



Ingeniería Informática

Sistema de Información Geográfica para la gestión de documentos técnicos de la Empresa Filial de Investigaciones Aplicadas Cienfuegos



Autores:

Jahdiel López Fiffe
Karlen Jesús Fernández Bisquet

Tutores:

Ing. Isali Azpiri Medina. Universidad de Cienfuegos.
MSc. Elio Cruz. EFIA Cienfuegos

Consultantes:

Dr.C. Eduardo Concepción Morales. Universidad de Cienfuegos
MSc. Richard Darian Sánchez Rivero. Universidad de Cienfuegos
MSc. Vivian Escudero Rodríguez. EFIA Cienfuegos
Ing. Gilberto Lasoncel. EFIA Cienfuegos

**Cienfuegos, Cuba
Curso 2024**



*“La tecnología SIG es la herramienta para implantar el
paradigma del cambio, no es la fuerza que lo maneja”*

W.Huxhold, Allan Levinshon

AGRADECIMIENTOS

En primer lugar, deseamos expresar nuestros más sinceros agradecimientos a Isali Azpiri, tutora de la tesis por su guía, apoyo y valiosos consejos a lo largo de todo el proceso de investigación y redacción. Su experiencia y conocimientos han sido fundamentales para el desarrollo de este trabajo. A nuestros consultantes, los profesores del Departamento de Informática de la Universidad de Cienfuegos, Richard Sánchez, Eduardo Concepción y Domingo Valladares, deseamos agradecerle por sus consejos, sugerencias y apoyo en el desempeño de este trabajo.

Asimismo, agradecemos a Gilberto Lasoncel, Vivian Escudero, y a Elio Cruz, por su disposición a ayudarnos y guiarnos durante la realización de este proyecto. Sus aportes han enriquecido significativamente este trabajo.

Un agradecimiento especial a la Empresa Filial de Investigaciones Aplicadas Cienfuegos por darnos el espacio para realizar esta investigación facilitar el acceso a datos y permitir realizar las prácticas en su institución. Su contribución ha sido clave para la concreción de esta tesis.

Finalmente, queremos expresar nuestra gratitud a nuestra familia y amigos cercanos por su apoyo incondicional, paciencia y comprensión durante la realización de este proyecto. Ellos han sido un pilar fundamental en este recorrido.

A todos ustedes, muchas gracias.

*Karlen Jesús Fernández Bisquet
Jahdiel Lopez Fiffe*

RESUMEN

La presente investigación estuvo orientada al desarrollo de un Sistema de Información Geográfica para el tratamiento de la información del Archivo Técnico de la Empresa Filial de Investigaciones Aplicadas Cienfuegos, el cual permite recuperar y actualizar la información relacionada con las obras constructivas de la provincia y las perforaciones en el terreno para los estudios geofísicos que se realizan en esta entidad. Esta información se encuentra en diferentes formatos y en ocasiones en estado de deterioro, información muy valiosa que permite analizar condiciones de futuras construcciones por los trabajos realizados por la empresa durante años y que con la implementación de esta herramienta podrá unificarse con todas las facilidades de su actualización. La información será reflejada de forma geoespacial en mapas con temáticas de estudio en el municipio de Aguada de Pasajeros de la provincia de Cienfuegos, desarrollándose complementos en la aplicación QGIS y el Sistema Gestor de Bases de Datos PostgreSQL con su extensión espacial PostGIS. El Sistema se implementa en arquitectura cliente-servidor a través de la metodología de desarrollo de software Proceso Unificado Ágil.

Palabras clave: Archivo Técnico, obras constructivas, perforaciones en el terreno, estudios geofísicos, información geoespacial, mapas, Sistema Gestor de Bases de Datos, extensión espacial.

ABSTRACT

The present research was oriented to the development of a Geographic Information System for the treatment of the information of the Technical File of the Cienfuegos Applied Research Subsidiary Company, which allows recovering and updating the information related to the construction works of the province and the perforations in the field for the geophysical studies that are carried out in this entity. This information is in different formats and sometimes in a state of deterioration, very valuable information that allows analyzing conditions of future constructions due to the works carried out by the company for years and that with the implementation of this tool can be unified with all the facilities of its updating. The information will be reflected in a geospatial way in maps with study themes in the municipality of Aguada de Pasajeros in the province of Cienfuegos, developing complements in the QGIS application and the PostgreSQL Database Management System with its PostGIS spatial extension. The system is implemented in client-server architecture through the Agile Unified Process software development methodology.

Keywords: Technical File, construction works, field drilling, geophysical surveys, geospatial information, maps, Database Management System, spatial extension.

ÍNDICE

INTRODUCCIÓN	1
1 - FUNDAMENTOS TEÓRICOS	7
1.1 - INTRODUCCIÓN	7
1.2 - DESCRIPCIÓN DEL DOMINIO DEL PROBLEMA	7
1.2.1 - <i>Conceptos asociados al dominio del problema</i>	7
1.3 - FLUJO ACTUAL DE LOS PROCESOS Y ANÁLISIS CRÍTICO DE LA EJECUCIÓN DE ESTOS	17
1.3.1 - <i>Descripción del flujo actual de los procesos:</i>	17
1.3.2 - <i>Análisis crítico de la ejecución actual de los procesos</i>	17
1.4 - DESCRIPCIÓN DE LOS SISTEMAS EXISTENTES	18
1.4.1 - <i>Antecedentes Internacionales</i>	19
1.4.2 - <i>Antecedentes Nacionales:</i>	20
1.4.3 - <i>Softwares utilizados en el desarrollo de SIG's:</i>	20
1.5 - METODOLOGÍAS, TECNOLOGÍAS ACTUALES Y ARQUITECTURA DE SOFTWARE	21
1.5.1 - <i>Metodología AUP</i>	22
1.5.2 - <i>Arquitecturas de software</i>	25
1.5.3 - <i>Entorno de desarrollo y herramientas utilizadas</i>	26
1.5.4 - <i>Lenguajes de programación utilizados</i>	26
1.5.5 - <i>Otras tecnologías o librerías utilizadas</i>	27
1.5.6 - <i>Sistema Gestor de Base de Datos</i>	27
1.5.7 - <i>PostgreSQL</i>	28
1.6 - CONCLUSIONES PARCIALES	30
2 - PROPUESTA DE SOLUCIÓN	31
2.1 - INTRODUCCIÓN	31
2.2 - CONCEPCIÓN GENERAL DEL SISTEMA	31
2.2.1 - <i>SIG personalizado</i>	32
2.2.2 - <i>Capas temáticas del SIG</i>	32

2.2.3 - Recopilación y organización de los datos	39
2.2.4 - Desarrollo de complementos para un SIG	40
2.3 - REQUISITOS	43
2.3.1 - Requerimientos funcionales	43
2.3.2 - Requerimientos no funcionales	45
2.3.3 - Historias de usuario	46
2.4 - ANÁLISIS Y DISEÑO	49
2.4.1 - Descripción de la arquitectura del software	49
2.4.2 - Patrón de arquitectura Multicapas	49
2.4.3 - Patrones de diseño	51
2.4.4 - Construcción de la solución propuesta	52
2.4.5 - Diagrama del modelo físico de datos	53
2.4.6 - Diagramas de componentes	54
2.4.7 - Diagramas de clases	55
2.5 - CONCLUSIONES PARCIALES	56
3 - VALIDACIÓN DE LA SOLUCIÓN	57
3.1 - INTRODUCCIÓN	57
3.2 - DIAGRAMA DE DESPLIEGUE	57
3.3 - VALIDACIÓN DE LA SOLUCIÓN	58
3.3.1 - Métrica aplicada a los requisitos	58
3.3.2 - Pruebas de caja negra	59
3.3.2 - Pruebas de caja blanca	63
3.3.3 - Pruebas de aceptación	74
3.3 - CONCLUSIONES PARCIALES	75
CONCLUSIONES GENERALES	77
RECOMENDACIONES	78
REFERENCIAS BIBLIOGRÁFICAS	79

GLOSARIO DE TÉRMINOS

84

ANEXOS

ÍNDICE DE FIGURAS

1 Búfers a puntos, líneas y polígonos.....	15
2 Flujo actual de los procesos de la EFIA Cienfuegos	17
3 Fases de AUP	24
4 Capas Obras y Calas del municipio Aguada de Pasajeros.....	33
5 Capa Relieve.....	35
6 Capa Viales	36
7 Capa Ríos.....	36
8 Capa Embalses y Microembalses	37
9 Capa Geología.....	38
10 Capa Poblados	38
11 Capa Municipios	39
12 Estructura general de un plugin de QGIS	42
13 Arquitectura Cliente - Servidor.....	49
14 Modelo físicos de datos.....	53
15 Diagrama de componentes.....	54
16 Diagrama de clases.....	55
17 Diagrama de despliegue	57
18 Grafo de la prueba de caja blanca.....	68
19 Vista general del SIG.....	86
20 Plugin Obras y Calas y sus herramientas.....	86
21 Ejemplo de un búfer creado con el plugin	87
22 Resultados de la consulta realizada con el plugin.....	87
23 Herramienta Empaquetar para QField.....	88
24 Captura realizada desde un móvil de la aplicación QField.....	89
25 Herramienta Filtrar datos	90

ÍNDICE DE TABLAS

1 Tabla de requisitos no funcionales	45
2 Historia de Usuario RF1	47
3 Historia de usuario RF2	48
4 Caso de prueba 1 (caja negra)	61
5 Caso de prueba 2 (caja negra)	61
6 Caso de prueba 3 (caja negra)	62
7 Caso de prueba 4 (caja negra)	62
8 Caso de prueba 5 (caja negra)	62
9 Caso de prueba 6 (caja negra)	63
10 Caso de prueba 1 (caja blanca)	71
11 Caso de prueba 2 (caja blanca)	71
12 Caso de prueba 3 (caja blanca)	72
13 Caso de prueba 4 (caja blanca)	73
14 Tabla de pruebas de aceptación.....	75
15 Historia de usuario RF3	91
16 Historia de usuario RF4	92
17 Historia de usuario RF5	93
18 Historia de usuario RF6	94
19 Historia de usuario RF7	95
20 Historia de usuario RF8	96
21 Historia de usuario RF9	97
22 Historia de usuario RF10	98
23 Historia de usuario RF11	99
24 Historia de usuario RF12	100
25 Historia de usuario RF13	101
26 Historia de usuario RF14	102

INTRODUCCIÓN

La era actual ha sido testigo de una revolución tecnológica sin precedentes, impulsada significativamente por los desarrollos en informática y comunicaciones. Estos avances han llevado a una difusión masiva de tecnologías innovadoras, la expansión global de redes de comunicación y un acceso sin fronteras a la información. Además, la capacidad de integrar datos espaciales¹ en mapas digitales ha revolucionado la manera en que analizamos y visualizamos información, permitiendo una comprensión más profunda y detallada del mundo que nos rodea. [1]

A finales del siglo XX, el desarrollo comercial de los Sistemas de Información Geográfica (SIG) impulsó el perfeccionamiento de la planificación y el ordenamiento espacial en áreas relacionadas con el medio ambiente y los recursos naturales. Esto permitió abrir nuevos horizontes al explotar la capacidad de estos sistemas para procesar datos de diversa índole en estrecha relación con la cartografía.

Desde entonces, la evolución de los SIG ha estado vinculada al avance de las tecnologías de la información y las comunicaciones. En 1990, el National Center for Geographic Information and Analysis (NCGIA) de Estados Unidos definió a los SIG como "*un sistema de hardware, software y procedimientos diseñado para facilitar la obtención, gestión, manipulación, análisis, modelado, representación y salida de datos espacialmente referenciados, con el fin de resolver problemas complejos de planificación y gestión*". Esta definición concibe a la tecnología SIG como un recurso necesario para optimizar las acciones y análisis relacionados con la planificación y gestión en una amplia gama de campos de la sociedad. [2]

Un Sistema de Información Geográfica (SIG) es un sistema de hardware, software, datos, personas, organizaciones y convenios institucionales para la recopilación, almacenamiento, análisis y distribución de información de territorios de la Tierra. El uso

¹ **datos espaciales:** también conocidos como geodatos, son datos que incluyen información relacionada con ubicaciones en la superficie de la Tierra.

de estas técnicas ha dotado a los profesionales con la capacidad de crear sus propios mapas digitales y representar información sobre el mundo real de una manera visual. [3] Estas técnicas también revelan patrones, tendencias y relaciones sobre cualquier elemento que tenga un contexto geoespacial. Los SIG se han convertido en poderosos sistemas para la elaboración, mantenimiento y visualización de información espacial, dando paso a los conocidos SIG. [4]

La Empresa Filial de Investigaciones Aplicadas Cienfuegos (EFIA Cienfuegos) [5] es subordinada a la Empresa Nacional de Investigaciones Aplicadas (INVESCONS) [6], perteneciente al Grupo Empresarial de Diseño e Ingeniería de la Construcción (GEDIC), adscrito al Ministerio de la Construcción de Cuba (MICONS), esta entidad presta una amplia gama de servicios vinculados al sector de la Construcción que comprenden:

- Investigaciones ingeniero-geológicas
- Estudios Ambientales
- Laboratorio de Materiales de la Construcción
- Laboratorio de Suelos
- Control de las Obras Viales
- Patología y Restauración de Edificaciones
- Control de la Calidad de Estructuras y de Hormigón hidráulico
- Estudios Topográficos² y Perforación

Con todos estos servicios prestados, la EFIA ha acumulado una gran cantidad de información técnica desde la década de los 70 del siglo XX. Esta información, que es crucial para los estudios en las obras de construcciones tanto existentes como proyectadas, se encuentra almacenada en diferentes formatos y tecnologías, tanto en copia dura³ como digital. Sin embargo, gran parte de esta información se encuentra en estado de deterioro debido al tiempo de existencia y las condiciones de conservación.

² **estudio topográfico:** es la primera fase del estudio técnico y descriptivo de un terreno.

³ **copia dura:** es una reproducción física permanente de datos que se puede sostener y leer. La copia dura se realiza en papel.

Además, la información digital se encuentra dispersa en diferentes tecnologías, lo que dificulta su consulta y evaluación.

Por lo tanto, el desafío que se plantea en esta tesis es cómo mejorar el tratamiento de esta valiosa información técnica, teniendo en cuenta las limitaciones actuales y las necesidades futuras de la EFIA Cienfuegos. Esto podría implicar la exploración de nuevas tecnologías y metodologías para la gestión de datos, así como la implementación de mejores prácticas para la conservación y organización de la información.

Por lo tanto, el **problema a resolver** es:

¿Cómo mejorar el tratamiento de la información técnica de la Empresa Filial de Investigaciones Aplicadas Cienfuegos?

Esto podría implicar la exploración de nuevas tecnologías y metodologías para la gestión de datos, así como la implementación de prácticas más novedosas y eficientes para la conservación y organización de la información.

La investigación tiene como **objeto de estudio**: la información del archivo técnico de la EFIA Cienfuegos. Delimitando el **campo de acción** en el proceso de organización y actualización de la información de forma espacial y centralizada en la EFIA Cienfuegos.

Para la realización de esta investigación se propone como **idea a defender** la creación de un nuevo **sistema** de integración de datos geográficos de diferente procedencia y estructura, utilizando tecnologías actuales en un Sistema de Información Geográfica (SIG) de forma informatizada. Este sistema facilitaría la recuperación, actualización y mantenimiento de los datos en el sistema, lo que a su vez permitiría brindar un mejor servicio con esta información a la Empresa Filial de Investigaciones Aplicadas Cienfuegos (EFIA).

Para resolver el problema se traza el siguiente **objetivo general**:

Desarrollar un Sistema de Información Geográfica para la recuperación y mantenimiento de la información del Archivo Técnico de la Empresa Filial de Investigaciones Aplicadas Cienfuegos.

Para cumplir con este objetivo general se plantean los siguientes **objetivos específicos**:

1. Investigar las tecnologías, metodologías y mecanismos para digitalizar y recopilar la información del Archivo Técnico.
2. Diseñar un sistema informático para la gestión de la información del Archivo Técnico.
3. Implementar las funcionalidades requeridas en el sistema diseñado.
4. Validar las funcionalidades de la propuesta de solución.

Para alcanzar estos objetivos se precisa la realización de diferentes **tareas** dentro de la investigación, como son:

1. Realizar un estudio del proceso actualización de la información del Archivo Técnico de la EFIA Cienfuegos.
2. Entrevistar a los trabajadores que intervienen en los procesos de actualización y utilización de la información.
3. Estudiar los sistemas existentes relacionados con el problema a resolver.
4. Seleccionar las herramientas, metodologías, lenguajes y tecnologías para la implementación del sistema.
5. Estudiar la base de datos para la persistencia de la información.
6. Diseñar la interfaz gráfica de la aplicación.
7. Implementar las funcionalidades del Sistema de Información Geográfica.
8. Validar la propuesta de solución.
9. Documentar la información generada durante el negocio, diseño e implementación del sistema.

El **aporte práctico** de esta investigación consiste en la utilidad que brindará la puesta en acción de este sistema unificando toda la información de forma espacial y centralizada

para su recuperación, conservación y actualización. Este SIG también agilizará considerablemente el tiempo empleado para en la gestión de la información y para dar respuesta a las tareas devenidas de su uso, permitiendo una rápida y efectiva toma de decisiones a los técnicos y directivos de la entidad.

A continuación, se explican los métodos de la metodología de investigación científica empleados para la obtención de información y el futuro desarrollo del SIG.

Métodos teóricos:

1. **Histórico-Lógico:** Análisis histórico de los Sistemas de Información Geográfica desarrollados. Para la realización de la investigación se hizo necesario estudiar la evolución del problema y la existencia de metodologías, procedimientos y sistemas informáticos similares, determinando cuales son las tendencias actuales para el desarrollo de los SIG.
2. **Analítico-Sintético:** Posibilitó realizar un análisis de las distintas partes que afectan el objeto de estudio y sintetizar los elementos más significativos.

Métodos empíricos:

1. **Observación:** Se empleó para identificar características en el proceso de actualización y empleo de la información del archivo de la forma tradicional.
2. **Entrevista:** Se realizó para poder obtener información sobre cómo se realiza el proceso de actualización de la información del Archivo Técnico.

El trabajo presenta la siguiente estructura: Introducción, Tres Capítulos, Conclusiones Generales, Recomendaciones, Referencias Bibliográficas y Anexos.

Capítulo 1: Fundamentos teóricos

Fundamentación teórica del Sistema de Información Geográfica referente al trabajo con los documentos del Archivo Técnico. En el capítulo se definen los conceptos relacionados con el dominio de la investigación. De igual forma, se realizó un estudio sobre los SIG relacionados con el tema de investigación para identificar posibles soluciones para el

problema definido. También se definió la metodología y las herramientas a emplear en la construcción del SIG.

Capítulo 2: Propuesta de solución

En el capítulo se describe la propuesta de solución del problema de investigación, se especificaron los requisitos funcionales y no funcionales que debe cumplir el software. Se realizaron las historias de usuarios, el modelo físico de los datos y el diagrama de clases y de paquetes para guiar la implementación del SIG.

Capítulo 3: Validación de solución

En el capítulo se realizó el modelo de implementación del Sistemas de Información Geográfica para la descripción de la estructura física del sistema y la relación entre sus elementos. Se describieron las pruebas realizadas para comprobar las funcionalidades del software en los diferentes escenarios, para verificar que los resultados sean los esperados.

1 - FUNDAMENTOS TEÓRICOS

1.1 - Introducción

En este capítulo, se presentarán los fundamentos teóricos que sustentan el desarrollo de un Sistema De Información Geográfica (SIG) en la Empresa Filial de Investigaciones Aplicadas Cienfuegos (EFIA Cienfuegos) en Cuba. El objetivo de este capítulo es proporcionar una base sólida de conocimientos teóricos necesarios para comprender la importancia y funcionalidad de un SIG, así como los desafíos y oportunidades que implica su implementación. Al adentrarnos en el dominio del problema, podremos comprender mejor el contexto en el que se desarrolla el proyecto y analizar críticamente el flujo actual de los procesos involucrados.

Además, se realizará un análisis comparativo de los sistemas existentes en el país y en el extranjero, o en la propia entidad, que estén relacionados con el objeto de estudio de esta tesis. Esto permitirá identificar las fortalezas y debilidades de esas soluciones existentes, y destacar cómo la propuesta se diferencia y supera a las alternativas disponibles. Asimismo, se abordarán las metodologías, tecnologías actuales y arquitectura de software que respaldan la implementación del SIG. Estar al tanto de las tendencias y enfoques innovadores en este campo permitirá tomar decisiones informadas y utilizar las herramientas más adecuadas para desarrollar la propuesta.

1.2 - Descripción del dominio del problema

1.2.1 - Conceptos asociados al dominio del problema

Sistema de Información Geográfica

Es una herramienta que integra datos geográficos y no geográficos para su captura, almacenamiento, análisis y visualización en un contexto espacial. Los SIG son fundamentales para tomar decisiones basadas en información geoespacial, ya que permiten gestionar y analizar datos relacionados con ubicaciones geográficas. Facilitan

el análisis de patrones espaciales⁴, la superposición de capas⁵ y la visualización de datos en forma de mapas, lo que ayuda a comprender mejor el entorno y tomar decisiones informadas en diversos campos. [7]

Componentes de un SIG

- **Datos:** La materia prima del SIG, que pueden provenir de diversas fuentes como sensores remotos, GPS, fotografías aéreas, archivos CAD, etc.
- **Software:** Programas informáticos especializados para gestionar y analizar los datos geográficos.
- **Hardware:** Equipos como computadoras, servidores, dispositivos de entrada/salida, etc. que soportan el funcionamiento del SIG.
- **Personas:** Los usuarios, analistas y administradores que operan el sistema.
- **Métodos:** Procedimientos y protocolos para capturar, almacenar, analizar y presentar la información geográfica.

Funciones y Objetivos de un SIG

- Integrar datos de diferentes fuentes y formatos en una base de datos geográfica.
- Analizar las relaciones espaciales entre los fenómenos representados en los datos.
- Generar mapas, escenas 3D y otros productos de información para apoyar la toma de decisiones.
- Resolver problemas complejos relacionados con la gestión del territorio y los recursos.

Datos geográficos

Los datos geográficos en un SIG se refieren a la información relacionada con ubicaciones espaciales en la Tierra, como coordenadas, límites de áreas y topografía. Son fundamentales para el funcionamiento efectivo de un SIG, ya que permiten representar y

⁴ **patrones espaciales:** se refieren a la disposición de objetos en el espacio físico y cómo se relacionan entre sí.

⁵ **superposición de capas:** combinación de geometrías y atributos de varias capas en una sola capa resultante.

analizar información en un contexto espacial. Los datos geográficos precisos y actualizados son esenciales para evitar errores y tomar decisiones informadas. [8]

Tipos de datos geográficos en un SIG

- Datos vectoriales: Representan objetos geográficos mediante puntos, líneas y polígonos. Permiten analizar propiedades topológicas⁶.
- Datos raster: Representan el espacio mediante una malla de celdas o píxeles, como imágenes, fotografías aéreas y modelos digitales de elevación.

Fuentes de datos geográficos

- Sensores remotos (satélites, drones, aviones)
- Sistemas de Posicionamiento Global (GPS)
- Fotografías aéreas
- Archivos digitales (shapefiles⁷, CAD⁸, hojas de cálculo)

Organización de los datos geográficos

- Los datos se organizan en capas temáticas (parcelas, carreteras, edificios, etc.) que se pueden activar y desactivar para su análisis.
- Cada capa contiene una colección homogénea de entidades geográficas (puntos, líneas, polígonos) con atributos descriptivos asociados.

Importancia de los datos geográficos

- Son la materia prima fundamental para construir mapas y realizar análisis espaciales en un SIG.
- Permiten integrar y relacionar información de diversas fuentes para obtener inteligencia accionable.

⁶ **propiedades topológicas:** son características geométricas de los objetos geográficos que describen cómo se relacionan entre sí en el espacio.

⁷ **shapefiles:** son un formato de archivo utilizado para almacenar datos geoespaciales en sistemas de información geográfica.

⁸ **archivo CAD:** es un tipo de archivo utilizado en el diseño asistido por computadora (CAD, por sus siglas en inglés). Estos archivos son fundamentales para la creación de dibujos técnicos, planos arquitectónicos y otros tipos de representaciones gráficas.

- Son esenciales para que un SIG sea útil y pueda resolver problemas complejos relacionados con la gestión del territorio y los recursos.

Análisis espacial

El análisis espacial en un SIG es el proceso de examinar y comprender los patrones y relaciones geoespaciales en los datos. Se utilizan técnicas como la superposición de capas, la interpolación y el análisis de proximidad. Estas herramientas permiten identificar coincidencias, estimar valores entre puntos y analizar la conectividad en una red. El análisis espacial es relevante para comprender el entorno geoespacial y tomar decisiones informadas. [9]

Tipos de análisis espacial

- **Ubicación:** Determinar "¿Qué hay allí?"
- **Condición:** Responder "¿Dónde puedo encontrar estos elementos?"
- **Patrones:** Analizar "¿Cómo están relacionados estos elementos en aquella zona?"
- **Tendencias:** Evaluar "¿Cómo ha cambiado este elemento a lo largo del tiempo?"
- **Modelamiento:** Predecir "¿Qué pasaría si...?"

Funciones de análisis espacial

- **Reclasificación:** Asignar nuevos valores a categorías existentes en una capa
- **Distancia y proximidad:** Calcular distancias euclidianas entre elementos
- **Sobreposición:** Combinar varias capas para generar nueva información

Proceso de análisis espacial

1. **Exploración:** Trabajar con datos espaciales existentes o geocodificar nuevos datos.
2. **Modelado:** Calcular valores y formas, como mapas de isócronas.
3. **Comparación:** Procesar múltiples capas para determinar áreas de superposición o límite.
4. **Predicción:** Detectar patrones y tendencias a lo largo del tiempo para generar mapas predictivos.

Cartografía digital

La cartografía digital se refiere a la representación gráfica de datos geográficos en forma de mapas digitales. Está relacionada con los SIG y ofrece ventajas en términos de visualización interactiva, manipulación y análisis de datos, actualización en tiempo real, almacenamiento eficiente y posibilidad de compartir y colaborar. Permite codificar y gestionar datos geoespaciales utilizando bases de datos, aplicaciones estadísticas, programas de diseño y teledetección. [10]

Ventajas de la cartografía digital

- Permite actualizar los mapas de forma más rápida y frecuente que la cartografía analógica en papel.
- Ofrece mayor nivel de detalle y precisión, evitando la generalización de información.
- Facilita la combinación e integración de múltiples capas de información geográfica.
- Posibilita el análisis espacial y la toma de decisiones basada en datos georreferenciados.

Aplicaciones de la cartografía digital

- Planificación territorial y urbanística.
- Análisis de recursos naturales y medioambiente.
- Gestión de infraestructuras y redes.
- Navegación y logística.
- Agricultura de precisión.

Capa temática

Se refiere a una capa de información geoespacial específica que representa un tema o fenómeno geográfico. Cada capa temática contiene entidades geográficas relacionadas con ese tema, como puntos, líneas o polígonos. Las capas temáticas permiten la representación y análisis de fenómenos geográficos específicos y facilitan la visualización y el estudio de relaciones espaciales. [11]

Organización de la información en capas

- En un SIG, la información geográfica se divide y organiza en múltiples **capas temáticas**.
- Cada capa contiene un tipo de **información** específico, como puntos, líneas o polígonos, con sus atributos descriptivos.
- Las capas se pueden **activar**, **desactivar** y **combinar** según las necesidades del análisis.

Ventajas de las capas temáticas

- Permiten una estructura más organizada y atomizada de los datos geográficos.
- Evitan la redundancia de información al separar los datos por temáticas.
- Facilitan la integración y el análisis de información de diversas fuentes.
- Posibilitan la visualización y el análisis de relaciones espaciales entre diferentes variables.

Ejemplos de capas temáticas

- Red vial (carreteras, calles, caminos)
- Hidrografía (ríos, lagos, embalses)
- Usos del suelo (agrícola, forestal, urbano)
- Topografía (curvas de nivel, elevaciones)
- Infraestructuras (redes eléctricas, telecomunicaciones)
- Límites administrativos (municipios, provincias, países)

Sistemas de Referencia de Coordenadas (SRC)

Marco utilizado para definir, representar y medir la ubicación espacial de entidades en la superficie terrestre. El sistema de coordenadas es un método estandarizado de referenciar e integrar datos espaciales de varias fuentes y garantiza relaciones espaciales precisas entre entidades. Incluye un punto de referencia (origen), ejes de coordenadas (como latitud y longitud o rumbo este y distancia ecuatorial) y unidades de medida (como grados, metros o pies). Los sistemas de coordenadas comunes que se utilizan en SIG son los sistemas de coordenadas geográficas, proyectadas y verticales.

Proyecciones Cartográficas

Las proyecciones cartográficas son técnicas utilizadas para representar la superficie esférica de la Tierra en un formato plano. Existen diferentes familias de proyecciones, incluyendo:

- **Proyecciones cilíndricas**
- **Proyecciones cónicas**
- **Proyecciones planas**

Cada tipo de proyección tiene ventajas y desventajas, y la elección de la proyección adecuada depende de la escala del mapa y de los objetivos específicos del análisis.

Precisión y Distorsiones

Las proyecciones cartográficas introducen distorsiones en conformidad angular, distancia y área. Es imposible preservar todas las propiedades en una sola proyección, por lo que se debe seleccionar una proyección que minimice las distorsiones relevantes para el tipo de análisis que se va a realizar.

Tipos de SRC

Los SRC se dividen en dos categorías principales:

1. **Sistemas de Referencia de Coordenadas Geográficas:** Utilizan grados de latitud y longitud. El sistema más común es el WGS 84, que define ubicaciones en la Tierra mediante coordenadas esféricas.
2. **Sistemas de Referencia de Coordenadas Projectados:** Definidos en un plano bidimensional, estos sistemas utilizan ejes X e Y. Un ejemplo destacado es el sistema Universal Transverse Mercator (UTM), que divide el mundo en zonas para minimizar distorsiones.

Uso del UTM

El sistema UTM es ampliamente utilizado para representar áreas en todo el mundo, dividiendo el planeta en 60 zonas de 6 grados de ancho. Cada zona tiene su propio

sistema de coordenadas que permite una representación precisa de ubicaciones en la superficie terrestre.

Búfer

Los búfers son herramienta de análisis espacial que se utiliza para crear una zona alrededor de una entidad geográfica, como un punto, una línea o un polígono. Esta zona se define a partir de una distancia específica desde la entidad original. [12]

Tipos

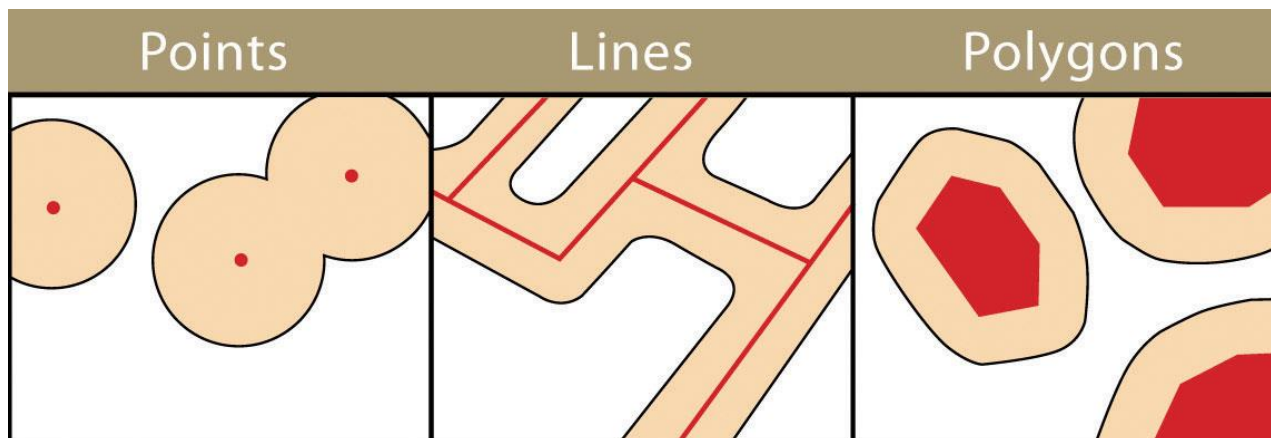
- Búfer simple: Una única zona alrededor de la entidad.
- Búfer múltiple: Varias zonas concéntricas a diferentes distancias.
- Búfer interior y exterior: Zonas tanto dentro como fuera de una distancia específica

Importancia

Los búfers son muy útiles para realizar estudios de antecedentes y análisis de impacto en áreas de incidencia. Al crear un búfer alrededor de una entidad geográfica, se puede identificar y analizar todas las obras, infraestructuras, o elementos que caen dentro de esa zona.

Aplicaciones Comunes

- Evaluación de Impacto Ambiental (EIA): Determinar cómo una nueva construcción afectará el entorno natural y las infraestructuras existentes.
- Planificación Urbana: Analizar cómo una nueva carretera o edificio afectará a las áreas circundantes.
- Gestión de Recursos: Evaluar la distribución de recursos naturales y cómo se verán afectados por nuevas obras.



1 Búfers a puntos, líneas y polígonos

Como observamos en la imagen 1 [13], los búfers pueden realizarse a tres tipos de geometrías, las puntuales, las lineales y las poligonales.

Aplicaciones y campos de uso de los SIG

Algunas de las aplicaciones de los SIG son: [14]

Planificación Urbana y Ordenación del Territorio

- Gestión de datos catastrales y de propiedad
- Planificación de infraestructuras y servicios públicos
- Análisis de crecimiento y desarrollo urbano
- Gestión de patrimonio cultural y natural

Gestión de Recursos Naturales

- Análisis de cuencas hidrográficas y gestión del agua
- Monitoreo y conservación de ecosistemas
- Gestión forestal y agrícola
- Evaluación de riesgos naturales

⁹ Imagen tomada de <https://openpress.usask.ca/introgeomatics/part/introduction-to-analysis/>

Gestión de Emergencias y Desastres

- Mapeo de riesgos y vulnerabilidades
- Planificación y respuesta ante emergencias
- Evaluación de daños y coordinación de ayuda

Transporte y Logística

- Planificación y optimización de rutas de transporte
- Análisis de accesibilidad y conectividad
- Gestión de flotas y seguimiento de vehículos

Salud Pública

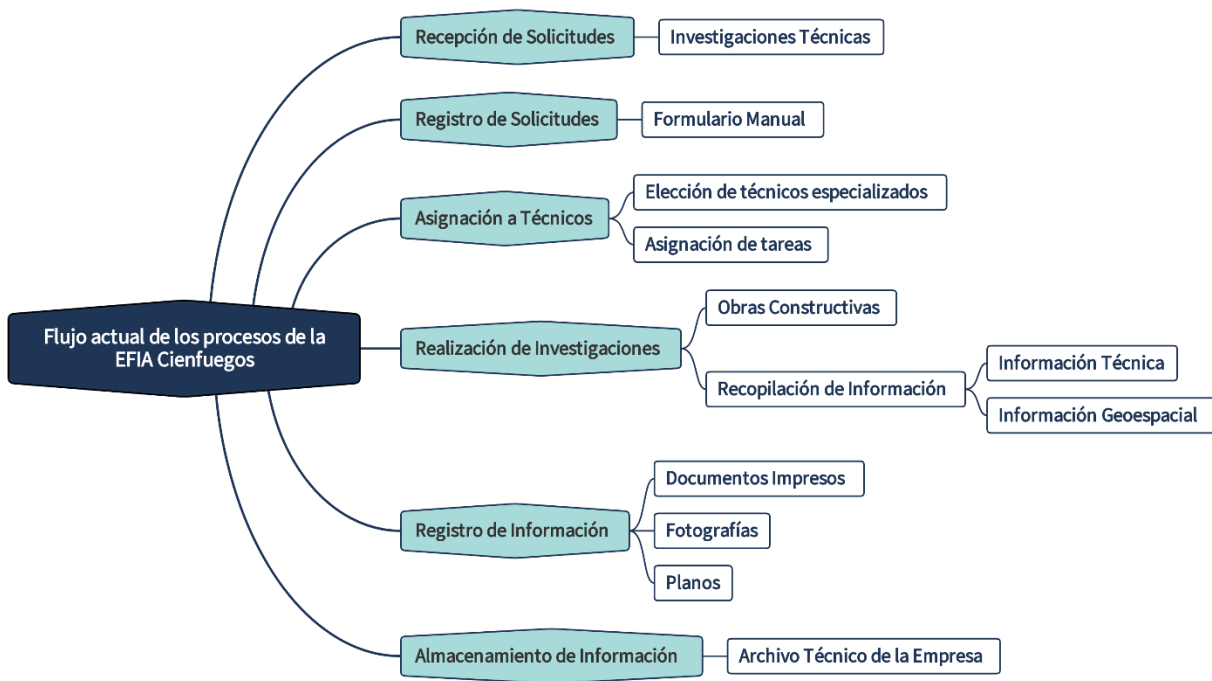
- Análisis de patrones de enfermedades
- Planificación de servicios de salud
- Seguimiento de brotes epidemiológicos

Agricultura de Precisión

- Monitoreo de cultivos y rendimiento
- Optimización del uso de insumos
- Análisis de aptitud de tierras

1.3 - Flujo actual de los procesos y análisis crítico de la ejecución de estos

1.3.1 - Descripción del flujo actual de los procesos:



2 Flujo actual de los procesos de la EFIA Cienfuegos

Debido a la falta de un sistema centralizado y actualizado de gestión de la información, el archivo técnico de la EFIA Cienfuegos se encuentra en un estado de deterioro. Los documentos se encuentran duplicados y algunos se han perdido con el tiempo. Esto dificulta la recuperación y actualización de la información, lo que afecta negativamente el desempeño de los técnicos y la toma de decisiones por parte de los directivos de la entidad.

1.3.2 - Análisis crítico de la ejecución actual de los procesos

El análisis crítico de cómo se ejecutan actualmente los procesos revela diversas deficiencias. Entre las principales **causas** de esta situación problemática se pueden identificar las siguientes:

- **Falta de un sistema de gestión de la información:** La ausencia de un sistema centralizado y actualizado para gestionar la información técnica y geoespacial dificulta la organización, recuperación y actualización de los datos en el archivo técnico.
- **Uso de métodos manuales:** El registro manual de las solicitudes de investigaciones y la documentación de los resultados de las investigaciones en diferentes formatos implica un mayor riesgo de errores, pérdida de información y duplicación de esfuerzos.
- **Mal estado del archivo técnico:** La falta de un sistema adecuado de almacenamiento y organización de la información ha llevado a que los documentos se encuentren en mal estado, lo que dificulta su acceso y consulta.

Las **consecuencias** derivadas de esta situación problemática son las siguientes:

- **Dificultades en la recuperación de la información:** La falta de centralización de la información dificulta la localización y recuperación de los datos necesarios para llevar a cabo las investigaciones técnicas.
- **Pérdida de datos:** Debido a la falta de un sistema de gestión adecuado, se han producido casos de pérdida de información, lo que implica la pérdida de conocimiento y experiencia acumulados en la empresa.
- **Retraso en la toma de decisiones:** La dificultad para acceder y analizar la información de manera rápida y efectiva retrasa la toma de decisiones por parte de los directivos de la EFIA Cienfuegos, lo que puede tener un impacto negativo en la eficiencia de las obras constructivas.

1.4 - Descripción de los sistemas existentes

A continuación, se detallan algunos sistemas geográficos existentes en el mundo y en Cuba y los diferentes softwares que se pueden utilizar para desarrollar un SIG.

1.4.1 - Antecedentes Internacionales

- Sistema de Información Geográfica con interfaz web, para la coordinación del seguimiento a la calidad educativa en la dirección departamental de educación San Vicente. 2012, Universidad de El Salvador. Este SIG en escenario web no sigue la línea de esta investigación. [15]
- Integración de capas de información geográfica en QGIS y desarrollo e implementación de funcionalidades para la planificación de rutas. 2019, Universidad Politécnica de Cartagena. [16]
- GISWater es un software de ingeniería hidráulica desarrollado por la Asociación española GISWater. La empresa es impulsada por el conocimiento y los desarrolladores de GISWater utilizan tecnología de código abierto¹⁰. Además, la empresa colabora con cofundadores y contribuyentes de todo el mundo a través de GitHub, donde se puede aportar el código fuente de GISWater. Este es un SIG que se apoya en la realización de un plugin para el tratamiento hidráulico. [17]
- Tracenet: Plugin en QGIS para Trazado de Líneas con Dirección de Flujo a Partir de Puntos Georreferenciados: Automatiza el trazado de líneas a partir de puntos georreferenciados. Se encuentra programado en PyQGIS para QGIS, con el cual se busca adicionalmente que el usuario pueda reducir los tiempos de procesamiento y aumentar el costo/beneficio. [18]
- MMQGIS desarrollado por Michael Minn, profesor asistente en el Departamento de Geografía y Ciencias de la Información Geográfica de la Universidad de Illinois un conjunto de complementos de Python para manipular capas de mapas vectoriales en Quantum GIS: entrada/salida/unión de CSV, geocodificación, conversión de geometría, almacenamiento en búfer, análisis de ejes, simplificación, modificación de columnas y animación simple. MMQGIS proporciona informes de progreso detallados, una interfaz de usuario intuitiva, acceso directo a archivos y algunas capacidades adicionales que faltan en otros conjuntos de complementos, como la caja de herramientas de procesamiento. [19]

¹⁰ **código abierto:** se refiere a un modelo de desarrollo de software en el cual el código fuente es accesible al público, permitiendo a los usuarios ver, modificar y redistribuir el software sin restricciones.

Todos estos Sistemas y herramientas estudiadas con tecnología de software libre han aportado a la investigación objeto de estudio para todo su desarrollo, lo que como son específicos de un tema o actividad no se utilizaron en este trabajo.

1.4.2 - Antecedentes Nacionales:

Se investigó el desarrollo de SIG por la Universidad de Cienfuegos y otras entidades, encontrándose algunas tesis y artículos, todos del 2009 y realizados en el software MapInfo Professional como los relacionados a continuación:

- Diseño participativo de un sistema de información geográfica para la optimización del uso de suelos en la agricultura urbana.

Sistemas desarrollados por la Empresa GeoCuba Cienfuegos en el 2009 en el software ArcView:

- Sistema de Información Geográfica para la distribución de productos farmacéuticos.
- Sistema de Información Geográfica para el cálculo de las Fajas Forestales Hidrorreguladoras.

Estos Sistemas se desarrollaron en software propietario que ya constituyen tecnología obsoleta, sin embargo, ha sido muy importante su estudio para la metodología a seguir en el SIG de esta investigación y marcan pautas para los temas relacionados con la personalización en el desarrollo de un SIG.

1.4.3 - Softwares utilizados en el desarrollo de SIG's:

Hay varios softwares especializados en Sistemas de Información Geográfica. Entre ellos se encuentran:

- **ArcGIS Pro:** Desarrollado por ESRI, ArcGIS Pro es una plataforma de SIG empresarial. Ofrece una amplia gama de herramientas para el análisis y visualización de datos geospaciales. La integración con otros productos de ESRI, como ArcMap y la extensa lista de extensiones especializadas, lo convierten en una herramienta poderosa para el análisis espacial. [20]

- **GRASS GIS:** es un SIG de código abierto que proporciona herramientas avanzadas para el procesamiento de datos geoespaciales y el análisis cuantitativo. Es altamente modular y ofrece soporte para diversas proyecciones y sistemas de referencia.
- **MapInfo Pro:** es un software SIG desarrollado por Pitney Bowes. Ofrece numerosas herramientas para la cartografía, análisis espacial y visualización de datos, además de capacidades avanzadas para la gestión de activos y datos empresariales.
- **Global Mapper:** es una solución SIG robusta que ofrece un conjunto de herramientas para el procesamiento de datos geoespaciales en 3D, cartografía y análisis espacial. Además de ser un SIG, también es conocido por su compatibilidad con una amplia gama de formatos de datos.
- **MapTiler:** es una plataforma SIG que permite la visualización y el procesamiento de datos cartográficos en línea. Ofrece herramientas para convertir datos geoespaciales en mapas basados en la web, así como para la integración y personalización de mapas interactivos. Comparación: - QGIS es una plataforma SIG de código abierto con una amplia comunidad de usuarios y desarrolladores. Ofrece capacidades avanzadas de análisis y visualización de datos geoespaciales.
- **QGIS:** es un SIG de software libre y de código abierto. Cuenta con soporte para la extensión espacial de PostgreSQL, PostGIS. Maneja archivos vectoriales Shapefile, ArcInfo coverages, MapInfo, GRASS GIS, DXF, etc. Tiene soporte para un importante número de tipos de archivos ráster (GRASS GIS, GeoTIFF, TIFF, JPG, etc.). Permite la creación de complementos con C++ o Python.

1.5 - Metodologías, tecnologías actuales y arquitectura de software

El uso de metodologías de software representa un elemento clave en el desarrollo de software para guiar, organizar, facilitar, informatizar y optimizar un proyecto determinado. Proporciona garantía de calidad y también le permite estimar y gestionar los costos de

desarrollo del proyecto. Al ser un proceso estructurado, también organiza la forma en que se lleva a cabo el proyecto, obligándonos a comprobar la exactitud de los resultados antes de continuar y a fijar objetivos intermedios para gestionar el avance del proyecto. De esta forma se consigue una mayor eficiencia en el uso de los recursos, es decir, se invierte lo mínimo para obtener el máximo a cambio. Existen dos tipos principales de metodologías: ágiles (también llamadas ligeras) y tradicionales (también conocidas como duras). La duración de un proyecto, el pleno aprovechamiento de sus beneficios y la calidad con la que se desarrolla el proceso en cuestión hasta conseguir el producto final dependen en gran medida de la correcta selección, buen uso y habilidad que tenga el equipo de desarrollo sobre los mismos. Las siguientes tecnologías se utilizan en el proceso de desarrollo de software:

1.5.1 - Metodología AUP

El Proceso Unificado Ágil (AUP) es una metodología de desarrollo de software que combina los principios del Proceso Unificado Racional (RUP) con métodos ágiles¹¹ como Scrum y XP. AUP se centra en la simplicidad, la facilidad de uso, la adaptación al cambio y la entrega temprana de software funcional. Su objetivo principal es proporcionar un marco adaptable que permita a los equipos de desarrollo crear software de alta calidad de manera eficiente y rápida.

A diferencia de una metodología rígida y estructurada, AUP utiliza una estrategia iterativa y paso a paso. Los equipos trabajan en ciclos cortos conocidos como iteraciones y al final de cada iteración crean una característica del software. Este enfoque le permite obtener comentarios tempranos de los usuarios y adaptarse a medida que haya nueva información disponible o cambien los requisitos. Al combinar los principios del proceso unificado de Rational con métodos ágiles, AUP proporciona un marco flexible y adaptable. La colaboración entre los miembros del equipo y las partes interesadas es una prioridad, fomentando la comunicación constante y la participación de los usuarios. Además, se

¹¹ **métodos ágiles:** las metodologías ágiles permiten cambiar las prioridades de cada fase del proyecto, según los objetivos y necesidades del cliente, y está orientada a obtener resultados tangibles desde el principio.

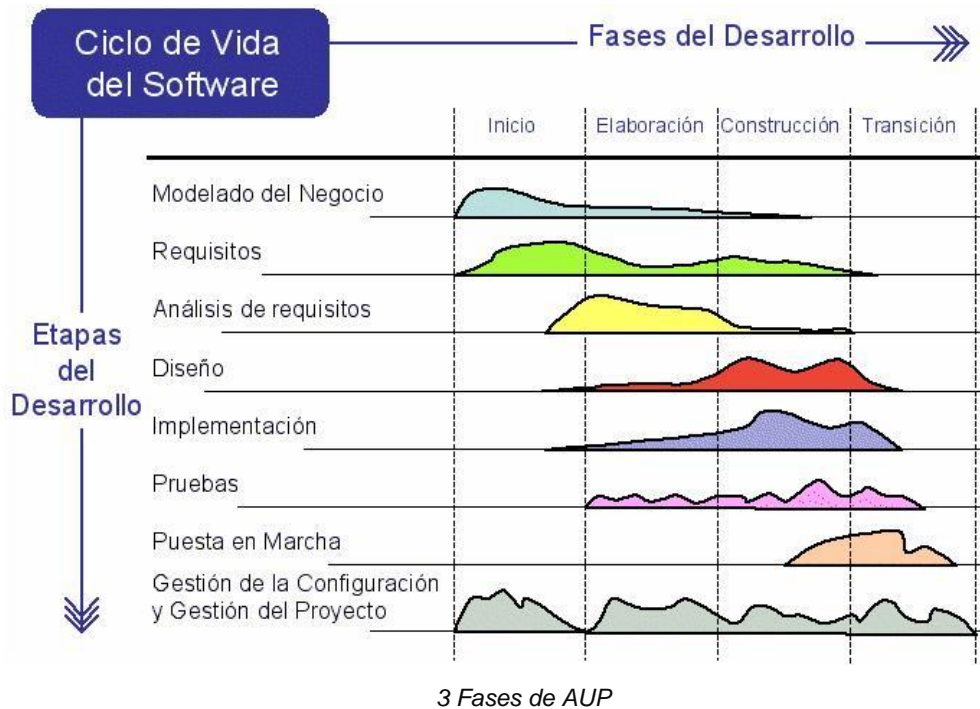
fomenta la entrega continua de software funcional, lo que permite obtener beneficios en las primeras etapas del proyecto. [21]

En el desarrollo de este SIG se utiliza la metodología AUP.

Fases de AUP

AUP se compone de las siguientes fases:

- **Inicio:** En esta fase se definen los objetivos del proyecto, se identifican las partes interesadas y se crea una visión general del sistema. También se realiza un análisis de riesgos y se elabora el plan preliminar de desarrollo.
- **Elaboración:** En esta fase se realiza un análisis y diseño detallado del sistema. Se identifican y priorizan los casos de uso, se definen las arquitecturas y se crea un plan de desarrollo más detallado. Al final de la fase de desarrollo, se espera tener una arquitectura estable y una lista de requisitos claramente definidos.
- **Construcción:** En esta fase el sistema se implementa en sucesivas iteraciones. Se desarrollan casos de uso prioritarios y se realizan pruebas continuas para garantizar la calidad del software.
- **Transición:** En esta fase, el software se implementa en el entorno de producción. Se realizan las pruebas finales, se realiza la formación de los usuarios y se realiza la migración de datos. Al final de esta fase, el software está listo para su uso en producción.



Escenarios de AUP

La metodología Agile Unified Process (AUP) establece cuatro escenarios para modelar sistemas en función de cómo se modela el negocio. Estos escenarios son cruciales para determinar cómo se encapsulan los requisitos del sistema. A continuación, se detallan los escenarios [22]:

- **Escenario 1:** Proyectos que modelen el negocio con Casos de Uso del Negocio (CUN) solo pueden modelar el sistema con Casos de Uso del Sistema (CUS). Este enfoque asegura que los requisitos del negocio se traduzcan directamente en funcionalidades del sistema.
- **Escenario 2:** Proyectos que modelen el negocio con Modelo Conceptual (MC) solo pueden modelar el sistema con Casos de Uso del Sistema (CUS). Similar al escenario anterior, este escenario permite que un modelo conceptual bien definido guíe el desarrollo de los casos de uso del sistema.
- **Escenario 3:** Proyectos que modelen el negocio con Descripción de Proceso de Negocio (DPN) solo pueden modelar el sistema con Descripción de Requisitos por Proceso (DRP). Esto implica que los procesos de negocio definidos guiarán la descripción de los requisitos del sistema de manera estructurada.

- **Escenario 4:** Proyectos que no modelen negocio solo pueden modelar el sistema con Historias de Usuario (HU). Este enfoque es más flexible y permite capturar requisitos de manera más informal y centrada en el usuario, sin un modelo de negocio formal.

Considerando lo mencionado anteriormente sobre los escenarios de la disciplina de requisitos de software en la metodología AUP, se opta por el **escenario número 4**. Esta elección se debe a que el equipo de desarrollo es reducido y mantiene una interacción directa con el cliente.

1.5.2 - Arquitecturas de software

En los inicios de la Ingeniería de Software, el desarrollo de software se realizaba libremente, pero con el tiempo se han ido descubriendo y desarrollando nuevos modelos y estándares, con base a los cuales se puedan resolver las problemáticas modernas. A estos, se les ha denominado arquitectura de software, porque, a semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software.

Una arquitectura de [software], también denominada *arquitectura [lógica]*, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan un marco definido y claro para interactuar con el código fuente del software.

Una arquitectura de software se selecciona y diseña con base en objetivos (requisitos) y restricciones. Los objetivos son aquellos prefijados para el sistema de información, pero no solamente los de tipo funcional, también otros objetivos como el mantenimiento, la auditoría, flexibilidad e interacción con otros sistemas de información. Las restricciones son aquellas limitaciones derivadas de las tecnologías disponibles para implementar sistemas de información. Unas arquitecturas son más recomendables de implementar con ciertas tecnologías mientras que otras tecnologías no son aptas para determinadas arquitecturas. Por ejemplo, no es viable emplear una arquitectura de software de tres capas para implementar sistemas en tiempo real.[23]

Arquitecturas más comunes:

- Descomposición Modular. Donde el software se estructura en grupos funcionales muy acoplados.
- Cliente-servidor. Donde el software reparte su carga de cómputo en dos partes independientes, pero sin reparto claro de funciones.
- Arquitectura de tres niveles. Especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes (o capas) con un reparto claro de funciones: una capa para la presentación (interfaz de usuario), otra para el cálculo (donde se encuentra modelado el negocio) y otra para el almacenamiento (persistencia). Una capa solamente tiene relación con la siguiente.

1.5.3 - Entorno de desarrollo y herramientas utilizadas

El **entorno de desarrollo** usado fue Visual Studio Code en su versión 1.84.2. Visual Studio Code es ideal para el desarrollo del código de un complemento para QGIS debido a su sólido soporte para lenguajes de programación, integración con herramientas esenciales, amplia gama de extensiones, efectiva gestión de proyectos y archivos, así como una comunidad activa y un sólido respaldo de soporte. Esta combinación de características hace que Visual Studio Code sea una herramienta altamente efectiva y productiva para el desarrollo de complementos para QGIS.

Utilizamos la extensión de Python de VSC. [24]

1.5.4 - Lenguajes de programación utilizados

El **lenguaje de programación** usado fue Python en su versión 3.9. Python es la mejor opción para desarrollar un complemento para QGIS debido a su facilidad de aprendizaje, amplio soporte en QGIS, la disponibilidad de bibliotecas y herramientas para análisis espacial, su flexibilidad y versatilidad, así como el respaldo de una comunidad activa y extensa documentación. Estas cualidades lo convierten en un lenguaje ideal para abordar el desarrollo de complementos y herramientas personalizadas en QGIS. [25]

1.5.5 - Otras tecnologías o librerías utilizadas

Librerías:

- **qgis.core** La biblioteca qgis.core de Python es la piedra angular del desarrollo de complementos y scripts para QGIS. Ofrece una amplia gama de clases, módulos y funciones que permiten interactuar con los datos geoespaciales, las capas, los proyectos y la interfaz de usuario de QGIS.
- **PyQt5**: PyQt5 es un conjunto de herramientas de Python que permite el desarrollo rápido de aplicaciones gráficas de usuario (GUI) utilizando la biblioteca Qt. Es ampliamente utilizado debido a su capacidad para crear aplicaciones multiplataforma y su facilidad de uso, especialmente con la herramienta Qt Designer, que permite un diseño visual mediante arrastrar y soltar.

Aplicación de diagramas: draw.io.

Aplicación para crear prototipos: Figma¹².

1.5.6 - Sistema Gestor de Base de Datos

Un sistema gestor de base de datos [26] es un software que permite almacenar, gestionar y manipular grandes cantidades de datos de manera eficiente y segura. Proporciona una interfaz para crear y consultar bases de datos, garantiza la seguridad y la integridad de los datos, y ofrece herramientas para administrar y respaldar los datos de manera confiable.

Algunos ejemplos de SGBD [27] populares son:

- **MySQL**: Sistema de gestión de bases de datos relacional de código abierto, muy utilizado en aplicaciones web.
- **Microsoft SQL Server**: Gestor de bases de datos relacional desarrollado por Microsoft, compatible con Windows y entornos cloud. Se destaca por su seguridad y facilidad de uso.

¹² **Figma**: es una herramienta de diseño colaborativo basada en la nube que permite a los diseñadores crear interfaces de usuario, prototipos y otros elementos visuales

- **PostgreSQL:** SGBD relacional de código abierto, con soporte para múltiples plataformas
- **Oracle Database:** Uno de los SGBD más utilizados, especialmente en entornos empresariales.
- **Microsoft Access:** Sistema gestor de bases de datos relacional, fácil de usar e integrado con Microsoft Office

Las principales funciones y características de un SGBD [28] son:

- Definir la estructura de la base de datos (tipos, estructuras y restricciones de datos).
- Almacenar y gestionar los datos de forma eficiente.
- Permitir el acceso y manipulación de los datos (consultas, actualizaciones, informes).
- Garantizar la seguridad, integridad y consistencia de los datos.
- Controlar los permisos y accesos de los usuarios.
- Proporcionar mecanismos de respaldo y recuperación ante fallos.
- Ofrecer abstracción de la información, ocultando detalles de bajo nivel a los usuarios.

1.5.7 - PostgreSQL

PostgreSQL es un sistema de gestión de **bases de datos relacional**¹³ de código abierto y gratuito. Es altamente escalable, confiable y compatible con los estándares del lenguaje SQL. Proporciona una forma eficiente y segura de almacenar, gestionar y manipular grandes volúmenes de datos.

Características de PostgreSQL:

- **Modelo relacional:** PostgreSQL utiliza el modelo relacional, lo que significa que los datos se organizan en tablas con relaciones definidas entre ellas. Esto facilita la gestión y consulta de los datos de manera estructurada.

¹³ **bases de datos relacional:** bases de datos relacionales son un tipo de sistema de gestión de bases de datos que organizan la información en tablas, las cuales están compuestas por filas y columnas.

- **Extensibilidad:** PostgreSQL es altamente extensible, lo que permite a los usuarios definir nuevos tipos de datos, funciones y operadores. Esto permite adaptar el sistema a necesidades específicas y ampliar sus capacidades.
- **Transacciones ACID:** PostgreSQL garantiza la integridad de los datos mediante el soporte de transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad). Esto asegura que las operaciones se realicen de manera confiable y que se mantenga la consistencia de los datos.
- **Alta disponibilidad y tolerancia a fallos:** PostgreSQL ofrece mecanismos de replicación para lograr alta disponibilidad y tolerancia a fallos. Esto permite configurar réplicas de la base de datos para asegurar la continuidad del servicio en caso de fallas en el sistema principal.
- **Compatibilidad con estándares:** PostgreSQL es compatible con los estándares SQL y soporta numerosas características avanzadas, como consultas complejas, vistas, funciones almacenadas y disparadores. Esto facilita la migración de aplicaciones y la interoperabilidad con otros sistemas. [29]

Ventajas de PostgreSQL:

- **Licencia de código abierto:** PostgreSQL es un sistema de código abierto, lo que significa que es gratuito y se puede modificar y distribuir libremente. Esto reduce los costos de licenciamiento y permite una mayor flexibilidad en su uso.
- **Escalabilidad:** PostgreSQL es altamente escalable y puede manejar grandes volúmenes de datos y cargas de trabajo intensivas. Puede soportar sistemas con múltiples usuarios concurrentes y bases de datos de gran tamaño.
- **Estabilidad y confiabilidad:** PostgreSQL es conocido por su estabilidad y confiabilidad. Es utilizado en entornos de producción de misión crítica debido a su capacidad para mantener la integridad de los datos y mantener un alto rendimiento incluso en situaciones de alta carga.
- **Comunidad activa:** PostgreSQL cuenta con una comunidad de desarrolladores y usuarios muy activa y comprometida. Esto se traduce en un amplio soporte, actualizaciones regulares, corrección de errores y desarrollo continuo de nuevas características.

- **Amplia gama de extensiones y complementos:** ofrece una amplia gama de extensiones y complementos desarrollados por la comunidad. Estas extensiones proporcionan funcionalidades adicionales, como soporte para datos geoespaciales, indexación de texto completo y análisis avanzado, lo que amplía aún más las capacidades del sistema.

1.6 - Conclusiones parciales

En este capítulo se presentaron los fundamentos teóricos necesarios para comprender el contexto y los desafíos relacionados con la implementación de un Sistema de Información Geográfica (SIG) en la Empresa Filial de Investigaciones Aplicadas Cienfuegos (EFIA Cienfuegos). Las principales conclusiones son las siguientes:

- Se identificaron las dificultades derivadas de la dispersión y desorganización de la información técnica en la empresa.
- Se realizó un análisis crítico del flujo de los procesos actuales, destacando deficiencias en la recuperación y actualización de datos espaciales.
- Se constató la falta de un sistema unificado y centralizado para el manejo eficiente de la información geográfica.
- Se exploraron metodologías, tecnologías actuales y arquitectura de software relevantes para la implementación del SIG.

En base a esto, se puede afirmar que la implementación de un SIG, que integre los datos geográficos de manera centralizada y actualizada constituye una gran utilidad para EFIA Cienfuegos. Este sistema permite mejorar la recuperación, conservación y actualización de los datos, así como agilizar la toma de decisiones en la entidad.

2 - PROPUESTA DE SOLUCIÓN

2.1 - Introducción

En este capítulo, se presenta la propuesta de solución que tiene como objetivo desarrollar un SIG para la EFIA Cienfuegos. El SIG propuesto busca centralizar y mantener actualizada la información técnica en un entorno espacial, permitiendo una consulta y evaluación más eficiente de los datos. Se aborda la concepción general del sistema, los requisitos funcionales y no funcionales, así como el análisis y diseño de la solución propuesta. Se presentan el diagrama del modelo físico de datos, el diagrama de clases y de componentes. Se presentan los requisitos funcionales, las historias de usuarios y los prototipos de interfaz.

2.2 - Concepción general del sistema

La **creación de un SIG** puede basarse en tres tipos de **escenarios** para el desarrollo de software [30]:

- 1. Desarrollo de extensiones y complementos:** Este enfoque implica agregar funcionalidades adicionales al software SIG existente en el escritorio. Se desarrollan extensiones y complementos que se integran con el software principal, proporcionando características y herramientas adicionales para el análisis y la visualización de datos geoespaciales.
- 2. Desarrollo de herramientas web de visualización y generación de mapas:** En este escenario, se desarrollan aplicaciones web que permiten a los usuarios visualizar y generar mapas geoespaciales a través de un navegador web. Estas herramientas web pueden ofrecer funcionalidades de visualización interactiva, consulta de datos y análisis espacial, brindando acceso a la información geográfica de forma accesible y fácil de usar.
- 3. Desarrollo de aplicaciones independientes en el escritorio empleando componentes:** En esta opción, se desarrollan aplicaciones de escritorio independientes que utilizan componentes de software SIG para gestionar y analizar datos geoespaciales. Estas aplicaciones pueden aprovechar las

capacidades de los sistemas SIG para realizar análisis espacial, visualizar mapas y realizar tareas específicas en el dominio del problema.

Los escenarios 2 y 3 se utilizan para sistemas en los cuales hay que desarrollar todas sus funcionalidades mientras que en el escenario 1 se desarrollan las funcionalidades propias de la problemática tratada como una extensión de la plataforma SIG, pudiéndose aprovechar todas las facilidades de dicha plataforma. El SIG objeto de nuestra investigación se desarrolló en el **primer escenario** en la plataforma QGIS debido a que va dirigido a **usuarios especializados** de la temática.

2.2.1 - SIG personalizado

Un SIG personalizado se refiere a un SIG que ha sido adaptado o configurado específicamente para satisfacer las necesidades y requisitos de un usuario u organización en particular. Algunas características clave de un SIG personalizado incluyen:

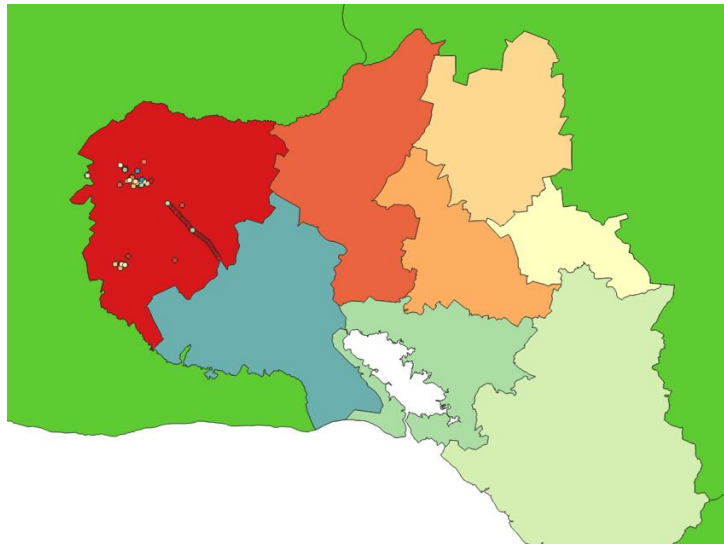
- **Adaptación a las necesidades específicas:** El SIG se diseña y configura para cumplir con los objetivos y requisitos únicos del usuario, como tipos de datos, análisis, visualización, etc.
- **Interfaz y funcionalidad a medida:** La interfaz de usuario y las herramientas del SIG se personalizan para facilitar el uso y la interacción del usuario final.
- **Integración con otros sistemas:** El SIG se integra con otros sistemas de información, bases de datos o aplicaciones que utiliza la organización.
- **Capacidades de análisis y modelado avanzadas:** El SIG puede incluir herramientas y funcionalidades de análisis espacial y modelado más sofisticadas, adaptadas a las necesidades específicas del usuario.

2.2.2 - Capas temáticas del SIG

El SIG desarrollado tiene las siguientes capas temáticas:

Obras y Calas

Dos de las capas trabajadas. En el SIG se representaron las obras y las calas del municipio de Aguada de Pasajeros de la provincia de Cienfuegos.



4 Capas Obras y Calas del municipio Aguada de Pasajeros

Obras: Se refiere a cualquier proyecto de construcción, ya sea de edificios, puentes, carreteras, etc. Incluye todas las actividades necesarias para llevar a cabo el proyecto, desde la planificación y diseño hasta la ejecución y finalización.

Atributos de obras:

- **geom:** Topología del objeto
- **obra:** Nombre de la obra
- **mun_obra:** Hace referencia al municipio en el que está contenida la obra
- **X:** Coordenada X
- **Y:** Coordenada Y
- **situación:** Situación geográfica donde se encuentra la obra
- **fecha:** fecha en que se realizó la obra
- **autor:** autor de la obra
- **etapa:** La etapa de ejecución de la obra
- **area:** Área que abarca la obra en metros cuadrados
- **escala:** La escala de los gráficos de la obra en los documentos
- **perf:** Perforación realizada

- **lab:** Ensayos de laboratorio realizados
- **pruebas:** Pruebas generales, como la de permeabilidad
- **nf:** Nivel freático (límite superior de la zona saturada de agua subterránea)
- **informe:** Ruta del documento informe
- **columna:** Ruta del documento columna
- **perfil:** Ruta del documento perfil
- **plano:** Ruta del documento plano

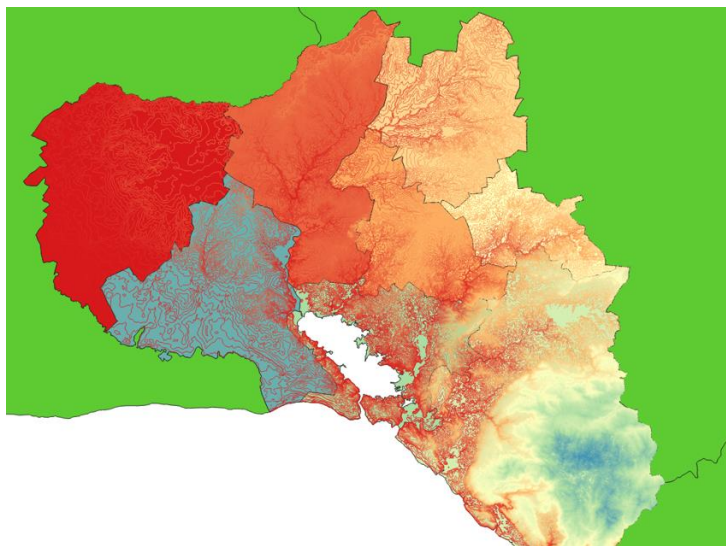
Calas: Son excavaciones realizadas en el terreno para estudiar su composición y estructura. Estas excavaciones permiten conocer las características geológicas del suelo y determinar la profundidad a la que se encuentra un terreno firme para apoyar la fundación de un edificio¹. Las calas son esenciales para asegurar que la construcción se realice sobre una base sólida y segura.

Atributos de calas:

- **geom:** Topología del objeto
- **mun_obra_cala:** Hace referencia al municipio, y a la obra en la que está contenida la cala
- **obra_cala:** Identificador de obra y cala
- **idcala:** Identificador de la cala
- **x:** Coordenada X
- **y:** Coordenada Y
- **z:** Altura en metros
- **nf:** Nivel freático (límite superior de la zona saturada de agua subterránea)
- **cota nf:** Cota del nivel freático
- **prof:** Profundidad de la cala
- **cota_final:** Cota del final de la cala o de profundidad
- **pdf:** Ruta de los PDF asociados a la cala
- **cod_mun:** Código del municipio al que pertenece la cala
- **idobra:** Identificador de la obra a la que pertenece la cala

Relieve

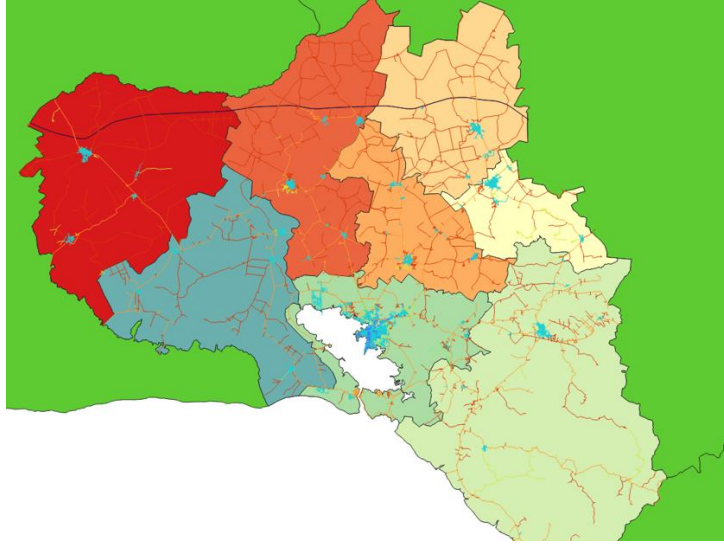
Capa visual que representa el relieve de la provincia Cienfuegos. El relieve terrestre es el término que define a las formas que tiene la corteza terrestre o litosfera en la superficie, tanto en relación con las tierras emergidas como en cuanto al relieve oceánico, es decir, al fondo del mar.



5 Capa Relieve

Viales

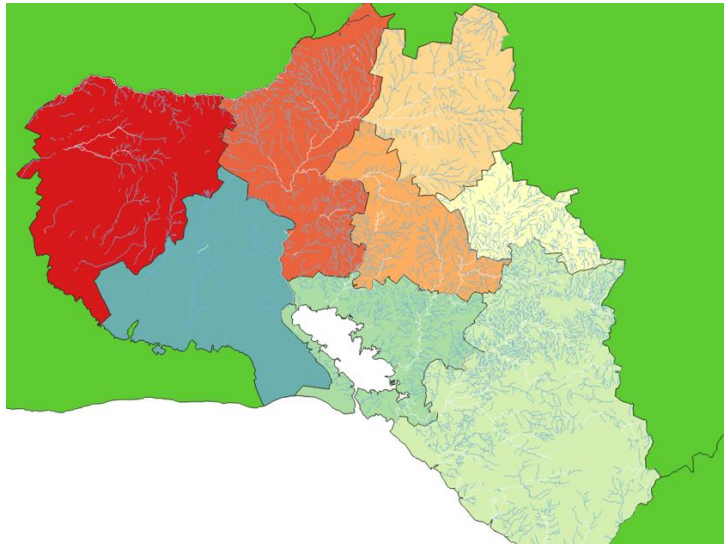
Capa visual que representa los viales de la provincia Cienfuegos. Los viales representan las carreteras, vías y calles.



6 *Capa Viales*

Ríos

Capa visual que representa los ríos de la provincia Cienfuegos.



7 *Capa Ríos*

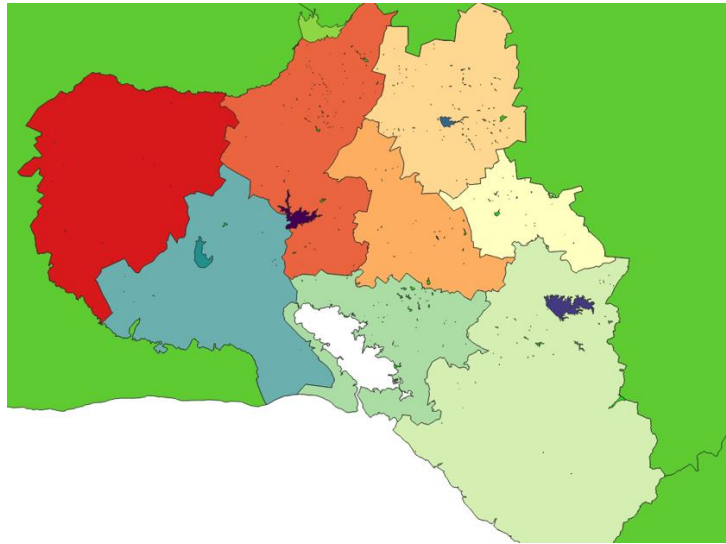
Embalses y microembalses

Capa visual que representa los embalses y microembalses de la provincia Cienfuegos.

Embalses: Son grandes masas de agua creadas por el hombre mediante la construcción de una presa en un río o en la salida de un lago. Se utilizan para almacenar agua y

asegurar su disponibilidad durante todo el año para diversos usos como el riego, el abastecimiento de agua potable, la generación de energía hidroeléctrica y la recreación.

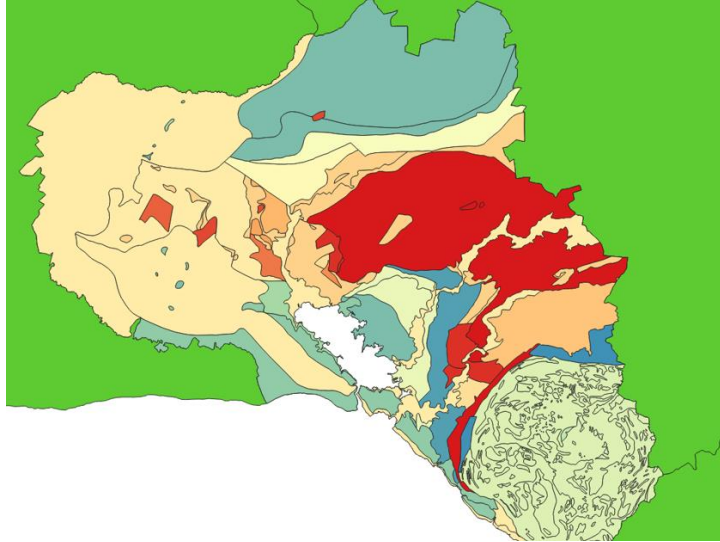
Microembalses: Son embalses de menor tamaño que se utilizan principalmente para el riego agrícola y el abastecimiento de pequeñas comunidades. Su construcción y mantenimiento son más sencillos y económicos en comparación con los grandes embalses.



8 Capa Embalses y Microembalses

Geología

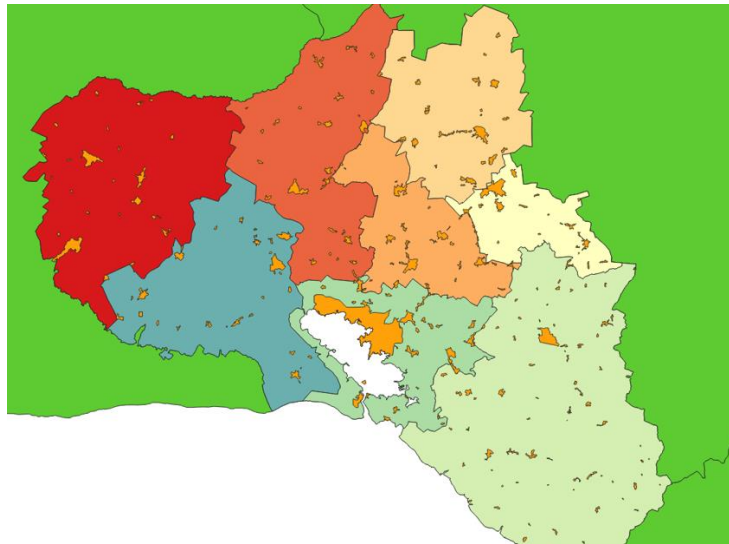
Capa visual que representa la geología de la provincia Cienfuegos. La geología de una localidad se refiere al estudio de la composición, estructura y procesos geológicos que han dado forma a una región específica. Esto incluye el análisis de rocas, minerales, fósiles y formaciones geológicas presentes en el área. La geología de una localidad puede proporcionar información valiosa sobre la historia geológica, los recursos minerales y las condiciones ambientales de la región



9 Capa Geología

Poblados

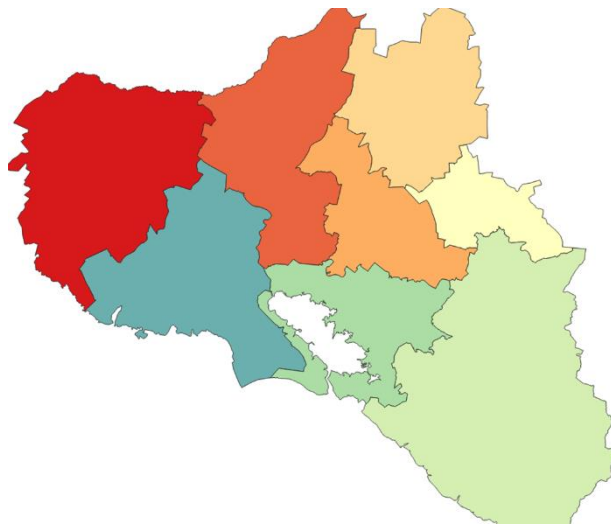
Capa visual que representa los poblados de la provincia Cienfuegos.



10 Capa Poblados

Municipios

Capa visual que representa los municipios de la provincia Cienfuegos. En este caso se trabajó con el municipio de Aguada de Pasajeros. Las obras y las calas representadas anteriormente son las de este municipio.



11 Capa Municipios

2.2.3 - Recopilación y organización de los datos

Estos pasos constituyeron la antesala de la creación del SIG.

1. **Recopilación de información:** Se recopiló toda la información del Archivo Técnico de la EFIA (Copia Dura, Digital).
2. **Tabla Excel:** Se conformaron tablas Excel para el registro de la información.
3. **Escaneo de documentos:** Algunos documentos físicos que tenían alto riesgo de perderse se escanearon en formato PDF.
4. **Validación de datos:** Se validaron los datos de las tablas Excel (Obras y Calas) en la plataforma de QGIS.
5. **Conversión de datos a CSV:** Se convirtieron los datos validados a formato CSV. A fin de que sean compatibles con QGIS todos los caracteres de los datos, se abrieron en Bloc de Notas y salvaron con UTF-8.
6. **Importar Capas Espaciales:** Añade las capas espaciales a QGIS. Estas capas deben contener la información de geometría (coordenadas X y Y).
7. **Visualización en el Mapa:** Verifica que las capas se representen correctamente en el mapa.

8. **Importar Capas Literales:** Añade las capas que solo contienen información literal. Estas no tendrán representación en el mapa, pero son importantes para el análisis.
9. **Organización de Capas:** Asegura que todas las capas estén bien organizadas en el panel de capas de QGIS.
10. **Creación de la base de datos:** Se crea una base de datos, usando PostgreSQL.
11. **Conexión con PostGIS:** Se habilitó en PostgreSQL la extensión PostGIS, con el fin de conectar la base de datos con QGIS de forma espacial..
12. **Importar Capas a PostgreSQL:** Importa las capas espaciales y no espaciales a PostgreSQL desde QGIS.
13. **Refrescar Conexión:** Al refrescar la conexión a PostgreSQL, las tablas y capas importadas se añadirán automáticamente al mapa de QGIS.
14. **Eliminar Capas CSV:** Una vez que los datos están en PostgreSQL, se pueden eliminar las capas CSV de QGIS si ya no son necesarias.
15. **Importar Estructura Vacía:** Si es necesario, puedes importar una estructura vacía desde un CSV a QGIS.
16. **Adición de Datos Nuevos:** Utiliza la opción "Agregar datos" para empezar a añadir datos desde cero en la capa vacía de PostgreSQL.

2.2.4 - Desarrollo de complementos para un SIG

El desarrollo de complementos (plugins) es una parte importante en los Sistemas de Información Geográfica (SIG) como QGIS. Estos complementos permiten extender las funcionalidades del software SIG y adaptarlo a necesidades específicas. Algunos aspectos clave sobre el desarrollo de complementos en SIG [31] :

Lenguajes de programación

Los complementos de QGIS se desarrollan principalmente utilizando el lenguaje Python y la API PyQGIS. Esto permite aprovechar las capacidades de Python para el análisis geoespacial.

Estructura y arquitectura

Los complementos siguen una estructura y arquitectura definida por las pautas de desarrollo de QGIS. Incluyen archivos de metadatos, código Python principal, recursos, traducciones, etc.

Integración con bases de datos

Los complementos pueden integrarse con bases de datos geográficas como PostgreSQL/PostGIS para almacenar y gestionar datos espaciales. Esto permite crear aplicaciones SIG completas.

Ejemplos de complementos

Existen complementos para inventario urbano que permiten gestionar elementos como árboles, mobiliario, etc. a través de una interfaz intuitiva. También se desarrollan complementos para consumir servicios web geoespaciales estándares como WPS (Web Processing Service).

Beneficios

- El desarrollo de complementos permite adaptar los SIG a necesidades específicas de usuarios y organizaciones.
- Facilita la gestión de datos geográficos complejos a través de interfaces amigables.
- Aprovecha las capacidades de análisis espacial avanzado de los SIG.

Estructura general de los complementos

1. Código fuente

- El código fuente es el corazón del complemento, escrito en Python.
- Contiene las clases y funciones que implementan la lógica del complemento.

2. Metadatos

- Los metadatos describen el complemento, como su nombre, autor, versión, etc.
- Se almacenan en un archivo de metadatos como metadata.txt.

3. Interfaz de usuario

- El complemento puede tener una interfaz de usuario para interactuar con el usuario.
- Se construye usando Qt Designer y se integra en el código Python.

4. Recursos

- Los recursos son archivos adicionales utilizados por el complemento, como íconos, imágenes, etc.
- Se referencian en el código y se incluyen en el paquete del complemento.

5. Archivo de proyecto

- El archivo de proyecto (.qgis2 o .qgz) contiene la configuración del proyecto QGIS.
- Puede incluir datos específicos del complemento, como configuraciones o resultados.

Un complemento típico tiene la siguiente estructura de directorios [32]:

```
complemento/  
├─ complemento.py  
├─ __init__.py  
├─ metadata.txt  
├─ resources.qrc  
├─ ui  
│   └─ complemento_dialog_base.ui  
└─ resources  
    └─ iconos  
        └─ icono.png
```

12 Estructura general de un plugin de QGIS

- `complemento.py`: contiene la lógica principal del complemento
- `__init__.py`: inicializa el complemento como un paquete Python
- `metadata.txt`: archivo de metadatos del complemento

- resources.qrc: archivo de recursos que referencia archivos adicionales
- ui/: directorio que contiene archivos de interfaz de usuario Qt
- resources/: directorio que contiene recursos estáticos como iconos

Es importante mencionar que esta es una estructura general y los plugins de QGIS pueden variar en su estructura dependiendo de los requisitos y funcionalidades específicas que se deseen implementar. Más adelante se menciona de manera específica la estructura de los complementos desarrollados.

En este caso, se ha decidido enfocarse en la vertiente del desarrollo de extensiones y complementos para abordar los objetivos específicos de la EFIA Cienfuegos. Esto implica la creación de algunas herramientas que cumplan con los requisitos establecidos por la entidad.

2.3 - Requisitos

2.3.1 - Requerimientos funcionales

Un requisito funcional de software es una especificación que describe una función o capacidad específica que debe ser cumplida por un sistema de software. Estos requisitos se centran en qué debe hacer el software, es decir, en las acciones que debe llevar a cabo y las operaciones que debe realizar.

RF1. Adicionar en el mapa elementos de la capa Obras

Descripción: El sistema permite que se añadan elementos nuevos a la capa Obras.

RF2. Adicionar en el mapa elementos de la capa Calas

Descripción: El sistema permite que se añadan elementos nuevos a la capa Calas.

RF3. Adicionar en el mapa elementos de la capa Obras mediante coordenadas X Y.

Descripción: El sistema permite que se añadan elementos nuevos a la capa Obras introduciendo sus coordenadas "X" y "Y".

RF4. Adicionar en el mapa elementos de la capa Calas mediante coordenadas X Y.

Descripción: El sistema permite que se añadan elementos nuevos a la capa Calas introduciendo sus coordenadas “X” y “Y”.

RF5. Adicionar elementos a una capa desde otra capa.

Descripción: El sistema permite que se añadan elementos desde una capa origen hacia una capa destino.

RF6. Editar los datos de la capa Obras

Descripción: El sistema permite editar los datos de los elementos de la capa Obras.

RF7. Editar los datos de la capa Calas

Descripción: El sistema permite editar los datos de los elementos de la capa Calas.

RF8. Editar los datos de una capa desde otra capa.

Descripción: El sistema permite editar los datos de una capa desde otra capa.

RF9. Seleccionar elementos de la capa Obras en un área de análisis.

Descripción: El sistema permite seleccionar las Obras que estén contenidas en el área de análisis.

RF10. Exportar datos a ficheros CSV.

Descripción: El sistema permite exportar datos de las capas a formato CSV.

RF11. Realizar consultas literales sobre las capas.

Descripción: El sistema permite realizar consultas literales a las capas.

RF12. Filtrar datos.

Descripción: El sistema permite filtrar los datos que desean mostrar en el mapa.

RF13. Empaquetar el proyecto para QField.

Descripción: El sistema permite crear un paquete que pueda visualizarse en la aplicación móvil QField¹⁴.

RF14. Mostrar la ayuda del sistema.

Descripción: El sistema permite mostrar una ayuda.

2.3.2 - Requerimientos no funcionales

Un requisito no funcional de software es una especificación que describe aspectos o características del sistema que no están relacionados directamente con su funcionalidad, sino más bien con sus propiedades globales, restricciones o atributos de calidad.

1 Tabla de requisitos no funcionales

Nro.	Descripción	Tipo de requisito
RNF1	El plugin tiene de forma clara y explícita todos los campos que han de ser rellenados.	Usabilidad
RNF2	El plugin se ejecuta perfectamente en dispositivos de bajo rendimiento.	Rendimiento
RNF3	El plugin se puede usar en cualquier versión de QGIS 3 y no presenta fallos.	Portabilidad
RNF4	El plugin es confiable, ya que no requiere conexión con internet. Solo se usará en un entorno offline.	Confiability

¹⁴ **QField**: es una aplicación de código abierto diseñada para la recolección de datos geoespaciales en dispositivos móviles.

RNF5	Es necesario tener instalado QGIS en su versión 3 en adelante para poder usar el plugin.	Software
RNF6	El código fuente debe estar documentado y seguir estándares de codificación.	Mantenibilidad
RNF7	El acceso al sistema debe estar protegido por autenticación de usuario y contraseña. Los datos sensibles deben estar encriptados tanto en reposo como en tránsito.	Seguridad

2.3.3 - Historias de usuario

Las historias de usuario son una técnica clave en la metodología Agile Unified Process (AUP) utilizada para la especificación de requisitos funcionales en el desarrollo de software.

Número: 1	Requisito: Adicionar en el mapa elementos de la capa Obras
Programador: Jahdiel López Fiffe	Iteración Asignada: 1
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 80 horas

Descripción:

- El usuario a través de la herramienta de adicionar puntos de QGIS adiciona un nuevo punto en una ubicación del mapa.
- Tras la selección de la ubicación, el sistema invoca un formulario diseñado para la entrada de datos específicamente para la capa temática **Obras**, donde el usuario podrá llenar los campos requeridos.
- Al completar el formulario y confirmar la entrada, el sistema guarda los datos en la capa correspondiente de la base de datos.

Observaciones: N/A**Prototipo de interfaz:**

Obras - Atributos del objeto espacial

Acciones

Id de la Obra.

Nombre de la Obra:

Coordenada X:

Coordenada Y:

Situación:

FECHA

Autor:

Etapa

Area (km 2)

Escala (1:Esc):

Perforación:

Pruebas de Laboratorio. F

Realización de Pruebas:

Nivel Freatico

Informe: NULL

Columna Litologica NULL

Perfil NULL

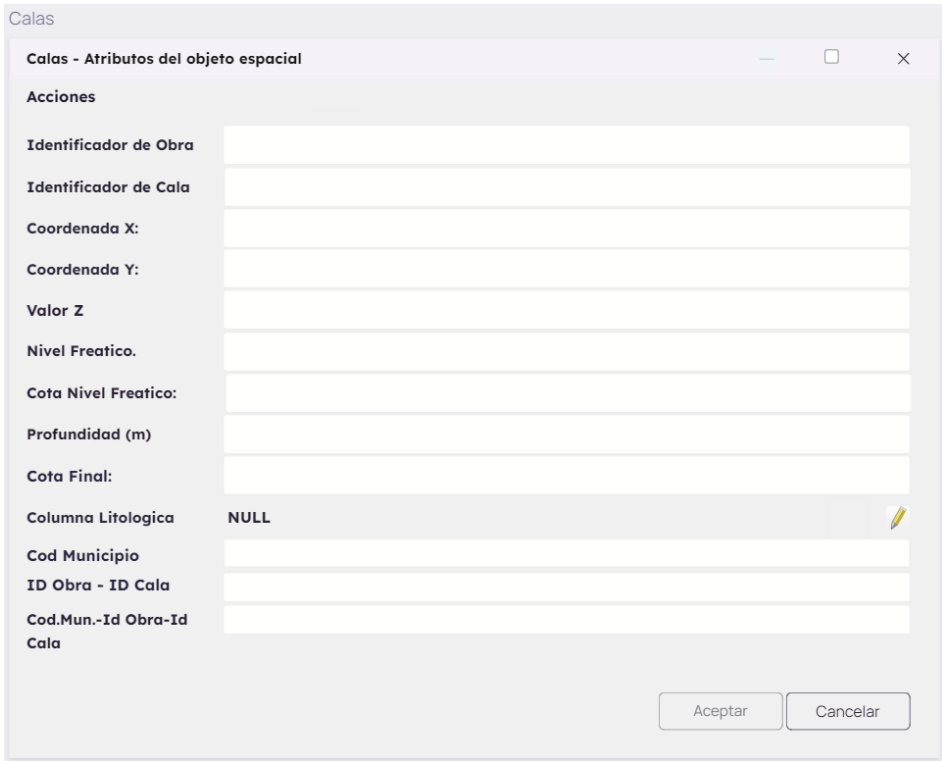
Plano NULL

Codigo Municipio:

CodigoMunicipio - IdObra

Aceptar Cancelar

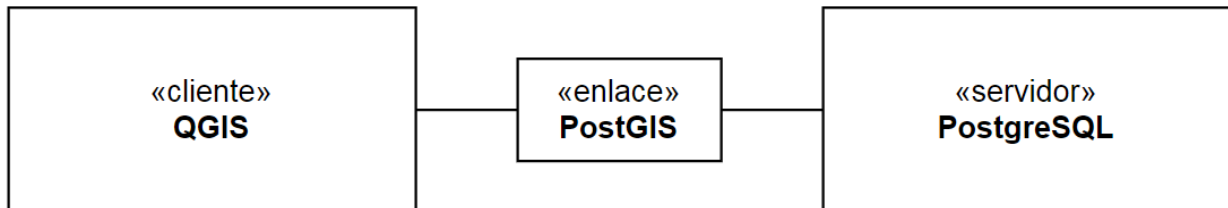
3 Historia de usuario RF2

Número: 2	Requisito: Adicionar en el mapa elementos de la capa Calas	
Programador: Jahdiel López Fiffe		Iteración Asignada: 1
Prioridad: Alta		Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio		Tiempo Real: 80 horas
<p>Descripción:</p> <ul style="list-style-type: none"> - El usuario a través de la herramienta de adicionar puntos de QGIS adiciona un nuevo punto en una ubicación del mapa. - Tras la selección de la ubicación, el sistema invoca un formulario diseñado para la entrada de datos específicamente para la capa temática Calas, donde el usuario podrá llenar los campos requeridos. - Al completar el formulario y confirmar la entrada, el sistema guarda los datos en la capa correspondiente de la base de datos. 		
Observaciones: N/A		
<p>Prototipo de interfaz:</p> 		

2.4 - Análisis y Diseño

2.4.1 - Descripción de la arquitectura del software

En el contexto de un proyecto de desarrollo de complementos para QGIS con Python, la arquitectura de software modelo cliente-servidor se manifiesta de la siguiente manera.



13 Arquitectura Cliente - Servidor

El **servidor** es PostgreSQL, el gestor de base de datos, que contiene toda la información que se le brinda al cliente.

El **punto** entre cliente y servidor es la extensión de QGIS y PostgreSQL, llamada PostGIS: Permite su conexión y que los datos de PostgreSQL se representen de manera espacial en QGIS.

El **cliente** es QGIS, ya que contiene la interfaz de cara al usuario, tiene las funcionalidades para que el usuario interactúe y representa de forma la información brindada por el servidor.

2.4.2 - Patrón de arquitectura Multicapas

El patrón de arquitectura multicapas se presenta como una solución efectiva en el diseño de sistemas de software, especialmente en el contexto de un Sistema de Información Geográfica (SIG) implementado bajo la arquitectura cliente-servidor. Este patrón no debe confundirse con la arquitectura en sí, ya que se refiere a una forma específica de organizar la lógica y las interacciones dentro de la aplicación.

Nexo entre Arquitectura Cliente-Servidor y Patrón Multicapas

La arquitectura **cliente-servidor** establece un marco en el que los clientes (usuarios finales) interactúan con un servidor que centraliza la lógica de negocio y los datos. En este escenario, el patrón de arquitectura multicapas se integra de manera natural, permitiendo una clara separación de responsabilidades y mejorando la escalabilidad y mantenibilidad del sistema.

En el SIG desarrollado para la Empresa Filial de Investigaciones Aplicadas Cienfuegos, se pueden identificar las siguientes capas en el patrón multicapas:

- **Capa de Presentación:** Se encarga de la interfaz de usuario, donde se visualizan los datos geoespaciales y se realizan consultas. Esta capa asegura que los usuarios puedan interactuar de manera intuitiva y eficiente con el sistema.
- **Capa de Lógica de Negocio:** Aquí se implementan las reglas y procesos que gestionan la manipulación de datos. En el contexto de la tesis, esta capa es crucial para la recuperación y actualización de la información técnica, garantizando que las operaciones realizadas sean coherentes y eficientes.
- **Capa de Acceso a Datos:** Esta capa actúa como intermediaria entre la lógica de negocio y la base de datos, facilitando la gestión de datos a través de PostgreSQL y su extensión espacial PostGIS. Su diseño optimiza la interacción con los datos geoespaciales, permitiendo un acceso rápido y eficiente. [33]

Beneficios del Patrón Multicapas en el SIG

La implementación del patrón de arquitectura multicapas en el SIG ofrece varias ventajas:

- **Mantenibilidad:** La separación de las capas permite realizar modificaciones en una sin afectar a las demás, lo que facilita el mantenimiento del sistema.
- **Escalabilidad:** Cada capa puede escalarse de forma independiente, lo que es esencial para manejar un aumento en la cantidad de datos y usuarios.
- **Reusabilidad:** Las capas pueden ser reutilizadas en otros proyectos, lo que optimiza el tiempo de desarrollo y los recursos.
- **Flexibilidad:** Permite la integración de nuevas tecnologías en cualquiera de las capas sin necesidad de rediseñar todo el sistema.

En conclusión, el patrón de arquitectura multicapas se adapta perfectamente a las necesidades del SIG, proporcionando una estructura organizada que mejora la calidad del software y optimiza la gestión de la información técnica. Este enfoque no solo facilita la recuperación y actualización de datos, sino que también potencia la toma de decisiones en la Empresa Filial de Investigaciones Aplicadas Cienfuegos.

2.4.3 - Patrones de diseño

El patrón de diseño GRASP (General Responsibility Assignment Software Patterns) es un conjunto de principios que ayudan a asignar responsabilidades a las clases y objetos en el diseño orientado a objetos. Este patrón es especialmente útil en el contexto de un Sistema de Información Geográfica (SIG) implementado bajo la arquitectura cliente-servidor y el patrón de arquitectura multicapas, ya que promueve un diseño desacoplado y cohesivo.[34]

Principios de GRASP

Se basa en varios principios fundamentales para asignar responsabilidades:

- **Experto:** Una clase debe ser responsable de la información que posee y de las tareas que puede realizar. En el SIG, esto se traduce en que cada capa debe encargarse de las responsabilidades específicas de su dominio, como la presentación, la lógica de negocio o el acceso a datos.
- **Creador:** Una clase debe ser responsable de crear instancias de otras clases. En el SIG, el patrón Creador podría utilizarse para instanciar objetos relacionados con la manipulación de datos geospaciales o la generación de informes.
- **Bajo acoplamiento:** Las clases deben estar lo menos acopladas posible entre sí. Esto se logra asignando responsabilidades de manera que se minimicen las dependencias entre clases. En el SIG, un bajo acoplamiento entre capas permite que los cambios en una capa no afecten a las demás, mejorando la mantenibilidad y la escalabilidad del sistema.
- **Alta cohesión:** Las responsabilidades asignadas a una clase deben estar estrechamente relacionadas y ser lógicamente consistentes. En el SIG, cada capa

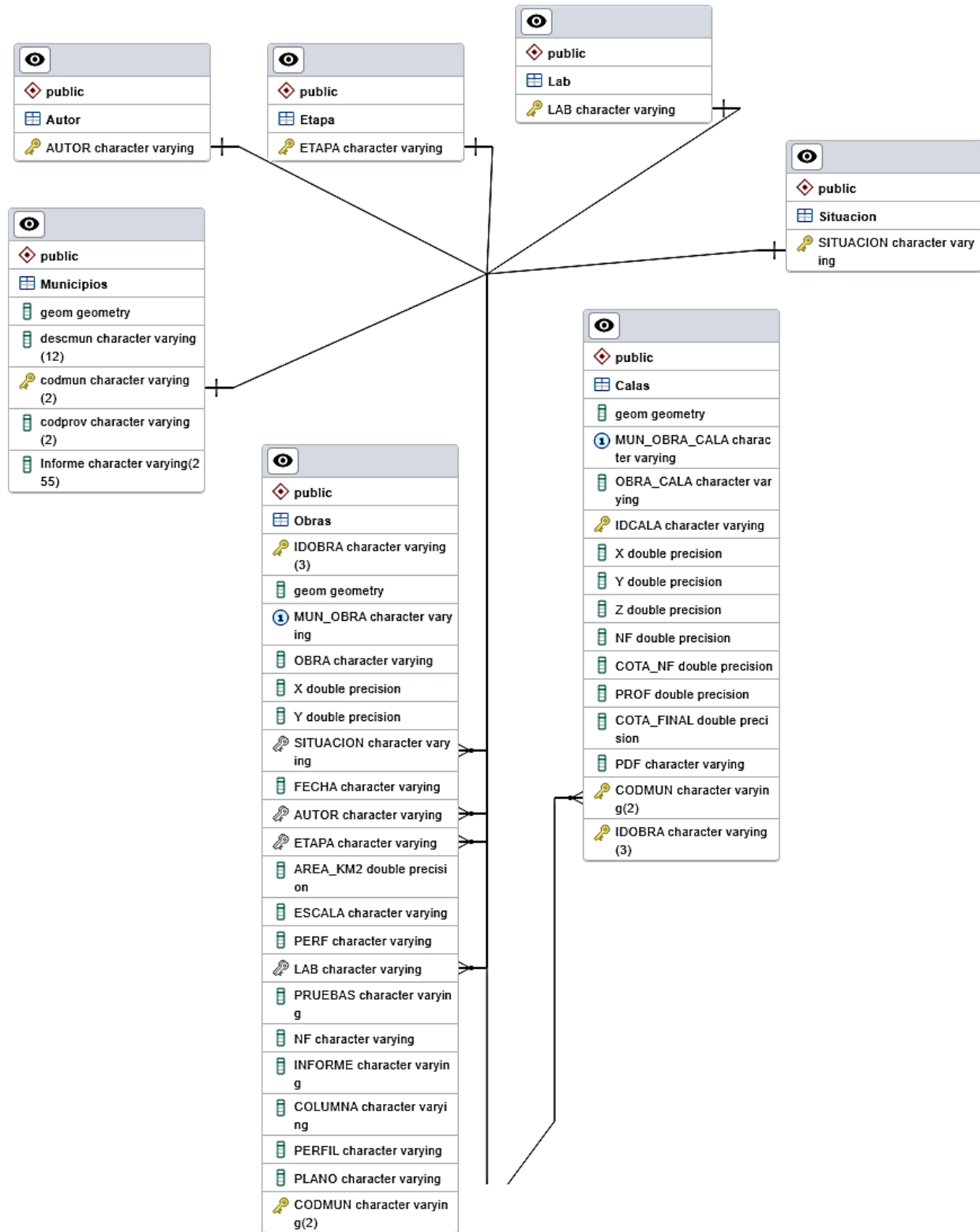
debe tener una responsabilidad bien definida y coherente, como la presentación de datos, la lógica de negocio o el acceso a la base de datos.

- Polimorfismo: Las responsabilidades relacionadas con el comportamiento de las clases deben asignarse a las subclases concretas. En el SIG, el polimorfismo podría aplicarse para manejar diferentes tipos de datos geoespaciales o para proporcionar múltiples formas de visualizar la información.

2.4.4 - Construcción de la solución propuesta

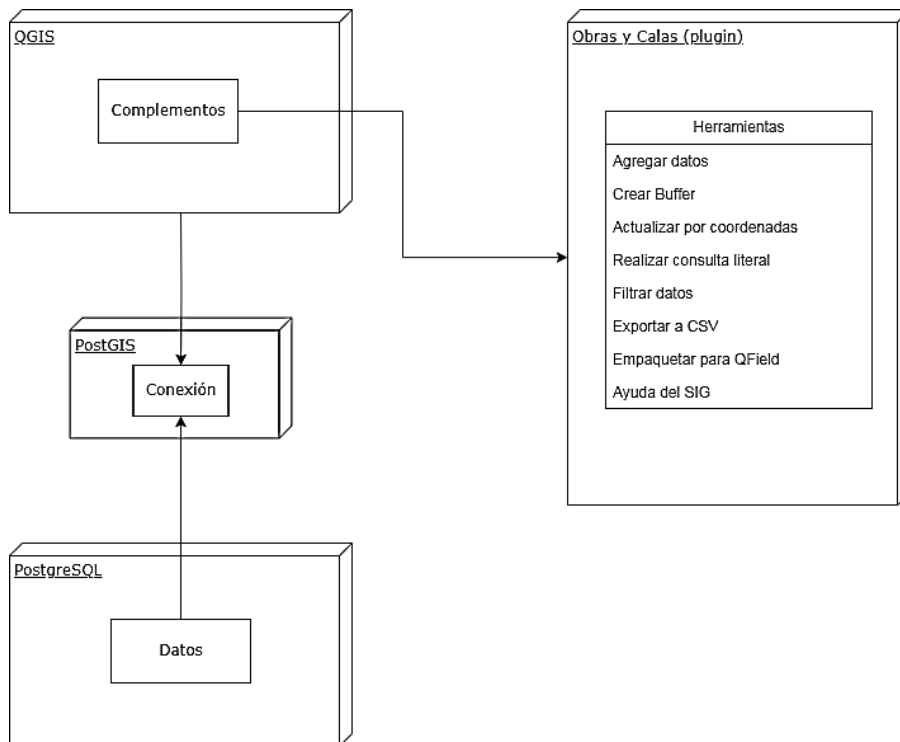
A continuación, aparecen los posibles diagramas que se pueden modelar.

2.4.5 - Diagrama del modelo físico de datos



2.4.6 - Diagramas de componentes

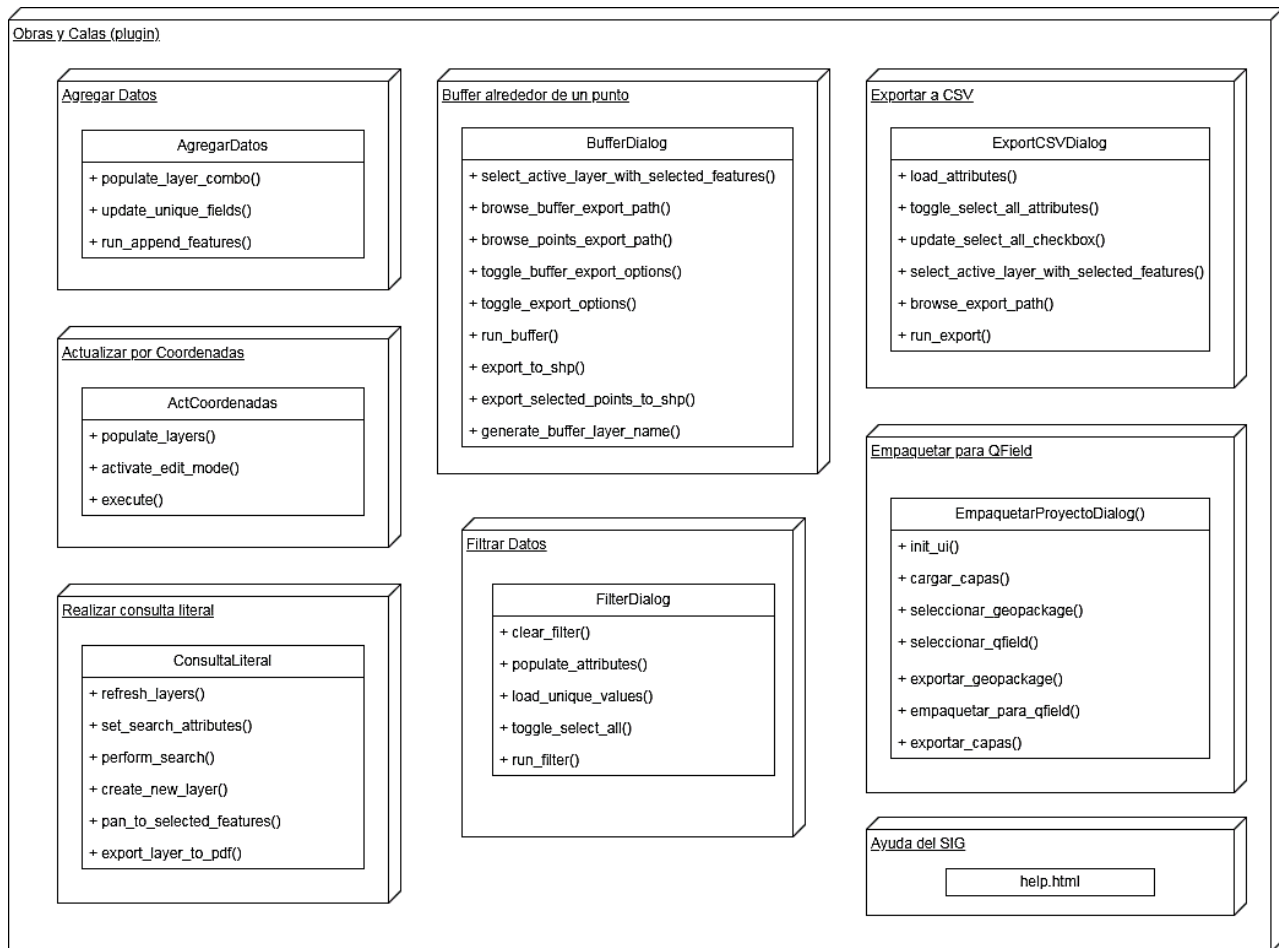
Modela cómo se conectan entre sí los componentes para formar un sistema informático y las interfaces de esos componentes. Ejemplos de componentes son las bibliotecas de terceros, los archivos, los ejecutables, etc.



15 Diagrama de componentes

2.4.7 - Diagramas de clases

En este diagrama están modelados las clases de las herramientas desarrolladas y sus métodos principales. Hay 8 herramientas que le darán solución a los requisitos funcionales.



16 Diagrama de clases

2.5 - Conclusiones parciales

En este capítulo se presentó la propuesta de solución para abordar los problemas identificados en el Capítulo 1 y desarrollar un (SIG) eficiente en EFIA Cienfuegos. A continuación, se resumen las principales conclusiones obtenidas:

- Se estableció una concepción general del sistema propuesto, definiendo los objetivos del SIG, que incluyen la integración de datos geográficos, la mejora en la recuperación y actualización de la información y la agilización de la toma de decisiones en la entidad.
- Se identificaron los requisitos funcionales necesarios para el desarrollo del SIG, describiendo las acciones y funcionalidades específicas que debe tener el sistema.
- Se establecieron los requisitos no funcionales, abordando aspectos relacionados con el rendimiento, seguridad, usabilidad y otros aspectos relevantes para garantizar la calidad y la eficiencia del sistema.
- Se realizó un análisis y diseño detallado de la solución propuesta, describiendo la arquitectura del software, los principios de diseño adoptados y se explicó cómo se llevará a cabo la construcción de la solución.
- Se sentaron las bases para la implementación y prueba del SIG propuesto, considerando las características y requisitos específicos de EFIA Cienfuegos.

En base a estas conclusiones, se puede afirmar que la propuesta de solución planteada en este capítulo se ajusta a las necesidades de EFIA Cienfuegos y tiene el potencial de mejorar significativamente la gestión de la información geográfica en la entidad.

3 - VALIDACIÓN DE LA SOLUCIÓN

3.1 - Introducción

En este capítulo se aborda la validación de la solución propuesta para el Sistema de Información Geográfica (SIG) en EFIA Cienfuegos. Se presentarán los contenidos del capítulo y se explicará el enfoque utilizado para la validación, centrándose en las técnicas de pruebas de caja negra y pruebas de caja blanca. Estas técnicas permiten evaluar el sistema desde diferentes perspectivas y garantizar su funcionamiento óptimo. Se detallarán los procedimientos, escenarios de prueba y resultados obtenidos. Finalmente, se presentarán las conclusiones derivadas de la validación.

3.2 - Diagrama de despliegue

Un diagrama de despliegue es un tipo de diagrama UML que muestra la arquitectura de ejecución de un sistema, incluyendo nodos como entornos de ejecución de hardware o software, y el middleware que los conecta.

Los diagramas de despliegue se utilizan normalmente para visualizar el hardware y el software físico de un sistema. Usándolo puedes entender cómo el sistema se desplegará físicamente en el hardware.

Los diagramas de despliegue ayudan a modelar la topología de hardware de un sistema en comparación con otros tipos de diagramas UML, que en su mayoría esbozan los componentes lógicos de un sistema. [35]



17 Diagrama de despliegue

El servidor tiene la base de datos en PostgreSQL. Las computadoras clientes tienen instalado el SIG en QGIS y se conectan a la base de datos para recibir la información de las capas, que se carga en QGIS. Los usuarios interactúan con QGIS desde las computadoras clientes y pueden modificar la información de la base de datos.

3.3 - Validación de la solución

3.3.1 - Métrica aplicada a los requisitos

Con el objetivo de medir la calidad de la especificación de los requisitos del sistema, se aplicó una de las métricas denominada Calidad de la Especificación (CE). Para determinar cuán entendibles y precisos son los requisitos, primeramente, se calculó el total de requisitos de la especificación, utilizando la siguiente fórmula:

$$Nr = Nf + Nnf$$

Donde:

- Nr: El total de requisitos de especificación
- Nf: Cantidad de requisitos funcionales
- Nnf: Cantidad de requisitos no funcionales

Sustituyendo los valores correspondientes al sistema en cuestión, se obtuvo:

$$Nr = 14 + 7$$

$$Nr = 21$$

Para determinar la Especificidad de los Requisitos (ER), es decir, la ausencia de ambigüedad en los mismos, se realizó la siguiente operación:

$$ER = Nui / Nr$$

Donde Nui es el número de requisitos para los cuales todos los revisores tuvieron interpretaciones idénticas. Mientras más cercano a 1 esté el valor de ER, menor será la ambigüedad.

En el caso de los 21 requisitos obtenidos para el SIG, solo uno produjo contradicción en las interpretaciones. Sustituyendo las variables, se obtuvo:

$$ER = 20 / 21$$

$$ER = 0.95$$

Este resultado final es satisfactorio, indicando que el grado de ambigüedad de los requisitos es sumamente bajo (5%), ya que el 95% es entendible. Los requisitos ambiguos fueron modificados y validados para garantizar su correcta comprensión.

3.3.2 - Pruebas de caja negra

La prueba de caja negra, test funcional o prueba comportamental es un tipo de prueba de software directa, cuya finalidad es analizar la compatibilidad entre las interfaces de cada uno de los componentes del software. No tiene en consideración la lógica interna del sistema. Permite la revisión final de las especificaciones y codificación de un programa. La prueba es considerada aceptable cuando su ejecución conlleva una probabilidad elevada de encontrar un error y es satisfactoria cuando lo detecta.

El propósito es reducir el número de casos de prueba, pero manteniendo la efectividad de ésta, es decir, conseguir localizar el mayor número de errores posible. Esto se consigue mediante la exhaustiva elección de las condiciones de entrada y salida válidas y no válidas, que cubren todas las funcionalidades del sistema.

Este tipo de prueba detecta las siguientes tipologías de errores: funciones incorrectas o ausentes, errores de inicialización y terminación, de rendimiento, errores en estructuras de datos o accesos a bases de datos y en la interfaz.[36]

La técnica utilizada para desarrollar las pruebas de caja negra fue la partición de equivalencia. Se considera como clase de equivalencia al conjunto de datos de entrada que se encarga de definir los estados válidos y no válidos del sistema, considerando como clase válida aquella que genera el valor esperado.

Las condiciones de entrada pueden ser estructuras de datos de diversos tipos, tanto un valor específico, como un intervalo, un conjunto de valores o una condición lógica.

Descripción

Los datos de entrada y los resultados de salida se dividen en diferentes categorías, donde todos los miembros de una categoría están relacionados. Cada una de estas categorías es una partición de equivalencia, en la que el programa actúa de la misma manera para cada miembro de la categoría. Se asume que probar un valor representativo de cada categoría es equivalente a probar cualquier otro valor de la misma categoría.

Clases de equivalencia

Cada condición de entrada o salida (generalmente una frase o sentencia en la especificación) se divide en uno o más grupos. Se identifican dos tipos de clases de equivalencia: válidas e inválidas. Una combinación específica de clases de entrada resultará en una combinación de clases de salida.

A continuación, se presentan las clases de equivalencia para los campos de entrada del formulario del Requisito Funcional RF9: Seleccionar elementos de la capa Obras en un área de análisis.

Campo “Distancia del buffer”:

- Particiones válidas: Números positivos (e.g., 100, 20000.00)
- Particiones inválidas: Números negativos (e.g., -100), letras (e.g., “abc”), caracteres especiales (e.g., “@#\$”)

Campo “Seleccionar capa”:

- Particiones válidas: Opciones disponibles en el menú desplegable
- Particiones inválidas: Opciones no disponibles en el menú desplegable

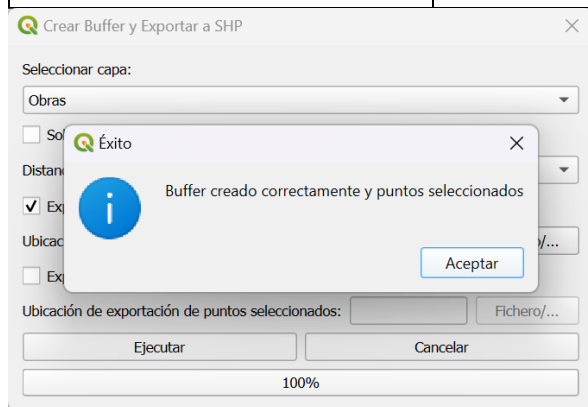
Casos de prueba

Campo “Distancia del buffer”:

Caso de prueba 1

4 Caso de prueba 1 (caja negra)

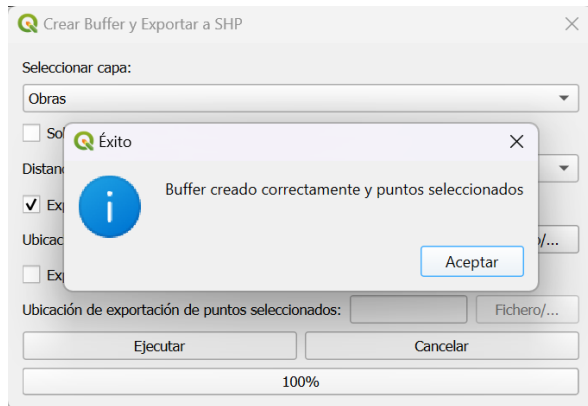
Valores de entrada	Resultado esperado	Resultado obtenido
100	Satisfactorio	Satisfactorio



Caso de prueba 2

5 Caso de prueba 2 (caja negra)

Valores de entrada	Resultado esperado	Resultado obtenido
200.00	Satisfactorio	Satisfactorio



Caso de prueba 3

6 Caso de prueba 3 (caja negra)

Valores de entrada	Resultado esperado	Resultado obtenido
sadls	El sistema no permite introducir cadenas de texto.	El sistema no permitió introducir cadenas de texto.

Caso de prueba 4

7 Caso de prueba 4 (caja negra)

Valores de entrada	Resultado esperado	Resultado obtenido
&#\$*&!*	El sistema no permite introducir caracteres especiales.	El sistema no permitió introducir caracteres especiales.

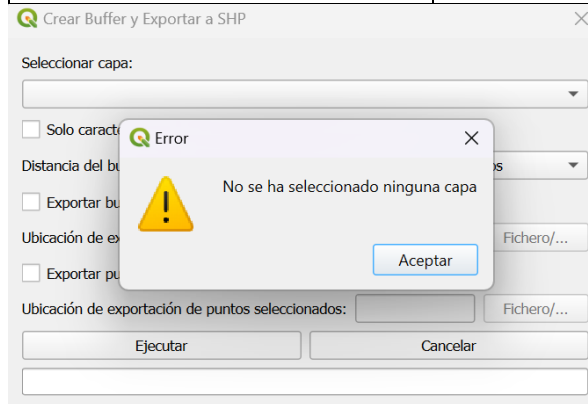
Campo “Seleccionar capa”:

Caso de prueba 5

8 Caso de prueba 5 (caja negra)

Valores de entrada	Resultado esperado	Resultado obtenido
--------------------	--------------------	--------------------

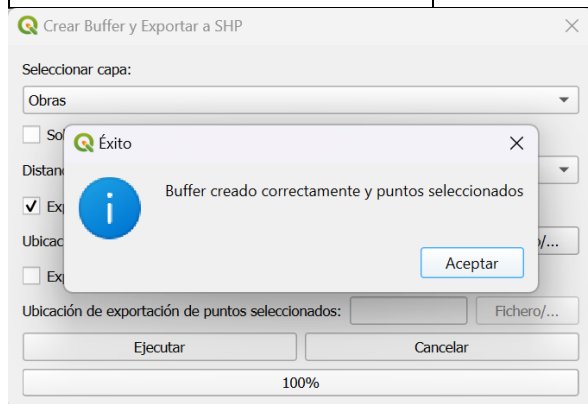
Ninguna capa seleccionada	El sistema debe mostrar un diálogo emergente indicando que debe seleccionar una capa.	El sistema mostró un diálogo emergente indicando que debe seleccionar una capa.
---------------------------	---	---



Caso de prueba 6

9 Caso de prueba 6 (caja negra)

Valores de entrada	Resultado esperado	Resultado obtenido
Se selecciona una capa	Satisfactorio	Satisfactorio



3.2.2 - Pruebas de caja blanca

En este epígrafe se procederá a la realización de pruebas de caja blanca a la función `perform_search` del módulo `consulta_dialog.py`, cuyo propósito es buscar elementos en capas geospaciales dentro de una aplicación de QGIS. A través de estas pruebas, se busca garantizar que la función se comporte de manera correcta y eficiente bajo

diferentes condiciones. Para ello, se utilizará la metodología estructurada de análisis de flujo, cálculo de complejidad ciclomática, y la definición de casos de prueba basados en caminos básicos.

Análisis del Código

En esta sección, se realizará un análisis detallado de la función `perform_search`, identificando las operaciones clave y los posibles caminos lógicos que se pueden tomar durante su ejecución. Este análisis es fundamental para comprender el comportamiento del código y preparar el terreno para la construcción del grafo de flujo.

```
def perform_search(self):
    self.found = []
    self.results.clear()
    self.found_features = {}

    max_items = 5000

    if not str(self.value.displayText()):
        QtWidgets.QMessageBox.warning(self, "Error", "Por favor, ingrese un
valor de búsqueda.")
        return

    for layer in self.iface.mapCanvas().layers():
        if (layer.type() != QgsMapLayer.VectorLayer) or \
            ((self.input_layer_names.currentText() != "[Todas las capas]") and \
             (layer.name() != str(self.input_layer_names.currentText()))):
            continue

        expression = ""
        value = self.value.displayText().strip().replace("'", "")
        comparison = self.comparison.currentText().lower()

        # Construir la expresión de consulta a partir de los campos
        for field in layer.fields():

            if expression != "":
                expression += " OR "

            if comparison == "contiene":
                expression += f'to_string("{field.name()}") ILIKE \'%{value}%\''
```

```

        elif comparison == "empieza con":
            expression += f'LOWER(LEFT(to_string("{field.name()}"),
{len(value)})) = \'{value.lower()}\''

        elif value.isnumeric():
            expression += f'"{field.name()}" {comparison} {value}'
        else:
            expression += f'LOWER(to_string("{field.name()}")) {comparison}
\{value.lower()}\''

    # Seleccionar en esta capa
    layer.selectByExpression(expression)

    # Transformar coordenadas y agregar resultados a la lista
    transform = QgsCoordinateTransform(layer.crs(),
                                      self.iface.mapCanvas().mapSettings().
destinationCrs(),
                                      QgsProject.instance())

    for feature in layer.selectedFeatures():
        if len(self.found) >= max_items:
            break

        if feature.geometry().isNull():
            continue

        name = str(feature.attribute(self.attributes.currentText()))

        field_name = self.id_field.currentText() # Obtener el nombre del
campo seleccionado
        field_value = str(feature.attribute(field_name)) # Obtener el
valor del campo seleccionado

        try:
            transform.transform(feature.geometry().boundingBox().center()
                               center =
self.found.append([layer.name(), feature.id(), center])
            self.results.addItem(f"{layer.name()}: {name} ({{field_name}}:
{field_value})")
        except Exception as e:
            # Capture any exceptions that may occur during the coordinate
transformation
            print(f"Error transforming coordinates for feature
{feature.id()}: {e}")
            continue

```

```

        # Almacenar la característica en el diccionario
        self.found_features[layer.name()] =
self.found_features.get(layer.name(), []) + list(layer.selectedFeatures())

        # Crear una nueva capa con los puntos filtrados
        self.new_layer = self.create_new_layer(layer,
self.found_features[layer.name()])

        # Limpiar la selección en la capa
        layer.removeSelection()

```

La función ``perform_search`` realiza las siguientes operaciones:

- Inicializa variables de resultado y establece un límite de ítems (``max_items``).
- Verifica si se ha ingresado un valor de búsqueda, mostrando un mensaje de error si no es así.
- Itera sobre las capas del mapa, filtrando las capas según ciertos criterios.
- Construye una expresión de consulta basada en los campos de la capa y el valor de búsqueda.
- Selecciona los elementos de la capa que coinciden con la consulta.
- Transforma las coordenadas y agrega los resultados a la lista si cumplen con los criterios.
- Limita la cantidad de resultados a ``max_items``.

Grafo de Flujo

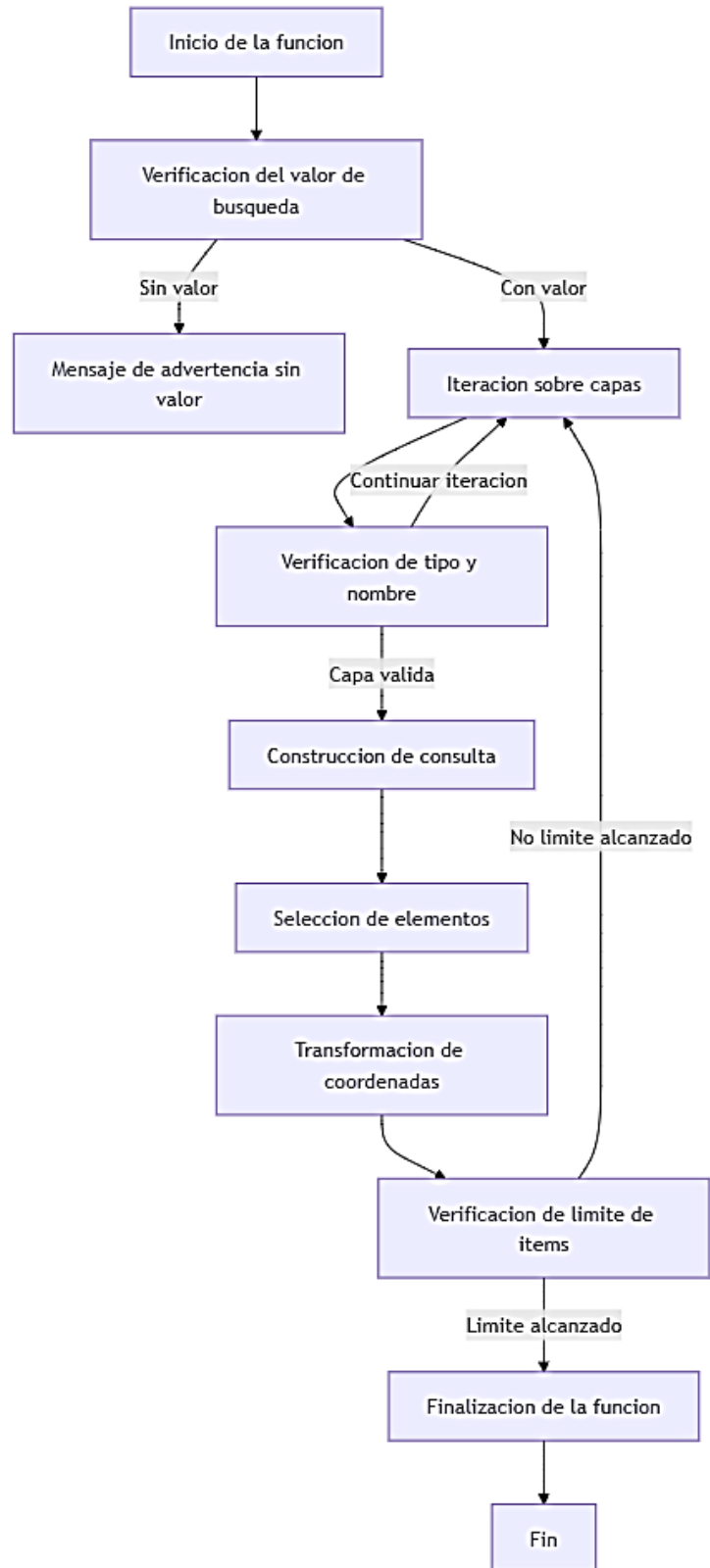
Con el fin de visualizar y entender mejor la estructura de control de la función ``perform_search``, se construirá un grafo de flujo. Este grafo representará gráficamente las decisiones y los procesos que tienen lugar dentro de la función, facilitando la identificación de los caminos independientes y posibles puntos de fallo. La construcción de este grafo es un paso esencial para la siguiente etapa de cálculo de la complejidad ciclomática.

Nodos:

- N1: Inicio de la función.
- N2: Verificación del valor de búsqueda.
- N3: Mensaje de advertencia si no hay valor de búsqueda.
- N4: Iteración sobre las capas del mapa.
- N5: Verificación del tipo de capa y nombre de capa.
- N6: Construcción de la expresión de consulta.
- N7: Selección de elementos en la capa según la consulta.
- N8: Transformación de coordenadas y agregado de resultados.
- N9: Verificación del límite de ítems (`max_items`).
- N10: Finalización de la función.

Aristas:

- A1: N1 → N2
- A2: N2 → N3 (si no hay valor de búsqueda)
- A3: N2 → N4 (si hay valor de búsqueda)
- A4: N4 → N5 (inicio de iteración sobre capas)
- A5: N5 → N4 (continuación de iteración sobre capas)
- A6: N5 → N6 (si la capa es válida)
- A7: N6 → N7
- A8: N7 → N8
- A9: N8 → N9
- A10: N9 → N4 (si no se ha alcanzado el límite de ítems)
- A11: N9 → N10 (si se ha alcanzado el límite de ítems)
- A12: N10 → Fin



18 Grafo de la prueba de caja blanca

Complejidad Ciclomática

A partir del grafo de flujo previamente construido, se calculará la complejidad ciclomática de la función `perform_search`. La complejidad ciclomática es una métrica que indica el número de caminos independientes dentro de una función y proporciona una medida de la complejidad del código. Este cálculo es crucial para determinar el número mínimo de casos de prueba necesarios para garantizar la cobertura total de los caminos lógicos en la función.

Utilizando la fórmula para calcular la complejidad ciclomática:

$$V(G) = E - N + 2P:$$

Donde:

- E = 12 -> (Número de aristas)
- N = 10 ->(Número de nodos)
- P = 1 -> (Número de componentes conectados)

$$V(G) = 12 - 10 + 2 * 1 = 4$$

Esto indica que se necesitan al menos 4 caminos básicos para cubrir todos los caminos independientes en el código.

Evaluación del Riesgo

Complejidad Ciclomática	Evaluación del Riesgo
1-10	Método Simple, sin mucho riesgo
11-20	Más complejo, riesgo moderado
21-50	Complejo, Método de alto riesgo
50	Método no testeable, Muy alto riesgo

Dado que la complejidad ciclomática es 4, el código se encuentra en el rango de 1-10. Según la tabla proporcionada, esto significa que el método es simple y no presenta mucho riesgo.

Caminos Básicos

Basándose en el análisis del código y el cálculo de la complejidad ciclomática, se identificarán los caminos básicos que cubren todos los caminos independientes en la función `perform_search`. La identificación de estos caminos es clave para la definición de casos de prueba que aseguren una evaluación exhaustiva y precisa del comportamiento del código.

Camino Básico 1:

- N1 → N2 → N3 → N10 → Fin (sin valor de búsqueda)

Camino Básico 2:

- N1 → N2 → N4 → N5 → N6 → N7 → N8 → N9 → N4 → Fin (con valor de búsqueda y capas)

Camino Básico 3:

- N1 → N2 → N4 → N5 → N6 → N7 → N8 → N9 → N4 → N10 → Fin (con valor de búsqueda, capas y límite de ítems alcanzado)

Camino Básico 4:

- N1 → N2 → N4 → N5 → N10 → Fin (sin capas válidas)

Casos de Prueba

Finalmente, se definirán y ejecutarán los casos de prueba correspondientes a cada uno de los caminos básicos identificados. Estos casos de prueba están diseñados para verificar que la función `perform_search` se comporta correctamente bajo diferentes condiciones y entradas, asegurando la integridad y fiabilidad del sistema. Cada caso de

prueba se describirá detalladamente, especificando las condiciones de ejecución, las entradas, y los resultados esperados.

Caso de Prueba 1:

10 Caso de prueba 1 (caja blanca)

Descripción: Verificar la ejecución cuando no se ingresa un valor de búsqueda.		
Condición de ejecución: Sin valor de búsqueda.	Entrada: <ul style="list-style-type: none"> • El campo de entrada de búsqueda `self.value` está vacío. • No se realizan selecciones adicionales, ya que la función termina anticipadamente. 	Resultado esperado: Se muestra un mensaje de advertencia indicando que no se ha ingresado un valor de búsqueda.
Resultado obtenido: Satisfactorio		

Caso de Prueba 2:

11 Caso de prueba 2 (caja blanca)

Descripción: Verificar la ejecución con un valor de búsqueda válido y capas seleccionadas.		
Condición de ejecución: Con valor de búsqueda y capas válidas.	Entrada: <ul style="list-style-type: none"> • Se ingresa un valor de búsqueda válido en el campo `self.value`. • Se selecciona "[Todas las capas]" o una capa específica en `self.input_layer_names`. • Se selecciona una comparación válida en `self.comparison` (por ejemplo, "contiene", "empieza con"). 	Resultado esperado: Se muestran los resultados encontrados.

	<ul style="list-style-type: none"> Se selecciona un atributo específico en <code>`self.attributes`</code>. 	
Resultado obtenido: Satisfactorio		

Caso de Prueba 3:

12 Caso de prueba 3 (caja blanca)

Descripción: Verificar la ejecución con un valor de búsqueda válido, capas seleccionadas y alcanzando el límite de ítems.		
Condición de ejecución: Con valor de búsqueda, capas válidas y más de 5000 ítems encontrados.	Entrada: <ul style="list-style-type: none"> Se ingresa un valor de búsqueda válido en el campo <code>`self.value`</code>. Se selecciona "[Todas las capas]" o una capa específica en <code>`self.input_layer_names`</code>. Se selecciona una comparación válida en <code>`self.comparison`</code> (por ejemplo, "contiene", "empieza con"). Se selecciona un atributo específico en <code>`self.attributes`</code>. Se simula la existencia de más de 5000 elementos que coinciden con los criterios de búsqueda. Se selecciona un atributo específico en <code>`self.attributes`</code>. 	Resultado esperado: Se limitan los resultados a 5000 ítems.
Resultado obtenido: Satisfactorio		

Caso de Prueba 4:*13 Caso de prueba 4 (caja blanca)*

Descripción: Verificar la ejecución cuando no hay capas válidas para la búsqueda.		
Condición de ejecución: Sin capas válidas.	Entrada: <ul style="list-style-type: none"> • Se ingresa un valor de búsqueda válido en el campo <code>`self.value`</code>. • Se selecciona una capa en <code>`self.input_layer_names`</code> que no corresponde a ninguna capa en el mapa actual. • Se selecciona una comparación válida en <code>`self.comparison`</code> (por ejemplo, "contiene", "empieza con"). • Se selecciona un atributo específico en <code>`self.attributes`</code>. 	Resultado esperado: No se encuentran resultados.
Resultado obtenido: Satisfactorio		

Las pruebas de caja blanca realizadas sobre la función ``perform_search``, han demostrado que el código es robusto y cumple con los requisitos funcionales esperados. A través de la elaboración y análisis del grafo de flujo, se identificaron todos los caminos lógicos posibles dentro de la función, lo que permitió la definición precisa de casos de prueba que cubrieron completamente el comportamiento del código. El cálculo de la complejidad ciclomática proporcionó una base sólida para asegurar que se evaluaron todas las rutas críticas del código. Cada caso de prueba fue ejecutado con éxito, confirmando que el sistema responde de manera adecuada tanto a entradas válidas como a situaciones de error.

3.3.3 - Pruebas de aceptación

El principal objetivo de las pruebas de aceptación es asegurar que el sistema funcione como se espera y que satisfaga las necesidades del usuario. Esto implica verificar que todas las funcionalidades requeridas están presentes y operativas, además de identificar posibles errores o áreas de mejora antes de la implementación final. [37]

Proceso de Realización

Definición de Criterios: Las pruebas se basan en criterios definidos previamente, que pueden incluir requisitos del usuario, especificaciones del sistema y procesos comerciales.

Preparación del Entorno: Se establece un entorno de prueba que simula el entorno real donde el software será utilizado. Esto permite a los usuarios realizar pruebas en condiciones controladas.

Ejecución y resultados de las Pruebas: Los usuarios finales ejecutan las pruebas diseñadas para verificar la funcionalidad y usabilidad del software.

Definición de criterios

Los usuarios interactúan con el sistema y prueban lo siguiente:

1. Correcta visualización de las capas temáticas del SIG
2. Correcto funcionamiento de todos los requisitos funcionales implementados en el sistema y en el complemento desarrollado.
3. Cumplimiento de los requisitos no funcionales en el flujo de trabajo del SIG.

Preparación del Entorno

Se preparó el entorno de la EFIA para la implementación del SIG, siguiendo el patrón del diagrama de despliegue. Se utilizó una PC cliente para instalar el sistema y se instaló la base de datos en un servidor de la empresa.

Ejecución y resultados de las Pruebas

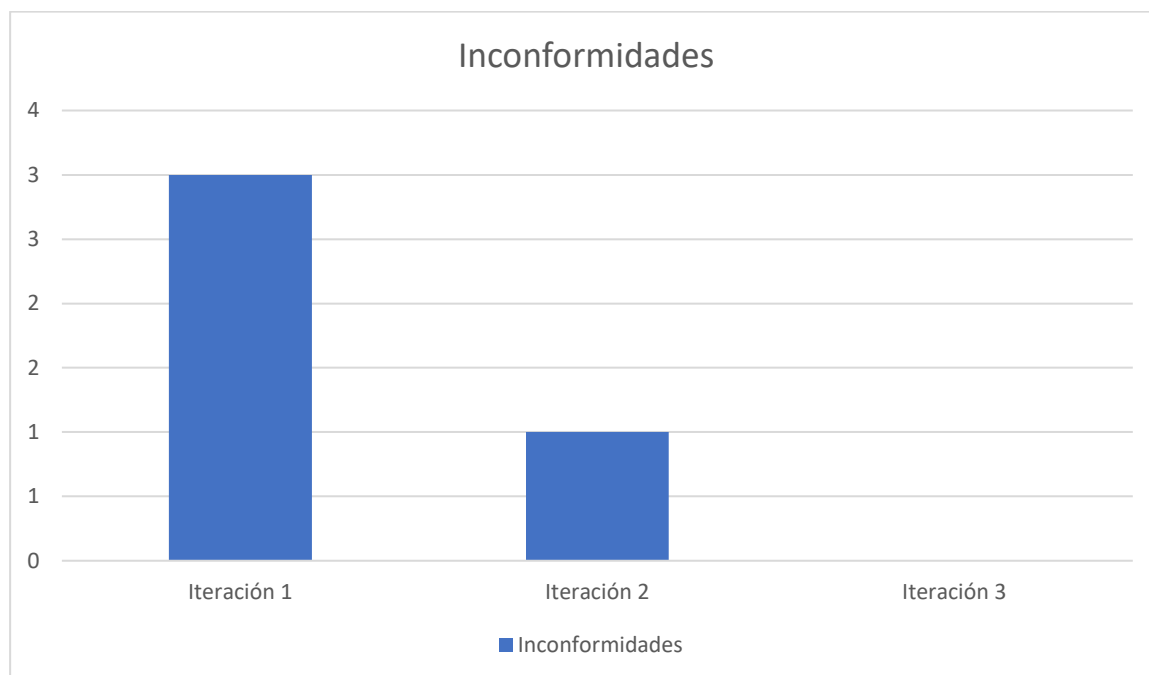
Se ejecutaron las pruebas en 3 iteraciones.

En la primera iteración se detectaron 3 inconformidades por parte de los requisitos funcionales 1 y 3.

En la segunda iteración se le dio solución a las 3 inconformidades y se detectó 1 inconformidad nueva en el requisito funcional 1.

En la tercera iteración no se detectaron inconformidades y los resultados fueron exitosos.

14 Tabla de pruebas de aceptación



3.3 - Conclusiones parciales

En este capítulo, se llevó a cabo la validación de la solución propuesta para el (SIG) en EFIA Cienfuegos. A continuación, se presentan las conclusiones derivadas de este proceso de validación:

- Se realizó la validación de la solución mediante la aplicación de técnicas de pruebas de caja negra y pruebas de caja blanca. Estas técnicas permitieron

evaluar el sistema desde diferentes perspectivas y garantizar un amplio alcance de la validación.

- Las pruebas de caja negra se enfocaron en probar las funcionalidades del SIG sin necesidad de conocer su implementación interna. Se diseñaron escenarios de prueba exhaustivos para evaluar la integridad y el rendimiento del sistema.
- Las pruebas de caja blanca se centraron en evaluar la estructura interna del sistema, analizando su lógica de programación y asegurando que cumpla con los requisitos de diseño y calidad establecidos.
- Los resultados de las pruebas demostraron que la solución propuesta para el SIG en EFIA Cienfuegos cumplió con los requisitos establecidos y proporcionó resultados confiables y precisos.
- La validación permitió identificar y corregir posibles problemas y deficiencias en el sistema antes de su implementación completa, asegurando su funcionamiento óptimo y su capacidad para brindar los beneficios previstos.

En conclusión, la validación de la solución propuesta para el SIG en EFIA Cienfuegos fue exitosa. Las pruebas de caja negra y caja blanca permitieron evaluar exhaustivamente el sistema, garantizando su cumplimiento de los requisitos y su funcionamiento confiable. Estos resultados respaldan la efectividad y la calidad de la solución propuesta.

CONCLUSIONES GENERALES

La investigación realizada para desarrollar un Sistema de Información Geográfica para la recuperación y mantenimiento de la información del Archivo Técnico de la Empresa Filial de Investigaciones Aplicadas Cienfuegos permitió llegar a las siguientes conclusiones:

- Se ha cumplido con el objetivo general de desarrollar un Sistema de Información Geográfica para la gestión de la información del Archivo Técnico de la Empresa Filial de Investigaciones Aplicadas Cienfuegos, el cual permite una representación geoespacial eficiente de los datos, facilitando su recuperación, actualización y mantenimiento.
- Se investigaron diversas tecnologías, metodologías y mecanismos para digitalizar y recopilar la información del Archivo Técnico. Este análisis permitió identificar las herramientas más adecuadas para la implementación del Sistema de Información Geográfica (SIG).
- Se diseñó un Sistema de Información Geográfica para la gestión de la información del Archivo Técnico, lo que facilitó la organización y estructuración de los datos geográficos. Este diseño consideró las necesidades particulares de la Empresa Filial de Investigaciones Aplicadas Cienfuegos (EFIA).
- Se implementaron las funcionalidades requeridas en el sistema diseñado, asegurando que el Sistema de Información Geográfica pudiera realizar tareas de recuperación, actualización y mantenimiento de datos que contribuya a mejorar la operatividad del sistema y su utilidad para los usuarios.
- Se validaron las funcionalidades del SIG a través de pruebas de usuario y retroalimentación, lo que garantizó que el sistema cumpliera con las expectativas y requerimientos de los usuarios finales. Esta validación fue fundamental para asegurar la efectividad y usabilidad del sistema en la gestión de la información técnica de la EFIA Cienfuegos.

RECOMENDACIONES

Algunas recomendaciones para la mejora y evolución del Sistema de Información Geográfica (SIG) en la EFIA Cienfuegos son:

- Realizar actualizaciones regulares del software y la implementación de nuevas funcionalidades según los requisitos emergentes.
- Proporcionar capacitaciones y entrenamientos a los usuarios del SIG para asegurar un uso efectivo del sistema.
- Establecer un sistema de retroalimentación de usuarios para recopilar comentarios y sugerencias.
- Desarrollar otras funcionalidades requeridas al sistema.

REFERENCIAS BIBLIOGRÁFICAS

- [1] «Revolución tecnológica», *Wikipedia, la enciclopedia libre*. 19 de mayo de 2024. Accedido: 27 de mayo de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Revoluci%C3%B3n_tecnol%C3%B3gica&oldid=160214758
- [2] «Historia de los SIG | Línea temporal de los inicios de los SIG y su futuro». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://www.esri.com/es-es/what-is-gis/history-of-gis>
- [3] «Definiciones de un SIG | Introducción a los SIG». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: https://www.educagis.com/files/gis_basico/tema1/definiciones_de_un_sig.html
- [4] «Sistema de información geográfica», *Wikipedia, la enciclopedia libre*. 23 de enero de 2024. Accedido: 21 de mayo de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Sistema_de_informaci%C3%B3n_geogr%C3%A1fica&oldid=157572860
- [5] *Empresa Filial de Investigaciones Aplicadas Cienfuegos | Cienfuegos*. Accedido: 23 de mayo de 2024. [En línea Video]. Disponible en: <https://www.facebook.com/EFIA.Cienfuegos>
- [6] «Empresa Nacional de Investigaciones Aplicadas, Invescons, Cuba | Havana | Facebook». Accedido: 23 de mayo de 2024. [En línea]. Disponible en: <https://www.facebook.com/INVESCONS.CUBA/>
- [7] «2. Introducción a SIG — documentación de QGIS Documentation -». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: https://docs.qgis.org/3.34/es/docs/gentle_gis_introduction/introducing_gis.html#overview
- [8] «¿Qué son los datos geoespaciales? - Datos geoespaciales - AWS», Amazon Web Services, Inc. Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/geospatial-data/>
- [9] «Análisis Espacial: Tipos Y Prácticas De Aplicación». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://eos.com/es/blog/analisis-espacial/>

- [10] «Cartografía digital: qué es, importancia y principales mapas - Mappa». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://mappa.ag/es/blog/cartografia-digital-que-es/>
- [11] «Capa (información geográfica)», *Wikipedia, la enciclopedia libre*. 20 de agosto de 2023. Accedido: 27 de mayo de 2024. [En línea]. Disponible en: [https://es.wikipedia.org/w/index.php?title=Capa_\(informaci%C3%B3n_geogr%C3%A1fica\)&oldid=153201757](https://es.wikipedia.org/w/index.php?title=Capa_(informaci%C3%B3n_geogr%C3%A1fica)&oldid=153201757)
- [12] «Análisis Espacial Vectorial (Buffers)». Accedido: 31 de agosto de 2024. [En línea]. Disponible en: https://docs.qgis.org/2.18/es/docs/gentle_gis_introduction/vector_spatial_analysis_buffers.html
- [13] S. Bell, «Introduction to Analysis», Accedido: 31 de agosto de 2024. [En línea]. Disponible en: <https://openpress.usask.ca/introgeomatics/part/introduction-to-analysis/>
- [14] «Aplicaciones de los Sistemas de Información Geográfica». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://imasgal.com/aplicaciones-sistemas-informacion-geografica/>
- [15] J. V. Rodríguez Serrano y C. R. Ríos Guzmán, «Sistema de información geográfica con interfaz web, para la coordinación del seguimiento a la calidad educativa en la Dirección Departamental de Educación San Vicente.», engd, Universidad de El Salvador, 2012. Accedido: 17 de julio de 2024. [En línea]. Disponible en: <https://oldri.ues.edu.sv/id/eprint/1346/>
- [16] M. Martínez Molina, «Integración de capas de información geográfica en GPIS y desarrollo e implementación de funcionalidades para la planificación de rutas», oct. 2019, Accedido: 17 de julio de 2024. [En línea]. Disponible en: <http://hdl.handle.net/10317/8099>
- [17] «Giswater, solución abierta para la gestión del ciclo del agua». Accedido: 17 de julio de 2024. [En línea]. Disponible en: <https://www.nosolosig.com/articulos/giswater-solucion-abierta-para-la-gestion-del-ciclo-agua>
- [18] D. A. Moreno López, E. C. Rodríguez Mojica, y L. V. Salamanca Urrego, «Tracenet: Plugin en QGIS para Trazado de Líneas con Dirección de Flujo a Partir de Puntos

Georreferenciados», nov. 2015, Accedido: 17 de julio de 2024. [En línea]. Disponible en: <http://repository.udistrital.edu.co/handle/11349/2787>

[19] «MMQGIS». Accedido: 13 de septiembre de 2024. [En línea]. Disponible en: <https://www.michaelminn.com/linux/mmqgis/>

[20] «ArcGIS Online | Cloud-Based GIS Mapping Software Solution». Accedido: 13 de septiembre de 2024. [En línea]. Disponible en: https://www.esri.com/en-us/c/product/arcgis-online?aduc=Advertising&sf_id=7015x000001I5kOAAS&adut=ppcbrand_general&aduco=general&aduca=DGEXPArcGISOnline&utm_id=7015x000001I5kOAAS&adum=PPC&utm_campaign=DGEXPArcGISOnline&adusf=Bing&utm_term=ppcbrand_general&utm_source=Advertising&utm_medium=PPC&utm_content=general&ef_id=:G:s&s_kwid=AL18948!3!!p!!o!!esri%20arcgis%20online&_bk=esri%20arcgis%20online&_bt=&_bm=p&_bn=o&_bg=1311718514686607&aducop=esri%20arcgis%20online-p&gclid=491f2e6a0a471e38d363488ec2004757&gclsrc=3p.ds&msclkid=491f2e6a0a471e38d363488ec2004757

[21] «AUP – Metodología». Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://metodologia.es/aup/>

[22] D. B. Morales, J. B. Borrell, y L. J. Armas, «Aplicación móvil para el análisis de la información captada en SIGEv3.0», *Serie Científica de la Universidad de las Ciencias Informáticas*, vol. 12, n.º 6, Art. n.º 6, jul. 2019.

[23] «Arquitectura de software», *Wikipedia, la enciclopedia libre*. 13 de agosto de 2024. Accedido: 31 de agosto de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Arquitectura_de_software&oldid=161852918

[24] «Visual Studio: IDE y Editor de código para desarrolladores de software y Teams». Accedido: 16 de septiembre de 2024. [En línea]. Disponible en: <https://visualstudio.microsoft.com/es/>

[25] «¿Qué es Python y para qué se usa? Guía para principiantes | Coursera». Accedido: 16 de septiembre de 2024. [En línea]. Disponible en: <https://www.coursera.org/mx/articles/what-is-python-used-for-a-beginners-guide-to-using-python?msckid=0f805443e0dd630627c3406fe1ac626c>

- [26] «Gestor de Base de datos: Qué es, Funcionalidades y Ejemplos». Accedido: 6 de julio de 2024. [En línea]. Disponible en: <https://intelequia.com/es/blog/post/gestor-de-base-de-datos-qu%C3%A9-es-funcionalidades-y-ejemplos>
- [27] «5 gestores de bases de datos para diversas aplicaciones». Accedido: 6 de julio de 2024. [En línea]. Disponible en: <https://docpath.com/5-gestores-de-bases-de-datos-para-diversas-aplicaciones/?lang=es>
- [28] «Introducción al sistema gestor de base de datos (SGBD)», IONOS Digital Guide. Accedido: 6 de julio de 2024. [En línea]. Disponible en: <https://www.ionos.com/es-us/digitalguide/hosting/cuestiones-tecnicas/sistema-gestor-de-base-de-datos-sgbd/>
- [29] «PostgreSQL», *Wikipedia, la enciclopedia libre*. 6 de mayo de 2024. Accedido: 27 de mayo de 2024. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=PostgreSQL&oldid=159952341>
- [30] «Construcción de un visor de Shapefiles con herramientas libres: QGIS, Python y Qt – GeoTux». Accedido: 15 de julio de 2024. [En línea]. Disponible en: <https://geotux.tuxfamily.org/2009/03/14/construccion-de-un-visor-de-shapefiles-con-herramientas-libres-qgis-python-y-qt/>
- [31] «25.1. Complementos de QGIS — documentación de QGIS Documentation -». Accedido: 16 de septiembre de 2024. [En línea]. Disponible en: https://docs.qgis.org/3.16/es/docs/user_manual/plugins/plugins.html
- [32] «16.1. Estructurar Complementos de Python — documentación de QGIS Documentation -». Accedido: 16 de septiembre de 2024. [En línea]. Disponible en: https://docs.qgis.org/3.34/es/docs/pyqgis_developer_cookbook/plugins/plugins.html
- [33] «Arquitectura multicapa», *Wikipedia, la enciclopedia libre*. 10 de junio de 2023. Accedido: 3 de septiembre de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Arquitectura_multicapa&oldid=151757295
- [34] «Patrón de diseño», *Wikipedia, la enciclopedia libre*. 25 de abril de 2024. Accedido: 3 de septiembre de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Patr%C3%B3n_de_dise%C3%B1o&oldid=159701893
- [35] S. Siriwardhana, «Tutorial de Diagrama de Despliegue | ¿Qué es un Diagrama de Despliegue», Creately Blog. Accedido: 3 de septiembre de 2024. [En línea].

Disponible en: <https://creately.com/blog/es/diagramas/tutorial-de-diagrama-de-despliegue/>

[36] R. KeepCoding, «¿Qué son las pruebas de caja negra? | KeepCoding Bootcamps». Accedido: 3 de septiembre de 2024. [En línea]. Disponible en: <https://keepcoding.io/blog/que-son-las-pruebas-de-caja-negra/>

[37] «Pruebas de aceptación (informática)», *Wikipedia, la enciclopedia libre*. 17 de febrero de 2020. Accedido: 16 de septiembre de 2024. [En línea]. Disponible en: [https://es.wikipedia.org/w/index.php?title=Pruebas_de_aceptaci%C3%B3n_\(inform%C3%A1tica\)&oldid=123617459](https://es.wikipedia.org/w/index.php?title=Pruebas_de_aceptaci%C3%B3n_(inform%C3%A1tica)&oldid=123617459)

GLOSARIO DE TÉRMINOS

Listado de términos ordenados alfabéticamente, que son usados en el trabajo de diploma y cuyo significado es poco conocido. Se indica para cada término su significado.

Búfer: Definición: Una operación común en SIG que crea un área alrededor de una entidad geográfica, como un punto, línea o polígono, basada en una distancia dada. Esto se usa para analizar zonas de influencia alrededor de elementos geográficos.

Capa: En un SIG, una capa es una representación visual de datos geoespaciales en forma de mapas. Puede consistir en puntos, líneas, polígonos, imágenes u otras representaciones geográficas.

Complemento de QGIS: En el contexto de un SIG, un complemento es una extensión o herramienta adicional desarrollada para extender las capacidades del software SIG principal, en este caso, QGIS, mediante el uso de Python o C++.

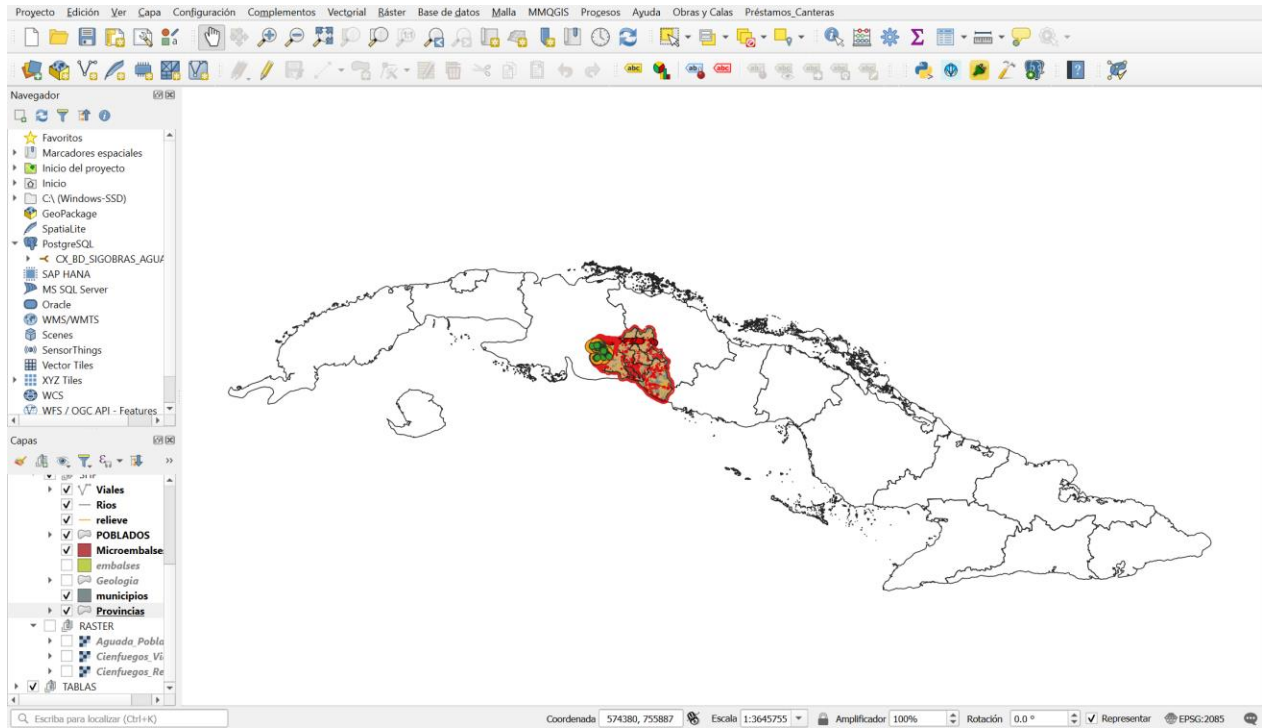
Geometría: En un contexto SIG, la forma y posición de los elementos geográficos, como puntos, líneas o polígonos, que se utilizan para representar componentes del mundo real en un entorno digital.

Interfaz gráfica de usuario (GUI): La porción de un programa de ordenador que permite a los usuarios interactuar con dispositivos de entrada como teclados, ratones y otros dispositivos para realizar acciones y recibir retroalimentación visual.

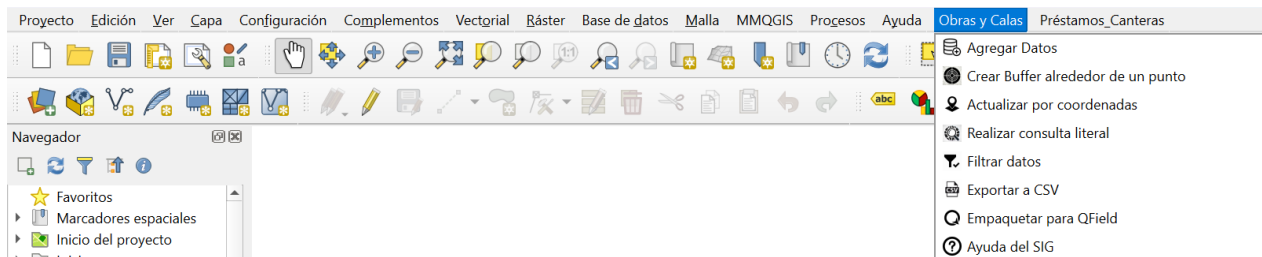
Python: Un lenguaje de programación ampliamente utilizado que es especialmente adecuado para el desarrollo de complementos y scripts en QGIS y otros sistemas de información geográfica.

Sistema de Información Geográfica (SIG): Un sistema de hardware, software, datos, personas, procedimientos y contextos organizativos para recopilar, almacenar, procesar, analizar, modelar y representar datos geoespaciales.

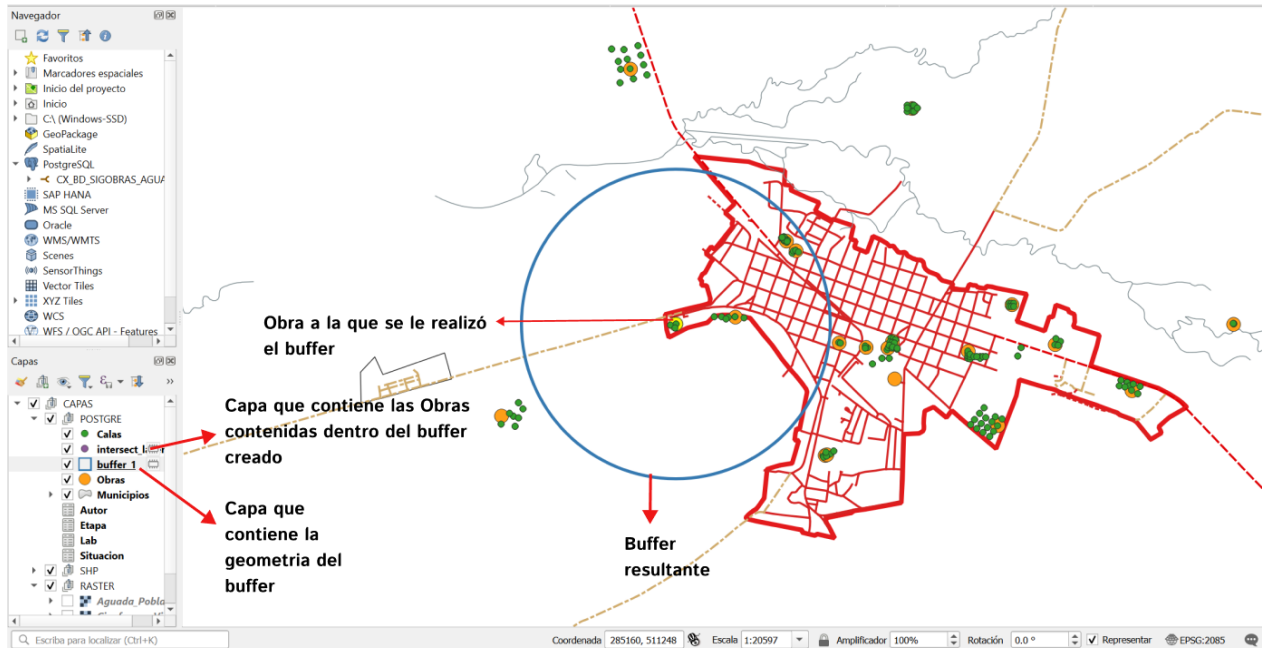
ANEXOS



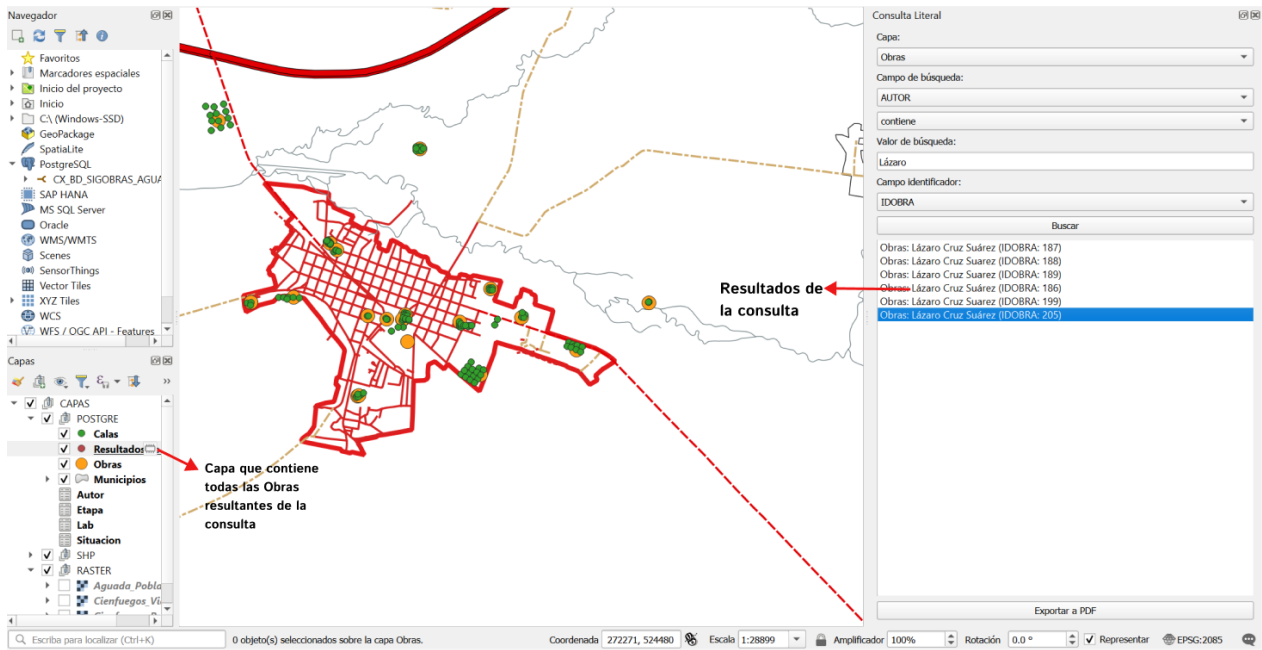
19 Vista general del SIG



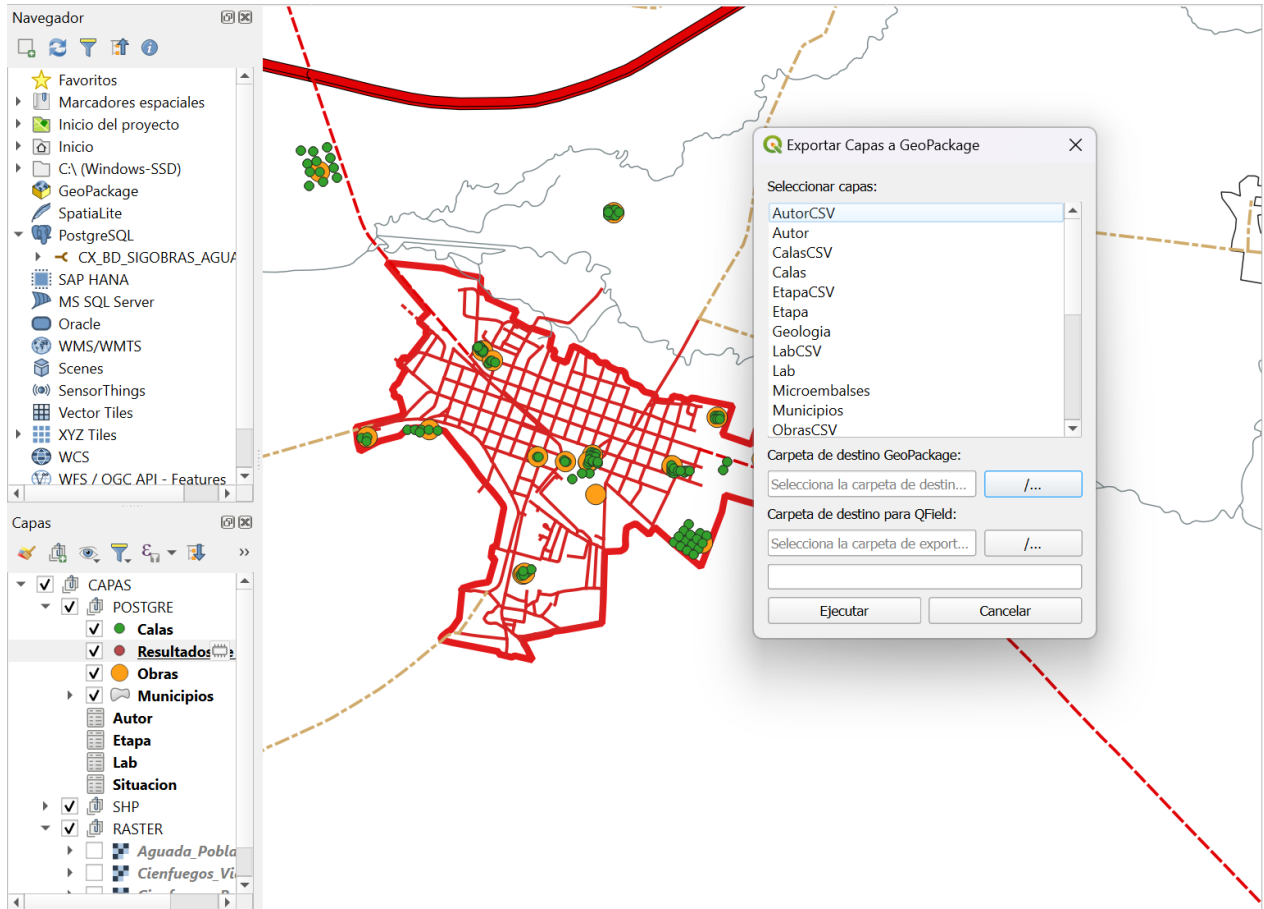
20 Plugin Obras y Calas y sus herramientas



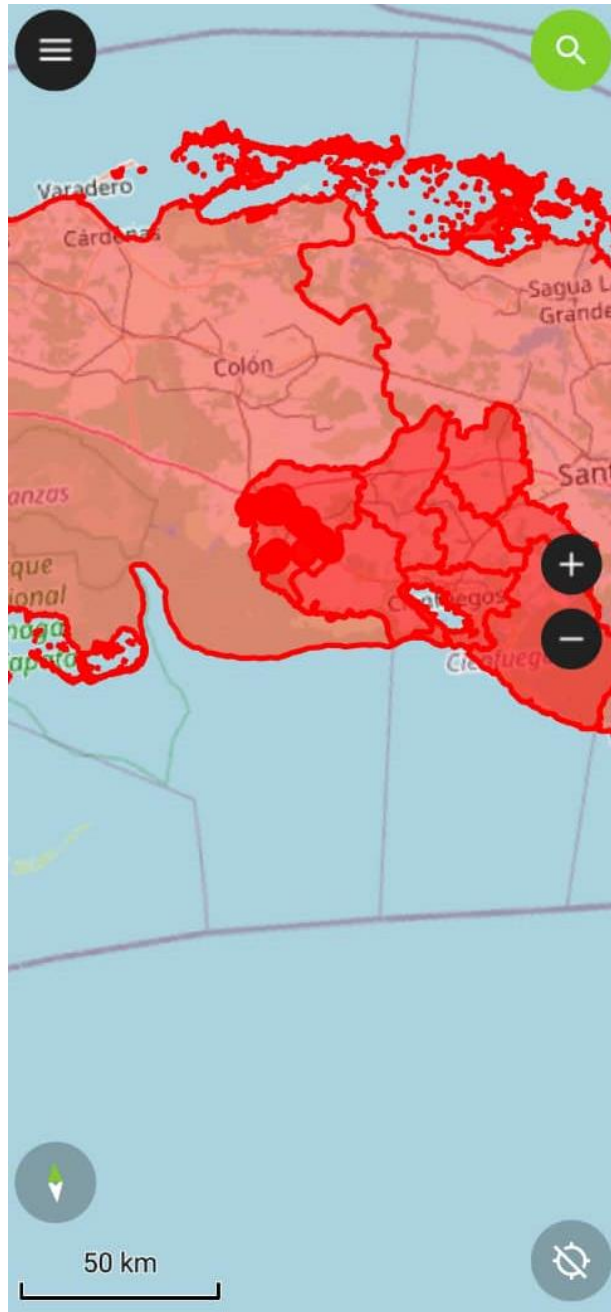
21 Ejemplo de un búfer creado con el plugin



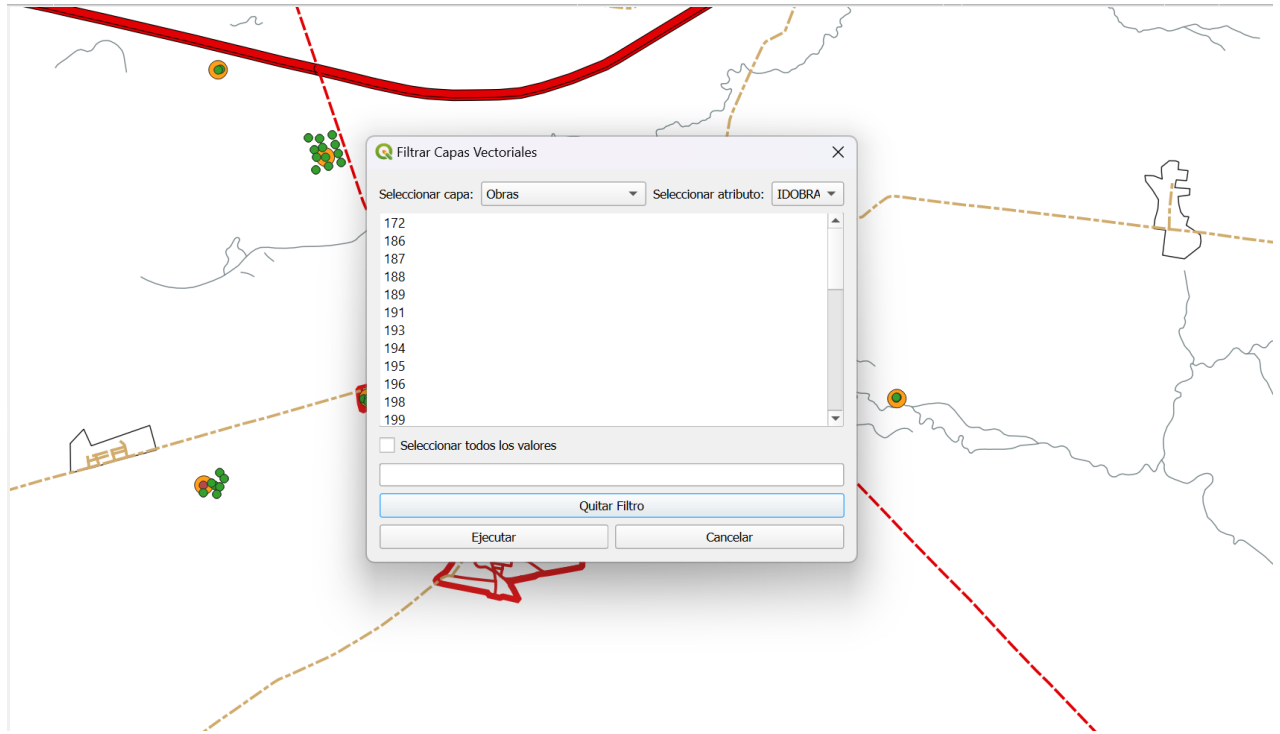
22 Resultados de la consulta realizada con el plugin



23 Herramienta Empaquetar para QField



24 Captura realizada desde un móvil de la aplicación QField



25 Herramienta Filtrar datos

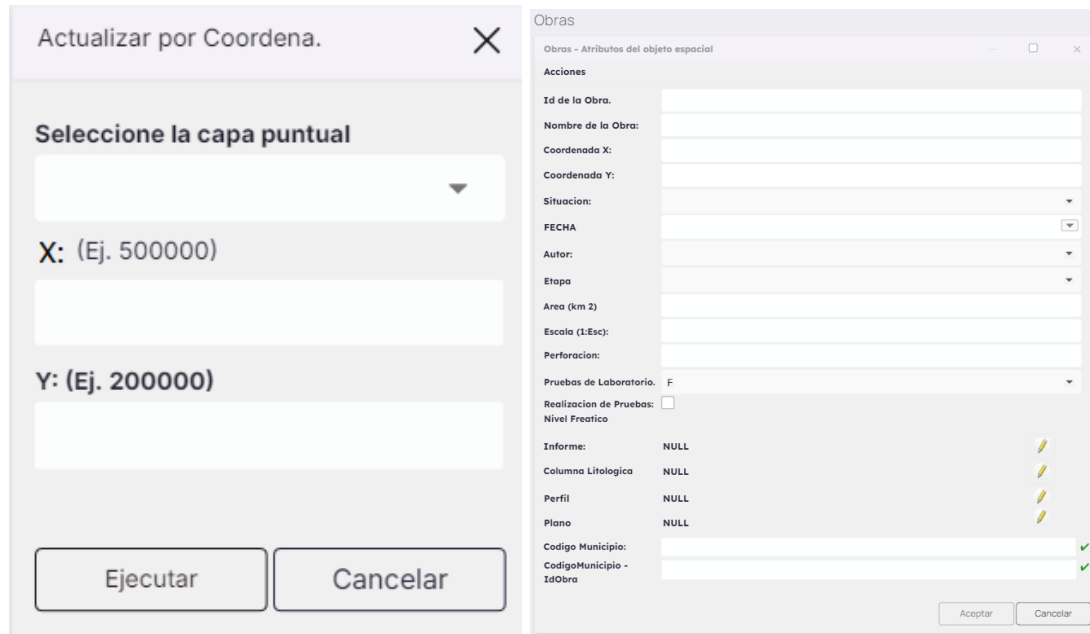
Número: 3	Requisito: Adicionar en el mapa elementos de la capa Obras mediante coordenadas X Y
Programador: Jahdiel López Fiffe	Iteración Asignada: 2
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas

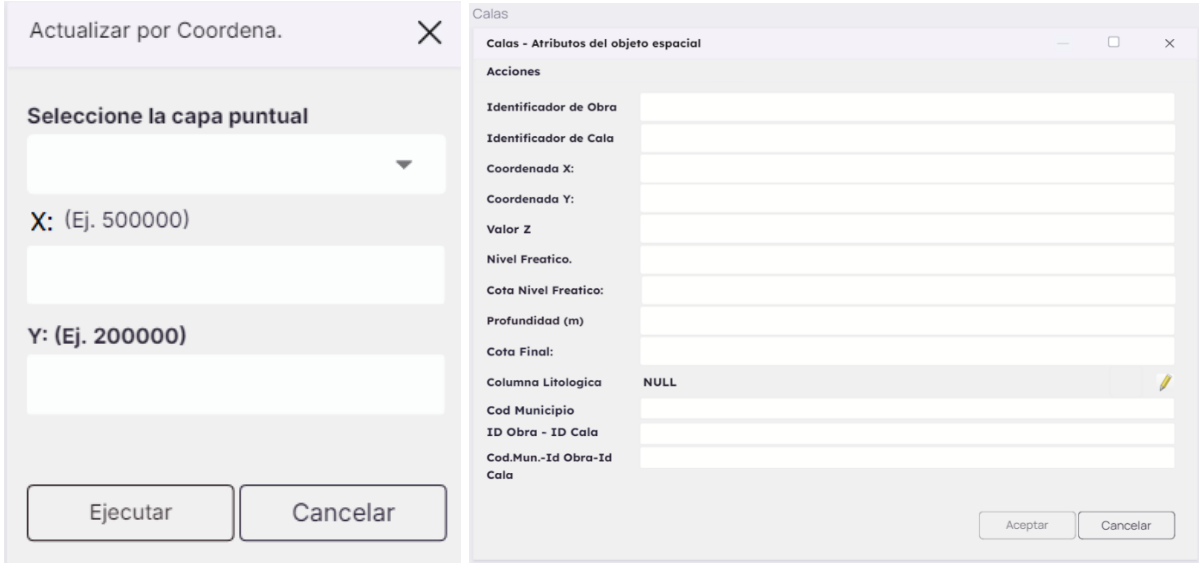
Descripción:

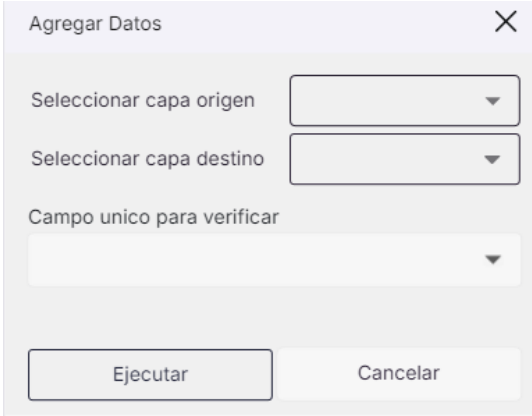
- El sistema solicita al usuario que ingrese las coordenadas X e Y del nuevo punto.
- El sistema invoca el mismo formulario de entrada de datos de la capa **Obras** para que el usuario complete la información adicional.
- Al confirmar la entrada de datos, el sistema guarda la información en la capa correspondiente.

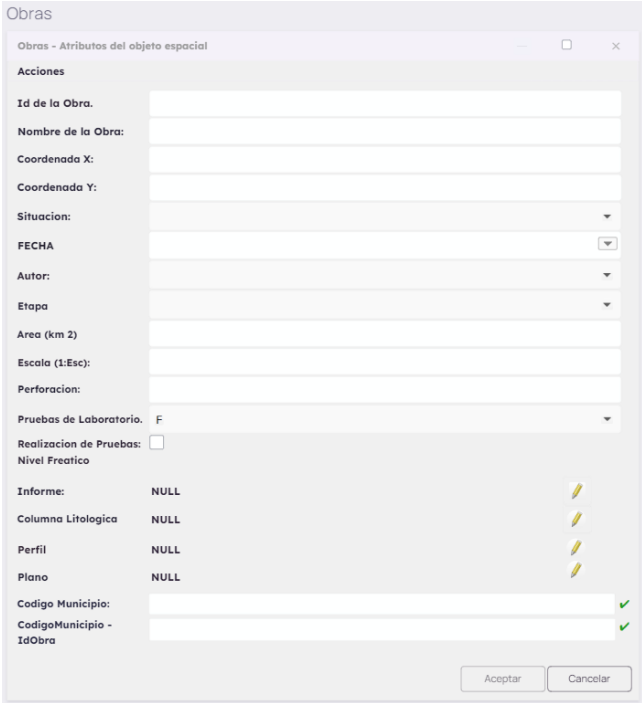
Observaciones: N/A

Prototipo de interfaz:



Número: 4	Requisito: Adicionar en el mapa elementos de la capa Calas mediante coordenadas X Y
Programador: Jahdiel López Fiffe	Iteración Asignada: 2
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: <ul style="list-style-type: none"> - El sistema solicita al usuario que ingrese las coordenadas X e Y del nuevo punto. - El sistema invoca el mismo formulario de entrada de datos de la capa Calas para que el usuario complete la información adicional. - Al confirmar la entrada de datos, el sistema guarda la información en la capa correspondiente. 	
Observaciones: N/A	
Prototipo de interfaz: 	

Número: 5	Requisito: Adicionar elementos a una capa desde otra capa.	
Programador: Jahdiel López Fiffe		Iteración Asignada: 2
Prioridad: Alta		Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio		Tiempo Real: 48 horas
<p>Descripción:</p> <ul style="list-style-type: none"> - El usuario importa datos al mapa desde un archivo CSV a través de una opción de QGIS. - La capa donde se encuentran los datos a adicionar debe tener la misma estructura de la capa destino. - El sistema permite al usuario seleccionar esta capa importada del archivo CSV y verificar la existencia de los datos en la capa correspondiente a través de un campo único. - Si los datos no existen, el sistema adiciona los nuevos puntos a la capa. 		
Observaciones: N/A		
<p>Prototipo de interfaz:</p>  <p>El prototipo muestra una ventana modal titulada 'Agregar Datos' con un botón de cerrar (X) en la esquina superior derecha. Dentro de la ventana, hay tres campos de selección por menú desplegable: 'Seleccionar capa origen', 'Seleccionar capa destino' y 'Campo unico para verificar'. En la parte inferior de la ventana, hay dos botones: 'Ejecutar' y 'Cancelar'.</p>		

Número: 6	Requisito: Editar los datos de la capa Obras
Programador: Jahdiel López Fiffe	Iteración Asignada: 3
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
<p>Descripción:</p> <ul style="list-style-type: none"> - Esta edición se realiza a través de un formulario. - El sistema permite al usuario seleccionar un punto existente y activar el formulario de entrada de datos para la capa Obras al hacer clic sobre el punto. - Al invocar el formulario, el sistema debe cargar automáticamente los datos actuales del punto seleccionado para que el usuario pueda editarlos. - El sistema permite al usuario modificar los datos literales en el formulario y, al confirmar la entrada, guarda los cambios en la capa correspondiente. 	
Observaciones: N/A	
<p>Prototipo de interfaz:</p> 	

Número: 7	Requisito: Editar los datos de la capa Calas
Programador: Jahdiel López Fiffe	Iteración Asignada: 3
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas

Descripción:

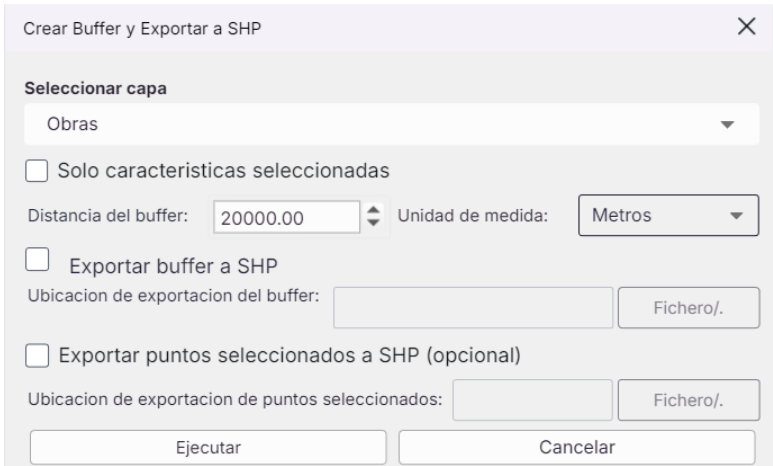
- Esta edición se realiza a través de un formulario.
- El sistema permite al usuario seleccionar un punto existente y activar el formulario de entrada de datos para la capa **Calas** al hacer clic sobre el punto.
- Al invocar el formulario, el sistema debe cargar automáticamente los datos actuales del punto seleccionado para que el usuario pueda editarlos.
- El sistema permite al usuario modificar los datos literales en el formulario y, al confirmar la entrada, guardar los cambios en la capa correspondiente.


Observaciones: N/A

Prototipo de interfaz:

The screenshot shows a web form titled "Calas - Atributos del objeto espacial" with a standard window header (minimize, maximize, close). The form is organized into two columns. The left column lists the following fields: "Acciones", "Identificador de Obra", "Identificador de Cala", "Coordenada X:", "Coordenada Y:", "Valor Z", "Nivel Freatico.", "Cota Nivel Freatico:", "Profundidad (m)", "Cota Final:", "Columna Litologica" (with the value "NULL" and a pencil icon), "Cod Municipio", "ID Obra - ID Cala", and "Cod.Mun.-Id Obra-Id Cala". The right column contains empty input fields corresponding to these labels. At the bottom of the form, there are two buttons: "Aceptar" and "Cancelar".

Número: 8	Requisito: Editar los datos de una capa desde otra capa
Programador: Jahdiel López Fiffe	Iteración Asignada: 3
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
<p>Descripción:</p> <ul style="list-style-type: none"> - El usuario importa datos al mapa desde un archivo CSV a través de una opción de QGIS. - La capa donde se encuentran los datos a modificar debe tener la misma estructura de la capa destino. - El sistema permite al usuario seleccionar esta capa importada del archivo CSV y verificar la existencia de los datos en la capa correspondiente a través de un campo único. - Si los datos existen, el sistema modifica los puntos de la capa. - El sistema al terminar la edición guarda los datos en la tabla correspondiente a la capa. - También existe la posibilidad de cambiar la ubicación de los puntos de la capa a través de la herramienta de edición de vértices de QGIS, esta modificación no invoca al formulario por lo que para modificar los datos literales en este caso después del cambio de posición se procederá a la opción de Editar los datos de la capa. 	
Observaciones: N/A	

Número: 9	Requisito: Seleccionar elementos de la capa Obras en un área de análisis.
Programador: Jahdiel López Fiffe	Iteración Asignada: 3
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: - El sistema permite al usuario realizar un búfer de una distancia determinada alrededor de uno o varios puntos para seleccionar las Obras dentro de esta área para su posterior análisis. - Todos los resultados pueden ser guardados y se visualizan en el mapa.	
Observaciones: N/A	
Prototipo de interfaz: 	

Número: 10	Requisito: Exportar datos a ficheros CSV
Programador: Jahdiel López Fiffe	Iteración Asignada: 4
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: - El sistema permite al usuario seleccionar una capa y el separador y exportar esa estructura y datos a un fichero CSV que posteriormente puede ser analizado en MS Excel.	
Observaciones: N/A	
Prototipo de interfaz:  El prototipo de interfaz muestra una ventana de diálogo titulada 'Exportar Atributos a CSV' con un botón de cerrar (X) en la esquina superior derecha. Dentro de la ventana, hay un menú desplegable 'Seleccionar capa' con un icono de flecha hacia abajo. Debajo de este menú hay dos opciones de selección con casillas de verificación: 'Solo características seleccionadas' y 'Seleccionar todos los atributos'. A continuación, se encuentra el texto 'Seleccionar atributos:' seguido de un espacio en blanco. Más abajo, hay un campo 'Seleccionar separador:' con un menú desplegable. En la parte inferior de la ventana, hay dos botones: 'Ejecutar' y 'Cancelar'. A la izquierda de 'Ejecutar' y 'Cancelar' se encuentran los textos 'Ubicación de exportación' y 'Seleccionar ubicación' respectivamente.	

Número: 11	Requisito: Realizar consultas literales sobre las capas.
Programador: Jahdiel López Fiffe	Iteración Asignada: 4
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas

Descripción:

- El sistema permite al usuario seleccionar de una capa, y un atributo de la misma. Este atributo se compara con un criterio o valor introducido por el usuario y el sistema realizará una búsqueda espacial de las características que cumplan con determinado criterio.
- El sistema posibilitará mostrar en el mapa los elementos que cumplen con esta consulta.
- El sistema realiza zoom a estos elementos seleccionados.
- El sistema permite exportar un archivo PDF con los resultados.

Observaciones: N/A**Prototipo de interfaz:**

Consulta Literal Mapa 11

Capa:

Campo de búsqueda


Valor de búsqueda

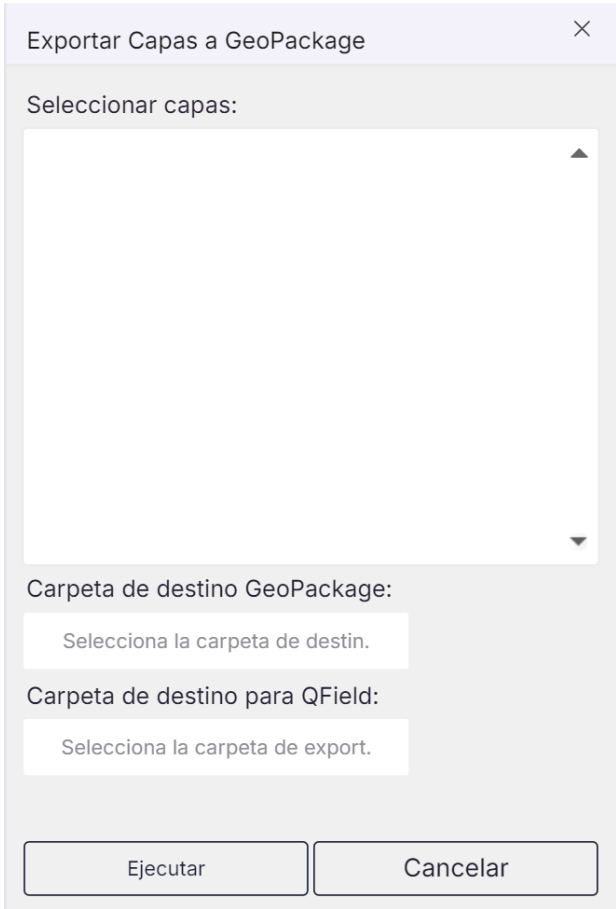
Campo identificador:

Buscar

Resultados:

Exportar a PDF

Número: 12	Requisito: Filtrar datos.
Programador: Jahdiel López Fiffe	Iteración Asignada: 4
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: <ul style="list-style-type: none">- El sistema permite al usuario seleccionar de una capa los datos que desea visualizar en el mapa.- Se seleccionan los atributos que se desean mostrar y se aplica el filtro.- El usuario puede eliminar este filtro.	
Observaciones: N/A	
Prototipo de interfaz:  <p>El prototipo de interfaz muestra un diálogo de usuario con el título "Filtrar Capas Vectoriales". En la parte superior derecha hay un botón de cerrar (X). A continuación, hay dos campos de selección: "Seleccionar capa:" con un menú desplegable y "Seleccionar atributo:" con un menú desplegable. Debajo de estos campos hay un área grande y vacía que probablemente contendrá una lista de valores. En la parte inferior izquierda, hay un checkbox con el texto "Seleccionar todos los valores". En la parte inferior central, hay un botón que dice "Quitar Filtro". En la parte inferior, hay dos botones: "Ejecutar" a la izquierda y "Cancelar" a la derecha.</p>	

Número: 13	Requisito: Empaquetar el proyecto para QField.
Programador: Jahdiel López Fiffe	Iteración Asignada: 4
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: - El sistema le permite al usuario seleccionar las capas que desea empaquetar para la aplicación móvil QField. - QField permite la visualización del SIG de manera offline.	
Observaciones: N/A	
Prototipo de interfaz:  <p>Exportar Capas a GeoPackage</p> <p>Seleccionar capas:</p> <p>Carpeta de destino GeoPackage: Selecciona la carpeta de destin.</p> <p>Carpeta de destino para QField: Selecciona la carpeta de export.</p> <p>Ejecutar Cancelar</p>	

Número: 14	Requisito: Mostrar la ayuda del SIG.
Programador: Jahdiel López Fiffe	Iteración Asignada: 4
Prioridad: Alta	Tiempo Estimado: 72 horas
Riesgo en desarrollo: Medio	Tiempo Real: 48 horas
Descripción: - El usuario podrá consultar una ayuda para las diferentes herramientas desarrolladas en el sistema para comprender su funcionamiento.	
Observaciones: N/A	
Prototipo de interfaz: 