



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 6

**HERRAMIENTA INFORMÁTICA QUE PERMITE DESCRIBIR Y
RELACIONAR DE MANERA ORDENADA LOS ARCHIVOS
CONTENEDORES DE CARTOGRAFÍAS DIGITALES RASTER Y
VECTORIALES.**

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

AUTOR: Gelsys González Souto.

TUTOR: Ing. Jesse Daniel Cano.

CO-TUTOR: Ing. Joel Macías Roque.

La Habana, 23junio, 2011.

Año 53 de la Revolución.

DEDICATORIA

Dedico este trabajo de diploma a mis padres, por ser las personas más importantes de mi vida, gracias por todo su cariño y apoyo, toda mi vida se la debo a ustedes, y en especial esta carrera, por ser siempre un ejemplo a seguir desde pequeña en mi formación profesional y en la vida, los quiero mucho.

AGRADECIMIENTOS

A mis padres por ser mi guía y ejemplo en cada instante de mi vida.

A mis abuelos, por darme todo su amor y comprensión y darle la satisfacción de verme graduada como siempre lo han soñado.

A mis tías Liliana y Magalys, gracias por contribuir a mi formación profesional y apoyarme siempre con sus conocimientos.

A mis tíos, por ayudarme en todo lo que está a su alcance e incluso en lo que no también.

A toda mi familia por formar parte de este triunfo.

A Roly, por ser mi apoyo incondicional, demostrándome que soy capaz de superar cualquier obstáculo y darme las fuerzas necesarias para seguir adelante y lograr mis objetivos. Gracias por tu comprensión y darme los mejores consejos; sabes que no siempre tienes la razón....TE AMO.

A mis suegros, que se han convertido en mi segunda familia, gracias por todos sus consejos.

A Joel, gracias por tu ayuda, a pesar de caerte tanto arriba y dedicarme tanto tiempo, de lo contrario no sé qué hubiera podido pasar con esta tesis y su tesista.

A Jesse, mi tutor, por guiarme y defenderme en cada fase de desarrollo de la tesis.

A mis grandes amigos Humberto, José Ángel, Alejandro, Raiza, Lianet: por compartir los momentos de alegrías y tristeza, brindándome toda su comprensión y apoyándome en mis decisiones, los quiero mucho, nunca los olvidaré.

A mis amistades Alain Jiménez, Alain Sánchez, Eric, Carlos Luis, Yania, Yosnay, Yenía, Lisandra Sosa, Anaivis por estar al tanto de mí y poner un granito de arena a lo largo de estos cinco años.

A todo el proyecto SIG-Desktop y especialmente a mi Jefe de Proyecto y amigo Yoandry Lazo, por apoyarme en el desarrollo de la tesis y a lo largo de toda mi estancia en el proyecto.

A los miembros del tribunal por todo el apoyo y los consejos durante la investigación y desarrollo de este trabajo de diploma.

A todos las personas que de una forma u otra que han compartido estos cinco años de carreras inolvidables en la UCI muchas gracias.

DECLARACIÓN DE AUDITORÍA

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

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

Gelsys González Souto.

Ing. Jesse Daniel Cano Otero.

OPINIONES Y AVALES

DATOS DE CONTACTO.

TUTOR:

Nombre: Ing. Jesse Daniel Cano Otero.

Correo Electrónico: jdcano@uci.cu

Universidad de las Ciencias Informáticas.

CO-TUTOR

Nombre: Ing. Joel Macías Roque.

Correo Electrónico: jmroque@uci.cu

Universidad de las Ciencias Informáticas.

OPINIÓN DEL TUTOR

RESUMEN

El continuo avance de las Tecnologías de la Información y las Comunicaciones (TIC), y el aumento de información geográfica, han traído como consecuencia, el uso a mayor escala de los Sistemas de Información Geográfica (SIG), convirtiéndose en una herramienta poderosa, para el análisis y toma de decisiones.

La Universidad de las Ciencias Informáticas está inmersa en el estudio y desarrollo de los SIG, al frente de esta tarea se encuentra el Departamento de Geoinformática, que pertenece al Centro de Desarrollo de Geoinformática y Señales Digitales (GEYSED). Actualmente se trabaja en el desarrollo de aplicaciones de escritorio para SIG, dicha actividad es desempeñada por el proyecto productivo SIG-Desktop. Todo este proceso necesita un sinnúmero de cartografías digitales, que dan lugar a importantes volúmenes de archivos raster y vectoriales. Hoy en día la búsqueda y clasificación de dichos archivos se realiza de forma no automatizada, lo que trae como consecuencia incumplimiento en los cronogramas planificados y en la entrega de información, al manejar y controlar, gran cantidad de archivos raster y vectoriales, es por ello que surge la necesidad de implementar un sistema informático que permita describir y relacionar de manera ordenada los archivos contenedores de cartografías digitales raster y vectoriales, proporcionando la información necesaria para interpretar y procesar los datos .

Para ello se llevó a cabo un estudio minucioso de las tendencias y tecnologías que existen a nivel mundial, donde se seleccionó las herramientas y tecnologías correctas para llevar a cabo un buen desarrollo de la aplicación informática, con la mayor calidad, bajo tecnologías libres en su mayoría. Se cuenta con una documentación técnica, donde describe todo los artefactos generados correspondientes a cada flujo de trabajo.

Palabras Claves: Sistemas de Información Geográfica (SIG), Tecnologías Libres. Tecnologías de la Información y las Comunicaciones (TIC), archivos raster, archivos vectoriales.

INDÍCE

INTRODUCCIÓN	1
CAPÍTULO1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 Introducción.....	5
1.2 Conceptos asociados al dominio del sistema.	5
1.2.1 Sistema de Información Geográfica (SIG).....	5
1.2.2 Datos Espaciales.	6
1.2.3 Metadatos.....	6
1.2.4 Catálogo de Datos Geográficos.	7
1.2.5 Cartografía.....	7
1.3 Descripción General del Objeto de estudio.....	8
1.3.1 La representación de los datos en los diferentes SIG.	8
1.3.2 Tipos de Cartografía Digital.	14
1.4 Situación Problemática.	15
1.5 Análisis de otras soluciones existentes.....	15
1.6 Conclusiones Parciales.	17
CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	18
2.1 Introducción.....	18
2.2 Metodología de desarrollo de software: Rational Unified Process (RUP).....	18
2.3 Lenguaje de Modelación: Unified Modeling Language (UML).....	20
2.5 Herramienta CASE: Visual Paradigm.	22
2.6 Sistema de Base Datos: SQLite.	23
2.7 Lenguaje de Programación: C++.....	24
2.8 Entorno de Trabajo: Qt. Creator.	25
2.9 Conclusiones Parciales.	27
CAPÍTULO 3: PRESENTACIÓN DE LA SOLUCIÓN.	28
3.1 Introducción.....	28
3.2 Modelo de dominio.	28
3.2.1 Diagrama de clases del Modelo de Dominio	29

3.2.2	Glosario de Términos del Dominio.....	29
3.3	Requisitos Funcionales.....	31
3.4	Requisitos no Funcionales.....	32
3.5	Descripción del sistema propuesto.....	33
3.5.1	Descripción de los actores del sistema.....	33
3.5.2	Diagrama de casos de usos del sistema.....	34
3.5.3	Descripción de los casos de usos del sistema.....	34
3.6	Conclusiones Parciales.....	37
CAPÍTULO 4 CONSTRUCCIÓN DE LA PROPUESTA DE SOLUCIÓN.....		38
4.1	Introducción.....	38
4.2	Diseño.....	38
4.2.1	Diagramas de Clases del Diseño.....	39
4.3	Arquitectura de software.....	40
4.4	Arquitectura 3 Capas.....	40
4.5	Diseño de la Base de Datos.....	41
4.5.1	Diagrama de Clases Persistentes.....	42
4.5.2	Modelo Entidad-Relación.....	43
4.6	Modelo de Despliegue.....	44
4.6.1	Diagrama de Despliegue.....	44
4.7	Modelo de Implementación.....	44
4.7.1	Diagrama de Componentes.....	45
4.8	Conclusiones Parciales.....	46
CONCLUSIONES.....		47
RECOMENDACIONES.....		48
REFERENCIAS BIBLIOGRÁFICAS.....		49
BIBLIOGRAFÍA.....		53
GLOSARIO DE TÉRMINOS.....		54
ANEXOS.....		56

INDÍCE DE TABLAS Y FIGURAS

INDÍCE DE TABLAS

Tabla 1: Descripción de los actores del sistema.	33
Tabla 2: Descripción textual de CU Categorizar Archivos.	34
Tabla 3: Descripción textual de CU Autenticar Usuario.	56
Tabla 4 : Descripción textual de CU Actualizar Ruta.	57
Tabla 5 : Descripción textual de CU Vista Previa.	62
Tabla 6 : Descripción textual de CU Visualizar Metadatos.	65

INDÍCE DE FIGURAS

Figura 1.Modelo de datos de un SIG.....	8
Figura 2.Representación exhaustiva.	9
Figura 3.Run-length encoding.	10
Figura 4.Representación Vectorial.	11
Figura 5.Lista de coordenadas espagueti.	12
Figura 6.Diccionario de vértices.	13
Figura 7.Ficheros DIME.	13
Figura 8.Arco / Nodo.....	14
Figura 9.Fases e Iteraciones de la Metodología RUP.	20
Figura 10.Diagrama de Clases del Modelo de Dominio.....	29
Figura 11.Diagrama de casos de usos del sistema.	34
Figura 12.Diagrama del Diseño CU Categorizar Archivos.....	39
Figura 13.Diagrama de Clases Persistentes.	42
Figura 14.Diagrama de Entidad – Relación.....	43
Figura 15.Diagrama de Despliegue.....	44

Figura 16.Diagrama de Componentes CU Categorizar Archivos.....	45
Figura 17 .Diagrama del Diseño CU Autenticar Usuario.....	68
Figura 18. Diagrama del Diseño CU Visualizar Metadatos.....	69
Figura 19. Diagrama del Diseño CU Vista Previa.....	70
Figura 20 .Diagrama de Componente CU Vista Previa.	71
Figura 21 .Diagrama de Componente CU Autenticar Usuario.	72

INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones (TIC) han permitido un avance increíble en el estudio de la naturaleza, tanto es así que desde la aparición de las TIC se han realizado casi tantos descubrimientos como en toda la historia antes de su creación. El papel de la TIC se fundamenta en su potencial para manejar información como recurso esencial en el desarrollo humano.

El conjunto de datos que representa la información geográfica también se ha visto beneficiada con la creación de estas herramientas y tecnologías, como ejemplo de esto se tienen los Sistemas de Información Geográfica(SIG), los cuales permiten generar, procesar o representar dicha información así como almacenarla para su posterior consulta.(Chuvieco et al, 2005)

En el marco de la globalización, el estudio de la información geográficamente referenciada se hace cada vez más importante, ya que diversas decisiones a nivel global requieren de la existencia, el procesamiento y la posibilidad de compartir dicha información, como datos de calentamiento global, principales fuentes de contaminación, rutas de transporte, entre otros tipos de información de gran utilidad para toda la humanidad. Por este motivo uno de los retos de la sociedad actual es lograr la difusión de la información geoespacial y sus tecnologías asociadas, lo cual conlleva una serie de estrategias que van desde capacitar a la sociedad en general en el uso de los SIG hasta convencer a los responsables de la toma de decisiones de las ventajas del uso de dichos sistemas.

Esta tecnología, en un período de tan solo treinta años, se ha convertido para muchos en herramienta fundamental de análisis y de toma de decisiones y ha entrado a formar parte del acervo cultural de la población en general.

En Cuba existen aislados proyectos que estudian y tratan de dar solución al desarrollo del SIG en diferentes plataformas y con funcionalidades distintas, sin embargo para garantizar homogeneidad, organización y control de estos procesos surge la idea de crear el Departamento de Geoinformática que pertenece al Centro de Desarrollo de Geoinformática y Señales Digitales (GEYSED) el cual pertenece a la actualmente Facultad 6 de la Universidad de las Ciencias Informáticas.

En dicho departamento se encuentra el proyecto productivo SIG-Desktop, el cual tiene como objetivo proveer un producto desarrollado en su totalidad con herramientas de licencia libre el cual servirá como soporte al desarrollo de aplicaciones de SIG en plataforma de escritorio. La utilización de las diferentes herramientas y tecnologías libres representan un gran beneficio para los clientes.

A medida que crezca el interés de los clientes por el desarrollo de un SIG, traerá como resultado, el aumento de cartografías digitales a utilizar, lo que da lugar a la aparición de gran diversidad de archivos raster y vectoriales, para su posterior búsqueda y clasificación. Actualmente este proceso se realiza de forma no automatizada, lo que provoca atrasos en la producción y en la entrega de información, por lo que surge la necesidad de desarrollar una herramienta capaz de clasificar estos formatos de manera automática para lograr una mayor agilidad a la hora de su uso.

A partir del análisis de la situación problemática de la investigación se define como el **Problema a resolver** la siguiente interrogante:

¿Cómo agilizar el proceso para llevar a cabo la clasificación de los formatos de archivos contenedores de cartografías digitales para sistemas de información geográfica?

El problema planteado está enmarcado dentro del **Objeto de Estudio**:

Automatización del proceso de representación de terreno.

El **Campo de Acción** de la presente investigación está centrado en la automatización del proceso de clasificación de archivos cartográficos.

La meta a alcanzar en el presente trabajo para solucionar el problema queda sujeta al siguiente:

Objetivo General:

Desarrollo de una herramienta informática que permita describir y relacionar de manera ordenada los archivos contenedores de cartografías digitales para su posterior representación.

La **Idea a Defender** queda planteada de la siguiente manera:

Si se desarrolla una herramienta informática que permita describir y relacionar de manera ordenada los archivos contenedores de cartografías digitales se disminuirá el esfuerzo que implica clasificar dichos archivos para su posterior representación con el uso de un sistema de información geográfica.

Para cumplir con el objetivo trazado en esta investigación, es necesario llevar a cabo un conjunto de **Tareas Investigativas** que permitan, de manera sistemática y creciente, ir avanzando en el desarrollo de la investigación para lograr dar solución al problema científico que le dio origen a la misma:

1. Elaborar el marco metodológico y teórico.
2. Caracterizar las tendencias y tecnologías actuales a desarrollar.
3. Caracterizar las tendencias y tecnologías actuales a desarrollar.
4. Identificar los principales aspectos a automatizar en el proceso obtención de información geográfica.
5. Realizar las actividades de Diseño que tribute a la implementación del producto.
6. Implementar un producto que cumpla con las especificaciones identificadas y de acuerdo al diseño confeccionado.

Los posibles resultados que se quieren obtener al finalizar el trabajo de diploma son: un catálogo de archivos contenedores de cartografías digitales y la documentación técnica asociada al desarrollo de la herramienta propuesta.

Durante el transcurso y desarrollo de esta investigación científica, se utilizarán algunos **métodos teóricos** que se detallan a continuación:

- **Histórico - Lógico:** En la primera fase de la investigación se desarrolla un estudio del estado del arte de la problemática analizada, revisando de forma crítica cada uno de los documentos para lograr un mejor entendimiento de lo que debemos desarrollar. Asimismo, se permitió conocer detalladamente la trayectoria y desarrollo de aplicaciones SIG teniendo en cuenta sus precedentes históricos, las investigaciones realizadas y los resultados obtenidos por otros autores al efectuar dichas investigaciones.
- **Analítico- sintético:** Para descomponer el problema de investigación en partes, para un mejor entendimiento de la situación y luego poder sintetizarlos para la confección de la solución propuesta.
- **Modelación:** Se utiliza para la modelación de diagramas, representar el proceso de desarrollo y propiciar mejor entendimiento de la solución a implementar.

En toda investigación científica se utilizan métodos empíricos específicamente las Entrevistas a los jefes del Departamento de Geoinformática y al líder del proyecto productivo SIG-Desktop con el objetivo de recopilar la mayor información que nos puedan brindar a partir de sus experiencias , para realizar una mejor trabajo.

CAPÍTULO1: Fundamentación Teórica.

1.1 Introducción.

En el presente capítulo se describen los conceptos asociados al dominio del problema, los cuales serán de gran utilidad para el entendimiento del desarrollo de esta investigación. Se abordan los temas relacionados con la fundamentación teórica de la investigación, haciendo referencias a los principales elementos del estudio de los SIG.

1.2 Conceptos asociados al dominio del sistema.

1.2.1 Sistema de Información Geográfica (SIG).

Con la evolución acelerada de los SIG, estos se han convertido en parte importante de las distintas esferas de la sociedad, y los resultados que con estos se pueden alcanzar ha creado la necesidad de profundizar en esta tecnología, que ha revolucionado el mundo de la ciencias, por lo que se propone varias definiciones para conocer exactamente lo que es un SIG .

Un Sistema de Información Geográfica es una integración organizada de hardware, software, datos geográficos y personal, diseñado para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y gestión. También puede definirse como un modelo de una parte de la realidad referido a un sistema de coordenadas terrestre y construido para satisfacer unas necesidades concretas de información. (Yagüez y Langhi, 2002).

1.2.1.1 Otras definiciones de SIG:

Un SIG se define (Aronoff1987) como un sistema computacional para la entrada; manejo (almacenamiento y recuperación de información), manipulación, análisis y representación de datos geográficos.

Actualmente, nuevas modificaciones a estos conceptos clásicos de SIG se han producido para destacar el papel de la diseminación de los datos como una función ineludible de los sistemas de información geográfica en ambientes distribuidos y globales de acceso de datos y en el entorno WWW (Delgado2000).

Un sistema de hardware, software y procedimientos diseñados para soportar la captura, gestión, manipulación, análisis, modelado y visualización de datos espacialmente referenciados para resolver problemas complejos de planeamiento y gestión (Cowen1989).

Después de analizar algunos de los conceptos existentes, se puede concretar qué: Un Sistema de Información Geográfica es un sistema que por medio de computadoras y datos geográficos que ayuda al mejor entendimiento del mundo real y permite resolver los problemas que a diario se presentan.

1.2.2 Datos Espaciales.

Los Datos Espaciales o Datos Cartográficos Digitales son informaciones sobre la localización y las formas de un objeto geográfico y las relaciones entre ellos, normalmente con coordenadas y topología, es el tipo de datos necesarios para crear mapas y estudiar relaciones espaciales, son el componente fundamental de SIG. Se refieren a entidades o fenómenos que cumplen los siguientes principios básicos:

- **Posición absoluta:** sobre un sistema de coordenadas (x, y, z).
- **Posición relativa:** frente a otros elementos del paisaje (topología: incluido, adyacente, cruzado, etc.)
- Figura geométrica que representan (punto, línea, polígono)
- Atributos que describen (características del elemento o fenómeno).

“Es evidente que los datos son el principal activo de cualquier sistema de información. Por ello el éxito y la eficacia de un S.I.G. se miden por el tipo, la calidad y vigencia de los datos con los que opera” (Linde, 2000).

1.2.3 Metadatos.

La palabra Metadatos del griego “Meta” significa cambio y “Datos” describe los orígenes y rastrea los cambios hacia los datos. También es utilizado para describir información resumida o características de un conjunto de datos.

El término metadato se define como un conjunto formalizado de propiedades descriptivas que es compartido por una comunidad, que incluye una guía sobre supuestas, definiciones, capacidad de repetición y condicionalidad de los elementos.

1.2.3.1 Metadatos Geográficos.

Los Metadatos Geográficos son los recursos que permiten la descripción, clasificación y recuperación de la información Geoespacial, y por lo tanto constituyen, conjuntamente con los Datos, uno de los componentes fundamentales de la Infraestructura de Datos Espaciales (IDE).

1.2.4 Catálogo de Datos Geográficos.

Uno de los principales componentes necesarios para la construcción de una infraestructura espacial de información geográfica, son los catálogos de información geográfica, que permiten que usuarios o aplicaciones software que puedan buscar la información existente en algún lugar dentro de un entorno computacional distribuido. Los catálogos son la solución para publicar descripciones de datos geográficos mediante métodos estándar que posibilitan la realización de búsquedas a lo largo de múltiples servidores.

Los catálogos son considerados como el corazón de una infraestructura de datos espaciales, puesto que en ellos se almacenan los metadatos que permiten las búsquedas y el manejo de la información espacial.

El consorcio OpenGeoespacial(OGC, anteriormente denominado Open GIS Consortium,[OGC,1996], define un catálogo como el conjunto de servicios que soporta la organización, la búsqueda y el acceso a la información espacial.

Se puede pensar en que un catálogo como una base de datos especializada de información referente a recursos geoespaciales (metadatos) disponibles para una comunidad de usuarios. (NOGUERAS et al., 2005).

1.2.5 Cartografía.

La cartografía (del griego *chartis* = mapa y *graphein* = escrito) es la ciencia que se encarga del estudio y de la elaboración de los mapas geográficos, territoriales y de diferentes dimensiones lineales. También se denomina cartografía a un conjunto de documentos territoriales referidos a un ámbito concreto de estudio.

La cartografía constituye un conjunto de operaciones que permiten a partir de observaciones y mediciones, la representación de una parte o la totalidad de la Tierra. La representación de toda o parte de la superficie terrestre en una superficie plana constituye un mapa.

1.3 Descripción General del Objeto de estudio.

1.3.1 La representación de los datos en los diferentes SIG.

A través del estudio de los SIG se puede identificar dos grandes categorías: los SIG raster y los SIG vectoriales, los cuales se diferencian por el tipo de datos que tratan.

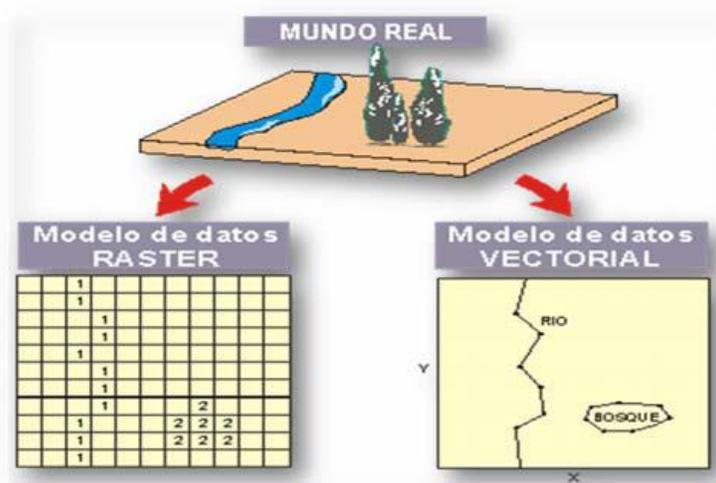


Figura 1. Modelo de datos de un SIG.

Los tipos de datos raster pueden ser cualquier tipo de imagen digital representada en mallas de rectángulos o cuadrados a los que se les denomina células o retículas, cada retícula posee información alfanumérica asociada que representa las características de la zona o superficie geográfica que cubre. Este tipo de archivo divide el espacio en celdas regulares donde cada una de ellas representa un único valor, se componen de filas y columnas de celdas.

Las estructuras raster pueden implicar en ocasiones un incremento del espacio de almacenamiento, estas almacenan en cada una de las celdas de la matriz, una entidad o simplemente de un espacio vacío sin tener en cuenta el tipo.

Los datos raster se almacenan en diferentes formatos, desde un archivo estándar basado en la estructura de JPG, PNG, GIF, TIFF/GeoTIFF [TIFF], EPPL7 y otros vía GDAL.

En un modelo raster cuanto mayor sean las dimensiones de la celda menor es la precisión o detalle de resolución de la representación del espacio geográfico. Dicho formato se centra en las propiedades del espacio más que en la precisión de la localización.

El modelo raster está compuesto por dos estructuras:

La enumeración exhaustiva es una estructura de datos, donde el valor de cada píxel se registra individualmente, de forma que no se aplica ningún método de compresión cuando el mismo valor numérico aparece reiteradas veces seguidas.



Figura 2. Representación exhaustiva.

La segunda estructura utilizada por este modelo es Run-length encoding (funcionar-longitud de codificación) que es un método de compresión de imágenes. En el caso de que existan celdas contiguas con valores numéricos idénticos, esta estructura compacta la información. En vez de registrar el valor de cada celda individualmente, para cada fila se recoge el valor temático que existe y el número de celdas con dicho valor.

Si sólo existe una celda con ese valor el tamaño se duplica, pero se reduce considerablemente en el caso de tres o más celdas idénticas. Cuanto mayor y más frecuentes sean las series de datos repetitivos, mayor compresión se logrará. Esta técnica es especialmente útil para codificar imágenes monocromo o binarias (Chrisman, 1997).

representación raster

A	A	A	A	0	0	0	0
A	A	A	A	A	0	0	0
A	A	A	A	0	B	0	0
A	A	A	A	0	0	0	0
A	A	A	0	0	0	C	C
0	0	0	0	0	C	0	0
C	C	C	C	C	0	0	0
0	0	0	0	0	0	0	0

A,	4	0,	8				
A,	5	0,	8				
A,	4	0,	5	B,	6	0,	8
A,	4	0,	8				
A,	3	0,	6	C,	8		
0,	5	C,	6	0,	8		
C,	5	0,	8				
0,	8						

Figura 3.Run-length encoding.

Los SIG raster en el manejo de sus datos poseen grandes ventajas por ejemplo: las estructuras y las operaciones de superposición son sencillas y simples y sus formatos son óptimos para las variaciones altas de datos .

El formato raster se obtiene cuando se "digitaliza" un mapa o una fotografía o cuando se obtienen imágenes digitales capturadas por satélites. En ambos casos se obtiene un archivo digital de esa información. La captura de la información en este formato se hace mediante los siguientes medios: scanner, imágenes de satélite, fotografía aérea, cámaras de video entre otros.

El modelo vectorial es una estructura de datos utilizada para almacenar datos geográficos. Los datos vectoriales constan de líneas o arcos, definidos por sus puntos de inicio y fin, y los puntos, son donde se cruzan varios arcos y los nodos. La localización de los nodos y la estructura topológica se almacena de forma explícita. Las entidades quedan definidas por sus límites solamente y los segmentos curvos se representan como una serie de arcos conectados. El almacenamiento de los vectores implica el almacenamiento explícito de la topología, sin embargo solo almacenan aquellos puntos que definen las entidades y el resto no es considerado.

Un SIG vectorial se define por la representación vectorial de sus datos geográficos. De acuerdo a las peculiaridades de este modelo de datos, los objetos geográficos se representan explícitamente y, junto a sus características espaciales, se asocian sus valores temáticos.

Hay dos formas de organizar esta base de datos doble (espacial y temático). Normalmente, los sistemas vectoriales tienen dos componentes: uno que almacena los datos espaciales y otro los datos temáticos. A éste se le denomina sistema de organización híbrido, por unir una base de datos relacional, para los aspectos temáticos, con una base de datos topológica, para los geográficos. Un elemento clave en este tipo de sistemas es el identificador de cada objeto. Éste es único y diferente para cada objeto y permite la conexión entre ambas bases de datos.

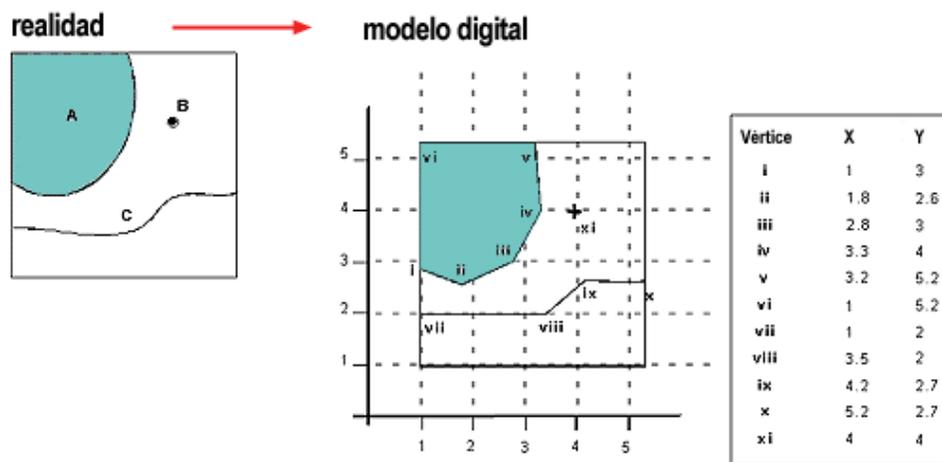


Figura 4. Representación Vectorial.

En los SIG vectoriales, representan la información por medio de pares ordenados de coordenadas, este ordenamiento da lugar a las entidades universales con las que se representan los objetos gráficos, sus datos se centran principalmente en la precisión de la localización de los elementos geográficos sobre el espacio.

Los datos vectoriales se representan a través tres elementos geométricos: el punto, la línea y el polígono. Los puntos se utilizan para aquellas entidades geográficas que mejor pueden ser expresadas por un único punto de referencia. Los mismos transmiten la menor cantidad de información de estos tipos de archivo y no son posibles las mediciones. También se pueden utilizar para representar zonas a una escala pequeña.

Las líneas se utilizan en pequeñas escalas para representar polígonos, estos elementos lineales pueden medir distancia y son usadas para rasgos lineales: como curvas de nivel, ferrocarriles etc.

La última entidad que conforman los datos vectoriales son: los polígonos bidimensionales que tienen gran utilidad ya que representan elementos geográficos que cubren un área particular de la tierra, los mismos transmiten el mayor volumen de información en archivos vectoriales y en ellos se puede medir el perímetro y el área.

Los datos vectoriales se almacenan en diferentes formatos ESRI shapefiles, PostGIS/SRI, ArcSDE, GML y otros muchos vía OGR.

Existen diferentes estructuras de datos vectoriales. Cada una de ellas tiene diferentes ventajas e inconvenientes.

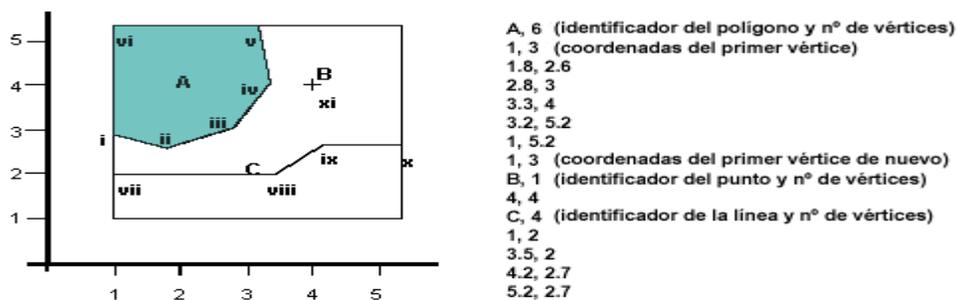


Figura 5. Lista de coordenadas espagueti.

Las listas de coordenadas son estructuras fáciles de operar, sencillas, no almacenan topología, son muy utilizadas en la cartografía automática.

El diccionario de vértices es otra de las estructuras del modelo vectorial en el mismo no existen operaciones redundantes, pero tampoco almacena la topología.

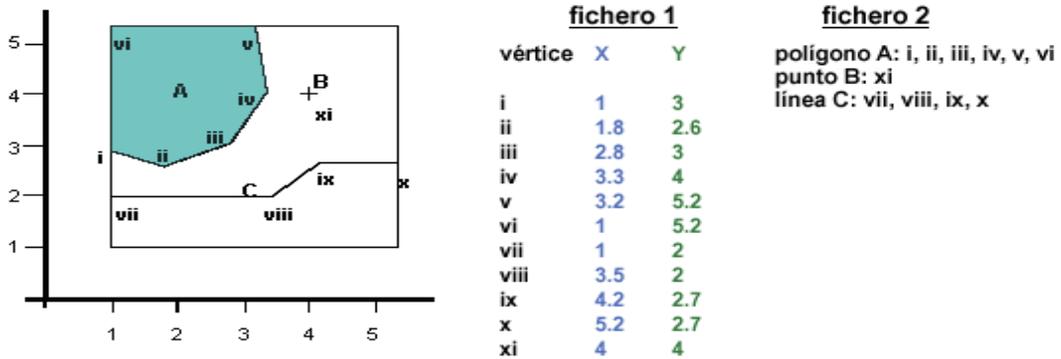


Figura 6. Diccionario de vértices.

Esta es la tercera estructura, que está conformada por nodos (intersecciones de líneas). Se identifican con códigos, a los nodos asignándoles un código direccional de la forma "from node" (nodo origen) y "to node" (nodo final).

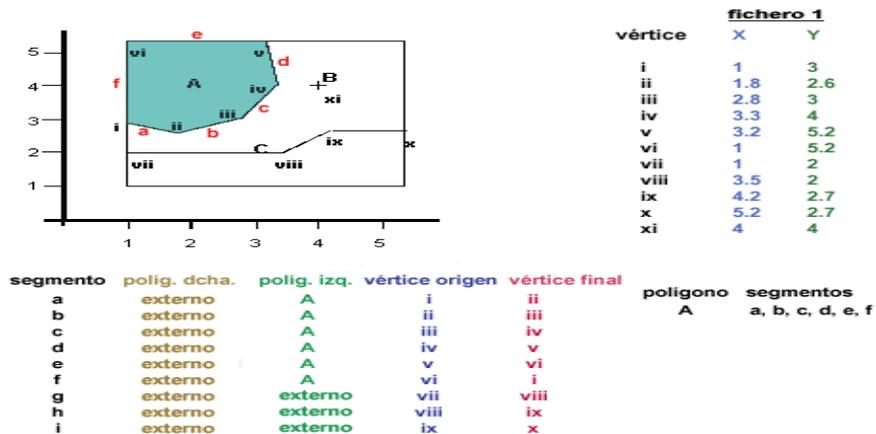


Figura 7. Ficheros DIME.

La última estructura del modelo vectorial son los **arco / nodo**.

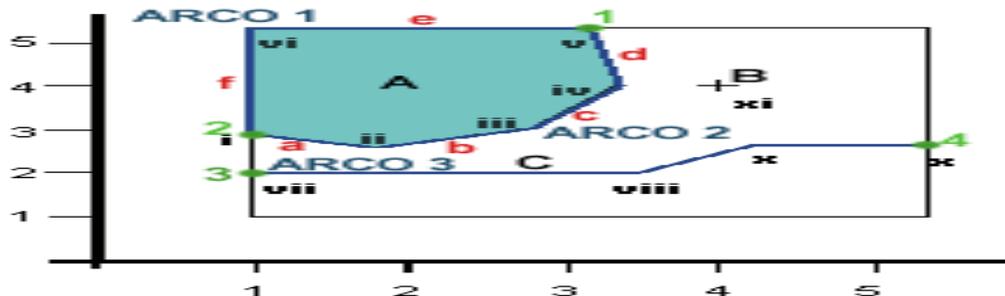


Figura 8. Arco / Nodo.

Algunas ventajas que posee este modelo es que utiliza una estructura compacta, sus datos solamente almacenan elementos digitalizados porque lo que necesita menos memoria para su tratamiento. Estos datos representan a sus elementos como gráficos vectoriales los cuales no pierden definición si se amplía la escala de visualización.

1.3.2 Tipos de Cartografía Digital.

Un elemento básico para el éxito de un Sistema de Información Geográfica es la cartografía o los mapas digitales que soportan las operaciones de un SIG, puede proceder de la digitalización de un mapa de papel (cartografía “a medida”) o de ficheros de la imagen de un mapa.

La oferta de cartografía digital en Internet es enorme y está en continuo crecimiento. La cartografía digital puede descargarse desde la web, servidores cartográficos, páginas personales, programas de compartir ficheros. A través de las investigaciones se ha llegado a la conclusión que la cartografía digital tiene dos grandes clasificaciones las cartografía raster y las cartografía vectoriales.

La cartografía raster es una imagen del mapa en formato TIF-Georeferenciado, se procede a escanearla y se hacen las correcciones geométrica, para que se puedan medir las coordenadas correctamente.

La cartografía vectorial, están compuesta por archivos vectoriales, que se utilizan en potentes aplicaciones profesionales de cartografía SIG. Los ficheros SIG vectoriales almacenan diferentes informaciones como si fueran “capas”. Cada capa contiene datos, procedentes de distintas fuentes u organismos, de todo tipo: geográficos, geológicos, sociales, medioambientales o biológicos. Estos datos se pueden visualizar, manipular, analizar o extrapolar individual o colectivamente.

1.4 Situación Problemática.

En la actualidad los Sistemas de Información Geográfica han adquirido una demanda extraordinaria en el mercado internacional, este ha llevado a que las grandes empresas de software, sientan la necesidad de hacerle frente a estas peticiones, que cada día continúan acrecentándose de una forma vertiginosa.

La Universidad de las Ciencias Informáticas no se ve exenta de este proceso que se está desarrollando en el mundo informático y de inmediato se da a la tarea de crear un espacio para el estudio de los SIG .

La responsabilidad recae Departamento de Geoinformática que pertenece al Centro de Desarrollo de Geoinformática y Señales Digitales (GEYSED) de la Facultad 6, en el cual se encuentra el proyecto productivo SIG-Desktop, siendo su principal tarea la creación de aplicaciones SIG de escritorio bajo tecnologías de licencia libre .

El proyecto trabaja arduamente en el desarrollo de un SIG de escritorio, todo este proceso utiliza un gran número de cartografías digitales y a medida que crezca el interés de los clientes aumentará considerablemente el número de mapas a manejar, generando un volumen importante de formatos contenedores de cartografías digitales tanto raster como vectoriales.

Hoy en día la búsqueda y clasificación de dichos formatos se realiza de forma no automatizada, lo que provoca grandes atrasos en producción y en el cronograma planificado, al existir gran cantidad de archivos raster y vectoriales que constantemente se van generando.

Para darle solución a la problemática, el proyecto SIG-Desktop propone desarrollar una herramienta informática que permita describir y relacionar de manera ordenada los archivos contenedores de cartografías digitales para su posterior representación, que permita encontrar de manera sencilla todos los formatos geográficos disponibles y sus características, lo que facilitaría una manera rápida de efectuar análisis, que hechos manualmente resultarían largos y engorroso. De esta forma se lograría la minimización de costos de operación e incremento de la productividad.

1.5 Análisis de otras soluciones existentes.

ArcCatalog es una aplicación propietaria que ayuda a organizar y administrar todos los datos SIG como los mapas, globos, conjuntos de datos, modelos, metadatos, y servicios. Incluye la búsqueda y hallazgo

de información geográfica; el registro, visualización, y manejo de metadatos, definición, la búsqueda y descubrimiento de datos SIG en las redes locales y la Web.

ArcCatalog permite buscar, visualizar el documento y organizar los datos geográficos. La razón más importante para el uso ArcCatalog es para la gestión de archivos. Fue desarrollada en una versión atrasada de la norma ISO 19115, por lo que los metadatos creados no son compatibles con otras herramientas que utilizan la versión actualizada del estándar.

GeoNetwork es una aplicación de catálogo para administrar los recursos con referencias espaciales. Se prevé la edición de metadatos de gran alcance y las funciones de búsquedas, así como un visualizador interactivo de mapa web. Se utiliza actualmente en numerosas iniciativas de infraestructura de datos espaciales en todo el mundo.

GeoNetwork ha sido desarrollado para conectar comunidades de información espacial y sus datos son usados en una arquitectura moderna, que es al mismo tiempo potente y de bajo costo, basada en los estándares internacionales y abiertos para servicios y protocolos (de ISO/TC211 y OGC). El software posibilita utilizar la interfaz web para la búsqueda de datos geoespaciales a través de múltiples catálogos, combinando y distribuyendo los servicios de mapas en el visualizador de mapas integrados, otra facilidad que brinda es publicar datos geoespaciales utilizando las herramientas de edición de metadatos en línea y opcionalmente el servidor incrustado GeoServer mapa.

LiberMaps: es una aplicación diseñada con el objetivo de brindar servicios de catálogo de mapas en la web, con el fin de gestionar las propiedades de los mapas y configurarlos en función de las necesidades del usuario. Al integrar el catálogo con la plataforma de desarrollo GeneSig se logra personalizar el sistema a los usuarios de acuerdo al perfil de cada uno, controlando el acceso a los datos sensibles y a las funcionalidades disponibles en cada caso. El usuario puede además configurar su perfil y definir de los datos a los que tiene acceso cuáles desea tener visibles (generalmente aquellas capas temáticas que utiliza con más frecuencia). Esto le permite adaptar el entorno de trabajo a sus necesidades inmediatas. LiberMaps permite personalizar la información georreferenciada de acuerdo a la necesidad del usuario, además de brindar mecanismos de seguridad sobre los datos.

Permite mantener el control de cada objeto de un fichero.map de MapServer, y además posibilita la personalización de los mapas de acuerdo a perfiles definidos por usuarios y roles, garantizando de esta forma la protección en su consulta y manipulación. Es una herramienta útil para integrar con los Sistemas

de Información Geográfica (SIG) que utilizan como servidor de mapas a MapServer. Este sistema está desarrollado en un ambiente web con tecnologías libres.

Luego de realizar el análisis de las diferentes herramientas se decide descartar ArcCatalog debido a su condición de ser una herramienta propietaria, la cual no permite utilizar su código y va contra todos los principios que actualmente se abogan, así mismo pasa con LiberMaps a pesar de estar desarrollada con licencias libres, es un sistema sustentado sobre tecnologías web y no se podrán utilizar para el desarrollo de la aplicación. Ninguna de las herramientas planteadas reúne las condiciones necesarias para ser el punto de partida para el desarrollo de la aplicación.

Actualmente no existe un software de escritorio, que se encargue del trabajo con los archivos raster y vectoriales por lo que se propone la construcción de esta aplicación que agilizará los procesos de análisis y clasificación de dichos archivos como parte de unos de los módulos del GeoQ, que es el corazón del proyecto SIG-Desktop.

1.6 Conclusiones Parciales.

Luego de realizar el estudio de los conceptos y definiciones más importantes sobre el contexto de la investigación, se logró un mejor entendimiento de los mismos, lo que garantiza continuar con el proceso investigativo. Se efectuó un estudio a nivel mundial de las diferentes herramientas que permiten catalogar los archivos raster y vectoriales, arrojando que no existe actualmente, ningún software de escritorio que brinde estas soluciones, por lo que se fundamenta, proponer una herramienta en Software Libre, para la automatización del proceso de clasificación de archivos cartográficos tanto raster como vectoriales en un SIG de escritorio.

CAPÍTULO 2: Tendencias y tecnologías actuales.

2.1 Introducción.

Este capítulo tiene como objetivo seleccionar las herramientas correctas que garanticen, construir un software de alta calidad, que se ajuste a las necesidades del proyecto, teniendo en cuenta estas premisas se realiza un análisis minucioso de las diferentes metodologías de software que existen a nivel mundial, los lenguaje de modelado, los lenguajes de programación, los gestores de bases de datos, entre otros aspectos que conformarán la propuesta de solución esta investigación.

2.2 Metodología de desarrollo de software: Rational Unified Process (RUP).

La metodología de desarrollo de software (RUP), proporciona disciplinas, en las cuales se encuentran artefactos con lo cual se podrá contar con guías para poder documentar e implementar de una manera fácil y eficiente, todas las guías para un buen desarrollo, todo esto dentro de las respectivas fases con las cuales cuenta. (CHACÓN, 2006).

Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

RUP unifica todo el equipo de desarrollo de software y optimiza su comunicación proporcionando a cada miembro de una aproximación al desarrollo de software con una base de conocimiento en línea adaptable de acuerdo a las necesidades específicas del proyecto. Usando la navegación en línea del buscador, cada miembro del equipo tiene acceso instantáneo a la base de conocimiento y guía de procesos del RUP desde su escritorio.

RUP es un proceso para el desarrollo de un proyecto de software que define claramente quién, cómo, cuándo y qué debe hacerse en el proyecto, con 3 características esenciales, está dirigido por los casos de usos: que orientan al proyecto a la importancia para los usuarios y lo que este quiere; está centrado en la arquitectura: que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden; y es iterativo e incremental: dividiéndose el proyecto en mini proyectos .

La metodología RUP está compuesta por 4 fases fundamentales:

- **Concepción:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **Elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **Construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- **Transición:** se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

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

En RUP se han agrupado las actividades en grupos lógicos definiendo nueve flujos de trabajo principales. Los seis primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo. (CHACÓN, 2006).

1. Modelado del negocio.
2. Requisitos.
3. Análisis y diseño.
4. Implementación.
5. Prueba.
6. Despliegue.
7. Configuración y manejo de cambios.
8. Administración del proyecto.
9. Gestión del entorno.

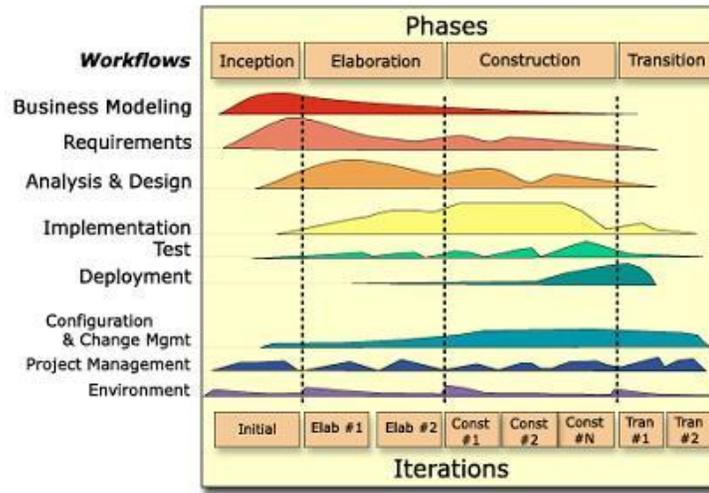


Figura 9. Fases e Iteraciones de la Metodología RUP.

RUP es la metodología, más factible para llevar a cabo el desarrollo de software de esta investigación. Sus características se adaptan correctamente a las condiciones del proyecto, ya que con solo la plataforma RUP proporciona un framework de proceso configurable que permite seleccionar e implantar los componentes específicos necesarios para proporcionar un proceso consistente y personalizado para el proyecto a través de sus fases y flujos de trabajos. RUP define, para cada etapa y cada disciplina, el flujo de trabajo, los trabajadores que intervienen, las actividades que realizan y los artefactos que se necesitan o producen con el objetivo de asegurar un software con la más alta calidad, que reúna las necesidades de los usuarios dentro del cronograma planeado y la inversión prevista. Dicha metodología proporciona una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas de desarrollo que permiten desplegar de manera efectiva cada paso que se dé para el desarrollo de la aplicación ya que define claramente quién, cómo, cuándo y qué debe hacerse en el proyecto para llevar a cabo un buen desarrollo de software que satisfaga, al finalizar la aplicación, las necesidades del cliente, que constituye la premisa fundamental del proyecto.

2.3 Lenguaje de Modelación: Unified Modeling Language (UML).

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

UML es un lenguaje que permite visualizar, especificar, construir y documentar los artefactos de un sistema de software. A través de este lenguaje se puede modelar, lo mismo sistemas de software, hardware u organizaciones del mundo real. Este modelo está compuesto por tres clases de bloques de construcción: Elementos, Relaciones y Diagramas.

Algunas características esenciales del lenguaje UML son:

- Concurrencia, es un lenguaje distribuido y adecuado a las necesidades de conectividad actuales y futuras.
- Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- Modela estructuras complejas.
- Las estructuras más importantes que soportan tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- Emplea operaciones abstractas como guía para variaciones futuras, añadiendo variables si es necesario.
- Comportamiento del sistema: casos de uso, diagramas de secuencia y de colaboraciones, que sirven para evaluar el estado de las máquinas.

UML constituye una de las innovaciones conceptuales más importantes en el mundo tecnológico del desarrollo de software que más expectativas ha generado a lo largo de muchos años, comparable con la aparición e implantación de los lenguajes COBOL8, Basic, Pascal, C++, y actualmente con los más recientes Java, XML, C#.

Se considera el lenguaje estándar para el análisis y diseño de sistemas de cómputo. Entre más complejo es el sistema que se desea crear más beneficios presenta el uso de UML, las razones de esto son evidentes:

- Diseño y documentación
- Código reutilizable.
- Descubrimiento de fallas.
- Ahorro de tiempo en el desarrollo del software.

- Mucho más fáciles las modificaciones.
- Más fácil la comunicación entre programadores.

Es importante resaltar que UML es un lenguaje para especificar y no para describir métodos o procesos. Además UML es desde finales de 1997, orientado a objeto de acuerdo con el Object Management Group, siendo utilizado diariamente por grandes empresas de desarrollo de software tales como: Microsoft, Oracle, Rational.

2.5 Herramienta CASE: Visual Paradigm.

Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite generar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación del proyecto automáticamente en varios formatos como HTML o PDF, y permite control de versiones. El Visual Paradigm tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de clases. Además, la herramienta es colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto.

Características principales de la herramienta:

- Producto de calidad.
- Soporta aplicaciones web.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones

Posee 7 ediciones: Enterprise Edition, Professional Edition, Standard Edition, Modeler Edition, Personal Edition, Community Edition y Viewer Edition. Soporta patrones de diseño, conexión a repositorios como el CVS y el Subversion, integración a varios Entornos Integrados de Desarrollo, (Integrated Development Environment, IDE) como el NetBeans, el Borland JBuilder y el Eclipse. Facilita la ingeniería inversa de varios códigos fuentes a través del Instant Reverse, como por ejemplo de Java, C++, .NET, XML y PHP

5.0, además brinda la posibilidad de generar el código de varios lenguajes incluyendo los mencionados anteriormente.

Visual Paradigm posee interfaces intuitivas agrupando los diagramas por categorías. El usuario puede configurar el estilo y el formato de los diagramas incorporando imágenes y estereotipos. Posibilita la interoperabilidad permitiendo importar proyectos de Visual Paradigm, Rational Rose, Erwin, genera imágenes en varios formatos como JPG y PNG, es soportado en varias plataformas como Windows y Linux, y además posee una arquitectura abierta a través del soporte de plug-ins para Java.

El Visual Paradigm, es una herramienta desarrollada con tecnologías libres, factor fundamental por lo que aboga la Universidad para garantizar la soberanía e independencia tecnológica, multiplataforma, y se ajusta perfectamente a las necesidades de este trabajo. Además posibilita la creación de diagramas para UML; diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad; capacidades de ingeniería directa e inversa; modelo y código que permanece sincronizado en todo el ciclo de desarrollo; disponibilidad de múltiples versiones y posibilita integrarse en los principales IDE.

2.6 Sistema de Base Datos: SQLite.

SQLite es un sistema de gestión de bases de datos relacional contenida en una relativamente pequeña (~275 kib) biblioteca en C. SQLite creado por D. Richard Hipp, es un dominio gratuito, público, herramienta de código abierto visual utilizado para crear, diseñar y editar archivos de base de datos. La biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de base de datos al principio de cada transacción.

Características esenciales de SQLite:

SQLite usa un sistema de tipos inusual. En lugar de asignar un tipo a una columna como en la mayor parte de los sistemas de bases de datos SQL, los tipos se asignan a los valores individuales, se puede insertar un string en una columna de tipo entero (a pesar de que SQLite tratará en primera instancia de convertir la cadena en un entero). Algunos usuarios consideran esto como una innovación que hace que la

base de datos sea mucho más útil, sobre todo al ser utilizada desde un lenguaje de scripting de tipos dinámicos.

Para el acceso a la información de manera rápida, eficiente y actualizada, se necesita tener una base de datos correctamente diseñada, pero a su vez se requiere un eficaz sistema de gestión de base de datos para complementar sus funciones. Por lo que se decide utilizar SQLite, ya que constituye la herramienta apropiada para el manejo y control de la información que se desea almacenar. La base de datos de la aplicación no constará de numerosas tablas por lo que no se requiere de un gestor más potente, en cuanto el tamaño de almacenamiento. Con SQLite se garantiza la persistencia de los datos ya que permite guardar como un sólo fichero toda la información asociada y cargarlo en la memoria de la máquina, permitiendo realizar de manera efectiva todas las peticiones de los usuarios. El sistema de gestión de base datos, es una herramienta desarrollada con tecnologías libres, multiplataforma y responde a las peticiones que pretende llevar a cabo Cuba, para garantizar la plena soberanía tecnológica.

2.7 Lenguaje de Programación: C++.

C++ es un lenguaje imperativo orientado a objetos derivado del C fue creado en los años 80 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. Posteriormente se añadieron facilidades de programación genérica, que se sumó a los otros dos paradigmas que ya estaban admitidos (programación estructurada y la programación orientada a objetos). Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma.

Dicho lenguaje que se han añadido nuevos tipos de datos, clases, plantillas, mecanismo de excepciones, sistema de espacios de nombres, funciones inline, sobrecarga de operadores, referencias, operadores para manejo de memoria persistente, y algunas utilidades adicionales de librería (en realidad la librería Estándar C es un subconjunto de la librería C++).

Respecto a su antecesor, se ha procurado mantener una exquisita compatibilidad hacia atrás por dos razones: poder reutilizar la enorme cantidad de código C existente, y facilitar una transición lo más fluida posible a los programadores de C clásico, de forma que pudieran pasar sus programas a C++ e ir modificándolos (haciéndolos más "++") de forma gradual. De hecho, los primeros compiladores C++ lo que

hacían en realidad era traducir (pre procesar) a C y compilar después (las consecuencias se dejan sentir todavía en el lenguaje).

Una particularidad del C++ es la posibilidad de redefinir los operadores (sobrecarga de operadores), y de poder crear nuevos tipos que se comporten como tipos fundamentales.

La Librería Estándar C++ se ha usado ampliamente, esta dualidad (ser mezcla de un lenguaje tradicional con elementos de POO), lo que ha permitido un modelo muy avanzado de programación.

A pesar de todo, ha experimentado un extraordinario éxito desde su creación. De hecho, muchos sistemas operativos compiladores e intérpretes han sido escritos en C++ (el propio Windows y Java).

Tanto sus fervientes defensores como sus acérrimos detractores han hecho correr ríos de tinta ensalzando sus cualidades o subrayando sus miserias, aunque todo el mundo parece estar de acuerdo en que es largo y complejo. Ha servido de justificación para el diseño de otros lenguajes que intentan eliminar sus inconvenientes como por ejemplo C# y Java, y una de sus última incorporaciones, las plantillas ha sido origen de un nuevo paradigma de programación (meta programación).

El lenguaje de programación con que se llevará a cabo el desarrollo de la aplicación es C++, que es catalogado como un lenguaje de programación multiparadigma de gran aceptación a nivel mundial desde su creación, que ha permitido un modelo muy avanzado de programación genérica. Cumple también con el paradigma de programación orientada a objetos y posee además una gran cantidad de documentación en la red y una numerosa comunidad de desarrollo en todo el mundo para su constante actualización. Una de las principales razones que permitió la selección de dicho lenguaje, es que la plataforma Geoq que constituye el producto principal del proyecto SIG-Desktop está implementada en C++ y el plugins a desarrollar será una extensión de dicha plataforma, que permitirá el trabajo con los archivos raster y vectoriales. De esta manera existirá plena uniformidad y compatibilidad entre las dos aplicaciones.

2.8 Entorno de Trabajo: Qt. Creator.

Qt. es una biblioteca de software que desarrolla Nokia para crear interfaces gráficas de usuario. Aplicaciones como: Google Earth, Skype, Adobe Photoshop Álbum y VirtualBox entre muchas otras, hacen uso de ésta. Para los desarrolladores, Nokia ofrece Qt Creator, un entorno de desarrollo (IDE) multiplataforma muy completo. Qt Creator, es distribuido bajo tres tipos de licencias: Qt Commercial Developer License, Qt. GNU LGPL v. 2.1, Qt GNU GPL v. 3.0 y está disponible para las plataformas:

Linux, Mac OSX; Windows, Windows CE, Symbian y Maemo. La librería Qt posee numerosos módulos que permite trabajar con bases de datos, Gráficos, Imágenes, Sonido, etc.

Es un excelente IDE multiplataforma para el desarrollo aplicaciones en C++ de manera sencilla y rápida. Como su nombre lo indica, está basado en la librería Qt y cuenta con las siguientes características principales:

- Posee un avanzado editor de código C++.
- Además soporta los lenguajes: C#/.NET Lenguajes (Mono), Python: PyQt y PySide, Ada, Pascal, Perl, PHP y Ruby.
- Posee también una GUI integrada y diseñador de formularios.
- Herramienta para proyectos y administración.
- Ayuda sensible al contexto integrado.
- Depurador visual.
- Resaltado y auto-completado de código.
- Soporte para refactorización de código.

Editor de código avanzado.

Qt. Creator se centra en proporcionar características que ayudan a los nuevos usuarios de Qt. aprender y comenzar a desarrollar rápidamente, también aumenta la productividad de los desarrolladores con experiencia en Qt.

- Editor de código con soporte para C++, QML y ECMAScript.
- Herramientas para la rápida navegación del código.
- Resaltado de sintaxis y auto-completado de código.
- Control estático de código y estilo a medida que se escribe.
- Soporte para refactoring de código.
- Ayuda sensitiva al context.

Depurado Visual

El depurador visual (visual debugger) para C ++ es consciente de la estructura de muchas clases de Qt., lo que aumenta la capacidad de mostrar los datos de Qt con claridad.

- Interrupción de la ejecución del programa.
- Ejecución línea por línea o instrucción a instrucción.
- Puntos de interrupción (breakpoints).
- Examinar el contenido de llamadas a la pila (stack), los observadores y de la variables locales y globales.

Diseñador de Interfaces Gráficas de Usuario (GUI).

Entorno integrado para la creación y diseño de formas para proyectos C++ que permite diseñar rápidamente widgets y diálogos usando los mismos widgets que se usaran en su aplicación.

Se decide utilizar para la realización del sistema el entorno de trabajo QT Creator: es un producto desarrollado con tecnologías libres y gratuitas sin restricciones de uso. Además de ser un potente IDE, posibilita el desarrollo de aplicaciones en C++ de manera rápida y efectiva, aspecto favorable, ya existe plena compatibilidad entre el lenguaje de programación y el entorno de trabajo. Permite construir aplicaciones completas para el cliente, brindando opciones de crear ventanas, menús, barras de herramientas y otras acciones de manera fácil, brindando mayores facilidades para el diseño de aplicaciones de escritorio.

2.9 Conclusiones Parciales.

En este capítulo realizó un análisis minucioso de las tendencias tecnológicas actuales a nivel mundial, con el objetivo de seleccionar las herramientas adecuadas para el desarrollo de la aplicación basándose en las características de las mismas y se arriba a la conclusión de que la propuesta como Metodología sería RUP para su desarrollo, UML como lenguaje de modelado, Visual Paradigm como herramienta CASE, SQLite como sistema gestión de bases de datos y C++ como lenguaje de programación, teniendo como premisa fundamental que las herramientas cumplirán con la condición de ser desarrolladas con tecnologías libres para garantizar la política de migración que se está llevando a cabo en Cuba .

CAPÍTULO 3: Presentación de la solución.

3.1 Introducción

En este capítulo se comienza el desarrollo de la aplicación donde se especifican las principales funcionalidades que el sistema debe cumplir para obtener resultados exitosos, se mostrarán los principales conceptos del entorno a través del modelo de dominio así como la descripción textual de cada uno de estos conceptos y se identifican y establecen un conjunto de requisitos tanto funcionales como no funcionales que constituyen la base del sistema.

3.2 Modelo de dominio.

Un modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema. (Jacobson, y otros, 2000).

Debido a que no se están definidas los procesos del negocio se propone realizar el modelo del dominio que tiene como objetivo fundamental comprender y describir las clases dentro del contexto del sistema e identificar todos los conceptos que se utilizarán en el diagrama, mediante un glosario de términos que ayudará a los usuarios a mantener una terminología común, lo que les permite compartir el conocimiento de forma más consistente.

3.2.1 Diagrama de clases del Modelo de Dominio

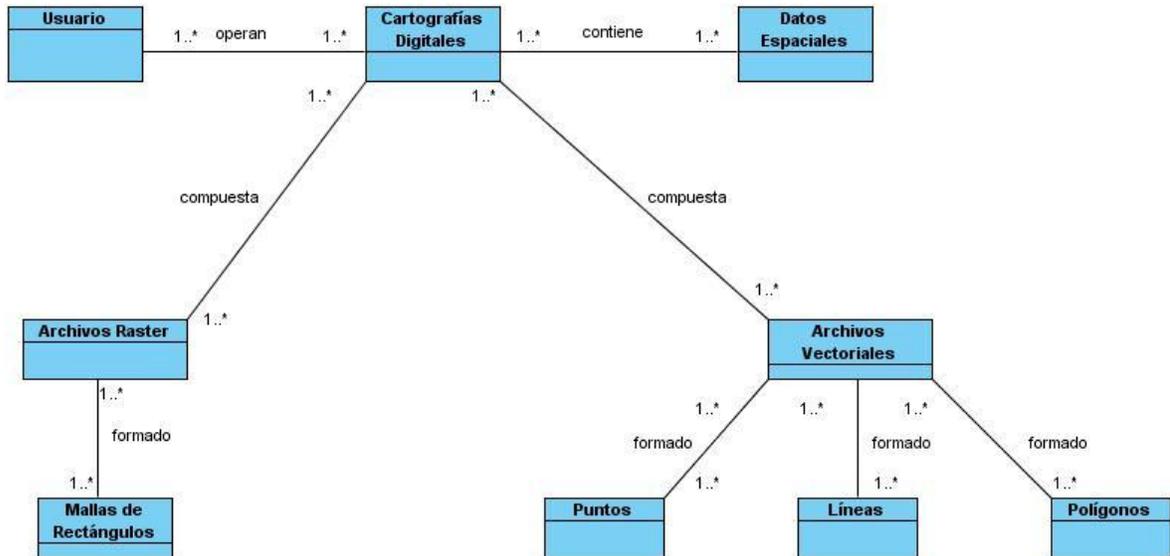


Figura 10. Diagrama de Clases del Modelo de Dominio.

3.2.2 Glosario de Términos del Dominio.

Cartografía digital:

La cartografía constituye un conjunto de operaciones que permiten a partir de observaciones y mediciones, la representación de una parte o la totalidad de la Tierra. La representación de toda o parte de la superficie terrestre en una superficie plana constituye un mapa.

Datos Espaciales:

Los Datos Espaciales o Datos Cartográficos Digitales son informaciones sobre la localización y las formas de un objeto geográfico y las relaciones entre ellos, normalmente con coordenadas y topología, es el tipo

de datos necesarios para crear mapas y estudiar relaciones espaciales, son el componente fundamental de SIG.

Archivos Raster:

Los archivos raster pueden ser cualquier tipo de imagen digital representada en mallas de rectángulos o cuadrados a los que se les denomina células o retículas, cada retícula posee información alfanumérica asociada que representa las características de la zona o superficie geográfica que cubre, divide el espacio en celdas regulares donde cada una de ellas representa un único valor, se componen de filas y columnas de celdas.

Malla de Rectángulo:

Las mallas de rectángulos o cuadrados a los que se les denominan células que pueden ser cualquier imagen digital que es la representación de un archivo raster.

Archivos Vectoriales:

Los archivos vectoriales representan la información por medio de pares ordenados de coordenadas, este ordenamiento da lugar a las entidades universales con las que se representan los objetos gráficos, sus datos se centran principalmente en la precisión de la localización de los elementos geográficos sobre el espacio. Los archivos vectoriales se representan a través tres elementos geométricos: el punto, la línea y el polígono.

Punto:

Los puntos se utilizan para aquellas entidades geográficas que mejor pueden ser expresadas por un único punto de referencia. Los mismos transmiten la menor cantidad de información de estos tipos de archivo y no son posibles las mediciones. También se pueden utilizar para representar zonas a una escala pequeña.

Líneas:

Las líneas se utilizan en pequeñas escalas para representar polígonos, estos elementos lineales pueden medir distancia y son usadas para rasgos lineales: como curvas de nivel, ferrocarriles .

Polígonos:

Los polígonos bidimensionales es la porción de plano limitada por una línea poligonal cerrada, tienen gran utilidad ya que representan elementos geográficos que cubren un área particular de la tierra, los mismos transmiten el mayor volumen de información en archivos vectoriales y en ellos se puede medir el perímetro y el área.

Usuarios:

Persona, Organismo u Organización que necesite acceder y trabajar los archivos contenedores de cartografías raster y vectoriales.

3.3 Requisitos Funcionales.

Los requisitos funcionales se pueden definir como una condición o capacidad que un usuario necesita para resolver un problema o lograr un objetivo.

A continuación se muestran los Requisitos Funcionales:

RF1. Permitir al usuario identificarse ante el sistema, este verifica la validez de los datos y da los permisos, para que puedan acceder a las secciones necesarias.

RF2. Permitir al usuario actualizar la ruta de los archivos tanto raster como vectoriales de la base de datos, donde se encuentran almacenados.

RF3. Permitir al usuario realizar Vista Previa de la información de los diferentes archivos contenedores de cartografías digitales tanto raster como vectoriales.

RF4. Permitir al usuario mostrar la información de los archivos contenedores de cartografías digitales tanto raster como vectoriales mediante la visualización de los metadatos.

RF5. Permitir al usuario categorizar los archivos por el tamaño del archivo.

RF6. Permitir al usuario categorizar los archivos en dependencia de su proveedor.

RF7. Permitir al usuario categorizar los archivos tanto raster como vectoriales por el tipo de archivo.

RF8. Permitir al usuario categorizar los archivos tanto raster como vectoriales por el tipo de almacenamiento del archivo.

RF9. Permitir al usuario categorizar los archivos raster como vectoriales por el origen del archivo.

RF10. Permitir al usuario categorizar los archivos tanto raster como vectoriales por el nombre del lugar asociado.

3.4 Requisitos no Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

A continuación se muestra los Requisitos no Funcionales:

Requisitos de interfaz externa:

El sistema debe tener una apariencia profesional y un diseño gráfico sencillo.

Requisitos de usabilidad:

Debe poseer una interfaz amigable y que muestre de forma clara todas las funcionalidades que brinda la aplicación.

Requisitos de rendimiento:

Debe tener una alta disponibilidad así como velocidad de respuesta atendiendo a la importancia de hacer eficiente y rápido el proceso de entrega y recibo de información así como la obtención de la misma.

Requisitos de soporte:

El sistema debe permitir posteriores modificaciones y actualizaciones a fin de alcanzar mayor funcionalidad o dado que cambien algunos elementos del negocio.

Requisitos de seguridad:

El sistema debe estar protegido de accesos no autorizados atendiendo al tipo de información que se maneja, por lo que los usuarios sólo podrán acceder a la información que necesite única y exclusivamente. El sistema debe proveer protección de los datos de corrupción e inconsistencias y debe garantizar el acceso a la información por los usuarios en tiempo.

Requisitos de software:

El sistema debe disponer de los Sistemas operativos GNU/Linux o se debe disponer de uno de los siguientes sistemas operativos: Windows XP, Windows 7, Windows NT, Windows 2000, Windows 2003.

Requisitos de hardware:

Se debe contar con una computadora de 1 GHz en adelante como velocidad del microprocesador y una RAM mínima de 128 MB.

3.5 Descripción del sistema propuesto.

3.5.1 Descripción de los actores del sistema.

Tabla 1: Descripción de los actores del sistema.

Actor	Descripción
Usuario	El usuario, es el encargado de realizar cualquier operación de búsqueda sobre los archivos según el criterio especificado, puede mostrar la información de los metadatos y realizar cualquier otra operación que el sistema brinda.

3.5.2 Diagrama de casos de usos del sistema.

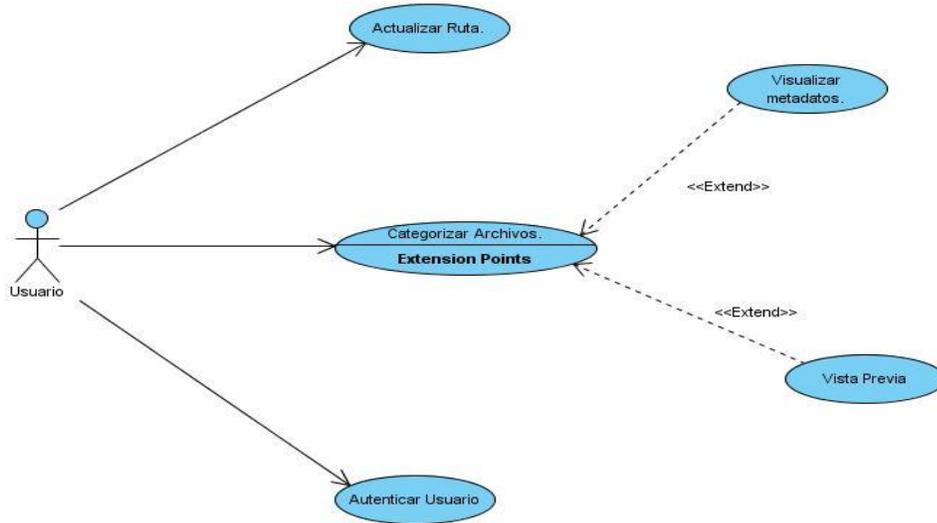


Figura 11. Diagrama de casos de usos del sistema.

3.5.3 Descripción de los casos de usos del sistema.

Tabla 2: Descripción textual de CU Categorizar Archivos.

Caso de Uso:	Categorizar Archivos.
Actores:	Usuario.
Resumen:	El CU se inicia cuando el usuario desea categorizar los archivos por las diferentes clasificaciones, para lograr obtener una mayor organización de los mismos.
Referencias	RF 5, RF 6, RF 7, RF 8, RF 9, RF 10.
Prioridad	Crítico.
<i>Flujo Normal de Eventos</i>	

Acción del Actor	Respuesta del Sistema
1. El usuario solicita la opción Categorizar Archivos.	2. El sistema ofrece un formulario con diferentes campos para realizar la categorización de los archivos: Tipo de Fichero: Raster o Vectorial, la Cantidad de Archivos a buscar, con los diferentes Criterios de Clasificación: tamaño, tipo de almacenamiento, proveedor y lugar asociado según los archivos, Criterio de Comparación, y el Valor que desea comparar.
3. El usuario llena el formulario para realizar la categorización de los archivos por el: Tipo de Fichero, Cantidad de Archivos, Criterios de Clasificación, Criterio de Comparación y Valor. 4. El usuario presiona el botón Aceptar.	5. El sistema muestra una tabla con los diferentes Criterios de Búsqueda, mediante los filtros realizados.
6. El usuario presiona el botón OK.	5. El sistema muestra una interfaz con los resultados de las búsquedas por los criterios de clasificación: tamaño, tipo de almacenamiento, proveedor y lugar asociado según los archivos.
7. El usuario presiona el botón Cancel.	

QgsCatalogue v0.1

Archivo Opciones

Construir consulta

Tipo de Fichero: **Raster** Cantidad: **0**

Campo: **Proveedor** Criterio: **=** Valor: **Geoq**

Aceptar

Criterios de Búsquedas

	Campo	Criterio	Valor
1	Proveedor	=	Geoq

Aceptar **Cancelar**

QgsCatalogue v0.1

Archivo Opciones

Resultado de la Categorización

	Nombre	Tipo	Cartografía	Lugar	Proveedor
1	Escala	Raster	UCIRaster 1	cuba	Geoq
2	leyenda	Raster	UCIRaster 1	cuba	Geoq
3	Manzanas	Raster	UCIRaster 1	cuba	Geoq

Visualizar Metadato

Flujo Alternativo de Eventos

9. El usuario desea eliminar de la Tabla de Criterios de Búsqueda, una fila que representa un criterio de búsqueda presionando el botón eliminar.	10. El sistema elimina la fila especificada por el usuario de la Tabla de Criterios de Búsqueda.
---	--

3.6 Conclusiones Parciales.

Con el desarrollo de este capítulo se generaron los artefactos correspondientes a los primeros flujo de trabajo que forman parte de la metodología de desarrollo RUP: Modelamiento del Negocio y Requisitos. Al no poder identificar correctamente los procesos del negocio se decide realizar el modelo de dominio. Se modeló el Diagrama de Casos de Uso del Sistema describiendo al Actor y Casos de Uso que forman parte de este, conjuntamente se cuenta con una documentación organizada, lo que posibilita un mayor entendimiento de los procesos a automatizar, y las características y restricciones que debe poseer el sistema para cumplir con los requisitos establecidos por el cliente. Todo lo expuesto anteriormente conlleva a que el sistema a desarrollar pueda transitar con eficiencia a la fase construcción de la misma.

Capítulo 4 Construcción de la propuesta de solución.

4.1 Introducción.

En este capítulo se detalla la construcción de la solución propuesta, a través de los flujos de Diseño e Implementación, que tiene como objetivo principal transformar los requerimientos en un diseño consecuente con la implementación del sistema, diseñando hacia un enfoque al rendimiento, así como establecer una línea base de la arquitectura que soporte a los requisitos del sistema a un coste razonable y en un plazo razonable, mediante los diagramas que constituyen la guía vital para el desarrollo correcto de la aplicación.

4.2 Diseño

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

4.2.1 Diagramas de Clases del Diseño.

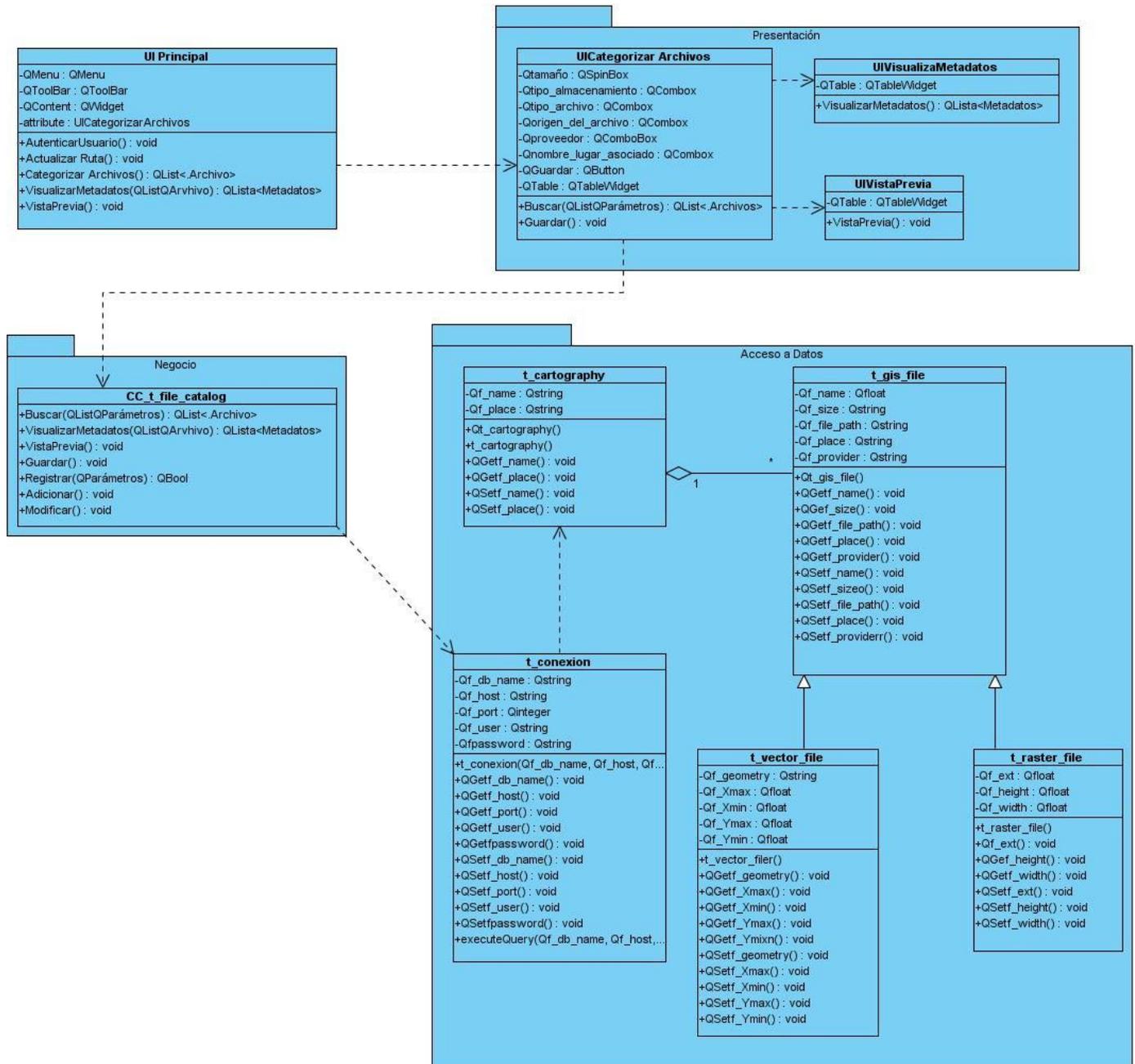


Figura 12. Diagrama del Diseño CU Categorizar Archivos.

4.3 Arquitectura de software.

La arquitectura de una aplicación es la vista conceptual de la estructura de ella misma. Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos. La arquitectura de las aplicaciones difiere según como está distribuido este código. (Cornejo 2001).

4.4 Arquitectura 3 Capas.

La arquitectura de tres capas se basa en la división en el nivel de acceso a datos, nivel de lógica de negocio y nivel de presentación o aplicación.

La capa de presentación: Esta capa se encarga de proveer una interfaz entre el sistema y el usuario. Básicamente, se responsabiliza de que se le comunique información al usuario por parte del sistema y viceversa, manteniendo una comunicación exclusiva con la capa de negocio.

La capa de negocio: Es la capa que contiene los procesos a realizar con la información recibida desde la capa de presentación, las peticiones que el usuario ha realizado, y responsabilizándose de que se le envíen las respuestas adecuadas a la capa de presentación. Se considera una capa intermedia, a medio camino entre la capa de presentación y la capa de datos, puesto que se relaciona con ambas y por supuesto, procesa también la información devuelta por la capa de datos.

La capa de acceso a datos: Por último, la capa donde se almacenan los datos. Mediante la capa de negocio, se puede encargarse de ofrecer, modificar, almacenar, borrar y recuperar datos, mediante el gestor de bases de datos que la aplicación requiera.

Las principales ventajas de este modelo son las siguientes:

Separación de funciones: esta ventaja permite que no se mezcle en una capa código correspondiente a otra.

Reutilización: el código correspondiente a una capa puede ser reutilizado desde varias partes de la capa inmediatamente superior.

Escalabilidad: Al ubicar donde está el código correspondiente a cada capa, se puede realizar modificaciones dentro de una capa para mejorar o aumentar el tamaño del sistema de software, con un mínimo impacto en las capas restantes.

Facilidad de mantenimiento: mediante esta división, es mucho más sencillo localizar errores en el código o efectuar mejoras.

4.5 Diseño de la Base de Datos.

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada, logrando que no existan datos redundantes, que aumentan la probabilidad de que se produzcan errores e incoherencias. Para eso el proceso de diseño de una base de datos se identifican todas las clases persistentes, específicamente en el modelo de clases persistentes; mostrando la estructura lógica de la base de datos mediante clases, traduciendo sus atributos a columnas de las tablas y el modelo entidad – relación, que permite definir un conjunto de conceptos, reglas y convenciones bien definidos que permiten aplicar una serie de abstracciones a fin de describir y manipular los datos de los archivos raster y vectoriales que se desea almacenar en la base de datos.

4.5.1 Diagrama de Clases Persistentes.

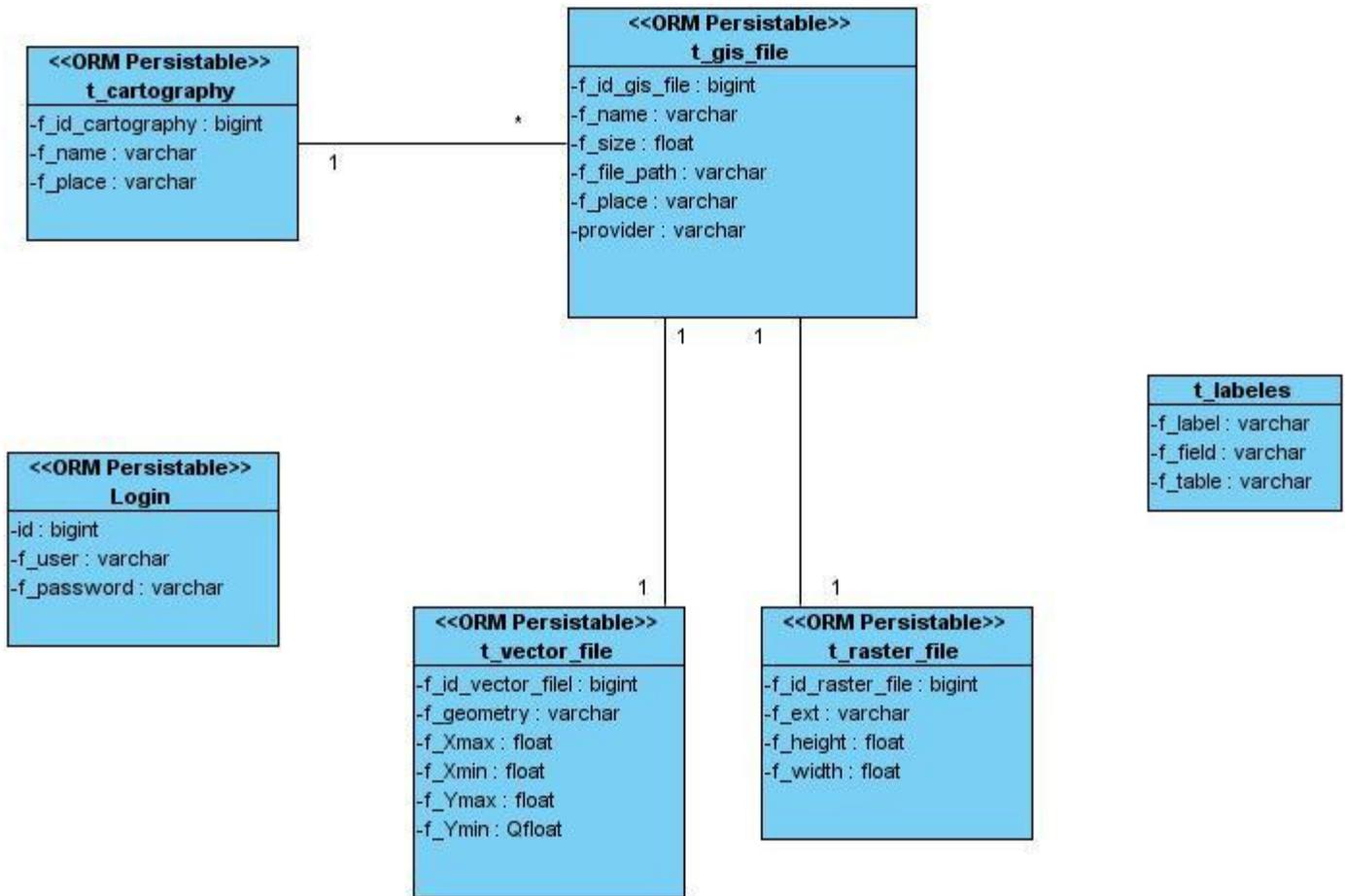


Figura 13. Diagrama de Clases Persistentes.

4.5.2 Modelo Entidad-Relación.

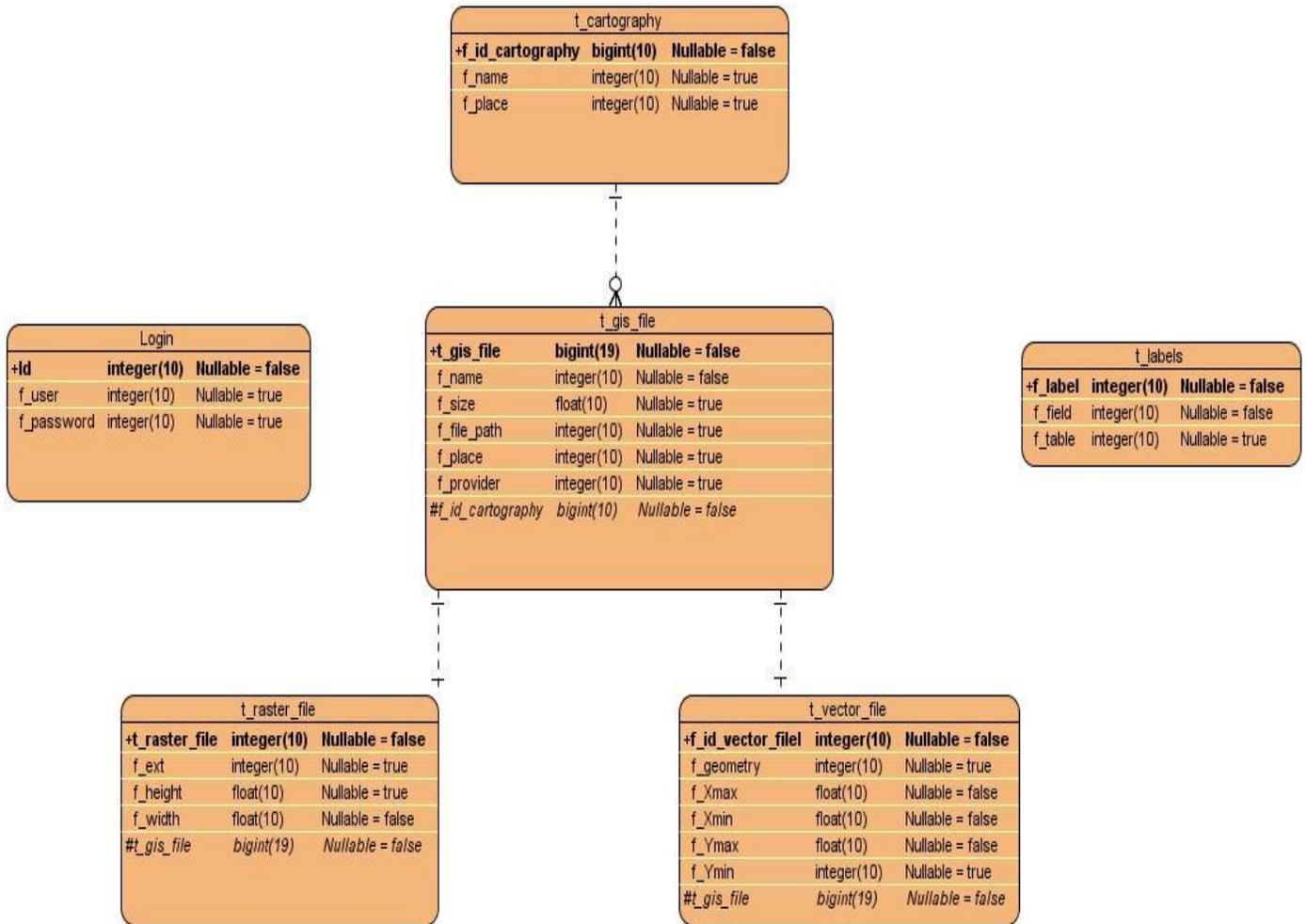


Figura 14. Diagrama de Entidad – Relación.

4.6 Modelo de Despliegue.

El diagrama de despliegue es un artefacto que modela la arquitectura en tiempo de ejecución de un sistema. En él se muestra la disposición de las particiones físicas del sistema de información y la asignación de los componentes de software a estas particiones y las relaciones físicas entre los componentes software y hardware en el sistema a entregar.

4.6.1 Diagrama de Despliegue.

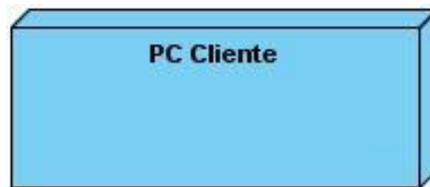


Figura 15. Diagrama de Despliegue.

4.7 Modelo de Implementación.

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

El diagrama de componentes proporciona una visión física de la construcción del sistema. Muestra la organización de los componentes software, sus interfaces y las dependencias entre ellos.

4.7.1 Diagrama de Componentes.

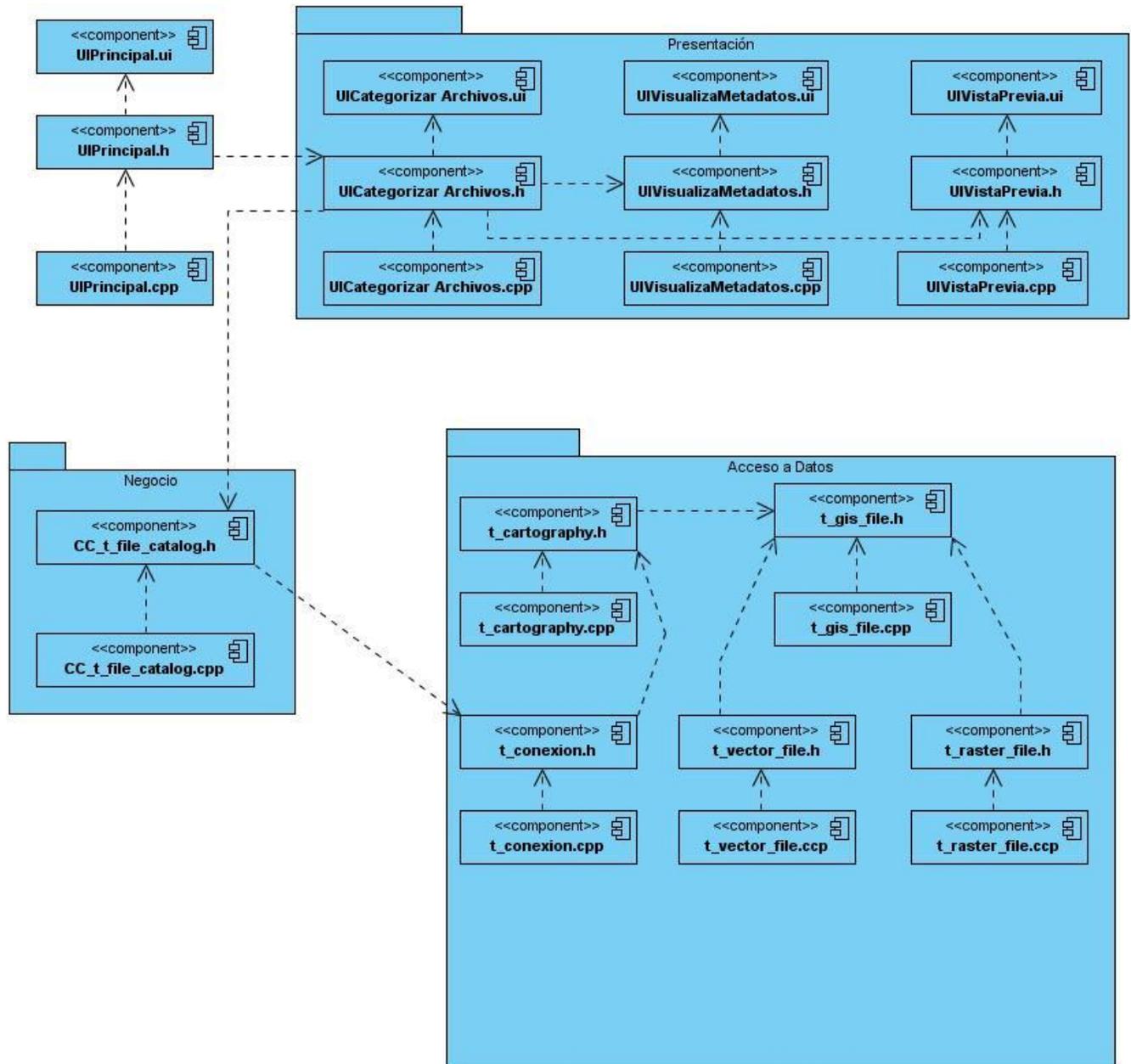


Figura 16. Diagrama de Componentes CU Categorizar Archivos.

4.8 Conclusiones Parciales.

En este capítulo arrojaron resultados importantes mediante el desarrollo de los flujos de trabajo: Diseño e Implementación, donde se generaron los artefactos pertinentes a cada uno de ellos como son el Modelo de Diseño, el cual permitió representar la estructura de la aplicación mostrando sus clases, atributos y las relaciones entre ellos. Se confeccionó el Modelo de Despliegue donde se representan la distribución física del sistema y sus conexiones, el Modelo de Datos mediante este se obtiene de manera eficiente y rápido la información necesaria. Para concluir este capítulo se expuso el Modelo de Implementación, donde se especificaron los distintos componentes creados para desarrollar la aplicación y la relación entre ellos, lográndose traducir el diseño en términos de componentes ejecutables. De esta forma se cuenta con la base principal en la cual apoyar la construcción del sistema.

CONCLUSIONES

Durante el desarrollo de esta investigación se logró cumplir con el objetivo general propuesto: “Desarrollar de una herramienta informática que permita describir y relacionar de manera ordenada los archivos contenedores de cartografías digitales. Conjuntamente se cuenta con una documentación técnica que describe claramente todos los artefactos generados, convirtiéndose en una guía para llevar a cabo una correcta implementación del software y lograr un mejor entendimiento de los procesos a automatizar. Para llevar a cabo la construcción de la aplicación se realizó un estudio minucioso de las tendencias y tecnologías actuales a nivel mundial, con el objetivo de seleccionar las herramientas adecuadas, para garantizar un software con la más alta calidad, regidas por los criterios de selección de tecnologías libres y multiplataforma, aspecto fundamental que impulsa la universidad y el país en sentido general, de esta forma se garantiza un producto con mayor tiempo de vida, más reutilizable y eficiente.

Los resultados alcanzados a lo largo del proceso investigativo, permiten concluir que la herramienta diseñada, constituye un aporte importante, al considerarse el único software informático sobre plataforma de escritorio, que permita categorizar la gran variedad de archivos raster y vectoriales. Se logró un diseño y una arquitectura de la información de manera sencilla, fácil de usar, intuitiva que proporciona un entorno agradable para los usuarios finales. Todos los requisitos funcionales y no funcionales capturados en el momento correspondiente fueron debidamente implementados. Con la construcción de dicho sistema, permitió agilizar de manera automatizada mediante un catálogo el proceso de búsqueda y clasificación de los archivos raster y vectoriales, garantizando el desarrollo en tiempo y forma las actividades que se llevan a cabo en el proyecto productivo SIG- Desktop.

RECOMENDACIONES

Luego de concluir la propuesta de solución de esta investigación y a ver cumplidos los objetivos trazados, se proponen las siguientes recomendaciones:

- Implementar las pruebas diseñadas y documentar los defectos en función de lograr un producto más robusto y libre de errores.
- Continuar el ciclo de desarrollo de la herramienta, con la identificación de nuevos requisitos funcionales, en aras de ampliar las posibilidades de trabajo del sistema desarrollado, que permita obtener una herramienta competitiva en el mercado del software.
- Presentar los resultados de la investigación en eventos de carácter científico.

REFERENCIAS BIBLIOGRÁFICAS.

- **(Aronoff1987):** Aronoff, S. Geographical Information Systems: A management perspective. WDL Pub., 1987.
- Arquitectura basada en 3capas :Consultado 2010; Disponible en:<http://geeks.ms/blogs/jkpelaiez/archive/2009/05/29/arquitectura-basada-en-capas.aspx>
- Cartografía y Mapas Digitales: Consultado 2010; Disponible en:<http://www.mapas.com.co/VisorMPC2008/contenido/cartografiamd.jsp>
- **(Cornejo 2001):** Arquitectura en Capas. Un camino hacia los procesos Distribuidos.
- **(Cowen1989):** Cowen D. Lectura NCGIA. , 1989.
- **(Chacón):** CHACÓN, JULIO CÉSAR RUEDA. 2006. APLICACIÓN DE LA METODOLOGÍA RUP PARA EL DESARROLLO RÁPIDO DE APLICACIONES BASADO EN EL. Guatemala: Universidad de San Carlos de Guatemala. Facultad de Ingeniería. Escuela de Ingeniería en Ciencias, 2006.
- **(Chrisman, 1997):** Chrisman, N.R. (1997) Exploring Geographic Systems. John Wiley and Sons.
- **(Chuvieco):** Chuvieco, E., 2005. Teledetección y medio ambiente, Ariel, Geografía, Barcelona.
- Consulta ,Edición y Análisis Espacial con ArGis 9.2.Tomol: Teoría (Consultado 2010; Disponible en: <http://www.scribd.com/doc/38966709/7/ARCCATALOG>
- **(Delgado2000):** Delgado, T. Infraestructura Cubana de Datos Geoespaciales: Una necesidad nacional para la integración y diseminación de datos geoespaciales. Memorias del II Congreso Internacional Geomántica 2000, 2000.
- DIFERENCIAS ENTRE METODOLOGIAS XP Y RUP(Consultado 2010; Disponible en<http://oness.sourceforge.net/proyecto/html/ch05.html>
- Durán, Diana (2004) Educación geográfica. Cambios y continuidades.

- Buenos Aires. Lugar Editorial. (Consultado 2010; Disponible en:<http://www.scribd.com/doc/3069256/Cartografia-trabajo-entregado-225>)
- Distinctive Features of SQLite. SQLite (3 de Marzo, 2008). Consultado el 5 de Abril, 2010.
 - Entorno de desarrollo Qt Creator: (Consultado 2010; Disponible en http://developer.qt.nokia.com/wiki/Category:Tools::QtCreator_Spanish
<http://pixelcoblog.com/qt-creator-completo-entorno-de-desarrollo-multiplataforma>)
 - GeoNetwork opensource. (Consultado 2010; Disponible en <http://geonetwork-opensource.org>)
 - Geográfica (S.I.G.). 2002, Disponible en:http://www.inta.gov.ar/barrow/info/documentos/SIG/que_es_sig.htm
 - **(GSI)**: GSI. 2007. Herramientas y soluciones IBM. www.rational.com.ar. [En línea] 2007. [Citado el: 5 de marzo de 2010.] <http://www.rational.com.ar/herramientas/rup.html>
 - Introducción a los SIG. (Consultado 2010; Disponible en:http://www.geogra.uah.es/gisweb/1modulosespanyol/IntroduccionSIG/GISModule/GIST_Vector.htm)
 - Introducción a Rational Rose. [En línea] [Citado el: 12 de Marzo de 2010.] <http://sistemaronald.blogspot.com/2008/08/introduccion-rational-rose.html>.
 - **(Jacobson, y otros, 2000)**: Jacobson, I., G. Booch, and J. Rumbaugh. El Proceso Unificado de Desarrollo de Software. 2000.
 - LAJARA, M. M. y SALINAS, J. G. S. Panorama actual del ecosistema de software libre para SIG. 2007, Disponible en:<http://www.sigte.udg.es/jornadassiglibre2007/comun/1pdf/12.pdf>.
 - La Arquitectura tres Capas: Consultado 2010; Disponible en: <http://davidguru.wordpress.com/2010/02/08/la-arquitectura-de-tres-capas-introduccion/>
 - Lenguaje de Programación C++:(Consultado 2010; Disponible en: http://www.zator.com/Cpp/E1_2.htm)
 - (Linde): Linde, J. M. M. ¿QUÉ ES UN S.I.G.? LA EXPERIENCIA DEL I.M.I. DEL AYUNTAMIENTO DE PALMA Disponible en: http://www.mappinginteractivo.com/plantillaante.asp?id_articulo=133.

- LOCKHART, T. PostgreSQL Tutorial. 1998, Disponible en: <http://www.eskimo.com/support/PostgreSQL/tutorial/>.
- Metadatos Geográficos (Consultado 2010; Disponible en: <http://metadatosdeco1960.blogspot.com/>
- Metodologías de Desarrollo de Software. [En línea] [Citado el: 25 de Febrero de 2008.] http://static.scribd.com/docs/dsmodb48dz9x5.swf?INITIAL_VIEW=EW=width.
- Metodologías tradicionales vs. Metodologías ágiles. . [En línea] [Citado el: 4 de Marzo de 2010.] http://www.mygnet.net/manuales/software/metodologias_tradicionales_vs_dot_metodologias_agile_s.
- Microsoft Solution Framework (MSF) ; Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software
- Naturaleza de los Datos espaciales. (Consultado 2009; Disponible en: <http://www.udistrital.edu.co>
- **(Nogueras):** NOGUERAS et al. (2005) "OGC Catalog Services: Akey element for the development of Spatial Data Infrastructures".
The Geospatial Data Abstraction Library (Consultado 2010; Disponible en: <http://www.springerlink.com/content/n55l25g2h0r586m2/>.
- OMG. 2009. Unified Modeling Language. [En línea] 2009. [Citado el: 3 de marzo de 2010.] www.uml.org.
- Paradigm, V. Visual Paradigm for UML. [Disponible en: <http://www.visual-paradigm.com/product/vpuml/>.
 - (Popkin Software and Systems): Popkin Software and Systems. 2008. Modelado de Sistemas con UML. 2008.
- RAMSEY, P. Manual PostGIS. Disponible en: <http://postgis.refractory.net/documentation/postgis-spanish.pdf>.

- Tomlin, C.D. (1991) Cartographic modeling en D.J. Maguire; M.F. Goodchild and D.W. Rhind (Eds.); 1991 Geographical Information Systems: Principles and Applications, John Wiley & sons, Chichester www.wiley.co.uk/wileychi/gis/resources.html
- Visual Paradigm Internacional. [En línea] [Citado el: 10 de Febrero de 2010.][http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29)
- **(Yagüez y Langhi)** YAGÜEZ, J. C. D. y LANGHI, R. Sistema de Información.

BIBLIOGRAFÍA.

- De Meers, M.N. (2002) GIS Modeling in Raster John Wiley & sons, Chichester 203 pp. Tomlin, C.D. (1990) Geographic Information Systems and Cartographic Modeling Prentice Hall, 249 pp.
- Laurini, R. and Thompson, D. (1992) Fundamentals of Spatial Information Systems. London, Academy Press.
- Larman, C., ed. *UML Y PATRONES. Introducción al análisis y diseño orientado a objetos.*
- Peuquet, D.J. and Marble, D.F. (eds.) (1990) Introductory. Readings in Geographic Information Systems. London, Taylor and Francis.
- Rolando Alfredo Hernández León, Sayda Coello González. El Paradigma Cuantitativo de la Investigación Científica. Ciudad de la Habana: EDUNIV Editorial Universitaria, 2002. ISBN959-16-0343-6.
- Valle, Yunier Martínez, LiberMaps: Sistema de Catálogo para la Gestión de Mapas en la Web.

GLOSARIO DE TÉRMINOS

CU: caso de uso.

ESRI Shapefiles: es un formato de archivo informático propietario de datos espaciales desarrollado por la compañía ESRI, quien crea y comercializa software para Sistemas de Información Geográfica.

GDAL: es una colección de traductor para los formatos de datos raster geoespaciales que se distribuye bajo de código abierto de licencia por la Open Source Geospatial Foundation.

Geo TIFF: es un formato de metadatos, que proporciona información geográfica a asociarse con los datos de imagen.

GeneSig: es una plataforma que ha sido desarrollada en la Universidad de las Ciencias Informáticas para la futura creación de aplicaciones SIG en entornos web.

GEYSED: Centro de Desarrollo de Geoinformática y Señales Digitales.

GIF: es un formato de archivos de gráficos de mapa de bits (una trama).

GUI: Diseñador de Interfaces Gráficas de Usuario.

IDE: Infraestructura de Datos Espaciales.

ISO: Organización Internacional para la Estandarización o ISO, cuyo nombre en inglés es International Organization for Standardization.

JPG: es un formato de compresión de imágenes, tanto en color como en escala de grises, con alta calidad (a todo color).

Multiplataforma: sistema informático que corre sobre varios sistemas operativos, sin prescindir de ninguna de sus funcionalidades.

OGR: simples operaciones de biblioteca es un C + + de código abierto de la biblioteca, de línea de comandos herramientas la prestación de leer.

Open Source: Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente.

PDF: es un formato de documento portátil.

PNG: es un formato de archivos de gráficos de mapa de bits (una trama).

PostGIS: Es una extensión al sistema de base de datos objeto-relacional PostgreSQL.

PLUG-INS: es un módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande.

RAM: memoria de acceso aleatorio, es la memoria desde donde el procesador recibe las instrucciones y guarda los resultados.

RF: Requisito Funcional.

RUP: Proceso Unificado de Rational, o en sus siglas en inglés: Rational Unified Process.

SIG: Sistema de Información Geográfica.

UML: Lenguaje Unificado de Modelado.

.

ANEXOS

ANEXO I

Descripción de los casos de usos del sistema.

Tabla 3: Descripción textual de CU Autenticar Usuario.

Caso de Uso:	Autenticar Usuario.
Actores:	Usuario.
Resumen:	El CU se inicia cuando el usuario introduce en el sistema su usuario y contraseña para acceder a la aplicación, el sistema verifica la validez de estos datos para determinar si tiene privilegios de acceso.
Referencias	RF 1.
Prioridad	Crítico.
<i>Flujo Normal de Eventos</i>	
Acción del Actor	Respuesta del Sistema
1. El usuario accede al sistema.	2. El sistema solicita su usuario y contraseña.
3. El usuario introduce los datos solicitados.	4. El sistema verifica que los datos sean correctos y permite el acceso a la aplicación.



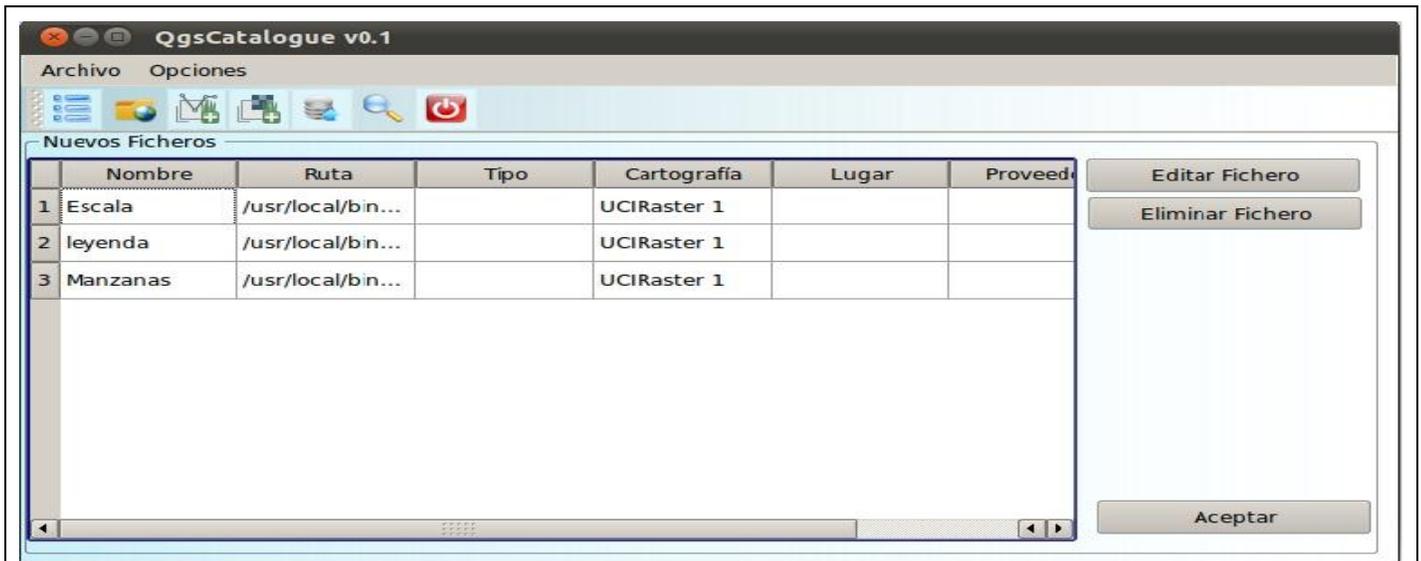
Flujo Alternativo

Acción del Actor	Respuesta del Sistema
	<p>5. El sistema verifica que los datos son incorrectos y da la posibilidad de que el usuario introduzca los datos correctos.</p> <p>Pasa directamente a la acción 2.</p>

Tabla 4 : Descripción textual de CU Actualizar Ruta.

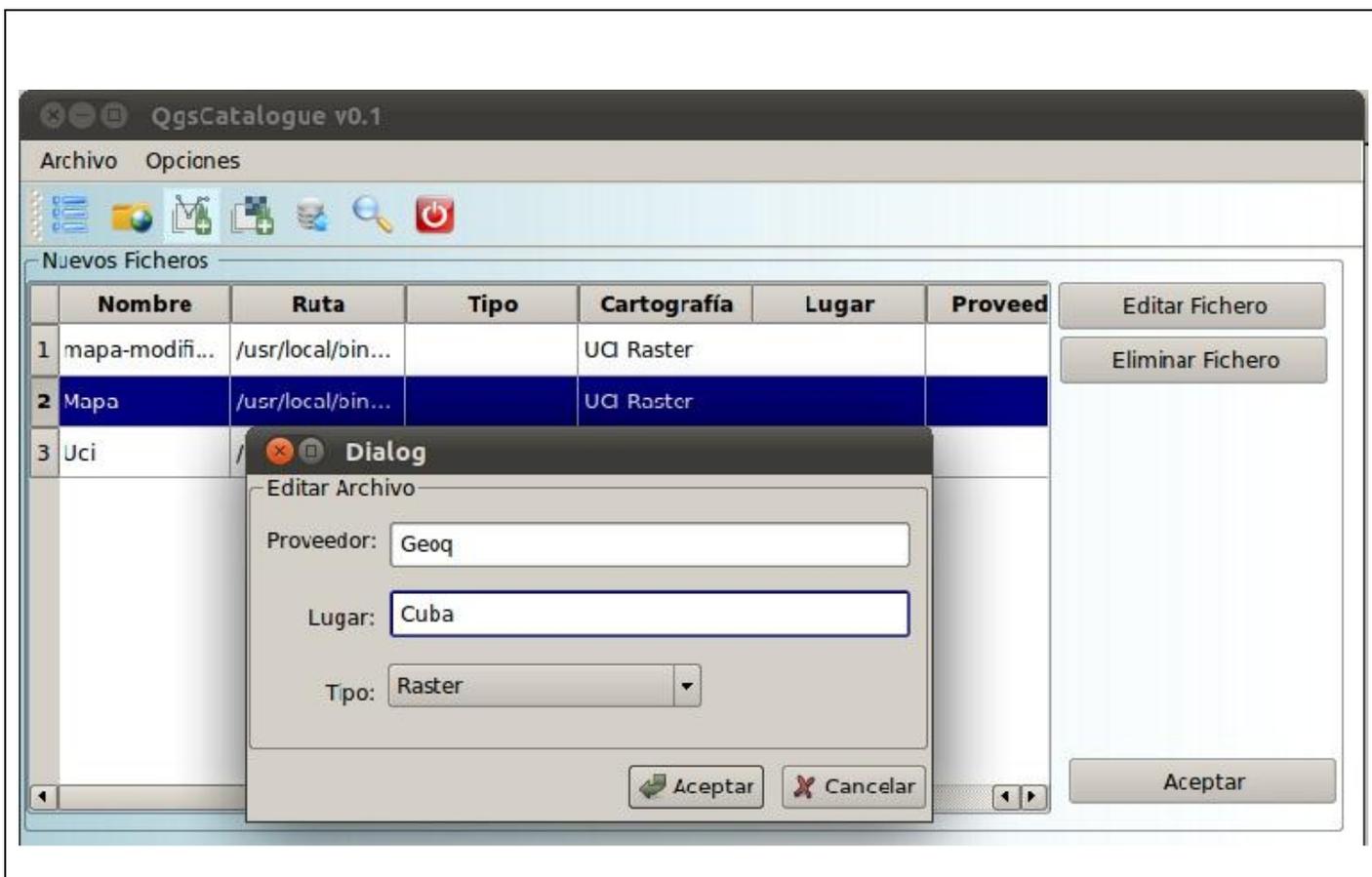
Caso de Uso:	Actualizar Ruta.
Actores:	Usuario
Resumen:	El CU se inicia cuando el usuario desea actualizar la base de datos con los nuevos archivos que se van incorporar, que estaban copiados en la carpeta predefinida para las cartografías digitales.
Referencias	RF 2.

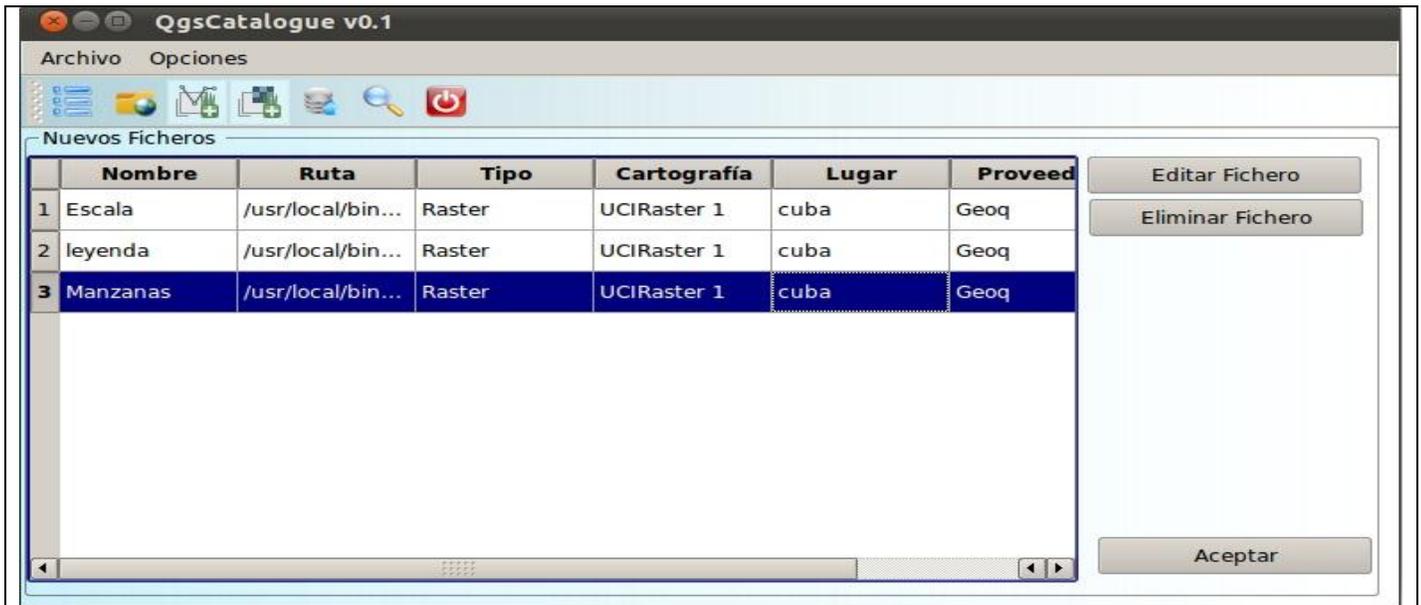
Prioridad	Crítico.
<i>Flujo Normal de Eventos</i>	
Acción del Actor	Respuesta del Sistema
1. El usuario solicita la opción Actualizar ruta.	2. El sistema muestra un formulario con los nuevos archivos que desean almacenar en la base de datos. 3. El sistema da la opción de Editar ficheros o Eliminar ficheros.
4. Escoge opción Editar haciendo clic sobre el botón "Editar Ficheros". Sección 1. 5. Escoge opción Eliminar ficheros seleccionando el archivo que desea eliminar y haciendo clic sobre el botón "Eliminar ficheros". Sección 2	



Sección1 Editar Ficheros.

Acción del Actor	Respuesta del Sistema
	6. El sistema ofrece una interfaz con los nuevos archivos que se desean añadir a la base datos.
6. El usuario selecciona los archivos a editar. 7. El usuario presiona el botón Editar Ficheros.	8. El sistema muestra un formulario que los nuevos campos a editar: Proveedor, Lugar y Tipo.
9.El usuario presiona el botón Aceptar	8. El sistema muestra los nuevos ficheros con los campos editados.
10. El usuario presiona el botón Aceptar	11. El sistema muestra una interfaz con los ficheros existente en la base de datos.





Flujo Alternativo de Eventos

10. El sistema verifica que se haya seleccionado al menos un archivo para realizar la acción de editar de lo contrario el sistema muestra un mensaje "Debe seleccionar un fichero".
Pasa directamente a la acción 6.

Sección 2 Eliminar Ficheros.

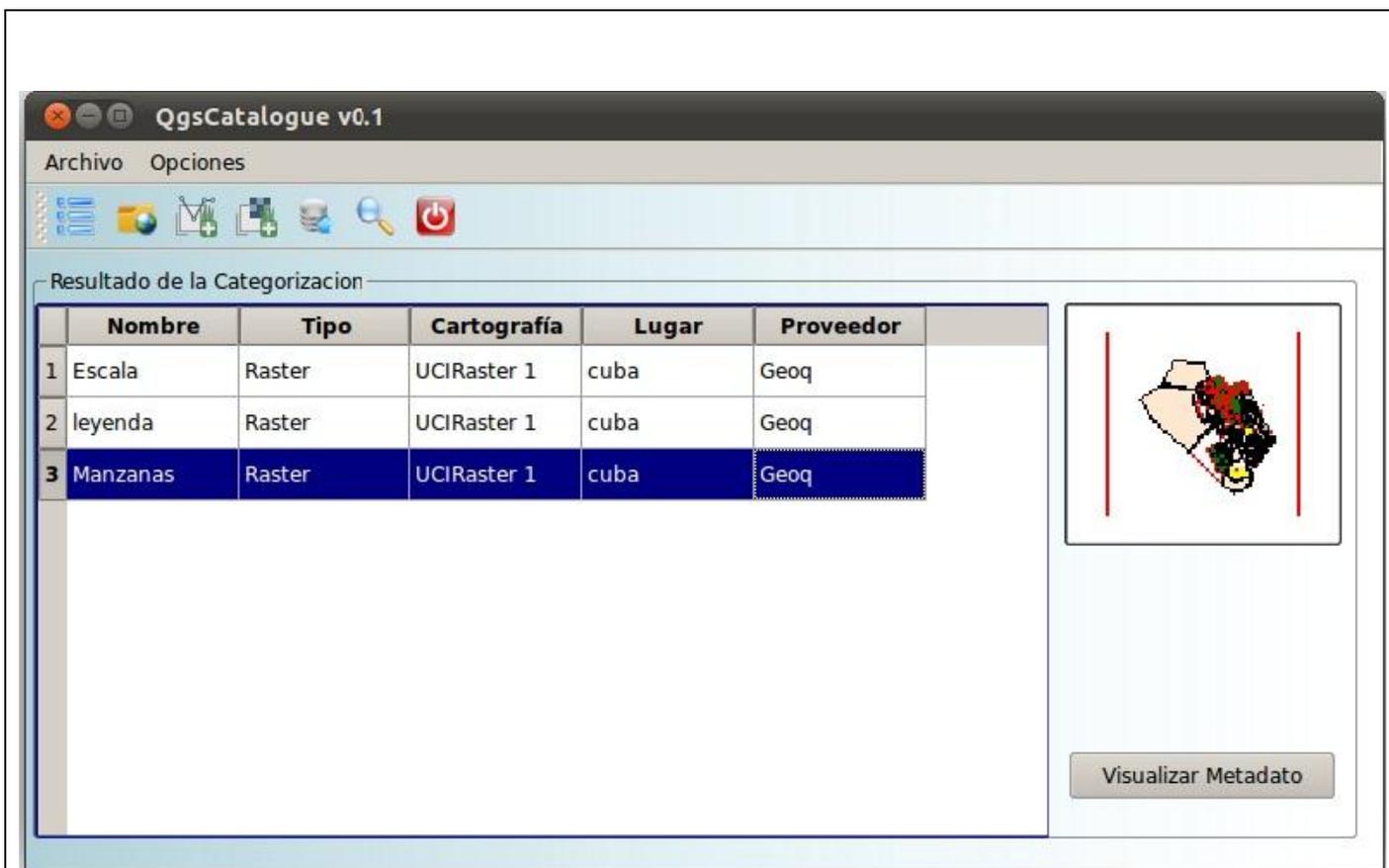
Acción del Actor	Respuesta del Sistema
	11. El sistema ofrece una interfaz con los nuevos archivos que se desean añadir a la base datos.

12. El usuario selecciona los archivos a eliminar. 13. El usuario presiona el botón Eliminar Ficheros.	13. El sistema va eliminando los ficheros existentes que no se desean almacenar en la base dato.
14. El usuario presiona el botón Guardar.	15. El sistema guarda los cambios realizados.
Flujo Alternativo de Eventos	
	17. El sistema verifica que se haya seleccionado al menos un archivo para realizar la acción de eliminar de lo contrario el sistema muestra un mensaje "Debe seleccionar un fichero". Pasa directamente a la acción 11.

Tabla 5 : Descripción textual de CU Vista Previa.

Caso de Uso:	Vista Previa.
Actores:	Usuario.
Resumen:	El CU se inicia cuando el usuario desea obtener una Vista Previa de la

	información de los archivos de los archivos raster como vectoriales.
Referencias	RF 3.
Precondiciones	Se tiene registrado la categorización de los archivos.
Prioridad	Secundarios.
<i>Flujo Normal de Eventos</i>	
Acción del Actor	Respuesta del Sistema
1. El usuario solicita la opción Vista previa.	2. El sistema muestra una interfaz con los resultados de la categorización de los archivos.
3. El usuario selecciona el archivo para mostrar su vista previa.	4. El sistema muestra la vista previa del archivo seleccionado por el usuario.



Flujo Alterno

Acción del Actor

Respuesta del Sistema

8.El sistema verifica que se haya seleccionado al menos un archivo para mostrar la vista previa de lo contrario el sistema muestra un mensaje "Debe seleccionar un archivo".

Pasa directamente a la acción 2.

Tabla 6 : Descripción textual de CU Visualizar Metadatos.

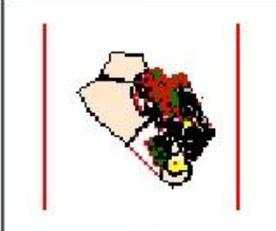
Caso de Uso:	Visualizar metadatos.	
Actores:	Usuario.	
Resumen:	El CU se inicia cuando el usuario desea obtener una visualización de los metadatos de los diferentes archivos contenedores de cartografía digital tanto raster como vectoriales.	
Referencias	RF 4.	
Precondiciones	Se tiene registrado la categorización de los archivos.	
Prioridad	Secundarios.	
<i>Flujo Normal de Eventos</i>		
Acción del Actor	Respuesta del Sistema	
1. El usuario solicita la opción Visualizar Metadatos.	2. El sistema muestra una interfaz con los resultados de la categorización de los archivos.	
3. El usuario selecciona el archivo para mostrar sus metadatos. 4. El usuario presiona el botón Visualizar Metadatos.	5. El sistema ofrece una interfaz con las propiedades del archivo seleccionado: Simbología, Mapa Color, General, Metadatos e Histogramas.	
6. El usuario selecciona el botón Aceptar.	7. El sistema cierra la interfaz con las propiedades del archivo.	

QgsCatalogue v0.1

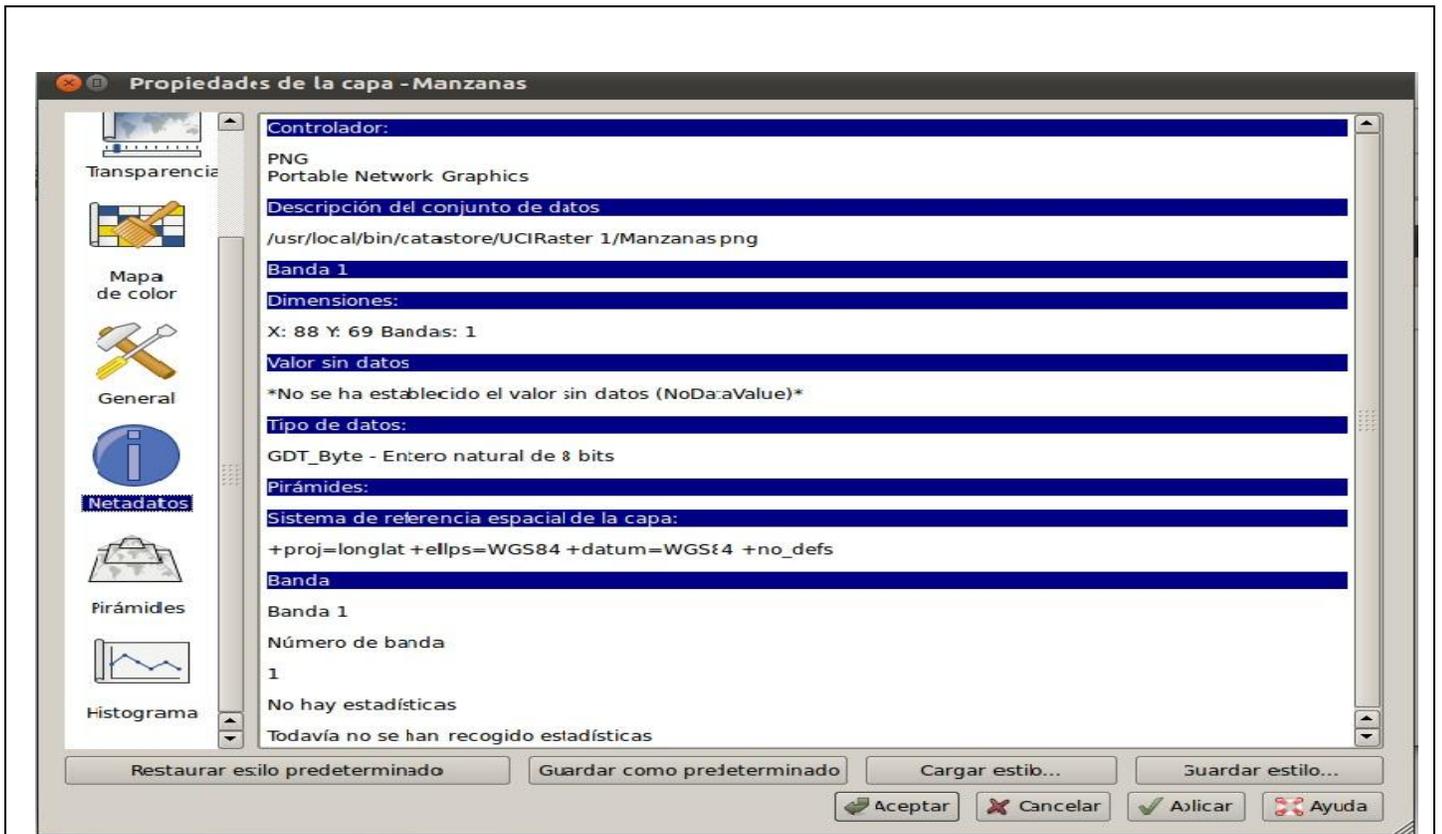
Archivo Opciones

Resultado de la Categorización

	Nombre	Tipo	Cartografía	Lugar	Proveedor
1	Escala	Raster	UCIRaster 1	cuba	Geoq
2	leyenda	Raster	UCIRaster 1	cuba	Geoq
3	Manzanas	Raster	UCIRaster 1	cuba	Geoq



Visualizar Metadato



Flujo Alternativo de Eventos

9. El sistema verifica que se haya seleccionado al menos un archivo para mostrar la vista previa de lo contrario el sistema muestra un mensaje

“Debe seleccionar un archivo”.

Pasa directamente a la acción 2.

ANEXO II Diagramas Clases del Diseño.

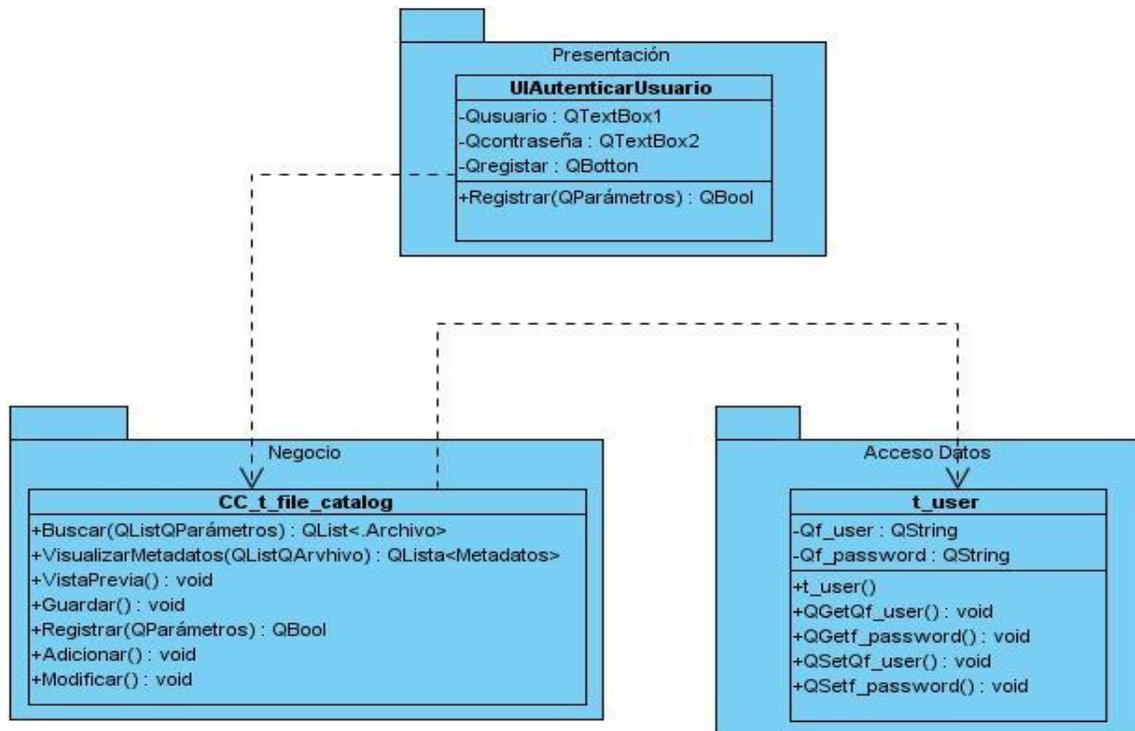


Figura 17 .Diagrama del Diseño CU Autenticar Usuario.

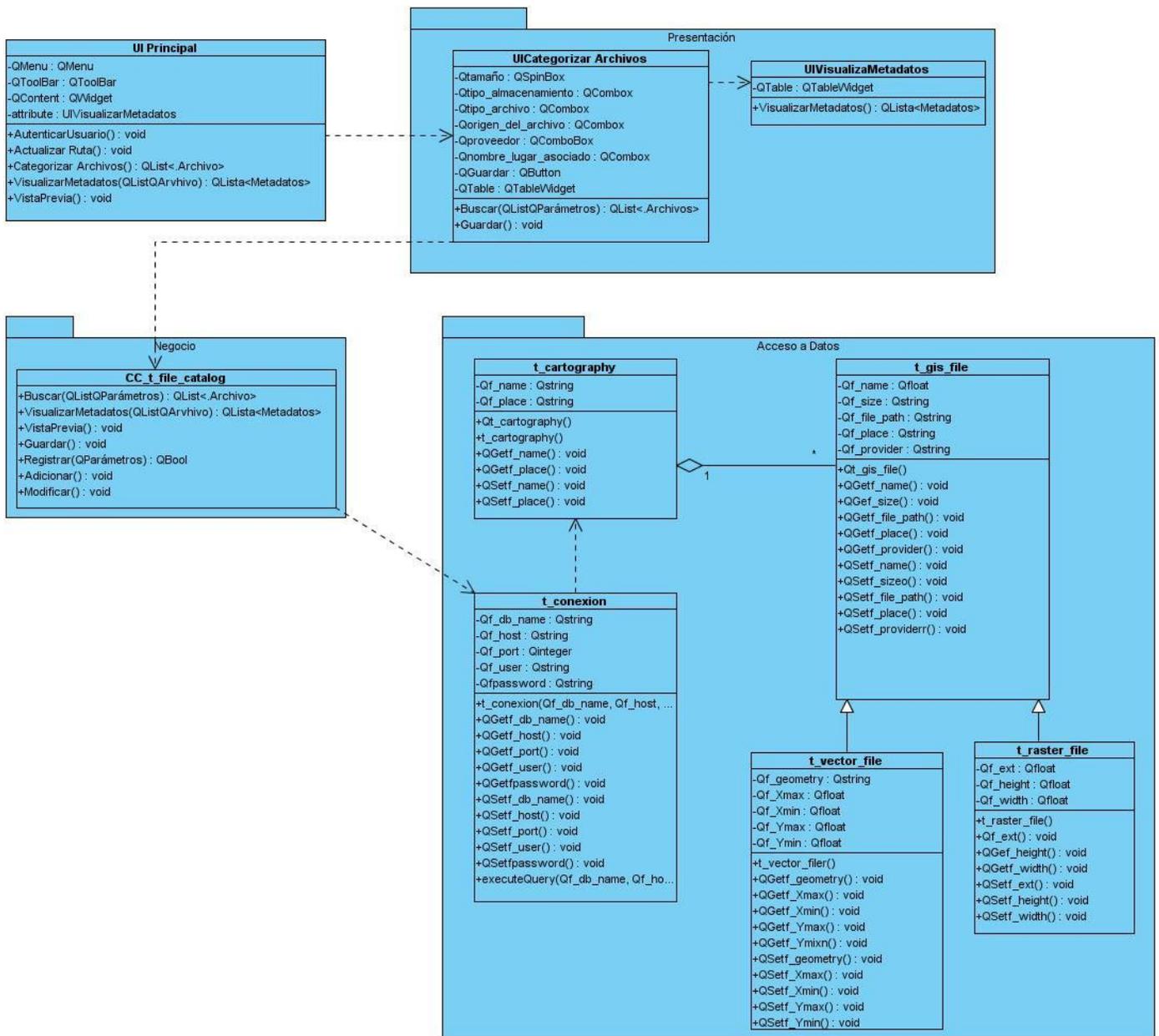


Figura 18. Diagrama del Diseño CU Visualizar Metadatos.

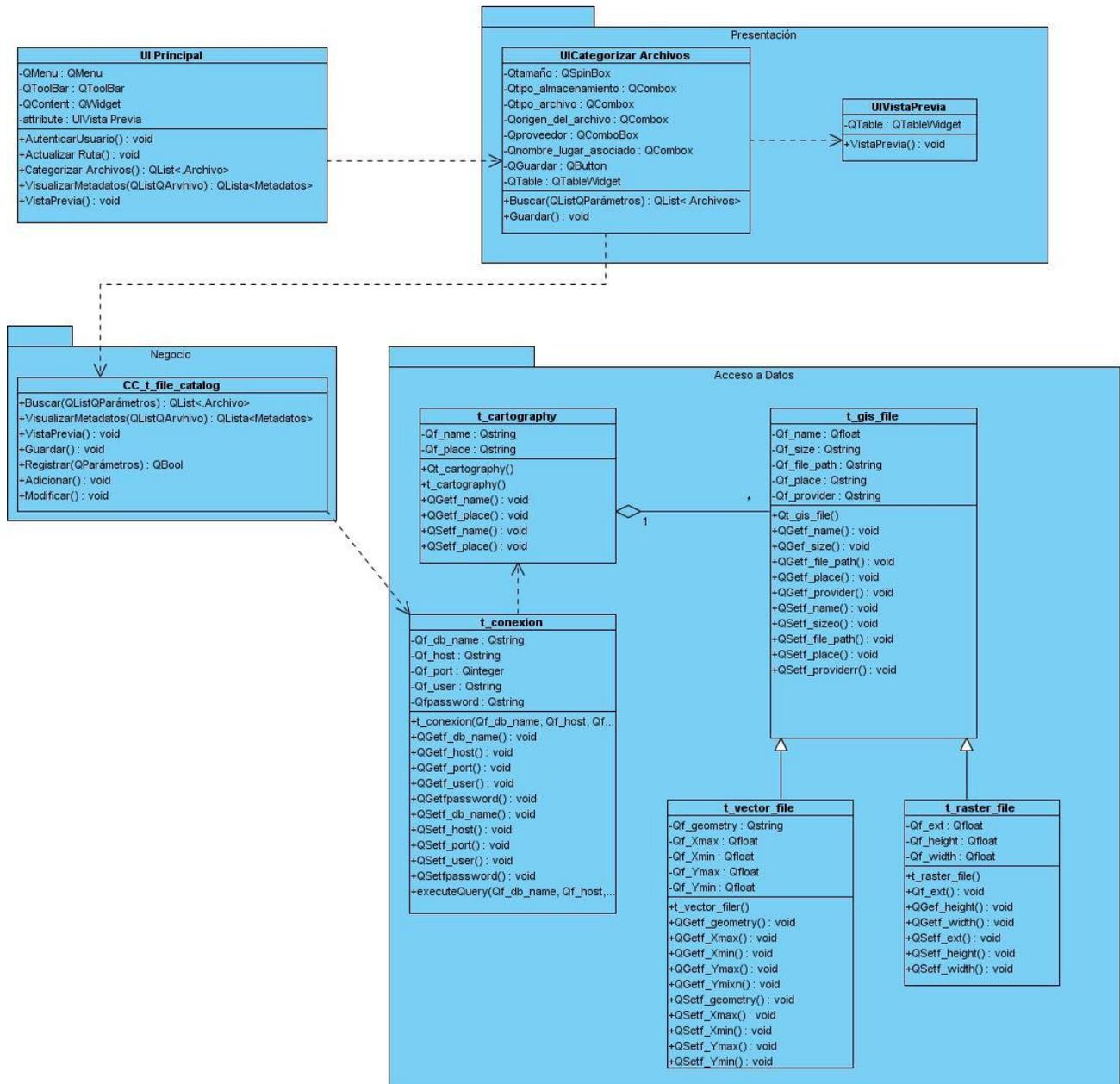


Figura 19. Diagrama del Diseño CU Vista Previa.

ANEXO III Diagramas de Componentes.

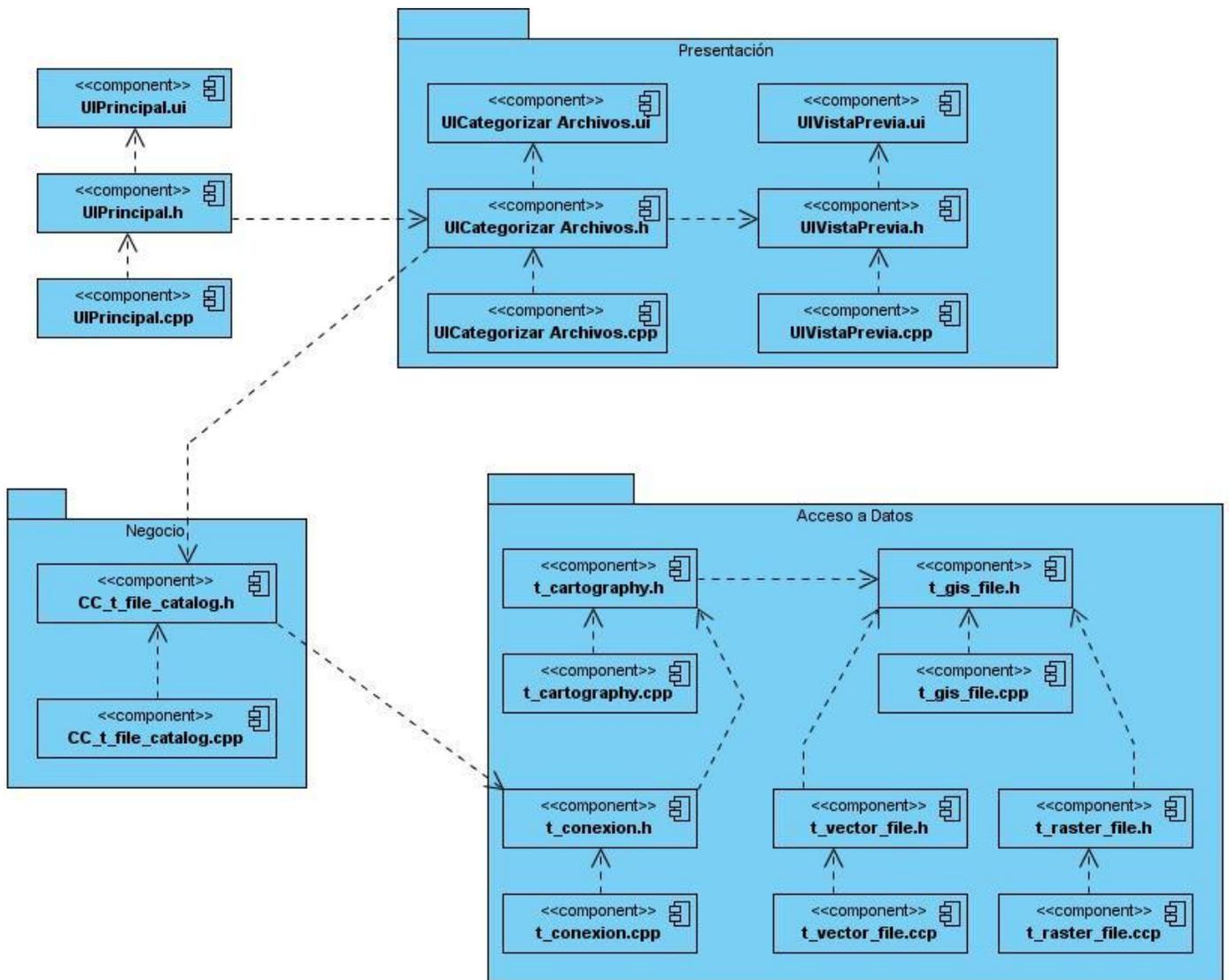


Figura 20 .Diagrama de Componente CU Vista Previa.

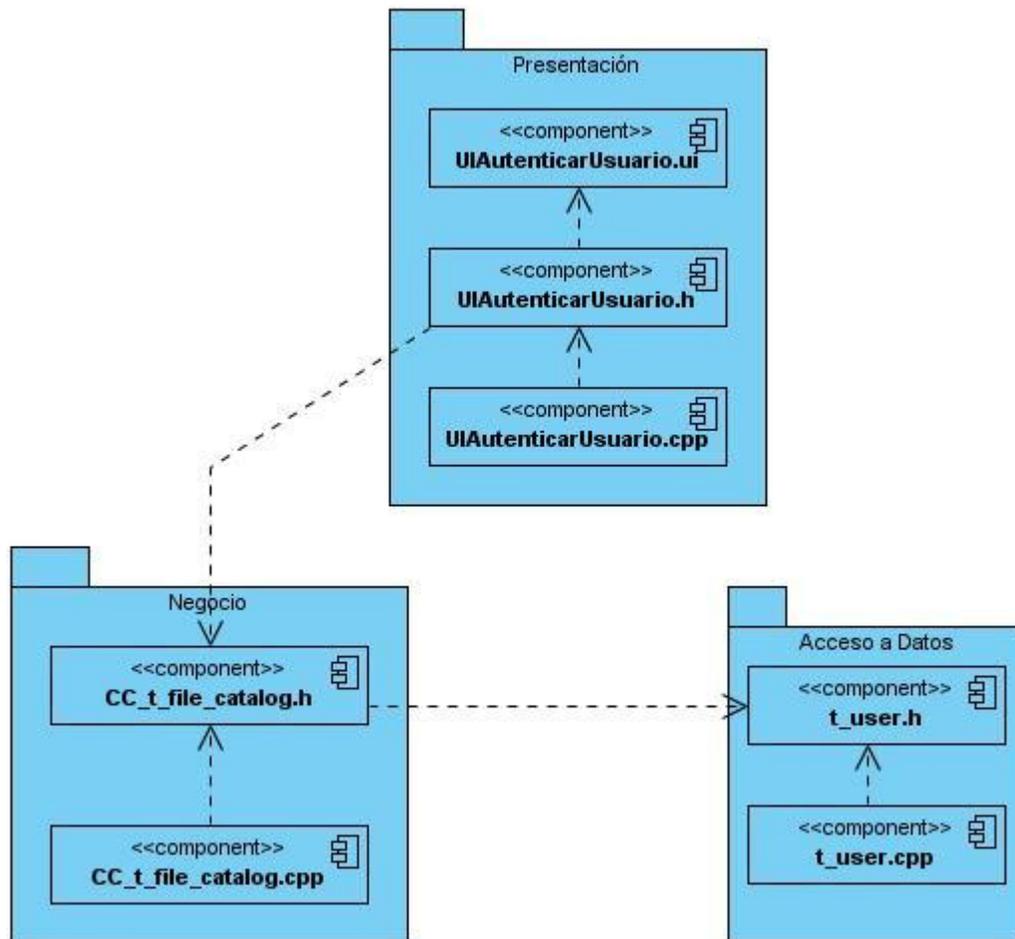


Figura 21 .Diagrama de Componente CU Autenticar Usuario.