

República de Cuba



Universidad de las Ciencias Informáticas

Facultad 1

“Centro de Gestión de Emergencias y Seguridad Ciudadana (171)”

“Módulo de Mapificación de la Información”

Trabajo de Diploma

presentado para optar por el título de

Ingeniero Informático

Autor: Wilfredo Ríos Milanés

Tutor: Ing. Yordanis Tornes Medina

“Año de la Revolución Energética en Cuba”

Ciudad de la Habana, Cuba.

2006

Dedicatoria:

...A mis Padres

Agradecimientos.

A mis padres y a mi abuela, por el cariño y el amor que me han brindado toda la vida, por ser mis principales educadores, por ser el motivo de inspiración de llevar todo este sueño adelante y poder siempre contar con ellos.

A mi Hermano, por ser mas que mi sangre, por ser mi orgullo y para que tome el ejemplo y algún día me de la alegría de invitarme a su tesis.

A mi tío Rolandito, a mi tía Elena, Evita, Nivia y a toda mi familia por siempre preocuparse por mí, por brindarme su amor y apoyo incondicional.

A mis amigos Mark, Henry, Landor y Tamarita, que tanto han luchado porque este día fuera realizado y sin su apoyo nada de esto fuera realidad, por ser mis compañeros ante cualquier situación y por enseñarme el verdadero significado de la amistad.

A Ismary, por siempre estar a mi lado, por brindarme su ayuda en todo momento y por ser la persona con la que he compartido muy buenos momentos, te agradezco de todo corazón permitirme contar contigo.

A Mongui y Alexis, por darme todo su apoyo, sus consejos y por mostrarme el camino que me hizo llegar hasta acá.

A mi tutor Yordanis, por su consagración y apoyo para que este trabajo resultara lo mejor posible.

A Yisel, por estar en todo momento, por su paciencia y por ofrecerme sus conocimientos incondicionalmente.

A Meneses, mi amigo y compañero de batalla en esta lucha, gracias por estar acá.

A Molina, a Julián, a Alexander y a todos mis compañeros de cuarto, por ofrecerme su amistad, por mostrar siempre su interés y preocupación ante cualquier situación y por compartir buenos y malos momentos.

A mis compañeros de aula, por haber tenido la posibilidad de compartir estos años con ellos y haber conocido personas maravillosas.

A Adrián, Yahima, Daniel y Adonis por ofrecerme su ayuda en todo momento.

A todos los compañeros del proyecto, por su interés para que todo saliera bien y por darme su amistad.

A la Universidad de Oriente, a la Universidad de las Ciencias Informáticas y a la Revolución, por educarnos y prepararnos un poco mejor ante la vida, con el simple objetivo de convertirnos en mejores personas.

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas para que haga el uso que estime pertinente con el mismo.

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

Wilfredo Ríos Milanes

Ing. Yordanis Tornes Medina

Firma del Autor

Firma del Tutor

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: **Centro de Gestión de Emergencias y Seguridad Ciudadana (171). Módulo de Mapificación de la Información.**

Autor: **Wilfredo Ríos Milanes**

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia**
- Originalidad**
- Creatividad**
- Laboriosidad**
- Responsabilidad>**

<Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de **<nota>**. **<Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>**

Ing. Yordanis Tornes Medina

Firma

Fecha

Resumen.

La seguridad ciudadana es uno de los principales deberes desde el punto de vista social que todo gobierno debe garantizar, una forma de lograr tal propósito es hacer que las demandas de emergencias efectuadas por la población sean atendidas de manera rápida y efectiva por los órganos de seguridad del estado. Actualmente, en el Área Metropolitana de la República Bolivariana de Venezuela, no existe un centro de gestión de emergencias y seguridad ciudadana automatizado que integre a los órganos de seguridad, que atienda de manera urgente y eficiente las emergencias solicitadas y visualice en un mapa digital del área, los incidentes que ocurran tanto en tiempo real como con carácter histórico. Para dar respuesta a esta problemática se promueve la realización del **Centro de Gestión de Emergencias y Seguridad Ciudadana 171 (Centro 171)**, en su conjunto estará compuesto por varios sistemas de tecnología actual y un sistema informático que gestionará el buen funcionamiento del mismo. Esta aplicación informática está integrada por varios módulos: Módulo de Administración, Módulo de Mapificación de la Información, entre otros.

El principal objetivo de este trabajo es analizar, diseñar e implementar un Sistema de Información Geográfico (SIG) como parte del sistema informático del Centro 171, Módulo de Mapificación de la Información.

La implementación de un SIG para este sistema es necesaria porque permite ubicar geográficamente en un mapa digital del área las emergencias, hechos, faltas, móviles con GPS y otras situaciones de interés que requieran su visualización, tanto en tiempo real como con carácter histórico. Posibilitando de esta forma una mejor apreciación de la situación en el territorio y por tanto la adopción de decisiones más precisas.

El Módulo de Mapificación de la Información contribuirá a que el Centro 171 brinde un mejor servicio a la población y servirá de ayuda para el cumplimiento de lo establecido en el artículo 322 de la Constitución Nacional Venezolana, que establece *“La organización de los órganos de seguridad ciudadana como medio para garantizar la protección de los ciudadanos y sus hogares en disfrute de los derechos fundamentales”* [7].

Índice.

Introducción.....	1
Capítulo I: Fundamentación Teórica.....	6
1.1 Introducción.....	6
1.2 Seguridad Ciudadana.....	6
1.3 Centros de Gestión de Emergencias.	6
1.3.1 Centros de Gestión de Emergencias en Venezuela.	7
1.4 Sistemas de Posicionamiento.	8
1.4.1 Sistema de Posicionamiento Global (GPS).	8
1.4.2 GLONASS.	9
1.4.3 Galileo.	9
1.5 Sistemas de Información Geográficos (SIG).....	10
1.5.1 Sistemas de Información Geográficos especializados en Gestión de Emergencias y Seguridad Ciudadana.	11
1.6 Metodología utilizada.	11
1.6.1 Proceso Unificado de Desarrollo de Software (RUP).	12
1.6.2 Lenguaje de Modelado Unificado (UML).	13
1.7 Microsoft .NET.	13
1.7.1 Visual Studio .NET 2003.....	14
1.7.2 Lenguajes.	14
1.8 Plataforma Java.	16
1.8.1 Java.	16
1.10 Sistemas Gestores de Bases de Datos (SGBD).	17
1.10.1 Microsoft SQL Server.	17
1.10.2 Oracle.	18
1.11 Bases de Datos Geográficas (BDG).	19
1.11.1 Oracle Spatial.	19
1.12 MapInfo.	20
1.12.1 MapInfo MapXtreme 2004.....	21
1.13 Conclusiones.....	22
Capítulo II: Descripción de la solución propuesta.....	23
2.1 Introducción.....	23
2.2 Modelo de Dominio.	23
2.3 Requerimientos.	25
2.3.1 Requerimientos funcionales.	25
2.3.2 Requerimientos no funcionales.	30
2.4 Descripción de la solución propuesta.....	32
2.5 Modelo del Sistema.....	34
2.5.1 Modelo de Casos de Uso del Sistema.....	34
2.5.2 Diagrama de Casos de Uso del Sistema.....	37

2.5.3 Casos de Uso expandidos.....	40
2.6 Conclusiones.....	55
Capítulo III: Construcción de la solución propuesta.....	56
3.1 Introducción.....	56
3.2 Arquitectura propuesta para la implementación del sistema informático del Centro 171. ...	56
3.2.1 Enfoque Horizontal.....	56
3.2.2 Enfoque Vertical.....	58
3.3 Diagrama de Clases del diseño.....	59
3.3.1 Diagrama de Clases del paquete Servicio.....	59
3.3.2 Diagrama de clases del paquete Mapa.....	60
3.3.3 Diagrama de clases Interacción entre Paquetes.....	61
3.4 Diseño de la Base de Datos.....	61
3.4.1 Descripción de las Tablas.....	61
3.5 Principios de Diseño de la Aplicación.....	62
3.5.1 Estándares en la interfaz de la aplicación.....	63
3.5.2 Estándares de codificación.....	63
3.5.3 Tratamiento de errores.....	68
3.6 Conclusiones.....	70
Capítulo IV: Implementación.....	71
4.1 Introducción.....	71
4.2 Modelo de Despliegue.....	71
4.3 Modelo de Implementación.....	72
4.3.1 Diagrama de Componentes del paquete Servicio.....	72
4.3.2 Diagrama de Componentes del paquete Mapa.....	73
4.4 Conclusiones.....	74
Conclusiones.....	75
Recomendaciones.....	76
Referencias Bibliográficas.....	77
Bibliografía.....	79
Anexos.....	82
Glosario de Términos.....	87

Introducción.

La seguridad de los ciudadanos debe ser atendida de acuerdo al mandato establecido en el artículo 55 de la Constitución de la República Bolivariana de Venezuela, el cual establece: *“Toda persona tiene derecho a la protección del estado, a través de los Órganos de Seguridad Ciudadana regulados por Ley, frente a situaciones que constituyan amenazas, vulnerabilidad o riesgos para la integridad física de las personas, sus propiedades, el disfrute de sus derechos y el cumplimiento de sus deberes”* [11].

El Área Metropolitana de La República Bolivariana de Venezuela en la actualidad carece de un centro automatizado que integre a todos los órganos de seguridad ciudadana y que atienda con efectividad y rapidez las demandas de emergencias formuladas por la población.

Actualmente existen varios números telefónicos para reportar emergencias, ofrecidos por empresas privadas de telecomunicaciones, gobernaciones y alcaldías. Algunos de estos centros funcionan las 24 horas, los 365 días del año, pero la mayoría sólo responden en horas de oficina, ninguno está interconectado entre sí, actuando como centros independientes, lo que demora la respuesta y repercute en el bienestar y seguridad de la población. Tampoco existe una aplicación que muestre en un mapa digital de la región Metropolitana las emergencias, hechos, faltas, móviles con GPS y otras situaciones de interés que requieran su visualización, tanto en tiempo real como con carácter histórico, es decir, carecen de un SIG que los ayude a darle una respuesta más eficiente a la población.

Para darle respuesta a estas situaciones el Ministerio del Interior y Justicia, atendiendo a su misión institucional de garantizar la seguridad ciudadana, promueve la formulación y puesta en marcha del **Centro de Gestión de Emergencias y Seguridad Ciudadana 171 (Centro 171)**.

El Centro 171 estará compuesto por tecnología novedosa de Telecomunicaciones, grabación de audio y video digital, sistemas de posicionamiento global, etc. Además contará con un sistema automatizado que será el encargado de integrar cada uno de los sistemas tecnológicos y gestionar el buen funcionamiento de estos. Por la diversidad de roles y de tareas que se realizan en el proceso de atención de las emergencias, la aplicación informática consta de los siguientes módulos:

- Módulo de Operadores
- Módulo de Despachadores
- Módulo de Administración

- Módulo de Comunicaciones
- Módulo de Video Vigilancia
- Módulo de Análisis Estadístico
- Módulo de Mapificación de la Información
- Módulo de Ubicación de los Móviles por GPS
- Módulo de Control y Administración de Recursos

En una primera parte, el Centro 171 en su conjunto va a brindar servicios y facilidades priorizando los siguientes aspectos:

- Capacidad de recepción de la información y denuncias de la población u otras fuentes.
- Despachar la atención de las emergencias hacia los órganos de atención directa, y garantizar su seguimiento hasta lograr la asistencia solicitada.
- Mantener un registro de información que permita a los órganos de seguridad ciudadana, consultar y realizar análisis de la situación delictiva y de emergencias tanto en tiempo real como con carácter histórico.
- Posibilitar el análisis geográfico de la actividad delictiva y de emergencias, así como la localización en tiempo real es histórico de los recursos en servicio.
- Activar un proyecto piloto de cámaras de televisión digital operativa en áreas de interés ubicadas en el Distrito Metropolitano, para evaluar la factibilidad de implementación en etapas posteriores hacia el resto de los municipios.
- Acceso desde el centro a información de interés disponible en otros organismos.
- Disponer de un sistema de radiocomunicaciones TETRA que facilite la intercomunicación entre los diferentes cuerpos policiales de la región y el centro de control.
- Disposición de una sala situacional que permita la atención de emergencias en caso de catástrofes y desastres naturales.

El Módulo de Mapificación de la Información es imprescindible para que el Centro 171 brinde los servicios antes planteados, pues es el encargado de:

- Visualizar todos los sitios de interés, como por ejemplo centros hospitalarios, cuerpos de bomberos, centros educacionales, comisarías de la policía, entre otros.
- Mostrar los Móviles en Servicio y garantizar el seguimiento de las unidades destinadas a un incidente hasta que se logre la asistencia solicitada.
- Visualizar las emergencias que requieran de la atención de una entidad encargada de la seguridad ciudadana.
- Mostrar en un mapa temático del área las estadísticas de los reportes. Esto permite consultar y realizar análisis de la situación delictiva y de emergencias tanto en tiempo real como con carácter histórico y por tanto la adopción de decisiones más precisas.
- Mostrar el mapa digital del Área Metropolitana u otro de interés. Por defecto se mostrará automáticamente desde un lugar centralizado, pudiendo ser desde una Base de datos o un dispositivo de almacenamiento local, permitiendo de esta forma que todos utilicen el mismo mapa y la versión más actualizada del mismo.
- Salvar una vista de lo que está visible en el mapa, pudiendo ser en un dispositivo local de almacenamiento o en un lugar centralizado. Esto permite predefinir una situación excepcional o de interés, y después visualizarlas cuando el usuario lo estime conveniente.
- Comunicarse con varios módulos y recibir información de estos, del Módulo de Operadores recibe la dirección de la emergencia que inmediatamente se muestra en el mapa digital, del Módulo de Administración se recibe la notificación de que se actualizaron los sitios de interés, y si se están mostrando se informa al usuario que se van a actualizar.

Para darle solución a los problemas antes planteados se definió como objetivo general de este trabajo analizar, diseñar e implementar un Sistema de Información Geográfico (SIG) como parte del sistema informático del Centro 171, Módulo de Mapificación de la Información. Para dar cumplimiento al mismo se definieron los siguientes objetivos específicos:

- Efectuar sobre el mapa digital cualquier acción básica, por ejemplo: Acercar, Alejar, Centrar, Mover, Seleccionar, etc.
- Permitir la búsqueda de cualquier elemento georreferenciado.
- Visualizar la ubicación de todos los Móviles en Servicio, la posición de cualquier sitio georreferenciado y mostrar cada hecho en tiempo real y con carácter histórico.
- Almacenar en una Base de Datos Geográfica los mapas, las vistas y los reportes diarios.

El cumplimiento de los objetivos antes planteados permitirá agilizar la respuesta de los órganos de seguridad ciudadana ante una situación de emergencia, pues hará posible que ante un hecho: el agente de seguridad ciudadana más cercano al lugar sea quien preste la atención debida, informar el Hospital más cercano para llevar un herido, etc. También ante situaciones de incendio se puede informar a los cuerpos de bomberos donde se encuentran los hidrantes o los locales que puedan contribuir a la expansión del fuego e impliquen algún peligro para la población como las estaciones de gasolina, etc.

En la concepción de este Módulo se plantearon un grupo de tareas que permitiesen realizar una implementación satisfactoria del sistema, las cuales son:

- Investigar sobre las herramientas a utilizar para el desarrollo de un SIG, así como el proceso de instalación, configuración y utilización de estas.
- Evaluar cada herramienta disponible y las posibles alternativas según su funcionalidad y desempeño observado.
- Diseñar, implementar y evaluar una serie de pruebas que permitan fortalecer el conocimiento sobre la programación y la comunicación entre los distintos componentes de información georreferenciada.
- Reunir los conocimientos y la experiencia de la realización de las pautas anteriormente planteadas y seleccionar un prototipo final.

El presente documento está estructurado en: resumen, introducción y desarrollo, el cual está compuesto por 4 capítulos:

- Capítulo I denominado “**Fundamentación Teórica**”, se incluyen todos los aspectos teóricos que soportan este proyecto y se analizan algunas de las herramientas y lenguajes de programación más utilizadas en el mundo para el desarrollo de las aplicaciones, con esta información se definen

las que se usarán para realizar esta aplicación. Además se plantea la metodología a seguir en la elaboración de la misma.

- Capítulo II denominado “**Descripción de la solución propuesta**”, donde se definen las reglas del negocio y los conceptos fundamentales del Módulo de Mapificación de la Información. Se determinan además los requerimientos funcionales y no funcionales del mismo, agrupándolos en casos de usos y se da una idea general del funcionamiento de esta aplicación.
- Capítulo III denominado “**Construcción de la solución propuesta**”, donde se determinan las clases que se utilizarán en la implementación del sistema y la relación entre ellas. Además se muestra todo el proceso de obtención de la Base de Datos y se definen los principios de diseño de interfaz, codificación y tratamientos de errores.
- Capítulo IV denominado “**Implementación**”, aquí se definen la topología de hardware sobre la que se sustenta la aplicación y sus componentes, se verifica su integridad y ajuste a los requerimientos planteados.

También está compuesto por las conclusiones donde se realiza un breve resumen del resultado obtenido durante el transcurso del trabajo, las recomendaciones que abordan las posibles mejoras al sistema y los aspectos a profundizar, se incluye un glosario de términos definido para el sistema en el cual se añaden aquellos términos que se utilizan en el documento y no son comunes, las referencias bibliográficas y la bibliografía que no son más que una lista de libros, artículos, documentos, sitios en Internet, etc. que han sido utilizados como referencia durante el desarrollo de este trabajo. Por último se encuentran los anexos, y documentos que complementan el cuerpo del trabajo.

Capítulo I: Fundamentación Teórica.

1.1 Introducción.

En este capítulo se analizan aspectos teóricos, que en algún momento en la concepción de todo proyecto fue necesario investigar.

Se abordan aspectos relacionados con el uso de las nuevas tecnologías de la informática y las comunicaciones. Se explica la importancia de la seguridad ciudadana y de los centros de gestión de emergencia; específicamente los que existen en la República Bolivariana de Venezuela y el Área Metropolitana de Caracas.

También es objeto de interés analizar las ventajas y características que presentan los Sistemas de Información Geográficos y las herramientas que facilitan la realización de los mismos, por ser estos elementos de vital importancia en la toma de decisiones y para la elaboración del sistema.

Se determinará la metodología a utilizar para el análisis y diseño de la aplicación, definiéndose la notación y los artefactos a utilizar.

1.2 Seguridad Ciudadana.

Debe entenderse como Seguridad Ciudadana el grado de respeto que se otorga al conjunto de derechos de los ciudadanos, no solo por parte del Estado, sino también de parte de las personas e instituciones públicas y privadas, que tienen que garantizar el bienestar de todos los componentes de las estructuras a las cuales están vinculadas, a las cuales prestan servicios o de las que depende su seguridad.

1.3 Centros de Gestión de Emergencias.

Los centros de gestión de emergencias son un servicio que se brinda a la población con el objetivo de ofrecer soluciones efectivas a las situaciones problemáticas de la sociedad, y con esto brindar mayor confianza en la seguridad ciudadana de cada individuo. Para lograr tales expectativas estos centros deben ser un ente integrador de los organismos de seguridad y emergencias, encargado de recibir las llamadas de auxilio de la ciudadanía las 24 horas del día durante los 365 días del año, manteniendo un servicio de

comunicaciones que permita garantizar la adecuada supervisión y capacidad de respuesta de los organismos.

En la actualidad esto se realiza con la ayuda de sistemas automatizados que se encargan de realizar la mayoría de las funciones de manera inmediata, estos se llaman Sistemas de Gestión de Emergencias.

Estos sistemas cuentan con diferentes subsistemas de cómputo, telefonía, radio e información operativa, ofreciendo a la población beneficios como la disminución del tiempo de respuesta a las demandas de emergencias formuladas, detectar la localización de las llamadas e informar a la(s) institución(es) del estado que corresponde atender el incidente. También visualizan toda la región que se atiende, los hechos que ocurren, cada llamada que se reciba, cuales móviles están en servicio, entre otras. Así se brinda un mejor servicio y contribuye de una forma u otra a que los ejecutivos tomen mejores decisiones para el bienestar y la seguridad del pueblo.

1.3.1 Centros de Gestión de Emergencias en Venezuela.

Hoy en día, en la república Bolivariana de Venezuela existen varios estados que cuentan con centros que poseen sistemas de emergencia, algunos de estos funcionan las 24 horas de los 365 días del año, la mayoría sólo responde en horas de oficina, no están interconectado entre sí, por lo que actúan como centros independientes. Tampoco presentan un sistema que visualice en un mapa digital de la ciudad o del estado una información detallada de los incidentes que ocurren a diario o con anterioridad. Todo esto demora la respuesta por parte de los órganos de seguridad y repercute en el bienestar, la confianza y seguridad de la población.

En la capital de este país (Caracas) actualmente no existe ningún centro de gestión de emergencias que funcione como es debido, los que están funcionando no integran a los órganos de seguridad, demoran su respuesta y por tanto resultan ineficientes a la hora de otorgar una respuesta a un ciudadano.

A continuación se mencionan algunos de los centros que funcionan actualmente en Venezuela:

- Sistema Integral de Emergencia de Táchira 171.
- Alcaldía Metropolitana 864-7191
- Alcaldía Libertador 545-4513 / 542-1711 / 409-8632
- Alcaldía Sucre 237-6343 / 271-0253 / 272-3360 / 0-800-76547

1.4 Sistemas de Posicionamiento.

Desde siempre el ser humano ha tratado de encontrar la forma de orientarse y saber el lugar exacto de su localización. Para resolver este problema se han buscado distintas vías de solución como orientarse por las constelaciones, por la posición del sol, etc. Hoy en día gracias a la tecnología, ya no se trata de buscar estrellas que nos sitúen el Norte, sino de localizar satélites que están en órbita sobre la Tierra y nos indican nuestra posición exacta. Eso se consigue gracias a los receptores de los sistemas de navegación por satélite que permiten conocer con gran precisión dónde nos encontramos.

1.4.1 Sistema de Posicionamiento Global (GPS).

Una de las maneras más utilizadas para capturar información geográfica e incorporarla a un Sistema de Información Geográfico lo constituye el Sistema de Posicionamiento Global. El término GPS procede del acrónimo de la expresión inglesa '**Global Positioning System**' (Sistema de Posicionamiento Global). Se trata de un sistema que permite calcular las coordenadas de cualquier punto de la superficie terrestre a partir de la recepción de señales emitidas desde una constelación de satélites en órbita. Básicamente, su principal funcionalidad es que permite al usuario conocer, mediante un receptor, su posición en cualquier parte del planeta [10].

El sistema GPS funciona gracias a la constelación NAVSTAR, formada por 24 satélites situados a unos 20.000 kilómetros de altura, mantenidos por la fuerza aérea estadounidense. Estos satélites llevan a bordo un reloj atómico de gran precisión y emiten señales personalizadas que indican la hora en que partieron. El receptor en tierra, reconoce el satélite que ha emitido la señal, determina el tiempo que ha tardado en llegar y así calcula la distancia que le separa de él. Una vez que el receptor GPS ha contactado con al menos cuatro satélites es capaz de determinar su posición y altura.

Un receptor GPS normal presenta un margen de error entre 3 y 15 metros [1]. Para algún usuario que requiera un uso normal de este equipo esta diferencia no es muy significativa, pero para un sistema de urgencias como el nuestro, en situaciones excepcionales, este margen de error puede significar una catástrofe, por lo cual debe ser prácticamente nulo.

El GPS Diferencial (DGPS) reduce el margen de error a menos de un metro de diferencia con la posición indicada. Este tipo de receptor, además de recibir y procesar la información de los satélites, recibe y procesa, simultáneamente, otra información adicional procedente de una estación terrestre situada en un

lugar cercano y reconocido por el receptor. Esta información complementaria permite corregir las inexactitudes que se puedan introducir en las señales que el receptor recibe de los satélites. En este caso, la estación terrestre transmite al receptor GPS los ajustes que son necesarios realizar en todo momento, éste los contrasta con su propia información y realiza las correcciones mostrando en su pantalla los datos correctos con una gran exactitud [4].

1.4.2 GLONASS.

La segunda alternativa al GPS estadounidense la constituye el sistema de satélites de navegación global GLONASS (**GLO**bal **NA**avigation **S**atellite **S**ystem) de administración rusa, cuyas funciones son similares a las del GPS. Al igual que el sistema estadounidense, el GLONASS tiene aplicación tanto en el campo militar como en el civil, aunque en este último su uso es bastante limitado. El control de este sistema lo ejerce el gobierno de la Federación Rusa por mediación de las fuerzas espaciales. El sistema GLONASS comenzó a operar oficialmente el 24 de septiembre de 1993. Este sistema se compone de 24 satélites (21 activos y 3 de reserva), los cuales están distribuidos en sus respectivas órbitas de forma tal que siempre existen entre 4 ó 5 de ellos a la vista de los receptores, cubriendo el 97% de toda la superficie terrestre. Existen actualmente receptores duales que trabajan tanto con el GPS como con el GLONASS [4].

1.4.3 Galileo.

La tercera alternativa de posicionamiento global es el sistema Galileo, controlado por la Unión Europea y que se espera esté en explotación en el año 2008. Este sistema, actualmente en fase de desarrollo por la Agencia Espacial Europea, rinde honor con su nombre al famoso físico y astrónomo italiano Galileo Galilei (1564-1642).

El principio de funcionamiento del sistema europeo será idéntico al sistema norteamericano. Estará formado por 30 satélites geoestacionarios distribuidos en tres órbitas circunferenciales situadas aproximadamente a 24 mil kilómetros de altura sobre la Tierra. De ese total de satélites en órbita se encontrarán siempre operativos 27, mientras los 3 restantes se mantendrán en reserva.

Una diferencia sustancial entre el Galileo, comparado con el GPS y el GLONASS, es que su origen es completamente civil y no estará controlado por un solo país, sino por todos los países que integran la Unión Europea.

La Agencia Espacial Europea prevé que el sistema Galileo sea mucho más preciso que el GPS teniendo en cuenta la tecnología de los satélites de nueva generación y los sistemas de control que se utilizarán desde Tierra [4].

1.5 Sistemas de Información Geográficos (SIG).

Hoy en día los Sistemas de Información Geográficos (SIG) son una herramienta de software de uso masivo. Los tipos de aplicaciones y análisis que pueden realizarse por un SIG son tan amplias y diversas como los conjuntos de datos geográficos disponibles.

Definición.

Un Sistema de Información Geográfico (SIG) es un Software que nos permite el almacenamiento, recuperación, análisis y despliegue de información geográfica digital.

Un SIG consta principalmente de 4 componentes: Software, Hardware, base de datos y personal.

- **Software:** Es el encargado de llevar a cabo de manera eficiente las operaciones de almacenamiento, recuperación, análisis y despliegue de la información geográfica. Esta componente contiene 4 aspectos principales, que son:
 - Herramientas para la entrada y manipulación de la información geográfica.
 - Herramientas que soportan consultas, análisis y visualización de informaciones geográficas.
 - Sistema manejador de base de datos geográficos.
 - Una interfaz gráfica amigable para el usuario y de fácil acceso a las herramientas de trabajo.
- **Hardware:** Está compuesto por todos los dispositivos externos sobre los cuales opera el sistema. Por ejemplo el CPU, el monitor, el teclado, el servidor de Bases de Datos, etc.
- **Base de datos:** Las bases de datos en los SIG son llamadas bases de datos geográficas. Ofrecen las mismas funcionalidades que las base de datos pero además soportan un nuevo tipo de datos

para trabajar con los datos geométricos. Estas son la esencia del sistema y no es más que una colección de datos acerca de objetos localizados en un área determinada.

De igual forma se permite acceder a información geográfica que esté ubicada fuera de una Base de Datos, en estos casos la información está distribuida en ficheros, cuando son accedidos desde el sistema su tratamiento es igual al empleado cuando se accede desde una Base de Datos, esto es debido a que el almacenamiento interno de estos ficheros responde, analógicamente, a la estructura de las tablas de un Modelo Relacional.

- Personal: Es el encargado de operar, desarrollar y administrar el sistema. Además realiza análisis complejos siguiendo los criterios impuestos en la realización del Software y de toma de decisiones.

1.5.1 Sistemas de Información Geográficos especializados en Gestión de Emergencias y Seguridad Ciudadana.

El SIG muestra un mapa de la zona que se requiera atender; sobre esta área se van a representar tanto las llamadas fijas y de móviles que se reciben en el centro, como la ubicación donde ha ocurrido el incidente, la ubicación de las agencias (Policías, Bomberos, etc.) encargadas de tratar los incidentes y los recursos en servicio (vehículos) de cada una de las agencias. Aparte de una mera visualización de información, desde el SIG se podrán realizar operaciones relacionadas con el proceso de atención de llamadas diario que se realiza en el centro, como por ejemplo, comunicarse con una patrulla y brindarle alguna información. Además ofrece reportes de hechos o incidentes que permiten a los órganos de seguridad ciudadana consultar y realizar análisis de la situación delictiva y de emergencias, tanto en tiempo real como con carácter histórico, de esta forma se pueden tomar decisiones más rápidas y adecuadas para el bienestar de la sociedad y la tranquilidad del pueblo.

1.6 Metodología utilizada.

Actualmente los sistemas que se construyen son cada vez más grandes y complejos, se quiere un software que este mejor adaptado a nuestras necesidades y en el menor tiempo posible.

Sin embargo la mayoría de los desarrolladores de hoy en día hacen sus sistemas con los mismos métodos de años atrás donde muchos proyectos fracasaban, sin percatarse que el elemento que hace que un software triunfe es tener un proceso bien definido y bien gestionado.

Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software [8] (Ver figura 1.1).

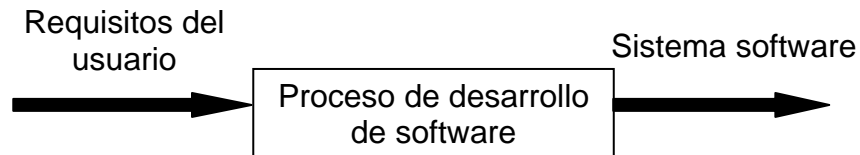


Figura 1.1. Proceso de desarrollo de software.

1.6.1 Proceso Unificado de Desarrollo de Software (RUP).

El Proceso Unificado es más que un simple proceso; es un marco genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organización, diferentes niveles de aptitud y diferentes tamaños de proyectos [8].

Los verdaderos aspectos definitorios del Proceso Unificado se resumen en 3 frases claves: Dirigido por Casos de Uso, Centrado en la Arquitectura, Iterativo e Incremental. Esto es lo que hace único al Proceso Unificado [8].

- Dirigido por Casos de Uso.
Para construir un sistema con éxito debemos conocer lo que sus futuros usuarios necesitan y desean [8]. Un caso de uso es un servicio que el actor requiere del sistema y este le proporciona un resultado. Representan los requerimientos funcionales de la aplicación. Los casos de uso guían el diseño, la implementación y prueba, es decir, guían el proceso de desarrollo.
- Centrado en la Arquitectura.
La arquitectura en un sistema de software se describe mediante diferentes vistas del sistema en construcción. Esta se refleja en los casos de uso pues cada uno describe una funcionalidad y una forma.
- Iterativo e Incremental.

Es mucho más sencillo y práctico dividir el trabajo en pequeños módulos, los que no son más que iteraciones que resultan en un incremento.

Una iteración es una secuencia de actividades con un plan establecido y criterios de evaluación, cuyo resultado es una versión del software

Beneficios de la iteración:

- Reduce el coste del riesgo al coste de un solo incremento.
- Menos riesgo de no sacar el producto al mercado en fecha.
- Acelera el ritmo de desarrollo.
- Las necesidades del usuario y correspondientes requisitos no pueden definirse completamente al principio. Se requieren iteraciones sucesivas.

1.6.2 Lenguaje de Modelado Unificado (UML).

Es un lenguaje de propósito general para el modelado orientado a objetos. UML es también un lenguaje de modelación visual que permite una abstracción del sistema y sus componentes, que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software, UML no es un lenguaje de programación.

Permite la modificación de todos sus miembros mediante estereotipos y restricciones. Un estereotipo nos permite indicar especificaciones del lenguaje al que se refiere el diagrama de UML. Una restricción identifica un comportamiento forzado de una clase o relación, es decir mediante la restricción estamos forzando el comportamiento que debe tener el objeto al que se le aplica.

UML es un Standard, su utilización es independiente del lenguaje de programación y de las características de los proyectos, ya que UML ha sido diseñado para modelar cualquier tipo de proyecto.

1.7 Microsoft .NET.

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software, de manera que los lenguajes de programación y modelo de componentes con los que hayan sido desarrollados puedan comunicarse y combinarse sin importar la plataforma, a ella se le denominó **plataforma .NET.**

Microsoft, para crear aplicaciones escalables y distribuidas ha publicado el Framework .NET SDK, que incluye las herramientas necesarias tanto para el desarrollo de aplicaciones como para su distribución y ejecución. Visual Studio.NET es la herramienta por excelencia que permite el desarrollo de aplicaciones desde una interfaz visual basada en ventanas.

1.7.1 Visual Studio .NET 2003.

Para .NET se distribuye el paquete Visual Studio .NET (VS.NET), con un sin fin de utilidades para el desarrollador.

Posee varios lenguajes, esto permite elegir el lenguaje más adecuado para la tarea a desarrollar. Proporciona eficaces herramientas para crear con rapidez las aplicaciones.

Los equipos de desarrollo pueden mantener el control de versiones y compartir código fuente y documentación. Se puede analizar el rendimiento y la escalabilidad de las aplicaciones. El .NET tiene una plataforma escalable y confiable para aplicaciones distribuidas por lo que los sistemas creados son seguros, confiables y de alto rendimiento.

1.7.2 Lenguajes.

VS.NET, nos ofrece la posibilidad de programar en lenguajes como Visual C++ 7.0, Visual C#, Visual Basic .NET, ASP.NET y JSCRIPT, (además de incluir ADO.NET y muchas herramientas), todos con la posibilidad de utilizar la BCL. A continuación se hace una breve explicación de algunos de los lenguajes más utilizados.

- **Visual Basic .NET (VB)**

De los lenguajes usados en versiones anteriores de VS.NET, éste ha sido el más beneficiado y el que más ha cambiado. Gracias a la BCL, VB ha adquirido características que nunca antes había presentado, por ejemplo: la programación multi-hilo. Se han quitado instrucciones típicas de Basic reemplazándolas por unas más intuitivas y se han agregado y/o modificado otras. Ahora las aplicaciones de Visual Basic.NET son mucho más robustas.

- **C#, (o Visual C#)**

Este es el nuevo lenguaje inspirado en C/C++, similar a Java. Este lenguaje tiene una sintaxis basada en C/C++ con una estructura similar a Java pero con características especiales que lo hacen muy estructurado, sencillo, poderoso y de alto nivel. Hace uso de las BCL lo cual sirve para cualquier otro lenguaje .NET.

Ofrece una serie de ventajas basadas fundamentalmente en aspectos tales como:

- Sencillez: Es un lenguaje simple y de fácil aprendizaje,
- Soporta el trabajo con punteros manteniéndolos en espacio de código que el usuario declara como unsafe (inseguro).
- Orientado a Objetos: Es un lenguaje diseñado para la implementación de software orientado a objetos. Implementa conceptos como la herencia, el tratamiento de estructuras, la abstracción, la herencia, el polimorfismo, la encapsulación, entre otros.
- Modernidad: Es un lenguaje joven, surgido en el año 2000 y muy similar al java.
- Distribuido: Está concebido para trabajar en un entorno conectado en red. Cuenta con una amplia biblioteca de clases para comunicarse mediante TCP/IP: HTTP, FTP, etc.
- Gestión automática de memoria.
- Seguridad de tipos.
- Instrucciones seguras.
- Sistema de tipos unificado.
- Extensibilidad de tipos básicos.
- Orientación a componentes.
- Extensibilidad de operadores.
- Extensibilidad de modificadores.
- Versionabilidad.
- Eficiencia.
- Compatibilidad.

1.8 Plataforma Java.

Posee dos componentes fundamentales: La Máquina Virtual Java (JVM) que es el intérprete Java y La Interfaz de Programación de Aplicaciones (API). El API Java es un conjunto de clases ya desarrolladas que ofrecen un gran abanico de posibilidades al programador.

Hoy en día esta plataforma ha evolucionado en concordancia con el avance tecnológico y se ha convertido en una de las plataformas de programación mas usadas por los desarrolladores. Su principal ventaja es que su entorno de desarrollo es independiente de la plataforma sobre la que se trabaje, es decir, sus aplicaciones son funcionales tanto en Linux como en Windows. Sin embargo Java no permite la interoperabilidad de múltiples lenguajes. Es posible compilar a bytecode desde múltiples lenguajes, pero no es posible alcanzar el nivel de interacción que se da en .NET. Además la intercomunicación entre aplicaciones es segura pero a su vez muy difícil, esto ocurre por el hecho de que tiene que cumplir con los protocolos de seguridad de los Sistemas Operativos sobre los que funciona. Actualmente existen distintas ediciones de la plataforma Java como por ejemplo: J2ME (Java2 Micro Edition), J2EE (Java2 Enterprise Edition) y J2SE (Java2 Standard Edition), esta última es la más usada por los desarrolladores.

1.8.1 Java.

Java es un lenguaje de programación surgido en 1991. Los creadores de Java se basaron en C++, pero eliminaron la mayoría de sus complejidades como por ejemplo: la herencia múltiple y la creación de punteros. Java presenta características que lo convierten en un lenguaje seguro, estándar y de alto nivel, algunas de las principales características se muestran a continuación:

- Orientado a Objetos.
- Distribuido.
- Interpretado.
- Robusto.
- Seguro.
- Portabilidad.
- Dinámico.

1.10 Sistemas Gestores de Bases de Datos (SGBD).

En 1964, se conciben los primeros SGBD, con estos se pretendió dar un viraje a los Sistemas de Archivos, los cuales se limitaban a la estructuración del almacenamiento físico de los datos. Con los SGBD se logró por medio de actividades integradas verlos físicamente en un solo almacenamiento, pero lógicamente se manipulan a través de esquemas compuestos por estructuras donde se establecen vínculos de integridad, métodos de acceso y organización física sobre los datos, permitiendo así obtener valores agregados de utilización.

En 1970 el Dr. Edgar F. Codd propuso el Modelo de Datos Relacional, este modelo es el que ha marcado la línea de investigación por muchos años y representa al mundo real mediante tablas relacionadas entre sí por columnas comunes. Viendo la necesidad de mejorar este estándar se desarrollaron los Sistemas Gestores de Base de Datos Relacionales (SGBDR) cuyas características hacen al sistema mucho más eficiente que los sistemas de manejo de archivos.

Un SGBDR es un conjunto de datos relacionados entre sí y un grupo de programas para tener acceso a esos datos, permitiendo concurrencia y recuperación. La persistencia de un SGBDR hace referencia a la conservación de los datos después de la finalización del proceso que los creó y la concurrencia se refiere a la capacidad del sistema para gestionar a múltiples usuarios interactuando al mismo tiempo. Entre los SGBDR más comunes se encuentran SQL Server y Oracle, a continuación se dará una breve explicación de los mismos.

1.10.1 Microsoft SQL Server.

SQL Server es un sistema de gestión de bases de datos relacionales (SGBDR) basada en el lenguaje SQL, capaz de poner a disposición de muchos usuarios, grandes cantidades de datos de manera simultánea.

Entre sus características figuran:

- Soporte de transacciones.
- Gran estabilidad.
- Gran seguridad.
- Escalabilidad.
- Soporta procedimientos almacenados.

- Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de otros servidores de datos

Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle o Sybase.

Es común desarrollar completos proyectos complementando Microsoft SQL Server y Microsoft Access a través de los llamados ADP (Access Data Project). De esta forma se completa una potente base de datos (Microsoft SQL Server) con un entorno de desarrollo cómodo y de alto rendimiento a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows.

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, incluyendo .NET.

Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas Operativos de Microsoft. Como política de desarrollo de Cuba y también de nuestro país hermano Venezuela, se encuentra la migración a software libre, por lo que utilizar un sistema gestor de bases de datos que dependa en su implantación de un sistema operativo propietario no se presenta como una opción factible.

1.10.2 Oracle.

Oracle es un manejador de BD relacional que hace uso de los recursos del sistema informático en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargados de información. También proporciona la capacidad de almacenar y acude a los datos de forma consecuente con un modelo definido como relacional (...). Además es una suite de productos que ofrece una gran variedad de herramientas [3].

Es el mayor y más usado Sistema Manejador de Base de Dato Relacional (RDBMS) en el mundo, e incluye cuatro generaciones de desarrollo de aplicación, herramientas de reportes y utilitarios [3].

Entre las características de Oracle se destaca su escalabilidad y alta disponibilidad, aportando un sistema de administración completo para gestionar todas las situaciones críticas de una Base de Datos, por ejemplo presenta: Sistema de seguridad basados en usuarios, grupos y roles, alertas, backups y restauración de datos. Oracle corre en computadoras personales (PC), mainframes y computadoras con

procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de hardware y software distintos sin tener la necesidad de cambiar una sola línea de código. Esto es porque más del 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos [3].

1.11 Bases de Datos Geográficas (BDG).

La esencia de un SIG está constituida por una base de datos geográfica que no es mas que una colección de datos almacenados acerca de objetos localizados en una determinada área de interés en la superficie terrestre, organizados de forma tal que puede servir eficientemente a una o varias aplicaciones.

Una base de datos geográfica requiere de un conjunto de procedimientos que permitan hacer un mantenimiento de ella tanto desde el punto de vista de su documentación como de su administración.

La eficiencia está determinada por los diferentes tipos de datos almacenados en diferentes estructuras.

1.11.1 Oracle Spatial.

Oracle Spatial ha sido desarrollado para facilitar el uso de datos espaciales y hacerlos accesibles a todas las compañías y usuarios de los SIG. Es una extensión del gestor de base de datos Oracle, proporciona un esquema SQL de las funciones que facilitan el almacenamiento, la recuperación, la actualización y las consultas de los datos espaciales hacia la base de datos. Sus principales componentes son:

- Un tipo de datos (MDSYS) que prescribe el almacenaje, la sintaxis, y la semántica de los tipos de datos geométricos apoyados.
- Un mecanismo espacial de indexación.
- Un sistema operador y las funciones para realizar consultas de la sección de interés, consultas join espaciales.
- Utilidades administrativas.

El modelo de datos de Oracle Spatial es una jerarquía consistente en **Elementos, Geometrías, Capas y Sistema Coordinado**. También presenta un índice de búsqueda basado en criterios espaciales tales como intersección y contención. Por defecto utiliza una estructura de índice espacial R-tree que aproxima

cada geometría a un solo rectángulo llamado el rectángulo de limitación mínimo e incluye como mínimo la geometría.

1.12 MapInfo.

Fundada en 1986, MapInfo es líder de mercado en herramientas de software para la visualización y análisis de datos corporativos a través de mapas.

Los diferentes productos de MapInfo han sido diseñados para que los usuarios visualicen y examinen los datos desde una perspectiva geográfica, superponiéndolos en mapas digitales con diferentes niveles. De esta forma, es posible utilizar la información geográfica de una empresa (por ejemplo, ciudades, nombres de calles, códigos postales...) para descubrir modelos de comportamiento y tendencias que resultan difíciles de identificar de otra manera.

MapInfo ofrece una amplia gama de soluciones y software para PC e Internet, herramientas de desarrollo de aplicaciones y datos.

Los archivos de MapInfo se estructuran alrededor de las denominadas tablas que contienen la información espacial y la de atributos. De esta forma, por una tabla lo mismo podemos entender un mapa o una capa. La información se almacena en tablas independientes que generalmente conforman 4 ó 5 ficheros enlazados entre si, los cuales se crean automáticamente y mantienen el mismo nombre, estos serán:

- Nombre de la tabla.TAB (fichero de textos con el encabezamiento de toda la información)
- Nombre de la tabla.DAT (fichero binario y de textos con los datos alfanuméricos)
- Nombre de la tabla.MAP (fichero binario con los datos espaciales)
- Nombre de la tabla.ID (fichero binario para el enlace de los demás ficheros)
- Nombre de la tabla.IND (fichero binario con los campos de la base de datos que han sido indexados) (opcional).

A continuación se mencionan algunos de los principales productos de MapInfo que de una forma u otra facilitan el trabajo a los usuarios de los SIG.

MapInfo MapBasic: Entorno de desarrollo de aplicaciones para MapInfo Professional.

MapInfo MapX: La forma más fácil y económica de añadir funcionalidad de gestión de mapas a otras aplicaciones y soluciones.

MapInfo MapXtreme 2004: Es la novedad principal del entorno de desarrollo basado en ubicaciones. Permite crear aplicaciones personalizadas sobre la plataforma .NET

MapInfo SpatialWare: Herramienta de gestión de información para almacenar, gestionar y manipular los datos basados en ubicación.

1.12.1 MapInfo MapXtreme 2004.

MapInfo MapXtreme 2004 permite desarrollar aplicaciones de manipulación de mapas digitales, en entornos Web y Desktop basados en Microsoft Windows.

MapXtreme 2004 ofrece:

- Un solo producto para el desarrollo de aplicaciones Windows, integrando funcionalidades geográficas para su implementación en la Web o en Desktop.
- Un desarrollo de aplicaciones más sencillo mediante la integración sin inconvenientes de MapInfo en el entorno Visual Studio.NET Development Environment, extendiendo así sus capacidades para el trabajo con mapas digitales.
- Una verdadera interoperabilidad gracias a su conformidad con los estándares de la industria geoespacial y de Tecnología de la Información.
- Una herramienta de desarrollo rica en funciones que permite desarrollar aplicaciones con una complejidad y flexibilidad sin precedentes.

MapXtreme 2004 está diseñado para los programadores de Microsoft Windows con conocimientos del entorno de desarrollo de Microsoft .NET, está integrada completamente con el entorno de desarrollo Visual Studio.NET. La solución MapXtreme 2004 SDK (FrameWork) permite desarrollar aplicaciones y soluciones que incorporan la ubicación dentro de los sistemas y la infraestructuras existentes. MapXtreme 2004 SDK es un único producto de Mapinfo que ofrece la capacidad de desarrollar aplicaciones Web o de escritorio para los entornos de smart- client, cliente-servidor o intranet/extranet utilizando la plataforma .NET.

Para el desarrollo de aplicaciones de escritorio, MapXtreme 2004 incluye varias plantillas de aplicaciones de mapas que configuran y cargan automáticamente el entorno de creación de mapas para el tipo de desarrollo seleccionado. Esto incluye herramientas de control de mapas que se utilizan con frecuencia, aplicaciones de muestra y funcionalidades de desarrollo. La opción de escritorio MapXtreme 2004 es completamente compatible con los lenguajes .NET de Microsoft como, por ejemplo, VB.NET, C# y Visual C++.

MapInfo MapXtreme es un producto totalmente compatible con los estándares de TI (Tecnología de la Información) como Microsoft .NET. Facilita el acceso directo a la información mediante sistemas de acceso a datos estándar y protocolos tales como ADO.NET, ODBC, Oracle Spatial y Microsoft SQL Server.

MapInfo MapXtreme 2004 es compatible con una amplia gama de bases de datos espaciales y no espaciales, ejemplo: SQL Server 2000, MS Access, Oracle 10g, etc.

1.13 Conclusiones.

Al realizar el estudio de los fundamentos teóricos que sustentan esta investigación, se constató la influencia de los SIG y sus herramientas de desarrollo en los centros de gestión de emergencias y seguridad ciudadana, se aborda como el desarrollo de esta tecnología ha revolucionado en ese campo y la importancia que tiene en la toma de decisiones en base a mejorar el servicio de las entidades encargadas de la seguridad ciudadana.

También se analizaron las ventajas y la importancia de los centros de emergencia y seguridad ciudadana en la República Bolivariana de Venezuela y como se ha ido introduciendo poco a poco el uso de estos servicios en todos los estados, buscando de esta forma una mejor atención al ciudadano y que este se sienta identificado con su gobierno.

Además se realizó un análisis detallado de la metodología a utilizar, de la herramienta para modelar el problema, de los principales lenguajes de programación, de las plataformas sobre las cuales se podía implementar un software con las características de un SIG y cuales gestores de base de datos serían más útiles.

Después de un extenso proceso de investigación, se decidió que el SIG para el Centro 171 se implementará utilizando: plataforma de desarrollo .NET, lenguaje de programación C#, metodología RUP y UML como lenguaje de modelación, gestor de base de datos Oracle y MapInfo MapXtreme 2004 como framework para mapas digitales.

Capítulo II: Descripción de la solución propuesta.

2.1 Introducción.

En este capítulo, se realiza un análisis del proceso de gestión de emergencias desde el punto de vista de la implementación de un SIG, ofreciendo una mayor comprensión del mismo y sus características. El desarrollo de la aplicación se centra en RUP, haciendo uso de UML para la modelación de los artefactos. Ha sido de gran utilidad el empleo de la herramienta CASE Rational Rose, que asiste al desarrollo de software para una mayor calidad de éste.

Se presenta el modelo de dominio como alternativa al modelo de negocio, debido a la poca estructuración de los procesos que describen el negocio de los centros de gestión de emergencias en el Área Metropolitana de la República Bolivariana de Venezuela. Además, este modelo ayuda a entender el contexto en que se emplaza el sistema y por tanto contribuye a la comprensión de los requisitos del sistema.

También se presenta la propuesta del sistema a desarrollar y se especifican los requerimientos funcionales y no funcionales. Se determinan los casos de uso del sistema y los actores que interactúan con ellos, elaborándose una descripción en formato expandido de todos.

2.2 Modelo de Dominio.

Debido a que en la República Bolivariana de Venezuela no existe un centro de gestión de emergencias que presente un SIG, se propone un Modelo de Dominio donde se mostrarán los principales conceptos a utilizar en el desarrollo de la aplicación. Esto ayuda a los usuarios, clientes, desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema, para capturar correctamente los requisitos y poder construir un sistema correcto.

Conceptos del Modelo del Dominio:

Elementos: Un elemento es la unidad de información básica de una capa. Los tipos espaciales son puntos, líneas, y polígonos.

Capas: Una capa es una colección de elementos homogéneos, es decir, tienen el mismo conjunto de atributos.

Mapa: Un mapa es un conjunto de capas superpuestas.

Usuarios: Son actores que utilizan el mapa y en dependencia del rol que posean consultarán unos u otros elementos del mapa.

Los conceptos de más relevancia durante el desarrollo de la aplicación se relacionan entre sí para dar a conocer el dominio de la información del sistema. Esto se representa a continuación:

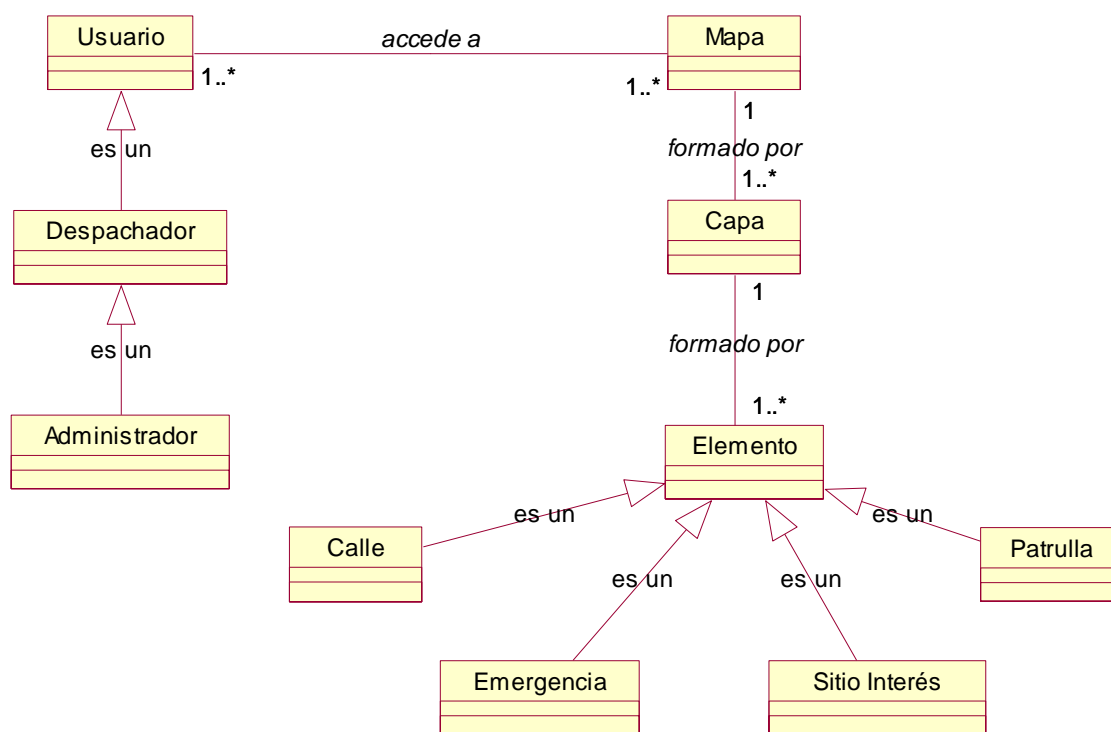


Figura 2.1. Modelo de Dominio del Módulo de Mapificación de la Información.

Descripción del Modelo de Dominio.

Los usuarios que interactúan con el mapa pueden ser administradores, un despachadores o usuarios estándar. Un administrador consulta cualquiera de los elementos del mapa, mientras que un despachador presta especial atención a las patrullas y un usuario estándar solamente interactúa con las calles, emergencias y sitios de interés.

Un mapa está formado por varias capas y cada una de estas por un conjunto de elementos de una misma geometría, es decir, un mapa está compuesto por varias figuras de diferentes tipos, líneas, polígonos y puntos, separadas en capas independientes que están compuestas por elementos de un mismo tipo. En el mapa se muestran las calles, edificaciones, sitios de interés, emergencias, patrullas y todos los elementos inherentes a él.

2.3 Requerimientos.

2.3.1 Requerimientos funcionales.

Los requerimientos funcionales no son más que las condiciones o capacidades que el sistema debe cumplir, el punto de partida para identificar qué debe hacer el sistema [8]. Deben comprenderlo tanto los desarrolladores como los usuarios.

Los requerimientos funcionales que debe cumplir la aplicación a desarrollar son los que a continuación se relacionan:

R1: Iniciar Aplicación

1. Mostrar la aplicación al usuario con las opciones a las que tiene acceso según los roles o permisos que tiene asignado.

R2: Activar Aplicación

2. Activar aplicación.
 - 2.1. Los datos necesarios son:
 - 2.1.1. No de serie de la oficina: número único generado aleatoriamente por el sistema (Módulo de Administración) para identificar unívocamente cada oficina.
 - 2.1.2. Nombre de usuario.
 - 2.1.3. Contraseña.
 - 2.1.4. Servidor: nombre del servidor de Base de datos.
 - 2.2. La aplicación se activa en la Base de Datos y en el registro del sistema operativo.

R3: Autenticar Usuario

3. Autenticar Usuario.
 - 3.1. Los datos necesarios son:
 - 3.1.1. Nombre de usuario.
 - 3.1.2. Contraseña.
 - 3.2. Validar datos introducidos.
 - 3.3. Obtener roles y permisos asociados al usuario.

R4: Abrir Mapa

4. Abrir un Mapa.
 - 4.1.1. El mapa puede encontrarse en la Base de Datos o en cualquier dispositivo de almacenamiento.

R5: Manipular Vistas

5. Manipular Vista.
 - 5.1. Abrir una Vista.
 - 5.1.1. Se puede abrir una Vista desde cualquier dispositivo de almacenamiento.
 - 5.2. Salvar una Vista.
 - 5.2.1. Se puede salvar una Vista en cualquier dispositivo de almacenamiento.
- Una Vista es un conjunto de capas que se visualizan en un momento dado.

R6: Gestionar Capas

6. Gestionar Capas.
 - 6.1. Adicionar una capa al mapa visualizado.
 - 6.1.1. Una capa está formada por objetos de un mismo tipo geométrico, los cuales pueden estar representados por: vehículos, calles, sitios de interés, emergencias, entre otros.
 - 6.2. Eliminar una capa del mapa visualizado.

R7: Realizar Operaciones Básicas

7. Realizar Operaciones Básicas sobre el mapa, a continuación se muestran las principales:

7.1. Seleccionar un objeto del mapa.

7.1.1. Existen tres vías de selección:

7.1.1.1. Selección puntual.

Selecciona el objeto sobre el cual se ha presionado el botón principal del Mouse.

7.1.1.2. Selección circular.

Selecciona los objetos que quedan en el interior del círculo dibujado.

7.1.1.3. Selección poligonal.

Selecciona los objetos que quedan en el interior del polígono dibujado.

7.2. Aumentar el tamaño del mapa visualizado.

7.2.1. Se aumenta a partir de un área seleccionada por el usuario.

7.3. Disminuir el tamaño del mapa visualizado.

7.3.1. Se disminuye a partir de un área seleccionada por el usuario.

7.4. Centrar un área del mapa.

7.4.1. Se centra a partir de un área seleccionada por el usuario.

7.5. Mover mapa.

7.6. Asociar una etiqueta a una región del mapa para identificarla.

7.6.1. Esta área puede ser una dirección, una parroquia, un municipio, un estado, un sitio de interés, etc.

R8: Buscar Elemento

8. Efectuar la búsqueda de uno o varios elementos que estén accesibles desde cualquier capa en el mapa, y mostrar su ubicación en éste.

8.1. La búsqueda puede ser en tiempo real o con carácter histórico.

8.2. Estos elementos pueden ser:

8.2.1. Un hecho.

8.2.2. Una patrulla.

8.2.3. Un sitio de interés.

8.2.4. Una dirección.

R9: Generar Reportes

9. Generar Reportes

9.1. Generar reporte de Hechos Delictivos.

9.1.1. La información de este reporte es:

- 9.1.1.1. Fecha.
- 9.1.1.2. Emergencia.
- 9.1.1.3. Tipo de emergencia.
- 9.1.1.4. Lugar en que ocurrió el hecho.
- 9.1.1.5. Total de hechos.
- 9.1.1.6. Cantidad de víctimas.

9.2. Generar reporte de Recursos.

9.2.1. La información de este reporte es:

- 9.2.1.1. Cantidad de patrullas.
- 9.2.1.2. Cantidad de ambulancias.

R10: Encuestar Patrulla

10. Encuestar datos de una patrulla.

10.1. Los datos que se obtienen son:

- 10.1.1. Posición.
- 10.1.2. Velocidad de traslación.

R11: Llamar Patrulla

11. Enviar notificación al módulo de comunicaciones sobre llamada a un patrullero.

11.1. Los datos de la notificación son:

- 11.1.1. Identificador de la patrulla.

R12: Definir Procedimientos Generales

12. Ubicar geográficamente recursos en servicio en un estado predefinido.

12.1. Los recursos son:

- 12.1.1. Patrullas.
- 12.1.2. Ambulancias.

12.1.3. Bomberos.

12.2. Esta ubicación se realiza para situaciones excepcionales como: desastres naturales, contingencias militares, etc.

R13: Asignar área geográfica de cada despachador

13. A cada despachador se asocian uno o varios municipios unidos geográficamente.

R14: Actualizar Patrulla

14. Visualizar en el mapa la posición en tiempo real de una o varias patrullas en servicio.

14.1. La posición en tiempo real de las patrullas se obtiene por medio del sistema GPS.

14.2. Se debe mostrar el estado del vehículo: ocupado o disponible.

R15: Actualizar Emergencias

15. Actualizar Emergencias.

15.1. Visualizar en el mapa la ubicación de las emergencias.

15.1.1. Las emergencias pueden ser, según su naturaleza:

15.1.1.1. Emergencia policial.

15.1.1.2. Emergencia de bomberos.

15.1.1.3. Emergencia médica.

15.1.1.4. Emergencia mixta.

15.2. Muestra la ubicación de la emergencia hasta que sea atendida.

R16: Actualizar Sitios de Interés

16. Visualizar en el mapa la ubicación de los sitios de interés.

16.1. Los sitios de interés pueden ser:

16.1.1. Estaciones de policía.

16.1.2. Hospitales.

16.1.3. Escuelas.

16.1.4. Museos.

16.1.5. Etc.

2.3.2 Requerimientos no funcionales.

Los requerimientos no funcionales especifican propiedades o cualidades que el producto de software debe tener, como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, facilidad de mantenimiento, entre otras [8].

Entre los requerimientos no funcionales del sistema propuesto se encuentran:

Apariencia o interfaz externa: La interfaz de usuario del sistema será amigable a los usuarios finales, cumpliendo con los requisitos siguientes:

- Las ventanas del sistema contendrán claro y bien estructurados los datos, al mismo tiempo permitirán la interpretación correcta e inequívoca de la información.
- La interfaz contará con teclas de función, teclas de atajo y menús desplegables que faciliten y aceleren su utilización.
- El sistema usará una norma que permita la distinción visual entre los elementos en las ventanas, a través del uso de colores, tamaño de las fuentes, así como otras técnicas.
- La corrección de errores en la introducción de datos por parte del usuario será clara y fácilmente identificable por él. El sistema hará uso del color rojo y de una imagen específica para mostrarlos.
- Todos los textos y mensajes en pantalla aparecerán en idioma español.
- Su funcionamiento será intuitivo y requerirá de información mínima.

Soporte: Garantía de instalación y prueba del sistema, además de un entrenamiento a los futuros usuarios.

Usabilidad: Para utilizar el sistema es necesario poseer conocimientos elementales de computación y conceptos propios de la información que se visualiza.

Rendimiento: El tiempo de respuesta no debe exceder los cinco segundos ante cada una de las acciones que solicite el usuario, debido a que es un Centro de Gestión de Emergencias, las cuales deben ser atendidas de manera urgente, es decir, el sistema debe ser capaz de procesar y mostrar la información lo más rápido posible.

Seguridad y privacidad:

- Se mantendrá seguridad y control a nivel de usuarios y contraseñas, garantizando el acceso de los mismos sólo a los niveles establecidos de acuerdo a la función que realizan. Las contraseñas sólo podrán ser cambiadas por el administrador del sistema.

- Se mantendrá un segundo nivel de seguridad a nivel de estación de trabajo, garantizándose la ejecución sólo de las aplicaciones que hayan sido establecidas para la estación en cuestión.
- El sistema permitirá registrar todas las acciones que se realizan, llevando el registro de las actividades de cada usuario en todo momento.
- La transmisión de datos por la red es a través de un protocolo seguro, TCP/IP.
- Se utilizan transacciones para garantizar el trabajo con los datos.

Software: Para el adecuado funcionamiento del sistema, se requieren las siguientes licencias de software:

Software	Requisitos
Sistema Operativo	Windows XP Professional SP I.
Plataforma de Desarrollo	Microsoft .NET Framework 1.1. MapInfo MapXtreme Framework 2004 SDK.
Programas	Oracle 10g y Oracle Spatial.

Tabla 2.1: Requisitos mínimos de software para el adecuado uso de la aplicación.

Hardware: Se requiere disponer, con vistas a que el sistema funcione como se tiene concebido, de una PC con las características que se muestran en la tabla.

Componente de Hardware	Requisitos
Procesador	3.00 - GHz o superior Pentium IV.
Memoria RAM	512 MB o más.
Espacio libre en disco duro	100 MB o más.
Tarjeta de Video	Cualquiera estándar con procesamiento interno y como mínimo 128 MB, se recomienda tecnología PCI-Express.
Monitor LCD	VGA, Súper VGA o LCD de 15" mínimo.

Tabla 2.2: Requisitos mínimos de hardware para el adecuado funcionamiento de la aplicación.

Restricciones para el Diseño y la Implementación: Para el desarrollo del sistema se utilizan las herramientas que brinda MapInfo MapXtreme 2004, como plataforma de programación el Visual Studio .Net 2003 y se emplea como lenguaje de programación visual C#. Los mapas deben estar en formato de MapInfo (*.TAB, *.MAP, *.ID, *.DAT). El sistema gestor de base de datos es Oracle 10g con la extensión Oracle Spatial para el tratamiento de los datos geométricos.

2.4 Descripción de la solución propuesta.

La ausencia de procedimientos que garanticen atender con efectividad las demandas de emergencias formuladas por la población de la Gran Caracas, relacionadas con la policía, los bomberos y salud pública; provocó que el Ministerio del Interior y Justicia planteara la necesidad de implantar un sistema que logre la coordinación de diferentes órganos policiales e instituciones relacionadas con la seguridad ciudadana, para garantizar una mejor y eficaz respuesta a las situaciones de emergencia que puedan ocurrir en el Área Metropolitana de este Estado. A este sistema se le denominó **Centro de Gestión de Emergencias y Seguridad Ciudadana 171 (Centro 171)**.

Este centro tiene como misión desarrollar un sistema con alto grado de automatización, eficiencia y profesionalidad, que brinde soluciones efectivas a las problemáticas de la población e integre diferentes subsistemas de cómputo, telefonía, radio e información operativa; que en su conjunto, lo conviertan en un sistema integral de emergencias y seguridad ciudadana. Los subsistemas de cómputo tienen entre sus funciones prioritarias, permitir la coordinación y visualización en tiempo real de las principales acciones que se desarrollen en la ciudad, a través de un mapa digitalizado de la misma. Este mapa también visualiza otros elementos de interés que lo convierten en una herramienta eficaz para responder a las demandas de los ciudadanos, y para que los jefes y directivos tomen decisiones más adecuadas. Para darle solución a estas funciones se crea el **Módulo de Mapificación de la Información**.

¿Cómo deberá funcionar el Módulo de Mapificación de la Información en el sistema informático del Centro 171?

El Módulo de Mapificación de la Información en el Centro 171 funciona de varias maneras en dependencia del rol que presente el usuario que acceda al sistema, es decir, cada usuario tiene asignado privilegios de acceso que le impiden o le permiten trabajar con todas las funcionalidades de la aplicación.

A continuación se explica detalladamente el funcionamiento del módulo de acuerdo a los usuarios que tienen acceso al sistema.

Cuando un usuario entra al sistema, el módulo muestra automáticamente el mapa de toda la región que se encuentra almacenado en una base de datos geográfica. Si este usuario es un:

- **Usuario Estándar:**

Una vez dentro del sistema ya está viendo el mapa en la aplicación y los hechos que ocurren, sobre estos puede realizar operaciones básicas como ampliar o disminuir una región, almacenar en el disco duro una vista de lo que está ocurriendo, conocer el nombre de un objeto, etc. Además se le ofrece la posibilidad de personalizar el mapa, es decir, se le permite adicionar y eliminar capas, las cuales no serán más que objetos que se visualizan, puede realizar búsquedas para localizar elementos que no se muestren en el área que está visualizando, de forma inmediata se le mostrará con un punto la ubicación del elemento. En la base de datos se almacenan reportes diarios de lo ocurrido, el usuario, dada una fecha, puede ver que lo sucedido y de esta forma sacar conclusiones estadísticas de criminalidad, accidentes, etc.

- **Despachador:**

El despachador recibe la información que envía el módulo de la operadora y de manera automática se muestra en el mapa la ubicación del hecho y la categoría que presenta. Este usuario puede realizar todas las operaciones descritas en el punto anterior y además, como es el encargado de mantener una conexión constante con los órganos de seguridad, desde el mapa digital puede iniciar una llamada al móvil más cercano para canalizar la emergencia y brindar la información que vaya recibiendo en tiempo real, también es posible encuestar un móvil que seleccione y de esta forma es capaz de conocer datos como la velocidad que transita el vehículo, el número de patrulla, si está disponible o no, ubicación geográfica, etc.

- **Administrador del Sistema:**

Este usuario es el encargado de administrar todo el sistema, particularmente crea vistas de posibles escenarios excepcionales, en los cuales hace falta que los órganos de seguridad y los vehículos en servicio adquieran cierta posición predeterminada en un área específica, además

otorga a cada despachador su área de trabajo. Es el máximo responsable de que esta aplicación funcione correctamente bajo cualquier situación, es el único usuario con privilegios para modificar una capa en la base de datos, determinar cuales capas se muestran y de tomar decisiones que de cierta forma afectan el funcionamiento de la aplicación.

De igual forma esta aplicación se vincula con otros módulos para adquirir información, recibe del Sistema GPS la posición de los móviles en servicio, el cual, cada cierto tiempo le envía automáticamente la localización. Desde el Módulo de Administración se actualizan los sitios de interés, notificando a todas las instancias del Módulo de Mapificación que hubo un cambio en esa capa de información y debe ser actualizada.

2.5 Modelo del Sistema.

2.5.1 Modelo de Casos de Uso del Sistema.

El modelo de casos de uso es un modelo del sistema que contiene actores, casos de uso y sus relaciones. Representa un esquema donde se recogen las funcionalidades del negocio que se automatizan y determina cómo será utilizado desde el punto de vista del usuario (Actor), pues se construye sobre la base de sus necesidades [8].

Los actores representan a los usuarios del sistema y otras aplicaciones que interactúan con él, es decir, representan terceros fuera del sistema que interactúan con éste. Estos suelen corresponderse con trabajadores o actores del negocio. Los actores definidos en el sistema son los siguientes:

Actores	Justificación
Usuario Estándar	Persona que interactúa con el sistema para consultar información.
Despachador	Persona a la que va dirigida fundamentalmente el sistema y que utiliza las funcionalidades más importantes del software, es el encargado de realizar el despacho de las emergencias hacia las unidades de competencia.
Administrador	Persona con privilegios sobre la base de datos, es el

	encargado de registrar los procedimientos generales a realizar en situaciones excepcionales y otorgarle, a cada despachador, su área de trabajo en el mapa.
Sistema GPS	Sistema automatizado que brinda información sobre la ubicación geográfica de los recursos en servicio.
Módulo de Administración	Sistema automatizado que brinda información sobre las actualizaciones relacionadas con los sitios de interés.
Módulo del Op - Desp	Sistema automatizado que brinda información sobre las actualizaciones relacionadas con las emergencias.

Tabla 2.3: Actores del sistema

A continuación se presentan los Casos de Uso (CU) determinados para satisfacer los requerimientos funcionales del sistema:

Casos de Uso (CU)	Propósito	Actores	Referencias
Iniciar Aplicación	Iniciar el subsistema de mapificación de la información y mostrar el formulario principal con las opciones a las que el usuario autenticado tiene acceso.	Usuario	R1
Autenticar Usuario	Permitir el acceso al sistema de mapas, determinando del usuario autenticado el rol que le corresponde.	Usuario	R2
Activar Aplicación	Activar la aplicación en la Base de Datos y en el registro del sistema operativo.	Usuario	R3

Abrir Mapa	Abrir los mapas de trabajo, ya sea que estén ubicados en la Base de Datos o en cualquier otro dispositivo de almacenamiento. Si se abre el mapa de la Base de Datos se tienen en cuenta los permisos definidos para el usuario.	Usuario	R4
Manipular Vistas	Abrir una vista almacenada en cualquier dispositivo de almacenamiento, o salvar una vista del mapa mostrado en el área de trabajo.	Usuario	R5
Gestionar Capas	Adicionar o eliminar una capa del mapa mostrado en el área de trabajo, según una petición del usuario.	Usuario	R6
Realizar Operaciones Básicas.	Realizar operaciones básicas que permitan al usuario trabajar con el mapa que se está mostrando.	Usuario	R7
Buscar Elemento	Buscar cualquier elemento que esté dentro del mapa.	Usuario	R8
Generar Reportes	Generar reportes de hechos delictivos y de los recursos en servicio. Se pueden hacer en tiempo real o con carácter histórico.	Usuario	R9
Encuestar Patrulla	Encuestar a un móvil, ofrece	Despachador	R10

	datos como: velocidad de traslación, ubicación y número de identificación.		
Llamar Patrulla	Inicia una comunicación con el móvil seleccionado.	Despachador	R11
Definir Procedimientos Generales	Ubica geográficamente los recursos en un mapa, que será utilizado cuando ocurra una situación excepcional.	Administrador	R12
Distribuir área del Despachador	Otorga un área de trabajo a cada despachador.	Administrador	R13
Actualizar Patrulla	Visualiza la posición y la disponibilidad en tiempo real de los recursos en servicio.	Sistema GPS	R14
Actualizar Emergencias	Visualiza en el mapa las emergencias según su naturaleza.	Módulo del Op - Desp	R15
Actualizar Sitios de Interés	Visualiza en el mapa todos los sitios de interés.	Módulo de Administración	R16

Tabla 2.4: Casos de Uso del Módulo de Mapificación de la Información.

2.5.2 Diagrama de Casos de Uso del Sistema.

A continuación se representa el diagrama donde se muestra la relación existente entre los actores y los casos de uso:

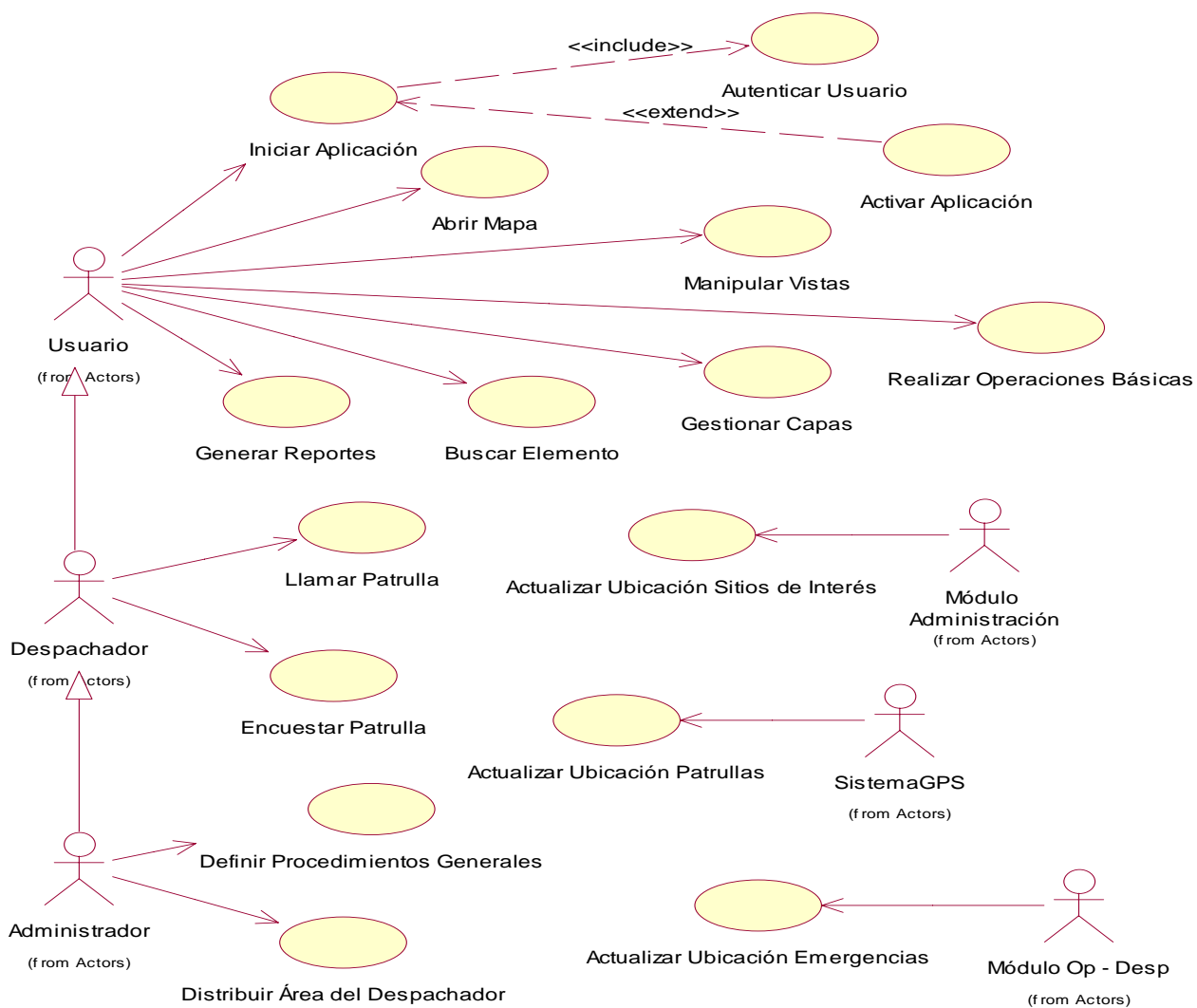


Figura 2.2 Diagrama de Casos de Uso del Sistema.

Basado en las características de RUP, se hizo una división de los CU en 3 ciclos de desarrollo. A continuación se muestra una tabla con el nombre de cada CU, el ciclo al que corresponde y la justificación de cada uno.

Los casos de uso Iniciar Aplicación, Activar Aplicación y Autenticar Usuario, se incluyen en el Modelo del Sistema debido a que constituyen los elementos básicos de la seguridad del módulo, y contribuyen a una mejor comprensión sobre el funcionamiento general del mismo. Sin embargo, no se encuentran en los ciclos de desarrollo porque no van a ser implementados como parte de éste módulo, forman parte de una línea horizontal de desarrollo para la arquitectura del sistema informático del Centro 171.

Nombre de Caso de Uso	Ciclo	Justificación de la selección
Abrir Mapa	1	Estos CU se encuentran implicados en el primer ciclo porque en su conjunto conforman las funcionalidades básicas de un SIG. A partir de este ciclo el módulo se va a perfeccionar en base a las condiciones que debe cumplir el sistema informático del Centro 171.
Manipular Vistas		
Gestionar Capas		
Realizar Operaciones Básicas		
Buscar Elemento		
Actualizar Ubicación Emergencias	2	En este ciclo comienzan a darse respuesta a los objetivos principales del Centro 171. Estos CU están en un segundo paso porque todos los actores son módulos que se están realizando en paralelo con el primer ciclo de este módulo.
Actualizar Ubicación Sitios de Interés		
Actualizar Ubicación Patrullas		
Distribuir área del Despachador		
Llamar Patrulla	3	Estos CU forman un tercer ciclo y último porque en su funcionamiento utilizan los CU del segundo ciclo. Llamar Patrulla: Localiza a la patrulla a través del mapa, posibilitando informar la ubicación de la emergencia y los sitios de interés más cercanos al incidente.
Encuestar Patrulla		
Generar Reportes		
Definir Procedimientos Generales		

Tabla 2.5: Descripción de los CU por ciclos.

2.5.3 Casos de Uso expandidos.

Mediante los casos de uso expandidos se describe paso a paso la secuencia de eventos que los actores utilizan para completar un proceso a través del sistema. A continuación se describen los CU del primer ciclo.

Caso de uso	
CU -1 Iniciar Aplicación	
Propósito	Iniciar el subsistema de mapificación de la información y mostrar el formulario principal con las opciones a las que el usuario autenticado tiene acceso.
Actores: Usuario (inicia).	
Resumen: El CU se inicia cuando el usuario va a utilizar el subsistema de mapificación de la información. El sistema determina que usuario ha iniciado la aplicación y entonces se visualiza la ventana principal con las opciones que se corresponden al nivel de acceso del mismo.	
Referencias:	R1
Acción del actor:	Respuesta del sistema:
1 - El usuario accede al sistema.	1.1 - El sistema determina que la aplicación está activada en esa estación de trabajo.
	1.2 – El sistema realiza el CU Autenticar.
	1.3 - El sistema muestra el formulario principal visualizando las opciones que se corresponden con el nivel de acceso del usuario.
Flujo alternativo	
Acción 1.1	El sistema determina que la aplicación no está activada en ese punto. El sistema realiza el CU Activar Aplicación. Ir a la acción 1.2.

Tabla 2.6. Descripción del CU Iniciar Aplicación.

Caso de uso	
CU -2 Autenticar Usuario	
Propósito	Permitir el acceso al sistema de mapas, determinando del usuario autenticado los roles que le corresponden.
Actores: Usuario	
Resumen: El CU se inicia cuando el usuario introduce en el sistema su usuario y contraseña, el sistema verifica la validez de estos datos y determina los roles del usuario.	
Referencias:	R2
Acción del actor:	Respuesta del sistema:
	1 - El sistema muestra el formulario Autenticar Usuario, solicitando nombre de usuario y contraseña.
2 - El usuario introduce los datos solicitados.	
3 - El usuario presiona el botón Aceptar.	3.1 - El sistema verifica que los datos sean correctos y determina los roles del usuario.
Flujo alternativo	
Acción 3	El usuario presiona el botón Cancelar.
Acción 3.1	El sistema determina que los datos son incorrectos, muestra un mensaje de error y cuenta el intento de autenticación, al llegar a tres se cancela la acción. Pasa directamente a la acción 2.

Tabla 2.7. Descripción del CU Autenticar Usuario.

Caso de uso	
CU -3 Activar Aplicación	
Propósito	Activar la aplicación en la Base de Datos y en el registro del sistema

	operativo.
Actores: Usuario	
Resumen: El CU se inicia cuando se va a utilizar la aplicación por primera vez y esta se encuentra desactivada. Se requiere tener la aplicación instalada en el sistema.	
Referencias:	R3
Acción del actor:	Respuesta del sistema:
	<p>1. – El sistema muestra el formulario Activar Aplicación solicitando al usuario la información necesaria para activar la misma:</p> <ul style="list-style-type: none"> - Número de Serie de la Oficina. - Usuario. - Contraseña. - Servidor. <p>Este usuario debe tener asignado los permisos necesarios para poder activar la aplicación.</p>
2 – El usuario proporciona la información solicitada y presiona el botón Validar.	<p>2.1 – El sistema verifica que los datos sean correctos.</p> <p>2.2 – El sistema muestra el nombre de la oficina asociada al número de serie proporcionado y la descripción del punto.</p>
3 – El usuario consulta la información proporcionada y presiona el botón Aceptar.	3.1 – El sistema activa la aplicación en la Base de Datos y en el registro del sistema operativo. Se cierra el formulario Activar Aplicación.
Flujo alternativo	
Acción 2.1	El sistema verifica que los datos no son correctos, y vuelve al estado inicial.
Acción 3	<p>El usuario consulta la información proporcionada y presiona el botón Cancelar.</p> <p>El sistema no realiza ninguna acción y cierra el formulario Activar Aplicación.</p>

Tabla 2.8. Descripción del CU Activar Aplicación.

Caso de uso	
CU – 4 Abrir Mapa	
Propósito	Abrir los mapas de trabajo ya sea que estén ubicados en la Base de Datos o en cualquier otro dispositivo de almacenamiento. Si se abre el mapa desde la Base de Datos se tienen en cuenta los permisos definidos para el usuario.
Actores: Usuario (Inicia)	
Resumen: El CU se inicia cuando el usuario desea abrir un mapa. Si selecciona la opción Abrir Mapa Inicial, el sistema determina los permisos del usuario y carga automáticamente el mapa desde la Base de Datos con las características que se han definido para él. Si selecciona Abrir Mapa, el sistema solicita la localización del mismo y lo carga desde cualquier dispositivo de almacenamiento.	
Referencias:	R4
Acción del actor:	Respuesta del sistema:
1 - El usuario necesita abrir un mapa y selecciona una de las opciones disponibles: <ul style="list-style-type: none"> • Abrir Mapa Inicial (Ir a Sección Abrir Mapa Inicial). • Abrir Mapa (Ir a Sección Abrir Mapa). 	
Sección Abrir Mapa Inicial	
	1.1 - El sistema verifica si hay algún mapa cargado y de ser así lo cierra.
	1.2 – El sistema determina las características de configuración que se han definido para el usuario

	y carga el mapa desde la Base de Datos. La configuración de cada usuario depende de su rol.
Flujo alternativo	
Acción 1.2	En caso de que exista algún problema al cargar el mapa se muestra un mensaje de error.
Sección Abrir mapa	
	1.1 - El sistema verifica si hay algún mapa cargado y de ser así lo cierra.
	1.2 - El sistema muestra una ventana de diálogo solicitando la localización del mapa que se desea abrir. El mapa puede estar ubicado en cualquier dispositivo de almacenamiento.
2 - El usuario localiza el mapa que desea abrir y pulsa el botón aceptar de la ventana de diálogo.	2.1 - El sistema carga el mapa seleccionado.
Flujo alternativo	
Acción 2	El usuario busca el mapa que desea abrir y pulsa el botón cancelar de la ventana. El sistema no carga ningún mapa.
Acción 2.1	En caso de que exista algún problema al cargar el mapa se muestra un mensaje de error.

Tabla 2.9. Descripción del CU Abrir Mapa.

Caso de uso	
CU – 5 Manipular Vistas	
Propósito	Abrir una vista almacenada en cualquier dispositivo de almacenamiento o

	salvar una vista del mapa mostrado en el área de trabajo.
Actores: Usuario (Inicia)	
Resumen: El usuario selecciona abrir o salvar una vista, en cualquier caso el sistema solicita una localización de la vista y ejecuta la acción que se le ha solicitado.	
Referencias:	R5
Acción del actor:	Respuesta del sistema:
1 – El usuario necesita abrir o salvar una vista y selecciona una de las opciones disponibles: <ul style="list-style-type: none"> • Abrir Vista (Ir a Sección Abrir Vista). • Guardar Vista (Ir a Sección Guardar Vista). 	
Sección Abrir Vista	
	1.1 - El sistema muestra una ventana de diálogo solicitando la ubicación de la vista que se desea abrir.
2 - El usuario localiza la vista que desea abrir y pulsa el botón aceptar de la ventana de diálogo.	2.1 - El sistema cierra el mapa que se está mostrando y carga la vista seleccionada.
Flujo alternativo	
Acción 2	El usuario localiza o no la vista que desea abrir y pulsa el botón cancelar de la ventana de diálogo. El sistema no carga ninguna vista, mostrando el mapa como se encontraba antes de iniciar la acción.
Acción 2.1	En caso de que exista algún problema al cargar la vista se muestra un mensaje de error.
Sección Guardar Vista	

	1.1 - El sistema muestra una ventana de diálogo solicitando la ubicación donde se desea guardar la vista.
2 - El usuario selecciona la ubicación donde desea guardar la vista del mapa y define un nombre para identificarla.	
3 - El usuario pulsa el botón Aceptar de la ventana de diálogo.	3.1 - El sistema guarda en esta ubicación la vista.
Flujo alternativo	
Acción 2	El usuario pulsa el botón Cancelar de la ventana de diálogo. El sistema no salva la vista y muestra el mapa como se encontraba antes de iniciar la acción.
Acción 3.1	En caso de que exista algún problema al guardar la vista se muestra un mensaje de error.

Tabla 2.10. Descripción del CU Manipular Vistas.

Caso de uso	
CU – 6 Gestionar Capas	
Propósito	Adicionar o eliminar una capa del mapa mostrado en el área de trabajo según una petición del usuario.
Actores: Usuario (Inicia)	
Resumen: El usuario selecciona adicionar o eliminar una capa al mapa de trabajo visible. Puede adicionar una capa desde la Base de Datos o desde cualquier dispositivo de almacenamiento.	
Referencias:	R6
Acción del actor:	Respuesta del sistema:
1 - El usuario necesita adicionar o	1.1 El sistema muestra el formulario Gestionar

eliminar una capa al mapa de trabajo visible y selecciona la opción Gestionar Capas.	Capas con el listado de capas visibles en el mapa.
<p>2 - El usuario selecciona en el formulario Gestionar Capas una de las opciones disponibles:</p> <ul style="list-style-type: none"> • Adicionar Capa (Ir a Sección Adicionar Capa). • Eliminar Capa (Ir a Sección Eliminar Capa). 	
Sección Adicionar Capa	
	2.1 - El sistema muestra el formulario Adicionar Capas con el listado de capas a las que el usuario tiene permisos de acceso y no están visibles en el mapa.
3 - El usuario selecciona una de las capas del listado.	
4 - El usuario presiona el botón Adicionar del formulario Adicionar Capas.	4.1 - El sistema cierra el formulario Adicionar Capas y añade la capa seleccionada al listado de capas visibles del formulario Gestionar Capas.
5 - El usuario presiona el botón Aceptar del formulario Gestionar Capas.	5.1 - El sistema visualiza en el mapa la capa seleccionada y cierra el formulario Gestionar Capas.
Flujos Alternativos	
Acción 2.1	<p>Se muestra la lista de capas vacía y un mensaje que indica que todas las capas se encuentran visibles.</p> <p>La opción adicionar de este formulario Adicionar Capas no está disponible.</p>

Acción 4	El usuario presiona el botón Cancelar y el sistema cierra el formulario Adicionar Capas.
Acción 4	<ul style="list-style-type: none"> - El usuario presiona el botón Examinar del formulario Adicionar Capas. - El sistema abre una ventana de diálogo solicitando la ubicación de la capa que se desea abrir. - El usuario localiza la capa y presiona el botón aceptar. <p>Ir a la acción 4.1.</p> <p>El usuario puede presionar el botón Cancelar cuando esta localizando la capa, en cuyo caso se cierra la ventana de diálogo Pasa directamente a la acción 2.</p>
Acción 5	El usuario presiona el botón Cancelar y el sistema cierra el formulario Gestionar Capas, cancelando todas la acciones que se han iniciado si no han sido aplicadas.
Acción 5	El usuario presiona el botón Aplicar y el sistema visualiza en el mapa las capas seleccionadas sin cerrar el formulario Gestionar Capas. Ir a la acción 2.
Sección Eliminar Capa	
3 – El usuario selecciona una de las capas del listado.	
4 – El usuario presiona el botón Eliminar del formulario Gestionar Capas.	4.1 - El sistema muestra un mensaje solicitando confirmación para la eliminación de la capa seleccionada.
5 – El usuario confirma la eliminación.	5.1 – El sistema elimina la capa seleccionada del

	<p>listado de capas visibles del formulario Gestionar Capas.</p> <p>Los pasos del 3 al 5.1 pueden repetirse tantas veces como desee el usuario.</p>
6 - El usuario presiona el botón Aceptar del formulario Gestionar Capas.	6.1 - El sistema actualiza el mapa eliminando las capas seleccionadas y cierra el formulario Gestionar Capas.
Flujos Alternativos	
Acción 5	El usuario rechaza la eliminación y el sistema cancela la acción.
Acción 6	El usuario presiona el botón Cancelar y el sistema cierra el formulario Gestionar Capas, cancelando todas la acciones que se han iniciado si no han sido aplicadas.
Acción 6	<p>El usuario presiona el botón Aplicar y el sistema actualiza el mapa eliminando las capas seleccionadas sin cerrar el formulario Gestionar Capas.</p> <p>Pasa directamente a la acción 2.</p>

Tabla 2.11. Descripción del CU Gestionar Capas.

Caso de uso	
CU -7 Realizar Operaciones Básicas	
Propósito	Realizar operaciones básicas que permitan al usuario trabajar con el mapa que se está mostrando.
Actores: Usuario (Inicia)	
Resumen: El CU se inicia cuando el usuario desea hacer alguna operación sobre el mapa que se muestra. El sistema identifica cual operación va a realizar el usuario sobre el mapa y da respuesta de esta.	

Referencias:	R7
Acción del actor:	Respuesta del sistema:
<p>1 – El usuario desea realizar alguna de las operaciones básicas sobre el mapa, selecciona una de las opciones del menú:</p> <ul style="list-style-type: none"> • Acercar (Ir a Sección Acercar). • Alejar (Ir a Sección Alejar). • Centrar (Ir a Sección Centrar). • Seleccionar Región (Ir a Sección Seleccionar Región). • Selección Circular (Ir a Sección Selección Circular). • Selección Poligonal (Ir a Sección Selección Poligonal). • Identificar Objeto (Ir a Sección Identificar Objeto). • Mover Mapa (Ir a Sección Mover Mapa). • 	
Sección Acercar	
2 - El usuario presiona el botón principal del mouse sobre la zona del mapa que desea ampliar.	2.1 – El sistema determina las coordenadas del clic.
	2.1 - El sistema centra y amplía el mapa en la pantalla usando las coordenadas determinadas.
Flujo Alternativo	
Acción 2	<p>El usuario presiona el botón principal del mouse sobre una zona del mapa.</p> <p>Sin soltar el botón va a ampliar la diagonal del rectángulo que se creará hasta que el área que</p>

	<p>desea ampliar quede incluida en este y entonces libera el botón del mouse.</p> <p>El sistema determina las coordenadas del punto inicial (donde se presionó el botón del mouse) y final (donde se liberó el botón del mouse), centra y amplía el área del mapa que fue seleccionada en la pantalla, mostrándose solo esta zona.</p>
Sección Alejar	
2 - El usuario presiona el botón principal del mouse sobre la zona del mapa que desea reducir.	2.1 - El sistema determina las coordenadas del clic.
	2.2 - El sistema reduce el mapa en la pantalla usando las coordenadas.
Flujo Alternativo	
Acción 2	<p>El usuario presiona el botón principal del mouse sobre una zona del mapa.</p> <p>Sin soltar dicho botón va a ampliar la diagonal del rectángulo que se creará hasta que el área que desea disminuir quede incluida en este y se libera el botón del mouse.</p> <p>El sistema determina las coordenadas del punto inicial (donde se presionó el botón del mouse) y final (donde se liberó el botón del mouse).</p> <p>El sistema reduce el área del mapa que fue seleccionada en la pantalla.</p>
Sección Centrar	
2 - El usuario presiona el botón principal del mouse sobre la zona del mapa que desea centrar.	2.1 - El sistema determina las coordenadas del clic.
	2.2 El sistema ubica estas coordenadas en el centro de la pantalla.
Sección Selección Región	

2 - El usuario da clic sobre la zona del mapa que desea seleccionar.	2.1 - El sistema determina las coordenadas del clic.
	2.2 - El sistema determina a que región pertenecen estas coordenadas y rellena el área con otro color.
Sección Selección Circular	
2 – El usuario presiona el botón principal del mouse sobre una región o área del mapa. Sin soltar el botón principal va a ampliar el radio del círculo que se creará hasta que el área que desea seleccionar quede incluida en este y libera el botón del mouse.	2.1 - El sistema determina todas las regiones cuyo centro se encuentren dentro del círculo trazado y las rellena con un color diferente.
Flujo Alternativo	
Acción 2.1	El sistema no rellena ningún área de color diferente porque en el círculo trazado no han quedado incluidos centros de región.
Sección Selección Poligonal	
2 - El usuario realiza una serie de clic tratando que la mayor parte del área o región del mapa que desea seleccionar quede incluida en el interior del polígono que se va creando.	3.1 - El sistema va marcando con un pequeño punto cada clic y con una recta la unión de cada uno de ellos.
4 - El usuario realiza doble clic determinando que ese es su punto final.	4.1 - El sistema une mediante una recta el primer punto y el último, cerrando el polígono.
	4.2 - El sistema determina todas las regiones cuyo centro se encuentren dentro del polígono trazado y las rellena con un color diferente.
Flujo Alternativo	

Acción 4.2	El sistema no rellena ningún área de color diferente porque en el polígono trazado no han quedado incluidos centros de región.
Sección Identificar Región	
2 - El usuario presiona el botón principal del mouse sobre la región del mapa que desea identificar.	2.1 - El sistema determina las coordenadas del clic.
	2.2 - El sistema determina a que región pertenecen estas coordenadas y muestra en el mapa, sobre el área seleccionada, una etiqueta con el nombre de la región. (Ejemplo: Si es una calle muestra su nombre, si es un sitio de interés muestra el tipo de sitio, etc.).
Sección Mover Mapa	
3 - El usuario presiona el botón principal del mouse sobre una zona del mapa y sin soltar dicho botón va a mover el Mouse en cualquier dirección que desee.	3.1 El sistema determina las coordenadas del clic y la dirección del movimiento del mouse. Con cada avance del mouse el sistema va repintando el mapa en la pantalla.

Tabla 2.12. Descripción del CU Realizar Operaciones Básicas.

Caso de uso	
CU – 8 Buscar Elemento	
Propósito:	Buscar cualquier elemento que esté visible en el mapa.
Actores: Usuario (Inicia)	
Resumen: El CU se inicia cuando el usuario elige la opción Buscar. El sistema muestra el formulario de búsqueda y automáticamente llena los campos por los que el mapa permite realizar ésta acción.	
Referencias:	R8
Acción del actor:	Respuesta del sistema:

<p>1 - El usuario selecciona la opción Buscar Elemento.</p>	<p>1.1 – El sistema muestra el formulario Buscar Elemento solicitando el criterio de búsqueda. El criterio de búsqueda está compuesto por los siguientes campos:</p> <ul style="list-style-type: none"> - Lista de Capas del mapa actual sobre el que se puede efectuar búsquedas. - Lista de Parámetros de Búsqueda de la capa seleccionada. - Lista de Elementos que posee el Parámetro de Búsqueda seleccionado. <p>Inicialmente la única lista que tiene elementos es la Lista de Capas.</p>
<p>2 - El usuario despliega la Lista de Capas y selecciona una capa.</p>	<p>2.1 - El sistema asigna a la Lista de Parámetros de Búsqueda los valores que se corresponden con la capa seleccionada y visualiza el primer parámetro de la lista.</p>
<p>3 - El usuario despliega la Lista de Parámetros de Búsquedas y selecciona un parámetro.</p>	<p>3.1 - El sistema asigna a la Lista de Elementos los valores que se corresponden con el parámetro seleccionado, dejando el campo Elemento a Buscar en blanco.</p>
<p>4 - El usuario comienza a teclear el nombre del elemento que desea buscar en el campo Elemento a Buscar.</p>	<p>4.1 - El sistema realiza una búsqueda sintáctica. A medida que el usuario va tecleando el sistema va sugiriendo el posible nombre del elemento. El usuario puede seleccionar la sugerencia o terminar de escribir la palabra. Si el usuario escribe una palabra para la que no hay sugerencia alguna, se inhabilita el botón Buscar.</p>
<p>5 - El usuario presiona el botón Buscar.</p>	<p>5.1 - El sistema realiza la búsqueda según el criterio seleccionado.</p>

	5.2 - El sistema muestra un punto en el lugar del mapa donde se encuentra el elemento buscado.
Flujo alternativo	
Acción 5	El usuario pulsa el botón Cancelar. El sistema cierra el formulario de búsqueda.

Tabla 2.13. Descripción del CU Buscar Elemento

2.6 Conclusiones.

El desarrollo de este capítulo ha permitido una mejor comprensión del contexto a automatizar, de las características y restricciones que deben existir en el sistema para cumplir con los requerimientos del cliente. Se realizó el Modelo del Dominio, descripción de la solución propuesta y se definieron los requisitos funcionales y no funcionales que debe cumplir el sistema. Se elaboró el diagrama de casos de uso del sistema donde se representan los actores y casos de uso, así como la relación entre ellos. Se describieron todos los casos de uso, a que ciclo pertenecía cada uno y posteriormente se realizó una descripción expandida de los casos de uso pertenecientes al primer ciclo de desarrollo.

Con el desarrollo de este flujo de trabajo y los artefactos obtenidos, se puede pasar al flujo de diseño para comenzar la construcción de la solución de software propuesta, el cual será presentado en el próximo capítulo.

Capítulo III: Construcción de la solución propuesta.

3.1 Introducción.

En este capítulo se realiza una exposición de cómo se construyó el SIG, basado en las condiciones específicas que debe cumplir el sistema informático del Centro 171, se describe la arquitectura de la aplicación, el diagrama de clases y el diseño de la base de datos que se utiliza.

También se ofrecerá una explicación de los principios de diseño de la aplicación, los cuales no son más que las pautas a seguir para el diseño de la interfaz y la programación, así como la estructuración de la ayuda y el tratamiento a seguir para cada posible error. De esta forma se brindan funcionalidades para que el sistema sea más robusto, legible y su uso sea más sencillo.

3.2 Arquitectura propuesta para la implementación del sistema informático del Centro 171.

La arquitectura del sistema informático del Centro 171 esta basada en dos características fundamentales. La primera es que presenta un modelo de desarrollo basado en casos de uso, y la segunda se refiere a que todo el ciclo de elaboración del sistema se ajustará a un modelo iterativo e incremental.

El sistema informático del Centro 171 está dividido en varios módulos: Módulo de Mapificación de la Información, Módulo de Administración, Módulo de Despacho, etc. Cada uno responde a un conjunto de funcionalidades específicas del cliente. Todos estos módulos interactúan entre sí y comparten datos de interés en dependencia de la funcionalidad de cada uno, siguen un estricto régimen de seguridad de la información y se desarrollan de forma paralela.

La arquitectura esta organizada desde dos enfoques, uno horizontal y otro vertical.

3.2.1 Enfoque Horizontal.

El desarrollo de cada módulo responde a un modelo multicapas (Véase Figura 3.1).

1. Capa interfaz.
2. Capa mediadora interfaz basada en acciones.
3. Capa lógica del negocio.

- 4. Capa acceso a datos.
- 5. Capa de datos.

A continuación se ilustra la arquitectura del sistema.

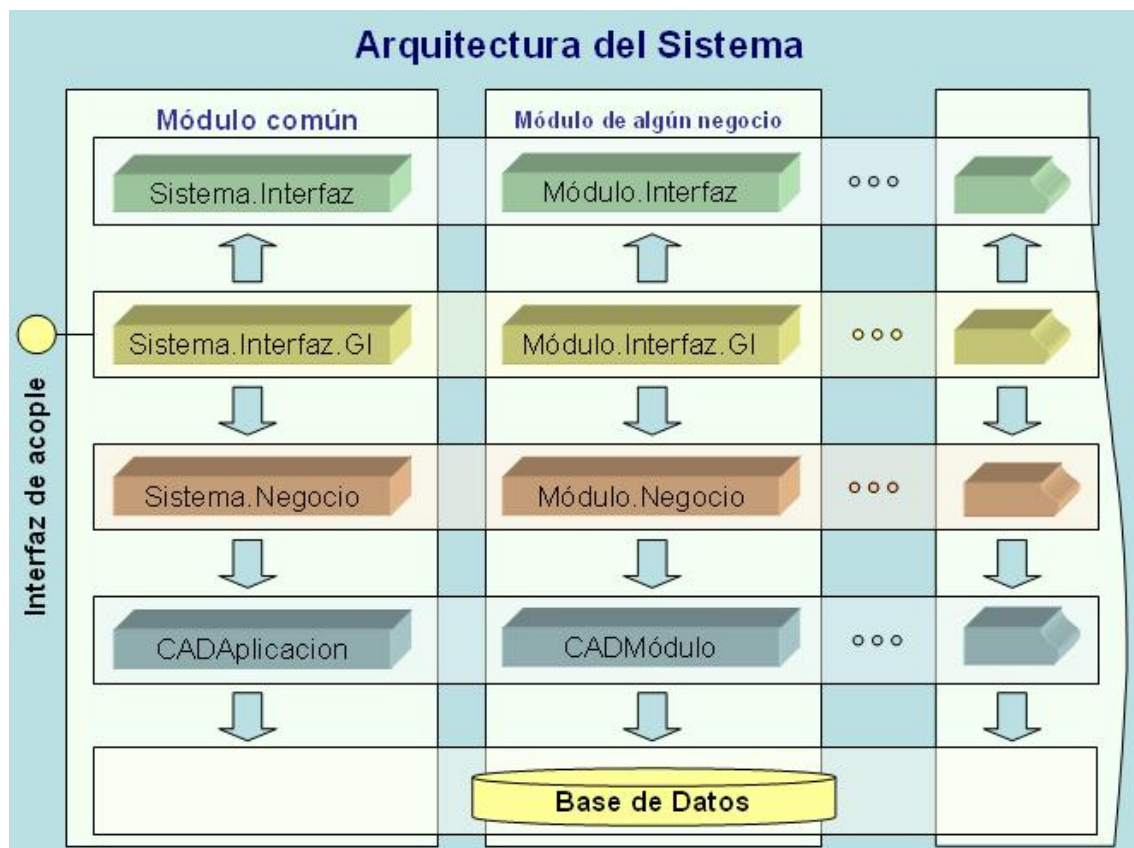


Figura 3.1: Arquitectura del Sistema.

Capa interfaz.

La capa de interfaz responde a un patrón estándar, todos los módulos tienen prácticamente el mismo diseño con vistas a facilitar el trabajo con ellos y la estandarización del sistema completo.

Capa mediadora interfaz basada en acciones.

Una acción es una clase que representa precisamente la ejecución de alguna tarea concreta. Estas tareas no deben ser excesivamente complejas y la clase debe:

- Heredar de la clase *Accion* o *AccionSegura*.
- Tener opcionalmente un formulario asociado.
- Atribuir propiedades en función del objetivo específico de la misma.
- Reescribir el método *CrearForma* con el objetivo de mostrar las características que se deseen en el formulario.
- En caso de que la acción tenga asignado un formulario, se deben programar dentro de la misma todos los eventos que puedan ocurrir a partir de la interacción con el usuario.

Capa lógica del negocio.

El diseño de esta capa depende directamente del negocio específico que se éste automatizando.

Es preciso se mantenga un estricto seguimiento de los documentos y artefactos de software definidos para la documentación del proceso, los cuales se explican más adelante.

Capa acceso a datos.

Esta capa esta compuesta por clases, que constituyen factorías de las entidades del negocio que deben persistir, debe garantizar todo el manejo de la información que requiera el negocio del problema y la conexión con la base de datos.

Capa de datos.

Esta capa corresponde a los almacenes de datos. A ella pertenecen las bases de datos en Oracle. Esta capa es única y común a todos los módulos del sistema general. Se debe caracterizar porque el modelo de datos debe estar en al menos la tercera forma normal.

3.2.2 Enfoque Vertical.

Cada modulo implementa sus funcionalidades y en caso que sea necesario se redefine aquella capa de la arquitectura que no se corresponda con su lógica de funcionamiento, por ejemplo para el Módulo de Mapificación de la Información fue necesario modificar la capa interfaz para aprovechar al máximo toda el área de trabajo.

3.3 Diagrama de Clases del diseño.

3.3.1 Diagrama de Clases del paquete Servicio.

Al iniciar la aplicación el sistema verifica si esta se encuentra activada en la estación de trabajo, si la condición anterior es correcta el usuario debe autenticarse. Una vez que el usuario se ha autenticado, el sistema determina el rol que desempeña y de esta forma se visualiza la ventana principal con las opciones a las que el usuario tiene permiso.

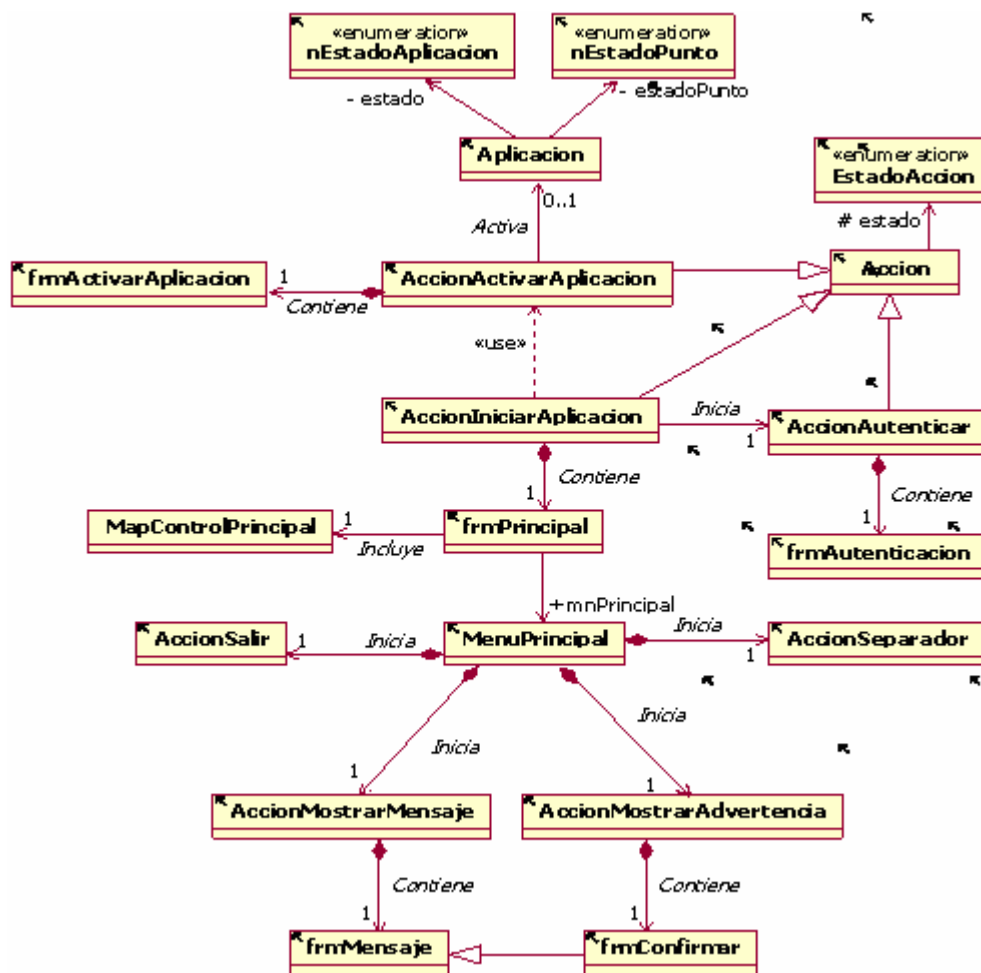


Figura 3.2: Diagrama de Clases del paquete Servicio.

3.3.2 Diagrama de clases del paquete Mapa.

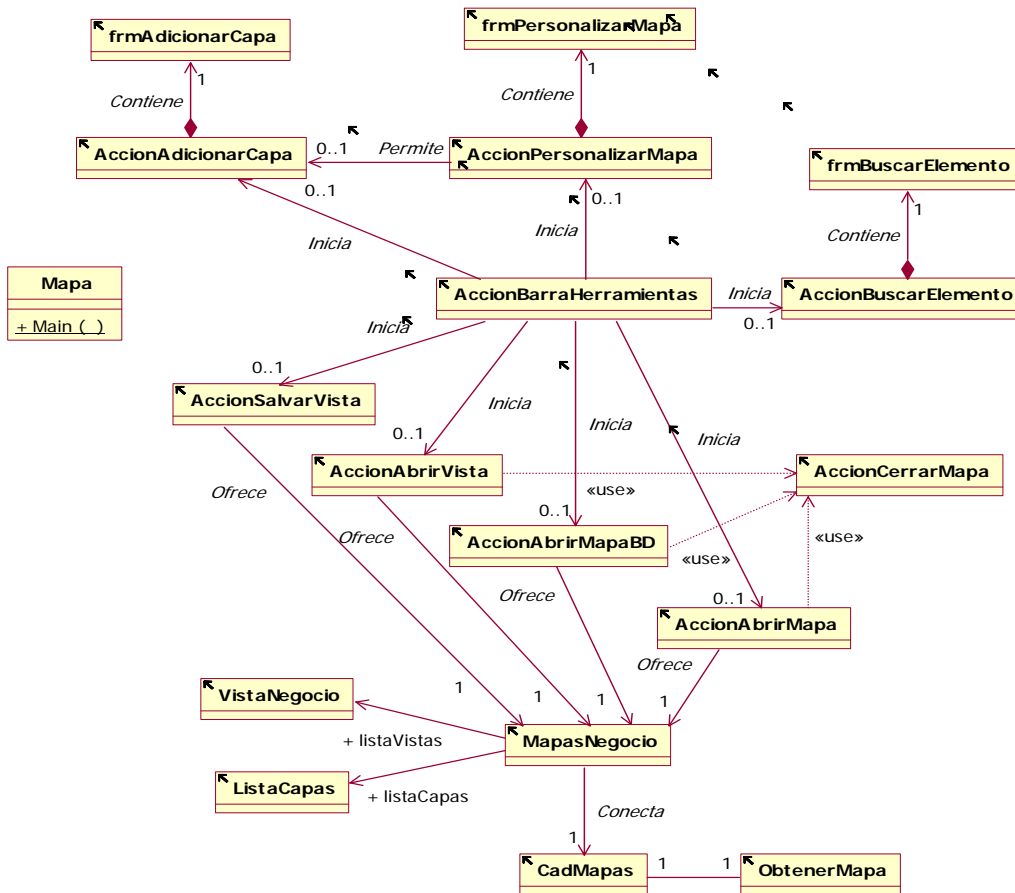


Figura 3.3: Diagrama de Clases del paquete Mapa.

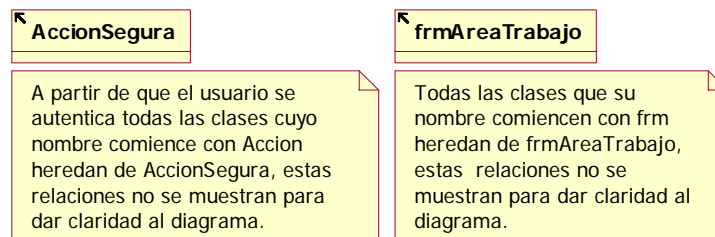


Figura 3.4: Notas necesarias para la comprensión del modelo.

3.3.3 Diagrama de clases Interacción entre Paquetes.

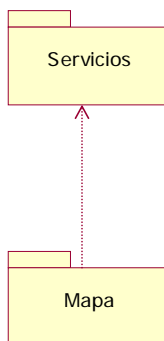


Figura 3.5: Diagrama de clases de interacción entre paquetes.

3.4 Diseño de la Base de Datos.

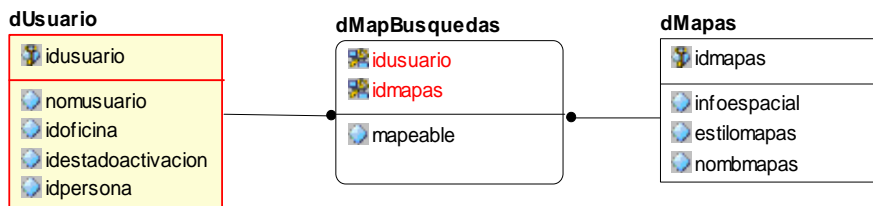


Figura 3.6: Diagrama de diseño de la Base de Datos.

3.4.1 Descripción de las Tablas.

Nombre: dUsuario		
Descripción: En esta tabla están registrados todos los usuarios del sistema.		
Atributo	Tipo de Datos	Descripción
idusuario	UNIQUEID	Identificador del usuario.
nomusuario	VARCHAR(20)	Nombre del usuario.
idoficina	UNIQUEID	Identificador de la oficina.

idestadoactivacion	NUMERIC(4,0)	Identificador del estado de activación.
idpersona	UNIQUEID	Identificador del usuario como persona.

Tabla 3.1: Descripción de la tabla dUsuario.

Nombre: dMapBusquedas		
Descripción: Tabla que relaciona los usuarios y los mapas, además dice si el mapa es de búsqueda.		
Atributo	Tipo de Datos	Descripción
idusuario	UNIQUEID	Identificador del usuario de la tabla dusuario.
idmapas	NUMERIC(10,0)	Identificador del mapa.
mapeable	VARCHAR(1)	Dice si el mapa es de búsqueda o no.

Tabla 3.3: Descripción de la tabla dMapBusquedas.

Nombre: dMapas		
Descripción: Tabla que almacena los mapas que se almacenan en la base de datos.		
Atributo	Tipo de Datos	Descripción
estilomapas	VARCHAR(254)	Define el tipo de estilo que va a tratar cada figura y es un campo de solo lectura.
nombmapas	VARCHAR(20)	Nombre de las capas.
infoespacial	MDSYS.SDO_GEOMETRY	Está la información espacial de la tabla, se almacenan las características geométricas de la figura.

Tabla 3.4: Descripción de la tabla dMapas.

3.5 Principios de Diseño de la Aplicación.

En todo proceso de desarrollo de software se hace necesario analizar los principios de diseño que determinarán las características del producto final. Las facilidades que brinda un software con un diseño coherente y consistente son elementos determinantes para una buena aceptación por parte de los usuarios.

Para el desarrollo del diseño del Módulo de Mapificación de la Información, se definen ciertas técnicas y principios con suficientes detalles como para permitir su interpretación y realización física, a continuación se explica el funcionamiento de cada una de estas.

3.5.1 Estándares en la interfaz de la aplicación.

Las interfaces del sistema informático del Centro 171 se implementan utilizando formularios de escritorio de Windows, de esta forma se permite procesar y dar formato a los datos de los usuarios, así como adquirir y validar los datos entrantes procedentes de éstos. En la construcción de los mismos se evitan las relaciones de modificación entre ellos, lo que aumenta el nivel de reutilización de los elementos de la interfaz de usuario, además, cada usuario que vaya a acceder al sistema debe validarse y en base al rol que desempeñe solamente se visualiza un menú con las acciones a las que tiene acceso, así desde la interfaz evita recorridos de ida y vuelta innecesarios a los componentes del lado del servidor al acceder a datos que no tiene permiso.

Se utiliza un único modelo de diseño, lo que facilita la conservación de la previsibilidad, el mantenimiento del diseño y la implementación por parte de todos los miembros del equipo de desarrollo.

3.5.2 Estándares de codificación.

Con vistas a garantizar la homogeneidad del código dentro del grupo de desarrollo, se establece el estilo de código descrito a continuación:

1. Organización de los ficheros

1.1 Ficheros de código

Clases/ficheros cortos, que no excedan las 2000 líneas de código.

Hacer las estructuras/clases claras.

Cada clase en un fichero independiente, con nombre igual que la clase.

1.2 Directorio

Crear un directorio para cada espacio de nombre. Por ejemplo, para **Mapificacion.Comunes.Negocio** usar **Mapificacion/Comunes/Negocio**, no usar puntos en el nombre de los espacios de nombre.

2. Indentación

2.1 Líneas largas

Cuando una expresión ocupa más de una línea, debemos dividir la misma atendiendo a los siguientes aspectos:

- Después de una coma.
- Después de un operador.
- Alinear la nueva línea al inicio de la expresión.

Ejemplos de expresiones aritméticas:

```
longName1 = longName2 * (longName3 + longName4 - longName5)
           + 4 * longname6;
var = a * b / (c - g + f) +
     4 * z;
```

Mal estilo (EVITAR):

```
var = a * b / (c - g +
             f) + 4 * z;
```

3. Declaración de clases e interfaces

Se deben seguir las siguientes reglas:

- No dejar espacios en blanco entre el nombre del método y el paréntesis '(' que comienza la lista de parámetros.
- La llave '{' que comienza el bloque de código del método no debe aparecer a continuación de la declaración.
- La llave que cierra el bloque de código del método debe estar sola en una línea. Indentada en correspondencia a la llave '{' que abre el bloque. Por ejemplo:

```
class Ejemplo : OtraClase, IUnaInterface
{
    int entero;

    public Ejemplo(int entero)
```



```
    {  
        this.entero = entero;  
    }  
  
    void Inc()  
    {  
        ++entero;  
    }  
    void MetodoVacio()  
    {  
    }  
}
```

4. Convenciones de declaración

4.1 Estilos de capitalización

4.1.1 Estilo de Pascal

Capitaliza la primera letra de cada palabra, ejemplo: UnContador

4.1.2 Estilo camello

Capitaliza la primera letra de cada palabra, excepto para la primera palabra, ejemplo unContador.

4.1.3 Mayúsculas

Usar este convencionalismo sólo para el caso de identificadores que consisten en abreviaturas de uno o dos caracteres de largo, por ejemplo:

```
public class Math  
{  
    public const PI = ...  
    public const E = ...  
    public const NumeroBaum = ...  
}
```

4.2 Directivas de declaración

- El uso de underscore es considerado una mala práctica. Fue utilizado un tiempo atrás para describir el tipo de una variable, pero ya debe considerarse obsoleto.
- Exceptuando los nombres de los controles de interfaz, un buen nombre de variable describe la semántica, no el tipo.
- Utilizar enumerados para los nomencladores.
- Todas las declaraciones a partir del espacio de nombre Sistema.
- Ejemplo: Sistema.Mapificacion, Sistema.Comunes.Excepciones, Sistema.Administracion, etc.

4.2.1 Clases

- Los nombres de las clases deben ser sustantivos o frases en sustantivo.
- Usar el estilo de Pascal.
- No usar ningún prefijo excepto para las clases que representen acciones que se debe utilizar el prefijo Accion.

4.2.2 Interfaces

- Utilizar sustantivos, frases en sustantivo o adjetivos que describan comportamiento.
- Usar el estilo de Pascal.
- Usar I como prefijo, seguido por una letra mayúscula (primera letra del nombre de la interfaz).

4.2.3 Enumerados

- Usar el estilo de Pascal.
- No poner prefijos (sufijos) al identificador ni a los valores.
- Identificadores en singular.

4.2.4 Campos de sólo lectura y constantes

- Utilizar sustantivos o frases en sustantivo.
- Usar el estilo de Pascal.
- Nombre completo en mayúsculas para las constantes.

4.2.5 Parámetros/campos no constantes

- Usar nombres descriptivos, deben ser suficientes para determinar el significado de la variable y su tipo, preferiblemente su significado.
- Usar el estilo camello.

4.2.6 Variables

- Utilizar i, j, k, l, m, n para contadores en ciclos triviales, en caso contrario utilizar nombres de variables más descriptivos.
- Utilizar el estilo camello.
- Utilizar el prefijo acc para las variables que representen acciones.

4.2.7 Métodos

- Nombrar los métodos con verbos o frases verbales.
- Utilizar el estilo de Pascal.

4.2.8 Propiedades

- Nombrar las propiedades utilizando sustantivos o frases en sustantivo.
- Utilizar el estilo de Pascal.
- Considerar nombrar las propiedades con el mismo nombre de su tipo.

4.2.9 Eventos

- Nombrar los manejadores de eventos con el sufijo Evento.
- Usar el estilo de Pascal.
- Nombrar los eventos que utilizan el concepto de presente y pasado utilizando el tiempo en que ocurren.

4.2.10 Sumario de capitalización

Tipo	Estilo	Notas
Clase / Estructura	Pascal	
Interfase	Pascal	Comenzando con I
Valores de un enumerado	Pascal	
Nombre de un enumerado	Pascal	
Eventos	Pascal	
Clases de excepciones	Pascal	Comenzando con Excepcion
Métodos	Pascal	
Espacios de nombres	Pascal	

Propiedades	Pascal	
Campos protegidos / privados	Camello	
Campos públicos	Camello	
Parámetros	Camello	
Variables locales	Camello	

Tabla 3.7: Sumario de Capitalización

4.3 Idioma

Todas las declaraciones deben ser en idioma español.

3.5.3 Tratamiento de errores.

El tratamiento de errores posibilita el buen funcionamiento de una aplicación dándole mejor apariencia ante los clientes. En el sistema se controlan las entradas de datos del usuario a nivel de interfaz y todas las posibilidades en las que pueda haber un error, cuando éste se produce, ya sea por la entrada incorrecta de un valor o por algún motivo desconocido para el usuario, se brinda información sobre lo ocurrido en la pantalla y donde se encuentra para que sea rectificado. Esto se hace por dos vías: utilizando el control ValidadorTextBox y la librería de Excepciones implementada, cuyas funcionalidades se explican a continuación.

3.5.4 ValidadorTextBox.

El ValidadorTextBox es un control no visual que extiende propiedades a los TextBox y ComboBox contenidos en un formulario, de esta forma permite validar los datos de entrada tanto por teclado como por asignación, y configurar la manera de señalar el TextBox o el ComboBox donde ha ocurrido un error. Esto puede hacerse de las siguientes formas: cambiar el color de fondo del control, utilizando un ErrorProvider para señalarlo, mostrar un ToolTip con el mensaje de error, o cualquier combinación de ellas.

4.5.5 Excepciones.

La librería Excepciones permite dar un adecuado tratamiento a las excepciones que se generan en una aplicación (Ver Figura 4.7), así como todos sus detalles (Ver Figura 4.8) lo que permite conocer el origen del error.

Los datos de las excepciones son almacenadas en un log de la base de datos, en caso de no poder insertarse en la base de datos, se crearía un log en el Registro de Eventos del sistema operativo con la misma información. La librería permite propagar las excepciones hacia capas superiores para su tratamiento.

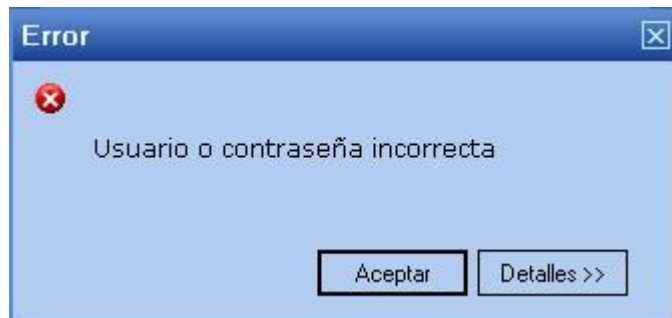


Figura 4.7: Mensaje de error emitido por el sistema.

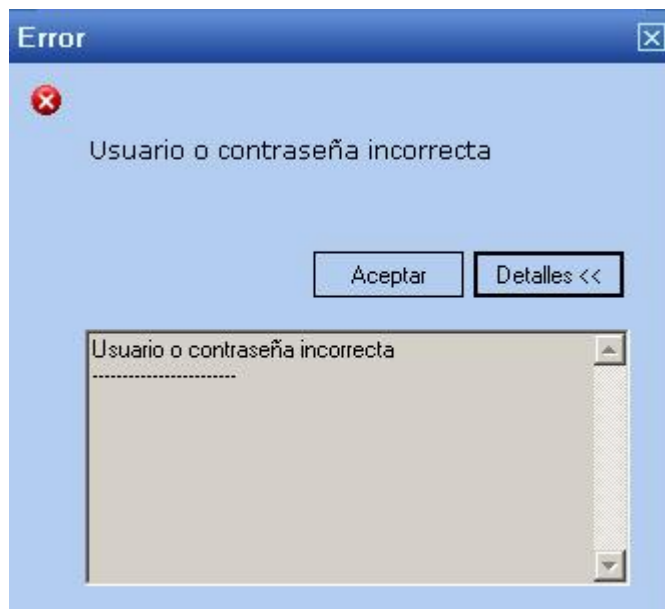


Figura 4.8: Mensaje de error detallado emitido por el sistema.

3.6 Conclusiones.

En el capítulo que se concluye se explicó detalladamente la arquitectura de la aplicación del sistema, los diagramas de clases se dividieron por paquetes para hacer más comprensible la organización del mismo, se obtuvo el modelo físico de la Base de Datos que gestionará la información, así como la descripción de cada tabla. Por último se analizaron los principios de diseño de la interfaz de la aplicación, los estándares de codificación y el tratamiento de errores a emplear durante la fase de construcción.

Capítulo IV: Implementación.

4.1 Introducción.

En este capítulo se ofrece el modelo de implementación dividido en paquetes y las componentes que pertenecen a cada uno, logrando una mejor descripción de los mismos. También se define el modelo de despliegue, los protocolos de comunicación entre los nodos que interactúan en el sistema y una breve descripción de ellos.

4.2 Modelo de Despliegue.

El Modelo de despliegue contiene los nodos (procesador o un dispositivo) que forman la topología hardware sobre la que se ejecuta el sistema. El Módulo de Mapificación de la Información envía o recibe información de dos servidores: el Servidor GPS y el Servidor de Base de Datos central. El Servidor de Base de Datos central utiliza el gestor de Base de Datos Oracle, el cual implementa su propio protocolo de comunicación (TNS).

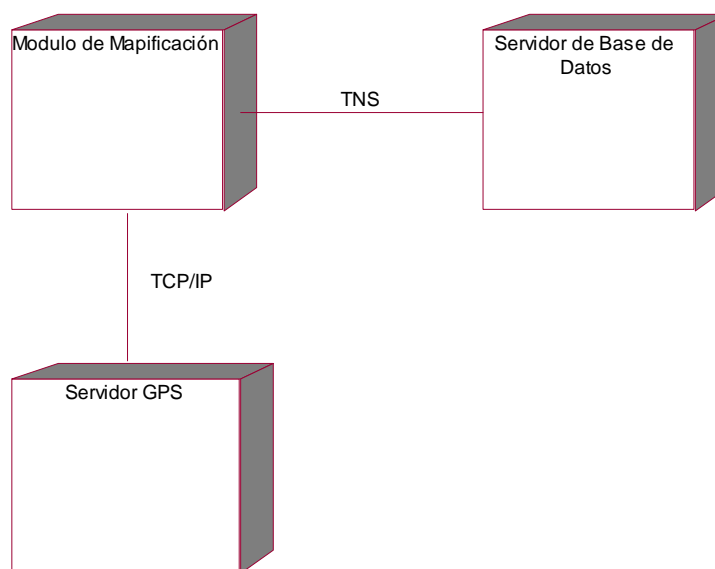


Figura 4.1: Modelo de Despliegue del Módulo de Mapificación de la Información.

4.3 Modelo de Implementación.

El modelo de implementación describe como los elementos del modelo de diseño se implementan en términos de componentes, ficheros de código fuente, ejecutables, etc.; organizándose con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y cómo dependen los componentes unos de otros.

Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases en el modelo de diseño. A continuación se muestra el Modelo de Implementación del Módulo de Mapificación de la Información:

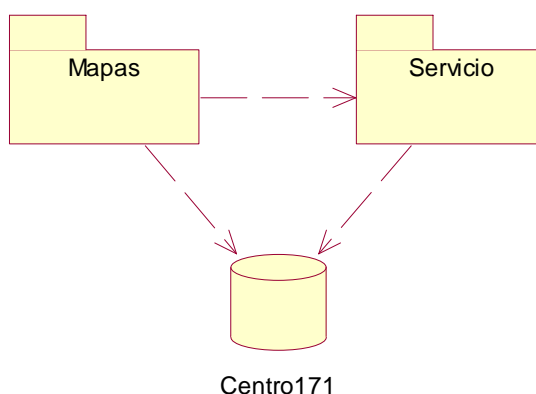


Figura 4.2: Modelo de Implementación.

4.3.1 Diagrama de Componentes del paquete Servicio.

En el paquete Servicio se agrupan un conjunto de componentes que definen las operaciones y acciones básicas para controlar el acceso de los usuarios a las aplicaciones, y brindar un entorno de interfaz común para todos los módulos, estandarizándose el comportamiento visual y funcional del sistema informático del Centro 171. De esta forma se logra la uniformidad que garantiza la calidad y profesionalidad mínimas requeridas para un producto software.

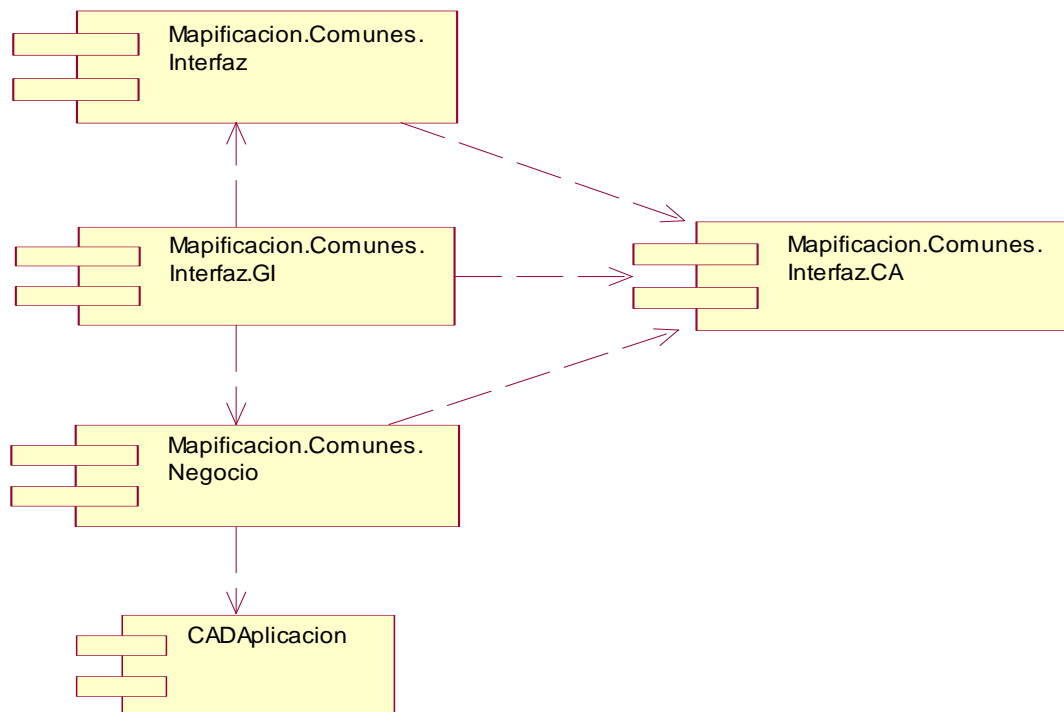


Figura 4.3: Diagrama de Componentes del paquete Servicio.

4.3.2 Diagrama de Componentes del paquete Mapa.

En el paquete Mapa se agrupan los componentes que definen las operaciones y acciones que permiten satisfacer los requerimientos funcionales definidos con el cliente.

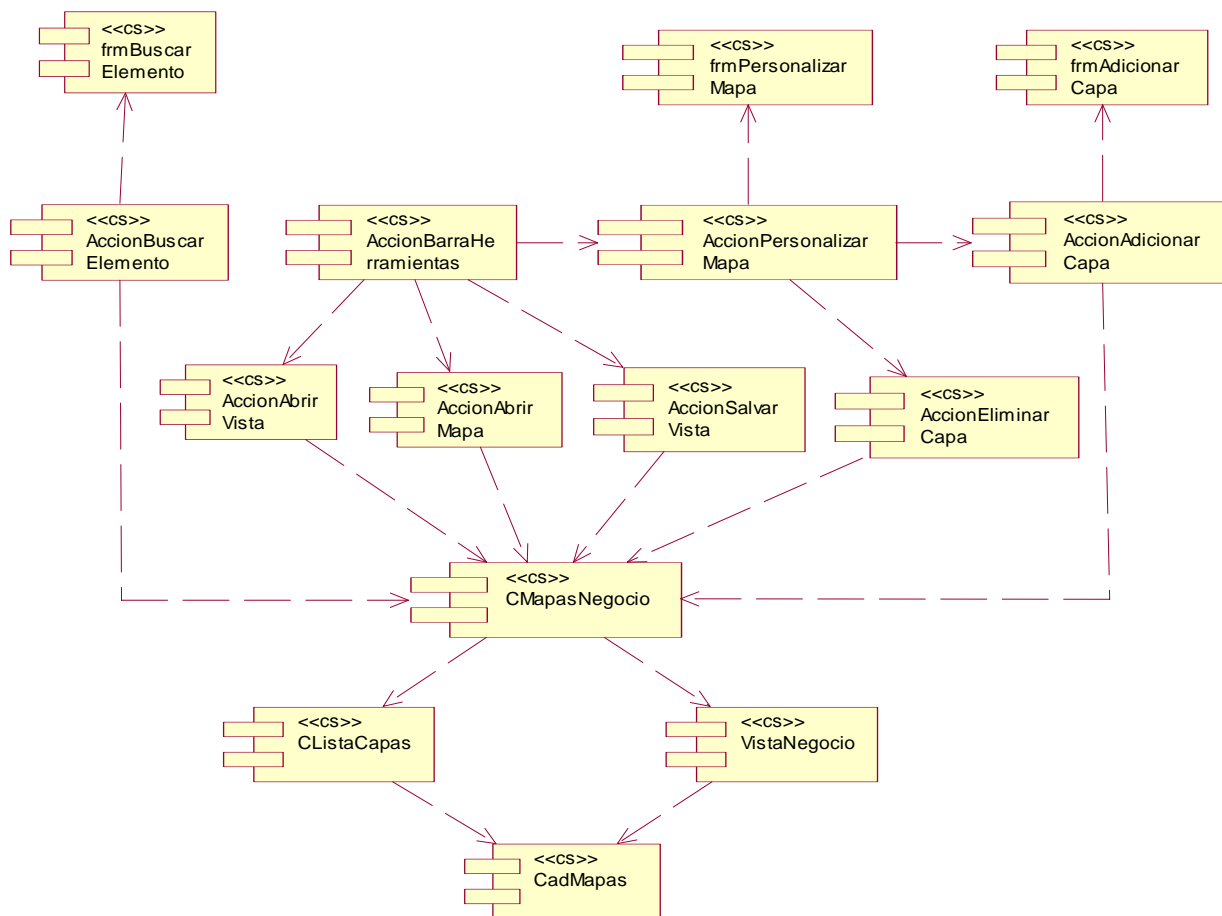


Figura 4.4: Diagrama de Componentes del paquete Mapa.

4.4 Conclusiones.

En este capítulo se mostró detalladamente el modelo de implementación, el cual se divide en paquetes para ofrecer una mejor descripción del mismo y sus componentes, también se realizó el modelo de despliegue que ofrece una visión de los protocolos de comunicaciones utilizados entre el cliente y los servidores.

Conclusiones.

Al finalizar el ejercicio profesional, el sistema informático del **Centro de Gestión de Emergencias y Seguridad Ciudadana (171)** se encuentra en una fase inicial de desarrollo. El Módulo de Mapificación de la Información como parte del sistema automatizado que conforma el Centro, está culminando su primer ciclo. Por ende podemos concluir que se le dio parcial cumplimiento al objetivo general, pues se analizaron, diseñaron e implementaron las funcionalidades correspondientes a los objetivos específicos:

- Permitir al usuario ver en el mapa digital la capa que desee y efectuar cualquier acción básica sobre este, por ejemplo: Acercar, Alejar, Centrar, Mover, Seleccionar, etc.
- Permitir la búsqueda y ubicación de cualquier elemento georreferenciado.
- Mostrar las capas almacenadas en una Base de Datos Geográfica.
- Almacenar en un dispositivo local las vistas seleccionadas.

Recomendaciones.

Continuar con el análisis, diseño e implementación de las funcionalidades del segundo y tercer ciclo de desarrollo del Módulo de Mapificación de la Información del sistema informático del Centro 171.

Migrar la herramienta de desarrollo MapXtreme de la versión 6.2 a la versión 6.5, por las mejoras que esta introduce en aspectos como: la impresión del mapa visualizado, manipulación de los vistas, entre otras.

En vista a la política de migración hacia software libre llevada a cabo por la República Bolivariana de Venezuela, se recomienda en etapas venideras, hacer un profundo estudio sobre la migración paulatina del módulo que presenta este sistema a software libre.

Referencias Bibliográficas.

- [1] Chile, GPS. (2005). “¿Qué margen de error tienen los GPS?” Fecha de acceso: 16-12-2005, disponible en: <http://www.gpschile.com/pagesection/Info.do?m=ver&id=4>
- [2] Consumer, R.(2004) "Los sistemas de navegación por satélites al alcance del bolsillo" Fecha de acceso: 15-12-2005, 2005, disponible en: <http://revista.consumer.es/web/es/20040301/internet/>
- [3] Corabel, S. (2004) “Manejadores de Base de Base de Datos - SQL, ORACLE, INFORMIX” Fecha de acceso: 24-11-2005, 2005, disponible en:
<http://www.ilustrados.com/publicaciones/EpZVVlyFyAbRDtMKhI.php>
- [4] Evidio, J. A (2006) “ASÍ FUNCIONA EL GPS” Fecha de acceso: 05-02-2006, 2006, disponible en: http://www.asifunciona.com/electronica/af_gps/af_gps_14.htm
- [5] Francisco Recio, D. P. (2005) “Arquitectura básica de la plataforma .Net. Descripción del Framework y sus principales componentes: Lenguajes, biblioteca de clases y CLR. ” Fecha de acceso: 23-11-2005, 2005, disponible en: <http://www.desarrolloweb.com/articulos/1328.php?manual=48>
- [6] Frias, Hugo Chávez (2001) “Gaceta Oficial de la República Bolivariana de Venezuela” Retrived 15-01-2006, 2006, disponible en:
<http://www.mintra.gov.ve/legal/leyesordinarias/lleycoordinaciondeseguridadciudadana.html>
- [7] Frias, Hugo Chávez (2001) “LEY DE COORDINACION DE SEGURIDAD CIUDADANA” Fecha de acceso: 28-3-2006, 2006, disponible en: <http://comunidad.vlex.com/pantin/0405.html>
- [8] Ivar Jacobson, B. C, James Runbaugh (2000) “El Proceso Unificado de Desarrollo de Software”

- [9] MapInfo. (2005) "What we Offer" Fecha de acceso: 20-10-2005, 2005, disponible en: <http://www.mapinfo.com/location/integration?txtTopNav=686a2545d8a37f00dev-vcm100001a031dc7&txtLeftNav=a4f43da100145010vcmprd00000034021dc7>
- [10] Ortiz, G (2005) "El funcionamiento del GPS" Fecha de acceso: 08-12-2005, 2005, disponible en: <http://recursos.gabrielortiz.com/index.asp?Info=039>
- [11] Venezuela, Gobierno (2002) "Artículo 55 de la Constitución de la República Bolivariana de Venezuela".

Bibliografía.

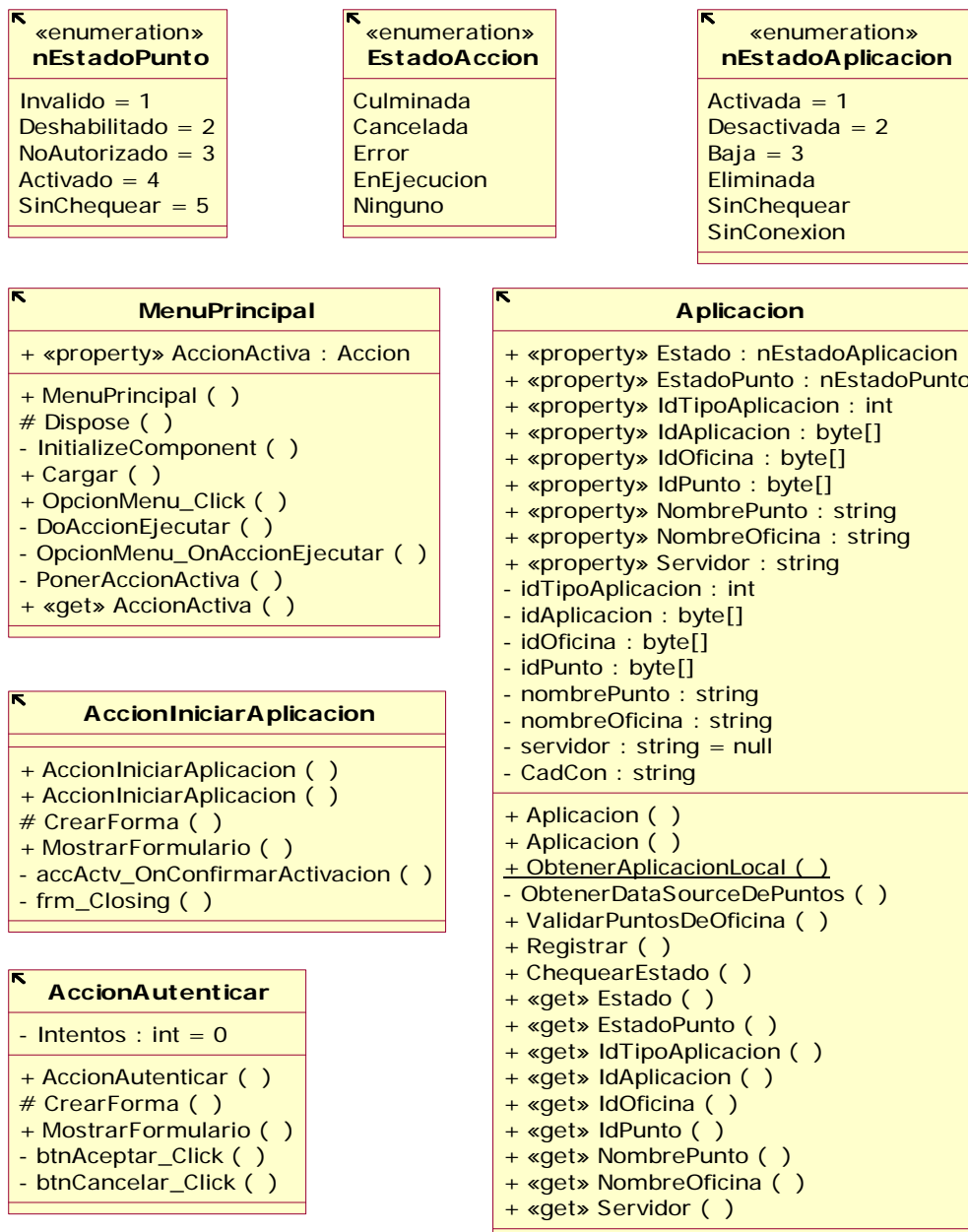
1. BECERRIL, F. *Java a su alcance*, 1999. [Disponible en: <http://bibliodoc.uci.cu/pdf/reg01327.pdf>.
2. BOSQUE SENDRA, J. *Sistemas de Información Geográfica*. Madr(BUZAI 2001)id, 1992. p 225.
3. BUZAI, G. D. *Geomarketing con sistemas de información geográfica*. Madrid, España, 2001. p 459.
4. CLIKEAR. *Arquitectura .NET y Metodologías de Programacion*, 2005. [Disponible en: <http://www.clikear.com/Arquitectura/>.
5. Grupo Asesor del Convenio de Colaboración Cuba Venezuela, *Proyecto Técnico Económico para la creación del Centro de Gestión de Emergencias y Seguridad Ciudadana 171 de la República Bolivariana de Venezuela.*, 2006. p 52.
6. GUTOVNIK, P. *Como funciona el GPS*, 2006. Disponible en: http://gutovnik.com/como_func_sist_gps.htm
7. HUXHOLD, W. y LEVINSOHN, A. *Managing geographic information system project*. New York, Oxford University Press, 1995. p 408.
8. IAN HEYWOOD, S. C., STEVE CARVER. *An Introduction to Geographical Information Systems*. New York, 2004. p. 527.
9. JAMES RUMBAUGH, I. J., GRADY BOOCH. *El lenguaje unificado de modelado. Manual de Referencia*, 2001. Disponible en: <http://bibliodoc.uci.cu/pdf/reg03050.pdf>
10. JEFF FERGUSON, B. P., JASON BERES. *La Biblia de C#*, 2002. Disponible en: <http://bibliodoc.uci.cu/pdf/8441514844.pdf>

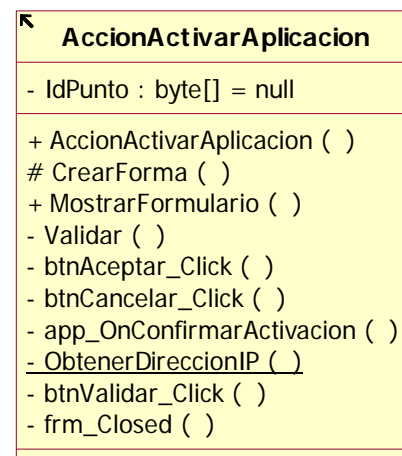
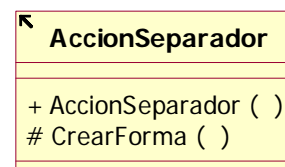
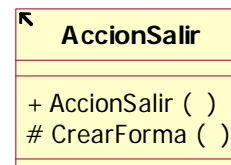
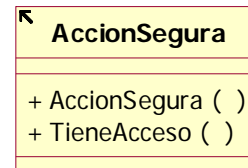
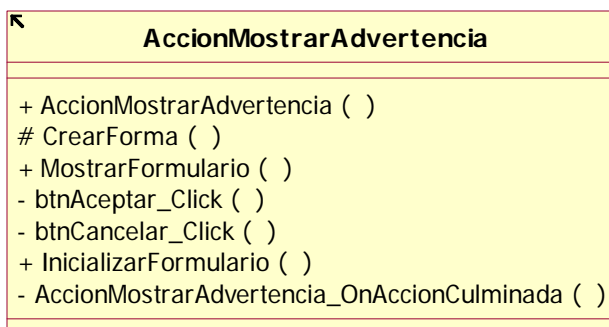
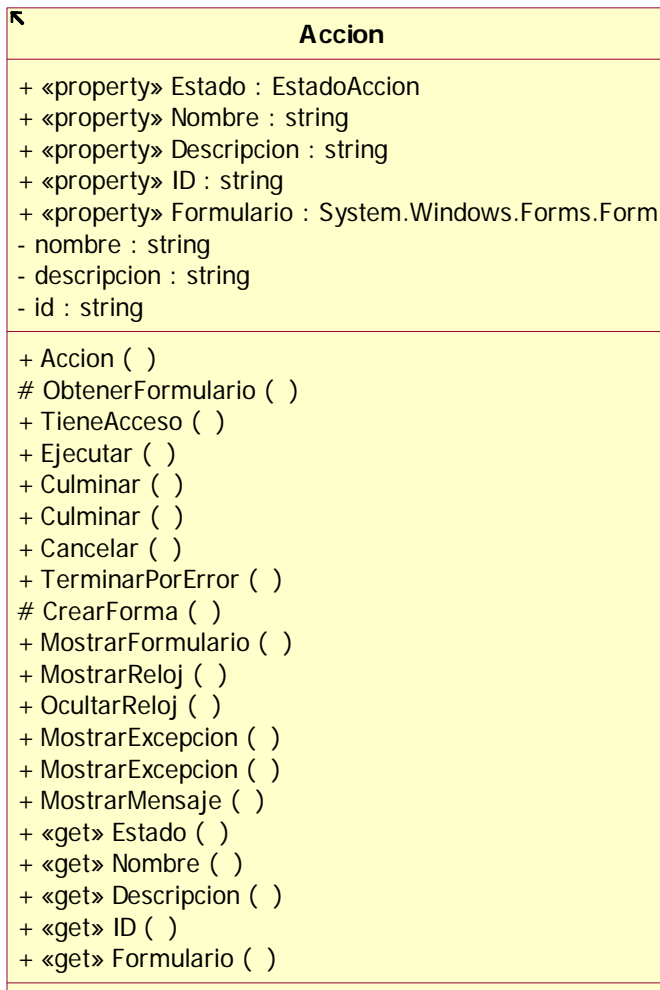
11. MAPINFO, CORPORATION. *MapInfo MapXtreme for Windows*, 2006. Disponible en: <http://extranet.mapinfo.com/support/documentation/manuals.cfm#mapxtremet>
12. MAPINFO, CORPORATION. *MapInfo Professional*, 2006. Disponible en: http://extranet.mapinfo.com/support/documentation/manuals.cfm#mi_pro
13. MAPTOOLS. *Desktop Tools*, 2006. Disponible en: <http://www.maptools.org/>
14. MICHAEL ABBEY, M. C., IAN ABRAMSON. *Oracle 9i. Guía de aprendizaje*, 2000. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02472.pdf>
15. MICROSOFT, CORPORATION. *SQL Server 2000*, 2004. Disponible en: <http://www.microsoft.com/sql/default.aspx>
16. MICROSOFT, CORPORATION. *What is .NET?*, 2005. Disponible en: <http://www.microsoft.com/net/basics.aspx>
17. MICROSOFT, CORPORATION. *.NET Development*, 2005. Disponible en: [http://msdn2.microsoft.com/en-us/library/ms310245\(en-us,MSDN.10\).aspx](http://msdn2.microsoft.com/en-us/library/ms310245(en-us,MSDN.10).aspx)
18. MINNESOTA, UNIVERSITY. *Utilizando Oracle Spatial con Mapserver*, 2006. Disponible en: <http://www.cartesia.org/article.php?sid=256>
19. ORACLE. *Oracle Ibérica*, 2005. Disponible en: <http://www.oracle.com/global/es/index.html>
20. ORACLE. *Oracle Spatial & Oracle Locator: Location Features for Oracle Database 10g*, 2006. Disponible en: <http://www.oracle.com/technology/products/spatial/index.html>
21. RALMALHO, J. A. *SQL Server. Iniciación y referencia*, 2000. [Disponible en: <http://bibliodoc.uci.cu/pdf/reg01719.pdf>

22. ROGER S, P. *Ingeniería del Software*, 2001. [1]. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02689.pdf>
23. SCHMULLER, J. *Aprendiendo UML en 24 horas*, 2000. [1]. Disponible en:
<http://bibliodoc.uci.cu/pdf/reg00004.pdf>
24. URMAN, S. *Oracle 8i. Programación avanzada con PL-SQL*, 2001. [Disponible en:
<http://bibliodoc.uci.cu/pdf/reg01431.pdf>

Anexos.

Clases del Paquete Servicio:







Clases del Paquete Mapas

```

AccionBuscarElemento
+ AccionBuscarElemento ( )
# CrearForma ( )
+ MostrarFormulario ( )
- LLenarComboNombreMapas ( )
- cbxNombreCapas_SelectedIndexChanged ( )
- cbxNombreCapas_TextChanged ( )
- cbxParametrosBusqueda_SelectedIndexChanged ( )
- lstbxElementosBusqueda_SelectedIndexChanged ( )
- lstbxElementosBusqueda_DoubleClick ( )
- btnCancelar_Click ( )
- btnBuscar_Click ( )
- MostrarPuntoEnMapa ( )
- cbxParametrosBusqueda_TextChanged ( )
- txtElementosBusqueda_KeyUp ( )
    
```

```

frmBuscarElemento
+ frmBuscarElemento ( )
# Dispose ( )
- InitializeComponent ( )
    
```

```

AccionAdicionarCapa
+ AccionAdicionarCapa ( )
# CrearForma ( )
+ MostrarFormulario ( )
- CargarCapasBD ( )
- btnExaminar_Click ( )
- btnCancelar_Click ( )
- btnAdicionar_Click ( )
    
```

```

AccionAbrirVista
+ AccionAbrirVista ( )
# CrearForma ( )
- AbrirVista ( )
- CrearVistaAuxiliar ( )
- LeerVista ( )
    
```

```

AccionPersonalizarMapa
+ AccionPersonalizarMapa ( )
# CrearForma ( )
+ MostrarFormulario ( )
- CargarCapasVisibles ( )
+ AdicionarCapas ( )
+ EliminarCapas ( )
- btnAdicionarCapa_Click ( )
- btnAplicar_Click ( )
- btnAceptar_Click ( )
- btnCancelar_Click ( )
- btnEliminarCapa_Click ( )
    
```

```

frmAdicionarCapa
+ frmAdicionarCapa ( )
# Dispose ( )
- InitializeComponent ( )
    
```

```

AccionSalir
+ AccionSalir ( )
# CrearForma ( )
    
```

```

frmPersonalizarMapa
+ frmPersonalizarMapa ( )
# Dispose ( )
- InitializeComponent ( )
    
```

```

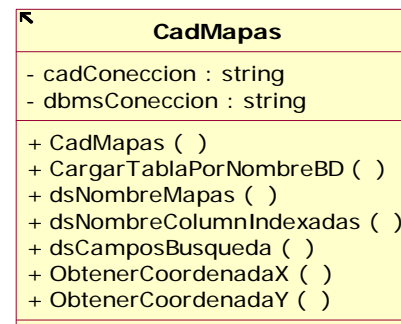
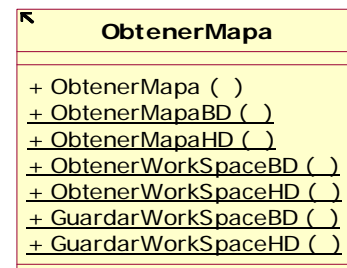
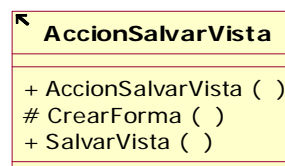
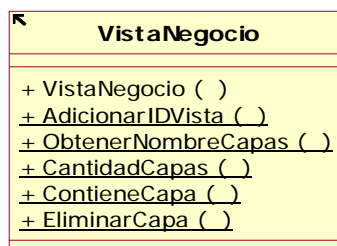
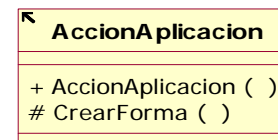
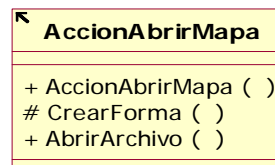
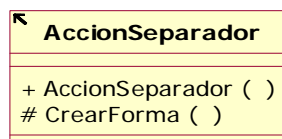
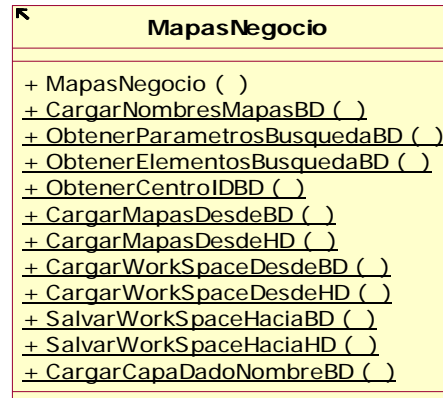
AccionCerrarMapa
+ AccionCerrarMapa ( )
# CrearForma ( )
+ Culminar ( )
    
```

```

AccionBarraHerramientas
- zoomMenor : MapInfo.Geometry.Distance
- zoomMayor : MapInfo.Geometry.Distance
+ AccionBarraHerramientas ( )
# CrearForma ( )
- mapToolBarPrincipal_ButtonClick ( )
- mnItemAbrirMapa_Click ( )
- mnItemAbrirMapaInicial_Click ( )
- mnItemAbrirVista_Click ( )
- DeshabilitarZoomMouse ( )
- Tools_Ending ( )
    
```

```

AccionAbrirMapaBD
+ AccionAbrirMapaBD ( )
# CrearForma ( )
+ AbrirMapaBD ( )
    
```



Glosario de Términos.

Elementos: Un elemento es la unidad de información básica de la geometría. Los tipos espaciales apoyados del elemento son puntos, líneas, y polígonos.

Geometrías: Es la representación de una característica espacial, modelada como conjunto ordenado de elementos primitivos. Puede consistir en un solo elemento o una colección (homogénea o heterogénea) de elementos.

Capas: Una capa es una colección de geometrías que tienen el mismo conjunto de atributos.

Sistema Coordinado: Permite la interpretación de un sistema de coordenadas como representación de una posición en un espacio verdadero del mundo.

Sistema Coordinado georreferenciado: Sistema coordinado relacionado con una representación específica de la tierra.

Sistema Coordinado no georreferenciado: Sistema coordinado cartesiano, y no está relacionado con una representación específica de la tierra.

Órganos de seguridad ciudadana: Entidades del estado que velan por el bienestar social de cada ciudadano.

- Las Policías de cada Estado.
- Las Policías de cada Municipio, y los servicios mancomunados de policías prestados a través de la Policía Metropolitana.
- El Cuerpo de Investigaciones Científicas, Penales y Criminalísticas.
- El Cuerpo de Bomberos y Administración de Emergencias de Carácter Civil.
- La Organización de Protección Civil y Administración de Desastre.

Móviles en Servicio: Carros de patrullas, bomberos, motos, etc. que presentan un dispositivo GPS.

SIG: Sistema de Información Geográfico.

BD: Base de Datos.

PC: Computador Personal.

BDG: Base de Datos Geográfica.

Centro 171: Centro de Gestión de Emergencias y Seguridad Ciudadana (171).

CU: Casos de Uso.

TNS: (Transparent Network Substrate, Sustrato de red transparente): Permite la comunicación entre los clientes y los servidores de bases de datos Oracle con independencia del protocolo de comunicaciones que se utilice.