es más difícil encontrar en qué lugar del sistema se

encuentran los errores, pero debido a que la interacción entre los casos de uso es relativamente poca, se decide que es el más conveniente.

4.2.3. Método utilizado.

Para la realización de las pruebas se utiliza el método de Caja Negra, el cual permite mediante la interacción con la interfaz demostrar que:

- Las funcionalidades con que cuenta el sistema son operativas.
- Los datos que se introducen son aceptados correctamente.
- El resultado de la operación es correcto.
- Se mantiene la integralidad de los datos con los que trabaja externamente el sistema.

Se ha decidido aplicar este método de prueba pues es una forma de comprobar que el sistema funciona correctamente sin tener que interactuar con el código.

4.2.4. Técnica utilizada.

Dentro del método de prueba de caja negra existen diferentes técnicas. Para la realización de las pruebas de este sistema se decidió aplicar la técnica de partición equivalente. Esta técnica permite dividir las entradas en clases de datos, lo cual permite diseñar las pruebas en dependencia de la acción del sistema y comprobar que funciona correctamente.

4.2.5. Diseño de casos de prueba.

4.2.5.1. CU_Calcular_Superficie.

Condiciones de Ejecución del CU_Calcular_Superficie.

El CU_Cargar_Aplicación tiene que haber sido inicializado.

Secciones a probar en el CU_Calcular_Superficie.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: "Calcular Perímetro"	EC 1.1: Calcular perímetro satisfactoriamente	Se selecciona la opción Calcular Perímetro y se seleccionan más de dos puntos sobre el mapa. El sistema realiza el cálculo y muestra el resultado en la Barra de Estado.	<ol style="list-style-type: none">1. El usuario selecciona la Barra de Herramientas de Información y Cálculo de Superficie.2. El sistema actualiza la interfaz principal mostrando las herramientas contenidas en dicha barra3. El actor escoge la opción Calcular Perímetro.4. El sistema muestra la Barra de Estado en la parte inferior de la pantalla sin valor alguno.5. El usuario marca más de dos puntos en el mapa.6. El sistema realiza el cálculo y muestra el resultado en la barra de estado.

	<p>EC 1.2: Fallo al calcular perímetro.</p>	<p>Se selecciona la opción Calcular Perímetro y se seleccionan uno o dos puntos sobre el mapa. El sistema muestra cero en la Barra de Estado como resultado.</p>	<ol style="list-style-type: none"> 1. El usuario selecciona la Barra de Herramientas de Información y Cálculo de Superficie. 2. El sistema actualiza la interfaz principal mostrando las herramientas contenidas en dicha barra 3. El actor escoge la opción Calcular Perímetro. 4. El sistema muestra la Barra de Estado en la parte inferior de la pantalla sin valor alguno. 5. El usuario marca uno o dos puntos sobre el mapa 6. El sistema muestra cero como resultado en la barra de estado.
--	---	--	---

Tabla # 13: Secciones a probar en el CU_Calcular_Superficie.

Matriz de de Caso de Prueba para el CU_Calcular_Superficie.

Escenario	Respuesta del Sistema	Resultado de la Prueba
<p>EC 1.1: Calcular perímetro satisfactoriamente.</p>	<p>Realiza el cálculo y muestra el resultado en la Barra de Estado.</p>	
<p>EC 1.2: Fallo al calcular perímetro.</p>	<p>El resultado en la Barra de Estado es cero.</p>	

Tabla # 14: Matriz de de Caso de Prueba para el CU_Calcular_Superficie

Matriz de Datos para el CU_Calcular_Superficie.

Escenario	Respuesta del Sistema	Resultado de la Prueba
EC 1.1: Calcular perímetro para más de dos puntos.	Realiza el cálculo y muestra el resultado en la Barra de Estado.	Satisfactorio
EC 1.2: Fallo al calcular perímetro.	El resultado en la Barra de Estado es cero.	Satisfactorio

Tabla # 15: *Matriz de Datos para el CU_Calcular_Superficie.*

Registro de defectos y dificultades detectados.

No se registran defectos ni dificultades para el caso de uso.

4.2.5.2. CU_Localizar_Edificio.

Condiciones de Ejecución del CU_Localizar_Edificio.

El CU_Cargar_Aplicación tiene que haber sido inicializado.

Secciones a probar en el CU_Localizar_Edificio.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC: Localizar Edificio.	1 EC: 1.1 Localizar edificio entrando una cadena.	El usuario entra una cadena y da clic en el botón buscar. El sistema lanza un mensaje de error: "El valor introducido no es correcto".	<ol style="list-style-type: none">1. El usuario entra una cadena y da clic en el botón Buscar.2. El sistema lanza un mensaje de error: "El valor introducido no es correcto".
	EC: 1.2 Localizar edificio sin conexión.	El usuario entra una cadena y da clic en el botón buscar. El sistema lanza un mensaje de error: "No se pudo establecer la conexión, inténtelo de nuevo"	<ol style="list-style-type: none">1. El usuario entra una cadena y da clic en el botón buscar.2. El sistema lanza un mensaje de error: "No se pudo establecer la conexión inténtelo de nuevo"

EC: 1.3 Datos de respuesta incompleta.	El usuario entra una cadena y da clic en el botón buscar, el servicio web le devuelve los datos .El sistema muestra el mensaje “No disponible” en aquellos que no existan.	<ol style="list-style-type: none">1. El usuario entra una cadena y da clic en el botón buscar.2. El servicio web devuelve los datos.3. El sistema muestra el mensaje “No disponible” en aquellos que no existan.
EC: 1.4 Capa inexistente.	El usuario entra una cadena, da clic en el botón buscar y luego en el botón cerrar, el sistema muestra un mensaje de error: “La capa de edificio no esta cargada”.	<ol style="list-style-type: none">1. El usuario entra una cadena, da clic en el botón buscar y luego en el botón cerrar.2. El sistema verifica que exista la capa de edificio, en caso de no existir muestra un mensaje de error: “La capa de edificio no esta cargada”.
EC: 1.5 No existe coincidencia	El usuario entra una cadena, hace clic en el botón buscar y luego en el botón cerrar, el sistema muestra un mensaje de error: “No se encontró coincidencia”.	<ol style="list-style-type: none">1. El usuario entra una cadena, da clic en el botón buscar y luego en el botón cerrar.2. El sistema muestra un mensaje de error: “No se encontró coincidencia”.
EC: 1.6 Localizar edificio con éxito.	El usuario entra una cadena, da clic en el botón buscar y luego en el botón Cerrar. El sistema	<ol style="list-style-type: none">1. El usuario entra una cadena, da clic en el botón buscar y luego en el botón Cerrar.

		dibuja de amarillo el edificio.	2. El sistema dibuja de amarillo el edificio
--	--	---------------------------------	--

Tabla # 16: Secciones a probar en el CU_Localizar_Edificio.

Descripción de variables del CU_Localizar_Edificio.

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Número Edificio	TextBox	Sí	Se introduce el número del edificio.

Tabla # 17: Descripción de variables del CU_Localizar_Edificio.

Matriz de de Caso de Prueba para el CU_Localizar_Edificio.

Escenario	Variable 1 Número Edificio	Respuesta del Sistema	Resultado de la Prueba
EC: 1.1 Localizar edificio entrando una cadena.	I	Lanza un mensaje de error: "El valor introducido no es correcto".	
EC: 1.2 Localizar edificio sin conexión.	V	Lanza un mensaje de error: "No se pudo establecer la conexión, inténtelo de nuevo"	
EC: 1.3 Datos de respuesta incompleta.	V	Muestra el mensaje "No disponible" en aquellos que no existan.	
EC: 1.4 Capa inexistente.	V	Muestra un mensaje de error: "La capa de edificio no esta cargada".	
EC: 1.5 No existe coincidencia	I	El sistema muestra un mensaje de error: "No se encontró coincidencia".	

EC: 1.6 Localizar Edificio con éxito.	V	El sistema muestra los datos del edificio en la interfaz Localizar Edificio y luego marca en amarillo el edificio en el mapa.	
---------------------------------------	---	---	--

Tabla # 18: Matriz de de Caso de Prueba para el CU_Localizar_Edificio.

Matriz de Datos para el CU_Localizar_Edificio.

Escenario	Variable 1 Número Edificio	Respuesta del Sistema	Resultado de la Prueba
EC: 1.1 Localizar edificio entrando una cadena.	Ruben	Lanza un mensaje de error: "El valor introducido no es correcto".	Satisfactorio.
	" "		Satisfactorio.
	12sdf55		Satisfactorio.
EC: 1.2 Localizar edificio sin conexión.	1	Lanza un mensaje de error: "No se pudo establecer la conexión, inténtelo de nuevo"	Satisfactorio.
EC: 1.3 Datos de respuesta incompleta.	63	Muestra el mensaje "No disponible" en aquellos que no existan.	Satisfactorio.
EC: 1.4 Capa inexistente.	1	Muestra un mensaje de error: "La capa de edificio no esta cargada".	Satisfactorio.

EC: 1.5 No existe coincidencia	200	El sistema muestra un mensaje de error: "No se encontró coincidencia".	Satisfactorio.
EC: 1.6 Localizar Edificio con éxito.	116	El sistema muestra los datos del edificio en la interfaz Localizar Edificio y luego marca en amarillo el edificio en el mapa.	Satisfactorio.

Tabla # 19: *Matriz de Datos para el CU_Localizar_Edificio.*

Registro de defectos y dificultades detectados.

No se registran defectos ni dificultades para el caso de uso.

4.3. Conclusiones parciales.

En este capítulo como parte de la implementación se realizaron los diagramas de despliegue y componentes. Como parte del flujo de pruebas se diseñaron los casos de pruebas, se realizó la descripción de las variables y se aplicaron las pruebas al sistema.

Conclusiones generales.

El desarrollo de la presente investigación ha permitido arribar a las siguientes conclusiones:

- La elaboración del marco teórico conceptual evidenció que en la actualidad es frecuente el uso de los SIG para el apoyo a la toma de decisiones por las funcionalidades que brinda este tipo de sistemas, además pudo constatar que a pesar de esto la Universidad no contaba con una aplicación que permitiera hacer uso de estas bondades desde dispositivos móviles.
- El desarrollo de un SIG para dispositivos móviles hace posible el acceso a la información geográfica de la Universidad desde cualquier lugar de esta, apoyando la toma de decisiones de los miembros de la comunidad universitaria.
- Las pruebas de caja negra realizadas al sistema permitieron comprobar el correcto funcionamiento de la aplicación y de este modo el cumplimiento de los requisitos funcionales.

Recomendaciones

Se recomienda:

- Implementar nuevas funcionalidades orientadas a la conexión que hagan uso de la base de datos utilizada por el SIGUCI web.
- Activar las funcionalidades de trabajo con GPS y de Edición de Capa para que los usuarios hagan un mejor uso de la herramienta.

1. (ESRI), E. S. R. I. 2010. How do I work with the maps in ArcGIS.com? Disponible en: www.arcgis.com. [Consultado el: 22-10-2010]
2. ---. GIS Dictionary. Disponible en: <http://resources.arcgis.com/glossary>. [Consultado el: 22-10-2010]
3. MICHIGAN, U. D. General GIS Concepts. Disponible en: <http://www.umich.edu/~ipcaa/GIS/General%20GIS%20Concepts.htm>. [Consultado el: 24-10-2010]
4. LTD, E. W. P. 2000. Raster Data. Disponible en: <http://www.cadresources.com.au/vector.htm>. [Consultado el: 25-10-2010]
5. ---. 1999. Vector Data. Disponible en: <http://www.cadresources.com.au/vector.htm>. [Consultado el: 25-10-2010]
6. COLUMBIA, U. D. Georreferenciación de una imagen en ArcGIS. Disponible en: <http://www.columbia.edu/acis/eds/gis/georef.html>. [Consultado el: 27-10-2010]
7. (EPA), U. S. E. P. A. 2009. Georreferenciación. Disponible en: <http://water.epa.gov/type/rsi/datait/waters/georef/index.cfm>. [Consultado el: 28-10-2010]
8. DENTON, C. E. *Acceso a Internet Inalámbrico: WAP, WML y VoiceXML*. http://catarina.udlap.mx/u_dl_a/tales/documentos/msp/enriquez_d_c/capitulo4.pdf [Consultado el: 30-10-2010]
9. C., B. G. 2004. UDDI (Universal Description Discovery and Integration). Disponible en: <http://www.desarrolloweb.com/articulos/1589.php>. [Consultado el: 1-11-2010]
10. ---. 2004. Registro en UDDI. Disponible en: <http://www.desarrolloweb.com/articulos/1599.php>. [Consultado el: 1-11-2010]
11. (AESIGT), A. E. D. S. D. I. G. Y. T. *Diccionario glosario de términos SIG*. Madrid: 1993. p: [Consultado el: 1-11-2010]
12. ARONOFF, S. *Geographic Information Systems: A Management Perspective*. WDL Publications, 1991, [Consultado el: 2-11-2010].
13. BURROUGH, P. A. y MCDONNELL, R. *Principles of geographical information systems*. Nueva York: Oxford University Press, 1997. p: [Consultado el: 2-11-2010]

14. (DOE), D. O. E. y CHORLEY, R. *Handling Geographic Information*. HMSO 1987. p:
http://openlibrary.org/books/OL22033639M/Handling_geographic_information.

[Consultado el:2-11-2010]

15. (NCGIA), N. C. F. G. I. A. A. *National Center for Geographical Information and Analysis*. Santa Bárbara: Universidad de California, 1990. vol. I, p:

[Consultado el:3-11-2010]

16. PASCUAL., A. R. *“Proposición de una definición profunda de SIG”*. *Los Sistemas de Información Geográfica en el umbral del Siglo XXI*. Madrid: II Congreso de AESIG, 1993. p: 127-143.

[Consultado el:5-11-2010]

17. SUNIAGA, J. Introducción a los SIG y los datos espaciales. *¿Qué es un SIG?*
http://www.etisig.chaco.gov.ar/descargas/Que_es_un_SIG.pdf

[Consultado el:5-11-2010]

18. SANTOS, M. Á. S. Definición y Desarrollo Histórico del SIG - Tema 4.
<http://www.ggyrna.geo.ucm.es/docencia/documentos/sig/definicion.pdf>

[Consultado el:7-11-2010]

19. ---. Conceptos básicos sobre Sistemas de Información Geográfica (SIG).
http://www.ggyrna.geo.ucm.es/docencia/.../Informatica_9.pdf

[Consultado el:7-11-2010]

20. SILVA, J. L. B. 2005. APLICACIÓN DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA EN CUBA. Disponible en: http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=1051. [Consultado el:7-11-2010]

21. LABSIG y ALICANTE, U. D. Documentación relacionada con SIGUA. 2011, Disponible en: <http://www.sigua.ua.es/web/presenta/doc/documentacion.php>. [Consultado el: 10-11-2010].

22. MORATO, R. B.; CALLE, M. D. L., *et al.* SIGATEX: adaptación del SIG de la Consejería de Cultura y Turismo. Junta de Extremadura. 2011, Disponible en: http://biblioteca.universia.net/html_bura/ficha/params/title/sigatex-adaptacion-sig-consejeria-cultura-turismo-junta-extremadura/id/42946792.html. [Consultado el: 2-5-2011].

23. RIGEL. Movípolis Disponible en:
http://www.rigelmsp.com/index.php?option=com_content&view=article&id=17&Itemid=89&lang=es. [Consultado el:10-11-2010]
24. ---. Ventajas de Movípolis Disponible en:
http://www.rigelmsp.com/index.php?option=com_content&view=article&id=21&Itemid=92&lang=es. [Consultado el:10-11-2010]
25. NÚÑEZ, M. H. M. F. 2006. SIG-ESAC: Sistema de Información Geográfica para la gestión de la estadística de salud de Cuba Disponible en:
http://scielo.sld.cu/scielo.php?pid=S1561-30032006000300003&script=sci_arttext. [Consultado el:10-11-2010]
26. VALDIVIA, A. R. *Relación de dependencias Funcionalidades-plugins y plugins-plugins de la plataforma GeneSIG*. UCID. [Consultado el: 10-11-2010]
27. ZALDÍVAR, Y. P. y MONTERO, L. H. *Documento Visión Plataforma GeneSIG*. UCI. [Consultado el: 10-11-2010]
28. MURCIA, U. D. Metodologías de desarrollo de software.
<http://www.um.es/docencia/barzana/IAGP/lagp2.html#BM3>. [Consultado el:12-11-2010]
29. JACOBSON, I.; BOOCH, G., *et al.* *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 16.
[Consultado el:12-11-2010]
30. FOWLER, M. 2003. La Nueva Metodología. Disponible en:
<http://www.programacionextrema.org/articulos/newMethodology.es.html>. [Consultado el:12-11-2010]
31. CANÓS, J. H.; LETELIER, P., *et al.* Metodologías Ágiles en el Desarrollo de Software.
<http://www.willydev.net/descargas/prev/TodoAgil.pdf>. [Consultado el:13-11-2010]
32. ACUÑA, K. B. *SELECCIÓN DE METODOLOGÍAS DE DESARROLLO PARA APLICACIONES WEB EN LA FACULTAD DE INFORMÁTICA DE LA UNIVERSIDAD DE CIENFUEGOS*. Universidad de Cienfuegos, 2009.[Consultado el:13-11-2010]
33. MICROSOFT. 2010. Información general. Disponible en:
<http://www.microsoft.com/downloads/details.aspx?familyid=A71AC896-1D28-45A4-880C-8B0CC8265C63>. [Consultado el:14-11-2010]

34. JACOBSON, I.; BOOCH, G., *et al.* *El proceso unificado de desarrollo de software*. España: Addison Wesley, 2000. p: 30.

[Consultado el:14-11-2010]

35. ---. *El proceso unificado de desarrollo de software*. España: Addison Wesley, 2000. p: 32.

[Consultado el:14-11-2010]

36. ---. *El proceso unificado de desarrollo de software*. España: Addison Wesley, 2000. p: 31.

[Consultado el:14-11-2010]

37. SWANSON, T. Mobile GIS Solutions. 2009, Disponible en: http://proceedings.esri.com/library/userconf/feduc09/papers/techworkshops/tw_arcgis-mobile-feduc09-fordist.pdf. [Consultado el: 15-11-2010].

38. (SOLGRAF), S. G. S. A. 11/2005. IntelliWhere OnDemand. Disponible en: <http://www.solgrafperu.com/productos/intelliwhere/ondemand.htm>. [Consultado el:15-11-2010]

39. GVSIG, A. 2009. GvSIG Mini Disponible en: <http://www.gvsig.com/productos/gvsig-mini>. [Consultado el:15-11-2010]

40. ---. 2009. GvSIG Mobile Disponible en: <http://www.gvsig.com/productos/gvsig-mobile>. [Consultado el:15-11-2010]

41. REY, A. D.; MONTESINO, M., *et al.* 2008. GvSIG Mobile como cliente móvil de IDEs. Disponible en: http://www.idee.es/resources/presentaciones/JIDEE08/ARTICULOS_JIDEE2008/articulo15.pdf. [Consultado el:15-11-2010]

42. ELORANTA, L.; KALLIO, E., *et al.* 2006. Una evaluación de la notación BPMN y diagramas de actividad UML. http://www.soberit.hut.fi/T-86/T-86.5161/2006/BPMN_vs_UML_final.pdf. [Consultado el:16-11-2010]

43. JACOBSON, I.; BOOCH, G., *et al.* *El Proceso unificado de Desarrollo de Software*. 2000. p: 422.

[Consultado el:16-11-2010]

44. GÓMEZ, O. S. G. Desarrollo de aplicaciones para dispositivos móviles utilizando J2ME. [Consultado el:17-11-2010]

45. W3C. 2007. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. Disponible en: <http://www.w3.org/TR/wsdl20/>. [Consultado el:17-11-2010]
46. ---. 2010. Guía Breve de Servicios Web. Disponible en: <http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>. [Consultado el:17-11-2010]
47. (GSI), G. D. S. I. 2007. Rational Rose Enterprise Disponible en: <http://www.rational.com.ar/herramientas/roseenterprise.html>. [Consultado el:16-11-2010]
48. LTD., V. P. I. 2010. Visual Paradigm for UML - Herramienta de UML para el desarrollo de software de aplicación. Disponible en: <http://www.visual-paradigm.com/product/vpuml/>. [Consultado el:16-11-2010]
49. ECLIPSE, F. 2010. Eclipse IDE para desarrolladores de Java. Disponible en: <http://www.eclipse.org/europa/>. [Consultado el:17-11-2010]
50. CARRERA, M. 2010. System requirements. Disponible en: <http://www.gvsig.org/web/projects/gvsig-mobile/official/piloto-gvsig-mobile-0.3/system-requirements>. [Consultado el:15-1-2011]
51. ---. 2010. Previous requirements. Disponible en: <http://www.gvsig.org/web/projects/gvsig-mobile/official/piloto-gvsig-mobile-0.3/previous-requirements>. [Consultado el:15-1-2011]
52. JACOBSON, I.; BOOCH, G., *et al.* *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 189.
[Consultado el:15-2-2011]
53. ---. *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 195.
[Consultado el:15-2-2011]
54. ---. *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 201.
[Consultado el:20-2-2011]
55. ---. *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 230.
[Consultado el:22-2-2011]
56. ---. *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 231.
[Consultado el:22-2-2011]
57. ---. *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 239.

[Consultado el:28-2-2011]

58. ALCALÁ, U. D. El formato Shapefile. Disponible en: http://www.geogra.uah.es/gisweb/practica-vectorial/Formato_Shapefile.htm. [Consultado el:5-3-2011]

59. JACOBSON, I.; BOOCH, G., *et al.* *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 276.

[Consultado el:3-5-2011]

60. GUERRERO, L. A. Arquitectura física: Diagramas de Componentes y Diagramas de Despliegue. *Taller de UML*.

61. JACOBSON, I.; BOOCH, G., *et al.* *El Proceso unificado de Desarrollo de Software*. Madrid: 2000. p: 301.

[Consultado el: 3-5-2011]

Glosario de Términos.

Canvas: Es el área donde se dibujan las diferentes capas que conforman un mapa.

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

Descripción Universal de Descubrimiento e Integración (Universal Description Discovery and Integration, UDDI): Directorio que provee interfaces para la utilización de servicios web. Permite la transferencia de información utilizando estándares como XML, SOAP, HTTP.

Interfaz de Programación de Aplicaciones (Application Programming Interface, API): Conjunto de rutinas, protocolos, y de herramientas para construir aplicaciones de software. Un API hace más fácil desarrollar un programa, proporcionando todos los bloques de la construcción.

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

J2ME (Java 2 Micro Edition): Versión desarrollada por la Sun Microsystems de Java, destinada a dispositivos de recursos limitados como PDA, teléfonos móviles, etc.

PDA (Personal Digital Assistant): Ordenador de pequeño tamaño cuya principal función era, en principio, mantener una agenda electrónica.

Protocolo de Aplicaciones Inalámbricas (Wireless Application Protocol, WAP): Protocolo de aplicación de telefonía inalámbrica que permite a los usuarios de teléfonos móviles el acceso a servidores web.

WSDL: Lenguaje de descripción de servicios Web (WSDL por sus siglas en inglés). Como su nombre lo indica permite la descripción del cuerpo del servicio, cumple la función de “acuerdo” entre el cliente y el servicio.