

Universidad de las Ciencias Informáticas
FACULTAD 6



**Sistema de Información Geográfica
para automatizar la gestión de la distribución de las
redes eléctricas en la Unión Nacional Eléctrica**

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

Autor: Sandor Escobar Ruiz
Tutor: Ing. Alberto Menendez Romero

Ciudad de La Habana, Junio 2012
“Año 54 de la Revolución”

Frase

“La posibilidad de realizar un sueño es lo que hace que la vida sea interesante”.

“Todos los días Dios nos da un momento en que es posible cambiar todo lo que nos hace infelices. El instante mágico es el momento en que un sí o un no pueden cambiar toda nuestra existencia”.

“Sólo una cosa vuelve un sueño imposible: el miedo a fracasar”.

Paulo Coelho

Declaración de Autoría

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Sandor Escobar Ruiz
Autor

Ing. Alberto Menendez Romero
Tutor

DATOS DEL CONTACTO

Tutor: Ing. Alberto Menendez Romero

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

Agradecimientos

AGRADECIMIENTOS

Agradezco la realización de este trabajo a las personas más importantes de mi vida:

A mi Mamá y mi Papá, Nancy y Justo, sin los cuales no estaría donde estoy, ni sería hoy quien soy, por ser esas personas tan maravillosas y a las que le debo la vida, por estar siempre apoyándome, dándome aliento en todos los momentos difíciles, por ayudarme siempre a seguir hacia adelante, por confiar en mí, por darme su amor, cariño y dedicación por sobre todas las cosas. Los quiero mucho...

A mi hermana y mi cuñado, Sandra y Julio, por ser guía de mi inspiración, ejemplos a seguir, por apoyarme en todo momento haciendo posible que haya cumplido este gran sueño, por confiar en mí, por darme su amor, cariño y dedicación, por haber estado ahí siempre que los necesité, gracias por todo. Los quiero mucho...

A mi novia, Lesdy, por ser tan buena amiga y compañera, por apoyarme, aguantarme y sobrellevarme todo este tiempo, por estar a mi lado cuando más lo he necesitado, por darme tantos momentos de felicidad. Te quiero mucho...

Al tribunal de esta Tesis por señalarme con paciencia mis errores. Gracias...

A mi tutor, Alberto Menéndez Romero, por su apoyo y ayuda en el desarrollo de la investigación. Gracias...

A todos los profesores que de una forma u otra contribuyeron a que este sueño se hiciera realidad y por brindarme todos sus conocimientos. Gracias...

A todas mis amistades y a esas personas que me han ayudado o han ayudado a las personas que quiero, directa o indirectamente. Gracias...

Dedicatoria

DEDICATORIA

Dedico este trabajo, todo mi esfuerzo, dedicación y méritos alcanzados a:

A mi mamá y a mi papá, Nancy y Justo, por brindarme todo su apoyo, confianza y amor, por ser las personas que más amo en el mundo, quienes me han dado todo lo que poseo, desde mi imagen hasta mi espíritu.

A mi hermana y mi cuñado, Sandra y Julio, por ser un ejemplo a seguir, por darme su confianza y apoyo en todo momento, por ser esas personas con las que siempre podré contar.

A lo más lindo de la familia, mi sobrina Claudia, la que adoro con la vida.

A mis abuelas y abuelos, Graciela, Irma, Ramón y Pablo, por amarme y quererme tanto, gracias por estar para mí.

A mi novia Lesdy, por estar siempre a mi lado por tener tanta confianza y paciencia conmigo.

A todos ustedes les dedico este momento de mi vida.

Los quiere Mucho. Sandor.

RESUMEN

Uno de los principales avances en el desarrollo de la ciencia y la tecnología en los últimos años ha sido la capacidad de uso y control de la energía eléctrica, esto implica la generación, transmisión y distribución de la corriente eléctrica a los consumidores. La planeación de las operaciones de los sistemas eléctricos para las condiciones existentes, así como las expansiones futuras, requieren actualmente de los Sistemas de Información Geográfica, capaces de visualizar con exactitud y en tiempo real las operaciones de administración de la red eléctrica. Con la presente investigación se pretende desarrollar un Sistema de Información Geográfica para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica. Lo que traerá como principal ventaja permitir la edición de la cartografía digital de las redes eléctricas en una aplicación de escritorio que brinde mayor calidad y rapidez, eliminando errores que pueden introducirse de forma manual y perfeccionar los resultados de análisis para la toma de decisiones en el proceso de distribución de las redes eléctricas. Además generar la documentación técnica, donde se describan todos los artefactos generados correspondientes a cada flujo de trabajo. Para darle cumplimiento a esta investigación se realizará un estudio minucioso de las tendencias y tecnologías que existen a nivel mundial, donde se elegirán las herramientas y tecnologías correctas para llevar a cabo un buen desarrollo del sistema propuesto, con mayor calidad y rapidez, bajo tecnologías libres y multiplataforma en su mayoría.

Palabras claves:

Automatizar, distribución, redes eléctricas, Sistemas de Información Geográfica.

TABLA DE CONTENIDO

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Introducción	6
1.2 Conceptos asociados al dominio del problema.....	6
1.2.1 Sistemas de Distribución Eléctrica.....	6
1.2.2 Sistema de Información Geográfica.....	6
1.3 Objeto de Estudio	7
1.3.1 Descripción General.....	7
1.3.2 Descripción actual del dominio del problema.....	9
1.3.3 Situación Problemática	10
1.4 Análisis de otras soluciones existentes.....	10
1.5 Conclusiones Parciales.....	12
CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A UTILIZAR	13
2.1 Introducción	13
2.2 GeoQ	13
2.3 Arquitectura de Software	13
2.3.1 Arquitectura Orientada a Objetos.....	14
2.3.2 Arquitectura en Capas	15
2.4 Metodología de desarrollo de software	16
2.4.1 Proceso Unificado Racional (RUP)	16
2.5 Lenguaje de Modelación.....	17
2.5.1 Lenguaje Unificado de Modelado (UML).....	17
2.6 Herramienta CASE	18
2.6.1 Visual Paradigm	19
2.7 Lenguaje de Programación	19
2.7.1 C++.....	20
2.8 Entorno de Desarrollo Integrado (IDE).....	20
2.8.1 Qt Creator	20
2.9 Sistemas Gestores de Base de Datos.....	21

2.9.1	PostgreSQL	21
2.9.2	PostGIS	22
2.10	Conclusiones Parciales.....	23
CAPÍTULO 3: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA		24
3.1	Introducción	24
3.2	Modelo de Negocio.....	24
3.2.1	Actor y trabajadores del negocio.....	24
3.2.2	Restricciones del negocio.....	26
3.2.3	Diagrama de caso de uso del Negocio	26
3.2.4	Diagrama de Actividades.....	26
3.2.5	Descripción textual de los casos de uso del Negocio.....	27
3.3	Requisitos Funcionales.....	28
3.4	Requisitos No Funcionales.....	31
3.4.1	Requerimientos de usabilidad.....	32
3.4.2	Interfaces de usuario.....	32
3.4.3	Interfaces de software.....	32
3.4.4	Interfaces de hardware	32
3.4.5	Interfaces de Comunicación.....	33
3.5	Descripción del Sistema Propuesto	33
3.5.1	Descripción del Actor del Sistema	33
3.5.2	Diagrama de casos de uso del Sistema.....	33
3.5.3	Descripción textual de los Casos de Uso del Sistema.....	34
3.6	Conclusiones Parciales.....	45
CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....		46
4.1	Introducción	46
4.2	Patrones de Diseño	46
4.3	Modelo de Diseño.....	47
4.3.1	Diagrama de Clases del Diseño.....	47
4.4	Principios de Diseño.....	48
4.4.1	Estándares de la Interfaz de Aplicación.....	48
4.5	Diseño de la Base de Datos.....	49

Índice

4.5.1	Diagrama de Clases Persistentes	50
4.5.2	Modelo Entidad-Relación.....	50
4.6	Generalidades de la Implementación	51
4.6.1	Modelo de Despliegue	51
4.6.2	Modelo de Implementación	51
4.7	Prueba del Sistema Propuesto	52
4.7.1	Prueba de Caja Negra	53
4.8	Conclusiones Parciales.....	58
CONCLUSIONES GENERALES.....		59
RECOMENDACIONES.....		60
TRABAJOS CITADOS		61
BIBLIOGRAFÍA		63
GLOSARIO DE TÉRMINOS.....		65

Índice de Tablas y Figuras

ÍNDICE DE FIGURAS

Figura 1: Modelo de Objetos del Negocio.	25
Figura 2: Diagrama de Caso de Uso del Negocio.	26
Figura 3: Diagrama de Actividades.	27
Figura 4: Diagrama de Casos de Uso del Sistema.	34
Figura 5: Diagrama de Clases del Diseño: Gestionar objeto eléctrico.	48
Figura 6: Diagrama de Clases Persistentes.	50
Figura 7: Modelo Entidad-Relación.	50
Figura 8: Modelo de Despliegue.	51
Figura 9: Diagrama Componente: Gestionar objeto eléctrico.	52

ÍNDICE DE TABLAS

Tabla 1: Actor del Negocio.	24
Tabla 2: Trabajadores del Negocio.	25
Tabla 3: Descripción del Caso de Uso del Negocio Reportar Incidencia.	28
Tabla 4: Descripción del actor del sistema.	33
Tabla 5: Descripción del caso de uso Ubicar objeto eléctrico.	36
Tabla 6: Descripción del caso de uso Gestionar objeto eléctrico.	45
Tabla 7: Secciones a probar en Guardar el estado de la red en un momento dado.	55
Tabla 8: Descripción de las variables en Guardar el estado de la red en un momento dado.	55
Tabla 9: Matriz de datos en Guardar el estado de la red en un momento dado.	55
Tabla 10: Secciones a probar en Mostrar el estado de la red en un momento dado.	56
Tabla 11: Descripción de las variables en Mostrar el estado de la red en un momento dado.	56
Tabla 12: Matriz de datos en Mostrar el estado de la red en un momento dado.	57
Tabla 13: Secciones a probar en Retornar al estado inicial.	57
Tabla 14: Matriz de datos en Retornar el estado inicial.	57
Tabla 15: Secciones a probar en Establecer un cambio anterior.	58
Tabla 16: Matriz de datos en Establecer un cambio anterior.	58

INTRODUCCIÓN

En la actualidad existe un elevado desarrollo en las Tecnologías de la Información y la Comunicación (TIC), permitiendo informatizar los principales problemas de la sociedad tales como: el fácil acceso a todo tipo de información, el desarrollo de diferentes canales de comunicaciones, el almacenamiento de grandes volúmenes de datos, la automatización de tareas, entre otras disímiles. Esto ha permitido gestionar grandes fuentes de información, posibilitando mayor precisión, evitando posibles errores que pueden ocurrir con el trabajo manual. Para contribuir al desarrollo de la informatización de la sociedad en Cuba; se crea en el año 2002 la Universidad de las Ciencias Informáticas (UCI), para potenciar y fomentar el desarrollo de software dentro del país y tratar de incluir al mismo en la élite mundial de la Informática. La UCI, cuenta con diversos centros de desarrollo, los cuales están enfrascados en la implementación de sistemas informáticos. La producción está repartida entre sus siete facultades, definiendo para cada una de ellas perfiles y centros productivos que agrupan un conjunto de proyectos a desarrollar. Dentro de estos centros se encuentra GEYSED dedicado al estudio de la Geoinformática y las Señales Digitales, el cual cuenta con el proyecto SIG-DESKTOP, encargado de desarrollar Sistemas de Información geográfica (SIG), donde se encuentra enmarcada la presente investigación. SIG-DESKTOP cuenta con GeoQ, una plataforma de escritorio para el manejo, análisis y visualización de información geográfica, desarrollada con tecnologías libres y multiplataforma. Esta plataforma permite el desarrollo de nuevas funcionalidades mediante personalizaciones, permitiendo adaptar el sistema a diferentes entornos de trabajo, contribuyendo a la informatización de la sociedad en Cuba.

Uno de los principales eslabones para el desarrollo de las TIC y que posibilita el avance de la informatización en el país, es el suministro de la energía eléctrica. Servicio que es brindado por la Unión Nacional Eléctrica (UNE) órgano rector en este sentido, encargado de la gestión y administración de las redes de distribución eléctrica en el país. La UNE brinda servicio eléctrico a clientes industriales, comerciales, rurales y urbanos cubriendo toda la demanda nacional. Para que esto sea posible la distribución de la energía eléctrica debe llevarse a cabo con redes bien diseñadas, ser proyectadas y construidas de manera que tengan flexibilidad suficiente para ampliarse progresivamente con cambios mínimos en las construcciones existentes, y así asegurar un servicio adecuado y continuo con un mínimo de costo de operación. El crecimiento continuo de las redes de distribución eléctrica ha ocasionado que su análisis y gestión requiera de herramientas de diseño asistido por computadoras. En general, la electricidad podría ser aceptada en el sentido de suministro de corriente eléctrica, esto implica la generación, transmisión y distribución de la corriente eléctrica a los consumidores. La electricidad es uno de los aspectos del sector de servicios públicos que es

esencial para el buen desarrollo de la informática y la sociedad, es indispensable para la economía y promueve el bienestar de los individuos. El funcionamiento eficiente de esta actividad es de suma importancia para el sustento de su crecimiento y su consecuente realización de los planes y objetivos de gestión. Por lo tanto, la toma de decisiones sabias es vital para las operaciones, el crecimiento y gestión de las instalaciones de distribución de energía eléctrica, la información debe ser recopilada y analizada en toda su extensión, tal información contribuye no sólo a servicios eficientes, sino también a las operaciones, el mantenimiento de los activos y a la planificación racional de las extensiones y nuevas obras.

La planeación de las operaciones de los sistemas eléctricos para las condiciones existentes, así como las expansiones futuras, requieren actualmente de los Sistemas de Información Geográfica, capaces de visualizar con exactitud y en tiempo real las operaciones de administración de la red eléctrica. Los SIG juegan un papel primordial en este sentido, con la facilidad para el manejo de la información, permiten a los usuarios crear consultas interactivas, analizar la información espacial y de la red, editar datos, mapas, representar los resultados de todas estas operaciones y pueden almacenar y manipular la información de forma visual y sencilla, facilitando el trabajo con grandes volúmenes de datos, ahorrando tiempo y costos.

La gestión de la distribución de la energía eléctrica en la Unión Nacional Eléctrica se realiza en diferentes departamentos. El departamento de despacho realiza el análisis del sistema diario, documentando todas las afectaciones que puedan ocurrir en la red eléctrica de forma manual, al finalizar la jornada de trabajo, la documentación es entregada al departamento de diseño con los cambios ocurridos en formato duro. En este departamento un grupo de dibujantes, con la ayuda de la herramienta informática de diseño AutoCAD, represente los nuevos cambios en el mapa de forma estático, para ser utilizado en la próxima jornada de trabajo para su correspondiente análisis. Esta tarea resulta compleja, pues los grandes volúmenes de información que se manejan en formato duro y el uso de la herramienta AutoCAD que no está diseñada para darle solución a este tipo de problema, pudieran introducir errores y resultados de análisis incorrectos para la toma de decisiones.

Es por ello que se hace necesaria la existencia de un Sistema de Información Geográfica para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica con el objetivo de aumentar la eficiencia y mejorar la operación y administración de la red.

Por lo anteriormente planteado se define como **problema a resolver**: el proceso de distribución de las redes eléctricas en la Unión Nacional Eléctrica, se realiza con el apoyo de herramientas informáticas

que no están diseñadas para darle solución a este tipo de problema, lo que posibilita en gran medida la ocurrencia de errores y, por ende, la obtención de resultados de análisis incorrectos para la toma de decisiones.

Planteándose como **objeto de estudio**: los procesos de la distribución de las redes eléctricas en la Unión Nacional Eléctrica, específicamente en la automatización de los procesos de gestión de la distribución de las redes eléctricas en la UNE, lo que constituye el **campo de acción** de la investigación.

Se define como **objetivo general** del presente trabajo de diploma: desarrollar un Sistema de Información Geográfica que permita automatizar el proceso de distribución de las redes eléctricas en la UNE.

Como parte de la investigación se propone la siguiente **idea a defender**:

El desarrollo de un Sistema de Información Geográfica para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica permitirá, eliminar errores que pueden introducirse de forma manual y perfeccionar los resultados de análisis para la toma de decisiones en el proceso de distribución de las redes eléctricas en la UNE.

Para darle cumplimiento a los objetivos trazados se determinó que las **tareas** a realizar en la presente investigación son:

1. Caracterizar los procesos de distribución de las redes eléctricas en la UNE.
2. Realizar una revisión de las tendencias y tecnologías actuales aplicables al proceso de negocio.
3. Modelar los procesos de Negocio, Diseño e Implementación del sistema.
4. Implementar la solución propuesta.
5. Probar y validar la solución propuesta.

Como parte del cumplimiento de las tareas investigativas se espera obtener los siguientes **resultados**:

1. Un Sistema de Información Geográfica para automatizar el proceso de gestión de la distribución de las redes eléctricas en la UNE.
2. La documentación técnica asociada al desarrollo de la solución propuesta.

A lo largo de la investigación se tendrán en cuenta métodos científicos que serán de suma importancia para la realización satisfactoria de la misma.

Los **métodos** empleados en el desarrollo de la presente investigación fueron:

Métodos Teóricos

- **Histórico - Lógico:** En la primera fase de la investigación se desarrolla un estudio del estado del arte de la problemática analizada, revisando de forma crítica cada uno de los documentos para lograr un mejor entendimiento de lo que se debe desarrollar.
- **Analítico - Sintético:** Se realiza un estudio con profundidad de toda la información acerca de las tecnologías, metodologías y herramientas posibles a ser utilizadas, pudiendo definir con mayor certeza las mismas, para un mejor entendimiento de la situación y luego poder sintetizarlos para la confección de la solución.
- **Modelación:** Se utiliza para la modelación de diagramas, representar el proceso de desarrollo y propiciar un mejor entendimiento de la solución a implementar.

Métodos Empíricos:

- **Observación:** Se realizarán varias visitas a la Unión Nacional Eléctrica (UNE) con el objetivo de observar los procesos que allí se desarrollan, las técnicas y mecanismos de representación de las redes eléctricas.
- **Entrevistas:** Se utilizará este método para establecer el alcance del presente trabajo, el objetivo, el flujo actual de la información, así como los requerimientos que contendrá la aplicación a desarrollar. Dentro de la entrevista se seleccionó el tipo de entrevista no estructurada, por ser más flexible y abierta; como población se cuenta con los ingenieros eléctricos de la UNE, y como muestra aleatoria 3 especialistas del departamento de despacho y 3 del departamento de diseño.

La investigación está dividida en cuatro **capítulos** fundamentales:

Capítulo 1: En este capítulo se describen brevemente los conceptos fundamentales relacionados con el dominio del problema. Se realiza un estudio acerca de los SIG y los sistemas eléctricos, además de hacer un análisis de otras soluciones existentes, nacionales e internacionales.

Capítulo 2: Se argumenta la selección de las tecnologías y herramientas que se utilizan en el proceso de desarrollo del software, lenguajes de programación y modelado, así como la arquitectura, la metodología a utilizar y el entorno de desarrollo.

Capítulo 3: En este capítulo se expone la solución propuesta con todos sus argumentos, el modelo de negocio, los requerimientos tanto funcionales como no funcionales, además de la descripción de los casos de usos y del actor del negocio y del sistema.

Capítulo 4: Se exponen los diagramas de clases del diseño para cada caso de uso del sistema, los diagramas de componentes, el diagrama de despliegue y se realizan los diseños de casos de prueba.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se describen los conceptos asociados al dominio del problema, así como los aspectos y conceptos generales relacionados con el tema de los SIG y su aplicación en las redes eléctricas, además una descripción del estado del arte del tema a tratar. Al finalizar el capítulo se dejan sentadas las bases necesarias para un correcto análisis durante el resto de la investigación.

1.2 Conceptos asociados al dominio del problema

Los conceptos asociados al dominio del problema se toman a partir de las definiciones más actualizadas establecidas internacionalmente por organizaciones mundiales y regionales de los SIG y los Sistemas Eléctricos; estos conceptos son útiles para el entendimiento del problema en cuestión, y ayuda a entender las relaciones que existen entre ellos.

1.2.1 Sistemas de Distribución Eléctrica

Un sistema de distribución eléctrica es el conjunto de elementos encargados de conducir la energía desde una subestación de potencia hasta el usuario. Básicamente, la distribución de energía eléctrica comprende las líneas primarias de distribución, los transformadores de distribución, las líneas secundarias de distribución, las acometidas y medidores. (Morón, 2009)

La función de los sistemas de distribución es recibir la energía eléctrica de las centrales de producción o estaciones primarias y distribuir a los usuarios a la tensión adecuada, con la conveniente continuidad y calidad de suministro para los distintos usos. (Sanabria, 1993)

1.2.2 Sistema de Información Geográfica

El término SIG procede del acrónimo de Sistema de Información Geográfica (en inglés GIS, Geographic Information System). Técnicamente se puede definir un SIG como una tecnología de manejo de información geográfica formada por equipos electrónicos (hardware) programados adecuadamente (software) que permiten manejar una serie de datos espaciales (información geográfica) y realizar análisis complejos con éstos siguiendo los criterios impuestos por el equipo científico (personal). (Sanabria, 2011)

1.3 Objeto de Estudio

1.3.1 Descripción General

Características de los Sistemas de Distribución

Las redes de distribución presentan características muy particulares. Entre éstas se distinguen: (Sanabria, 1993)

- Topologías radiales.
- Razón R/X alta (líneas de resistencia comparables a la reactancia).
- Múltiples conexiones (monofásicas, bifásicas, etc.).
- Estructura lateral compleja.
- Cargas de distinta naturaleza.
- Líneas sin transposiciones.
- Cargas distribuidas.

Clasificación de los Sistemas de Distribución

Los sistemas de distribución pueden clasificarse de diversas formas: (Sanabria, 1993)

- Según la carga: alumbrado público, industrial, comercial, residencial, mixto.
- Según la corriente: continua y alterna.
- Según la tensión: distribución primaria, distribución secundaria.
- Según su topología: radial, anillo, enmallada.
- Según el número de conductores: bifilar, trifilar, a cuatro hilos, etc.
- Según el tipo de instalación: aérea, subterránea.

Componentes de los Sistemas de Distribución

En todo sistema de distribución suelen encontrarse los siguientes elementos: alimentadores, transformadores, líneas y cables, capacitores o condensadores y equipos de protección. (Sanabria, 1993)

Modelos de datos de los SIG

Los SIG trabajan con diferentes tipos de datos, entre los principales: (Olaya, 2010)

- **Modelos Ráster:** Es una de las estructuras de datos más simple de organización celular de los datos espaciales. En una estructura Ráster un valor para el parámetro de interés (clase de uso de tierra, tipo de suelo, etc.) se desarrolla para cada celda un vector sobre el espacio. A cada celda en un archivo Ráster se le asigna solamente un valor, los atributos diferentes son almacenados como archivo de datos separados. Los modelos principales son: Modelo de cuadrícula, Mosaico regular, Mosaico anidado y Mosaico irregular.
- **Modelo Vectorial:** Un vector se define como una cantidad con una coordenada de inicio y una dirección y desplazamiento asociados. En el modelo Vectorial los objetos son representados por los puntos y líneas que definen sus límites. La posición de cada objeto se define por su ubicación en un mapa espacial que se organiza por un sistema de referencia coordinado. Los puntos son almacenados como líneas de longitud cero, las áreas o polígonos constituyen líneas con puntos comunes de inicio y final. Los modelos vectoriales más comunes son: la estructura de polígono completa y el modelo topológico.

Componentes de los SIG

Un SIG puede ser analizado desde el punto de vista de subsistemas, cada uno de ellos encargado de una serie de funciones particulares. Es habitual citar tres subsistemas fundamentales: (Olaya, 2010)

- **Módulo de datos:** Se encarga de las operaciones de entrada y salida de datos, y la gestión de estos dentro del SIG. Permite a los demás subsistemas tener acceso a los datos y realizar sus funciones en base a ellos.
- **Módulo de visualización y creación cartográfica:** Crea representaciones a partir de los datos (mapas, leyendas, etc.), permitiendo así la interacción con ellos. Entre otras, incorpora también las funcionalidades de edición.
- **Módulo de análisis:** Contiene métodos y procesos para el análisis de los datos geográficos.

Composición de los SIG

Los SIG descomponen la realidad en distintos temas o capas de información, cada uno de estos está compuesto por elementos geográficos a los que se les asocia una tabla de atributos donde se reflejan sus características o atributos. Una de las grandes ventajas de los SIG es que permiten relacionar y combinar distintas capas entre sí para determinar dónde se cumplen determinadas condiciones.

1.3.2 Descripción actual del dominio del problema

El sistema de distribución es de suma importancia para una empresa eléctrica por dos razones: su proximidad con el cliente final y su alto costo de inversión. El objetivo de la planificación de la distribución del sistema es asegurar que la creciente demanda de electricidad, con las tasas de crecimiento y la alta densidad de carga, se pueda satisfacer de manera óptima, sobre todo para lograr el mínimo del costo total de la expansión del sistema de distribución. El servicio eléctrico es responsabilidad de la compañía suministradora (distribuidora) que ha de construir y mantener las líneas necesarias para llegar a los clientes. El diseño de las redes de distribución eléctrica es una tarea cotidiana para los ingenieros de servicios públicos de electricidad, especialmente los del departamento de diseño. Estos diseños se realizan de forma computarizada utilizando mecanismos poco eficientes para una situación tan compleja. Este enfoque, por lo general, resulta en un sistema de distribución poco práctico para su actualización y mantenimiento, así como para la toma de decisiones.

En la UNE el proceso de gestión y control de la distribución de la energía eléctrica se efectúa de forma manual y poco convencional, utilizando herramientas informáticas para los procesos de diseño y planificación, que no son óptimas para este tipo de trabajo y que, a pesar de lograr su objetivo, lo hace de forma parcial y con mayor esfuerzo, pues no están diseñadas para resolver problemas con esta complejidad.

En la UNE la gestión de la distribución eléctrica se realiza en diferentes departamentos. La información es recopilada y analizada por el departamento de despacho donde se registran los cambios ocurridos en el día, que pueden ser accidentes, roturas imprevistas, cambios de voltajes, entre otras, mediante avisos o llamadas telefónicas. Esta información es almacenada en un registro de formato duro donde los despachadores toman decisiones y envían la orden al terreno guiándose por los mapas que se encuentran en formato AutoCAD y los registros en formato duro, donde la posibilidad de introducir errores es elevada, ya que la sincronización de la información de los registros con el mapa resulta compleja.

El departamento de despacho luego de realizar el análisis del sistema diario al finalizar el día lo entrega al departamento de diseño, donde se encuentran los dibujantes esperando por la información con los nuevos cambios en formato duro y el mapa. Los dibujantes con la ayuda de la herramienta informática de diseño AutoCAD representan los nuevos cambios en el mapa de forma estática, actualizando la cartografía digital para ser utilizada e impresa nuevamente. Diariamente los dibujantes

modifican la cartografía pudiendo ocasionar cambios innecesarios introduciendo errores que a simple vista no son perceptibles.

Al día siguiente el departamento de diseño entrega el mapa con los nuevos cambios incorporados al departamento de despacho para su correspondiente estudio y análisis. Este proceso se repite diariamente tras las solicitudes del departamento de despacho lo que lleva consigo gran esfuerzo y tiempo por parte del personal de ambos departamentos, llegando a ser un proceso complejo y tedioso, pasando por varios departamentos antes de ser utilizado.

1.3.3 Situación Problemática

La creación, actualización, mantenimiento y gestión general de la red de distribución eléctrica en términos de datos espaciales y no espaciales es una tarea compleja que se realiza con el apoyo de herramientas informáticas que no están diseñadas para darle solución a este tipo de problemas; lo que pudiera introducir errores y resultados de análisis incorrectos para la toma de decisiones. La gran cantidad de datos involucrados para mantener un registro adecuado resulta incómoda, y no pueden ser manejados efectivamente por un sistema tradicional de gestión de registros, además estos sistemas no cuentan con funcionalidades para representar los cambios en tiempo real. Todos estos problemas traen consigo un aumento en la probabilidad de errores en el análisis, en la toma de decisiones y en el diseño y creación de los mapas. Un mayor tiempo para realizar las operaciones de administración y gestión de las redes de distribución eléctrica. Realizar un esfuerzo extra por el personal del departamento de despacho y de diseño, haciendo un mayor uso de su experiencia, la utilización extra de recursos ofimático para el almacenamiento y gestión de la información.

Es por ello que se hace necesaria la existencia de un Sistema de Información Geográfica para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica y la elaboración de su respectiva documentación técnica para aumentar la eficiencia y mejorar la operación y administración de la red. Esto permitirá mayor calidad y rapidez en el proceso de gestión de las redes de distribución eléctrica y eliminar errores que pueden introducirse de forma manual, perfeccionando los resultados de análisis para la toma de decisiones y mejorar el proceso de diseño y actualización de los mapas.

1.4 Análisis de otras soluciones existentes

Actualmente existen disímiles herramientas que, de un modo u otro, tratan la problemática de la distribución de las redes eléctricas, aunque vale destacar que la mayoría son propietarias y requieren

de la compra de licencias para su utilización, mantenimiento y actualización, lo cual no permite utilizar su código para su posterior estudio y desarrollo, contrario a todos los principios que actualmente se siguen en Cuba para la informatización del país. Estas herramientas se centran en tareas específicas de análisis, diseño y planificación de los Sistemas de Distribución, pero no integran estos elementos con un Sistema de Información Geográfica capaces de gestionar y visualizar toda la información en tiempo real.

Programas de Análisis de Distribución de Energía Eléctrica (PADEE): Software para el análisis, diseño y planificación de las redes de distribución de energía eléctrica en media y baja tensión basado en CAD-CAE-GIS. (Matheus, y otros, 2008)

El Software toma los datos de las redes de distribución directamente de los planos en AutoCAD y los analiza (Software de flujo de carga), colocando los resultados directamente sobre el plano en forma automática e interactiva. Muy útil para el diseño, planificación, operación, mantenimiento y reducción de pérdidas de las Redes Eléctricas y es utilizado por empresas distribuidoras de electricidad o consultoras que trabajan en distribución.

Programa de análisis de redes eléctricas (CYME): El programa de análisis de redes eléctricas (CYME) es un conjunto de aplicaciones que constan de un editor de red, módulos de análisis y de bibliotecas de modelos personalizables mediante los cuales el usuario puede obtener la solución más eficiente.

Los módulos disponibles incluyen varias aplicaciones avanzadas y extensas bibliotecas para el análisis de redes industriales, de transporte o de distribución de energía eléctrica. (T&D, 2004)

Smallworld Electric Office: Es una plataforma integral que provee soporte geoespacial para el ciclo de vida de la gestión de activos de las empresas de redes, desde la planificación y el diseño hasta la mantención y el retiro de los elementos de la red.

Contiene modelos de datos, aplicaciones orientadas a la industria y herramientas de productividad que permiten realizar las actividades esenciales para la gestión de las redes eléctricas, incluyendo la ingeniería y el diseño de redes eléctricas, el análisis y la optimización de redes, la operación, la gestión de activos y la distribución de la información a través de aplicaciones Web.

Sistema Integral de Gestión de Redes (SIGERE): Sistema realizado en Cuba que forma parte del Sistema de Gestión de la Unión Nacional Eléctrica (SIGE) y está integrado por todos los equipos,

instalaciones, infraestructura y acciones que forman parte de la red de Transmisión y Distribución. El sistema debe recoger datos técnicos, económicos y de gestión que faciliten la dirección, operación, explotación y planificación de las redes. La aplicación está orientada al cliente permitiendo la reducción de costos operativos y mejorar la calidad de suministro. (Castro López, y otros, 2003)

1.5 Conclusiones Parciales

Se analizaron hasta el momento los principales conceptos y definiciones que circundan a los SIG y los Sistemas de Distribución Eléctrica, analizado la situación actual del dominio del problema, fundamentando el marco teórico metodológico, lo que garantiza continuar con el proceso investigativo.

Después de un análisis de las diferentes herramientas semejantes a nivel mundial, se concluye que actualmente no existe ningún software libre de escritorio que brinde una solución adecuada para erradicar los problemas de gestión de las redes de distribución existentes en la UNE, de visualización y manipulación de la información en tiempo real. Aunque existen herramientas que se encargan de dar solución en cierto modo a la problemática planteada, son en su mayoría propietarias y requieren de la compra de licencias para su utilización, o no cumplen con todas las especificaciones para un producto completo. Se concluye que se necesita un software que represente, en tiempo real, los cambios en la distribución de las redes eléctricas sobre el mapa, que evite en gran medida la introducción y manipulación de la información por parte de las personas para evitar introducir errores y resultados de análisis incorrectos, para la toma de decisiones y que se desarrolle sobre tecnologías libres, que contribuya al principio de la soberanía tecnológica, para su futuro estudio y desarrollo sin restricciones.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A UTILIZAR

2.1 Introducción

En este capítulo se realiza un análisis de los métodos, técnicas, herramientas, metodologías de desarrollo de software, lenguaje de modelado, lenguajes de programación y gestores de bases de datos, para el modelado del subsistema que se empleará en el desarrollo y documentación de la aplicación que permita optimizar y ganar en tiempo, en el desarrollo del producto.

2.2 GeoQ

GeoQ es un proyecto del centro de desarrollo de software GEYSED de la Universidad de las Ciencias Informáticas. Es un sistema de Información Geográfica (SIG) basado en el proyecto Quantum GIS (QGIS), es fácil de usar, de código abierto y bajo la licencia pública general (GNU/GPL). Es una solución desarrollada en C++ y de carácter multiplataforma (Linux, Unix, Mac OSX y Windows) que trabaja con distintos orígenes de datos tales como PostGIS y SpatiaLite, la mayoría de los formatos Vectoriales y formatos Ráster. GeoQ proporciona un número cada vez mayor de las capacidades proporcionadas por las funciones básicas y plugins. Permite visualizar, gestionar, editar, analizar datos y componer mapas para imprimir.

El proyecto GeoQ permite el desarrollo de nuevas soluciones que se ajusten a las necesidades que exija cualquier negocio que pueda existir mediante personalizaciones, en este caso, para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica.

Para darle solución a la problemática se utilizará la versión de GeoQ: **1.0**

2.3 Arquitectura de Software

David Garlan establece que “la Arquitectura de Software (AS) constituye un puente entre el requerimiento y el código, ocupando el lugar que en los gráficos antiguos se reservaba para el diseño”. (Reynoso, 2004)

Numerosas son las definiciones que los expertos en el tema precisan para AS sin embargo esta investigación se registró por la que aparece redactada en el documento de IEEE Std 1471-2000:

Capítulo 2. Tendencias y tecnologías actuales a utilizar

La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución.

Entre las tendencias actuales de los Patrones/Estilos arquitectónicos se encuentran:

- Estilos de Flujo de Datos.
 - ✓ Tubería y filtros.
- Estilos Centrados en Datos.
 - ✓ Arquitectura de Pizarra o Repositorio.
- Estilos de Llamada y Retorno.
 - ✓ Modelo Vista Controlador.
 - ✓ **Arquitectura en Capas.**
 - ✓ **Arquitectura Orientada a Objetos.**
 - ✓ Arquitectura Orientada a Componentes.
- Estilos de Código Móvil.
 - Arquitectura de Máquinas Virtuales.
- Estilos Heterogéneos.
 - ✓ Sistema de Control de Procesos.
 - ✓ Arquitectura Basada en Atributos.
- Estilos Peer-to-Peer.
 - ✓ Arquitectura Basada en Eventos.
 - ✓ Arquitectura Orientada a Servicios.
 - ✓ Arquitectura Basada en Recursos.

Dentro de este grupo de Patrones/Estilos arquitectónicos, se seleccionó la Arquitectura Orientado a Objetos y la Arquitectura en Capas. Para esta selección se tuvo en cuenta la solución que brindan ambos patrones y que la plataforma GeoQ con la que se pretende desarrollar el sistema propuesto tiene implementados estos patrones, por lo que su utilización facilitará continuar con su desarrollo y mantendrá la misma arquitectura de software.

2.3.1 Arquitectura Orientada a Objetos

Los componentes de este estilo son los objetos, o más bien instancias de los tipos de datos abstractos; se basan en principios OO: encapsulamiento, herencia y polimorfismo. Los objetos son asimismo las

unidades de modelado, diseño e implementación, y sus interacciones son el centro de las incumbencias en el diseño de la arquitectura y en la estructura de la aplicación. Las interfaces están separadas de las implementaciones. En general la distribución de objetos es transparente. (Reynoso, 2004)

Características

- Diseños modulares.
- Efectos laterales mínimos (encapsulamiento)
- Extensibilidad.
- Fácil de modificar.
- Orientado a datos.
- Explota la herencia (jerárquico).
- Reutilización de clases.

2.3.2 Arquitectura en Capas

Los sistemas o arquitecturas en capas constituyen una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior.

Este estilo se basa en tres capas principales:

- **Capa de Presentación:** Esta capa se comunica únicamente con la capa de negocio llevando y trayendo los datos o registros necesarios, es la interfaz gráfica del programa y debe ser lo más amena posible para una mejor comunicación con el usuario.
- **Capa de Negocio:** Constituye la capa donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todos los procesos que deben realizarse.
- **Capa de Acceso a Datos:** Constituye la capa donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Características

- Centralización de los aspectos de seguridad, que serían responsabilidad del modelo.
- No replicación de lógica de negocio en los clientes: esto permite que las modificaciones y mejoras sean automáticamente aprovechadas por el conjunto de los usuarios, reduciendo los costes de mantenimiento.
- Mayor sencillez de los clientes.

2.4 Metodología de desarrollo de software

Las metodologías de desarrollo de software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto software. (Carrillo Pérez, y otros, 2008)

Su principal objetivo es guiar a los desarrolladores en la creación de nuevas aplicaciones de probada calidad.

Metodologías pesadas: Son aquellas orientadas al control de los procesos, estableciendo rigurosamente las actividades a desarrollar, herramientas a utilizar y notaciones que se usarán.

Metodologías ágiles: Están más orientadas a la interacción con el cliente y el desarrollo incremental del software, mostrando versiones parcialmente funcionales del software al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando.

2.4.1 Proceso Unificado Racional (RUP)

Dentro de las metodologías pesadas sobresale el Proceso Unificado Racional (Rational Unified Process en inglés, usualmente abreviado como RUP) considerada entre las más tradicionales, centrada en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requiere una extensa documentación, ya que pretende prever todo de antemano.

La metodología RUP está compuesta por 4 fases fundamentales: (Tobarra Narro, 2003)

- **Concepción:** Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **Elaboración:** Se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **Construcción:** Se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.

- **Transición:** Se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

RUP es un proceso para el desarrollo de un proyecto de software que define claramente quién, cómo, cuándo y qué debe hacerse en el proyecto.

La metodología RUP cuenta con 3 características esenciales: (Tobarra Narro, 2003)

- **Dirigido por casos de uso:** Los casos de uso constituyen la guía fundamental establecida para las actividades a realizar durante todo el proceso de desarrollo del software.
- **Centrado en la arquitectura:** Se establece una arquitectura candidata al inicio del desarrollo del sistema que funcionará como guía y se irá perfeccionando en las restantes fases de desarrollo.
- **Iterativo e incremental:** El desarrollo iterativo garantiza la corrección de los errores en cada iteración, brindando la posibilidad de que los elementos sean integrados continuamente, lo que garantiza un producto más robusto y de mayor calidad.

RUP mejora la productividad del equipo de trabajo y entrega las mejores prácticas del software a todos los miembros del mismo. Es, además, una metodología de desarrollo de software que intenta integrar todos los aspectos a tener en cuenta durante el ciclo de vida del software, con el objetivo de abarcar tanto pequeños como grandes proyectos. (Tobarra Narro, 2003)

2.5 Lenguaje de Modelación

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. (Pockin, 2008)

2.5.1 Lenguaje Unificado de Modelado (UML)

UML (del inglés Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software. Sus creadores pretendieron, con este lenguaje,

unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas en un acercamiento estándar. (Pockin, 2008)

UML ayuda a los usuarios a entender la realidad desde un punto de vista de la tecnología y la posibilidad de que reflexione antes de invertir y gastar grandes cantidades de dinero en proyectos que no estén seguros en su desarrollo, reduciendo el costo y el tiempo empleado en la construcción de los módulos que construirán el software. (Rumbaugh, y otros, 2000)

Entre los rasgos principales que han contribuido a hacer de UML el estándar de la industria en la actualidad, se pueden mencionar: (Rumbaugh, y otros, 2000)

- Permite modelar sistemas utilizando técnicas orientadas a objetos.
- Adecuado a las necesidades de conectividades actuales y futuras.
- Es un lenguaje muy expresivo que cubre las vistas necesarias para desarrollar y luego desplegar los sistemas.
- Ampliamente utilizado por la industria del software.
- Reemplaza a decenas de notaciones empleadas por otros lenguajes.
- Modela estructuras complejas.
- Comportamiento del sistema: casos de usos, diagramas de secuencia, de colaboración, que sirven para evaluar el estado de las máquinas.

UML constituye una de las innovaciones conceptuales más importantes en el mundo tecnológico del desarrollo de software que más expectativas ha generado a lo largo de muchos años, comparable con la aparición e implantación de los lenguajes COBOL, Basic, Pascal, C++, y actualmente con los más recientes Java, XML, C#.

Para darle solución a la problemática se utilizará la versión de UML: **2.1**

2.6 Herramienta CASE

Las Herramientas CASE (Ingeniería de Software Asistida por Ordenador) en inglés (Software Engineering Computer Aided) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

2.6.1 Visual Paradigm

Visual Paradigm una herramienta CASE que utiliza UML como lenguaje de modelado. Está diseñada para una amplia gama de usuarios interesados en construir sistemas fiables con el uso del paradigma orientado a objetos, incluyendo actividades como ingeniería de software, análisis de sistemas y análisis de negocios. (Rumbaugh, y otros, 2000)

Sus características principales:

- Entorno de creación de diagramas para UML.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa (versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDEs.
- Disponibilidad en múltiples plataformas.

El Visual Paradigm, es una herramienta desarrollada con tecnologías libres, factor fundamental que garantiza la soberanía e independencia tecnológica, multiplataforma, posibilita la creación de diagramas para UML; diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad; capacidades de ingeniería directa e inversa; modelo y código que permanece sincronizado en todo el ciclo de desarrollo; disponibilidad de múltiples versiones y posibilita integrarse en los principales Entornos de Desarrollo.

Para darle solución a la problemática se utilizará la versión de Visual Paradigm: **8.0**

2.7 Lenguaje de Programación

Un lenguaje de programación es una técnica de comunicación estandarizada para expresar las instrucciones a una computadora. Se trata de un conjunto de reglas sintácticas y semánticas utilizadas para definir los programas de ordenador. (Cueva Lovelle, 1998)

Un lenguaje de programación de alto nivel se caracteriza por expresar los algoritmos necesarios para la creación de los programas informáticos de una manera adecuada a la capacidad cognitiva humana en lugar de a la capacidad ejecutora de las máquinas, es decir, facilita la comunicación con un

computador mediante signos convencionales cercanos a los de un lenguaje natural. (Cueva Lovelle, 1998)

2.7.1 C++

El C++ es un lenguaje versátil, potente y general. Su éxito entre los programadores profesionales le ha llevado a ocupar uno de los primeros puestos como herramienta de desarrollo de aplicaciones de escritorio. El C++ mantiene las ventajas del C en cuanto a riqueza de operadores y expresiones, flexibilidad, concisión y eficiencia. Además, ha eliminado algunas de las dificultades y limitaciones del C original. (García de Jalón, y otros, 1998)

El C++ es un lenguaje imperativo, orientado a objetos, que exige del programador un completo cambio de mentalidad. Mantiene una considerable potencia para programación a bajo nivel aunque se le han añadido elementos que le permiten también un estilo de programación con alto nivel de abstracción. Se le han incorporado nuevos tipos de datos, clases, plantillas, mecanismo de excepciones, sistema de espacios de nombres, funciones de expansión en línea, sobrecarga de operadores, referencias, operadores para manejo de memoria persistente, y algunas utilidades adicionales de librería.

2.8 Entorno de Desarrollo Integrado (IDE)

Un Entorno Integrado de Desarrollo (IDE) es un entorno de programación o programa informático compuesto por un conjunto de herramientas que utilizan los programadores para generar código. Las herramientas que oficialmente componen un IDE son: un editor de código, un compilador, un depurador, un constructor de interfaz gráfica y, eventualmente, un sistema de control de versiones. Tiene como objetivo acortar la brecha entre el usuario y el lenguaje de programación proporcionando un marco de trabajo amigable. Un IDE puede estar realizado para soportar un sólo lenguaje de programación o varios.

2.8.1 Qt Creator

Qt Creator constituye una multiplataforma que se ajusta a las necesidades de los desarrolladores. Se centra en proporcionar características que ayudan a los nuevos usuarios de Qt a aprender y comenzar a desarrollar rápidamente, también aumenta la productividad de los desarrolladores con experiencia en Qt. (Chaves Pérez, 2010)

Es un excelente IDE multiplataforma para el desarrollo de aplicaciones en C++ de manera sencilla y rápida. Entre sus principales características se encuentra: (Nicanor, 2007)

- Utiliza el lenguaje de programación orientado a objetos C++.
- Soporta los lenguajes: C#/.NET Lenguajes (Mono), Python: PyQt y PySide, Ada, Pascal, Perl, PHP y Ruby.
- Se basa en Qt, una librería multiplataforma y gratuita para la creación de interfaces gráfica, programación web, multihilo, bases de datos, etc.
- Permite realizar programación visual y programación dirigida por eventos.
- Características avanzadas de IDE: sintaxis coloreada, compleción automática de código, ayuda sensible al contexto, inspector de objetos, diseñador visual, compilador y depurador integrado, etc.
- Completamente orientado a objetos.

Qt Creator permite el desarrollo de aplicaciones en entornos MS Windows, Mac OS y Linux. Se caracteriza por ser abierto, gratuito y muy eficiente.

Para darle solución a la problemática se utilizará la versión de Qt Creator: **2.3.1**

2.9 Sistemas Gestores de Base de Datos

2.9.1 PostgreSQL

PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre. Es un Sistema Gestor de Base de Datos que ha sido desarrollado desde 1977. Comenzó como un proyecto en la Universidad Berkeley de California. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. Está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre. (González, 2011)

Entre sus principales características destacan: (González, 2011)

- **Soporte SQL Comprensivo:** PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92.
- **DBMS Objeto-Relacional:** Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas.

- **Integridad Referencial:** PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- **Altamente extensible:** Soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.
- **Tipos de Datos:** Además del soporte para los tipos base y estructura de datos de arreglos, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, entre otros. También permite la creación de tipos propios.
- **Lenguajes Procedurales:** Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

Muchos son los motores de datos existentes en el mercado, PostgreSQL se destaca por sus capacidades desde hace un tiempo y cada vez toma mayor fuerza. OpenGIS es soportado por este motor de datos por medio de PostGIS, que es una herramienta que le permite manipular geometrías, indispensables para los SIG.

2.9.2 PostGIS

Esta herramienta se incluye al servidor de datos PostgreSQL, permite manejar nuevos tipos de datos que no son parte del motor. Existen versiones de PostgreSQL para Linux y Windows, al igual existen versiones de PostGIS para ambos sistemas operativos.

PostGIS es un módulo que añade soporte de objetos geográficos a la base de datos objeto-relacional PostgreSQL, convirtiéndola en una base de datos espacial para su utilización en Sistema de Información Geográfica. Se publica bajo la Licencia pública general de GNU.

Ha sido desarrollado por la empresa canadiense Refraction Research, especializada en productos de código abierto. PostGIS es hoy en día un producto veterano que ha demostrado versión a versión su eficiencia. En relación con otros productos, ha demostrado ser muy superior a la extensión geográfica de la nueva versión de MySQL, y a juicio de muchos autores, es muy similar a la versión geográfica de la archiconocida Oracle. (Iglesia, 2011)

Un aspecto a tener en cuenta es que PostGIS ha sido certificado en el 2006 por el Open Geoespacial Consortium (OGC) lo que garantiza la interoperabilidad con otros sistemas también interoperables. PostGIS almacena la información geográfica en una columna del tipo GEOMETRY, que es diferente

del homónimo "GEOMETRY" utilizado por PostgreSQL, donde se pueden almacenar la geometría en formato **WKB** en inglés (Well-Known Binary), aunque hasta la versión 1.0 se utilizaba la forma **WKT** en inglés (Well-Known Text). (Iglesia, 2011)

Para darle solución a la problemática se utilizará la versión de PostgreSQL: **9.1** y la versión de PostGIS: **1.5**

2.10 Conclusiones Parciales

Para el desarrollo del sistema propuesta se utilizará como plataforma base GeoQ por su carácter de código abierto que permitirá nuevas soluciones que se ajusten a las necesidades que exija cualquier negocio que pueda existir. Esta herramienta permitirá la reutilización de código y una solución más rápida debido a las funcionalidades ya existentes en la aplicación base.

Para el desarrollo de esta personalización se utilizará el lenguaje de programación C++, como IDE de desarrollo a Qt Creator y siguiendo una arquitectura de software orientada a objetos. La metodología a usar es RUP, UML como lenguaje de modelado donde los diagramas serán diseñados utilizando Visual Paradigm de CASE. La selección de estas herramientas y tecnologías se llevo a cabo teniendo en cuenta que todas cumplen con los principios de soberanía tecnológica de la universidad, y que impulsa el país, además que el sistema propuesto tiene como base GeoQ que es una herramienta desarrollada en su mayoría utilizando estas herramientas y tecnologías, por lo que su utilización facilita la continuación de su desarrollo garantizando en gran medida una guía durante todo el proceso de desarrollo y la obtención de la documentación técnica asociada al sistema.

CAPÍTULO 3: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

3.1 Introducción

En este capítulo se describe el proceso llevado a cabo durante la fase de análisis y diseño. Se analiza la solución en términos de procesos de negocio, el actor y trabajadores del negocio unido a la justificación de su uso, además, se expone el Diagrama de Casos de Uso del Negocio con su descripción textual.

En un segundo momento, se presenta la solución desde la perspectiva del sistema, se define el actor del sistema, requisitos funcionales y no funcionales del software. A partir de los requisitos funcionales se exhibe el Diagrama de Casos de Uso del Sistema (DCUS) y luego sus descripciones textuales. Como parte del proceso que se realiza durante esta fase se generan los diferentes diagramas y artefactos correspondientes, para dar cumplimiento a los requisitos trazados.

3.2 Modelo de Negocio

El modelo de negocio describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización. Tiene como objetivo comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema, comprende los problemas actuales de la organización e identifica las mejoras potenciales. Asegura que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización y sepan derivar los requisitos del sistema.

3.2.1 Actor y trabajadores del negocio

Un actor del negocio representa un individuo, grupo, organización, máquina o sistema que interactúa con el negocio y se beneficia de su existencia.

Actor del Negocio	Descripción
 Cliente	Recibe el servicio eléctrico en su domicilio y en caso de una falla en el sistema eléctrico lo reporta a la empresa eléctrica.

Tabla 1: Actor del Negocio.

Capítulo 3. Presentación de la solución propuesta

Los trabajadores del negocio representan personas o sistemas que están involucrados en uno o más procesos del negocio, que participan en ellos, pero no obtienen ningún resultado de valor.

Trabajadores del Negocio	Descripción
 Despachador	Se encarga de recibir todos los reportes de incidencias de la población y procesarlos, atiende a los operarios en el terreno haciendo un seguimiento del estado del trabajo en las líneas eléctricas.
 Operario	Se encarga de darle solución a las fallas eléctricas en el sistema, se desplazan en el terreno y son supervisados desde la empresa eléctrica.
 Dibujante	Se encarga de diseñar, actualizar y mantener el estado de las redes de distribución eléctrica en los mapas o planos, usando herramientas informáticas.

Tabla 2: Trabajadores del Negocio.

La siguiente figura representa el **Modelo de Objetos del Negocio**.

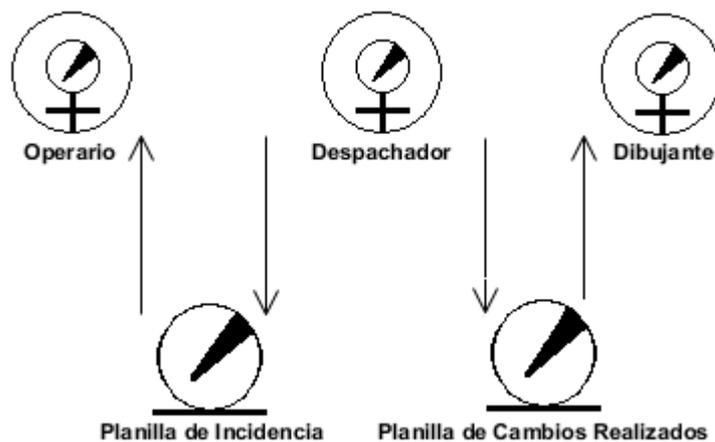


Figura 1: Modelo de Objetos del Negocio.

3.2.2 Restricciones del negocio

El proceso de negocio consiste en darle solución a las incidencias causadas en el sistema eléctrico que afectan a los clientes. Los despachadores son los encargados de velar por el buen funcionamiento de las redes eléctricas, atender las llamadas de los clientes y supervisar a los operarios en el terreno, en sus labores de reparación de las redes eléctricas. Los dibujantes son los encargados de mantener y actualizar los mapas de las redes de distribución eléctrica, llevando a estos los cambios ocurridos durante la jornada de trabajo.

Los despachadores para asistir a un operario deben apoyarse en todas las modificaciones que ha sufrido la red durante el día, los planos impresos de la red y su representación en el mapa. Los dibujantes son los encargados de diseñar y construir las redes de distribución eléctrica y de gestionar todos los objetos eléctricos en el mapa, llevando un control detallado de cada cambio. En caso de expansión de la red o modificaciones en el sistema de distribución son los encargados de la modificación de los mapas. Los dibujantes al finalizar cada jornada de trabajo y antes de comenzar la próxima jornada, tienen que reflejar todos los cambios ocurridos en el sistema eléctrico en formato digital (actualizar los mapas) y luego realizar su impresión para ser utilizado en el área de despacho.

3.2.3 Diagrama de caso de uso del Negocio

La siguiente figura representa el diagrama de **Casos de Uso del Negocio** que describe la relación que existe entre el caso de uso del negocio y el actor del negocio.



Figura 2: Diagrama de Caso de Uso del Negocio.

3.2.4 Diagrama de Actividades

Los diagramas de actividades ayudan a describir en detalle lo que ocurre dentro del negocio, permiten examinar los roles específicos que juegan las personas (trabajadores del negocio) y las actividades que realizan, también identifican qué funciones deberá asumir el producto y quiénes serán los actores del futuro sistema.

Capítulo 3. Presentación de la solución propuesta

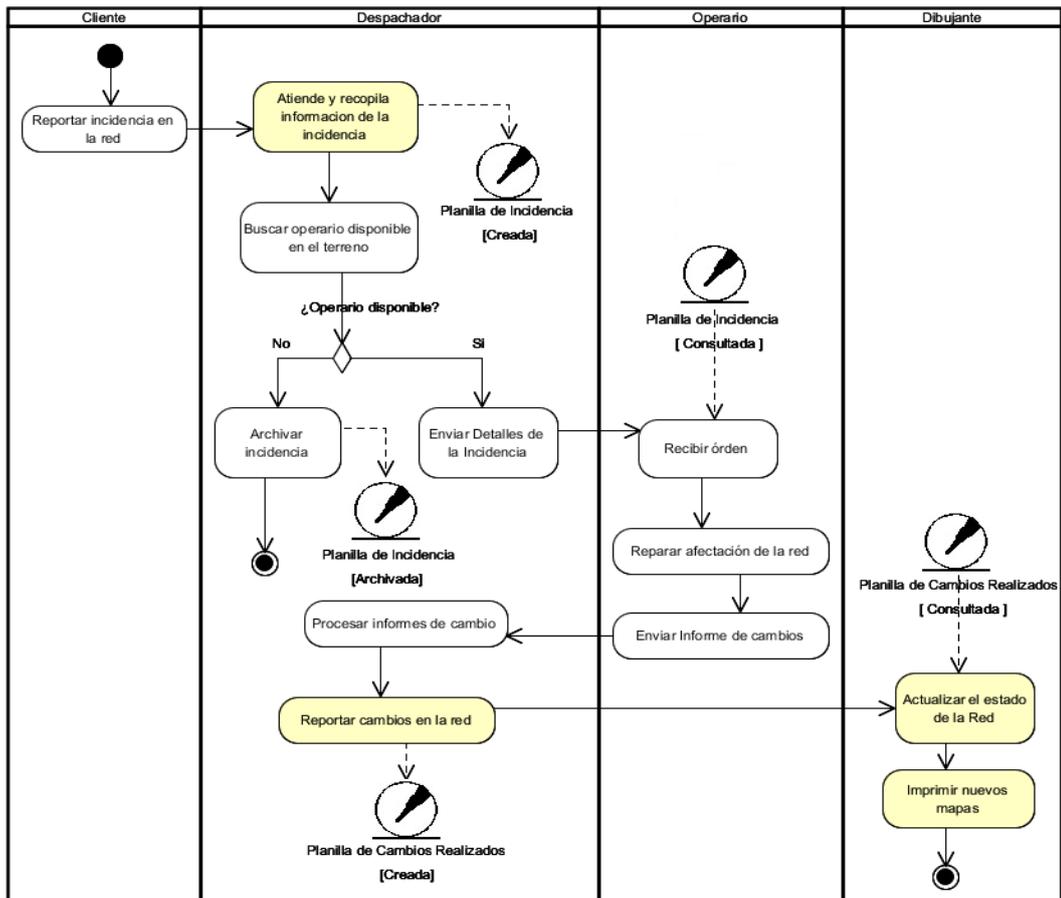


Figura 3: Diagrama de Actividades.

3.2.5 Descripción textual de los casos de uso del Negocio

A continuación se muestra la descripción textual del Caso de Uso del Negocio: **“Reportar Incidencia”**.

Caso de uso del negocio	Reportar incidencia
Actor	Cliente.
Trabajadores	Despachador, Operario, Dibujante.
Resumen	El CU inicia cuando el cliente reporta un incidente, en el sistema eléctrico de la red, y concluye cuando se actualiza el estado de la red eléctrica luego de darle solución.
Casos de uso asociados	Ninguno.
Curso Normal de Eventos	

Capítulo 3. Presentación de la solución propuesta

Acción del actor	Respuesta del proceso de negocio
1. Reporta incidente en el sistema eléctrico de la red.	2. El despachador atiende al cliente, recopilando y documentando los datos del incidente.
	3. El despachador localiza a los operarios disponibles en el terreno.
	4. El despachador envía la orden de solución al terreno y los detalles de la incidencia.
	5. Los operarios reciben la orden y reparan la afectación de la red.
	6. Los operarios envían un informe con los cambios realizados en la red, a los despachadores.
	7. Los despachadores procesan el informe de los cambios ocurridos y reportan los cambios a los dibujantes.
	8. Los dibujantes actualizan el estado de la red eléctrica e imprimen los nuevos mapas, finalizando así el caso de uso.
Flujos Alternos	3. En caso de no haber operarios disponibles en ese momento se archiva la incidencia.
Poscondiciones: Se le da solución a la incidencia reportada por el cliente y se actualizan los mapas.	

Tabla 3: Descripción del Caso de Uso del Negocio Reportar Incidencia.

3.3 Requisitos Funcionales

Los requisitos funcionales (en lo adelante RF) son capacidades o condiciones que el sistema debe cumplir. Como requisitos funcionales de la solución propuesta se tienen:

RF 1. Ubicar objeto eléctrico: El sistema debe permitir organizar los objetos eléctricos en su capa correspondiente, activando y desactivando la capa de forma automática, sin que el usuario tenga que preocuparse en que capa está trabajando.

RF 2. Añadir poste: El sistema debe permitir añadir un poste en una capa de postes predefinida por el usuario.

Capítulo 3. Presentación de la solución propuesta

El sistema debe mostrar en caso de error un mensaje indicando que la capa postes no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 3. Añadir transformador: El sistema debe permitir añadir un transformador en una capa de transformadores predefinida por el usuario.

El sistema debe mostrar en caso de error un mensaje indicando que la capa transformadores no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 4. Añadir línea aérea: El sistema debe permitir añadir una línea aérea en una capa de líneas aéreas predefinida por el usuario.

El sistema debe mostrar en caso de error un mensaje indicando que la capa líneas aéreas no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 5. Añadir línea soterrada: El sistema debe permitir añadir una línea soterrada en una capa de líneas soterradas predefinida por el usuario.

El sistema debe mostrar en caso de error un mensaje indicando que la capa líneas soterradas no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 6. Añadir alumbrado público: El sistema debe permitir añadir un alumbrado público en una capa de alumbrado público predefinida por el usuario.

El sistema debe mostrar en caso de error un mensaje indicando que la capa alumbrado público no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 7. Añadir interruptor: El sistema debe permitir añadir un interruptor en una capa de interruptores predefinida por el usuario.

Capítulo 3. Presentación de la solución propuesta

El sistema debe mostrar en caso de error un mensaje indicando que la capa interruptores no existe en el servidor.

- Mensaje (Formato: Alfabético).

RF 8. Eliminar objeto eléctrico: El sistema debe permitir eliminar un objeto eléctrico (poste, transformador, línea aérea, línea soterrada, alumbrado público, interruptor) en dependencia de la capa en que se encuentren.

El sistema muestra un mensaje indicando el objeto que desea borrar.

- Mensaje (Formato: Alfanumérico).

RF 9. Colorear estado de un circuito: El sistema debe permitir cambiar el color de los circuitos en dependencia de si tienen energía eléctrica, si están conectados o desconectados.

RF 10. Abrir dos circuitos: El sistema debe permitir abrir dos circuitos en dependencia de si tienen energía eléctrica para desconectarlos.

RF 11. Cerrar dos circuitos: El sistema debe permitir cerrar dos circuitos en dependencia de si tienen energía eléctrica para conectarlos.

RF 12. Añadir fotos, documentos, multimedia: El sistema debe permitir añadir fotos, documentos y multimedia, y asociarlo a un determinado objeto eléctrico.

Esta funcionalidad requiere los siguientes criterios de entrada:

- Dirección del archivo (Formato: Alfanumérico, Obligatorio: Sí).

RF 13. Modificar fotos, documentos, multimedia: El sistema debe permitir modificar fotos, documentos y multimedia, de un determinado objeto eléctrico.

Esta funcionalidad requiere los siguientes criterios de entrada:

- Dirección del archivo (Formato: Alfanumérico, Obligatorio: Sí).

RF 14. Eliminar fotos, documentos, multimedia: El sistema debe permitir eliminar fotos, documentos y multimedia, de un determinado objeto eléctrico.

RF 15. Guardar el estado de la red en un momento dado: El sistema debe permitir guardar el estado de la red en un momento dado, quedando almacenado el mapa tal y como se encuentra en ese momento.

Esta funcionalidad requiere los siguientes criterios de entrada:

- Nombre (Formato: Alfanumérico, Obligatorio: Sí).

RF 16. Mostrar el estado de la red en un momento dado: El sistema debe permitir mostrar el estado de la red en un momento dado, buscando en los mapas almacenados anteriormente.

Esta funcionalidad requiere los siguientes criterios de entrada:

- Nombre (Formato: Alfanumérico, Obligatorio: Sí).

RF 17. Retornar al estado inicial: El sistema debe permitir retornar al estado inicial del sistema, permitiendo alternar entre diferentes mapas.

RF 18. Establecer un cambio anterior: El sistema debe permitir establecer un cambio anterior ocurrido en el sistema, permitiendo alternar entre diferentes mapas.

RF 19. Seleccionar componente de la red: El sistema debe permitir seleccionar componentes de la red por diferentes criterios de selección, que pueden ser postes, transformadores, circuitos, entre otros.

El sistema muestra un mensaje indicando la cantidad de objetos seleccionados.

- Mensaje (Formato: Alfanumérico).

RF 20. Buscar componente de la red: El sistema debe permitir buscar componentes de la red por diferentes criterios de búsquedas, que pueden ser postes, transformadores, circuitos, entre otros.

El sistema muestra un mensaje indicando la cantidad de objetos encontrados.

- Mensaje (Formato: Alfanumérico).

3.4 Requisitos No Funcionales

Los Requisitos No Funcionales (en lo adelante RNF) son propiedades o cualidades que el producto debe tener. Normalmente están vinculados a requisitos funcionales, es decir, una vez que se conozca

lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser. (Pressman, 2005)

3.4.1 Requerimientos de usabilidad

La interfaz gráfica de usuario debe permitir completar las interacciones necesarias en la menor cantidad de secuencias posibles. El sistema podrá ser usado por personas con conocimientos básicos en el manejo de computadoras, conocimientos mínimos para entender la simbología y conceptos que se trabajan en una red eléctrica, así como conocimientos mínimos en el trabajo con los Sistemas de Información Geográfica.

3.4.2 Interfaces de usuario

El sistema debe tener una apariencia profesional con un diseño gráfico sencillo con tonalidades de colores claros que facilite la localización de las funciones del sistema. La interfaz deberá permitir al usuario cambiar fácilmente la configuración del sistema. La interfaz debe ser sensible y visualmente atractiva.

3.4.3 Interfaces de software

PC Cliente:

Es un software multiplataforma que se ejecuta en Linux y Windows, debe ser compatible con los Sistemas Operativos Microsoft Windows 2000 NT, Microsoft Windows XP o superior y cualquier distribución de GNU/Linux.

PC Servidor:

Los servidores deben tener instalado PostgreSQL en su versión 9.1 y PostGIS en su versión 1.5.

3.4.4 Interfaces de hardware

PC Cliente:

Las computadoras que utilizarán la aplicación deberán contar con un microprocesador de velocidad de procesamiento de un 1 GHz o superior y memoria RAM de 512 MB o superior.

PC Servidor:

Capítulo 3. Presentación de la solución propuesta

Las computadoras que utilizarán la aplicación deberán contar con un microprocesador de velocidad de procesamiento de un 1 GHz o superior y memoria RAM de 512 MB o superior.

3.4.5 Interfaces de Comunicación

Es una aplicación de Escritorio. No se proveen interfaces de comunicación con sistemas externos.

3.5 Descripción del Sistema Propuesto

Un caso de uso constituye una técnica utilizada para describir el comportamiento del sistema, a través de un documento narrativo que define la secuencia de acciones que obtienen resultados de valor para un actor que utiliza un sistema para completar un proceso, sin importar los detalles de la implementación.

3.5.1 Descripción del Actor del Sistema

Actor del Sistema	Descripción
 <p>Dibujante</p>	El dibujante es el encargado de diseñar el sistema de distribución eléctrico utilizando herramientas informáticas. Actualiza y mantiene el estado de los mapas o planos.

Tabla 4: Descripción del actor del sistema.

3.5.2 Diagrama de casos de uso del Sistema

A continuación se muestra el Diagrama de **Casos de Uso del Sistema** donde está representado el actor que interactúa con el sistema y a continuación la descripción textual de cada uno de los casos de uso.

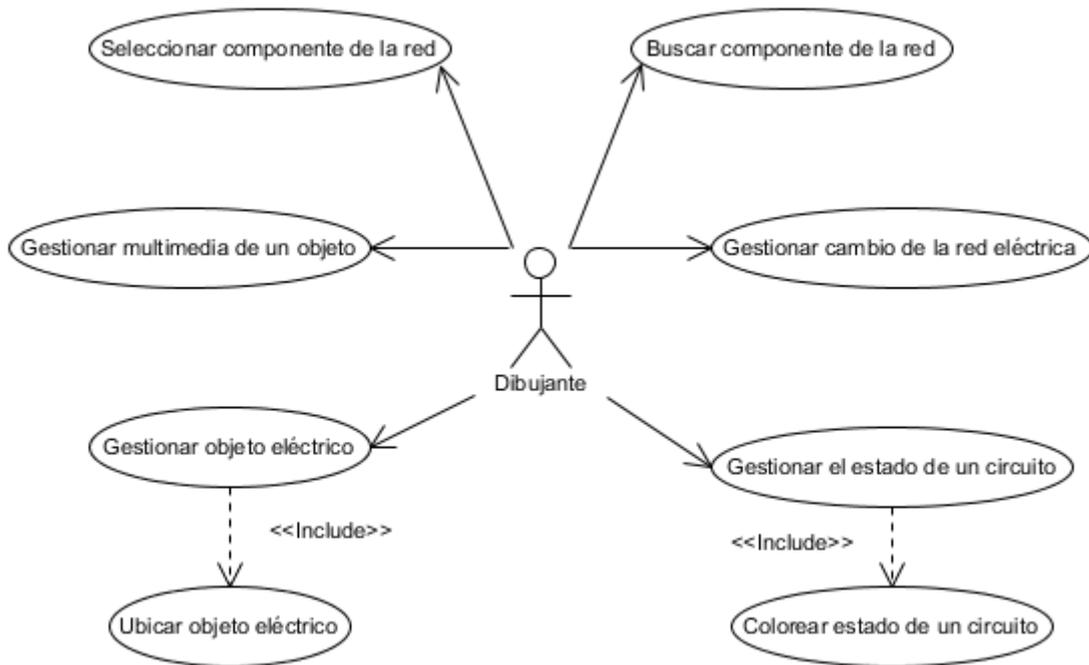


Figura 4: Diagrama de Casos de Uso del Sistema.

3.5.3 Descripción textual de los Casos de Uso del Sistema

A continuación se muestra la descripción textual del Caso de Uso del Sistema: “Ubicar objeto eléctrico”.

Caso de Uso	Ubicar objeto eléctrico.
Actor	Dibujante
Resumen	El caso de uso inicia cuando el dibujante necesita añadir algún objeto eléctrico en el mapa, sin tener en cuenta en que capa está trabajando, y termina cuando el dibujante añade un objeto eléctrico en el mapa y este se ubica en su capa correspondiente.
Precondiciones	Para lograr ubicar un objeto eléctrico en su capa correspondiente tiene que tener cargada la cartografía correspondiente a la red y añadirse algún objeto eléctrico.
Referencias	RF 1.
Prioridad	Secundario
Flujo Normal de Eventos	

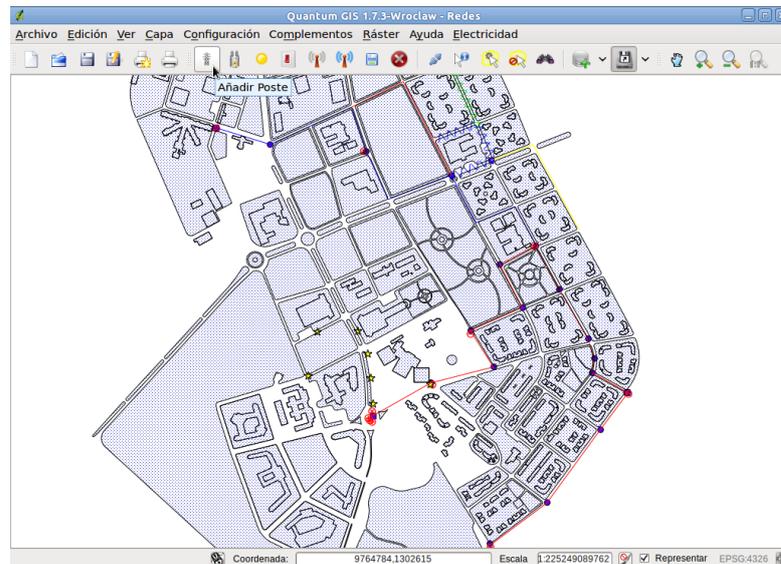
Capítulo 3. Presentación de la solución propuesta

“Ubicar objeto eléctrico”

Acción del Actor	Respuesta del Sistema
<p>1. El caso de uso inicia cuando el dibujante selecciona una de las siguientes opciones:</p> <ul style="list-style-type: none"> ✓ Añadir poste. ✓ Añadir transformador. ✓ Añadir alumbrado público. ✓ Añadir línea aérea. ✓ Añadir línea soterrada. ✓ Añadir interruptor. 	<p>2. El sistema selecciona la capa correspondiente a la opción seleccionada y activa el modo de edición. Ver prototipo de interfaz #1.</p>
<p>3. El dibujante añade el objeto eléctrico dando clic sobre el mapa.</p>	<p>4. El sistema ubica el objeto eléctrico en la capa que le pertenece.</p>

Prototipo de Interfaz

Interfaz #1



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>2. Si la capa relacionada con el objeto eléctrico no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #2.</p>

Capítulo 3. Presentación de la solución propuesta

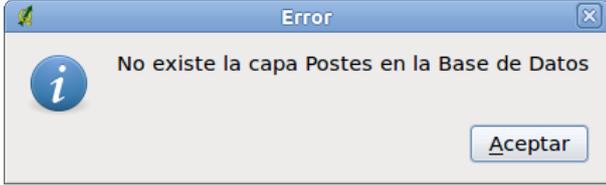
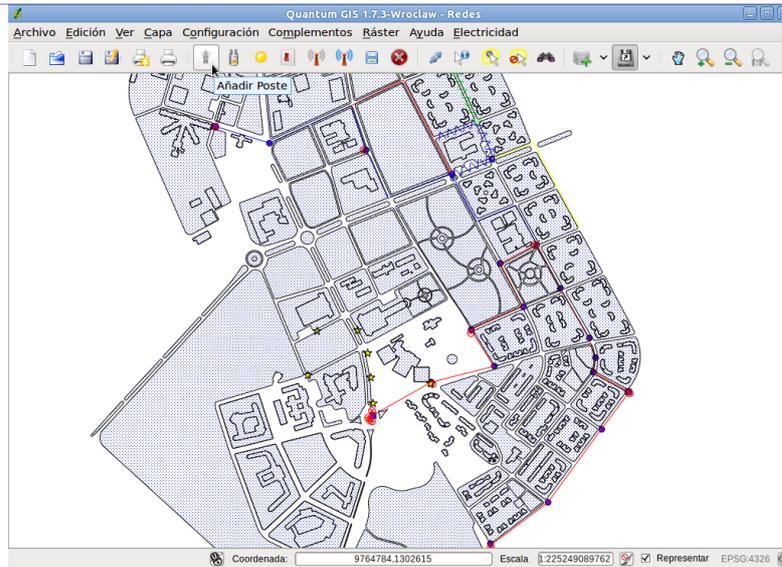
Prototipo de Interfaz	
Interfaz #2	
	
Poscondiciones	Se ubica el objeto eléctrico en su capa correspondiente.

Tabla 5: Descripción del caso de uso Ubicar objeto eléctrico.

Caso de Uso	Gestionar objeto eléctrico.
Actor	Dibujante
Resumen	El caso de uso inicia cuando el dibujante necesita gestionar los objetos eléctricos permitiendo añadir o eliminar “postes”, “transformadores”, “líneas aéreas”, “líneas soterradas”, “alumbrado público”, “interruptores”, en el mapa sobre la capa perteneciente al objeto seleccionado.
Precondiciones	Para lograr gestionar un objeto eléctrico se tiene que tener cargada la cartografía correspondiente a la red.
Referencias	RF 2, RF 3, RF 4, RF 5, RF 6, RF 7, RF 8.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando el dibujante selecciona una de las siguientes opciones: <ul style="list-style-type: none"> ✓ Añadir poste. ✓ Añadir transformador. ✓ Añadir línea aérea. ✓ Añadir línea soterrada. ✓ Añadir alumbrado público. ✓ Añadir interruptor. ✓ Eliminar objeto eléctrico. 	2. El sistema permite seleccionar las opciones de añadir o eliminar objetos eléctricos. <ul style="list-style-type: none"> - Si el dibujante selecciona “Añadir poste” ver sección “Añadir poste”. - Si el dibujante selecciona “Añadir transformador” ver sección “Añadir transformador”. - Si el dibujante selecciona “Añadir línea aérea” ver sección “Añadir línea aérea”. - Si el dibujante selecciona “Añadir línea soterrada” ver sección “Añadir línea

Capítulo 3. Presentación de la solución propuesta

	<p>soterrada”.</p> <ul style="list-style-type: none"> - Si el dibujante selecciona “Añadir alumbrado público” ver sección “Añadir alumbrado público”. - Si el dibujante selecciona “Añadir interruptor” ver sección “Añadir interruptor”. - Si el dibujante selecciona “Eliminar objeto eléctrico” ver sección “Eliminar objeto eléctrico”.
Sección “Añadir poste”	
Acción del Actor	Respuesta del Sistema
1. El dibujante selecciona la opción Añadir poste.	2. El sistema selecciona y activa el modo de edición en la capa “Postes”. Ver prototipo de interfaz #3.
3. El dibujante añade el elemento (Poste) en el área seleccionada por el sistema dando clic izquierdo sobre el mapa.	4. El sistema muestra un formulario para introducir los datos del elemento añadido.
5. El dibujante introduce los valores correspondientes a los atributos del poste.	6. El sistema guarda los atributos y concluye el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.
Prototipo de Interfaz	
Interfaz #3	



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #4.

Prototipo de Interfaz

Interfaz #4



Sección “Añadir transformador”

Acción del Actor	Respuesta del Sistema
1. El dibujante selecciona la opción Añadir transformador.	2. El sistema selecciona y activa el modo de edición en la capa “Transformadores”. Ver prototipo de interfaz #5.
3. El dibujante añade el elemento (Transformador) en el área seleccionada por el sistema dando clic izquierdo sobre el mapa.	4. El sistema muestra un formulario para introducir los datos del elemento añadido.
5. El dibujante introduce los datos	6. El sistema guarda los atributos y concluye

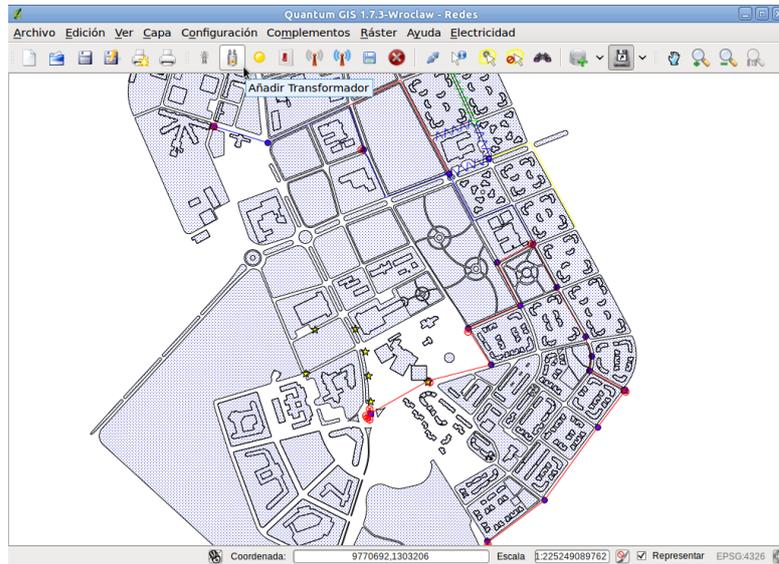
Capítulo 3. Presentación de la solución propuesta

correspondientes a los atributos del transformador.

el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.

Prototipo de Interfaz

Interfaz #5



Flujos Alternos

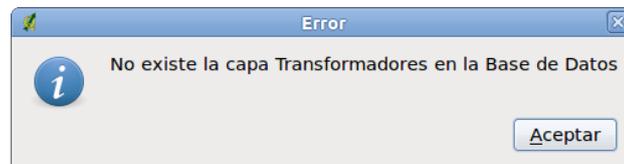
Acción del Actor

Respuesta del Sistema

2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #6.

Prototipo de Interfaz

Interfaz #6



Sección “Añadir línea aérea”

Acción del Actor

Respuesta del Sistema

1. El dibujante selecciona la opción Añadir línea aérea.

2. El sistema selecciona y activa el modo de edición en la capa “Líneas Aéreas”. Ver prototipo de interfaz #7.

3. El dibujante añade el elemento (Línea Aérea) en el área seleccionada por el

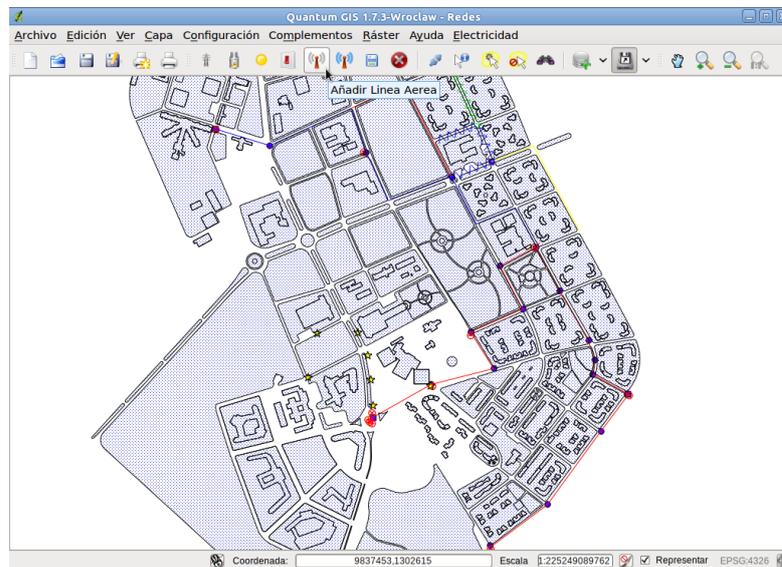
4. El sistema muestra un formulario para introducir los datos del elemento añadido

Capítulo 3. Presentación de la solución propuesta

<p>sistema dando clic continuo y clic derecho cuando ya haya culminado de pintar el objeto sobre el mapa.</p>	<p>(Línea Aérea).</p>
<p>5. El dibujante introduce los valores correspondientes a los atributos de la Línea Aérea.</p>	<p>6. El sistema guarda los atributos y concluye el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.</p>

Prototipo de Interfaz

Interfaz #7



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #8.</p>

Prototipo de Interfaz

Interfaz #8



Sección “Añadir línea soterrada”

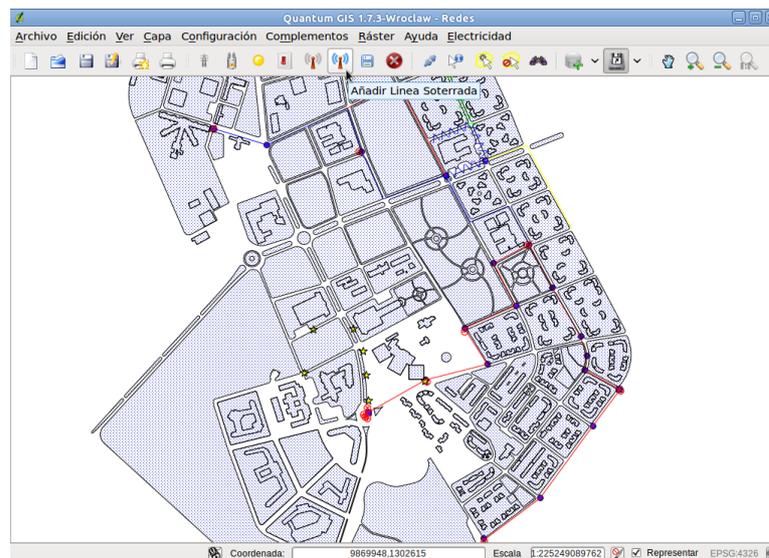
Acción del Actor	Respuesta del Sistema
<p>1. El dibujante selecciona la opción Añadir</p>	<p>2. El sistema selecciona y activa el modo de</p>

Capítulo 3. Presentación de la solución propuesta

línea soterrada.	edición en la capa “Líneas Soterradas”. Ver prototipo de interfaz #9.
3. El dibujante añade el elemento (Línea Soterrada) en el área seleccionada por el sistema dando clic continuo y clic derecho cuando ya haya culminado de pintar el objeto sobre el mapa.	4. El sistema muestra un formulario para introducir los datos del elemento añadido (Línea Soterrada).
5. El dibujante introduce los datos correspondientes a los atributos de la Línea Soterrada.	6. El sistema guarda los atributos y concluye el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.

Prototipo de Interfaz

Interfaz #9



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #10.

Prototipo de Interfaz

Interfaz #10

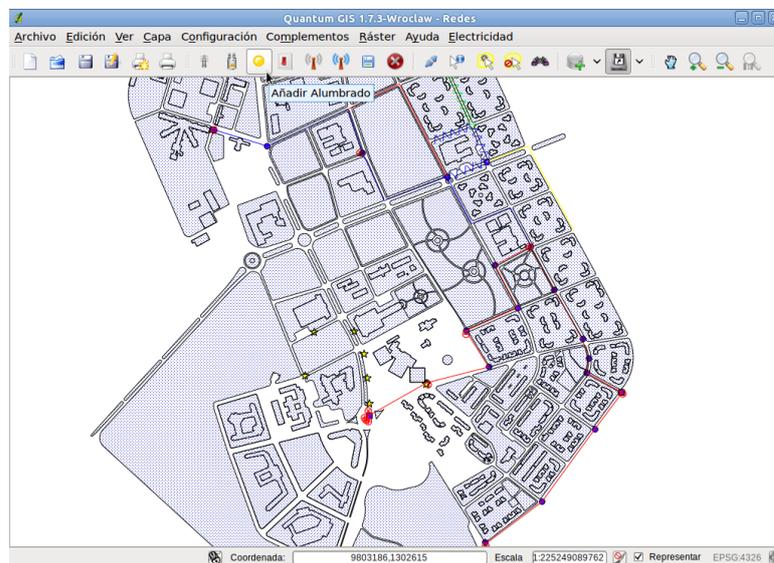


Sección “Añadir alumbrado público”

Acción del Actor	Respuesta del Sistema
1. El dibujante selecciona la opción Añadir alumbrado público.	2. El sistema selecciona y activa el modo de edición en la capa “Alumbrado Público”. Ver prototipo de interfaz #11.
3. El dibujante añade el elemento (Alumbrado público) en el área seleccionada por el sistema dando clic izquierdo sobre el mapa.	4. El sistema muestra un formulario para introducir los datos del elemento añadido (Alumbrado público).
5. El dibujante introduce los datos correspondientes a los atributos del Alumbrado Público.	6. El sistema guarda los atributos y concluye el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.

Prototipo de Interfaz

Interfaz #11



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver

Capítulo 3. Presentación de la solución propuesta

prototipo de interfaz #12.

Prototipo de Interfaz

Interfaz #12

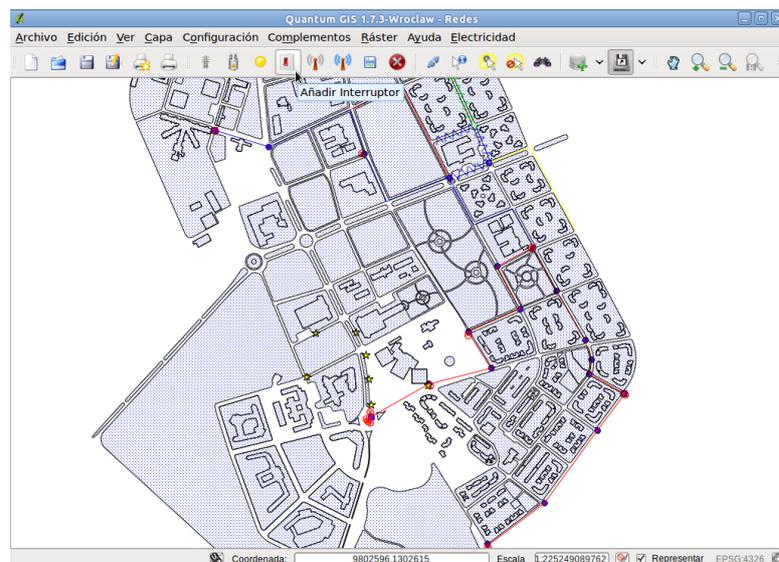


Sección “Añadir interruptor”

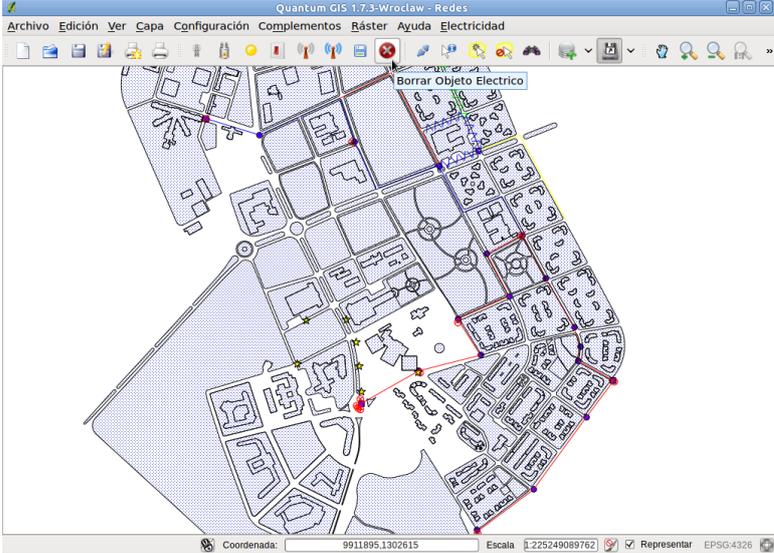
Acción del Actor	Respuesta del Sistema
1. El dibujante selecciona la opción Añadir interruptor.	2. El sistema selecciona y activa el modo de edición en la capa “Interruptores”. Ver prototipo de interfaz #13.
3. El dibujante añade el elemento (Interruptor) en el área seleccionada por el sistema dando clic izquierdo sobre el mapa.	4. El sistema muestra un formulario para introducir los datos del elemento añadido.
5. El dibujante introduce los datos correspondientes a los atributos del Interruptor.	6. El sistema guarda los atributos y concluye el caso de uso como se describe en el caso de uso “Ubicar objeto eléctrico”.

Prototipo de Interfaz

Interfaz #13



Capítulo 3. Presentación de la solución propuesta

Acción del Actor	Respuesta del Sistema
	2. Si la capa relacionada con el objeto no existe, el sistema muestra un mensaje de error. Ver prototipo de interfaz #14.
Prototipo de Interfaz	
Interfaz #14	
	
Sección “Eliminar objeto eléctrico”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. El dibujante hace clic en la opción Eliminar objeto eléctrico seleccionado. Ver prototipo de interfaz #15. 2. El dibujante selecciona el objeto a eliminar dando clic izquierdo sobre el objeto. 	3. El sistema muestra un cuadro de confirmación preguntando si desea eliminar el objeto.
4. El dibujante confirma haciendo clic en el botón aceptar. Ver prototipo de interfaz #16.	5. El sistema elimina el objeto eléctrico seleccionado, finalizando así el caso de uso.
Prototipo de Interfaz	
Interfaz #15	
	

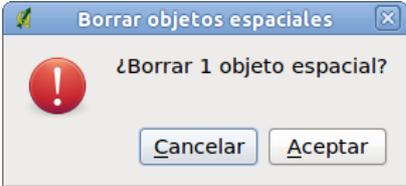
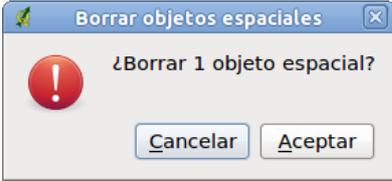
Interfaz #16	
	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
4. El dibujante hace clic en el botón cancelar para no eliminar el objeto. Ver prototipo de interfaz #17.	5. El sistema cancela el borrado del objeto, finalizando el caso de uso.
4. El dibujante hace clic en el botón cerrar (x) para no eliminar el objeto. Ver prototipo de interfaz #17.	5. El sistema cancela el borrado del objeto, finalizando el caso de uso.
Prototipo de Interfaz	
Interfaz #17 	
Poscondiciones	Se añade o elimina un objeto eléctrico al mapa

Tabla 6: Descripción del caso de uso Gestionar objeto eléctrico.

La descripción textual de los restantes casos de uso del sistema se pueden encontrar en el **Anexo 1**.

3.6 Conclusiones Parciales

En el presente capítulo se describió detalladamente el negocio, el cual se modeló a través del Diagrama de Casos de Uso del Negocio. Se precisaron las características del sistema, los requerimientos funcionales y no funcionales concluyendo que el software a desarrollar es de fácil utilización y puesta en práctica, finalmente, es soportado por la mayoría de las computadoras con que cuenta la sociedad cubana en la actualidad y el resto de los países del mundo. Se describió la perspectiva de la solución desde el ámbito del usuario, con el Diagrama de Casos de Uso del Sistema.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.1 Introducción

En este capítulo se detalla la construcción de la solución propuesta, a través de los flujos de Diseño e Implementación, que tiene como objetivo principal transformar los requerimientos en un diseño consecuente con la implementación del sistema. Se establece una línea base de la arquitectura que soporte a los requisitos del sistema a un costo y plazo razonable, mediante diagramas que constituyen la guía vital para el desarrollo correcto de la aplicación. Se define además, el modelo de despliegue originado por la selección de los artefactos más importantes para el sistema donde se precisan los componentes que conforman la estructura física de la aplicación.

4.2 Patrones de Diseño

Los patrones de diseño son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Son soluciones basadas en la experiencia y que se ha demostrado que funcionan. Los patrones de diseño no son fáciles de entender, pero una vez entendido su funcionamiento, los diseños serán mucho más flexibles, modulares y reutilizables. (Gracia, 2005)

Luego de haber realizado un análisis de los patrones de diseño se determinó la aplicación de los patrones GoF, y GRASP. Estos patrones facilitan la construcción y desarrollo del sistema, aportando elementos reutilizables en el diseño, evitando la repetición de búsquedas de soluciones a problemas ya conocidos y ya solucionados anteriormente y definiendo un lenguaje común entre desarrolladores. Los patrones GRSP por su parte brindan una solución elegante para la asignación de responsabilidades.

Patrones GoF:

Patrones de creación: tratan las formas de crear instancias de objetos. El objetivo de estos patrones es de abstraer el proceso de instanciación y ocultar los detalles de cómo los objetos son creados o inicializados.

- **Singleton:** Garantiza que una clase sólo tenga una instancia, y proporciona un punto de acceso global a ella.

Patrones de comportamiento: ayudan a definir la comunicación e iteración entre los objetos de un sistema. El objetivo de este patrón es reducir el acoplamiento entre los objetos, caracterizan las formas en las que interactúan y reparten las responsabilidades entre las distintas clases u objetos.

- **Observador (Observer):** Define una dependencia de uno-a-muchos entre objetos, de forma que cuando un objeto cambia de estado se notifica y actualizan automáticamente todos los objetos.

Patrones GRASP:

- **Alta Cohesión:** Cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable.
- **Bajo Acoplamiento:** Deben haber pocas dependencias entre las clases. Si todas las clases dependen de todas.
- **Controlador (Controller):** Asignar la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad, etc.). El controlador no realiza estas actividades, las delega a otras clases con las que mantiene un modelo de alta cohesión.

Se seleccionaron estos patrones porque cada uno permite que algunos de los aspectos de la estructura del sistema se pueda cambiar independientemente de otros aspectos. Facilitan la reusabilidad, extensibilidad y mantenimiento de los objetos dentro del sistema. Estos patrones le dan al sistema forma y estructura, logrando alcanzar una madures en el diseño.

4.3 Modelo de Diseño

El objetivo principal del flujo de trabajo de diseño es la construcción del modelo de diseño, que permite describir la realización física de los casos de uso, traduciendo los requisitos funcionales y no funcionales a una representación del sistema para crear una entrada apropiada y un punto de partida para las actividades de implementación. Durante esta etapa se generan todas las especificaciones necesarias para la programación del sistema.

4.3.1 Diagrama de Clases del Diseño

El diagrama de clases del diseño permite describir gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. Los diagramas de clases son los más utilizados en el modelo de sistemas orientado a objetos.

A continuación se muestra el **Diagrama de clases del diseño** del caso de uso “Gestionar objeto eléctrico”.

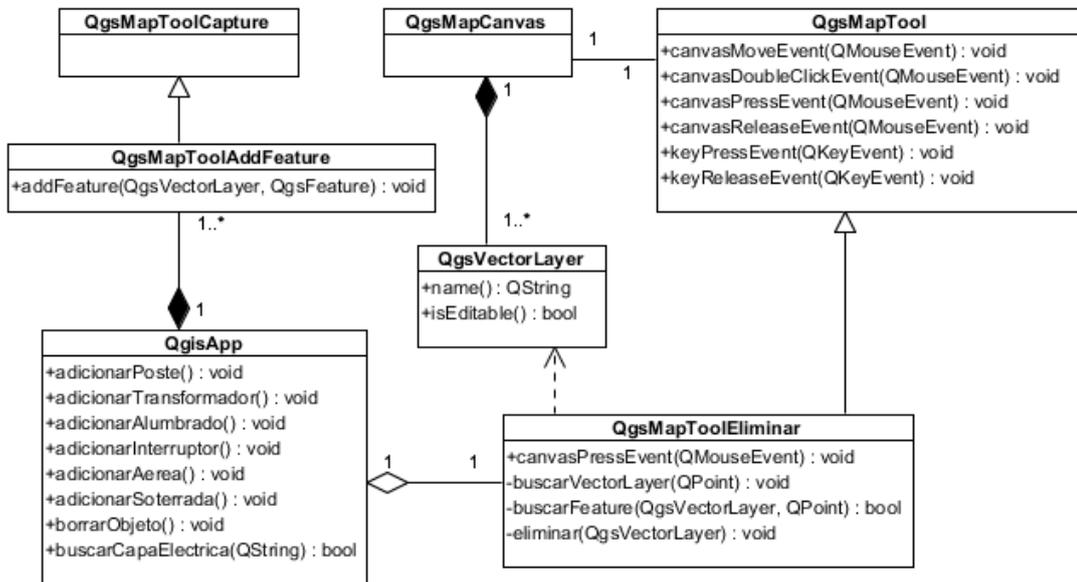


Figura 5: Diagrama de Clases del Diseño: Gestionar objeto eléctrico.

Los diagramas de clases del diseño de los restantes casos de usos del sistema se pueden encontrar en el **Anexo 2**.

4.4 Principios de Diseño

4.4.1 Estándares de la Interfaz de Aplicación

Las interfaces que presentará el subsistema de acuerdo con las normas establecidas dentro de la arquitectura definida y las consideraciones del diseño asumidas por el proyecto productivo, deben ir en concordancia con los colores presentes en la interfaz principal de la plataforma GeoQ, incluyendo sus logos, imágenes representativas y otros elementos. El contenido a mostrar se distribuirá en tres partes fundamentales:

- **El encabezado:** Estará situado ocupando toda la parte superior de las interfaces y en él se encontrará el logo de la aplicación y la identificación de la interfaz.
- **El menú:** El menú se encontrará en la parte superior de la interfaz en consonancia con las funcionalidades presentes, permitiendo al usuario configurar la vista de la interfaz cambiando la posición del menú hacia la parte lateral.
- **El área de trabajo:** Como cualquier diseño estándar, estará situada al centro, desde donde termina el menú hasta el final de la página. Ocupa la mayor parte del espacio ya que incluirá todas las actividades a considerar dentro del sistema que aquí se diseña.

Capítulo 4. Construcción de la solución propuesta

Para lograr un adecuado diseño de la aplicación propuesta se requiere seguir una serie de estándares que van dirigidos a garantizar la consistencia de dicha aplicación y una mayor aceptación por parte de los interesados. Estos estándares se listan a continuación:

- Brindar una interfaz sencilla, de manera tal que cualquier persona con un mínimo dominio de la computación pueda aprender a trabajar con la aplicación.
- Garantizar la legibilidad, de manera que exista contraste de los colores de los textos con el fondo y el tamaño de la fuente para que sea lo suficientemente adecuado a la vista del usuario.
- Mostrar al usuario, solamente aquellas opciones a las que, dado los permisos asignados a su rol, tiene derecho a acceder.
- Mostrar al usuario, siempre que vaya a realizar una acción relevante sobre el sistema, un mensaje de confirmación que le permita asegurarse que es correcta la opción seleccionada.

4.5 Diseño de la Base de Datos

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada, logrando que no existan datos redundantes que aumenten la probabilidad de que se produzcan errores e incoherencias. Para lograr un diseño de la Base de Datos lo más acorde a las necesidades reales propuestas se hace necesario seguir un conjunto de pasos que comienzan con definir las clases persistentes, luego refinarlas y clasificarlas junto con sus atributos, para más tarde realizar el diagrama de clases persistentes.

A continuación se muestra el diagrama de clases persistentes asociado a la aplicación a desarrollar y posteriormente se representa el diagrama entidad relación de la Base de Datos que se genera del diagrama anterior.

4.5.1 Diagrama de Clases Persistentes

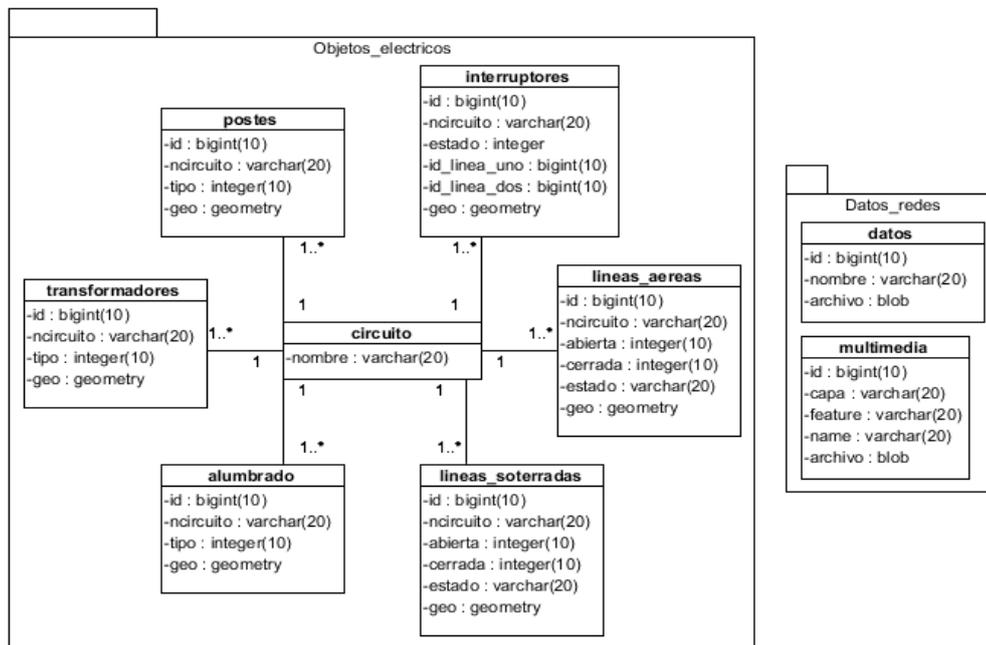


Figura 6: Diagrama de Clases Persistentes.

4.5.2 Modelo Entidad-Relación

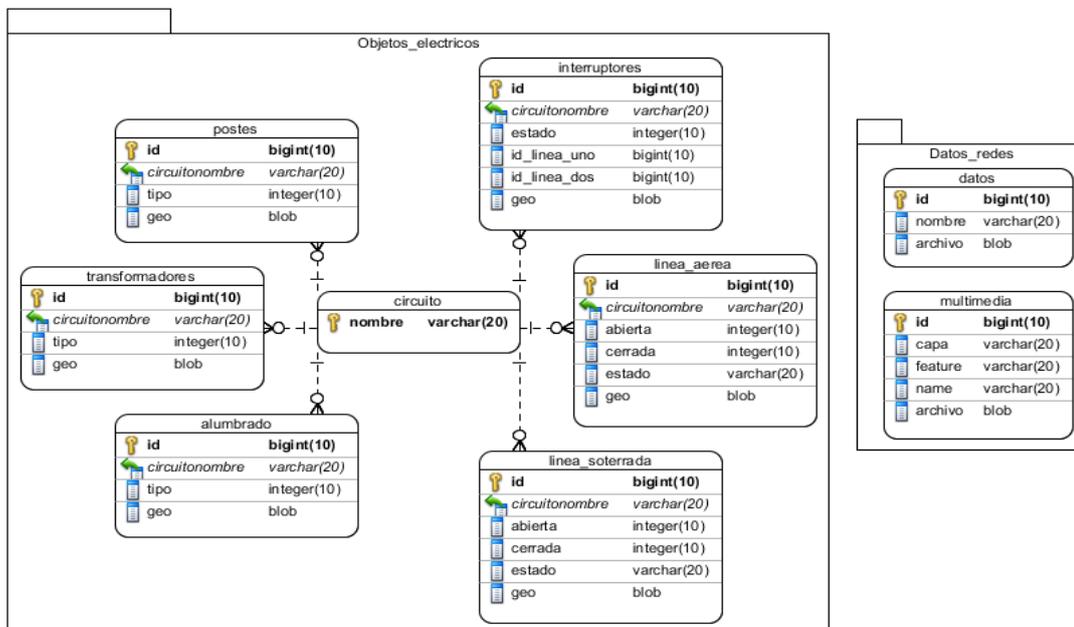


Figura 7: Modelo Entidad-Relación.

4.6 Generalidades de la Implementación

4.6.1 Modelo de Despliegue

El diagrama de despliegue muestra la disposición física de los distintos nodos que componen el sistema y el reparto de los componentes sobre dichos nodos. Un nodo es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, que generalmente tiene algo de memoria y, a menudo, capacidad de procesamiento.

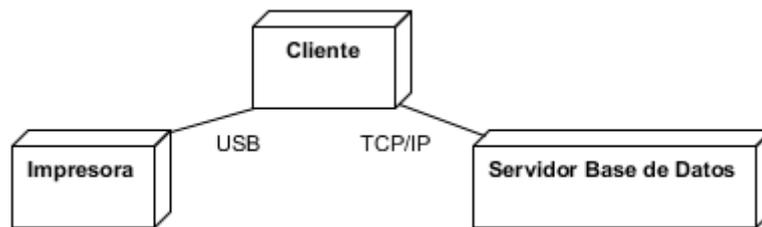


Figura 8: Modelo de Despliegue.

4.6.2 Modelo de Implementación

El modelo de implementación describe, cómo los elementos del modelo de diseño se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructura y modularización disponibles en el entorno de implementación, en los lenguajes de programación utilizados y cómo dependen los componentes unos de otros. (Rumbaugh, y otros, 2000)

A continuación se muestra el **Diagrama de componente** del caso de uso “Gestionar objeto eléctrico”.

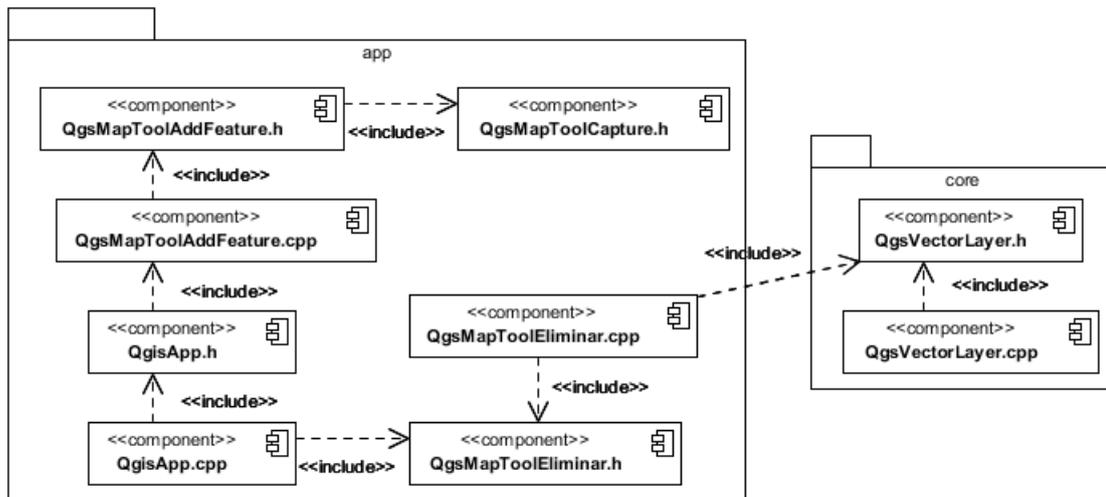


Figura 9: Diagrama Componente: Gestionar objeto eléctrico.

Los diagramas de componentes de los restantes casos de usos del sistema se pueden encontrar en el **Anexo 3**.

4.7 Prueba del Sistema Propuesto

La prueba es el proceso de ejecución de un programa con la intención de descubrir un error. Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones, del diseño y la codificación. La creciente percepción del software como un elemento del sistema y la importancia de los costes asociados a un fallo del propio sistema, están motivando la creación de pruebas minuciosas y bien detalladas. (Pressman, 2005)

Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones del diseño y de la codificación, estas tienen diferentes clasificaciones, algunas de ellas son:

- **Pruebas de Verificación:** Se verifica el cumplimiento de las especificaciones del diseño.
- **Pruebas de Validación:** Se encargan de velar por el cumplimiento de los requisitos del análisis.
- **Pruebas de Caja Blanca:** Se conoce el código y se trata de ejecutar cada uno de los elementos del mismo.
- **Pruebas de Caja Negra:** Solamente se conoce la interfaz y se trata de probar cada uno de los elementos que la componen.

Cualquier tipo de prueba tiene como objetivo principal encontrar errores o defectos en el software. Para probar el sistema propuesto se seleccionó la técnica de prueba de caja negra. Se escoge este tipo de técnica porque permite probar el sistema desde el punto de vista del usuario, lo que debe hacer y no como lo va a hacer. Esta técnica hace la prueba funcional de los requerimientos de la aplicación permitiendo encontrar aquellas funciones incorrectas o ausentes. En la prueba de caja negra se decide no tener en cuenta el funcionamiento interno de un sistema y solo se analizan sus entradas y salidas.

4.7.1 Prueba de Caja Negra

La prueba de Caja Negra, se refieren a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que la salida producida es correcta, ya que no es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar.

Muchos autores consideran que estas pruebas permiten encontrar: (Pressman, 2005)

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las Bases de Datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

Para preparar los casos de pruebas hacen falta un conjunto de datos que ayuden a la ejecución de las funcionalidades y que permitan que el sistema se ejecute en todas sus variantes, pueden ser datos válidos o inválidos, en dependencia de lo que se desee hallar, un error o probar una funcionalidad. Los datos se escogen atendiendo a las especificaciones del problema, sin importar los detalles internos del sistema, a fin de verificar que funcione correctamente.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están: (Pressman, 2005)

- **Técnica de Partición de Equivalencia:** Esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del sistema.
- **Técnica de Análisis de Valores Límites:** Esta técnica prueba la habilidad del sistema para manejar datos que se encuentran en los límites aceptables.
- **Técnica de Grafos de Causa-Efecto:** Es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Capítulo 4. Construcción de la solución propuesta

Dentro de la prueba de Caja Negra la técnica de Partición de Equivalencia es una de las más efectivas, pues permite examinar los valores válidos e inválidos de las entradas existentes en el sistema, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico.

Las clases de equivalencia se pueden definir de acuerdo con las siguientes directrices:

- Si un parámetro de entrada debe estar comprendido en un cierto rango, aparecen 3 clases de equivalencia: por debajo, en y por encima del rango.
- Si una entrada requiere un valor concreto, aparecen 3 clases de equivalencia: por debajo, en y por encima del rango.
- Si una entrada requiere un valor presente en un conjunto de valores, aparecen 2 clases de equivalencia: en el conjunto o fuera de él.
- Si una entrada es booleana, hay 2 clases: si o no.

Aplicando estas directrices se ejecutan casos de prueba para cada elemento de datos del campo de entrada a desarrollar, donde los casos de prueba se seleccionan de forma que ejerciten el mayor número de atributos de cada clase de equivalencia a la vez.

A continuación se muestra los **casos de prueba** utilizando la técnica de Partición de Equivalencia.

Caso de Uso: Gestionar cambio de la red eléctrica.

Descripción general del caso de uso: El caso de uso inicia cuando el despachador necesita gestionar el estado de la red eléctrica en un momento dado, guardando o mostrando el estado de la red o retornando a un cambio anterior.

Condiciones de Ejecución: Para lograr gestionar un cambio en la red eléctrica se tiene que tener cargada la cartografía correspondiente a la red.

Secciones a probar en el caso de uso

Nombre de la sección	SC 1: Guardar el estado de la red en un momento dado.	
Escenarios de la sección	Descripción de la funcionalidad	Flujo central
EC 1.1: Seleccionar "Guardar estado"	Al acceder a la opción "Guardar estado" del menú "Electricidad" o de la barra de herramientas se muestra una ventana, la	1. Ventana Principal. 2. Menú Electricidad. 3. Guardar Estado.

Capítulo 4. Construcción de la solución propuesta

		misma permite introducir el nombre con que se va a guardar el estado de la red. Complementariamente se tienen los botones “Aceptar” y “Cancelar”.	
EC 1.2:	Seleccionar “Aceptar”	Al seleccionar el botón “Aceptar” el sistema guarda el estado de la red en la base de datos.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Guardar Estado. 4. Aceptar.
EC 1.3:	Seleccionar “Cancelar”	Al seleccionar el botón “Cancelar” el sistema cerrará la ventana.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Guardar Estado. 4. Cancelar.

Tabla 7: Secciones a probar en Guardar el estado de la red en un momento dado.

Descripción de las variables.

No.	Nombre del campo	Clasificación	Requerido	Descripción
1	Nombre	Campo de texto	si	Especifica el nombre con que se va a guardar el estado de la red.

Tabla 8: Descripción de las variables en Guardar el estado de la red en un momento dado.

Matriz de datos (SC 1. Guardar el estado de la red en un momento dado)

Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EC 1.1: Seleccionar “Guardar estado”		Al acceder a la opción “Guardar estado” el sistema muestra una ventana.	Satisfactorio.
EC 1.2: Seleccionar “Aceptar”	Redes	Al seleccionar el botón “Aceptar” el sistema guarda el estado de la red en la base de datos.	Satisfactorio.
	24680		Satisfactorio.
EC 1.3: Seleccionar “Cancelar”		Al seleccionar el botón “Cancelar” el sistema cerrará la ventana.	Satisfactorio.

Tabla 9: Matriz de datos en Guardar el estado de la red en un momento dado.

Capítulo 4. Construcción de la solución propuesta

Secciones a probar en el caso de uso

Nombre de la sección	SC 2: Mostrar el estado de la red en un momento dado.	
Escenarios de la sección	Descripción de la funcionalidad	Flujo central
EC 2.1: Seleccionar “Mostrar estado”	Al acceder a la opción “Mostrar estado” del menú “Electricidad” o de la barra de herramientas se muestra una ventana, la misma permite seleccionar el nombre del estado que se va a mostrar en el mapa. Complementariamente se tienen los botones “Aceptar” y “Cancelar”.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Mostrar Estado.
EC 2.2: Seleccionar “Aceptar”	Al seleccionar el botón “Aceptar” el sistema carga y muestra el estado de la red en ese momento.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Mostrar Estado. 4. Aceptar.
EC 2.3: Seleccionar “Cancelar”	Al seleccionar el botón “Cancelar” el sistema cerrará la ventana.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Mostrar Estado. 4. Cancelar.

Tabla 10: Secciones a probar en Mostrar el estado de la red en un momento dado.

Descripción de las variables.

No.	Nombre del campo	Clasificación	Requerido	Descripción
1	Nombre	Campo de selección	si	Especifica el nombre del estado que se va a mostrar en el mapa

Tabla 11: Descripción de las variables en Mostrar el estado de la red en un momento dado.

Matriz de datos (SC 2. Mostrar el estado de la red en un momento dado)

Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EC 2.1: Seleccionar “Mostrar estado”		Al acceder a la opción “Mostrar estado” el sistema muestra una ventana.	Satisfactorio.

Capítulo 4. Construcción de la solución propuesta

EC 2.2: Seleccionar “Aceptar”	Red1	Al seleccionar el botón “Aceptar” el sistema carga y muestra el estado de la red en ese momento.	Satisfactorio.
EC 2.3: Seleccionar “Cancelar”		Al seleccionar el botón “Cancelar” el sistema cerrará la ventana.	Satisfactorio.

Tabla 12: Matriz de datos en Mostrar el estado de la red en un momento dado.

Secciones a probar en el caso de uso

Nombre de la sección		SC 3: Retornar al estado inicial.	
Escenarios de la sección		Descripción de la funcionalidad	Flujo central
EC 3.1: Seleccionar “Retornar estado”	Seleccionar	Al acceder a la opción “Retornar estado” del menú “Electricidad” o de la barra de herramientas el sistema regresa al mapa inicial.	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Retornar Estado.

Tabla 13: Secciones a probar en Retornar al estado inicial.

Matriz de datos (SC 3. Retornar estado)

Escenario	Respuesta del sistema	Resultado de la prueba
EC 3.1: Seleccionar “Retornar estado”	Al acceder a la opción “Retornar estado” el sistema retorna al mapa inicial.	Satisfactorio.

Tabla 14: Matriz de datos en Retornar el estado inicial.

Secciones a probar en el caso de uso

Nombre de la sección		SC 4: Establecer un cambio anterior.	
Escenarios de la sección		Descripción de la funcionalidad	Flujo central
EC 4.1: Seleccionar “Establecer estado”	Seleccionar	Al acceder a la opción “Establecer estado” del menú “Electricidad” o de la barra de herramientas el sistema establece el mapa que se muestra como mapa inicial actualizando todos los	<ol style="list-style-type: none"> 1. Ventana Principal. 2. Menú Electricidad. 3. Establecer Estado.

Capítulo 4. Construcción de la solución propuesta

	datos.	
--	--------	--

Tabla 15: Secciones a probar en Establecer un cambio anterior.

Matriz de datos (SC 4. Retornar estado)

Escenario	Respuesta del sistema	Resultado de la prueba
EC 4.1: Seleccionar "Establecer estado"	Al acceder a la opción "Establecer estado" el sistema establece el mapa que se muestra como mapa inicial actualizando todos los datos.	Satisfactorio.

Tabla 16: Matriz de datos en Establecer un cambio anterior.

Los casos de pruebas de los restantes casos de usos del sistema se pueden encontrar en el **Anexo 4**.

4.8 Conclusiones Parciales

En el presente capítulo se arribó a un sistema completamente diseñado y construido en términos de clases del diseño. Se generaron además cada uno de los artefactos y diagramas referentes al flujo de trabajo de implementación, culminando la modelación completa de la solución propuesta. De esta forma se implementó y validó cada una de las funcionalidades del sistema propuesto, obteniendo resultados satisfactorios.

CONCLUSIONES GENERALES

Con el desarrollo del Sistema de Información Geográfica para automatizar la gestión de la distribución de las redes eléctricas en la Unión Nacional Eléctrica, se cumplen los objetivos propuestos para este trabajo en función de las tareas trazadas para llevar a cabo el proceso investigativo.

Una vez concluida la investigación y analizados los resultados, es posible resaltar una serie de conclusiones que se enumeran a continuación:

1. Las tecnologías, herramientas y framework seleccionados para el desarrollo de la aplicación que se centra la investigación, obedecen a criterios de selección de tecnologías libres y multiplataforma, de acuerdo con las políticas que impulsa la universidad y el país en sentido general.
2. La puesta en explotación de la aplicación desarrollada proporcionará mayor calidad y rapidez, en el proceso de distribución de las redes eléctricas en la unión nacional eléctrica, eliminando errores que pueden introducirse de forma manual y perfeccionar los resultados de análisis para la toma de decisiones en el proceso de distribución de las redes eléctricas.
3. Todos los requisitos funcionales previstos fueron implementados satisfactoriamente, teniendo en cuenta las restricciones no funcionales especificadas a lo largo de la investigación.
4. La aplicación desarrollada es fácil de utilizar, intuitiva, con un diseño sencillo, puede ser ejecutado en computadoras con bajas prestaciones y desde cualquier sistema operativo.
5. El diseño de la prueba de caja negra permitirá validar el cumplimiento de los requisitos funcionales capturados y comprobar que el sistema realiza las acciones que debe realizar en el momento esperado.
6. Conjuntamente se cuenta con una documentación técnica que describe claramente todos los artefactos generados, logrando un mejor entendimiento de los procesos a automatizar.

RECOMENDACIONES

En correspondencia con los resultados obtenidos y la experiencia acumulada a lo largo de todo el proceso investigativo se proponen las siguientes recomendaciones:

1. Establecer un período de capacitación para cada usuario nuevo que deba utilizar el sistema para el desarrollo de sus actividades.
2. Continuar el ciclo de desarrollo de la herramienta, con la identificación de nuevos requisitos funcionales, en aras de ampliar las posibilidades de trabajo del sistema desarrollado.
3. Socializar el resultado de esta investigación en algún espacio de consulta, presentando los resultados en eventos de carácter científico.

TRABAJOS CITADOS

- Boions, Fernando, y otros. 2006.** *Simulación de Redes Eléctricas Integradas a un SIG*. Uruguay : s.n., 2006.
- Carrillo Pérez, Isaias y Pérez Gonzáles, Rodrigo. 2008.** *Metodología de Desarrollo de Software*. Mexico : s.n., 2008.
- Castro López, Alfredo y Meneses Acosta, Boris. 2003.** *Sistema de Información Geográfico para el control y operación de las Redes Eléctricas de Sancti Spíritus*. Sancti Spíritus : s.n., 2003.
- Chaves Pérez, Juan Manuel. 2010.** *GESTION INFORMÁTICA CON SOFTWARE LIBRE*. Cádiz : s.n., 2010.
- Cueva Lovelle, Juan Manuel. 1998.** *CONCEPTOS BÁSICOS DE PROCESADORES DE LENGUAJE*. España : SERVITEC, 1998.
- García de Jalon, Javier y Brazales, Alfonso. 1998.** *Aprenda C++ como si estuviera en primero*. España : s.n., 1998.
- Garlan, David y Shaw, Mary. 1994.** *An Introduction to Software Architecture*. New Jersey : World Scientific, 1994. 15213-3890.
- González, Carlos. 2011.** Curso PostgreSQL, SQL avanzado y PHP. [En línea] Diciembre de 2011. [Citado el: 5 de Diciembre de 2011.] <http://www.usabilidadweb.com.ar/postgre.php>.
- Gracia, Joaquin. 2005.** IngenieroSoftware. [En línea] 27 de Mayo de 2005. [Citado el: 20 de enero de 2012.] <http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>. IBNH/.
- Igbokwe, J.I y Emengini, E.J. 2002.** *SIG en la gestión de la Red de Distribución Eléctrica: Un estudio de caso de Onitsha-Norte LGA, el Estado de Anambra, Nigeria*. Anambra : s.n., 2002.
- Iglesia, Carlos. 2011.** sensagent. *sensagent*. [En línea] 2011. [Citado el: 20 de Noviembre de 2011.] <http://dictionary.sensagent.com/postgis/es-es/>.
- López, Alfredo Castro y Acosta, Boris Meneses. 2003.** *Sistema de Información Geográfico para el control y operación de las Redes Eléctricas de Sancti Spíritus*. Sancti Spíritus : s.n., 2003.
- Matheus, Jorge y Moreira, Arturo. 2008.** Ingeniería y Construcción MATMOR, C.A. *Ingeniería y Construcción MATMOR, C.A.* [En línea] 2008. [Citado el: 25 de Noviembre de 2011.] <http://matmor.dyndns.org/padee.html>.
- Meyer, Hector Hugo. 2009.** *Aplicación de Tecnologías GIS para optimizar la calidad de servicios en los Sistemas de Distribución Eléctricos Rural en la provincia de Córdoba*. Córdoba : s.n., 2009.
- Moreira, Ing. Jorge Matheus y Ing. Arturo.** Ingeniería y Construcción MATMOR, C.A. [En línea] [Citado el: 25 de Noviembre de 2011.] <http://matmor.dyndns.org/padee.html>.
- Morón, Juan Antonio Yebra. 2009.** *Sistemas Eléctricos de Distribución*. México : EDITORIAL REVERTÉ, 2009.

Referencias Bibliográficas

Nicanor, Lisandro Damián. 2007. *Introducción al desarrollo con Qt.* 2007.

Olaya, Víctor. 2010. *Sistemas de Información Geográfica.* Girona : s.n., 2010.

Pockin. 2008. *Modelado de Sistemas com UML.* 2008.

Pressman, Roger S. 2005. *Software Engineering.* New York : The MacGraw-Hill, 2005. 978-0-07-337598-7.

Reynoso, Carlos. 2004. *Introducción a la arquitectura de software.* Buenos Aires : s.n., 2004.

Rumbaugh, J., Booch, G. y Jacobson, L. 2000. *El Proceso Unificado de Desarrollo.* 2000.

Sanabria, Javier Manrique. 2011. *Conceptos de SIG y Cartografía.* Bogotá : s.n., 2011.

Sanabria, Raúl César Vilcahuamán. 1993. *Análisis Interactivo Gráfico de Sistemas Eléctricos de Distribución Primaria.* Santiago de Chile : s.n., 1993.

Skrlec, Davor, Krajcar, Slavoko y Blagajac, Snjezana. 1994. *Aplicación de la Tecnología de Optimización de los SIG en la Red de Distribución Eléctrica.* Croacia : s.n., 1994.

T&D, CYME International. 2004. Soluciones y software para sistemas eléctricos en potencia. [En línea] 9 de Julio de 2004. [Citado el: 20 de Noviembre de 2011.] <http://www.cyme.com/es/>.

Tobarra Narro, Manuel. 2003. *CMM y RUP: Una perspectiva común.* Albacete : s.n., 2003.

BIBLIOGRAFÍA

- Boions, Fernando, y otros. 2006.** *Simulación de Redes Eléctricas Integradas a un SIG*. Uruguay : s.n., 2006.
- Carrillo Pérez, Isaias y Pérez Gonzáles, Rodrigo. 2008.** *Metodología de Desarrollo de Software*. Mexico : s.n., 2008.
- Castro López, Alfredo y Meneses Acosta, Boris. 2003.** *Sistema de Información Geográfico para el control y operación de las Redes Eléctricas de Sancti Spíritus*. Sancti Spíritus : s.n., 2003.
- Chaves Pérez, Juan Manuel. 2010.** *GESTION INFORMÁTICA CON SOFTWARE LIBRE*. Cádiz : s.n., 2010.
- Cueva Lovelle, Juan Manuel. 1998.** *CONCEPTOS BÁSICOS DE PROCESADORES DE LENGUAJE*. España : SERVITEC, 1998.
- García de Jalon, Javier y Brazales, Alfonso. 1998.** *Aprenda C++ como si estuviera en primero*. España : s.n., 1998.
- Garlan, David y Shaw, Mary. 1994.** *An Introduction to Software Architecture*. New Jersey : World Scientific, 1994. 15213-3890.
- González, Carlos. 2011.** Curso PostgreSQL, SQL avanzado y PHP. [En línea] Diciembre de 2011. [Citado el: 5 de Diciembre de 2011.] <http://www.usabilidadweb.com.ar/postgre.php>.
- Gracia, Joaquin. 2005.** IngenieroSoftware. [En línea] 27 de Mayo de 2005. [Citado el: 20 de enero de 2012.] <http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>. IBNH/.
- Igbokwe, J.I y Emengini, E.J. 2002.** *SIG en la gestión de la Red de Distribución Eléctrica: Un estudio de caso de Onitsha-Norte LGA, el Estado de Anambra, Nigeria*. Anambra : s.n., 2002.
- Iglesia, Carlos. 2011.** sensagent. *sensagent*. [En línea] 2011. [Citado el: 20 de Noviembre de 2011.] <http://dictionary.sensagent.com/postgis/es-es/>.
- López, Alfredo Castro y Acosta, Boris Meneses. 2003.** *Sistema de Información Geográfico para el control y operación de las Redes Eléctricas de Sancti Spíritus*. Sancti Spíritus : s.n., 2003.
- Matheus, Jorge y Moreira, Arturo. 2008.** Ingeniería y Construcción MATMOR, C.A. *Ingeniería y Construcción MATMOR, C.A.* [En línea] 2008. [Citado el: 25 de Noviembre de 2011.] <http://matmor.dyndns.org/padee.html>.
- Meyer, Hector Hugo. 2009.** *Aplicación de Tecnologías GIS para optimizar la calidad de servicios en los Sistemas de Distribución Eléctricos Rural en la provincia de Córdoba*. Córdoba : s.n., 2009.
- Moreira, Ing. Jorge Matheus y Ing. Arturo.** Ingeniería y Construcción MATMOR, C.A. [En línea] [Citado el: 25 de Noviembre de 2011.] <http://matmor.dyndns.org/padee.html>.
- Morón, Juan Antonio Yebra. 2009.** *Sistemas Eléctricos de Distribución*. México : EDITORIAL REVERTÉ, 2009.

Nicanor, Lisandro Damián. 2007. *Introducción al desarrollo con Qt.* 2007.

Olaya, Víctor. 2010. *Sistemas de Información Geográfica.* Girona : s.n., 2010.

Pockin. 2008. *Modelado de Sistemas con UML.* 2008.

Pressman, Roger S. 2005. *Software Engineering.* New York : The MacGraw-Hill, 2005. 978-0-07-337598-7.

Reynoso, Carlos. 2004. *Introducción a la arquitectura de software.* Buenos Aires : s.n., 2004.

Rumbaugh, J., Booch, G. y Jacobson, L. 2000. *El Proceso Unificado de Desarrollo.* 2000.

Sanabria, Javier Manrique. 2011. *Conceptos de SIG y Cartografía.* Bogotá : s.n., 2011.

Sanabria, Raúl César Vilcahuamán. 1993. *Análisis Interactivo Gráfico de Sistemas Eléctricos de Distribución Primaria.* Santiago de Chile : s.n., 1993.

Skrlec, Davor, Krajcar, Slavoko y Blagajac, Snjezana. 1994. *Aplicación de la Tecnología de Optimización de los SIG en la Red de Distribución Eléctrica.* Croacia : s.n., 1994.

T&D, CYME International. 2004. Soluciones y software para sistemas eléctricos en potencia. [En línea] 9 de Julio de 2004. [Citado el: 20 de Noviembre de 2011.] <http://www.cyme.com/es/>.

Tobarra Narro, Manuel. 2003. *CMM y RUP: Una perspectiva común.* Albacete : s.n., 2003.

GLOSARIO DE TÉRMINOS

Arquitectura de software: Conjunto de decisiones significativas acerca de la organización de un sistema (software). La arquitectura no solo se interesa por la estructura y el comportamiento, sino también por las restricciones y compromisos de uso, funcionalidad, funcionamiento, flexibilidad al cambio, reutilización, compresión, economía y tecnología, así como por aspectos estéticos del software.

Artefacto: Pieza de información tangible que es creada, modificada y usada por los trabajadores al realizar actividades. Un artefacto puede ser un modelo, un elemento de un modelo o un documento.

Dirigido por caso de uso: En el contexto del ciclo de vida del software, indica que los casos de uso se utilizan como artefacto principal para definir el comportamiento deseado para el sistema, y para comunicar este comportamiento entre las personas involucradas en el sistema.

Diseño (flujo de trabajo): Flujo de trabajo fundamental cuyo propósito principal es el de formular modelos que se centran en los requisitos no funcionales y el dominio de la solución, y que prepara para la implementación y pruebas del sistema.

Fase: Periodo de tiempo entre dos hitos principales de un proceso de desarrollo.

Fase de construcción: Tercera fase del ciclo de vida del software, en la que el software es desarrollado a partir de una línea base de la arquitectura ejecutable, hasta el punto en el que está listo para ser transmitido a la comunidad de usuarios.

Fase de elaboración: Segunda fase del ciclo de vida en la que se define la arquitectura.

Fase de inicio: Primera fase del ciclo de vida del software.

Fase de transición: Cuarta fase del ciclo de vida del software, en la que este es puesto en manos de la comunidad de usuarios.

Flujo de trabajo: Realización de un caso de uso o parte de él. Puede describirse en términos de diagrama de actividad, que incluye a los trabajadores participantes, las actividades que realizan y los artefactos que producen.

Herramientas: Son los ambientes de apoyo necesario para automatizar las prácticas de Ingeniería de Software.

IEEE: Es el Instituto de Ingenieros Eléctricos y Electrónicos, es una Asociación Profesional técnica sin fines de lucro de más de 350,000 miembros individuales en 175 países. El IEEE es la principal autoridad en áreas técnicas que van de la Ingeniería en Computación, Ingeniería Eléctrica, Aeroespacial, Electrónica, Tecnología Biomédica, Telecomunicaciones, Electrónica de consumo, entre otros.

Implementación (flujo de trabajo): Flujo de trabajo fundamental cuyo propósito esencial es implementar el sistema en términos de componentes, es decir, código fuente, guiones, ficheros binarios, ejecutables etc.

Mapa: Es la representación de un territorio en un plano, que incluye la ubicación, características de magnitud, distribución y relaciones de los fenómenos naturales, geográficos y sociales, usando símbolos convencionales. Sus elementos principales son: proyección, ubicación, escala y orientación.

Métodos: Son las maneras que se efectúan las tareas de Ingeniería de Software o las actividades del ciclo de vida.

Proyecto: Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

Requisitos: Capacidades, condiciones o cualidades que el sistema debe cumplir y tener.

Transmisión: Es la parte del sistema de suministro eléctrico constituida por los elementos necesarios para llevar hasta los puntos de consumo y a través de grandes distancias la energía eléctrica generada en las centrales eléctricas.

Usuario: Individuo u organización que interactúa con un sistema.