

**DISEÑO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB BASADA EN
TECNOLOGÍAS DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADA
AL MANEJO DE INFORMACIÓN DE AGENCIAS DE BIENES RAÍCES**

**JEANETTE ROCIO CARREÑO VELASCO
MARGARITA LEAL JOYA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2008

**DISEÑO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB BASADA EN
TECNOLOGÍAS DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADA
AL MANEJO DE INFORMACIÓN DE AGENCIAS DE BIENES RAÍCES**

**JEANETTE ROCIO CARREÑO VELASCO
MARGARITA LEAL JOYA**

**PROYECTO DE GRADO PARA OPTAR POR EL TÍTULO DE
INGENIERO DE SISTEMAS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2008

UNIVERSIDAD INDUSTRIAL DE SANTANDER

PROYECTO DE GRADO

TÍTULO:

**DISEÑO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB BASADA EN
TECNOLOGÍAS DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADA
AL MANEJO DE INFORMACIÓN DE AGENCIAS DE BIENES RAÍCES**

PRESENTADO ANTE: Comité de proyectos de grado

ESCUELA : Ingeniería de Sistemas e Informática

FACULTAD : Ciencias Físico - Mecánicas

ELABORADO POR: Jeanette Rocío Carreño Velasco
Margarita Leal Joya

DIRECTOR: Ing. Enrique Sarmiento Moreno
Docente Escuela de Ingeniería de Sistemas
Grupo de Investigación GEMA

A Dios

A mis padres y a mi familia

A Margarita por todo el empeño que le puso y la paciencia que me tuvo

Al Profe Enrique por confiar en nuestro trabajo

A Xiomara por su constante colaboración

A Guille y a Diego por sus conocimientos en SIG

A todos los amigos y personas que estuvieron pendientes y preguntando “¿Para Cuando?”

JEANETTE ROCIO CARREÑO VELASCO

A Dios.

A mis Padres y hermanos.

A Juana por su paciencia.

Al profesor Enrique Sarmiento por la confianza
y el apoyo brindado

A Guillermo y Diego por su valiosa
colaboración.

A todos aquellos que confiaron en mi trabajo,
en especial a: una tal X, las LIC y en general a
todo el G7...

MARGARITA LEAL JOYA

AGRADECIMIENTOS

Ahora que finaliza una etapa nuestras vidas, queremos agradecer a todas aquellas personas que aportaron para que culmináramos con éxito esta meta planteada al comienzo del proyecto:

Queremos antes que nada agradecer a Dios, por estar siempre presente en nuestras vidas y ser nuestra luz, nuestro camino y nuestra fortaleza.

A la Universidad Industrial de Santander por la formación profesional y personal obtenida en el transcurso de nuestra vida universitaria

Al Profesor Enrique por la confianza depositada, por todas sus enseñanzas y por su apoyo incondicional.

A todos los compañeros del grupo de investigación GEMA, que nos brindaron su apoyo y fueron partícipes de este aprendizaje, principalmente a los ingenieros Guillermo Hernández y Diego Ordoñez por su valiosa colaboración en SIG.

A Xiomara porque siempre estuvo presente en todas las etapas del proyecto ayudándonos con los dibujos y las diferentes VX que se realizaron.

A nuestros amigos más cercanos, por acompañarnos durante este laaaargo, laaaaaaargo proceso, brindándonos su comprensión y apoyo.

Y también gracias a todos los que nos prestaron los portátiles en el momento preciso; a GOOGLE porque casi todo lo supo; a todos los que nos interrumpieron por Messenger cuando estábamos trabajando en el proyecto... y porque no, a nosotras también, por no desistir en los momentos más críticos y dedicarle los eternos fines de semana, donde sólo a veces, nos acompañó una "Perfecta".

CONTENIDO

INTRODUCCIÓN	16
1. ESTUDIO PRELIMINAR, DESCRIPCION DEL PROYECTO	17
1.1 PLANTEAMIENTO DEL PROBLEMA	17
1.2 OBJETIVOS DEL PROYECTO	19
1.2.1 Objetivo General	19
1.2.2 Objetivos Especificos	19
1.3 JUSTIFICACION	22
1.4 VIABILIDAD	23
2. MARCO TEORICO	24
2.1 SISTEMAS DE INFORMACION GEOGRAFICA	24
2.1.1 Definición	25
2.1.2 Componentes de un SIG	25
2.1.3 Modelos y Estructuras de Datos	27
2.1.4 SIG en Internet	32
2.2 BASE DE DATOS ESPACIALES	34
2.2.1 Tipos de Datos	34
2.2.2 Información Espacial	35
2.2.3 Información Alfanumérica	37
2.2.4 Sistema Manejador de Base de Datos (SMBD)	37
3. METODOLOGIA	40
3.1 PROTOTIPADO INCREMENTAL	40
3.2 LENGUAJE UNIFICADO DE MODELADO (UML)	40
3.2.1 Generalidades de UML	41
3.2.2 Elementos	44
3.2.3 Relaciones	47
3.2.4 Diagramas de UML	48
4. TECNOLOGÍAS SOFTWARE EMPLEADAS	51
4.1 MAPSERVER	51
4.2 PHP MAPSCRIPT	55
4.3 MAPLAB	56
4.4 LENGUAJES WEB	56
4.4.1 HTML	56
4.4.2 PHP	58
4.4.3 JAVASCRIPT	59
4.5 ALMACENAMIENTO DE DATOS	60
4.5.1 POSTGRESQL	60
4.5.2 POSTGIS	62

5.	ANÁLISIS DE REQUERIMIENTOS	65
5.1	USUARIO PÚBLICO	67
5.2	USUARIO CLIENTE	68
5.3	USUARIO ASESOR COMERCIAL	68
5.4	USUARIO ADMINISTRADOR DEL SISTEMA	68
6.	DISEÑO DE LA HERRAMIENTA	70
6.1	ACTORES	70
6.2	CASOS DE USO	71
7.	IMPLEMENTACIÓN	75
7.1	MODELO DE DATOS DEL PROTOTIPO	76
7.2	INTERFAZ GRÁFICA	76
7.2.1	Usuario Público	78
7.2.2	Usuario Administrador	80
7.2.3	Usuario Asesor Comercial	89
7.2.4	Usuario Cliente	91
8.	PRUEBAS	92
9.	CONCLUSIONES Y RECOMENDACIONES	103
9.1	CONCLUSIONES	103
9.2	RECOMENDACIONES	106
10.	BIBLIOGRAFIA	108
10.1	SISTEMAS DE INFORMACION GEOGRAFICA	108
10.2	BASES DE DATOS	108
10.3	DISEÑO	110
10.4	LENGUAJE DE PROGRAMACIÓN	110

TABLA DE FIGURAS

Figura 1	Diagrama de componentes SIG	26
Figura 2.	Modelo de Datos	28
Figura 3.	Esquema de los Modelos Lógicos	29
Figura 4.	Modelo raster	31
Figura 5.	Arquitectura Internet MapServer (IMS)	33
Figura 6.	Modelo Prototipado Incremental	41
Figura 7	Esquema de funcionamiento MapServer	52
Figura 8	Actores.	71
Figura 9	Casos de Uso del Actor Administrador	72
Figura 10	Casos de Uso del Actor Asesor Comercial.	73
Figura 11	Casos de Uso del Actor Cliente.	74
Figura 12	Casos de Uso del Actor Usuario Público.	75
Figura 13	Interfaz Principal del Software	78
Figura 14	Interfaz de Contáctenos	79
Figura 15	Interfaz de Acerca de.	80
Figura 16	Interfaz de Inicio de Sesión.	80
Figura 17	Menú principal del Administrador	81
Figura 18	Manejador de Inmuebles	81
Figura 19	Visualizar datos de Inmuebles	82
Figura 20	Consultas Espaciales	83
Figura 21	Interfaz de Eliminar Inmueble	83
Figura 22	Modificar Inmuebles	84
Figura 23	Crear Inmueble	85
Figura 24	Buscar Inmueble	85
Figura 25	Habilitar Inmuebles	86
Figura 26	Manejador de Usuarios	86
Figura 27	Visualizar Usuarios	87
Figura 28	Modificar Usuarios	87
Figura 29	Habilitar Usuarios	88
Figura 30	Crear Usuarios	88
Figura 31	Buscar Usuarios	88
Figura 32	Modificar Datos	89
Figura 33	Menú principal del Asesor Comercial	89
Figura 34	Visualizar Usuarios	90
Figura 35	Registrar Usuarios	90
Figura 36	Menú principal del Cliente	91
Figura 37	Lista de Inmuebles del Cliente	91
Figura 38	Imagen de prueba Web	92

LISTA DE TABLAS

Tabla 1	Funcionalidades de UML	43
Tabla 2	Elementos Estructurales de UML	46
Tabla 3.	Elementos de Comportamiento	46
Tabla 3	Elementos de Agrupación	46
Tabla 4	Elementos de Anotación	47
Tabla 5	Relaciones de UML	48
Tabla 6	Diagramas de UML	50
Tabla 7	Resultados del Módulo Usuario Administrador	97
Tabla 8	Resultados del Módulo Usuario Cliente.	99
Tabla 9	Resultados del Módulo Usuario Asesor Comercial.	100
Tabla 10	Resultados del Módulo Usuario Público.	102

LISTA DE ANEXOS

ANEXO A. OPENGIS	112
ANEXO B. MAPFILE	117
ANEXO C. SOFTWARE LIBRE	121
ANEXO D. FUNCIONES POSTGIS	125
ANEXO E. DICCIONARIO DE DATOS	135
ANEXO F. MANUAL DEL PROGRAMADOR	148
ANEXO G. NECESIDADES DEL SISTEMA	164
ANEXO H. MANUAL DE INSTALACION	167

RESUMEN

TITULO: DISEÑO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB BASADA EN TECNOLOGÍAS DE SISTEMAS DE INFORMACIÓN GEOGRÁFICA APLICADA AL MANEJO DE INFORMACIÓN DE AGENCIAS DE BIENES RAÍCES*

AUTORES: JEANETTE ROCIO CARREÑO VELASCO
MARGARITA LEAL JOYA**

PALABRAS CLAVES: Sistemas De Información Geográfica, Bases de Datos Espaciales, Web, Inmuebles, GPL, OpenSource, Metodología

CONTENIDO

Los Sistemas de Información Geográfica se han convertido en una herramienta casi indispensable para distintas áreas de la academia. Las herramientas Web para este tipo de aplicaciones, carecen de opciones que le faciliten al usuario ubicar espacialmente el producto que requiere y construir un bosquejo mental acerca de las implicaciones de la elección que está por realizar, ya que solo ofrecen información de tipo textual.

Se realiza una investigación de herramientas Web basadas en tecnologías de Sistemas de Información Geográfica y Bases de Datos Espaciales ligadas a ellas para efectos de la realización de análisis espaciales, y aplicarlas al desarrollo del proceso de oferta de inmuebles para el arriendo o venta que ofrece una inmobiliaria.

De igual forma la investigación se centrará en la exploración y uso de las herramientas SIG de licencia publica GPL, al igual que las bases de datos espaciales con opciones OpenSource, en este caso PostgreSQL con su extensión espacial PostGIS y la utilización de herramientas con el servidor de mapas Mapserver, la librería PHP/MapScript y el lenguaje de programación PHP y MapLab.

La metodología que se utiliza es la de Prototipado Incremental ya que se desarrolla primero las áreas de mayor riesgo del sistema, de tal forma que el prototipo pueda ser tomado como producto final una vez se llegue a su fin.

* Proyecto de Grado
** Facultad de Físico-Mecánicas
Escuela de Ingeniería de Sistemas e Informática

SUMMARY

TITLE:DESIGN AND IMPLEMENTATION OF A WEB TOOL SUPPORTED TECHNOLOGY GEOGRAPHIC INFORMATION SYSTEM INFORMATION MANAGEMENT APPLICATION THE REAL ESTATE AGENCY*

AUTHORS: JEANETTE ROCIO CARREÑO VELASCO **
MARGARITA LEAL JOYA**

KEY WORDS: Geographic Information System, Spatial Databases, Properties. GPL, OpenSource, Methodology.

CONTENT:

The Geographic Information Systems have become an almost indispensable tool for different areas of the academy. The tools for this type of Web applications, lack of options that will facilitate the user to spatially locate the product that requires mental and build a sketch about the implications of the choice that is accomplished, since only type offered textual information.

It conducts research tools based Web technologies of geographic information systems and spatial database linked to them for purposes of analysis space, and apply them in the development process of offering properties for sale or lease that provides a real estate.

Similarly research will focus on the exploration and use of GIS tools license GPL, as well as the spatial databases with options OpenSource, in this case PostgreSQL with its PostGIS spatial extent and use tool with the server maps Mapserver, the library PHP / MapScript and the programming language PHP and MapLab.

The methodology used is the Prototype Incremental as it develops the first areas of greatest risk of the system, so that the prototype could be taken as the final product once it reaches its end.

* Project of Degree
**Physical-Mechanics Faculty
Informatic and Systems Engineering School

INTRODUCCIÓN

En el presente trabajo se mostrará de forma general los temas tratados en la investigación propuesta a cerca de las herramientas Web basadas en tecnologías SIG y las Base de Datos Espaciales; de la misma forma se dará a conocer los pasos que se llevaron a cabo para realizar la aplicación de esta investigación en el prototipo software propuesto para el proceso de oferta de inmuebles para el arriendo o venta que ofrece una inmobiliaria, haciendo énfasis en incursionar principalmente en el manejo de la información geográfica que es con la que se puede demostrar la bondad de estas tecnologías.

A continuación se resume los temas tratados en cada capítulo del presente libro.

En el capítulo 1 se hace una introducción a la descripción general del proyecto, dando a conocer el planteamiento del problema, objetivo general y los específicos, la justificación y la viabilidad del proyecto. En el capítulo 2 se realiza el marco teórico que sirvió como base para el desarrollo del proyecto, los principales temas tratados acá son los Sistemas de Información Geográfica y el concepto de Base de Datos Espaciales. En el capítulo 3 se plantea y explica la metodología utilizada para el desarrollo del proyecto, que en este caso es el Prototipado Incremental. En el capítulo 4 se realiza una breve definición y se nombran sus principales características de las tecnologías software que se utilizaron para desarrollar el prototipo propuesto en el proyecto, entre las tecnologías utilizadas están: El servidor de mapas MapServer, el DBMS PostgreSQL y su extensión PostGis, El lenguaje de programación PHP y el módulo PHP/MapScript, HTML y JavaScript. En el capítulo 5 Se lleva a cabo el Análisis de requerimientos, identificando los actores del sistema, estableciendo sus funciones y restricciones. Limitando las funcionalidades que se van a tener finalmente para el desarrollo del prototipo propuesto. En el capítulo 6 se realiza el diseño del prototipo software, que va a ser el apoyo en la fase de codificación del prototipo. En el capítulo 7 se describe la implementación del prototipo diseñado, explicando el modelo de datos utilizado y la interfaz gráfica en cada uno de los módulos del prototipo. En el capítulo 8 se habla del resultado que arrojaron las pruebas hechas al prototipo. En el capítulo 9 se expresan las conclusiones a las que se llegó después del diseño, desarrollo e implementación del prototipo, teniendo en cuenta los objetivos propuestos inicialmente en el proyecto. En el capítulo 10 se realiza la referencia bibliográfica que se tuvo en cuenta durante el desarrollo del proyecto.

1. ESTUDIO PRELIMINAR, DESCRIPCION DEL PROYECTO

1.1 PLANTEAMIENTO DEL PROBLEMA

Los Sistemas de Información Geográfica se han convertido en una herramienta casi indispensable para distintas áreas de la academia. Una de las propiedades que les han dado fuerza es su gran versatilidad, ya que se puede utilizar información de diversas fuentes y en distintas formas; de igual forma es de destacar su gran capacidad para ejecutar análisis cada vez más complejos y manejar grandes volúmenes de datos.

Se destaca en particular la importancia que han tomado los SIG en Internet, este tipo de herramientas ofrecen gran cantidad de servicios que cada día son demandados por más personas, debido a que la tecnología informática y el acceso a Internet están cada vez más al alcance de la sociedad.

Pero a pesar de las evidentes ventajas que trae la utilización de este tipo de herramientas en los procesos de una empresa, su uso en nuestro medio es muy bajo o subutilizado, sobre todo en campos donde se hace evidente la necesidad del componente geográfico.

El Grupo GEMA tuvo dentro de sus electivas un curso de Introducción a SIG, y ha considerado que dentro de las herramientas que deben poder manejar en ese curso, se debe tener una que aplique experimentalmente esta tecnología a la solución de un problema frecuente en el área de los servicios, y que sirva de base a un proyecto posterior que suministre una herramienta comercialmente funcional para alguna empresa del área, dando origen a otros proyectos que proporcionarán la experiencia requerida para mostrar una Escuela fuerte en un área de su responsabilidad, y de la que ahora está marginada.

Un ejemplo donde se puede apreciar estas falencias es en la rama de las inmobiliarias, donde el proceso de oferta de inmuebles, ya sea para arriendo o venta, en la gran mayoría se sigue realizando aun de forma tradicional limitada por la utilización de procesos en los cuales se efectúan operaciones de información textual que no incluyen ningún tipo de manejo gráfico sobre consultas o actualización de la información que requieren, se convierte entonces, en una tarea tediosa donde se puede perder determinada información de tipo espacial que resultaría vital en el proceso, sin contar el gasto temporal y la no posibilidad de introducción de nuevos factores urbanísticos. Las herramientas Web para este tipo de aplicaciones, carecen de opciones que le faciliten al usuario ubicar espacialmente el producto que requiere y construir un bosquejo mental acerca de las implicaciones de la elección que está por realizar, ya que solo ofrecen información de tipo textual.

Se propone, en este proyecto realizar una investigación de herramientas Web basadas en tecnologías de Sistemas de Información Geográfica y Bases de Datos Espaciales ligadas a ellas para efectos de la realización de análisis espaciales, y aplicarlas al desarrollo del proceso de oferta de inmuebles para el arriendo o venta que ofrece una inmobiliaria.

Entre las aplicaciones que puede proporcionar el uso de los SIG en una inmobiliaria se encuentra la propiedad de geocodificación, donde se podrá no solamente ubicar un inmueble dada una dirección, sino poder realizar una exploración del entorno que rodea gráficamente al mismo, al tiempo que se podría acceder a la información alfanumérica asociada, todo integrado en un mismo programa. Otra aplicación interesante que ofrece el uso de los SIG a este tipo de industria es el análisis espacial, ya que proporcionan una visión general de la situación. Uno de los principales criterios de decisión de un usuario al elegir un inmueble está en función de su localización, es decir su decisión se puede

condicionar a la localización del inmueble en una determinada zona, caracterizada principalmente por sus propiedades ambientales (zonas verdes, parques, presencia de vías de comunicación, ausencia de ruidos, seguridad, entre otros) y de servicios (proximidad a centros comerciales y de ocio, hospital, centros educativos, iglesia, conectividad con las redes de comunicación, etc.).

Un sistema de información completo para una firma de finca raíz es complejo, requiere incluir también los elementos relacionados con la parte administrativa y financiera, pero teniendo en cuenta que éstos han sido manejados en otros proyectos por lo que son conocidos, no es el interés de este Proyecto manejarlos ahora. En esta investigación nos interesa incursionar en la información espacial que prácticamente no se ha venido incluyendo en estos sistemas de información, y para podernos concentrar en ese tipo de información, sólo esa información manejaremos en este proyecto. La idea es que si logramos darle un manejo adecuado al tema, en otro proyecto posterior de la Escuela se le puede incluir la información administrativa y financiera, para tener entonces un sistema integrado de información y se pueda ofrecer a las entidades del gremio, en tal forma que sea beneficioso económicamente, tanto para esas empresas, como para la Escuela.

De igual forma este proyecto de investigación se centrará en la exploración y uso de las herramientas SIG de licencia pública GPL, al igual que las bases de datos espaciales con opciones OpenSource, en este caso PostgreSQL con su extensión espacial PostGIS. La razón de esta preferencia es que el trabajar con opciones comerciales deja por fuera soluciones para empresas pequeñas y a la mayoría de proyectos de investigación por lo costoso que puede salir su licenciamiento. Sumado al bajo costo se encuentra la excelente funcionalidad con la que cuentan, en su mayoría ya están a la par con las soluciones comerciales.

1.2 OBJETIVOS DEL PROYECTO

1.2.1 Objetivo General

Investigar sobre las tecnologías de Sistemas de Información Geográfica empleadas en Internet para la publicación de información espacial a través de la Web, y aplicar estas tecnologías en el diseño e implementación de una herramienta orientada al proceso de oferta de servicios de venta y arrendamiento de bienes raíces que ejerce una inmobiliaria, con el propósito de demostrar su bondad y modo de utilizarlas adecuadamente.

1.2.2 Objetivos Específicos

- Profundizar sobre la extensión PostGIS del manejador de bases de datos PostgreSQL para el almacenamiento y administración de la información geográfica, y sobre el estándar de almacenamiento de datos geográficos OpenGis en el cual está basado la extensión PostGis.
- Estudiar diferentes alternativas de tecnologías Web aplicadas a SIG de origen GNU que se requieran para la ejecución del proyecto, y documentarlas adecuadamente.
- Identificar y codificar los principales requerimientos de información en el proceso de oferta de los servicios de venta y arrendamiento que realiza una inmobiliaria.
- Diseñar una herramienta Web basada en tecnologías SIG que permita realizar la administración, tratamiento, consulta y análisis de la información geográfica y alfanumérica definida en el objetivo anterior.

- Diseñar e implementar una base de datos espacial que soporte la herramienta web basada en sistemas de Información geográfica, que permita la centralización y distribución de los mapas utilizados.
- Implementar diferentes tipos de análisis espacial sobre la información geográfica pertinente a la aplicación propuesta.
- Desarrollar un prototipo software que lleve a cabo el proceso de oferta de inmuebles para la venta o arriendo de una inmobiliaria en el cual se aplique las funcionalidades de los SIG y las Bases de Datos espaciales, y que permita:
 - ✓ El tratamiento y administración de la información geográfica y alfanumérica.
 - ✓ La publicación de mapas en Internet.
 - ✓ Ubicar espacialmente los predios disponibles para arrendamiento o venta.
 - ✓ Realizar consultas espaciales que faciliten a los usuarios tomar decisiones más acertadas en el momento de decidirse por un inmueble, permitiendo acceder a información no tradicional, como las características generales complementarias del entorno del inmueble seleccionado cercanías a comodidades como zonas verdes, servicios, Colegios, bibliotecas, hospitales, parques, entre otros, o condiciones sobre existencia de facilidades, como la proximidad de alguna iglesia, o lejanía de una cárcel, por ejemplo.
 - ✓ Enlazar fotos, videos, planos y/o sonidos a la localización del predio o a

alguna de sus características, y que un supuesto usuario pueda ver u oír estos elementos multimediales y se forme una idea más completa del predio ofrecido.

1.3 JUSTIFICACION

El problema es importante desde el punto de vista de quien quiere buscar una vivienda o un local, por lo incómoda que se hace tal búsqueda con la metodología que se usa actualmente en la ciudad. Consideramos que una alternativa SIG puede ser la solución, pero antes de poderla ofrecer a una entidad inmobiliaria debemos verificar que es factible y funcional.

La EISI necesita proyectos que puedan rentarle, ya que la UIS no apoya financieramente en forma completa a las Escuelas, y la única forma que tiene una Escuela como la EISI para mejorar sus recursos es vendiendo servicios, como hacen otras Escuelas. Esos servicios pueden ser aplicaciones software, como muchas que se han desarrollado pero que no se han sabido comercializar.

En la ciudad existen muchas empresas inmobiliarias, ninguna tiene un sistema adecuado para mostrar a sus posibles clientes las opciones disponibles; en cada una de estas entidades existe una lista textual de alternativas, el cliente tendrá que iniciar un proceso consistente en seleccionar las posibles e ir a verlas, para lo cual debe hacerse acompañar de un funcionario de la inmobiliaria, esperar que haya disponibilidad de las llaves (generalmente hay un solo juego de llaves y no se usa que vayan dos clientes distintos a la vez a ver un mismo predio), no es frecuente que haya disponibilidad de las llaves y de un funcionario acompañante, por lo que en una mañana pocas veces pueden verse más de dos de tales predios.

En cuanto al uso de software SIG de licencia pública se justifica por la facilidad de

acceso a este debido al bajo costo económico que representan; el uso de aplicaciones SIG comerciales se traducen en inversiones económicas para las empresas o como en este caso para proyectos de investigación que no cuentan con estos recursos y necesitan usar esta información. Además, las herramientas SIG y gestores de bases de datos con extensiones espaciales de licencia GPL, cuentan con excelentes funcionalidades que los convierten en fuertes competidores de las soluciones comerciales.

1.4 VIABILIDAD

El proyecto es factible tecnológicamente debido a que existen en el mercado las herramientas con las que se realizará la implementación del proyecto, entre los que se encuentran: servidores de mapas, servidores Web, manejador de bases de Datos con extensión espacial y herramientas software para desarrollo Web. Todos ellos bajo licencia pública, lo cual disminuirá notablemente los costos del proyecto

La aplicación de tecnologías SIG será viable no solo a inmobiliarias, sino que podrá tener otras aplicaciones, ya que se beneficiará la comunidad educativa con la documentación de las herramientas que se investigarán para desarrollar otros proyectos.

Presenta viabilidad económica debido a que el proyecto se puede realizar con herramientas libres, el presupuesto incluirá solo mano de obra, es algo requerido por la comunidad, haciéndolo asequible.

2. MARCO TEORICO

2.1 SISTEMAS DE INFORMACION GEOGRAFICA

Un Sistema de Información Geográfica (SIG) es la unión de herramientas informáticas e información que se fundamenta en datos georreferenciados¹ para su análisis con unos objetivos concretos. También son conocidos por sus siglas en inglés GIS (Geographic Information System). Los aportes más importantes de los SIG respecto a otros sistemas de información, son el marco de referencia donde se organiza y su capacidad de realizar análisis espaciales.

Los Sistemas de Información Geográfica se han desarrollado a partir de la unión de diversos tipos de aplicaciones informáticas y científicas. Entre las aplicaciones informáticas se encuentran: la cartografía automática tradicional, los sistemas de gestión de bases de datos, las herramientas de análisis digital de imágenes, los sistemas de ayuda a la toma de decisiones y las técnicas de modelamiento físico. Entre las aplicaciones científicas están: Geografía, Ciencias Ambientales, Biología, Economía, Informática, Ingeniería, y otras.

La mayor utilidad de un SIG está íntimamente relacionada con la capacidad que posee éste de construir modelos o representaciones del mundo real a partir de las bases de datos digitales; esto se logra aplicando una serie de procedimientos específicos que generan aún más información para el análisis.

¹ Las coordenadas de un punto que permiten su ubicación en el espacio terrestre.

2.1.1 Definición

Un Sistema de Información Geográfica es el Conjunto de métodos, herramientas y actividades que actúan coordinada y sistemáticamente para soportar la captura, administración, manipulación, análisis, modelamiento, graficación y despliegue de información, tanto gráfica como descriptiva de los elementos considerados para resolver problemas complejos de planeación y administración.

Un SIG almacena información acerca del mundo como una colección de capas temáticas que pueden relacionarse geográficamente. Este concepto ha facilitado resolver muchos problemas del mundo real, desde rastrear vehículos o registrar detalles de aplicaciones de planificación, hasta modelos más complejos como la circulación atmosférica global.

La información geográfica contiene una referencia geográfica explícita (tal como latitud y longitud) o una referencia implícita (como el domicilio y código postal, un sitio o el nombre de una calle). Las referencias implícitas pueden derivarse de referencias explícitas utilizando un proceso automatizado llamado "geocodificación". Estas referencias geográficas permiten localizar en la superficie de la tierra para su análisis, características tales como negocios o bosques, o eventos como un terremoto.

2.1.2 Componentes de un SIG

Son cinco los elementos que componen un sistema de información geográfica:



Figura 1 Diagrama de componentes SIG ²

➤ **Hardware**

Los SIG corren en un amplio rango de tipos de computadoras, desde equipos centralizados hasta configuraciones individuales o de red. Una organización requiere de hardware suficientemente específico para cumplir con las necesidades de aplicación.

➤ **Software**

Los programas SIG proveen las herramientas y funcionalidades necesarias para almacenar, analizar y mostrar información geográfica. Los componentes principales del software SIG son:

- ✓ Sistema de manejo de base de datos.
- ✓ Una interfaz gráfica de usuarios (GUI) para el fácil acceso a las herramientas.
- ✓ Herramientas para captura y manejo de información geográfica.
- ✓ Herramientas para soporte de consultas, análisis y visualización de datos geográficos.

➤ **Información**

² Tomada de <http://www.monografias.com/trabajos25/sistema-geografico/sistema-geografico.shtml>

El componente más importante para un SIG es la información. Se requiere de adecuados datos de soporte para que el SIG pueda resolver los problemas y contestar a preguntas de la forma más acertada posible. La consecución de datos correctos generalmente absorbe entre un 60 a 80% del presupuesto de implementación del SIG, y la recolección de los datos es un proceso largo que frecuentemente demora el desarrollo de productos que son de utilidad. Los datos geográficos y alfanuméricos pueden obtenerse por recursos propios u obtenerse a través de proveedores de datos. Mantener, organizar y manejar los datos debe ser política de la organización.

Es indiscutible que los datos son el principal activo de cualquier sistema de información. Por ello el éxito y la eficacia de un SIG se miden por el tipo, la calidad y vigencia de los datos con los que opera.

➤ **Personal**

Las tecnologías SIG son de valor limitado si no se cuenta con los especialistas en manejar el sistema y desarrollar planes de implementación del mismo. Sin el personal experto en su desarrollo, la información se desactualiza y se maneja erróneamente, el hardware y el software no se manipula en todo su potencial.

➤ **Métodos**

Para que un SIG tenga una implementación exitosa debe basarse en un buen diseño y reglas de actividad definidas, que son los modelos y prácticas operativas exclusivas en cada organización.

2.1.3 Modelos y Estructuras de Datos

Al iniciar el estudio para diseñar un SIG, debe pensarse que se van a manejar objetos que existen en la realidad, tienen características que los diferencien y

guardan ciertas relaciones espaciales que se deben conservar; por lo tanto, no se puede olvidar en ningún caso que se va a desarrollar en el computador un modelo de objetos y relaciones que se encuentran en el mundo real.

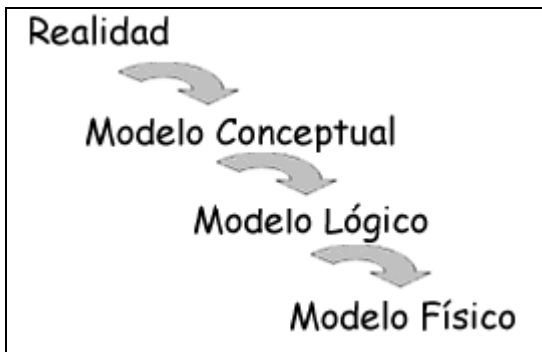


Figura 2. Modelo de Datos

Para garantizar que el esquema anterior se pueda obtener, se construye una serie de modelos que permitan manipular los objetos tal cual como aparecen en la realidad, con esto, se convertirán las imágenes de fenómenos reales en señales que se manejan en el computador como datos que harán posible analizar los objetos que ellas representan y extraerles información.

Normalmente se llevan a cabo tres etapas para pasar de la realidad del terreno al nivel de abstracción que se representa en el computador y se maneja en los SIG y que definen la estructura de los datos, de la cual dependerán los procesos y consultas que se efectuarán en la etapa de producción:

➤ **Modelo conceptual.**

Es la conceptualización de la realidad por medio de la definición de objetos de la superficie de la tierra (entidades) con sus relaciones espaciales y características (atributos) que se representan en un esquema describiendo esos fenómenos del mundo real. Para obtener el modelo conceptual, el primer paso es el análisis de la información y los datos que se usan; el siguiente paso es la determinación de las entidades y los atributos con las

relaciones que aquellas guardan, de acuerdo con el flujo de información en los diferentes procesos que se llevan a cabo en la empresa.

➤ **Modelo lógico**

Se puede definir como el diseño detallado de las bases de datos que contendrán la información alfanumérica y los niveles de información gráfica que se capturarán, con los atributos que describen cada entidad, identificadores, conectores, tipo de dato (numérico o carácter) y su longitud; además, se define la geometría (punto, línea o área) de cada una de ellas.

Es en esta etapa que se elaboran las estructuras en que se almacenarán todos los datos, tomando como base el modelo conceptual desarrollado anteriormente. Se trata de hacer una descripción detallada de las entidades, los procesos y análisis que se llevarán a cabo, los productos que se espera obtener y la preparación de los menús de consulta para los usuarios.

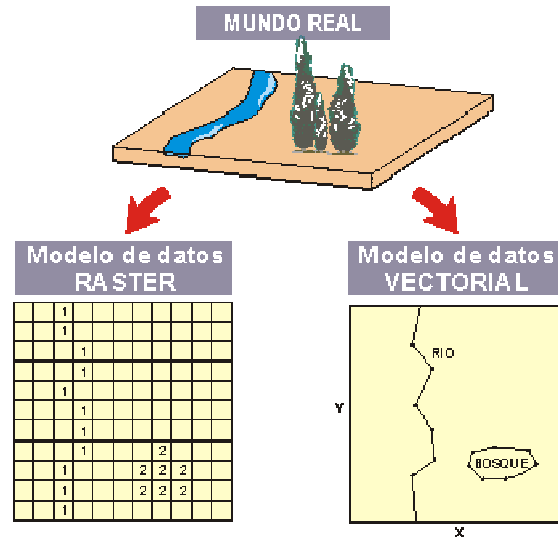


Figura 3. Esquema de los Modelos Lógicos³

³ Tomada de: <http://recursos.gabrielortiz.com/>

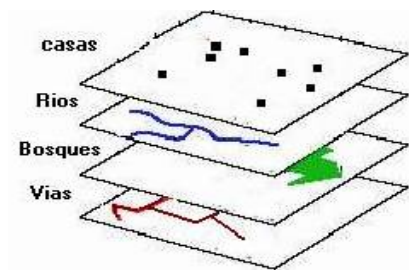
Una vez definido el modelo conceptual y el lógico, se conoce cuáles mapas se han de digitalizar y qué información alfanumérica debe involucrarse. Los sistemas de información geográfica funcionan con dos tipos fundamentales de modelos: **vectorial** y **raster**.

- **Modelo Vectorial**

En el modelo vectorial los datos están representados por un sistema de referencia (x,y) que corresponde a los sistemas de coordenadas representadas como latitud/longitud. Uno de los usos más comunes de archivos vectoriales es la digitalización de información geográfica de mapas. De este modo un mapa queda reducido a una serie de pares ordenados de coordenadas, cada uno con atributos asociados, utilizados para representar puntos, líneas y superficies.

Cada punto en el mapa de papel asume una de las tres formas en el archivo vectorial: punto, línea o polígono. Un punto

está representado por un solo par de coordenadas. Una línea está compuesta de un grupo de coordenadas (puntos) agregadas para formar el elemento que en un mapa pareciera no tener medidas de anchura (por ejemplo, rutas o ríos pequeños). Un polígono es un conjunto de coordenadas unidas por líneas, en el cual el primero y el último punto son iguales. El polígono siempre representa un área cerrada, como por ejemplo una zona de producción económica o una ciudad, las que ocupan un área que puede ser incluida en un mapa en una escala visible.



La captura de la información en el formato vectorial se hace por medio de: mesas digitalizadoras, convertidores de formato raster a formato vectorial, sistemas de geoposicionamiento global (GPS), entrada de datos

alfanuméricos, entre otros.

- **Modelo Raster**

En el modelo raster los mapas están representados por una rejilla. Un mapa está dividido en una serie de filas y columnas, horizontal y vertical, en las cuales cada elemento de la cuadrícula es llamado "celda". A cada celda se le asigna un valor que representa un elemento particular del mapa (por ejemplo, todas las celdas con un valor de 1 pueden representar algún tipo de suelo, agua las celdas con un valor de 2, etc.). Este modelo interpreta los puntos, líneas y polígonos de una forma diferente a como lo hace el sistema vectorial. En él, un punto está representado por una celda completa; una columna es un grupo de celdas contiguas y tiene la anchura de una celda; un polígono está representado por un grupo contiguo de celdas con la anchura de una o más celdas.

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: scanners, imágenes de satélite, fotografía aérea, cámaras de video entre otros.



Figura 4. Modelo raster

- **Modelo físico**

Es la implementación de los anteriores modelos en el software seleccionado y los equipos específicos en que se vaya a trabajar, por esto se realiza de acuerdo con sus propias especificaciones. El modelo físico determina en qué forma se debe almacenar los datos, cumpliendo con las restricciones y aprovechando las ventajas del sistema específico a utilizar.

2.1.4 SIG en Internet

La Internet ha evolucionado hasta convertirse en una plataforma informática. Para los usuarios de los sistemas de información geográfica con el desarrollo de la tecnología cliente- servidor ha hecho posible la implementación de aplicaciones que ha permitido hacer aplicaciones dinámicas.

El factor que ha permitido que esto se pueda llevar a cabo ha sido el uso de la tecnología Internet MapServer (IMS). A partir de esta tecnología, se han desarrollado varios sistemas como MapObjects, Arcview, ArcIMS, las cuales permite crear aplicaciones SIG en Internet/Intranet para visualizar, consultar y analizar la información geográfica a través de la red.

➤ Tecnología Internet MapServer (IMS)

Con la tecnología IMS, la información espacial publicada en la red es dinámica. La distribución de información geográfica vía Internet permite la integración en tiempo real de datos procedentes de cualquier parte del mundo. El usuario tiene acceso a los recursos de la red, se desplaza libremente por toda la información con herramientas funcionales, cambia la representación gráfica en línea, enlaza elementos gráficos con informaciones procedentes de bases de datos, y trabaja en tiempo real con funciones de análisis.

Las opciones de intercambiar, integrar o analizar datos de una nueva forma a

través de la red, facilitan, agilizan y favorecen el proceso para la toma de decisiones. Los usuarios pueden combinar datos e información accesible vía Internet con datos locales, visualizarlos, hacer consultas y el análisis pertinente.

Ofrece las siguientes ventajas:

- Compartir e intercambiar datos.
- Dar acceso a aplicaciones y herramientas para el análisis y toma de decisiones a un público mucho más amplio.
- Facilitar la actualización continuada de la información, ayudando a reducir redundancias (duplicaciones) y mejorando el acceso a bases de datos.
- Facilitar la actualización de aplicaciones e información divulgada.

La arquitectura IMS consta de tres niveles:



Figura 5. Arquitectura Internet MapServer (IMS)

Aplicaciones Cliente: Entorno de trabajo del usuario. Cualquier navegador que soporte el estándar HTML puede actuar como cliente. Será necesario que también soporte Applet (Plug-in) de Java o tecnología ActiveX si los servicios a los que se accede contienen estos componentes. A través de Internet y con el navegador como interfaz, el Cliente envía peticiones a la Aplicación Servidor para obtener la información que le interesa visualizar, consultar o analizar.

Aplicaciones Servidor: Son las encargadas de canalizar y atender las operaciones que el usuario solicita sobre los datos: ArcView IMS, MapObjects IMS, ArcIMS, etc.

Bases de Datos: Las aplicaciones acceden a los datos que pueden estar almacenados en archivos o en bases de datos espaciales.

2.2 BASE DE DATOS ESPACIALES

Una base de datos espacial (BDE) es un conjunto de datos referenciados espacialmente, relacionados entre sí, clasificados y agrupados según sus características; bajo control de redundancias e integrados para el desarrollo de aplicaciones y análisis sobre la información; dicho conjunto actúa como modelo de la realidad.

A partir de una BDE debe ser posible: generar un mapa, hacer operaciones espaciales simples (como cálculo de un área, determinar el punto con la máxima o mínima coordenada, entre otras), realizar una selección geométrica (seleccionar una ventana a partir de un mapa), establecer el cálculo de algunas métricas espaciales (como distancias, áreas, volúmenes.), ejecutar funciones topológicas (como encontrar los vecinos, determinar cuáles son visibles desde cierto punto, etc), efectuar una interpolación, extrapolación o localización (como seleccionar un sitio de acuerdo a los recursos, asignar recursos de acuerdo a características de mercadeo o de planeación urbana, por mencionar algunas).

2.2.1 Tipos de Datos

A los datos almacenados en una BDE se le asocian unos atributos que pueden ser *espaciales y alfanuméricos*.

Los atributos gráficos son las representaciones de los objetos geográficos asociados con ubicaciones específicas en el mundo real. La representación de los objetos se hace por medio de puntos, líneas o áreas. Los atributos alfanuméricos corresponden a las descripciones, calificaciones o características que nombran y

determinan los objetos o elementos geográficos.

Los datos espaciales (puntuales, lineales, poligonales) responden al dónde, qué y cuándo de un fenómeno espacial, y se refieren a entidades o fenómenos que cumplen los siguientes principios básicos:

- Tienen posición absoluta; sobre un sistema de coordenadas (x, y, z)
- Tienen una posición relativa; frente a otros elementos del paisaje (topología: incluido, adyacente, cruzado, etc.).
- Tienen una figura geométrica que las representan (punto, línea, polígono).
- Tienen atributos que lo describen (características del elemento o fenómeno).

2.2.2 Información Espacial

Se puede hablar de tres aspectos distintos dentro de la componente espacial: la localización geográfica, las relaciones espaciales y las propiedades espaciales.

- La localización, también denominada posición, respondería a preguntas como ¿dónde está localizada la entidad X? o ¿qué entidad se encuentra en la posición Y? Se trata de referenciar cualquier hecho geográfico respecto al globo terrestre. Esta referenciación adopta el nombre de Georreferenciación.
- Las relaciones espaciales, determinan las interrelaciones geométricas y topológicas entre elementos en el espacio. Es decir responden a preguntas como: ¿qué está junto a?, ¿Qué hay a menos de 500 metros del punto X?, ¿Qué distancia hay entre los puntos A y B?, ¿Cuál es la siguiente parada en la línea 1 de autobuses?, etc. Todas ellas se pueden responder a partir de la geometría o la topología. La topología, es el método matemático-lógico usado para definir las relaciones espaciales entre los objetos geográficos

Como resumen de lo anteriormente expuesto se podría hablar de cuatro grandes características, una geométrica y tres topológicas. La distancia entre dos puntos sería geométrica, y la contigüidad, conectividad e inclusión serían topológicas.

Las propiedades espaciales, vienen determinadas por el tipo de elemento espacial que contemplemos. Dependiendo de si hablamos de puntos, líneas o polígonos.

- Puntos: Se encuentran determinados por las coordenadas terrestres medidas por latitud y longitud. Por ejemplo, ciudades, accidentes geográficos puntuales, hitos. Es un objeto sin dimensiones que representa una unión topológica o un punto terminal y que especifica una localización geométrica; en cualquier caso, se trata de la entidad básica para representar entidades con posición pero sin dimensión.
- Líneas: Son objetos de una dimensión definidos por un nodo inicio y un nodo fin. Objetos abiertos que cubren una distancia dada y comunican varios puntos o nodos, aunque debido a la forma esférica de la tierra también se le consideran como arcos. Líneas telefónicas, carreteras y vías de trenes son ejemplos de líneas geográficas. Las líneas tienen como propiedades espaciales la longitud, forma, pendiente y orientación.
- Polígonos: Constituye el tipo de datos más común usado en SIG, Son objetos limitados y continuos de dos dimensiones. Son Figuras planas conectadas por distintas líneas u objetos que cubren un área determinada, como por ejemplo países, regiones o lagos. Los polígonos pueden tener características como el perímetro, la forma, la orientación y la pendiente.

La otra forma de expresar datos espaciales es mediante rasterización, la cual, a través de una malla que permite asociar datos a una imagen; es decir, se pueden relacionar paquetes de información a los píxeles de una imagen digitalizada.

En todos los casos debemos distinguir entre el objeto real y el elemento que lo representa en el proceso de su conceptualización digital. Por ejemplo un río puede

ser representado por una única línea cuyo ancho puede ser arbitrario mientras en la realidad el ancho está claramente determinado.

2.2.3 Información Alfanumérica

Constituida por los atributos asociados a cada una de las entidades espaciales. Los atributos alfanuméricos son aquellos que representan las propiedades de una entidad que no tienen relación con los aspectos geométricos y topológicos. Por ejemplo, el nombre, el tipo y el número de canales de una carretera. La gama de posibilidades es muy amplia y su clasificación heterogénea.

2.2.4 Sistema Manejador de Base de Datos (SMBD)

Es el conjunto de programas que administra una Base de Datos, (DBMS por sus siglas en inglés), que según su disposición física pueden ser centrales o distribuidas, y según su disposición lógica pueden ser relacionales u orientadas a objetos. Aunque también existen bases de datos de fichero simple, jerárquicas y en red, hoy en día para los SIG sólo se manejan de tipo relacional u orientado a objetos.

➤ Base de Datos Relacionales

De amplio uso en los SIG, garantizan la organización de todas las entidades con sus relaciones en un solo esquema de representación de las cosas. Con este modelo se obtiene un medio efectivo para mostrar los requerimientos de información, organización y documentación necesarios para desarrollar el SIG y las clases de datos que se estarán manipulando.

Una base de datos relacional es básicamente un conjunto de tablas, similares a las tablas de una hoja de cálculo, formadas por filas (registros) y columnas (campos). Los registros representan cada uno de los objetos

descritos en la tabla y los campos los atributos (variables de cualquier tipo) de los objetos. En el modelo relacional de base de datos, las tablas comparten algún campo entre ellas. Estos campos compartidos van a servir para establecer relaciones entre las tablas que permitan consultas complejas. Las BD relacionales hace énfasis en la computación lógica, es decir en el orden de procesamiento eficiente. La información clave acerca de un objeto en una Base de Datos Relacional es definida por el valor de un campo en un renglón de cada tabla. Cada renglón puede ser entendido de forma independiente.

Para poder explicar cómo funcionaría el Modelo Relacional en una Base de Datos en un SIG, hay que tener en cuenta que los Sistemas de Información Geográfica se basan en el manejo de Capas para administrar mejor la información; cada Capa representa un concepto que tiene unas características definidas, por ejemplo, los ríos; entonces, al guardar la información de un mapa, que está compuesto por la capa de ríos, la capa de carreteras y la capa de ciudades se generarían tantas tablas como capas se tengan y estas a su vez generan los renglones y columnas que se necesiten.

➤ **Base de Datos Orientada a Objetos**

Se llaman así porque cada elemento del paisaje se maneja efectivamente como un objeto al que se le definen sus componentes, variables y métodos de análisis propios. A diferencia de las demás, en las BDOO cada objeto sólo se define una vez y no hay necesidad de repetirlo en varias capas para obtener relaciones. Una línea se podría capturar una sola vez y asociarse como objeto con los atributos de río, vía y límite municipal, en vez de tener que tener estos atributos en tres capas diferentes.

Las BD OO intentan capturar de manera más fiel la semántica de los

objetos del mundo real, facilitando la construcción de sistemas complejos; esto debido a que la orientación a objetos de este modelo provee mejores conceptos y herramientas con el cual modela y representa el mundo real tan cercano como le sea posible. Las BD que están basadas en el modelo OO descomponen la realidad en objetos, esta descomposición depende del juicio y la naturaleza del problema. Los Objetos en las BDOO dentro de los SIG representan las entidades tanto espaciales como temáticas y su función primordial es el proceso de captura de propiedades y comportamiento de la entidad representada.

Las BD en un SIG que utiliza la metodología OO puede generar los tipos de datos necesarios para guardar y estructurar su información de una forma más eficiente; esto se logra gracias a los conceptos y herramientas que maneja el enfoque Orientado a Objetos, como objeto, clase, polimorfismo, herencia, subclase, superclase, especialización y agregación, y que a su vez aportan otras ventajas, como por ejemplo, el aumento del rendimiento de la base de datos gracias a la reutilización automática de métodos y datos de uno a otro objeto al implementar el concepto de herencia.

3. METODOLOGIA

3.1 PROTOTIPADO INCREMENTAL

Es un modelo que toma sus bases del prototipado simple, pero posee mayores controles sobre la calidad y desarrolla primero las áreas de mayor riesgo del sistema, de tal forma que el prototipo pueda ser tomado como producto final una vez se llegue a su fin. Es decir, en este modelo se desarrolla el concepto del sistema a medida que avanza el proyecto. A diferencia de otros tipos de prototipado, en el evolutivo no se descarta el código del prototipo; lo transforma en el código entregado finalmente.

El desarrollo de prototipos es continuo hasta que se decide que el prototipo es lo suficientemente bueno para entregar como producto final.

3.2 LENGUAJE UNIFICADO DE MODELADO (UML)



El Lenguaje Unificado de Modelado (Unified Modelling Language, UML) es el lenguaje estándar de modelado más conocido y utilizado en la actualidad para escribir planos de software. UML es el lenguaje gráfico para visualizar, especificar, construir y documentar un sistema que involucra gran cantidad de software; y es apropiado para modelar desde sistemas de información en empresas hasta aplicaciones distribuidas basadas en la Web.

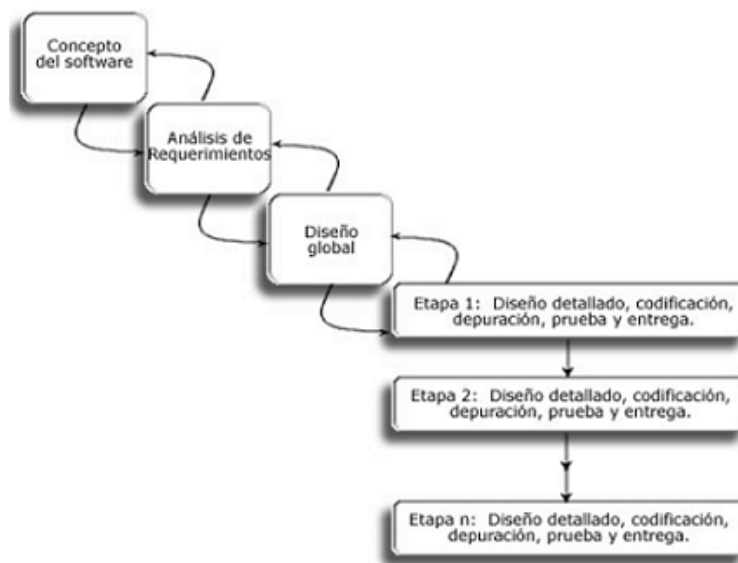


Figura 6. Modelo Prototipado Incremental⁴

El objetivo principal del Lenguaje Unificado de Modelado es entregar un material de apoyo que permita definir diagramas en el proceso de desarrollo de software, así como también facilitar el entendimiento del modelamiento de diagramas ya existentes.

UML es sólo un lenguaje y por tanto es tan sólo una parte de un método de desarrollo de software. UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

3.2.1 Generalidades de UML

UML es un lenguaje que proporciona un vocabulario y las reglas para combinar palabras de este vocabulario con el objetivo de posibilitar la comunicación. Un

⁴ El Proceso Unificado de Desarrollo de Software, BOOCH Grady, RUMBAUGH James, JACOBSON Ivar. Addison-Wesley. Pearson Educación S.A. Madrid 2000.

lenguaje de modelado es un lenguaje cuyo vocabulario y reglas se centran en la representación conceptual y física de un sistema.

UML no es sólo un montón de símbolos gráficos. Detrás de cada símbolo en la notación UML hay una semántica bien definida. De esta manera, un desarrollador puede escribir un modelo en UML, y otro desarrollador o incluso otra herramienta, puede interpretar ese modelo sin ningún problema, al escribir modelos en UML se facilita la comunicación.

El vocabulario y las reglas de un lenguaje como UML indican cómo crear y leer modelos bien formados, pero no dicen qué modelos se deben crear ni cuándo se deberían crear. Esta es tarea del proceso de desarrollo de software. Un proceso bien definido guiará a sus usuarios al decidir qué artefactos producir, qué actividades y qué personal se emplea para crearlos y gestionarlos, y cómo usar esos artefactos para medir y controlar el proyecto de forma global.

Los objetivos de UML son muchos, pero se pueden resumir sus funciones como lo muestra la siguiente tabla:

Visualizar	Representar mediante imágenes y/o símbolos el contenido y la organización de los conceptos que configuran un sistema. Hacer visible su naturaleza y su complejidad.
Especificar	UML permite especificar los procesos de análisis, diseño y codificación de un sistema software. También permite determinar modelos precisos, sin ambigüedades, detallando las partes esenciales.
Construir	A partir de los modelos especificados con UML se puede generar código en distintos lenguajes de programación y

	tablas en una base de datos a partir de modelos UML. Además permite simular el comportamiento de sistemas software.
Documentar	UML cubre la documentación de la arquitectura de un sistema y todos sus detalles. Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión. UML también proporciona un lenguaje para expresar requisitos y pruebas. También proporciona un lenguaje para modelar las actividades de planificación de proyectos

Tabla 1 Funcionalidades de UML

El lenguaje UML se compone de tres elementos básicos, los bloques de construcción, las reglas y algunos mecanismos comunes. Estos elementos interactúan entre sí para dar a UML el carácter de completitud y eliminar la ambigüedad.

Los bloques de construcción se dividen en tres partes: Elementos, que son las abstracciones de primer nivel, Relaciones, que unen a los elementos entre sí, y los Diagramas, que son agrupaciones interesantes de elementos.

Existen cuatro tipos de elementos en UML, dependiendo del uso que se haga de ellos: *elementos estructurales*, *elementos de comportamiento*, *elementos de agrupación* y *elementos de anotación*.

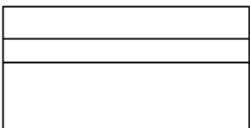
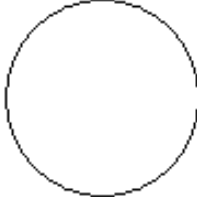

Las relaciones, a su vez se dividen para abarcar las posibles interacciones entre elementos que se nos pueden presentar a la hora de modelar usando UML, estas son: *relaciones de dependencia*, *relaciones de asociación*, *relaciones de generalización* y *relaciones de realización*.


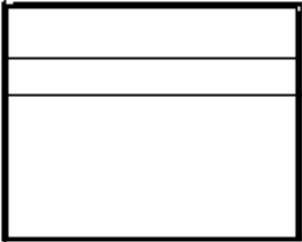
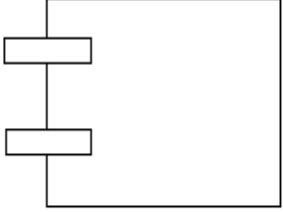
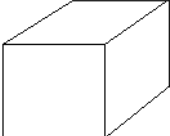
Se utilizan diferentes diagramas dependiendo de qué nos interese representar en cada momento, para dar diferentes perspectivas de un mismo problema.

A continuación se muestra una breve descripción de los Elementos, Relaciones y los Diagramas.

3.2.2 Elementos

- **Estructurales:** Son las partes estáticas del modelo y estos representan cosas materiales o conceptuales:

Clase	Descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica.	
Interfaz	Es una colección de operaciones que especifican un servicio de una determinada clase o componente, describe el comportamiento visible externamente de ese elemento y suele estar conectada a la clase o componentes que realiza.	
Colaboración	Define una interacción y es una sociedad de roles y otros elementos que colaboran para proporcionar un comportamiento cooperativo mayor que la suma de los comportamientos de sus elementos. Una clase puede participar en varias colaboraciones	

Caso de Uso	<p>Es la descripción de un conjunto de acciones que un sistema ejecuta y que produce un</p> <p>Determinado resultado que es de interés para un actor particular. Se utiliza para organizar los aspectos del comportamiento en un modelo. Es realizado por una Colaboración</p>	
Clase Activa	<p>Es una clase cuyos objetos tienen procesos o hilos de ejecución, pueden dar lugar a actividades de control. Es igual que una clase, excepto que sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos.</p>	
Componente	<p>Es una parte física y reemplazable de un sistema que conforma con un conjunto de interfaces y proporciona la implementación de dicho conjunto. Un componente representa típicamente el empaquetamiento físico de diferentes elementos lógicos, como clases, interfaces y colaboraciones.</p>	
Nodo	<p>Es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional que dispone de algo de memoria y de capacidad de procesamiento. Un conjunto de componentes puede residir en un nodo y</p>	

	puede migrar también de un nodo a otro.	
--	---	--

Tabla 2 Elementos Estructurales de UML

- **Comportamiento:** Son las partes dinámicas de los modelos UML y representan el comportamiento en el tiempo y en el espacio.



Interacción	Es un comportamiento que comprende un conjunto de mensajes intercambiados entre un conjunto de objetos. Una interacción involucra otros muchos elementos, incluyendo mensajes, secuencias de acción y enlaces.	
Máquina de Estados	Es un comportamiento que especifica las secuencias de estados por las que van pasando los objetos o las interacciones durante su vida en respuesta a eventos, junto con sus respuestas.	

Tabla 3 Elementos de Comportamiento

- **Agrupación:** Forman la parte organizativa de los modelos UML.


Paquete	Su función es organizar los elementos en grupos. Un paquete solo existe en tiempo de desarrollo	
----------------	---	---

Tabla 4 Elementos de Agrupación

- **Anotación:** Son las partes explicativas de los modelo UML. Son comentarios para describir, clarificar y hacer las observaciones pertinentes sobre cualquier elemento de un modelo.

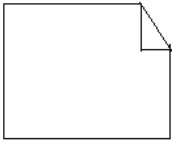


Nota	Es un símbolo para mostrar restricciones y comentarios junto a un elemento o un conjunto de elementos.	
-------------	--	---

Tabla 5 Elementos de Anotación

3.3.3 Relaciones

Existen cuatro tipos de relaciones entre los elementos, se describirán a continuación

Dependencia	Es una relación semántica entre dos elementos, donde un cambio a un elemento independiente puede afectar a la semántica del otro elemento dependiente.	
Asociación	Es una relación estructural que describe un conjunto de enlaces, los cuales son conexiones entre objetos. La agregación es un ejemplo especial de asociación y representa una relación estructural entre un todo y sus partes.	

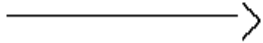
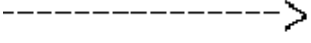
Generalización	Es una relación en la cual los objetos del elemento especializado pueden sustituir a los objetos del elemento general. De esta forma, el elemento especializado comparte la estructura y el comportamiento del elemento general.	
Realización	Se encuentran entre interfaces, las clases y los componentes que las realizan. También entre los casos de uso y las colaboraciones que los realizan.	

Tabla 6 Relaciones de UML

3.2.4 Diagramas de UML

Un diagrama es la representación gráfica de un conjunto de elementos, el cual se representa la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones). UML ofrece una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas. UML incluye los siguientes diagramas:

Diagrama de Clases	Muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones
Diagrama de Objetos	Muestra un conjunto de objetos y sus relaciones
Diagrama de Casos de Usos	Muestran un conjunto de casos de uso y actores (un tipo especial de clases) y sus

	relaciones. Estos diagramas son especialmente importantes en el modelado y organización del comportamiento de un sistema.
Diagrama de Secuencia y Diagrama de Colaboración	Son un tipo de diagrama de interacción que como su nombre lo indica muestra una interacción, que consta de un conjunto de objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos. Un diagrama de secuencia es un diagrama de interacción que resalta la ordenación temporal de los mensajes; un diagrama de colaboración es un diagrama de interacción que resalta la organización estructural de los objetos que envían y reciben mensajes.
Diagrama de Estados	Muestra una máquina de estados, que consta de estados, transiciones, eventos y actividades. Son importantes en el modelado del comportamiento de una interfaz, una clase o una colaboración y resaltan el comportamiento dirigido por eventos de un objeto, lo cual es útil en el modelado de sistemas reactivos.
Diagrama de Actividades	Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema. Son importantes al modelar el funcionamiento de un sistema y resaltan el flujo de control entre objetos.
Diagrama de Componentes	Muestra la organización y las dependencias

	entre un conjunto de componentes. Se relacionan con los diagramas de clases en que un componente se corresponde, por lo común, con una o más clases, interfaces o colaboraciones.
Diagrama de Despliegue	Muestra la configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. Se relacionan con los diagramas de componentes en que un nodo incluye, por lo común, uno o más componentes.

Tabla 7 Diagramas de UML

4. TECNOLOGÍAS SOFTWARE EMPLEADAS

4.1 MAPSERVER



MapServer es un ambiente de desarrollo que permite construir aplicaciones Web interactivas que permiten la visualización y consulta de información geográfica en forma de mapas. MapServer no es un software que posea por completo las características de un SIG, sin embargo proporciona la funcionalidad suficiente para soportar una amplia variedad de aplicaciones Web que ofrezcan algunas características propias de los sistemas de información geográfica. Es lo que se ha venido a denominar dentro de la industria SIG una aplicación de 'WebMapping'.

MapServer es un proyecto de carácter libre, distribuido bajo licencia **GPL**⁵. Este software fue desarrollado por la Universidad de Minnesota. Actualmente esta Universidad en compañía del proyecto TerraSIP, patrocinado por la NASA, y más de veinte desarrolladores, trabajan en su mantenimiento y evolución

El esquema de funcionamiento que presenta MapServer es el siguiente: El usuario ingresa a través de su navegador a la aplicación desarrollada sobre MapServer, dicha aplicación debe estar ubicada en un servidor Web, para que de esta forma se realice el enlace a MapServer, quien se encargará de permitir la manipulación de la información cartográfica a la que tiene acceso por medio de la aplicación.

⁵ Ver Anexo C

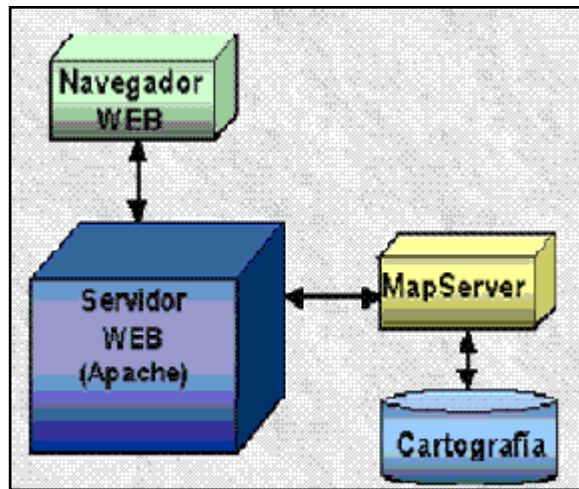


Figura 7 Esquema de funcionamiento MapServer

Mapserver puede operar en dos modos diferentes: CGI⁶ (Common Gateway Interface) y MapScript (API de MapServer).

En el modo CGI, MapServer funciona en ambiente servidor WEB como un script, es de fácil instalación y produce de forma rápida aplicaciones sencillas. Cuando se corre como CGI, MapServer puede mostrar mapas, mostrar atributos y realizar consultas espaciales simples.

En el modo MapScript, el API de MapServer es accesible desde Perl, Python, PHP o Java. La interfaz MapScript permite una aplicación flexible y significativamente más poderosa. En este ambiente MapServer puede realizar las mismas tareas que como si estuviera en ambiente CGI, pero también puede acceder a base de datos externas, realizar operaciones complejas y un largo repertorio de

⁶ Los programas CGI son un rasgo de la WWW que amplía enormemente la capacidad de documentos HTML sencillos al permitir la interacción con datos y/o aplicaciones informáticas que existen en un computador servidor conectado a la WWW. CGI es una norma para establecer comunicación entre el servidor WEB y el programa, de tal forma que este último pueda interactuar con Internet

comportamientos.

MapServer tiene una poderosa y sofisticada capacidad de consultas. Pero en modo CGI este carece de herramientas que le permitan proporcionar un buen análisis para un verdadero SIG.

Para instalar MapServer se recomienda los paquetes FGS para Linux y MS4W para Windows, que se pueden encontrar del sitio WEB <http://www.maptools.org>, ambos paquetes instalan MapServer con PHP/MapScript, librerías para manipulación de imágenes, entre otros archivos necesarios para el soporte de aplicaciones de mapas.

Principales Características:

- Cuenta con un archivo de configuración llamado Mapfile⁷, en el que se incluye una serie de parámetros que describen las capas disponibles y otros componentes del mapa.
- Funciona directamente con shapefiles⁸ que es el formato de almacenamiento de datos de SIG alternativo disponible en ArcView de ESRI⁹, posiblemente la entidad comercial más común en SIG.
- MapServer soporta diferentes tipos de datos vectoriales, incluyendo

⁷ Ver anexo A

⁸ El formato Shapefile es un formato de datos espaciales desarrollado por la compañía ESRI; es un formato vectorial de almacenamiento digital donde se guarda la localización de los elementos geográficos y los atributos asociados a ellos. Generalmente esta definido por tres archivos diferentes que poseen el mismo nombre, pero diferente extensión: .SHP(almacena las entidades geométricas), .SHX (almacena el índice de las entidades geométricas) y .DBF(almacena los atributos asociados a los objetos).

⁹ Empresa orientada a los SIG. La popularidad de sus productos ha supuesto la generalización de sus formatos de almacenamiento de datos espaciales en el campo de los Sistemas de Información Geográfica vectoriales, entre los que destaca el Shapefile. Su producto más conocido es ArcGIS.

shapefiles, geometrías PostGis, Oracle Spatial, ArcSDE, entre otras. Nativamente, MapServer soporta dos tipos de formato raster GeoTIFF y Tiff, pero puede leer otros vía el paquete GDAL¹⁰ (incluyendo bitmaps, GIFs, y JPEGs).

- Puede operar en dos modos diferentes: CGI (Common Gateway Interface) y MapScript (API de MAppserver), esta ultima puede ser adaptada a PHP (PHPMapScript), Python, Perl, Java, entre otros.
- Corre bajo plataformas UNIX/Linux, Mac OS X, Solaris, Windows 95 o superior.
- Soporta diferentes tipos de Base de datos: Oracle8i o superior (Oracle Spatial), PostgreSQL/PostGIS, ArcSDE, MySQL.
- Soporta Especificaciones Web de Open Gis Consortium (OGC)¹¹. MapServer tiene la habilidad para utilizar capas de otros servidores de mapas. En este caso, MapServer actúa como un cliente tipo WMS¹² o WFS¹³ que son estándares para la publicación de mapas en Internet

¹⁰ GDAL (Geospatial Data Abstract Layer) es una librería de código abierto para el manejo de datos geoespaciales que permite manejar una enorme cantidad de formatos de archivos raster, usando un único modelo abstracto de datos, lo que facilita en gran medida el desarrollo de aplicaciones.

¹¹ El **Open Geospatial Consortium** (OGC) agrupa a más de 250 organizaciones públicas y privadas. Su fin es la definición de estándares abiertos e interoperables dentro de los Sistemas de Información Geográfica. Persigue acuerdos entre las diferentes empresas del sector que posibiliten la interoperación de sus sistemas de geoprocesamiento y facilitar el intercambio de la información geográfica en beneficio de los usuarios.

¹² El estándar Web Map Service (WMS) define un mapa como una representación de la información geográfica en forma de un archivo de imagen digital conveniente para la exhibición en una pantalla de ordenador. Los mapas producidos por WMS se generan normalmente en un formato de imagen como PNG, GIF o JPEG, formato raster para la Web.

¹³ Web Feature Service (WFS) es un estándar hecho para mapas en formato vectorial, para compartir las capas utiliza GML, (Geography Markup Language), una extensión del formato XML, que facilita la transmisión de información geográfica incluyendo la geometría y las propiedades de los objetos, independientemente de la plataforma utilizada.

publicados por el OGC.

- Soporta proyecciones de mapas, proyecciones en tiempo real para más de mil proyecciones a través de la biblioteca proj4¹⁴
- Salida avanzada de cartografía, dibujo y etiquetado dependiente de la escala; valores de escala, símbolo y color adaptables; acceso en función de las características a datos sobre atributos; generación automática de leyendas.

4.2 PHP MAPSCRIPT

El módulo PHP/MAPSCRIPT es un módulo dinámico de PHP que hace las funciones y clases de MapScript de MapServer disponibles en un ambiente PHP. PhpMapScript sirve como puente de comunicación entre PHP y MapServer, es una extensión de PHP que permite acceder a la API de MapServer, MapScript. De esta forma el modulo PhpMapScript contiene funcionalidades que permiten realizar operaciones con datos espaciales, la reproyección de datos, la posibilidad de crear mapas dinámicamente o modificarlos si ya existen, entre muchas otras funcionalidades.

Mapscript está basado en un modelo de objetos, por lo cual el API se caracteriza propiamente como una colección de clases con métodos y atributos, Mapscript proporciona acceso a todos los servicios que provee MapServer. De esta forma el desarrollador puede crear sus propias aplicaciones utilizando el modelo de objetos

¹⁴ “Geographic Projection Library”. Es una librería de administración de proyecciones cartográficas. Esta librería se utiliza para poder realizar reproyecciones, proyección y cambio del sistema de coordenadas almacenados dentro del motor de datos. La librería proj4 se puede descargar en la dirección URL <http://www.remotesensing.org/proj/> ó en <http://proj.maptools.org/>

de MapServer gracias a la biblioteca de componentes MapScript. El API de MapScript puede ser utilizada además por otros lenguajes como Perl, Python o Java.

El módulo de PHP MapScript fue desarrollado y actualmente es mantenido por el Grupo DM Solutions y puede ser descargado en la siguiente dirección:

http://www2.dmsolutions.ca/mapserver/php_mapscript/index.html.

4.3 MAPLAB

Es un conjunto herramientas Web de tipo OpenSource que permite crear y administrar archivos de mapas para MapServer y aplicaciones Web Mapping. Cuenta con tres componentes principales: MapBrowser, GMapFactory y MapEdit. Este último provee una muy buena interfaz gráfica para crear y editar mapas para MapServer facilitando en gran medida esta tarea. MapLab es una alternativa fácil de usar para la creación y edición de archivos de mapas (*mapfiles*) y programación PhpMapScript en la creación de recursos para el servidor de mapas MapServer.

4.4 LENGUAJES WEB

4.4.1 HTML

HTML (hypertext markup language) es un lenguaje utilizado para desarrollar páginas y documentos Web y está compuesto por dos partes: la que se muestra en la pantalla (texto, imágenes, enlaces, etc.) y los códigos (formado por etiquetas o tags), transparentes al usuario y son los encargados de indicar cómo se debe mostrar la información en la pantalla.

Para editar o crear una página HTML y posteriormente visualizarla, se requiere de

un editor de texto ASCII (donde se incluyen las etiquetas necesarias para la presentación de la página) y un explorador Web como Internet Explorer o Firefox. Existen diversos editores de texto de HTML que van desde el Bloc de notas hasta otros más específicos que insertan automáticamente las etiquetas correspondientes al formato del texto que se escriba o a las imágenes que se inserten. Posteriormente, las páginas deben ser colocadas en un servidor Web para que otros usuarios puedan acceder a ellas.

Desde los inicios hasta la actualidad el lenguaje HTML ha evolucionado pasando desde un lenguaje estático hasta un lenguaje más dinámico, modificaciones aportadas sobre todo por las personas que lo utilizan. Así, una evolución en el lenguaje suele surgir de una propuesta que es adoptada por algunos clientes (browsers) y finalmente, con el uso, se verá si es eficiente y es adoptada y si es así, finalmente se incorpora al estándar.

Etiquetas básicas HTML

Las etiquetas o tags son las encargadas de indicar a los exploradores Web cómo tienen que mostrar el texto y los gráficos. Estas etiquetas normalmente se escriben entre los símbolos < y > y suelen haber dos etiquetas una que indica el comienzo y otra que indica el fin (<>; </>).

Dentro del documento hay dos zonas bien diferenciadas: el encabezamiento, delimitado por <HEAD> y </HEAD>, que sirve para definir diversos atributos de la página y el cuerpo, delimitado por <BODY> y </BODY>, donde reside la información del documento.

Con el uso de estas etiquetas podemos definir estructuras como el template la cual son objetos que se encargan de definir estructuras para mostrar y la forma de procesar dichos objetos.

4.4.2 PHP



El lenguaje PHP (Hypertext Preprocessor) es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web; es un lenguaje de estilo clásico, ya que utiliza variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como HTML o XML. Está más cercano a JavaScript o a C.

PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor como por ejemplo a una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador, normalmente en una página HTML.

Al ser PHP un lenguaje que se ejecuta en el servidor, no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

PHP es un lenguaje para la creación de sitios web del que se pueden destacar las siguientes características:

- Es un potente y robusto lenguaje de programación embebido en documentos HTML, De forma que se pueden introducir instrucciones php dentro de las páginas. Gracias a esto el diseñador gráfico de la web puede trabajar de forma independiente al programador. PHP es interpretado por el servidor (apache) generando un HTML con el resultado de sustituir las secuencias de instrucciones PHP por su salida.
- Dispone de librerías de conexión con la gran mayoría de los sistemas de gestión de base de datos para el almacenamiento de información permanentemente en el servidor.
- Proporciona soporte a múltiples protocolos de comunicaciones en Internet (HTTP, IMAP, FTP, LDAP, SNMP, etc).

- Código fuente abierto: el código del intérprete está accesible para permitir posibles mejoras o sugerencias acerca de sus desarrollo (PHP ha sido escrito en lenguaje C).
- Portable y multiplataforma: existen versiones del intérprete para múltiples plataformas (Windows 95, 98, NT, 2000, XP, Unix, Linux, etc.), esto permite que las aplicaciones puedan ser portadas de una plataforma a otra sin necesidad de modificar ni una sola línea de código.
- PHP permite, una fácil interacción con un gran número de sistemas de bases de datos relacionales (Oracle, DB2, MySQL, PostgreSQL, etc.), mientras mantiene una sintaxis sencilla y clara.
- Dispone de facilidades para el procesamiento de ficheros, funciones de tratamiento de textos, generación dinámica de imágenes, tratamiento de documentos XML etc.

4.4.3 JAVASCRIPT

JavaScript es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C.

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento, gracias a su compatibilidad con la mayoría de los navegadores modernos; con Javascript se puede crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones Javascript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

Javascript es el siguiente paso, después del HTML, que puede dar un programador de la web que decida mejorar sus páginas y la potencia de sus

proyectos. Es un lenguaje de programación bastante sencillo y pensado para hacer las cosas con rapidez, a veces con ligereza.

Entre las acciones típicas que se pueden realizar en Javascript se tienen dos vertientes. Por un lado los efectos especiales sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, Javascript permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo, etc.

4.5 ALMACENAMIENTO DE DATOS

4.5.1 POSTGRESQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) de código abierto, liberado bajo la licencia BSD (Distribución de Software Berkeley)¹⁵



Los sistemas de gestión de Bases de Datos relacionales tradicionales (DBMS,s) soportan un modelo de datos que consiste en una colección de relaciones con nombre, que contienen atributos de un tipo específico. En los sistemas comerciales actuales, los tipos posibles de datos incluyen numéricos de punto flotante, enteros, cadenas de caracteres, cantidades monetarias y fechas. PostgreSQL ofrece una potencia adicional sustancial al incorporar los siguientes conceptos adicionales básicos en una vía en la que los usuarios pueden extender fácilmente el sistema como clases, herencia, tipos y funciones. Además de otras características que aportan potencia y flexibilidad adicional que colocan a

¹⁵Pertenece al grupo de licencias de software libre, esta licencia consiste en que se puede ver el código, redistribuirlo y modificarlo; permite el uso del código fuente en software no libre.

PostgreSQL como la Base de Datos de código abierto más avanzada del mundo.

Debido a que PostgreSQL es de código abierto a diferencia de un software comercial, este puede ser extendido por el usuario en múltiples formas para adaptarlo a la aplicación en particular; por ejemplo, agregando nuevos tipos de datos, funciones, operadores, métodos de indexación, funciones de agregación y lenguajes procedurales. Además, debido a la licencia libre, PostgreSQL puede ser usado, modificado y distribuido libre de cargos para cualquier propósito, sea privado, comercial o académico.

- Principales características de PostgreSQL desde la versión 7.1.x:
 - Seguridad
 - Escalabilidad
 - Confiabilidad
 - Bajo costo total de operación
 - Compatibilidad con estándares (ANSI SQL92 y SQL99), incluye características avanzadas (Joins).
 - Transacciones (Transactions).
 - Disparadores (Triggers).
 - Restricciones (Constraints).
 - Replicación (Replication).
 - Respaldo y Recuperación (Backup & Recovery).
 - Reglas (Rules).
 - Procedimientos Almacenados/Funciones (Stored Procedures / Functions).
 - Logging. (Gestión de usuarios)
 - Extensivo y programable Soporta operadores, funciones, métodos y tipos de datos definidos por el usuario.
 - Orientado a Objetos. DBMS Objeto-Relacional, PostgreSQL aproxima los datos a un modelo objeto-relacional.

- Integridad Referencial.
- Lenguajes Procedurales. PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL.
- Cliente/servidor, entre otros.
- A partir de la versión 8, PostgreSQL se ejecuta en forma nativa en SO Windows sin requerir de una capa de emulación de ambiente Unix.

➤ Ventajas de PostgreSQL

- Por su arquitectura de diseño, escala muy bien al aumentar el número de CPUs y la cantidad de RAM.
- Soporta transacciones y claves foráneas (con comprobaciones de integridad referencial).
- Tiene mejor soporte para triggers y procedimientos en el servidor.
- Cuenta con unas interfaces gráficas, entre ellas PgAdmin para la administración de Bases de Datos.
- Tiene la capacidad de comparar la integridad referencial, así como también almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de BD de alto nivel como Oracle.

4.5.2 POSTGIS

PostGIS es una extensión de la base de datos relacional orientada a objetos PostgreSQL, que permite el almacenamiento y gestión de objetos geográficos. PostGIS más PostgreSQL forman una base de datos espacial donde se almacena y gestiona, tanto la geometría de los elementos geográficos, como los atributos temáticos de los mismos.



PostGIS ha sido desarrollado por Refractions Research Inc como un proyecto de investigación en tecnologías de bases de datos en código abierto. El licenciamiento del PostGIS es definido por la GNU GPL (General Public License), garantizando todas las libertades de un software libre.

Nativamente, PostgreSQL ya soporta geometrías, sin embargo PostGIS sigue la norma de almacenamiento de datos Geográficos OpenGis¹⁶ referente a bases de datos SQL conocida como Simple Features Specifications for SQL del consorcio internacional Open GeoSpatial (OGC), definiendo nuevos tipos de datos, creando dos tablas con información relevante al sistema (proyección de los datos y columna que posee la información geográfica) y definiendo también las funciones de manejo de información como procedimientos almacenados, garantizando a PostGIS interoperabilidad con incontables sistemas que también adoptan este estándar de almacenamiento. PostGIS cuenta con un gran número de funciones para análisis espaciales y/o topológicos que extienden el propio SQL del PostgreSQL.

De esta forma, PostGIS se encuentra al nivel de la base de datos SDE (Spatial Data Engine, motor de acceso a datos geográficos orientado a objetos) de ESRI o al de la versión de Oracle con su correspondiente extensión espacial (Oracle Spatial).

➤ Principales características:

- Almacenamiento y administración de datos geográficos que aparecen en las especificaciones OpenGIS, como por ejemplo puntos, líneas, polígonos, multilíneas, multipuntos y colecciones geométricas
- Implementación de diversas funcionalidades topológicas,

¹⁶ Ver anexo A

- Incluye funciones para el análisis básico de objetos SIG.
- Definición de nuevos tipos de datos para PostgreSQL
- Las geometrías pueden ser indexadas utilizando tanto la indexación nativa de PostgreSQL Gist (Generalized Search Trees) como la que implementa postgis R-Tree.
- Trae interfaces para servir de fuente de datos a MapServer, y a herramientas que utilizan OGR ¹⁷
- Actualmente no existe un interfaz gráfico de usuario que proporcione un entorno amigable y fácil de utilizar; Por ejemplo, para crear un nuevo elemento geográfico se debe utilizar la correspondiente sentencia SQL introduciendo tanto la geometría como los atributos. La solución estaría en utilizar otros paquetes software libres que proporcionen soporte para datos PostGIS, por ejemplo:
 - ✓ GRASS soporta actualmente PostGIS como fuente de datos a través del driver PostGRESS.
 - ✓ El visor JUMP tiene un plug-in para leer datos PostGIS.
 - ✓ El software QGIS presenta un buen soporte para PostGIS.

¹⁷ Optimal Golomb Ruler (OGR) es una librería que permite abrir y trabajar con ficheros vectoriales en varios formatos GIS. Integra geoprocesamientos, reproyección de datos y trabajo con los atributos

5. ANÁLISIS DE REQUERIMIENTOS

La fase de Análisis de Requerimientos es la primera etapa en el desarrollo de un proyecto o un Sistema de Información.

En esta etapa se formalizan las necesidades del cliente en torno a una situación específica (caso de estudio). Para esto; se identifican los actores¹⁸ del sistema, se establecen los requerimientos funcionales¹⁹, los no funcionales²⁰, así como las restricciones ó limitaciones externas que puede tener el sistema en relación con el caso de estudio definido.

Tomando la definición de requerimiento de la IEEE: “Condición o capacidad que un usuario requiere para suplir una necesidad, resolver un problema o lograr un objetivo”; y con el fin de levantar los requerimientos para el presente proyecto, se siguen los pasos que se enumeran a continuación:

- Obtener información acerca de las necesidades de los usuarios
- Planear el proyecto y los recursos que se usarán en él.
- Clasificar esas necesidades para comenzar a estructurar requerimientos
- Identificar los actores del sistema

¹⁸ Un actor es el rol o función que asume una persona, sistema o entidad que interactúa con el sistema que se desarrolla. Tiene la propiedad de ser externo a este. Se debe tener en cuenta que un usuario puede acceder al sistema como distintos actores o distintos roles.

¹⁹ Este tipo de requerimiento especifica algo que el sistema entregado debe ser capaz de realizar.

²⁰ Especifica algo propio del sistema, y cómo debe realizar sus funciones (rendimiento, calidad, etc.).

- Identificar los niveles de jerarquía del sistema y empezar a alojar los requerimientos ya clasificados en cada nivel.
- Especificar formalmente los requerimientos de acuerdo al nivel de audiencia que se desea.

Teniendo en cuenta que los requerimientos definen el alcance del sistema, se analizaron e identificaron los actores que intervienen en el mismo, para de acuerdo a esto, clasificar y estructurar como requerimiento las necesidades expuestas por los usuarios. De acuerdo a lo anteriormente mencionado se identificaron los siguientes actores ó roles:

Público: Es cualquier persona que entra al sistema sin haber cerrado negocios con la inmobiliaria, este usuario no necesita ningún tipo de clave para ingresar, solo puede consultar información acerca de inmuebles que están en oferta y solicitar mayor información acerca de los predios que le hayan interesado.

Cliente: Este usuario es quien ya ha cerrado algún negocio con la inmobiliaria, tiene acceso a una mayor información acerca del inmueble involucrado en la negociación.

Asesor Comercial: Es el encargado de contactar los nuevos posibles clientes, que son los usuarios públicos que han solicitado mayor información acerca de un predio en particular.

Administrativo: Este tipo de usuario es quien tiene acceso a la parte administrativa de la inmobiliaria.

Administrador del Sistema: Encargado del mantenimiento del sistema, debe mantener la información actualizada de los usuarios, de los inmuebles, entre otras.

Las necesidades²¹ del sistema se consultaron con varios expertos del área, y a partir de allí, y siguiendo los objetivos del presente proyecto, se establecieron los requerimientos del software a desarrollar.

Teniendo en cuenta que la aplicación que se propone en el proyecto es enfocada a la administración de información geográfica para la escogencia de los requerimientos, no se tendrán en cuenta muchas de las funcionalidades que son mas de gestión de la empresa que involucran a los usuarios definidos y en consecuencia no se tendrá en cuenta el usuario administrativo y otras funcionalidades de los diferentes perfiles. Por ende, los requerimientos para el desarrollo del prototipo serian los siguientes:

5.1 USUARIO PÚBLICO

- Buscar inmuebles ofrecidos en venta o arriendo por la inmobiliaria, que satisfagan condiciones de su escogencia; esta búsqueda se podría hacer teniendo en cuenta características como: Barrio, Rango de valor del inmueble, Rango de área del inmueble, Tipo de inmueble (Vivienda, Bodega, o Negocio específico), y seleccionar de la lista de inmuebles que dio como resultado la búsqueda.
- Visualizar información espacial de un inmueble seleccionado, localizándolo dentro del mapa de la ciudad.
- Visualizar información alfanumérica del inmueble seleccionado, como valor de venta o del canon de arriendo, disponibilidad o condiciones, áreas, servicios, estado, entre otros.
- Consultar información sobre existencia, distancia y otras características de utilidades cercanas al inmueble seleccionado, como iglesias, bancos,

²¹ Ver Anexo E.

escuelas y hospitales.

- Enviar solicitud de estudio y/o reserva para tomar el inmueble, ya sea para arrendamiento o venta.

Los roles Cliente, Asesor Comercial y Administrador del Sistema adicionalmente tienen acceso a funcionalidades anteriores del usuario público.

5.2 USUARIO CLIENTE

- Puede realizar las mismas funciones del usuario público.
- Registrarse. El usuario una vez es aprobado como cliente de la inmobiliaria, para ofrecer uno o varios inmuebles en arriendo o venta, puede identificarse y entrar al sistema para poder tener información adicional acerca del inmueble o enviar inquietudes.
- Modificar información del perfil personal, dirección, teléfonos, email.

5.3 USUARIO ASESOR COMERCIAL

- Puede realizar las mismas funciones del usuario público.
- Consultar información general de Clientes.
- Acceder al formato digital de inscripción de nuevos clientes y características del inmueble para venta o arriendo.
- Enviar información al administrador de formatos de nuevos inmuebles y clientes ya diligenciados.
- Modificar información del perfil personal, dirección, teléfonos, email.

5.4 USUARIO ADMINISTRADOR DEL SISTEMA

- Puede realizar las mismas funciones del usuario público.

- Consultar, ingresar, modificar o eliminar información sobre los usuarios registrados en el sistema.
- Asignar los respectivos nombres de usuarios y contraseña a los diferentes usuarios del sistema.
- Visualizar y habilitar información enviada por el asesor comercial de los formatos de nuevos inmuebles y clientes para que pueda ser vista por los demás usuarios del sistema según su perfil.
- Visualizar, ingresar, modificar o eliminar información alfanumérica y espacial de un inmueble.
- Realizar el mantenimiento de la información general que se maneja en la inmobiliaria.

6. DISEÑO DE LA HERRAMIENTA

Es la segunda etapa en la construcción de un proyecto, aquí tiene lugar su definición estructural y esto es el soporte y apoyo en la fase de codificación del sistema. Para diseñar el Sistema se empleó el Lenguaje Unificado de Modelado UML, enfocado en los diagramas de casos de uso.

Partiendo de los objetivos trazados y de los resultados obtenidos en la etapa de análisis de requerimientos se plantean los siguientes módulos:

- Módulo Administrador
- Módulo Asesor Comercial
- Módulo Cliente
- Módulo Usuario Público

Cada uno de los módulos está asociado a un tipo de usuario del prototipo del sistema propuesto: *Usuario Público, Administrador, Asesor Comercial, Cliente*; las acciones que cada uno de ellos puede realizar dentro del prototipo, dependen de sus propios intereses, es decir, según la información que cada usuario maneja; a continuación se presentan los diagramas de casos de uso teniendo en cuenta los requerimientos estipulados a desarrollar en el proyecto.

6.1 ACTORES

En los casos de Uso Críticos del proyecto se encuentran cuatro actores como se puede observar en la **Figura 7**, el actor Administrador quien es el encargado de hacer mantenimiento continuo de la información que se maneja, el actor Asesor Comercial que se encarga de revisar la información de los clientes, revisar

solicitudes y enviar información de nuevos clientes e inmuebles al administrador, el Cliente que puede dejar información de un inmueble o consultar sobre su inmueble y por último el Usuario Público que visualiza la información básica en pantalla.

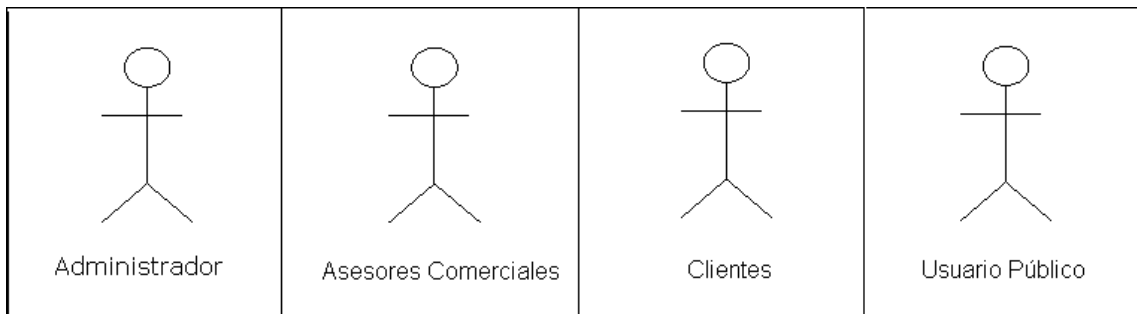


Figura 8 Actores.

6.2 CASOS DE USO

Los diagramas de Casos de Uso muestran las distintas operaciones que se espera de una aplicación y de cómo se relaciona con su entorno. A continuación se describen los casos de uso relacionados con cada uno de los actores presentes en el prototipo.

Casos de Uso del Actor Administrador: Los casos de uso que hacen referencia a las acciones que realiza el Actor Administrador cuando interactúa con el sistema se muestran en la **figura 8**. Ahí se definen entre otras, las tareas que implican la creación y mantenimiento de inmuebles y usuarios.

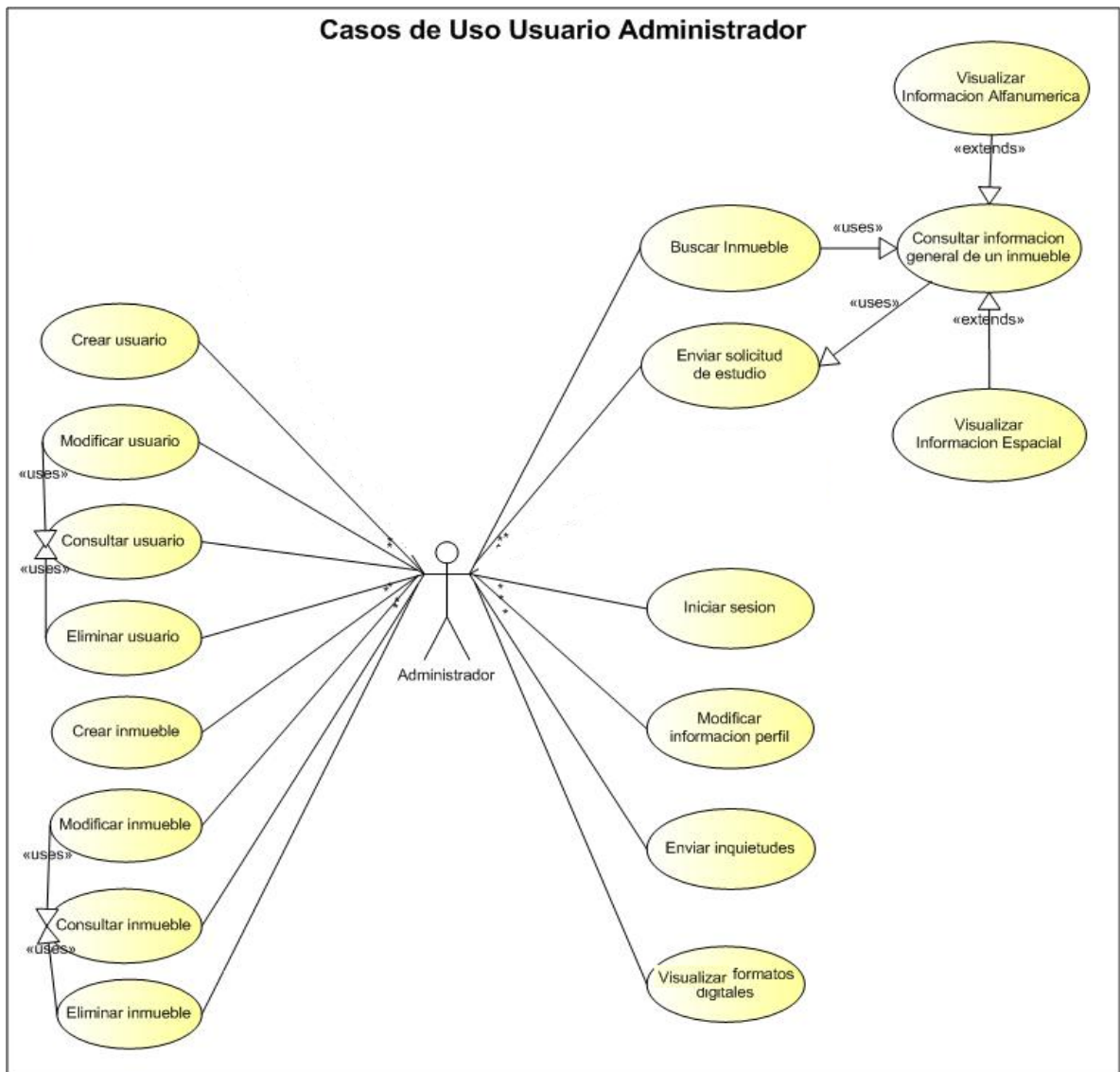


Figura 9 Casos de Uso del Actor Administrador

El usuario administrador tiene entre sus casos de uso, crear, modificar, eliminar y consultar la información de los usuarios, así mismo la de los inmuebles, modificar y visualizar la información espacial y hacer diferentes tipos de búsquedas.

Casos de Uso del Actor Asesor Comercial: Los casos de uso que hacen referencia a las acciones que realiza el Actor Asesor Comercial cuando este interactúa con el prototipo del sistema se muestran en la figura 9.

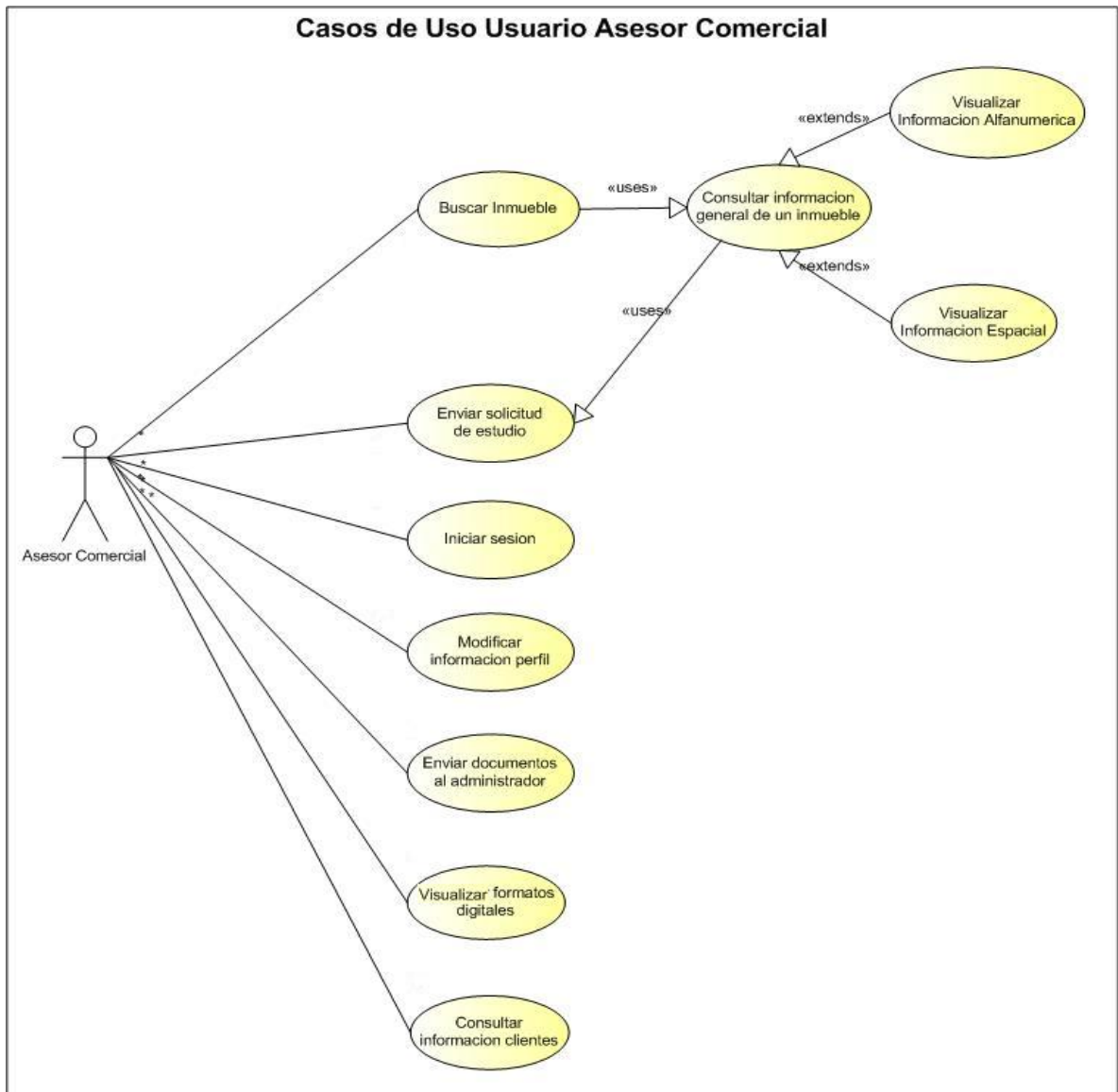


Figura 10 Casos de Uso del Actor Asesor Comercial.

El usuario Asesor Comercial tiene entre sus casos de uso, enviar documentos al administrador, estos documentos son el formulario de inscripción de nuevos inmuebles y clientes. Otro caso de uso es el de consultar información de clientes, donde además de la información general puede observar los inmuebles de los cuales es dueño el respectivo cliente.

Casos de Uso del Actor Cliente: Los casos de uso que hacen referencia a las acciones que realiza el Actor Cliente cuando éste interactúa con el prototipo del sistema se muestran en la figura 10.

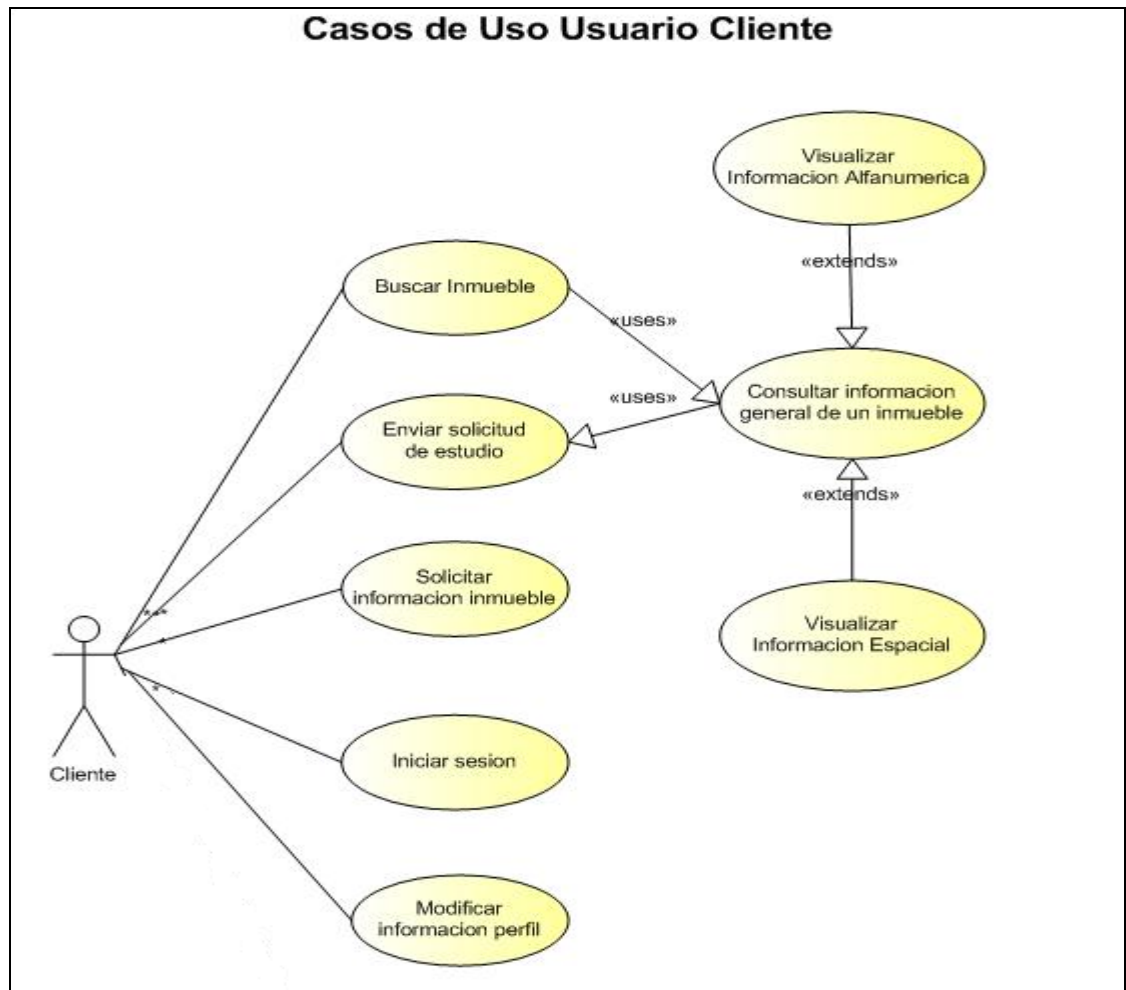


Figura 11 Casos de Uso del Actor Cliente.

El usuario Cliente tiene entre sus casos de uso, buscar inmueble, en esta búsqueda el cliente puede ver información adicional de los inmuebles de los cuales es dueño.

Casos de Uso del Actor Usuario Público: Los casos de uso que definen las operaciones que va a realizar el Actor Usuario Público cuando ingrese a la página

Web se muestran en la **figura 11**.

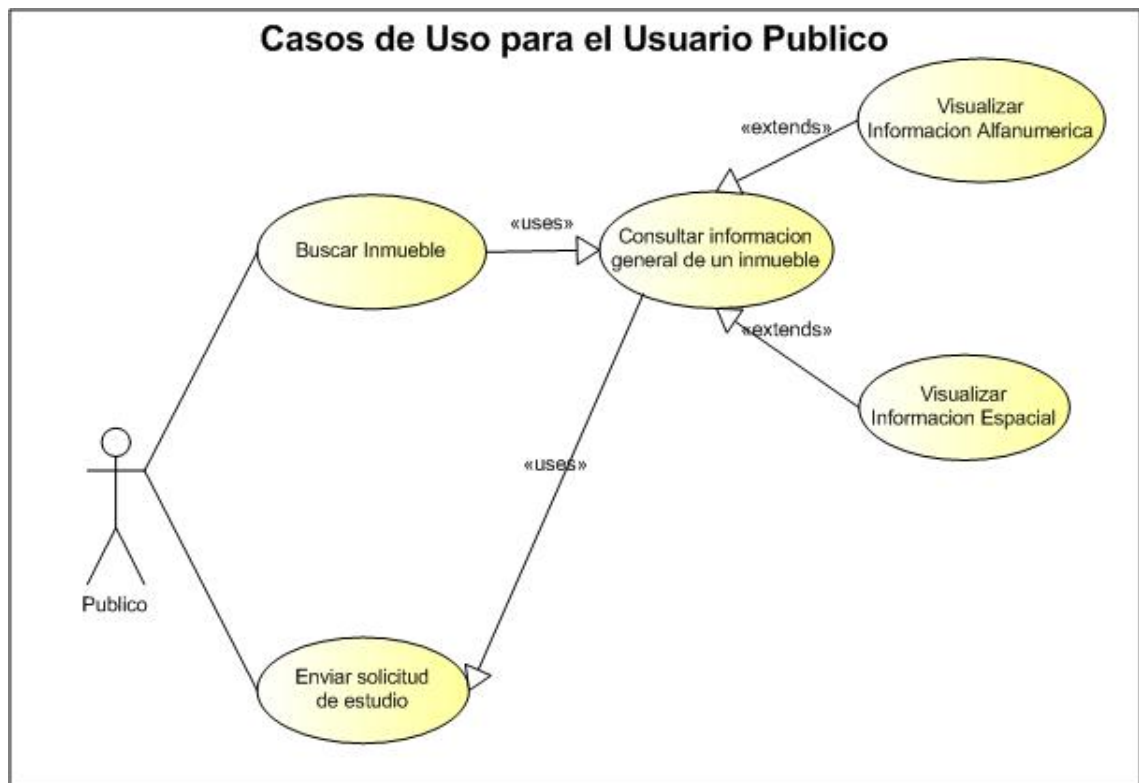


Figura 12 Casos de Uso del Actor Usuario Público.

Los casos de uso del Usuario Público son en su mayoría de visualización de información alfanumérica y espacial de inmuebles, teniendo la posibilidad de diferentes opciones de búsqueda según los criterios suministrados al sistema.

7. IMPLEMENTACIÓN

Ésta es la etapa donde todas las operaciones y funciones que debe realizar el prototipo se expresan como código. Ésta, se inicia con los resultados obtenidos en la etapa de diseño y su propósito general es definir las pautas para el desarrollo de la arquitectura y el sistema. La metodología de Prototipado Incremental trae consigo de forma explícita, una etapa de implementación para el proyecto desarrollado consistente en la codificación y depuración del prototipo siguiendo los resultados obtenidos en la etapa de diseño.

7.1 MODELO DE DATOS DEL PROTOTIPO

El desarrollo del prototipo implicó la construcción de un modelo de datos asociado al manejo de la aplicación donde se almacenan los datos que controlan su funcionamiento. La aplicación cuenta con una base de datos “inmobiliaria”, la cual se encarga de la gestión de los usuarios, almacenamiento de la información espacial y alfanumérica de las diferentes mapas que se manejan y que son básicos para poder implementar los diferentes análisis espaciales que se proponen en el prototipo y son principales protagonistas a la hora de acceder al sitio web. La descripción del modelo de datos junto con la imagen de la base de datos se encuentra en el Anexo E.

7.2 INTERFAZ GRÁFICA

Es la encargada de la comunicación entre el usuario y el computador, es esta característica donde se ve reflejada su importancia, ya que es la que se encarga

de transmitir las opciones que la herramienta pone a disposición de los usuarios; si dichas opciones no se muestran en forma clara, los usuarios no comprenderán los mensajes que se presentan, por lo tanto las funcionalidades que ésta ofrece no se aprovechan completamente. Es por esto que los menús, textos, gráficos, balance de colores, son trascendentales al desarrollar un proyecto como el que se presenta en este documento y contribuyen a mejorar la interacción usuario-máquina.

El prototipo está dividido en cuatro Módulos que son: El Módulo Usuario Público, el Módulo Asesor Comercial, el Módulo Cliente y el Módulo Administrador del Sistema. Para cada uno de ellos se diseñó la interfaz gráfica de acuerdo a las necesidades expresadas en los requerimientos planteados inicialmente en el capítulo 5 *Análisis de Requerimientos*, y después formalizados por medio de los diagramas de casos de uso vistos en el numeral 6.2 *Casos de Uso*.

Se diseñó una interfaz inicial a la cual tienen acceso todos los usuarios, es la página principal del prototipo. En la Figura 13 se observa:

- En el número 1 se observa el Marco Superior, el cual contiene una imagen panorámica de Bucaramanga que se visualiza en todo el prototipo Web y un menú de acceso rápido a las diferentes opciones que se presentan dependiendo del tipo de usuario que se presente.
- En el número 2 se observan las diferentes capas de mapas que puede cargar el usuario y que desee visualizar en el mapa.
- En el numeral 3 están las diferentes opciones de búsqueda de inmuebles incluidos por precio, área y/o barrio.
- En el número 4 se muestra el mapa principal de la ciudad, mostrando solamente las capas seleccionadas inicialmente (capa de barrios, comunas e inmuebles) y con los valores iniciales que se le asignan al mapa. En la parte

inferior se mostrarán los resultados que arroje las consultas realizadas en el menú del numeral 3 y las que se realicen interactuando con el mapa.

- En el numero 5 se muestran opciones para interactuar con el mapa directamente, cambiar las dimensiones del mapa, acercar, alejar, opciones de búsqueda directamente en el mapa, entre otras; se muestra además un mapa de referencia del mapa principal que permite localizar fácilmente el área visualizada.

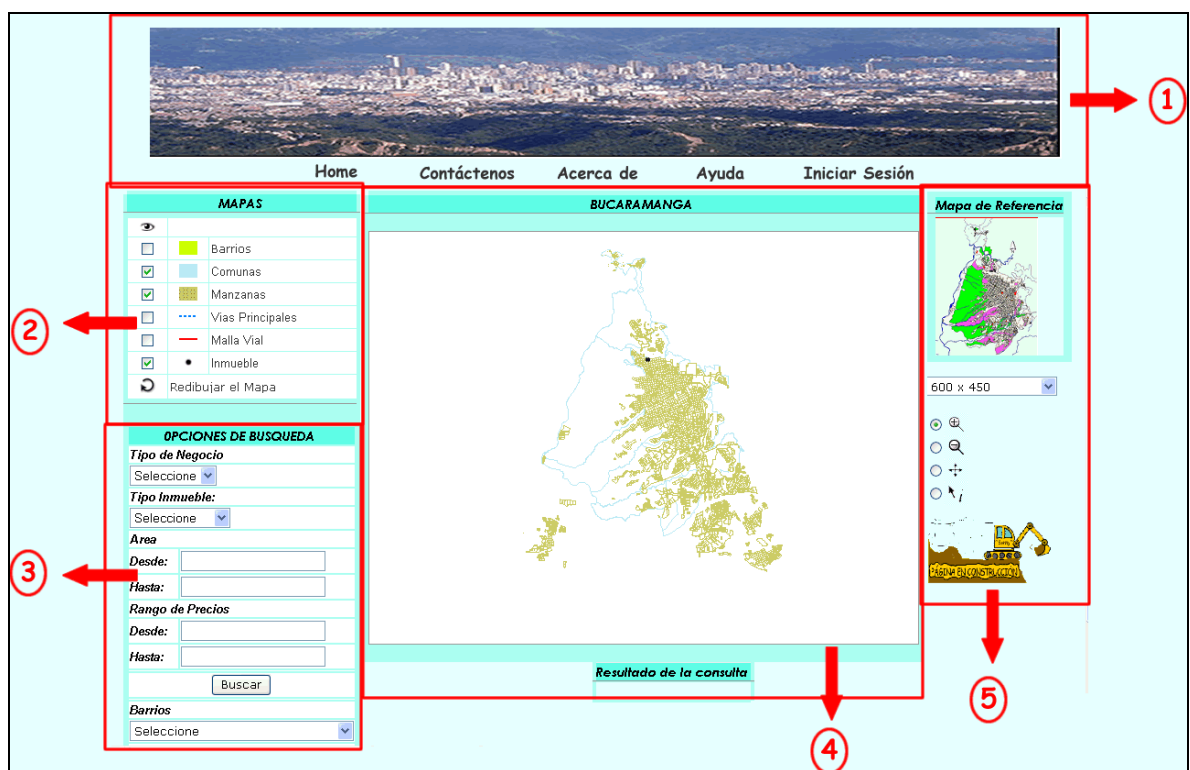


Figura 13 Interfaz Principal del Software

7.2.1 Usuario Público

Al módulo de Usuario Publico puede acceder cualquier tipo de usuario, fue diseñado para que cualquier persona que desee buscar un inmueble, pueda hacer la búsqueda por medio de una página Web de la forma más sencilla posible.

Luego de visualizar la página principal, tiene la opción de escoger entre las diferentes opciones que se encuentran en el menú de acceso rápido del marco superior. Para el caso del usuario público las opciones de menú son: *Home*²², *Contáctenos*, *Acerca de*, *Ayuda*, e *Inicio de Sesión*. La opción *Home*, es la que regresa siempre a la página de inicio; La opción *Ayuda* orienta al usuario para interactuar con el prototipo.

Mediante la interfaz de la opción *Contáctenos*, el usuario puede enviar información al administrador del sistema, solicitando o enviando información.



The image shows a contact form with the following fields and elements:

- *Nombre: Text input field
- *Apellido: Text input field
- *E-mail: Text input field
- *Motivo del mensaje: Text input field
- *Cuerpo del Mensaje: Text area with a scrollbar
- Caracteres disponibles: 800 (displayed in a small box)
- *Campos Obligatorios: Label indicating required fields
- Borrar: Button
- Enviar: Button

Figura 14 Interfaz de Contáctenos

También se presenta la opción de *Acerca de*, en donde se muestra la información general del proyecto, autores y entidades relacionadas en el desarrollo del proyecto y en dónde se realizó.

²² HOME: Hace referencia a la página principal o inicial, se usó en inglés para facilitar el aprendizaje, pues muchas interfaces lo usan así.

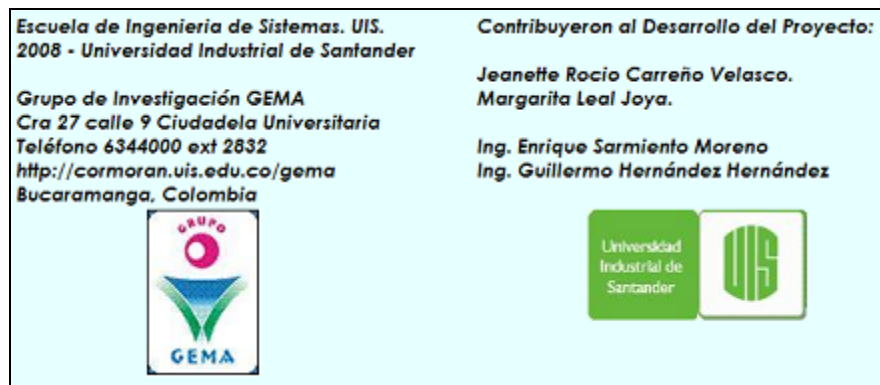


Figura 15 Interfaz de Acerca de.

En la siguiente ventana se observan los cuadros de texto para que el usuario registrado pueda ingresar su nombre de usuario y contraseña para tener acceso a las otras utilidades que presta el prototipo, bien sea como Administrador, Asesor Comercial o Cliente; a esta interfaz se ingresa con la opción de menú “iniciar sesión”.



Figura 16 Interfaz de Inicio de Sesión.

7.2.2 Usuario Administrador

Luego de que ha iniciado sesión un usuario que tenga perfil de Administrador, visualiza la página principal que se muestra en la Figura 17, con la diferencia que

el marco superior el menú cambia de opciones, las cuales describiremos a continuación:



Figura 17 Menú principal del Administrador

En el enlace de *inmueble* se dirige hacia una nueva pantalla donde nos lista los inmuebles que se encuentran disponibles, con información como dirección, barrio, el nombre del propietario y el estado del inmueble. También muestra las opciones de visualizar, modificar y eliminar dicho registro del inmueble. Se encuentran también otras opciones como Buscar, Habilitar y Crear.

Administración de Inmuebles					
Dirección del Inmueble	Barrio	Propietario	Estado		
calle 1 # 2 - 55	Comuneros	propietario uno	Arrendar		
calle 1 # 3 - 45	Comuneros	propietario uno	Arrendar		
<i>Número de Páginas : 1</i>					
<input type="button" value="Crear"/>		<input type="button" value="Habilitar"/>		<input type="button" value="Buscar"/>	

Figura 18 Manejador de Inmuebles

La Figura 19 nos muestra la interfaz de la información detallada del inmueble a la que se llega con el enlace *Ver Detalle* y da la opción de hacer consultas espaciales.

Información General	
Tipo de Inmueble:	apartamento
Dirección:	calle 1 # 2 - 55
Barrio:	Comuneros
Tipo de Negocio:	Arrendar
Valor:	500000
Información Detallada	
Area:	456.00
No. de Habitaciones:	5
No. de Baños :	2
Administración:	Si
Garaje:	No
No. de Closet :	4
Estrato:	2
Propietario:	propietario uno
Asesor Comercial:	vendedor1
Observaciones:	casa para arrendar
Realizar Consultas Especiales	










Figura 19 Visualizar datos de Inmuebles

En la opción de hacer consultas especiales, Se puede visualizar información adicional del inmueble acerca del entorno donde se halla ubicado, ya sea entidades culturales, bancos, iglesias, entidades de salud, entidades educativas, que tiene en un radio determinado. Se puede observar otra información como barrios que lo rodean, vías principales que atraviesan el barrio, y mínima distancia en metros que se encuentra ubicado el inmueble de una vía principal.

Información General	
Dirección :	calle 1 # 2 - 55
Comuna:	Comuna San Francisco
Barrio:	Comuneros
Estrato:	2
Información del Entorno del Inmueble	
En un radio de:	300 mt <input type="button" value="Buscar"/>
Bancos:	
Iglesias:	Asilo San Rafael , Convento Cristo Rey , Iglesia Adventista del S.ptimo D'ja , Iglesia Apostófica de Jesucristo ,
Ent. de Salud:	Centro de Salud Comuneros ,
Ent. Educativas:	Concent. Balbino Garcia , Concentracion 16 De Marzo , Concentracion Bucaramanga , Concentracion Comuneros ,
Ent. Culturales:	
Otra Información	
Barrios que lo rodean:	Uis , La Independencia , Chapinero , Mutualidad , Modelo , Universidad ,
Vías que atraviesan el barrio:	**IM** , Carrera 21 , Carrera 22 , Carrera-Diagonal 15 ,
minima dist a via principal:	4123 mt
<input type="button" value="Regresar"/>	



Figura 20 Consultas Espaciales

La opción de *Eliminar*, muestra la información no tan general del inmueble, para estar seguro de que esa es la información que se quiere visualizar.

Información General
Tipo de Inmueble: apartamento
Dirección: calle 1 # 2 - 55
Barrio: Comuneros
Tipo de Negocio: Arrendar
Valor: 500000
Información Detallada
Area: 456.00
No. de Habitaciones: 5
No. de Baños : 2
Administración: Si
Garaje: No
No. de Closet : 3
Estrato: 2
Propietario: propietario uno
Asesor Comercial: vendedor1
Observaciones: casa para arrendar
<input type="button" value="Cancelar"/> <input type="button" value="Eliminar"/>




Figura 21 Interfaz de Eliminar Inmueble

La opción de *modificar* muestra la información como dirección, barrio y estrato del inmueble, y permite arreglar datos como el número de baños, habitaciones, el tipo de inmueble, entre otras.

MODIFICAR INMUEBLE		
Datos Generales		
Dirección:	calle 1 # 2 - 55	
Barrio:	COMUNEROS	
Estrato:	2	
DATOS A MODIFICAR		
	Datos Actuales	Nuevos Datos
Dueño:	propietario uno	propietario uno
Asesor Comercial:	vendedor1	vendedor1
Tipo de Inmueble:	apartamento	casa
Estado del Inmueble:	Arrendar	arrendado
Valor:	500000	500000
Area:	456.00	456.00
No. de Habitaciones:	5	5
No. de Baños :	2	Seleccione
Administración:	Si	Seleccione
Garaje:	No	Seleccione
No. de Closet:	3	Seleccione
Foto:	Actual	<input type="text"/> Examinar
Observaciones:	casa para arrendar	
<input type="button" value="Cancelar"/> <input type="button" value="Modificar"/>		

Figura 22 Modificar Inmuebles

También el Administrador cuenta con las función de *Crear* inmuebles, luego que se digita la dirección (Figura 23), se abre una nueva ventana de Registro de Inmuebles (Figura 24), donde se ingresa características básicas del predio (# de habitaciones, precio, área, etc.), de igual forma da la opción de subir fotografías del inmueble y escribir observaciones generales.

Administración de Inmuebles		
Dirección del Inmueble:	calle 1	# calle 1 - <input type="text"/>
<input type="button" value="Validar Dirección"/>		

Figura 23 Crear Dirección del Inmueble

Figura 23 Crear Inmueble

El administrador puede hacer diferentes tipos de búsqueda de inmuebles, ya sea por barrio, propietario, tipo de inmueble, estado del inmueble y dirección, y de esa forma acceder de forma más rápida a la información.

Figura 24 Buscar Inmueble

En la opción *Habilitar* Se visualiza los inmuebles registrados por los Asesores

Comerciales y que falta ser habilitados por el Administrador. Para que puedan ser visualizados en el mapa principal por cualquiera de los usuarios. (Figura 26)





Administración de Inmuebles - Habilitar					
Dirección del Inmueble	Barrio	Propietario	Estado	Asesor Comercial	
calle 1 # 3 - 56	Comuneros	Arrendar	propietario uno	vendedor1	 
calle 1 # 6 - 34	Comuneros	Vender	propietario uno	vendedor1	 
Número de Páginas : 1					
Habilitar Todos los Inmuebles					

Figura 25 Habilitar Inmuebles

En el enlace de *Usuarios*, se visualiza la lista de personas que están registradas en la base de datos; allí se puede visualizar el número de cédula, nombre y el perfil; también cuenta con las opciones de visualizar, modificar y eliminar cada registro de usuario. Se encuentran también otras opciones como Buscar, Habilitar y Crear.
















Administración de Usuarios					
Cédula	Nombre	Perfil			
4444	Propietario Uno	Propietario			
44441	Propietario Dos	Propietario			
1111	Admon	Administrador			
2222	Vendedor1	Asesor Comercial			
45454654	Thghhh	Propietario			
Número de Páginas : 1					
Crear Habilitar Buscar					

Figura 26 Manejador de Usuarios

En la siguiente figura se puede ver los datos detallados del usuario, tales como el número de cédula, los nombres, nombre de usuario, perfil y la dirección, entre otros. A esta interfaz se accede con el enlace de *Ver Detalle*.

Administración de Usuarios - Detalle	
Cédula:	4444
Nombre y Apellidos:	propietario uno
Nombre de Usuario :	4444
Perfil:	Propietario
Teléfono:	1234
Dirección correspondencia:	calle 4 con 1
Correo Electrónico:	propietario1@hotmail.com
Contraseña:	4444
Observaciones:	propietario No.1
<input type="button" value="Aceptar"/>	

Figura 27 Visualizar Usuarios

En la opción de *modificar* los datos del usuario, el administrador tiene la opción de corregir el nombre, cambiar el perfil o arreglar otros datos si quedaron mal escritos o enviaron información errónea.

Administración de Usuarios - Modificar	
Perfil Actual:	Propietario
Cédula Usuario:	4444
Nombre de Usuario (*):	4444
Nombre y Apellidos (*):	<input type="text" value="propietario uno"/>
Nuevo Perfil (*):	<input type="text" value="Seleccione"/>
Teléfono (*):	<input type="text" value="1234"/>
Dirección correspondencia (*):	<input type="text" value="calle 4 con 1"/>
Correo Electrónico:	<input type="text" value="propietario1@hotmail.c"/>
Contraseña: (*)	<input type="text" value="4444"/>
Observaciones:	<input type="text" value="propietario No.1"/>
<input type="button" value="Cancelar"/> <input type="button" value="Modificar"/>	

Figura 28 Modificar Usuarios

En la opción *Habilitar* se visualiza los usuarios registrados por los Asesores Comerciales y que están pendientes de asignarles nombre de usuario y contraseña para que puedan acceder al prototipo como usuarios clientes.



Administración de Usuarios - Habilitar			
Cedula	Nombre	Perfil	
63458123	Eddy Fajardo Ortiz	Propietario	
37895632	Viviana Andrea Parada	Propietario	
Número de Páginas : 1			

Figura 88 Habilitar Usuarios

En la opción *crear* El prototipo permite al administrador crear un nuevo usuario, ingresando sus datos personales el nombre de usuario, la contraseña y el perfil o tipo de usuario.

Administración de Usuarios - Crear	
Cédula Usuario(*):	<input type="text"/>
Nombre y Apellidos(*):	<input type="text"/>
Perfil de Usuario(*)	Seleccione <input type="button" value="v"/>
Teléfono(*):	<input type="text"/>
Dirección correspondencia(*):	<input type="text"/>
Correo Electrónico:	@ <input type="text"/>
Nombre de Usuario (*):	<input type="text"/>
Contraseña: (*)	<input type="text"/>
Observaciones:	<input type="text"/>
<input type="button" value="Cancelar"/> <input type="button" value="Crear"/>	

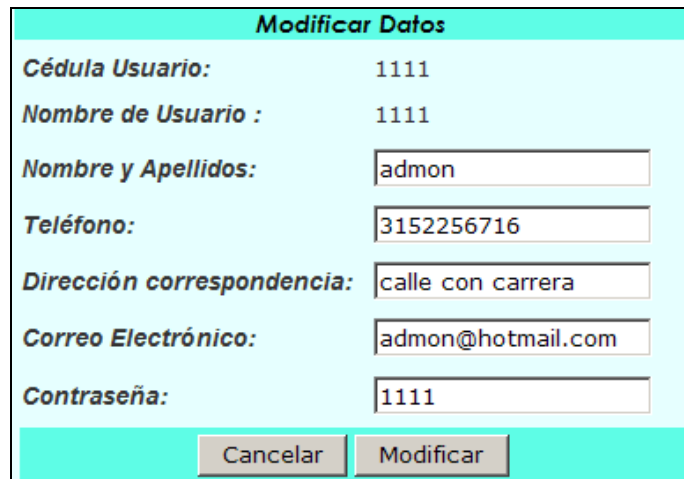
Figura 30 Crear Usuarios

Se pueden realizar búsquedas rápidas de los usuarios, bien sea por cédula, nombre o perfil y así visualizar la información que necesita.

Administración de usuarios - Búsqueda Avanzada	
<input type="radio"/> Cédula:	<input type="text"/>
<input type="radio"/> Nombre y Apellidos:	<input type="text"/>
<input type="radio"/> Perfil	Seleccione <input type="button" value="v"/>
<input type="button" value="Aceptar"/>	

Figura 31 Buscar Usuarios

Tanto el Administrador, como el Asesor Comercial y el Cliente, tienen la opción de modificar sus datos, para tener mayor control sobre su propia información y tener su clave, para una mejor seguridad. Además el Administrador puede modificar/eliminar los datos de todos, por si quien los creó ya no está disponible.



The image shows a web form titled "Modificar Datos" with a light blue background. It contains several input fields for user information:

Cédula Usuario:	1111
Nombre de Usuario :	1111
Nombre y Apellidos:	<input type="text" value="admon"/>
Teléfono:	<input type="text" value="3152256716"/>
Dirección correspondencia:	<input type="text" value="calle con carrera"/>
Correo Electrónico:	<input type="text" value="admon@hotmail.com"/>
Contraseña:	<input type="text" value="1111"/>

At the bottom of the form are two buttons: "Cancelar" and "Modificar".

Figura 32 Modificar Datos

7.2.3 Usuario Asesor Comercial

El usuario Asesor Comercial, tiene el siguiente marco superior, con las diferentes acciones que él puede realizar.





Figura 33 Menú principal del Asesor Comercial

En el enlace de *Inmuebles*, él tiene la opción de registrar un nuevo inmueble digitando la dirección y luego llenando todos los datos de dicho inmueble, como se

mostró en las figuras 19 y 20.

En la opción de *usuarios*, se puede visualizar los usuarios clientes de la inmobiliaria y ver información general de dichos usuarios (datos personales), de igual forma se puede ver los inmuebles que tienen consignados a su nombre en la inmobiliaria, como se muestra en la figura 35, En esta interfaz también se encuentra la opción de registrar un nuevo usuario (figura 36).

En la opción del menú contáctenos, el puede enviar información vía correo electrónico al administrador, solicitando datos o informándole a él de algún cambio. La interfaz es igual a la de la figura 14.

Usuarios			
Cédula	Nombre	Perfil	
4444	Propietario Uno	Propietario	
44441	Propietario Dos	Propietario	

Número de Páginas : 1

Nuevo usuario

Figura 34 Visualizar Usuarios

FORMATO DE REGISTRO DE USUARIOS	
Cedula Usuario: (*)	<input type="text"/>
Nombre y Apellidos: (*)	<input type="text"/>
Tipo de Cliente: (*)	Seleccione 
Telefono: (*)	<input type="text"/>
Dirección correspondencia: (*)	<input type="text"/>
Correo Electronico:	@ <input type="text"/>
Comentarios:	<input type="text"/>
Cancelar Aceptar	

Figura 35 Registrar Usuarios

7.2.4 Usuario Cliente

Al ingresar un Cliente, el marco superior cambia, para mostrar las opciones a las cuales el tiene acceso, como se muestra en la figura:



Figura 36 Menú principal del Cliente

El cliente tiene la opción de listar los inmuebles que tiene registrados en la inmobiliaria escogiendo la opción del menú *Mis Inmuebles*, para consultar el estado en que se encuentra, teniendo la opción de ver información detallada de sus inmuebles para poder saber si es correcta la información que se muestra.



<i>Dirección del Inmueble</i>	<i>Barrio</i>	<i>Estado</i>	
calle 1 # 2 - 55	Comuneros	Arrendar	
calle 1 # 3 - 45	Comuneros	Arrendar	
<i>Número de Páginas : 1</i>			

Figura 37 Lista de Inmuebles del Cliente

La opción de Cerrar Sesión en los diferentes perfiles del prototipo, al terminar dicha sesión los envía a la página principal del prototipo y quedaría con el perfil de usuario público.

8. PRUEBAS

Las pruebas del prototipo se realizaron de forma Web y local. Para llevar a cabo las pruebas Web del prototipo se utilizó como servidor Web cormoran.uis.edu.co, que es el servidor de la Escuela de Ingeniería de Sistemas UIS. Se pudo entrar al prototipo Web con la dirección:

'www.sistemas.uis.edu.co/inmobiliaria/index.htm'. (Figura 1).

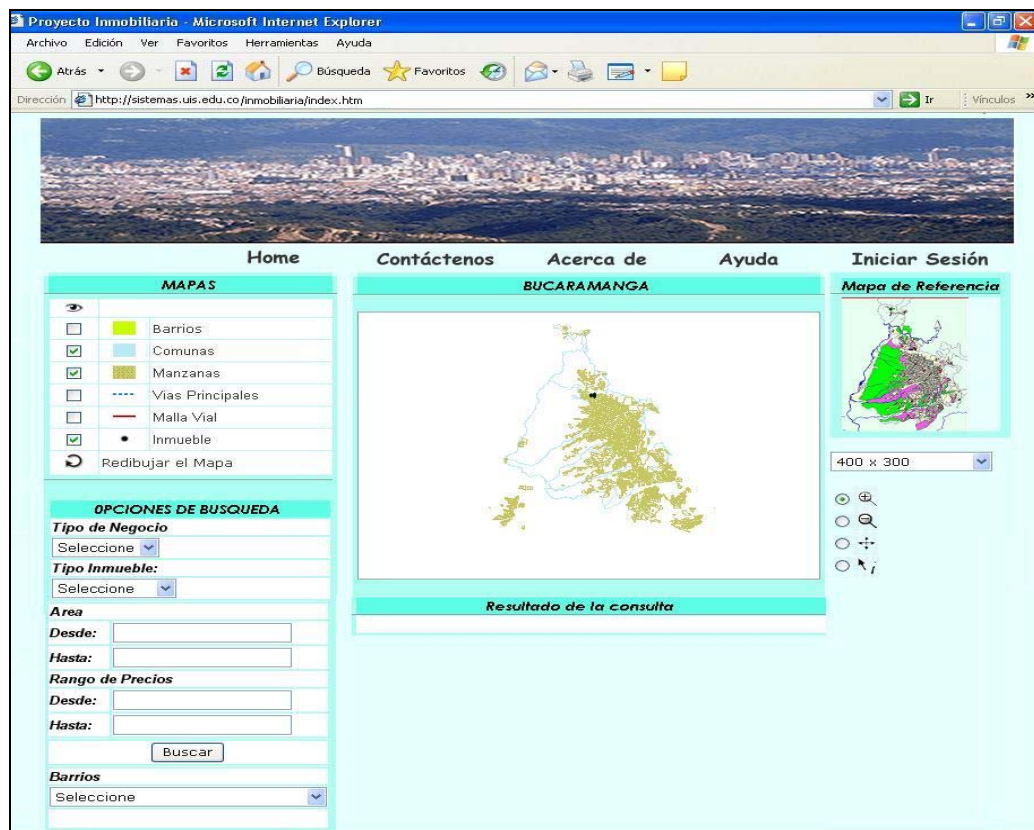


Figura 38 Imagen de prueba Web²³

²³ Prueba Web del prototipo ingresando a la página www.sistemas.uis.edu.co/inmobiliaria/index.htm mostrando la interfaz inicial del usuario publico

En el desarrollo de la prueba del prototipo en la WEB efectuada el día 16 de abril del 2008 participó personal vinculado al grupo de investigación GEMA desempeñando los diferentes tipos de usuarios del prototipo (Administrador, Asesor Comercial, Cliente y Público) previamente enteradas de las funcionalidades que debía cumplir cada uno de los roles propuestos en el prototipo y otras personas que accedieron al sitio como usuarios públicos.

Esta prueba se realizó para demostrar que el prototipo es Web, por lo tanto se puede acceder desde cualquier equipo de cómputo que tenga acceso a Internet. En esta prueba la información proveniente de la base de datos del prototipo Web fue recibida, almacenada y presentada a los usuarios que accedieron a él, en los diferentes tipos de usuarios.

Se probó que un usuario final, sin conocimientos en Sistemas de Información Geográfica, pudiera navegar en el prototipo y buscar un inmueble que cumpliera con sus expectativas. Se les asignó a personas, con conocimientos un poco más avanzados en el manejo de sistemas el perfil del administrador, para realizar dicho rol, y pudiesen ubicar un punto en el mapa, que podría llegar hacer la parte más complicada de realizar. En dichas prueba se evaluó el cumplimiento de los requisitos planteados para el prototipo y se hicieron ajustes al mismo.

Se realizaron pruebas de forma local durante el mes de marzo y abril del presente año con personal del Grupo Gema y otras personas, accediendo como los diferentes tipos de usuarios existentes en el prototipo.

Personal

Las principales personas que colaboraron con las pruebas del prototipo y respectivos observaciones al mismo fueron:

Nombre	Rol
Xiomara Santis Mancipe	Todos los roles
Frank Camacho	Todos los roles
Edwin Vargas	Todos los roles
Melba Susana Chaparro	Todos los roles
Andrea Duarte Daza	Todos los roles
Milena Celis	Todos los roles
Jhon A. Vargas	Todos los roles
Eddy Johana Fajardo	Usuario Público
Odry Johana Cubillos	Usuario Público
Viviana Andrea Parada	Usuario Público

Resultados de las pruebas

Las diferentes pruebas arrojaron como resultados las siguientes observaciones que se presentan organizadas por módulos, según las opciones de interacción con el prototipo de cada usuario; en la tabla 8 se muestran los resultados obtenidos para el módulo administrador, las observaciones del módulo del usuario cliente se muestran en la tabla 9, las del usuario asesor comercial en la tabla 10 y en la tabla 11 las del módulo usuario público.

CASO DE PRUEBA	RESULTADO
Opciones de búsqueda.	El prototipo permitió realizar búsquedas de inmuebles teniendo en cuenta características como: Tipo de inmueble, rango de área, rango de precios, tipo de negocio, y barrio.
Búsqueda en el mapa.	El prototipo permitió realizar búsquedas directamente en el mapa ya que se visualizan los inmuebles disponibles para arrendar o vender en forma de puntos.

Resultado de la Consulta	Se puede visualizar información de un inmueble dando clic en el enlace ' <i>ver más</i> ' de cada dirección de inmueble que da como resultado la consulta que se realiza ya sea por el menú de opciones de búsqueda o directamente en el mapa. La información del inmueble que se observa es más detallada que la de los otros usuarios, observando entre otros el dueño del inmueble, asesor comercial a cargo del negocio, observaciones, tipo de negocio, entre otros.
Realizar Consultas espaciales	Se puede visualizar información adicional del inmueble acerca del entorno donde se halla ubicado, ya sea entidades culturales, bancos, iglesias, entidades de salud, entidades educativas, que tiene en un radio determinado. Se puede observar otra información como barrios que lo rodean, vías principales que atraviesan el barrio, y mínima distancia en metros que se encuentra ubicado el inmueble de una vía principal.
Inmuebles	Se puede visualizar los inmuebles consignados en la inmobiliaria detallando la dirección, barrio, propietario y estado en el que se encuentra (arrendado, por arrendar, etc).
Crear (de la opción del menú Inmuebles)	El prototipo permite crear un nuevo inmueble, ingresando inicialmente una dirección, validando si la dirección digitada ya existe o si es correcta. Si es así se abre una nueva ventana de Registro de Inmuebles donde se ingresa características básicas del predio (# de habitaciones, precio, área, etc.), de igual forma da la opción de subir fotografías del inmueble y escribir observaciones generales. El nuevo inmueble creado, queda habilitado para ser mostrado en el mapa de la página principal de

	cualquier usuario si es para vender o arrendar.
Ver Detalle, Modificar y Eliminar (de la opción del menú Inmuebles)	El prototipo permite visualizar, modificar y eliminar inmuebles. Solo permite modificar información alfanumérica del inmueble, La información no modificable del inmueble es la dirección, barrio, comuna y estrato.
Habilitar (de la opción del menú Inmuebles)	Se visualiza los inmuebles registrados por los Asesores Comerciales y aun no habilitados por el administrador. Dando la opción de habilitarlos individualmente o todo el grupo. Al habilitar el inmueble ya se puede visualizar en el mapa de la página principal de cada usuario si es para arrendar o vender.
Buscar (de la opción del menú Inmuebles)	Permite realizar una búsqueda avanzada de los inmuebles ya sea por barrio, propietario, tipo de inmueble, estado del inmueble y dirección.
Usuarios	Se puede visualizar todos los usuarios de la inmobiliaria detallando la cédula, nombre y el perfil o tipo de usuario.
Ver Detalle, Modificar y Eliminar (de la opción del menú Usuarios)	El prototipo permite visualizar, modificar y eliminar usuarios. La información no modificable del usuario es la cédula y el nombre de usuario.
Crear (de la opción del menú Usuarios)	El prototipo permite crear un nuevo usuario, ingresando sus datos personales el nombre de usuario, la contraseña y el perfil o tipo de usuario.
Habilitar (de la opción del menú Usuarios)	Se visualiza los usuarios registrados por los Asesores Comerciales y aun no habilitados por el administrador. Es necesario habilitarlos individualmente ya que se necesita asignarles nombre de usuario y contraseña.
Buscar (de la opción del menú)	Permite realizar una búsqueda avanzada de los usuarios ya sea por cédula, nombre o perfil.

Usuarios)	
Home	Permite ir de forma rápida a la página principal dando clic en la opción 'Home'
Modificar Datos	Permite cambiar los datos personales del usuario y la contraseña, menos la cédula y el nombre de usuario.
Ayuda	En la opción 'ayuda' se puede encontrar claramente la información necesaria para realizar los procedimientos disponibles para el administrador.
Cerrar Sesión	El prototipo tiene la opción de cerrar la sesión iniciada, y regresar nuevamente a la página principal de usuario público.

Tabla 8 Resultados del Módulo Usuario Administrador

En modificar inmueble, los datos no modificables son dirección, barrio, comuna y estrato ya que estos datos son asignados automáticamente por el prototipo cuando se crea la geometría del punto que representa el inmueble.

CASO DE PRUEBA	RESULTADO
Opciones de búsqueda.	El prototipo permitió realizar búsquedas de inmuebles teniendo en cuenta características como: Tipo de inmueble, rango de área, rango de precios, tipo de negocio, y barrio.
Búsqueda en el mapa.	El prototipo permitió realizar búsquedas directamente en el mapa ya que se visualizan los inmuebles disponibles para arrendar o vender en forma de puntos.
Resultado de la Consulta	Se puede visualizar información de un inmueble dando clic en el enlace 'ver más' de cada dirección de inmueble que da como resultado la consulta que se realiza ya sea

	por el menú de opciones de búsqueda o directamente en el mapa.
Realizar Consultas espaciales	Se puede visualizar información adicional del inmueble acerca del entorno donde se halla ubicado, ya sea entidades culturales, bancos, iglesias, entidades de salud, entidades educativas, que tiene en un radio determinado. Se puede observar otra información como barrios que lo rodean, vías principales que atraviesan el barrio, y mínima distancia en metros que se encuentra ubicado el inmueble de una vía principal.
Mis Inmuebles	En la opción 'Mis Inmuebles' se puede observar la lista de inmuebles que tenga consignados a su nombre en la inmobiliaria con la opción de poder ver información adicional de sus inmueble como: asesor comercial a cargo, estado del inmueble (arrendado, por arrendar, por vender) observaciones varias.
Modificar Datos	Permite cambiar los datos personales del usuario y la contraseña.
Contáctenos	En la opción 'Contáctenos', permite enviar un correo electrónico con cualquier tipo de inquietud
Ayuda	En la opción 'Ayuda' se puede encontrar claramente la información necesaria para realizar los procedimientos disponibles para el usuario que ingresa al prototipo sin registrarse.
Cerrar Sesión	El prototipo tiene la opción de cerrar la sesión iniciada, y regresar nuevamente a la página principal de usuario público.
Home	Permite ir de forma rápida a la página principal dando clic en la opción 'Home'

Tabla 9 Resultados del Módulo Usuario Cliente.

CASO DE PRUEBA	RESULTADO
Opciones de búsqueda.	El prototipo permitió realizar búsquedas de inmuebles teniendo en cuenta características como: Tipo de inmueble, rango de área, rango de precios, tipo de negocio, y barrio.
Búsqueda en el mapa.	El prototipo permitió realizar búsquedas directamente en el mapa ya que se visualizan los inmuebles disponibles para arrendar o vender en forma de puntos.
Resultado de la Consulta	Se puede visualizar información de un inmueble dando clic en el enlace ' <i>ver más</i> ' de cada dirección de inmueble que da como resultado la consulta que se realiza ya sea por el menú de opciones de búsqueda o directamente en el mapa.
Realizar Consultas espaciales	Se puede visualizar información adicional del inmueble acerca del entorno donde se halla ubicado, ya sea entidades culturales, bancos, iglesias, entidades de salud, entidades educativas, que tiene en un radio determinado. Se puede observar otra información como barrios que lo rodean, vías principales que atraviesan el barrio, y mínima distancia en metros que se encuentra ubicado el inmueble de una vía principal.
Inmuebles	El prototipo permite registrar un nuevo inmueble, ingresando inicialmente una dirección, validando si la dirección digitada ya existe o si es correcta.
Formato de Registro de Inmuebles (de la	Se puede registrar un nuevo inmueble, ingresando características básicas del predio (# de habitaciones,

opción del menú Inmuebles)	precio, área, etc.), de igual forma da la opción de subir fotografías del inmueble y escribir observaciones generales. El nuevo inmueble aun no se visualiza en el mapa hasta que no sea habilitado por el Administrador.
Usuarios	Se puede visualizar los usuarios clientes de la inmobiliaria y ver información general de dichos usuarios (datos personales), de igual forma se puede ver los inmuebles que tienen consignados a su nombre en la inmobiliaria.
Nuevo Usuario (de la opción del menú Usuarios)	El prototipo permite registrar un nuevo usuario cliente, ingresando sus datos personales. El nuevo usuario solamente puede ingresar al sistema cuando el administrador lo habilita asignándole un nombre de usuario y una contraseña.
Home	Permite ir de forma rápida a la página principal dando clic en la opción 'Home'
Modificar Datos	Permite cambiar los datos personales del usuario y la contraseña.
Contáctenos	En la opción 'contáctenos', permite enviar un correo electrónico con cualquier tipo de inquietud
Ayuda	En la opción 'ayuda' se puede encontrar claramente la información necesaria para realizar los procedimientos disponibles para el usuario que ingresa al prototipo sin registrarse.
Cerrar Sesión	El prototipo tiene la opción de cerrar la sesión iniciada, y regresar nuevamente a la página principal de usuario público.

Tabla 10 Resultados del Módulo Usuario Asesor Comercial.

CASO DE PRUEBA	RESULTADO
Opciones de Búsqueda.	El prototipo permitió realizar búsquedas de inmuebles teniendo en cuenta características como: Tipo de inmueble, rango de área, rango de precios, tipo de negocio, y barrio.
Búsqueda en el mapa.	El prototipo permitió realizar búsquedas directamente en el mapa ya que se visualizan los inmuebles disponibles para arrendar o vender en forma de puntos.
Resultado de la Consulta	Se puede visualizar información de un inmueble dando clic en el enlace ' <i>ver más</i> ' de cada dirección de inmueble que da como resultado la consulta que se realiza ya sea por el menú de opciones de búsqueda o directamente en el mapa.
Realizar Consultas espaciales	Se puede visualizar información adicional del inmueble acerca del entorno donde se halla ubicado, ya sea entidades culturales, bancos, iglesias, entidades de salud, entidades educativas, que tiene en un radio determinado. Se puede observar otra información como barrios que lo rodean, vías principales que atraviesan el barrio, y mínima distancia en metros que se encuentra ubicado el inmueble de una vía principal.
Acerca de	En la opción 'Acerca de', permite observar información general de las personas y entidades involucradas en el desarrollo del prototipo.
Contáctenos	En la opción 'Contáctenos', permite enviar un correo electrónico con cualquier tipo de inquietud.
Ayuda	En la opción 'Ayuda' se puede encontrar claramente la información necesaria para realizar los procedimientos

	disponibles para el usuario que ingresa al prototipo sin registrarse.
Home	Permite ir de forma rápida a la página principal dando clic en la opción 'Home'.

Tabla 11 Resultados del Módulo Usuario Público.

Se revisaron las observaciones realizadas por las personas que participaron en las pruebas y se corrigieron los errores que se presentaron.

9. CONCLUSIONES Y RECOMENDACIONES

9.1 CONCLUSIONES

- La metodología adoptada en el desarrollo del prototipo fue adecuada, debido a que ésta permite el desarrollo incremental del proyecto, corregir el diseño y el desarrollo de acuerdo a las pruebas que se le realizan.
- Se desarrolló un prototipo software que lleva a cabo el proceso de oferta de inmuebles para la venta o arriendo de una inmobiliaria donde se aplicaron diversas funcionalidades de los SIG y Las Bases de Datos Espaciales, como: la publicación de mapas en Internet y la interacción con ellos en pantalla; la administración de información geográfica y alfanumérica; la realización de diversas consultas espaciales.
- Se diseñó e implementó una base de datos espacial, que permitió almacenar y administrar los mapas utilizados para el desarrollo del prototipo. La elaboración de esta base de datos se hizo en el DBSM PostgreSQL con su extensión Postgis que permitió el almacenamiento y administración de la información geográfica teniendo en cuenta el estándar de almacenamiento de datos geográficos OpenGis en el cual está basado dicha extensión.
- El diseño del prototipo permitió Implementar diferentes tipos de análisis espaciales sobre la información geográfica pertinente a la aplicación propuesta, ya que permitió el almacenamiento de todos los tipos de datos geográficos básicos permitidos por la extensión PostGis, puntos (inmuebles, bancos, entidades de salud, entidades educativas, centros de

salud y entidades culturales), líneas (malla vial y vías principales de la ciudad) y polígonos (barrios, estrato, comunas, manzanas). Estas operaciones geométricas se llevaron a cabo aplicando diferentes funciones espaciales con las que cuenta la extensión PostGis. La utilización de estas funciones permitió la realización de las siguientes funcionalidades del prototipo:

- El “direccionador”, la función que genera la geometría de un inmueble representada como un punto (x,y), dada una dirección (por ejemplo calle 1 # 3-15); En este caso lo que se hace es aplicar funciones espaciales al mapa de “malla vial” que contiene geometrías lineales (calles, carreras, etc.)
 - Realizar consultas acerca de la información del entorno del inmueble: colegios, hospitales, bancos, etc, que existen en un radio determinado.
 - Encontrar el barrio, estrato y comuna en el que se encuentra el inmueble generado por el “direccionador”.
- Para poder interactuar con los mapas en pantalla como se propuso para el prototipo del proyecto se utilizaron herramientas software de origen GNU como fueron el servidor de mapas MapServer y el módulo PhpMapScript de PHP; al utilizar este modulo se pudo acceder a todas las funcionalidades del API de MapServer: MapScript, que permite interactuar con los mapas en pantalla. En el prototipo la interacción con el mapa se da por ejemplo:
 - Cuando el usuario quiere consultar información de un inmueble visualizado en pantalla, los inmuebles son representados por puntos que se localizan geográficamente dentro del mapa de la ciudad, y al dar clic sobre el punto de interés puede ver información adicional.
 - Cuando al mapa visualizado se le realiza zoom in, zoom out,

ubicación por medio del mapa de referencia, escoger las capas que se quieren visualizar (barrios, comunas, manzanas, vías principales, inmuebles), tamaño del mapa a visualizar, entre otros.

- Para el desarrollo del prototipo se utilizaron herramientas software de origen GNU, como fueron el servidor de mapas MapServer, el servidor de archivos Apache, el editor de mapas MapLab, el lenguaje de programación PHP y su modulo PHP/MapScript, el DBMS PostgreSQL con su extensión PostGis. Sumado al bajo costo de estas herramientas se pudo probar su excelente funcionalidad al implementar el prototipo.
- Debido a que no se logró obtener el mapa real de la malla vial de la ciudad de Bucaramanga pues lo venden y no se tenían los recursos para ello, se analizó qué tan indispensable sería tener la malla vial, con el Director se encontró que se podía reemplazar su funcionalidad; entonces se optó por crear un archivo shape en la herramienta Arcview que representara de una forma muy básica la malla vial. Para la creación de este archivo se tuvo en cuenta los mínimos requerimientos necesarios para que contara con las características que se necesitaban para aplicarlo a las funcionalidades del prototipo. Esta capa fue necesario crearla ya que era básica para implementar las funciones del “direccionador”. Para la creación del archivo shape se tuvo en cuenta principalmente:
 - Que las dimensiones de la capa a crear coincidieran con las otras capas de la ciudad como la de barrios, comunas, manzanas.
 - La geometría del shape fuera lineal de tal forma que atravesara todo el mapa “manzanas” de forma horizontal, vertical y diagonal.
 - Cada línea contara con información alfanumérica adjunta como el nombre, la longitud y un índice único que lo identificara para poder realizar las diferentes operaciones geométricas requeridas en el prototipo.

- Durante el desarrollo del proyecto se fortalecieron conocimientos adquiridos durante la carrera de Ingeniería de Sistemas, en temas como el diseño y desarrollo de sistemas de información, diseño e implementación de Base de Datos, herramientas de desarrollo de software, entre otros.
- La funcionalidad en la Web del prototipo se pudo probar gracias a la colaboración del personal vinculado al servidor de la Escuela de Ingeniería de Sistemas y diferentes personas que accedieron al sitio, previamente enteradas de las funcionalidades que debía cumplir cada uno de los roles propuestos en el prototipo. En esta prueba la información proveniente de la base de datos del sitio Web fue recibida, almacenada y presentada a los usuarios que accedieron a él, en los diferentes tipos de usuarios (administrador, asesor comercial, público y cliente); en dicha prueba se evaluó el cumplimiento de los requisitos planteados para el prototipo y se hicieron ajustes menores al mismo, lo que permitió concluir que el ambiente software satisface los requerimientos básicos planteados al comienzo del proyecto. De la misma forma para la realización de pruebas al prototipo, ya de forma local, se contó con la colaboración de personas expertas en la rama de finca raíz. Estas mismas personas fueron también de gran ayuda a la hora de la toma de requerimientos.

9.2 RECOMENDACIONES

Se propone para posteriores versiones del prototipo de la Inmobiliaria desarrollado en el presente proyecto tener en cuenta los siguientes aspectos:

- Manejar técnicas de seguridad en las telecomunicaciones para proteger la transferencia de datos en la herramienta.
- Implementar el prototipo con la malla vial real de la ciudad para tener información más apropiada y apreciar resultados más reales.

- Incluir las funciones de gestión administrativa y financiera que debería llevar un Sistema de Información completo en una inmobiliaria para que se pueda apreciar aun mejor los beneficios que da la integración completa de la información geográfica y alfanumérica que se manejaría en un SIG para una inmobiliaria. y así poder ofrecer el sistema a las entidades del gremio, de tal forma que sea beneficioso económicamente, tanto para esas empresas, como para la Escuela de Ingeniería de Sistemas.
- Implementar funcionalidades de edición de capas de geometrías lineal y poligonal.
- Implementar otras funcionalidades de interacción con el mapa en pantalla, como crear inmuebles directamente haciendo clic en la pantalla, entre otras.

Se propone a la escuela de Ingeniería de Sistemas incursionar en el área de Sistemas de Información Geográfica y Base de Datos Espaciales, ya que es evidente el auge que ha tenido en las diferentes áreas de la academia debido a su gran versatilidad, ya que se puede utilizar información de diversas fuentes y en distintas formas, destacándose también su gran capacidad para ejecutar análisis cada vez más complejos y manejar grandes volúmenes de datos.

Seguir incursionando en el uso de las herramientas software libres, ya que son una excelente alternativa no solamente por su bajo costo sino por la excelente funcionalidad con la que cuentan, que los convierten en fuertes competidores de las soluciones comerciales y en su mayoría ya se encuentran a la par con ellas.

BIBLIOGRAFÍA

10.1 SISTEMAS DE INFORMACION GEOGRAFICA

An Introduction to GIS using ArcView, Environmental Systems Research Institute, Inc.[2003]

BRAVO, Javier. Breve Introducción a la Cartografía y a los Sistemas de Información Geográfica (SIG). Dpto. de Informática, Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (Ciemat). Ministerio de Ciencia y Tecnología. Madrid España, [2000].

HERNANDEZ, Guillermo. Introducción a los Sistemas de Información Geográfica, Grupo GEMA. Escuela de Ingeniería de Sistemas e Informática. Universidad Industrial de Santander. Bucaramanga, [2005].

KENNEDY, Melita; KOPP, Steve, Understanding Map Projections, Environmental Systems Research Institute, Inc, [2001].

SARRÍA, Francisco Alonso. Introducción a los Sistemas de Información Geográfica. Universidad de Murcia, España.[2003]

Shapefile Technical Description, Environmental Systems Research Institute, Inc. [2000].

10.2 BASES DE DATOS

ANDRADE, Roberto. Programación de funciones en PL/pgSQL para PostgreSQL, PostgreSQL 8.0 Documentation, The PostgreSQL Global Development Group,

[2005].

BLASBY, David. Building a Spatial Database in PostgreSQL, Refrations Research, [2005]

BLASBY, Dave; HODGSON, Chris. Developing Applications with PostGIS Mapserver User Meeting, Refrations Research Inc, [2003].

GUTING Ralf Hartmut. An Introduction to Spatial Database Systems, Informatica Practica IV. Universidad Hagen, Alemania, [1998].

HODGSON, Chris; RAMSEY, Paul. Developing Mapserver Applications with PostGIS, Refrations Research Inc., [2005].

MACDONALD, Andrew. Building a Geodatabase, Environmental Systems Research Institute, Inc, [2001]

MARTINEZ, Jose; COLL, Eloina. Análisis vectorial en PosGIS y Oracle Spatial: estado actual y evolucion de la especificacion Simple Feature for SQL, Universidad Politécnica de Valencia (España), [2006].

MOMJIAN, Bruce, PostgreSQL Introduction and Concepts, Addison Wesley, [2002].

Página Oficial de PostGis; [Web], <Dirección URL: <http://www.postgis.org>

Página Oficial de PostgreSQL; [Web], <Dirección URL: <http://www.postgresql.org>

RAMSEY, Paul. Introduction to PostGIS, Refrations Research Inc.[2005].

RAMSEY, Paul. PostGIS Introduction and Evaluation, Refrations Research Inc., [2004].

RIGAUX, Philippe; SCHOLL, Michel y AGNÉS Voisard. Spatial Database With Application to GIS. MORGAN KAUFMANN. [2003]

10.3 DISEÑO

BOOCH, Grady, Rumbaugh James, Jacobson Ivar, El Lenguaje Unificado de Modelado, Addison Wesley, [1999].

FIGUEROA, Pablo, Conceptos de un diagrama de Casos de Uso, [Web] <Dirección URL: http://www.cs.ualberta.ca/~pfiguero/soo/uml/casos_uso01.html>, [2005]

JACOBSON, Ivar, Booch Grady, Rumbaugh, El proceso Unificado de Desarrollo de Software, Addison Wesley, [1999].

SALINAS, Patricio, Casos de Uso (Use Case), [Web] <Dirección URL: <http://www.dcc.uchile.cl/~psalinas/uml/casosuso.html>>, [2005]

10.4 LENGUAJE DE PROGRAMACIÓN

GONZALEZ, Victor. Mapserver y su aplicacion a SIG, Equipo de SIG Universidad Tecnica particular de Loja, Ecuador, [2005].

KROPLA, Bill. Beginning MapServer: Open Source GIS Development, Apress, [2005].

NOZAWA, Jaime .PHP-Mapscript de Mapserver [Web], <Dirección URL:

<http://phpexperto.blogspot.com/archive1.html> >, [2007]

ONTIVEROS, Garcia,. Exportacion de datos Opengis. Trabajo de grado (Ing. De Sistemas Computacionales). Universidad de las Américas, México, [2000].

Página oficial de la organización Internacional Open GeoSpatial Consortium (OGC), [Web], <http://www.opengeospatial.org/>

Página oficial de MapLab <http://www.maptools.org/maplab/>

Página oficial de MapServer, [Web], <http://mapserver.gis.umn.edu/>

Página Oficial de PHP, [Web], <http://www.php.net/>

Página Oficial GNU, [Web], <http://www.gnu.org/>

PhpMapScript [Web], <Dirección URL:

<http://www.postgis.fr/documentation/win32/html/ape.html> >, [2007].

RIGAUX Philippe, SCHOLL Michel, VOISARD Agnés, Spatial Databases with Application to GIS. San Francisco: Morgan Kauffman Publishers, 2002

SKLAR, David; TRACHTENBERG, Adam. PHP Cookbook, O'Reilly, [2002].

TECHER, Jean. Initiation a Postgis et a Phpmascript [Web], <Dirección URL: <http://techer.pascal.free.fr/postgis/tutorialphpmapscriptpostgis/index.html>>, [2007]

TYLER, Mitchell, Web Mapping Illustrated, O'Reilly, [2005].

ANEXO A. OPENGIS

OpenGIS es un formato de almacenamiento de datos Geográficos. Tiene como propósito definir un estándar para un esquema SQL que soporte el almacenamiento, recuperación, consulta y actualización de una colección geoespacial de características simples. Una característica simple es definida por la especificación abstracta de OpenGis para tener atributos espaciales y no espaciales.

La especificación OpenGis es realizada por el Open Gis Consortium (OGC). OGC, es una organización sin ánimo de lucro, fundada en 1994 y dedicada a la promoción de nuevas aproximaciones técnicas y comerciales para geoprocesamiento abierto e interoperable. Las empresas que participaron en dicha construcción son:

- Environmental Systems Research Institute, Inc. (ESRI)
- **IBM Corporation**
- Informix Software, Inc.
- MapInfo Corporation.

Su objetivo es conseguir acuerdos sobre interfaces de software estándar abiertos que posibiliten la interoperación entre sistemas de geoprocesamiento de vendedores diferentes y de diferentes tipos (GIS/SIG, teledetección, cartografía automática, gestión de instalaciones, etc.). OpenGIS fue creado con la sola idea de unificar los formatos propietarios existentes suministrando una especificación libre para SQL con soporte espacial.

Principales características de la especificación OpenGis:

- Define tipos de objetos GIS estándar, los cuales son manipulados por funciones, y un conjunto de tablas de metadatos.

- Define las operaciones que se pueden realizar sobre estos datos geográficos: Area, Length, Boundary, Buffer, entre otras.
- Requiere que los objetos incluyan el identificador del sistema de referencia espacial (SRID). El SRID es requerido cuando se insertan un objeto espacial en la base de datos.
- Define las transformaciones de coordenadas, a través del SRID.

OpenGIS define dos formas de representar los objetos espaciales, las dos formas guardan información del tipo de objeto y sus coordenadas:

(WKT) Well-know text.

La representación Well-Known Text (WKT) de Geometrías está diseñada para intercambiar datos geométricos en formato ASCII.

Por ejemplo:

```
POINT (0 0 0)
LINESTRING (0 0, 1 1, 2 2)
MULTIPOINT (0 0 0, 1 2 1)
MULTILINESTRING ((0 0 0, 1 2 1), (1 2 3, 4 5 6))
MULTIPOLYGON (((0 0 0, 1 2 1), (1 2 3, 4 5 6)), ((0 0 0, 1 2 1), (1 2 3, 4 5 6)))
GEOMETRYCOLLECTION ( POINT(0 2 3), LINESTRING (0 0, 1 1, 2 2))
```

(WKB) Well-know binary

La representación Well-Known Binary (WKB) de valores geométricos está definida por la especificación OpenGIS. WKB se utiliza para intercambiar datos como cadenas binarias representadas por valores BLOB que contienen información geométrica WKB.

Por ejemplo, un valor WKB que corresponde a un POINT(1 1) consiste en esta secuencia de 21 bytes (cada uno representado aquí por dos dígitos hexadecimales):

0101000000000000000000F03F000000000000F03F

La secuencia puede descomponerse en los siguientes componentes:

Orden de byte: 01

Tipo WKB : 01000000

X : 000000000000F03F

Y : 000000000000F03F

El tipo WKB es un código que indica el tipo de geometría. Los valores del 1 al 7 significan Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, y GeometryCollection.

Existen dos tablas de meta-datos en la especificación OpenGIS:

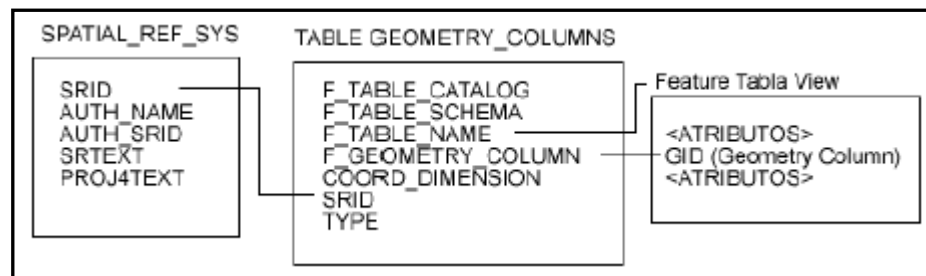


Figura 1. Esquema Conceptual del estándar OpenGis

SPATIAL_REF_SYS:

La tabla SPATIAL_REF_SYS contiene información de diferentes Sistemas de Referencia Espacial. Cada sistema de referencia espacial identifica el tipo de coordenadas para las geometrías almacenadas en las columnas y da significado a sus valores geométricos para cualquier instancia. En general esta tabla contiene un identificador numérico y una descripción

textual del sistema de coordenadas espacial de la base de datos. Los campos de esta tabla son:

- SRID: Identificador del sistema de referencia espacial.
- AUTH_NAME: Nombre del estándar para el sistema de referencia espacial. Por ejemplo: EPSG.
- AUTH_SRID: Identificador según el nombre del estándar del sistema de referencia espacial.
- SRTEXT: representación textual del sistema de referencia espacial.
- PROJ4TEXT: Es una librería que usa PostGis para transformar coordenadas. Esta columna contiene una cadena con definición de las coordenadas de Proj4 para un SRID dado.

Ejemplo:

sr_id int4	auth_name varchar	auth_sr_id int4	sr_text varchar	proj4_text varchar
2000	EPSG	2000	PROJCS["Anguilla 19	+proj=tmerc +lat_0=0 +lo

➤ GEOMETRY COLUMNS:

Esta tabla almacena la información de metadatos de las columnas de geometría de las tablas creadas en la BDE. Los campos de esta tabla son:

- F_TABLE_CATALOG,F_TABLE_SCHEMA,F_TABLE_NAME: distingue totalmente la tabla de características que contiene la columna geométrica.
- SRDI: Identificador del sistema de referencia espacial. Es una llave foránea que referencia SPATIAL_REF_SYS
- F_GEOMETRY_COLUMN: Es el nombre de la columna geométrica.
- COORD_DIMENSION: dimensión espacial de la columna geométrica (2D ó 3D).

- TYPR: Es el tipo de objeto espacial. POINT, LINESTRING, POLYGON, MULTYPOINT, GEOMETRYCOLLECTION. Para un tipo heterogéneo se debe usar el tipo GEOMETRY.

Ejemplo:

f_table_catalog varchar	f_table_schema varchar	f_table_name varchar	f_geometry_column varchar	coord_dimension int4	srid int4	type varchar
"	public	comunas	the_geom	2	-1	MULTIPOLYGON

➤ **FEATURE_TABLE_VIEW:**

Tabla creada por el usuario. Es la tabla de características o datos no espaciales de los temas, es decir los atributos de cada tema. En estas tablas se tiene la libertad de de agregar atributos de acuerdo alas necesidades de las aplicaciones que utilizan la especificación OpenGis.

ANEXO B. MAPFILE

El Mapfile es el archivo de mecanismo básico de configuración de MapServer define una colección de objetos de mapa, que juntos determinan la apariencia y comportamiento del mapa como es desplegado en la Web. El MapFile es un archivo de texto con extensión “*.map”,

Este archivo define las capas disponibles, estilo en que se presentan, su simbología, sistema de referencia, formato en el que se generara la imagen, fondo, símbolos, datos espaciales, nombres de las plantillas html a utilizar tamaño del mapa resultante.

El MapFile consta de varias secciones u "objetos". Cada objeto se inicia con el nombre del objeto y termina con la palabra END. El MapFile posee una estructura jerárquica de tal manera que algunos objetos residen dentro de otros, en la parte superior de la jerarquía se encuentra el objeto MAP, todos los otros objetos pertenece a este. Aquí se encuentran objetos como PROJECTION, que pueden pertenecer al objeto MAP o al objeto LAYER (el cual a la vez pertenece al objeto MAP).²⁴

Cada objeto puede definir otros objetos o palabras claves de determinados parámetros, algunos de los cuales son de obligatoria inclusión mientras que otros son opcionales o tienen un valor asignado por defecto.

²⁴ Ver Figura 1

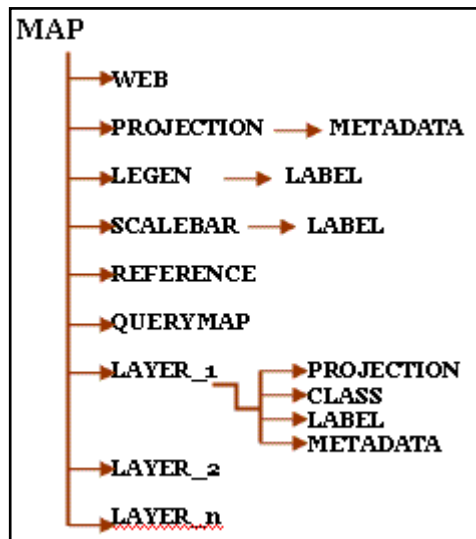


Figura 1 .Estructura general del MapFile

Uno de los principales objetos utilizados en el mapfile es el objeto Layer. En este objeto se definen las características de cada capa utilizada y que son dibujadas en el orden de aparición en el mapfile. Entre las secciones con las que cuenta el objeto layer se encuentra entre otras ‘el escenario de datos’ donde se define el formato de datos a utilizar ya sea Shapefile, OGR, Databases, WMS.

El *shapefile* es el formato de datos más común usado con MapServer y se accede a él en el mapfile dentro del objeto Layer en la sección DATA, allí se define el nombre y la ruta de acceso del archivo que contiene los datos espaciales La extensión del archivo no es necesario para los shapfile. Ejemplo de su utilización en el mapfile es:

```
DATA "bucaramanga/barrios"
```

El formato *OGR* permite a mapserver acceder a gran variedad de formatos de archivos de datos vectoriales en, Por ejemplo MapInfo TAB o Microstation DGN. Se accede en el mapfile dentro del objeto layer en la sección CONNECTIONTYPE y CONNECTION. Ejemplo de su utilización en el mapfile es:

```
CONNECTIONTYPE OGR
```

```
CONNECTION "data/tab/builtup_areas.tab"
```

- El formato *WMS* (Web Map Server) permite usar datos desde varios servidores y está habilitado para la creación de una red de servidores de mapas desde el cual el cliente pueda construir mapas personalizados. Se accede en el mapfile dentro del objeto layer en la sección CONNECTIONTYPE, CONNECTION y METADATA. Ejemplo de su utilización en el mapfile es:

```

CONNECTION "http://www2.dmsolutions.ca/cgi-bin/mswms_gmap?"
CONNECTIONTYPE WMS
METADATA
"wms_srs" "EPSG:42304 EPSG:42101 EPSG:4269 EPSG:4326
EPSG:42304"
"wms_name" "prov_bound"
"wms_server_version" "1.1.0"
"wms_formatlist" "image/gif,image/png,image/jpeg,image/wbmp"
"wms_format" "image/gif"

```

- Mapserver permite también acceder a datos espaciales almacenados en base de datos, por ejemplo en formato postgis, oraclespatial y SDE . Se accede en el mapfile dentro del objeto layer en la sección CONNECTIONTYPE, CONNECTION y DATA. Ejemplo de su utilización en el mapfile es:

Utilizando Postgis:

```

CONNECTIONTYPE postgis
CONNECTION "user=gis_user dbname=gis host=localhost" >
DATA "the_geom from s81"

```

En la sección CONNECTION, la línea de conexión a Postgis es básicamente una conexión a PostgreSQL, de la siguiente forma:
"user=nobody password=***** dbname=dbname host=localhost port=5432"

En la sección DATA Este parámetro debe ser de la forma:
"<nombre_columna> from <nombre_tabla>", donde <nombre_columna> es el nombre de la columna que contiene la geometría de los objetos y <nombre_tabla>" es el nombre de la tabla desde donde se lee la columna geométrica.

ANEXO C. SOFTWARE LIBRE

TABLA DE CONTENIDO

INTRODUCCION	¡ERROR! MARCADOR NO DEFINIDO.
1. DEFINICION	122
2. CLASIFICACION DEL SOFTWARE LIBRE	123
3. GNU/GPL	124

INTRODUCCIÓN

Actualmente se puede encontrar diversas aplicaciones de software libre fáciles de instalar, manejar y con interfaces agradables; aspectos que en años anteriores hacían que este tipo de software fuera poco atractivo para los usuarios.

Con la evolución de productos libres, aquellas empresas, personas u organizaciones que se atreven a explorar y confiar en este tipo de herramientas, consiguen gran cantidad de beneficios como la reducción de costos, estabilidad, seguridad; al mismo tiempo que pueden modificar y ajustar el código a la medida de sus necesidades. Esto sumado a que por ser código abierto muchos programadores pueden trabajar de forma conjunta, lo que hace que siempre se esté actualizando; evitando así que los programas queden obsoletos.

Esta característica del software libre hace que quienes lo utilizan tengan un gran número de personas que pueden brindar soporte de las aplicaciones o del mismo código.

1. DEFINICION

El software libre está amparado por la licencia GNU/GPL. Al hablar de software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Cuando se habla de libre, no precisamente quiere decir gratuitos, pues algunas personas pueden cobrar por distribuir, mejorar o adaptar programas o códigos; se refiere especialmente a cuatro clases de libertad para los usuarios de software:

- a. Ejecutarlo con cualquier propósito
- b. Estudiar cómo funciona y adaptarlo a sus necesidades
- c. Distribuir copias
- d. Mejorarlos y liberar esas mejoras al público

Con la única restricción del copyleft, es decir, cualquiera que redistribuya el software, con o sin cambios, debe dar las mismas libertades que antes, y con el requisito de permitir el acceso al código fuente (imprescindible para ejercer las libertades a y c)

2. CLASIFICACION DEL SOFTWARE LIBRE

De acuerdo al costo de adquisición: el Software Libre puede ser de las dos clases, es decir, de costo cero o de costo mayor que cero. Lo que lo diferencia del Software Propietario es que su costo es independiente del número de computadores que se poseen. Por ejemplo, en el caso del Sistema Operativo Microsoft Windows 3.1/95/98/Me/NT/2000/XP por cada computador en que se instale debo pagar una licencia. En cambio, si se utiliza en Sistema Operativo GNU/Linux (en cualquiera de sus distribuciones, como Red Hat, Mandrake, Debian, Ubuntu) se debe pagar una sola licencia independiente del número de equipos de computo en los que se instale (algunas licencias no tienen costo).

➤ **De acuerdo a la apertura del código fuente:** el Software Libre siempre es "open source", es decir, de código fuente abierto, ya que el acceso al código fuente es necesario para el ejercicio de las libertades a y c arriba descritas.

➤ **De acuerdo a su protección:** El Software Libre siempre está protegido con licencias, y más específicamente, con licencias de copyleft. Esto es, para que cuando se adquiera un Software libre se le haga alguna modificación y se compile no se pueda distribuir con código cerrado. No se protegen con licencias Copyright, porque existe la posibilidad que se le ponga alguna restricción y por lo tanto no va a seguir siendo Software libre

➤ **De acuerdo a su legalidad:** El Software Libre siempre es legal, porque al ser usado, estudiado, modificado, adaptado y/o mejorado no se está violando ninguna norma, ya que de por sí este tipo de software permite hacerlo, con la única salvedad de no poder agregarle ninguna restricción adicional cuando lo transfiera a otra persona.

La libertad para usar un programa significa la libertad para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicarlo al desarrollador o a alguna otra entidad específica.

3. GNU/GPL

GNU/GPL (Licencia Pública General) es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que limiten esas libertades a los usuarios.

Existen varias licencias "hermanas" de la GPL, como la licencia de documentación libre de GNU (GFDL) que cubre los artículos de la Wikipedia, la Open Audio License, para trabajos musicales, etcétera, y otras menos restrictivas, como la LGPL, o la LGPL (*Lesser General Public License* o *Library General Public License*), que permiten el enlace dinámico de aplicaciones libres a aplicaciones no libres.

ANEXO D. FUNCIONES POSTGIS

PostGIS cuenta con un gran número de funciones para análisis espaciales/topológicas que extienden el propio SQL del PostgreSQL. A continuación se hará una explicación de algunas de las principales funciones que se utilizaron para realizar el prototipo propuesto en este proyecto.

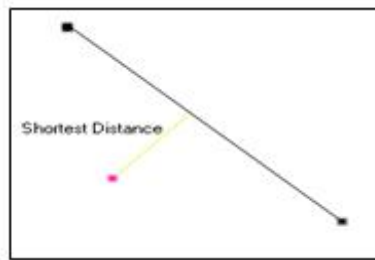
➤ **Función Distance:**

Distance(geometry, geometry)

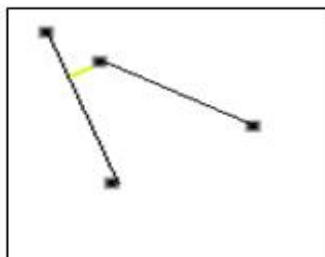
La función *distance* devuelve la distancia cartesiana entre dos geometrías²⁵. El valor que devuelve es la distancia más corta que separa a los objetos espaciales.



Distancia entre puntos



Distancia entre un punto y una línea



Distancia entre dos líneas.



Distancia entre dos polígonos

Figura 1. Distancias entre geometrías

²⁵ Tipos de geometrías que maneja PostGis: Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon.

Ejemplos de la aplicación de la función *distance*

Análisis: Distancia que hay de un determinado inmueble a "Via a la Costa";

Query:

```
select v.nombre, distance(i.the_geom, v.the_geom) from inmueble i, vias v where v.gid= 1 AND i.gid= 1;
```

Resultado:

"Via a la Costa";5581.42298883402

Análisis: Nombre de las entidades educativas que están en un radio de 500 mts de un inmueble específico.

Query

```
SELECT e.nombre FROM educacion e, inmueble i where distance(i.the_geom, e.the_geom) < 500 AND i.gid=1;
```

Resultado:

"BETHLEMITAS."

"ESCUELA ALBANIA."

➤ **Función Touches**

Touches(geometry, geometry)

La función *Touches* analiza si dos geometrías poseen alguna intersección y retorna verdadero si esto ocurre;

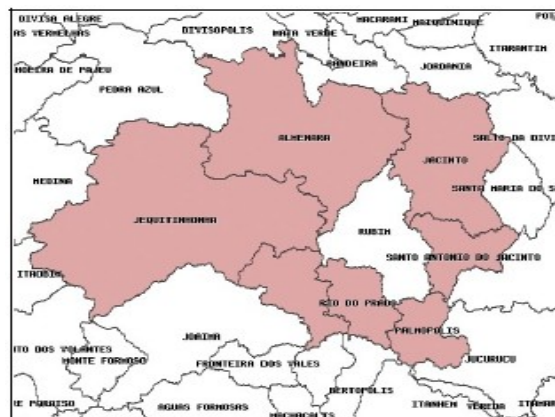


Figura 2. Ejemplo grafico del resultado de la función *Touches*

Ejemplos de la aplicación de la función *Touches*

Análisis: Barrios que rodean el barrio donde está ubicado determinado inmueble;

Query:

```
select b2.nombre from barrios b1, barrios b2 where touches (b1.the_geom,  
b2.the_geom)='t' AND b1.gid=55;
```

Resultado:

"EL PRADO", "MORRORICO",
"ALBANIA", "QUINTA BRIDADA"

➤ **Función Intersects**

Intersects(geometry,geometry)

La función *Intersects* analiza si dos geometrías poseen alguna intersección y retorna verdadero si esto ocurre, retorna un valor booleano

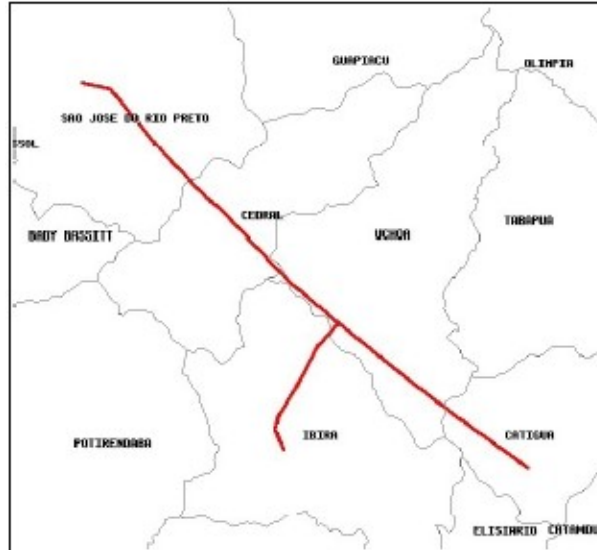


Figura 3. Ejemplo grafico de la función *Intersect*

Ejemplos de la aplicación de la función *Intersects*

Análisis: vías que itersectan el barrio 'QUINTA BRIGADA'.

Query:

```
select DISTINCT t2.nombre from barrios t1, vias t2 where intersects(t1.the_geom,  
t2.the_geom)='t' AND t1.nombre= 'QUINTA BRIGADA'.
```

Resultado:

"Avenida Quebrada Seca"

➤ **Función Contains**

contains(geometry, geometry)

La función *Contains* analiza si una geometría contiene a la otra y retorna verdadero si esto ocurre, retorna un valor booleano.

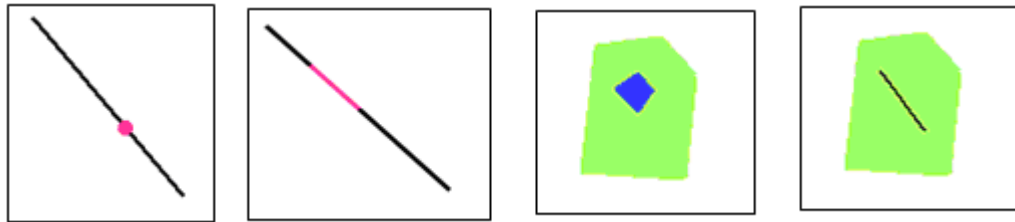


Figura 4. Ejemplos de Geometrías contenidas en otras

Ejemplos de la aplicación de la función *Contains*

Análisis: A qué comuna pertenece el barrio 'LAS AMERICAS'.

Query:

```
select t1.nombre_com from comunas t1 , barrios t2 where Contains (t2.the_geom,  
t1.the_geom)='t' AND t2.nombre='LAS AMERICAS';
```

Resultado:

"Comuna Oriental"

Análisis: A qué Barrio pertenece un determinado inmueble.

Query:

```
select b.nombre from inmueble i, barrios b where contains (b.the_geom,  
i.the_geom)='t' AND i.gid=1;
```

Resultado:

"LAS AMERICAS"

➤ **Función intersection**

Intersection(geometry, geometry)

La función *Intersection* retorna como resultado la geometría que representa el conjunto de puntos de intersección de una geometría con otra.

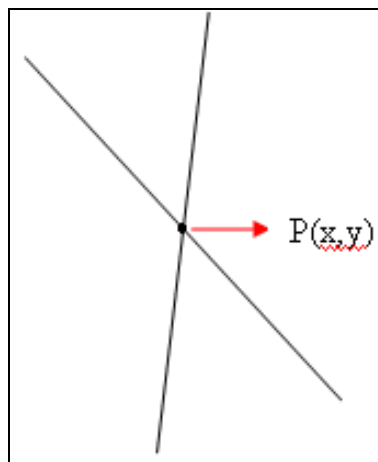


Figura 5. Ejemplo de la función *Intersection*²⁶

Ejemplos de la aplicación de la función *Intersection*

Análisis: Identifique el punto de intersección entre calle 4 con carrera 4.

Query:

```
select intersection(t1.the_geom,t2.the_geom) from malla_vial t1, malla_vial t2
where t1.nombre=calle 4 and t2.nombre=carrera 4;
```

Resultado:

"01010000008B6BCE909105FA405FD29EDCE47AF340"

²⁶ Ejemplo de intersección de dos líneas donde se genera un único punto con coordenadas (x,y)

➤ **Función length**

length(geometry)

La función *length* devuelve el valor de la longitud de la geometría, solo para geometrías lineales.

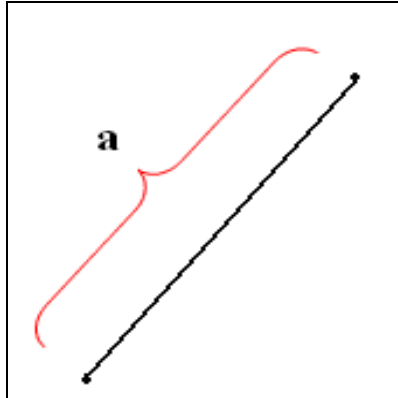


Figura 6. Ejemplo de la función *length*²⁷

Ejemplos de la aplicación de la función *length*

Análisis: Hallar la longitud de la calle 4.

Query:

```
select length(the_geom) from malla_vial where nombre = 'calle 4';
```

Resultado:

9551.41405277447

➤ **Función line locate_point**

line_locate_point(geometry, geometry)

La función *line_locate_point(geometry, geometry)* arroja la porción de la línea donde se encuentra ubicado un punto. Retorna un numero de 0 a 1; por ejemplo:

²⁷ Donde **a** es el valor de la longitud total de la línea

0.7865 quiere decir que el punto se ubica a un 78.65% de la longitud de la línea. Se debe tener en cuenta que la función exige que las geometrías sean una línea y un punto

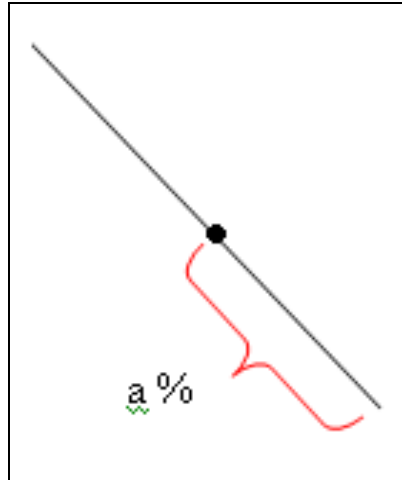


Figura 7. Ejemplo de la función `line_locate_point`

Ejemplos de la aplicación de la función `line_locate_point`

Análisis: Hallar la fracción que hay hasta el punto de intersección de la calle 4.

Query:

```
select line_locate_point(geometryN(the_geom,1),  
'01010000008B6BCE909105FA405FD29EDCE47AF340') from malla_vial where  
nombre = calle 4;
```

Resultado:

0.643193395344696

El tipo de geometría de la tabla `malla_vial` es `MultiLineString`, Si la función `line_locate_point` se hubiera utilizado de la siguiente forma:

Query:

```
Select line_locate_point(the_geom,  
'01010000008B6BCE909105FA405FD29EDCE47AF340') from malla_vial where
```

nombre = calle 4;

La consulta arrojaría el ERROR: line_locate_point: 1st arg isnt a line.

Este error se debe a que la función solo admite que sea haga con una geometría LineString y en el caso de esta consulta se está realizando con una geometría MultiLineString, entonces toca convertirlo a LineString con la función geometryN(the_geom,1), que se analiza a continuación.

➤ **Función geometryN.**

geometryN(geometry, int),

La función geometryN retorna la enésima geometría del objeto, si esta geometría es GeometryCollection, MultiPoint, MultiLineString, MultiPolygon, si no, retorna Null. 1 es 1ra geometría. Ejemplos de la aplicación de la función geometryN

Análisis: Mostrar cual es el tipo de geometría de del campo the_geom de la tabla malla_vial y luego convertir esa geometría de tipo MULTILINESTRING a LINESTRING.

Query:

Mostrar el tipo de geometría de la tabla malla_vial

```
select astext(the_geom) from malla_vial where gid = 4
```

Resultado:

```
"MULTILINESTRING((100606.091420726 78378.5940103117,109901.905829979  
80573.4390791631))"
```

Query:

Convertir la geometría

```
select astext(geometryN(the_geom,1)) from malla_vial where gid = 4
```

Resultado:

```
"LINESTRING(100606.091420726 78378.5940103117,109901.905829979  
80573.4390791631)"
```

➤ **Función line_interpolate_point**

line_interpolate_point(linestring, location)

La función line_interpolate_point da como resultado la geometría del punto que queda en esa porción (porcentaje) de la línea.

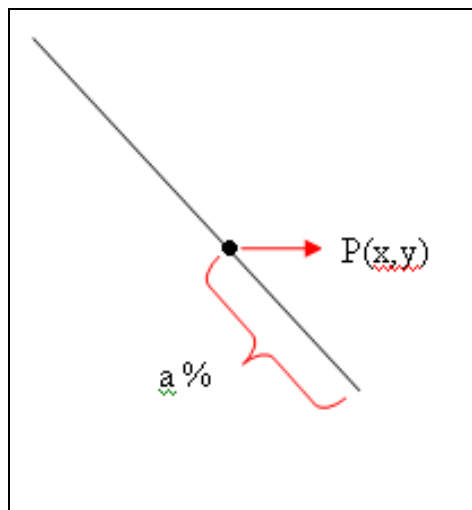


Figura 8. Ejemplo de la función line_interpolate_point

Ejemplos de la aplicación de la función line_interpolate_point

Análisis: Hallar la geometría de un punto sobre la calle 4 (gid=4).

Query:

```
select      (line_interpolate_point(geometryN(the_geom,1),0.695541665))      from  
malla_vial where gid=4;
```

Resultado:

```
"0101000000D327817BFB23FA40F00509341382F340"
```

Como en el caso de la función `line_locate_point(geometry, geometry)`.es necesario convertir la geometría de MULTILINESTRING (que es el tipo definido en la columna `the_geom` de la tabla `malla_vial`) a LineString con la función `geometryN(the_geom,1)`.

➤ **Función astext**

astext(geometry)

La función `astext` retorna la geometría en representación Well-Known Text (WKT), por ejemplo: "LINESTRING(100606.091420726

78378.5940103117,109901.905829979 80573.4390791631)"

Ejemplos de la aplicación de la función `astext`

Análisis: Mostrar el tipo e geometría de la tabla `malla_vial`

Query:

```
select astext(the_geom) from malla_vial where gid = 4
```

Resultado:

```
"MULTILINESTRING((100606.091420726 78378.5940103117,109901.905829979 80573.4390791631))".
```

ANEXO E. DICCIONARIO DE DATOS

A continuación se muestra la descripción de cada tabla, registros y relaciones de las tablas que hacen parte de la base de datos de bdinmobiliaria. Se explica también la relación que existe entre todas las tablas que tiene campos geométricos y las tablas de Metadatos SPATIAL_REF_SYS, GEOMETRY_COLUMNS, que junto al conjunto de funciones PostGis son los que permiten manipular los objetos geográficos.

- **Tabla barrios:** Almacena la información de los barrios de Bucaramanga

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
codigo	numeric		Código del barrio
nombre	character varying	30	Nombre del barrio
area	numeric		Área del barrio
perimeter	numeric		Perímetro del barrio
cobser	numeric		Cobertura de servicios públicos
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría, dimensión y proyección
barr_comuna	numeric		Llave foránea que referencia a la tabla comunas
xminimo	numeric	10	Extensión x min del polígono que forma el barrio
yminimo	numeric	10	Extensión y min del polígono que forma el barrio
xmaximo	numeric	10	Extensión x max del polígono que forma el barrio
ymaximo	numeric	10	Extensión y max del polígono que forma el barrio

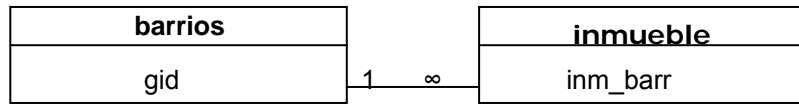
Relaciones

comunas



Tipo de Relación: Uno a muchos

inmueble



Tipo de Relación: Uno a muchos

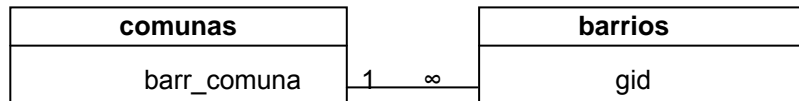
- **Tabla comunas:** Almacena la información de las comunas de Bucaramanga

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
area	double precision		Área de la comuna
perimeter	double precision		Perímetro de la comuna
nombre	smallint	30	Nombre de la comuna en numero
nombre_com	character varying	50	Nombre de la comuna
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría, dimensión y proyección

Relaciones

barrios



Tipo de Relación: Uno a muchos

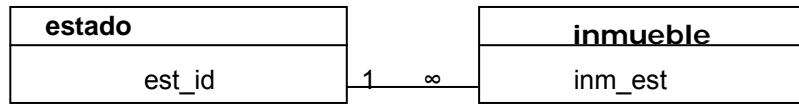
- **Tabla estado:** Almacena la información de los estados en que se puede encontrar un inmueble.

Columnas

Nombre	Tipo	Tamaño	Descripción
esta_id	numeric		id del tipo de inmueble , es la llave primaria.
esta_nombre	character varying	30	Nombre del tipo de inmueble

Relaciones

inmueble



Tipo de Relación: Uno a muchos

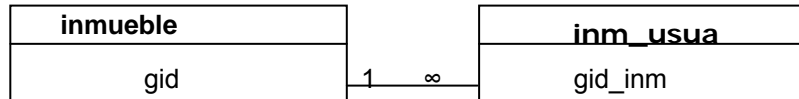
- **Tabla *inm_usua*:** Almacena las relaciones entre los inmuebles y los usuarios propietario y asesor comercial.

Columnas

Nombre	Tipo	Tamaño	Descripción
usua_id	bigint		llave foranea que hace referencia a la tabla usuarios
gid_inm	bigint		llave foranea que hace referencia a la tabla inmueble

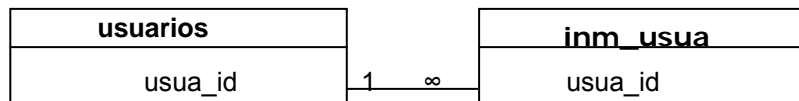
Relaciones

inmueble



Tipo de Relación: Uno a muchos

usuarios



Tipo de Relación: Uno a muchos

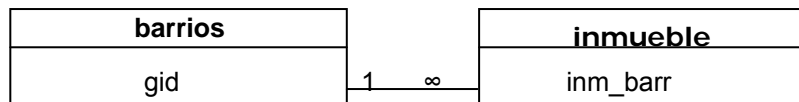
- **Tabla *inmueble*:** Almacena información de los inmuebles.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
inm_dir	character varying	60	direccion del inmueble
inm_barr	bigint		Llave foránea que referencia a la tabla barrios
tiinm_id	smallint		Llave foránea que referencia a la tabla tipo_inmueble
esta_id	smallint		Llave foránea que referencia a la tabla estado
inm_numh	integer		número de habitaciones
inm_precio	numeric	20	precio
inm_ban	numeric		número de baños
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría, dimensión y proyección
inm_area	numeric	5	area en mts cuadrados del inmueble
inm_estra	integer		Llave foránea que referencia a la tabla estrafin
inm_admon	character varying		almacena inf del pago de admon del inmueble
inm_closet	integer		número de closet
inm_garaje	character varying		almacena inf sobre la existencia de garaje en el inm
foto1	character varying		almacena la ruta donde se almacenan los archivos de fotos del inmueble
inm_revisado	integer		informa inf sobre el estado revisado del inmueble por parte del admon
inm_text	text		información adicional del inmueble

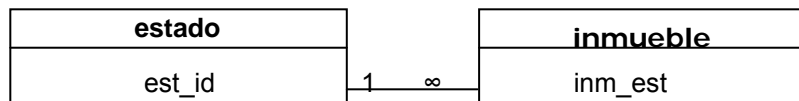
Relaciones

barrios



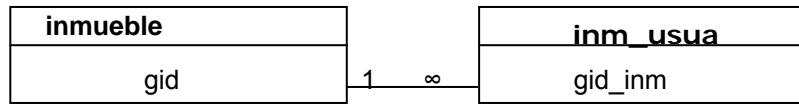
Tipo de Relación: Uno a muchos

estado



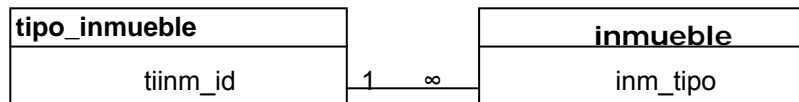
Tipo de Relación: Uno a muchos

Inm_usua



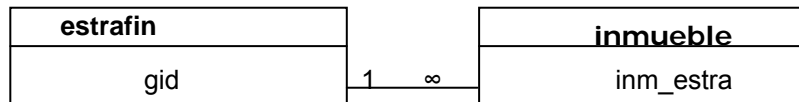
Tipo de Relación: Uno a muchos

Tipo_inmueble



Tipo de Relación: Uno a muchos

Estrafin



Tipo de Relación: Uno a muchos

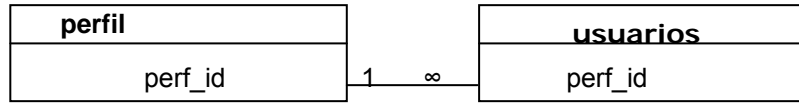
- **Tabla perfil:** Almacena la información de los perfiles de los usuarios

Columnas

Nombre	Tipo	Tamaño	Descripción
perf_id	integer		id del tipo de perfil de usuario , es la llave primaria.
perf_nombre	character varying	50	Nombre del tipo de perfil de usuario

Relaciones

usuarios



Tipo de Relación: Uno a muchos

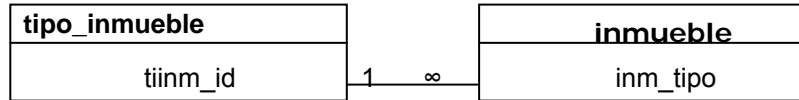
- **Tabla tipo_inmueble:** Almacena la información de los tipos de inmueble posibles en la inmobiliaria.

Columnas

Nombre	Tipo	Tamaño	Descripción
tiinm_id	numeric	4	id del tipo del tipo de inmueble, es la llave primaria.
tiinm_nombre	character varying	50	Nombre del tipo de inmueble

Relaciones

inmueble



Tipo de Relación: Uno a muchos

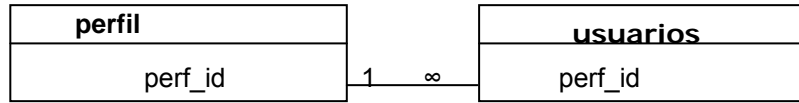
- **Tabla usuario:** Almacena la información de los usuarios registrados.

Columnas

Nombre	Tipo	Tamaño	Descripción
usua_id	bigint		identificador del usuario, es la llave primaria
usua_nombre	character varying	50	nombre del usuario
usua_pass	character varying	12	password del usuario
perf_id	integer		llave foránea que hace referencia a la tabla perfil
usua_dir	character varying	60	dirección de correspondencia del usuario
usua_tel	character varying	30	teléfono del usuario
usua_login	character varying	30	login de usuario
usua_correo	character varying	50	correo electrónico del usuario
usua_revisado	integer		estado revisado del inmueble por parte del admon
usua_text	text		información adicional del usuario

Relaciones

perfil



Tipo de Relación: Uno a muchos

A continuación se muestra la descripción de las otras tablas que tiene campos geométricos pero que no están relacionadas directamente con la tabla inmueble sino que se utiliza para realizar consultas espaciales. Estas tablas se relacionan entre si a partir las tablas de Metadatos SPATIAL_REF_SYS, GEOMETRY_COLUMNS, esta última almacenando toda la información geográfica de las tablas que contienen campos geométricos.

- **Tabla bancos:** Almacena la información de de las entidades bancarias de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
bancos	character varying	27	Nombre de la entidad bancaria
direccion	character varying	39	Dirección donde esta ubicada la entidad
dis_stan	character varying	25	Dirección estándar
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla cultura:** Almacena la información de de las entidades culturales de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
nombre	character varying	66	Nombre de la entidad cultural
direccion	character varying	44	Dirección donde esta ubicada la entidad cultural
dir_stan	character varying	25	Dirección estándar
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla educacion:** Almacena la información de de las entidades educativas de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
Codigo	character varying	6	Código de la entidad educativa
Uso	character varying	10	Uso, si es oficial o publico
Nombre	character varying	30	Nombre de la entidad cultural
Telefono	character varying	10	Teléfono de la institución
Direccion	character varying	60	Dirección donde esta ubicada la entidad educativa
nombre_mun	character varying	13	Nombre del municipio donde esta ubicado la entidad educativa
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla entidades_religion:** Almacena la información de de las entidades religiosas de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
Tipo	bigint		tipo de entidad religiosa
Nombre	character varying	59	Nombre de la entidad religiosa
Direccion	character varying	35	Dirección donde esta ubicada la entidad religiosa
dir_stan	character varying	25	Dirección en nomenclatura estándar
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla entidades_salud:** Almacena la información de las entidades de salud de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
Nombre	character varying	29	Nombre de la entidad de salud
Direccion	character varying	33	Dirección donde esta ubicada la entidad de salud
nivel	character varying	5	Nivel de complejidad de la entidad de salud
dir_stan	character varying	25	Dirección en nomenclatura estándar
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla estrafin:** Almacena la información de los estratos de Bucaramanga.

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
area	double precision		area que abarca e estrato
perimetro	double precision		perimetro del estrato
estra	smallint		Nombre del estrato en numero
label	character varying	18	Nombre del estrato
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla omzbuc:** Almacena toda la información de las manzanas de Bucaramanga

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
area	double precision		Área que abarca la manzana
perimeter	double precision		Perímetro de la manzana
omzbuc_id	bigint		Código de la manzana
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla vias:** Almacena toda la información de las principales vías de Bucaramanga

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
fnode_	bigint		identificador de inicio de arista
tnode_	bigint		identificador fin de arista
length	double precision		longitud de la vía
vias_id	bigint		identificador de la vía
nombre	character varying	100	nombre de la vía
clasificac	character varying	17	clasificación de la vía
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

- **Tabla mll_vial:** Almacena toda la información de la malla vial

Columnas

Nombre	Tipo	Tamaño	Descripción
gid	serial		Índice geométrico, es la llave primaria.
numero	integer		numero de vía
nombre	character varying	50	nombre de la vía
longitud	numeric	20	longitud de la vía
the_geom	geometry		Columna de tipo espacial, donde se especifica el tipo de geometría

A continuación se explican las tablas de Metadatos SPATIAL_REF_SYS, GEOMETRY_COLUMNS

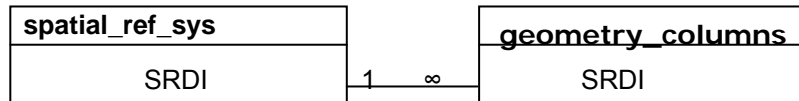
- **Tabla SPATIAL_REF_SYS:** contiene información de diferentes Sistemas de Referencia Espacial. Cada sistema de referencia espacial identifica el tipo de coordenadas para las geometrías almacenadas en las columnas espaciales y da significado a sus valores geométricos para cualquier instancia.

Columnas

Nombre	Tipo	Tamaño	Descripción
SRID	integer		identificador el sistema de referencia espacial
AUTH_NAME	character varying	256	nombre del autor del sistema de referencia espacial
AUTH_SRID	integer		identificador del autor del sistema de referencia espacial
SR_TEXT	character varying	2048	Representacion textual del sistema de referencia espacial
PROJ4TEXT	character varying	2048	Dimensión espacial de la columna geométrica

Relaciones

geometry_column



Tipo de Relación: Uno a muchos

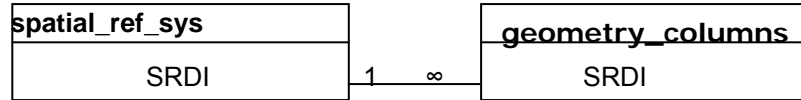
- **Tabla GEOMETRY_COLUMNS:** Almacena la información de metadatos de las columnas geométricas de las tablas creadas en la BDE. Esta tabla consiste en un renglón por cada tema. Cada tema puede ser de un tipo diferente de dato geométrico. Los datos almacenados en el tema incluyen información necesaria para navegar por las tablas.

Columnas

Nombre	Tipo	Tamaño	Descripción
F_TABLE_CATALOG	character varying	256	El catalogo para clasificar el tema
F_TABLE_SCHEMA	character varying	256	identificador de la tabla de características de la tabla.
F_TABLE_NAME	character varying	256	Nombre de la tabla geométrica
F_GEOMETRY_COLUMN	character varying	256	Es el nombre de la columna geométrica.
COORD_DIMENSION	integer		Dimensión espacial de la columna geométrica
SRDI	integer		Identificador del sistema de referencia espacial. Es una llave foránea que referencia a la tabla SPATIAL_REF_SYS
TYPE	character varying	30	Es el tipo de objeto espacial.

Relaciones

spatial_ref_sys



Tipo de Relación: Uno a muchos

La relación de la tabla **GEOMETRY_COLUMNS** con las demás tablas de la base de datos que cuentan con campos geométricos se realiza a través de los índices geométricos de cada tabla espacial. Mientras que la relación de las dos tablas de metadatos se da por el campo **SRID**.

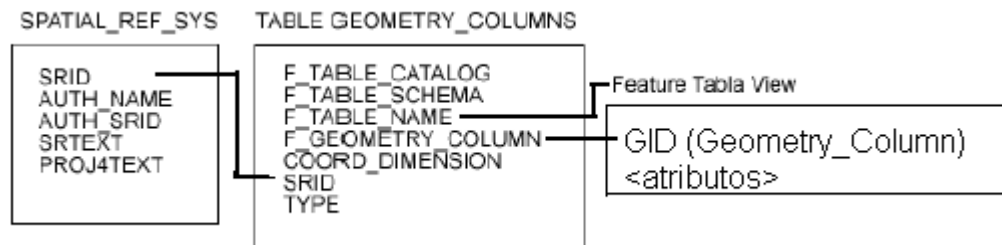


Figura 1. Esquema de las relaciones que existen entre las tablas de Metadatos y cada uno de las tablas de la base de datos con campos geométricos

ANEXO F. MANUAL DEL PROGRAMADOR

TABLA DE CONTENIDO

INTRODUCCION	149
1. COMANDO SHP2PGSQL	149
2. CREACIÓN DE UNA TABLA ESPACIAL	150
3. CONEXIÓN MAPSERVER CON POSTGIS	154
3. CONEXIÓN PHP CON MAPSERVER	155
4. CONEXION DE PHP CON POSTGRESQL	156
6. CONSULTAS ESPACIALES	157
7. DIRECCIONADOR	160

INTRODUCCION

En el presente documento se realiza una explicación de las instrucciones básicas que se llevaron a cabo para el desarrollo del prototipo, la conversión de archivos shapes a PostGIS, la construcción de la Base de datos espacial, las diferentes conexiones entre PostGIS, MapServer, PHP y PostgreSQL, la construcción del direccionador y realización de las consultas espaciales y uso de las funciones PHP/Mapscript para la interacción con el archivo Mapfile.

1. COMANDO SHP2PGSQL

El comando: “shp2pgsql” convierte archivos de figuras ESRI a SQL para su inserción en una base de datos PostGIS/PostgreSQL. Este comando se invoca desde la consola del sistema e importan tanto datos espaciales como temáticos (.dbf)

El comando de importación crea un único fichero .sql donde se incluyen todas las sentencias SQL necesarias para la creación de la tabla con su respectiva columna geométrica y la carga de cada uno de los registros

Sintaxis del comando shp2pgsql:

```
shp2pgsql [nombre.shp] [nombre_tablaEspacial] > [nombre.sql]
```

- [nombre.shp]: indica el nombre del archivo shape (*.shp), se puede especificar la ruta donde se encuentra, por defecto la ruta del archivo es la carpeta bin de PostgreSQL
- [nombre_tablaEspacial]: indica como se llamara la tabla de la base de datos.

- [nombre.sql]: Es el nombre que se le da al archivo donde se guardaran las instrucciones SQL generadas por el comando shp2pgsql, contiene toda la información del mapa en formato SQL.

Ejemplo del uso del comando shp2pgsql

Se tiene el shape de Barrios, entonces los tres archivos ESRI a convertir (barrios.shp, barrios.shx, barrios.dbf) se colocan previamente en la carpeta bin de PostgreSQL y desde una ventana de D.O.S. se ejecuta la siguiente sentencia:

```
shp2pgsql barrios.shp barrios > barrios.sql
```

Después de ejecutarla se observa que en la carpeta bin de PostgreSQL se crea un archivo SQL barrios.sql donde se encuentran las sentencias de la creación de la tabla 'barrios' con su respectiva columna geométrica 'the_geom', y las sentencias para la inclusión de cada uno de los registros de la tabla.

2. CREACIÓN DE UNA TABLA ESPACIAL

Las sentencias SQL para la creación de la tabla espacial, teniendo en cuenta el archivo generado con el comando shp2pgsql para el ejemplo anterior 'barrios.sql' son:

```
CREATE TABLE "barrios" (gid serial PRIMARY KEY,  
"codigo" numeric,  
"nombre" varchar(30),  
"area" numeric,  
"perimeter" numeric,  
"comuna" varchar(50));  
SELECT AddGeometryColumn('','barrios','the_geom','-1','MULTIPOLYGON',2);
```

La sentencia: `SELECT AddGeometryColumn(' ', 'barrios', 'the_geom', '-1', 'MULTIPOLYGON', 2);` Es la que crea la columna geométrica.

AddGeometryColumn es la función PostGIS que se encarga de añadir el campo de geometría del tipo seleccionado a una determinada tabla, también de actualizar la información de los metadatos contenidos en la tabla GEOMETRY_COLUMNS y añadir la restricción de tabla del tipo de la geometría en el campo correspondiente.

Sintaxis de la función AddGeometryColumn

AddGeometryColumn(<db_name>, <table_name>, <column_name>, <srid>, <type>, <dimension>);

- <db_name> nombre de la BD; en el ejemplo: ' ' por defecto donde se ejecute el archivo, no lo liga a una sola base de datos.
- <table_name> nombre de la tabla, nombre que se le dio en la sentencia usada con el comando shp2pgsql; en el ejemplo barrios.
- <column_name> nombre de la columna espacial; Por defecto siempre la llama 'the_geom'.
- <srid> Valor entero que identifica el sistema de referencia espacial. Este valor referencia un valor en la tabla SPATIAL_REF_SYS. En el ejemplo '-1', valor que coloca por defecto que quiere decir que es 'plano'.
- <type> tipo de objeto espacial; El tipo debe ser una cadena en mayúsculas que indica el tipo de geometría, como, 'POLYGON' o 'MULTILINESTRING'.; en el ejemplo 'MULTILINESTRING'
- <dimension> dimensión del mapa, en el ejemplo 2D

Las sentencias SQL para la inserción de los datos alfanuméricos y espaciales en la tabla 'barrios' en el archivo 'barrios.sql' depende de cómo se generaron los datos geométricos ya sea en como (WKT) Well-know text. Ó (WKB) Well-know

binary

- Si fue generado como WKT la sentencia sql seria

```
INSERT INTO "barrios" ("codigo","nombre","area","perimeter","comuna",the_geom)
VALUES ('137.00000','ROSALTA','7155.14900','423.79500',
GeometryFromText("MULTIPOLYGON(((104196.046875
82519.34375,104216.6796875 82535.3125,104244.0078125
82500.875,104251.1171875 82490.9140625,104211.2734375
82460.7421875,104216.3984375 82442.2421875,104206.1484375
82419.7578125,104192.4765625 82426.015625,104170
82444.515625,104143.53125 82478.671875,104196.046875 82519.34375))))"
,-1));
```

La función GeometryFromText, genera un 'string espacial' a partir de las coordenadas y el tipo de datos que recibe. Es la función para insertar los datos espaciales en la columna respectiva.

Sintaxis de la función GeometryFromText:

GeometryFromText (<type>,<srid>)

- <type> tipo de objeto espacial; El tipo debe ser una cadena en mayúsculas que indica el tipo de geometría, como, 'POLYGON' o 'MULTILINESTRING'. Aquí se especifica también las coordenadas geográficas del objeto.
- <srid> Valor entero que identifica el sistema de referencia espacial. Este valor referencia un valor en la tabla SPATIAL_REF_SYS. En el ejemplo '-1', valor que coloca por defecto que quiere decir que es 'plano'.
- Si fue generado como WKB la sentencia sql seria:

```
INSERT INTO "barrios" ("codigo","nombre","area","perimeter","comuna",the_geom)
VALUES ('137.00000','ROSALTA','7155.14900','423.79500',
,'01','0106000000010000000103000000010000001100000000000060AB7FF9400
```

```

00000A0C2F4F440000000805D7FF94000000040A0F6F440000000E0A182F9400
0000060FFF6F440000000C06084F940000000A0FEF5F44000000080C085F9400
00000E0B1F4F440000000A00C86F94000000060CCF1F44000000000E985F940
00000A0DDEEF440000000A01D83F940000000E0ADEEF440000000402783F94
00000080C1EFF440000000201483F940000000E0D5F1F44000000080BE82F94
000000408BF3F440000000809882F940000000E019F4F440000000803982F940
0000006095F4F440000000E0AA81F9400000000EBF4F440000000601C81F940
0000000FEF4F440000000202580F9400000000D8F4F44000000060AB7FF940
000000A0C2F4F440');

```

Cuando se ejecuta las sentencias SQL generadas por el comando shp2pgsql la tabla que se genera en la base de datos es la siguiente:

gid [PK] serial	codigo numeric	nombre character var	area numeric	perimeter numeric	comuna character var	the_geom geometry	barr_comuna numeric
1	137.00000	ROSALTA	7155.14900	423.79500	Comuna Norte	01060000000010000000	1
2	102.00000	COLORADOS	149787.75000	1542.26600	Comuna Norte	01060000000010000000	1
3	103.00000	EL ROSAL	90531.23400	2121.84100	Comuna Norte	01060000000020000000	1
4	101.00000	CAFE MADRID	457546.84400	3383.90100	Comuna Norte	01060000000010000000	1
5	105.00000	LAS HAMACAS	18568.88100	557.00200	Comuna Norte	01060000000010000000	1
6	118.00000	ALTOS DEL KENI	13008.74200	509.34300	Comuna Norte	01060000000010000000	1
7	130.00000	OLAS BAJAS	48158.64100	873.22400	Comuna Norte	01060000000010000000	1
8	2.00000	KENNEDY	222421.17200	2400.95100	Comuna Norte	01060000000010000000	1
9	123.00000	VILLA ROSA	222767.51600	2645.98200	Comuna Norte	01060000000010000000	1
10	136.00000	BOSCONIA	43092.19900	1009.47500	Comuna Norte	01060000000010000000	1
11	127.00000	VILLA HELENA	96627.21100	1439.81900	Comuna Nororie	01060000000010000000	2

Los atributos con los que se contaban en la tabla barrios.dbf antes de convertir el shape a postgis son:

Shape	Codigo	Nombre	Area	Perimeter	Comuna	barr_comuna
Polygon	37.00000	ROSALTA	155.14900	423.79500	Comuna Norte	01
Polygon	02.00000	COLORADOS	787.75000	1542.26600	Comuna Norte	01
Polygon	03.00000	EL ROSAL	531.23400	2121.84100	Comuna Norte	01
Polygon	01.00000	CAFE MADRID	546.84400	3383.90100	Comuna Norte	01
Polygon	05.00000	LAS HAMACAS	568.88100	557.00200	Comuna Norte	01
Polygon	18.00000	ALTOS DEL KENI	1008.74200	509.34300	Comuna Norte	01
Polygon	30.00000	OLAS BAJAS	158.64100	873.22400	Comuna Norte	01
Polygon	2.00000	KENNEDY	421.17200	2400.95100	Comuna Norte	01
Polygon	23.00000	VILLA ROSA	767.51600	2645.98200	Comuna Norte	01

Se observa que se adiciono un campo de tipo geométrico 'the_geom' que es el que contiene la información geográfica y se genera una llave primaria 'gid'.

De igual forma al ejecutar las sentencias SQL se actualiza la tabla de metadatos GEOMETRY_COLUMNS, adicionando el registro correspondiente a la nueva tabla espacial creada.

oid	f_table_catalog [PK] character varying(30)	f_table_schema [PK] character varying(30)	f_table_name [PK] character varying(30)	f_geometry_type [PK] character varying(30)	coord_dimension integer	srid integer	type character varying(30)
18971	"	public	barrios	the_geom	2	-1	MULTIPOLYGON

3. CONEXIÓN MAPSERVER CON POSTGIS

Una vez creada la base de datos con sus respectivas tablas espaciales se hace la conexión de PostGIS con MapServer para que muestre en pantalla los layers trayendo los datos directamente de la base de datos.

Esta conexión se realiza en el archivo base de MapServer el Mapfile²⁸ que aparte de definir el tipo de datos a usar, determina la apariencia y comportamiento del mapa como se va a desplegar en la Web.

En el objeto Layer del Mapfile se define el escenario de datos para acceder a los datos espaciales almacenados en la base de datos, su conexión con postgis se realiza dentro de la secciones CONNECTIONTYPE, CONNECTION y DATA de la la siguiente forma:

```
CONNECTIONTYPE POSTGIS
CONNECTION "host=localhost dbname=inmobiliaria user=postgres
```

²⁸ Ver anexo B

```
password=***** port=5432"  
DATA "the_geom from barrios"
```

En la sección CONNECTION, la línea de conexión a PostGIS es básicamente una conexión a PostgreSQL, de la siguiente forma: "user=**** password=**** dbname=dbname host=localhost port=5432"

El parámetro DATA debe ser de la forma: "<nombre_columna> from <nombre_tabla>", donde <nombre_columna> es el nombre de la columna que contiene la geometría de los objetos y <nombre_tabla>" es el nombre de la tabla desde donde se lee la columna geométrica.

Al cargar el modulo PHP/Mapscript se puede romper con la rigidez del archivo .map ya que se puede acceder a el para modificar, cambiar las capas que se deseen, hacer acercamientos o consultar datos relevantes

3. CONEXIÓN PHP CON MAPSERVER

En el archivo de PHP se carga el modulo PHP/MapScript que permitirá acceder a todas las funcionalidades MapScript de MapServer disponibles en ambiente PHP. También se llama el archivo Mapfile para poder desplegarlo en pantalla y poder interactuar con los objetos definidos en él.

// Cargando el modulo PHP/MapScript, verificando si la extensión ya existe por defecto.

```
if (!extension_loaded("MapScript"))  
    { dl("php_mapscript.so"); }
```

La función, *extension_loaded()* usa el nombre interno de extensión para probar si

una extensión determinada está disponible o no. La sintaxis es `extension_loaded (string $nombre)` y retorna un valor TRUE si la extensión identificada por *nombre* está cargada, FALSE de lo contrario. La sentencia `dl("php_mapscript.so");` carga la librería PHP/MapScript.

```
// Llamado al archive Mapfile
$gpoMap = ms_newMapObj("bmanga.map");
```

La función `ms_newMapObj (string map_file_name)` accede al archivo Mapfile de MapServer definido dentro del paréntesis, asignando en este caso, a la variable `$gpoMap` el objeto mapa con todas las características y objetos especificados dentro del Mapfile 'bmanga.map'. La función debe recibir al menos el parámetro de la ubicación del archivo MapFile.

4. CONEXION DE PHP CON POSTGRESQL

La conexión es la que se encarga de permitir el acceso a la información almacenada en la base de datos para visualizarla, modificarla o eliminarla según se haga necesario. Para abrir la conexión a la Base de datos de PostgreSQL se realiza con el siguiente código PHP.

```
$connection = pg_connect ("host=$host dbname=$db user=$user
password=$pass")
```

Los parámetros necesarios para acceder a la base de datos en PostgreSQL son:

```
$host = "localhost";
$user = "postgres";
$pass = "*****";
```

La cadena de conexión se encuentra en dos archivos denominados

“conexion.php” y “config.php” que están ubicados en:

- La carpeta raíz del proyecto
- inmobiliaria/Connections

Para cerrar La conexión a la Base de datos de PostgreSQL se realiza con el siguiente código PHP:

```
pg_close ();
```

5. CONSULTAS ESPACIALES

Para la realización del prototipo se especificaron consultas espaciales donde se apreciara la combinación de operaciones entre distintas geometrías, entre puntos, puntos con líneas, puntos con polígonos, líneas, líneas con polígono y entre polígonos. Estas combinaciones se explican a continuación.

a) Operaciones entre geometrías de puntos

- En un radio determinado (200 mts, 300 mts, 1 km, etc) del inmueble seleccionado, qué Bancos, Iglesias, Entidades de Salud, colegios o entidades culturales se encuentran.

Para la realización de esta consulta se utilizó la función Postgis ‘distance’. Un ejemplo de la utilización de la función distance en el caso de búsqueda de colegios que hay en un radio de 500 mts de un inmueble:

```
SELECT DISTINCT e.nombre FROM educacion e, inmueble i WHERE  
distance(i.the_geom, e.the_geom) < 500 AND i.gid=1;
```

La misma consulta se haría para el caso de buscar las entidades de salud, culturales, religiosas o bancos, solo cambiaría el nombre de la tabla educación.

b) Operaciones entre geometrías de puntos y polígonos

A que barrio, comuna o estrato pertenece el inmueble seleccionado.

Para la realización de esta consulta se utilizó la función Postgis 'contains'. Un ejemplo de la utilización de la función contains en el caso de conocer en que barrio se encuentra el barrio seleccionado:

```
SELECT b.nombre FROM inmueble i, barrios b WHERE contains  
(b.the_geom, i.the_geom)='t' AND i.gid=1;
```

La misma consulta se haría para el caso de buscar en que comuna o estrato se encuentra el inmueble seleccionado, solo cambiaría el nombre de la tabla barrio.

Cuantos inmuebles se encuentran en determinado barrio, comuna o estrato.

```
SELECT i.inm_dir FROM barrios b, inmueble i WHERE contains  
(b.the_geom, i.the_geom)='t' AND b.nombre='QUINTA BRIDADA';
```

La misma consulta se haría para el caso de buscar cuantos inmuebles de encuentran en determinada comuna o estrato, solo cambiaría el nombre de la tabla barrios.

c) Operaciones entre geometrías de puntos y líneas

➤ Mínima distancia que hay de determinada vía a un inmueble.

Para la realización de esta consulta se utilizó la función Postgis 'distance'. Un ejemplo de la utilización de esta función:

```
SELECT min(distance(i.the_geom, v.the_geom)) FROM inmueble i, vias v
```

```
WHERE v.nombre= 'Autopista Floridablanca' AND i.gid = 1;
```

d) Operaciones entre geometrías de líneas y polígonos

- Vías principales con las que cuenta el barrio donde se encuentra ubicado el inmueble.

Para la realización de esta consulta se utilizó la función Postgis 'intersects'. Un ejemplo de la utilización de esta función:

```
SELECT DISTINCT t2.nombre FROM barrios t1, vias t2 WHERE intersects(t1.the_geom, t2.the_geom)='t' AND t1.nombre= 'QUINTA BRIDADA';
```

e) Operaciones entre geometrías de líneas.

- Distancia que hay de la vía donde esta ubicado el inmueble a determinada vía principal.

Para la realización de esta consulta se utilizó la función Postgis 'distance'. Un ejemplo de la utilización de esta función:

```
SELECT v.nombre, distance(m.the_geom, v.the_geom) FROM mallas m, vias v WHERE v.gid= 1 AND m.gid= 1;
```

f) Operaciones entre geometrías de polígonos.

- Que barrios rodean al barrio donde se encuentra ubicado el inmueble.

Para la realización de esta consulta se utilizó la función Postgis 'touches'. Un ejemplo de la utilización de esta función:

```
SELECT b2.nombre FROM barrios b1, barrios b2 WHERE touches (b1.the_geom, b2.the_geom)='t' AND b1.gid=55;
```

A que comuna pertenece el barrio donde se encuentra ubicado el inmueble.

Para la realización de esta consulta se utilizó la función Postgis 'within'. Un ejemplo de la utilización de esta función:

```
SELECT t1.nombre_com FROM comunas t1 , barrios t2 WHERE within(t2.the_geom, t1.the_geom)='t' AND t2.nombre='LAS AMERICAS';
```

6. DIRECCIONADOR

En el prototipo, para que el administrador pueda crear un nuevo inmueble a partir de una dirección dada y almacenarlo en la base de datos en general se hizo el siguiente análisis:

Dada una dirección se realiza la intersección entre las dos líneas, por ejemplo calle con carrera y luego se mide desde ese punto el número de metros, ya sea sobre la calle o la carrera (dependiendo de la dirección), que se desee obtener.

Por ejemplo para hallar la geometría de un nuevo inmueble ubicado en la dirección Calle 4 No. 4 – 50 se siguen los siguientes pasos:

- a) Intersectar la calle cuarta con carrera cuarta (intersección entre dos líneas), utilizando la función PostGis intersection (geometry, geometry)

```
SELECT intersection(t1.the_geom,t2.the_geom) FROM malla_vial t1, malla_vial t2 WHERE t1.gid=4 and t2.gid=14;
```

Esta operación da como resultado la geometría de un punto (x,y), que corresponde al punto de intersección.

- b) Hallar la longitud de la calle 4, que es donde se ubicara el nuevo inmueble, según la dirección dada, utilizando la función PostGis length(geometry).

```
SELECT length(the_geom) FROM malla_vial WHERE gid=4;
```

Esta operación da como resultado la longitud en metros de la calle 4.

- c) Hallar la fracción de línea que hay hasta el punto de intersección de la calle 4, línea donde se va a colocar el nuevo punto con la función line_locate_point(geometry, geometry).

```
SELECT line_locate_point(geometryN(the_geom,1), GeometryFromText(' POINT
(416260.53987746 3770990.25156583)',-1))
FROM malla_vial WHERE gid = 4;
```

Para utilizar esta función se tiene en cuenta el resultado obtenido en el paso a, que es el punto de intersección, luego esta operación se estaría haciendo entre un punto y una línea, arrojando como resultado un numero entre 0 y 1, que equivale a la fracción de línea donde esta ubicado el punto de intersección.

- d) Hallar la fracción de la línea que hay hasta donde se va a colocar el nuevo punto. Para esto se debe sumar al resultado del paso c los metros que se especifican en la dirección, estos metros que se suman deben estar dados también en porcentaje con respecto a la calle donde se va a colocar el punto. Este paso solamente se utilizan operaciones matemáticas básicas de suma y división y arroja como resultado un número entre 0 y 1.
- e) Hallar la geometría (x,y) del nuevo inmueble. Con la función line_interpolate_point(geometry, float8)

```
SELECT      (line_interpolate_point(geometryN(the_geom,1),a))      FROM
malla_vial WHERE gid=4;
```

En esta sentencia SQL se tiene en cuenta el resultado del paso d (numero entre 0 y 1), que en el ejemplo es 'a'. Y da como resultado la geometría de un punto que corresponde a la nueva dirección que se adicionara a la base de datos.

7. FUNCIONES PHP/MAPSCRIPT

Entre las funciones PHP/Mapscript utilizadas para manipular el Mapfile con sus respectivos layers a través de la página Web se encuentran:

a) `numlayers` : int.

Un ejemplo de su uso es: `$gpoMap->numlayers`.

La variable `$gpoMap` es de tipo objeto mapa y contiene todas los objetos y características definidos en el Mapfile 'bumanga.map'; la función `numlayers` devuelve el numero de layers que se encuentran en este objeto, devuelve un numero entero.

b) `GetLayer(int)`

Un ejemplo de su uso es: `$gpoMap->GetLayer($i)`;

La función `GetLayer(int)` Devuelve un objeto layer del archivo Mapfile dado un valor de índice (0=el primer layer).

c) `getNumResults()` : int

Un ejemplo de su uso es: `$oLayer->getNumResults()`;

La variable `$oLayer` es de tipo objeto layer. La función `getNumResults()` devuelve el numero de resultados que tuvo la variable de tipo layer en la ultima consulta.

d) `getResult(int)`

Un ejemplo del uso de esta función es: `$oLayer->getResult($i)`.

La función `getResult()` toma la información que tenga el objeto `layer` dado un valor de índice (0=el primer `layer`).

e) `shapeindex`

Un ejemplo de su uso es `$oRes->shapeindex`;

La variable `$oRes` es de tipo objeto `layer`. La función `shapeindex`; recupera el índice geométrico de un `layer`.

f) `getlayerbyname(nombre_layer)`;

Un ejemplo de su uso es `$gpoMap->getlayerbyname(barrios)`;

La función `getlayerbyname(nombre_layer)` retorna un objeto `layer` a partir del parámetro donde se ingresa el nombre de la capa.

ANEXO G. NECESIDADES DEL SISTEMA

A continuación se presentan las necesidades identificadas para los usuarios definidos en la herramienta Web propuesta en el presente proyecto.

USUARIO PÚBLICO

- Buscar inmuebles ofrecidos en venta o arriendo por la inmobiliaria, que satisfagan condiciones de su escogencia; esta búsqueda se podría hacer teniendo en cuenta características como: Barrio, Rango de valor del inmueble, Rango de área del inmueble, Tipo de inmueble (Vivienda, Bodega, o Negocio específico), y seleccionar de la lista de inmuebles que dio como resultado la búsqueda.
- Visualizar información espacial de un inmueble seleccionado, localizándolo dentro del mapa de la ciudad.
- Visualizar información alfanumérica del inmueble seleccionado, como valor de venta o del canon de arriendo, disponibilidad o condiciones, áreas, servicios, estado, entre otros.
- Consultar información sobre existencia, distancia y otras características de utilidades cercanas al inmueble seleccionado, como iglesias, bancos, escuelas y hospitales.
- Enviar solicitud de estudio y/o reserva para tomar el inmueble, ya sea para arrendamiento o venta.
- Consultar información general de la inmobiliaria, ubicación, misión, visión, etc.

Los roles Cliente, Asesor Comercial y Administrador del Sistema adicionalmente tienen acceso a funcionalidades anteriores del usuario público.

USUARIO CLIENTE

- Puede realizar las mismas funciones del usuario público.

- Registrarse. El usuario una vez es aprobado como cliente de la inmobiliaria, para ofrecer uno o varios inmuebles en arriendo o venta, puede identificarse y entrar al sistema para poder tener información adicional acerca del inmueble o enviar inquietudes.
- Modificar información del perfil personal, dirección, teléfonos, email.
- Consultar información de interés a cerca del inmueble: estado de cuenta, términos del contrato, tiempo faltante para expiración de contrato, información sobre daños del inmueble.
- Enviar solicitudes o avisos, como daños, y averiguar el estado de tales solicitudes.

USUARIO ASESOR COMERCIAL

- Puede realizar las mismas funciones del usuario público
- Consultar información general de Clientes
- Acceder al formato digital de inscripción de nuevos clientes y características del inmueble para venta o arriendo.
- Enviar información al administrador de formatos de nuevos inmuebles y clientes ya diligenciados.
- Modificar información del perfil personal, dirección, teléfonos, email.
- Informar y consultar sobre gestiones con clientes o futuros clientes, avance de tales gestiones, plazos establecidos, compromisos.

USUARIO ADMINISTRATIVO

- Puede realizar las mismas funciones del usuario público.
- Realizar consultas estadísticas de ventas y arriendos.
- Consultar y contestar las solicitudes o inquietudes que han enviado los usuarios públicos.
- Consultar Informes de las actividades desarrolladas por los asesores comerciales.
- Consultar información general de Clientes.

- Consultar información general de asesores comerciales.
- Modificar información del perfil personal, dirección, teléfonos, etc.

USUARIO ADMINISTRADOR DEL SISTEMA

- Puede realizar las mismas funciones del usuario público
- Consultar, ingresar, modificar o eliminar información sobre los usuarios registrados en el sistema.
- Asignar los respectivos nombre de usuario y contraseña a los diferentes usuarios del sistema.
- Visualizar y habilitar información enviada por el asesor comercial de los formatos de nuevos inmuebles y clientes para que pueda ser vista por los demás usuarios del sistema según su perfil.
- Visualizar, ingresar, modificar o eliminar información alfanumérica y espacial de un inmueble.
- Realizar el mantenimiento de la información general que se maneja en la inmobiliaria.
- Consultar Informes de las actividades desarrolladas por los asesores comerciales.
- Consultar informes estadísticos de ventas y arriendos, filtrando la información solicitada por comuna, barrio, rango de valores, etc.

ANEXO H. MANUAL DE INSTALACION

Prerequisitos

Para poder realizar la insatacion del prototipo en un equipo que sirva como servidor lo primero que debe hacerse es insatalar y configurar las herraminetas software necesarias para que el prototipo funcione correctamente. El prototipo propuesto se desarrollo en el sistema operativo Windows XP y por tanto se instalaron herramientas software compatibles con dicho sistema operativo. Las versiones de las herramientas instalaron son las siguientes

Apache 2.2.4, PHP 5.2.1, Mapserver 4.1.0.1, Mapscript 4.1.0.1 con su respectiva librería php_mapscript.dll, PostgreSQL 8.2 y su extensión espacial PostGIS 1.1.6.

Creacion de la Base de Datos

Una vez ya se encuentren instalados y configuradas estas herramientas se debe crear una base de datos llamada 'inmobiliaria' en el DBMS PostgreSQL; con la ayuda de la herramienta pgAdminIII, que es una herramienta grafica para la administracione base de datos en PostgreSQL, se debe asegurar escojer en 'paIntilla' la opcion 'template_postgis', que es la que asegura que la bd se cree con las tablas de metadatos 'geometry_columns' y 'spatial_ref_sys' como se muestra en la **figura**.

Una vez creada la base de datos llamada 'inmobiliaria' se debe restaurar el archivo "bdinmobiliaria.backup" que se encuentra en el directorio "bdinmobiliaria", este archivo contiene las Instrucciones necesarias para la creación de las tablas y las relaciones de la base de datos del prototipo.

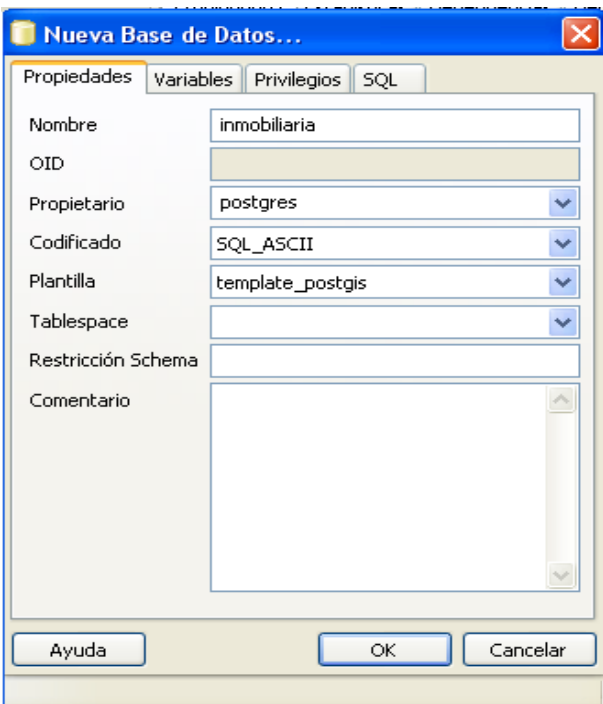


Figura. Creación de la Base de Datos.

Montar el Sitio Web

Para montar el sitio Web se debe copiar el contenido de la carpeta inmobiliaria, que está en el CD que contiene los archivos correspondientes al prototipo Web, en la carpeta htdocs de la carpeta de 'apache' que se encuentra en el caso de la instalación en el SO Windows en el directorio "C:\ms4w\Apache\htdocs"

Crear un Usuario Administrador

Crear un usuario administrador, en la Base de Datos 'Inmobiliaria', en la tabla de 'usuarios' se debe agregar un registro puede ser con la ayuda de la herramienta pgAdminIII se debe crear inicialmente un usuario administrador, para luego si ingresar al prototipo y empezar a poblar la base de datos.