



# **GEPASTO: Un Sistema de Información Geográfica Web para el Manejo Territorial del Municipio de Pasto**

**V2**

## **MANUAL DE USUARIO**



**UNIVERSIDAD DE NARIÑO  
SAN JUAN DE PASTO**

**2018**

## Tabla de contenido

1. ARQUITECTURA DE GEOPASTO.....	6
1.1. ARQUITECTURA DE CAPAS .....	6
1.1.1. Capa de Presentación. ....	7
1.1.2. Capa Kernel.....	7
1.1.3. Capa de Servicios.....	7
1.2. ARQUITECTURA DE PAQUETES.....	7
1.2.1. Paquete Common .....	8
1.2.2. Paquete Data .....	9
1.2.3. Paquete UI .....	14
1.2.4. Paquete Util .....	15
2. DISEÑO DE GEOPASTO.....	18
2.1. DIAGRAMA DE PAQUETES.....	18
2.1.1. Diagrama de Paquetes.....	18
2.1.2. Diagramas de Clases .....	19
2.2. DISEÑO DE LA BASE DE DATOS.....	23
2.2.1 Esquema APP. ....	23
2.2.2 Esquema GIS. ....	25
2.2.3 Esquema PUBLIC.....	27
3. MANEJO DE GEOPASTO.....	30
3.1. INICIO.....	30
3.2. MENU PRINCIPAL.....	31
3.3. MÓDULO CONFIGURACIÓN .....	32
3.3.1. Opción Generales .....	33
3.3.2. Editor de Menús.....	35
3.3.3. Biblioteca de Símbolos.....	39
3.3.4. Convenciones .....	42
3.4. MÓDULO SEGURIDAD .....	45
3.5. MÓDULO CARTOGRAFIA.....	48

3.6. MÓDULO HERRAMIENTAS .....	52
3.6.1. Capas de Información.....	52
3.6.2 Búsqueda Rápida .....	53
3.6.3 Información Útil.....	55
3.6.4 Opciones sobre el mapa .....	57
4. PRUEBAS DE FUNCIONALIDAD.....	58
4.1 Pruebas de Visualización.....	58
4.1.1 Herramientas de navegación .....	58
4.1.2 Herramienta Capas de información.....	61
4.2 Pruebas de Consulta. ....	63
4.2.1 Herramienta Búsqueda Rápida.....	63
4.2.2 Herramienta Información Útil .....	67

## TABLA DE FIGURAS

Figura 1. Arquitectura de GEOPASTO.....	6
Figura 2. Diagrama Paquete Principal .....	18
Figura 3. Diagrama Paquete Class .....	19
Figura 4. Diagrama de Clases Paquete Common .....	20
Figura 5. Diagrama de Clases Paquete Data.....	21
Figura 6. Diagrama de Clases Paquete UI.....	22
Figura 7. Diagrama de Clases Paquete UTIL.....	23
Figura 8. Diagrama entidad relación esquema APP.....	25
Figura 9. Diagrama entidad relación esquema GIS.....	26
Figura 10. Diagrama entidad relación esquema PUBLIC .....	29
Figura 11. Pantalla Iniciar Sesión .....	30
Figura 12. Mensaje de Error .....	30
Figura 13. Ventana Principal de la Aplicación .....	31
Figura 14. Módulo de Configuración .....	32
Figura 15. Pantalla Generales .....	33
Figura 16. Temas.....	34
Figura 17. Pantalla Editor de Menús .....	35
Figura 18. Selección de Módulo.....	36
Figura 19. Listado de Módulos.....	36
Figura 20. Edición del Módulo.....	37
Figura 21. Eliminar Módulo .....	38
Figura 22. Insertar Ítem.....	38
Figura 23. Eliminar Ítem.....	39
Figura 24. Opción Biblioteca de Símbolos .....	39
Figura 25. Pantalla biblioteca de símbolos.....	40
Figura 26. Nuevo Símbolo .....	40
Figura 27. Eliminar Símbolo.....	41
Figura 28. Opción Convenciones.....	42
Figura 29. Pantalla Convenciones .....	43
Figura 30. Nueva Convención.....	44
Figura 31. Eliminar Convención .....	45
Figura 32. Modulo Seguridad.....	45
Figura 33. Lista de Usuarios del Sistema.....	46
Figura 34. Pantalla de Privilegios.....	47
Figura 35. Asignación de Privilegios .....	47
Figura 36. Módulo Cartografía .....	48
Figura 37. Mapa Uso de Suelos.....	49
Figura 38. Mapa de Amenazas .....	50

Figura 39. Barra de herramientas .....	51
Figura 40. Pestaña Capas de Información .....	52
Figura 41. Búsqueda Rápida .....	53
Figura 42. Datos de Búsqueda .....	54
Figura 43. Resultado de la Búsqueda .....	54
Figura 44. Información Útil .....	55
Figura 45. Resultado de Consulta .....	56
Figura 46. Cuadro explicativo de Ayuda .....	56
Figura 47. Opciones sobre el mapa .....	57
Figura 48. Estado inicial del mapa .....	58
Figura 49. Resultado de la herramienta acercar .....	59
Figura 50. Resultado de la herramienta alejar .....	59
Figura 51. Resultado de la herramienta alejar .....	60
Figura 52. Resultado herramienta desplazamiento .....	61
Figura 53. Resultado de la activación de la capa comunas .....	61
Figura 54. Visualización de ítems de la capa comunas .....	62
Figura 55. Resultado visualización de ítems activos de la capa comunas .....	62
Figura 56. Panel de búsqueda rápida .....	63
Figura 57. Resultado ejecución de la búsqueda .....	64
Figura 58. Ampliación ventana con los resultados de la búsqueda .....	64
Figura 59. Ubicación de un registro del resultado de la consulta en el mapa .....	65
Figura 60. Visualización de todos los resultados de la consulta .....	65
Figura 61. Ampliación del panel de búsqueda rápida .....	66
Figura 62. Representación de los resultados de la búsqueda como una nueva capa de datos .....	66
Figura 63. Ampliación del panel de capas de información .....	67
Figura 64. Panel de información útil .....	67
Figura 65. Consultas especializadas disponibles en el mapa .....	68
Figura 66. Ventana de ayuda para la ejecución de consultas especializadas .....	68
Figura 67. Ampliación de la ventana de ayuda .....	69
Figura 68. Resultado de la ejecución de la consulta en el predio de mayor tamaño .....	69
Figura 69. Ampliación de la ventana con la respuesta a la consulta realizada .....	70
Figura 70. Resultado de la consulta que no depende de un punto específico .....	70
Figura 71. Ampliación de la ventana con el resultado de la consulta .....	71

# 1. ARQUITECTURA DE GEOPASTO

## 1.1 ARQUITECTURA DE CAPAS

GEOPASTO es un sistema de información geográfica web para el manejo territorial del municipio de Pasto. Este sistema fue desarrollado bajo los sistemas operativos Microsoft Windows y Linux OpenSuse, utilizando los lenguajes de programación PHP, HTML y JavaScript. Como gestor de base de datos fue utilizado PostgreSQL y su extensión para manejo de datos espaciales PostGIS. Finalmente como servidor de mapas fue utilizado MapServer, el cual interpreta los datos espaciales y facilita su publicación en la Web.

La arquitectura de GEOPASTO está compuesta por tres capas principales: *capa de presentación*, *capa kernel o núcleo* y *capa de servicios*, como se muestra en la figura 1.

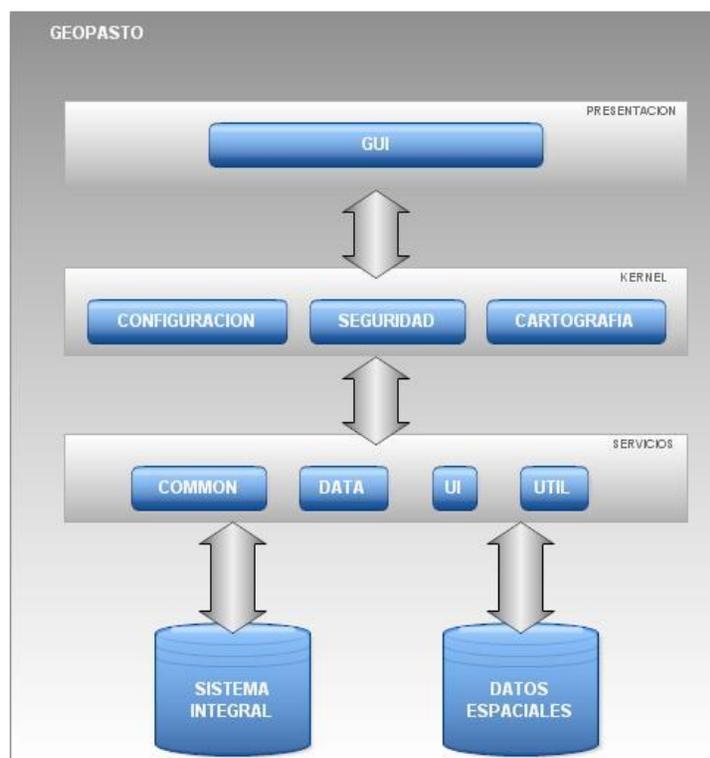


Figura 1. Arquitectura de GEOPASTO

### **1.1.1. Capa de Presentación.**

Conformada por el modulo de GUI o Interfaz Grafica de Usuario, el cual es el encargado de proporcionar la forma de interacción con el usuario final, de tal modo que este pueda realizar todas sus tareas en el sistema y satisfacer sus requerimientos.

### **1.1.2. Capa Kernel**

En esta capa se llevan a cabo los procesos específicos que constituyen la funcionalidad del sistema, procesos como edición de menús, gestión de privilegios de acceso al sistema por parte de los distintos tipos de usuarios, etc. Esta conformada por los módulos de *configuración, seguridad y cartografía*. El modulo de configuración, es el que permite la configuración de de variables necesarias para el funcionamiento del sistema. El modulo de seguridad, encargado de la administración de los privilegios de acceso de cada uno de los usuarios del sistema, y el modulo cartografía, en donde residen los mapas que conforman el sistema de información geográfica (SIG).

### **1.1.3. Capa de Servicios**

En esta capa se agrupan todos los servicios del sistema tales como: el acceso a bases de datos, el motor de plantillas necesarias para construir la interfaz de usuario, el servicio que facilita el uso de Ajax dentro de la herramienta, y el servicio que permite la comunicación con el servidor de mapas MapServer el cual facilita la publicación de cartografía en la web Conformada por el paquete *Common* que proporciona las funcionalidades del núcleo de la aplicación, tales como: construcción de la GUI, el acceso a bases de datos, etc. El paquete *Data* que contiene la definición del acceso a datos usando el patrón *ActiveRecord*. El paquete *UI* que contiene las clases encargadas de la construcción de la GUI y el paquete *Util* que contiene las clases responsables de la construcción de mapas, que finalmente constituye la característica más importante del sistema.

## **1.2 ARQUITECTURA DE PAQUETES**

La arquitectura de GEOPASTO se implementa en los siguientes paquetes:

- *Paquete Common:* dentro de este paquete se encuentran todas las clases base que proporcionan las funcionalidades primitivas del sistema, tales como: conexión a bases de datos, motor de plantillas para la interfaz de usuario, manejo de sesiones e integración de la aplicación.

- *Paquete Data*: en este paquete se agrupan todas las definiciones de las clases que encapsulan el acceso a bases de datos. Utilizando el patrón de diseño orientado a objetos conocido como *Active Record*.
- *Paquete UI*: este paquete contiene las clases responsables de la construcción de la interfaz gráfica de cada uno de los módulos de la aplicación.
- *Paquete Util*: dentro de este paquete se encuentran las clases que proporcionan la comunicación con el servidor de mapas, así como la base para la construcción de los mismos.

### 1.2.1 Paquete Common. Dentro de este paquete se encuentran las siguientes clases:

*AppSQLSettings*: clase utilizada para la configuración de las conexiones con el sistema gestor de bases de datos. Esta clase provee el manejo de información sobre una conexión, nombre del Host o servidor donde se encuentra el gestor de bases de datos, nombre de usuario de la base de datos, contraseña del usuario, nombre de la base de datos y el Driver o controlador que especifica el tipo de servidor de bases de datos, que para nuestro caso es PostgreSQL.

*AppSQL*: clase que provee la conexión con el sistema gestor de bases de datos implementando el patrón de diseño orientado a objetos conocido con el nombre de *Singleton*, el cual garantiza una única instancia para la conexión con la base de datos y un acceso global a esta. Contiene el siguiente método:

- *getInstance()* : construye y retorna una instancia del objeto que facilita las operaciones realizadas sobre la base de datos.

*AppActiveRecord*: clase que provee métodos de acceso para cada uno de los atributos de una tabla de la base de datos y la encapsulación de los mismos, de manera que se asocian filas únicas con objetos que facilitan la comprensión del proceso de almacenamiento y la lógica del dominio. Contiene los siguientes métodos:

- *getXajaxResponse()*: permite el acceso al objeto de utilizado para el manejo de ajax.
- *toArray()*: serializa todas las propiedades del objeto y las retorna en un array.

*AppTemplates*: clase que proporciona el acceso al motor de plantillas utilizado para construcción de la GUI y la separación de la capa de presentación y la capa de dominio.

*AppPage*: clase abstracta responsable de la base para la construcción de las diferentes GUI de los módulos de GEOPASTO. Contiene los siguientes métodos:

- *getXajaxResponse()*: permite el acceso al objeto de utilizado para el manejo de ajax.
- *renderTemplate()*: procesa una plantilla que contiene código HTML reemplazando variables dentro de esta y retornando el resultado para ser manipulado.
- *Index()*: punto de entrada para cada una de las interfaces de GUI, encargado de desplegar la pagina principal de cada modulo.

*AppSession*: clase encargada de controlar la sesión de los usuarios en el sistema y su seguimiento dentro del mismo. Contiene los siguientes métodos:

- *startSession()*: inicia el manejo de sesiones aplicando los valores de seguridad correspondientes a la configuración del sistema.
- *setData()*: almacena los datos de seguimiento del usuario una vez haya ingresado satisfactoriamente al sistema:
- *getData()*: retorna los datos de seguimiento del usuario.
- *isValid()*: verifica si la sesión del usuario aun está activa.
- *destroy()*: finaliza el manejo de sesiones del sistema.

*AppHome*: clase controlador responsable de la inicialización e integración de todos los módulos que componen el sistema, constituye el punto de entrada a la aplicación. Contiene los siguientes métodos:

- *DisplayLayout()*: construye el marco de la página principal de la aplicación mediante el uso del motor de plantillas.
- *DisplayWelcome()*: construye el mensaje de bienvenida para cuando el usuario haya ingresado satisfactoriamente al sistema.
- *exec()*: encargado de inicializar cada uno de los módulos y delegar las acciones correspondientes a estos.
- *isValidAction()*: verifica que el modulo y la acción que se desea ejecutar estén disponibles.

**1.2.2 Paquete Data.** Dentro de este paquete se encuentran las siguientes clases:

*Config*: clase que se encarga de las operaciones sobre la tabla que almacena los valores de configuración del sistema. Contiene los siguientes métodos:

- *getAllValues()*: recupera todos las variables de configuración del sistema.
- *getByKey()*: recupera una variable de configuración por su identificador.
- *setValue()*: almacena el valor de una variable de configuración.

*AppModules*: clase encargada de las operaciones sobre la tabla que almacena los módulos que componen GEOPASTO. Contiene los siguientes métodos:

- *getAllModules()*: recupera todos los registros de la tabla.
- *saveModule()*: almacena un nuevo registro en la tabla.
- *updateModule()*: actualiza un registro en la tabla.
- *deleteModule()*: elimina un registro de la tabla.
- *toJson()*: serializa las propiedades del objeto en formato JSON.
- *getMenus()*: recupera todos los menús asociados a un modulo en particular.
- *isAllowed()*: comprueba si un usuario tiene los privilegios para acceder a un modulo de la aplicación.

*AppModuleMenus*: clase encargada de las operaciones sobre la tabla que almacena los menús correspondientes a cada uno de los módulos que componen GEOPASTO. Contiene los siguientes métodos:

- *isAllowed()*: comprueba si un usuario tiene los privilegios para acceder a un ítem del menú correspondiente a un modulo en especial.
- *saveItem()*: almacena un nuevo registro en la tabla.
- *updateItem()*: actualiza un nuevo registro en la tabla.
- *deleteItem()*: elimina un registro de la tabla.

*AppMenu*: esta clase se encarga de recuperar los módulos y sus respectivos ítems del menú a los cuales un usuario tiene privilegios para acceder. Contiene los siguientes métodos:

- *getModules()*: recupera los datos de los módulos a los cuales un determinado usuario puede acceder.
- *getMenuTree()*: construye una estructura de árbol correspondiente a los módulos y sus menús, dependiendo de los privilegios de un usuario específico.

*Permisos*: clase encargada de las operaciones sobre la tabla que almacena los privilegios de cada uno de los usuarios del sistema. Contiene los siguientes métodos:

- *getSecurityTree()*: construye una estructura de árbol correspondiente a aquellos módulos y menús a los cuales un usuario tiene acceso.
- *removeAll()*: elimina todos los privilegios de un usuario determinado.
- *setRight()*: almacena los privilegios de un usuario determinado.
- *getRights()*: recupera los privilegios de un usuario determinado.

*Usuarios*: clase encargada de las operaciones sobre la tabla que almacena los usuarios del sistema. Contiene los siguientes métodos:

- *Encriptar()*: encripta la contraseña que el usuario utiliza para autenticarse en el sistema.
- *Desencriptar()*: desencripta la contraseña que el usuario utiliza para autenticarse en el sistema.
- *doLogin()*: realiza la autenticación de un usuario en el sistema.
- *doLogout()*: finaliza de manera segura la sesión de un usuario.
- *getAllUsers()*: recupera todos los usuarios del sistema.

*Simbolos*: clase encargada de las operaciones sobre la tabla que almacena los datos de los símbolos utilizados para definir las convenciones en los mapas de la aplicación. Contiene los siguientes métodos:

- *getAll()*: recupera todos los registros de la tabla.
- *addSym()*: crea un nuevo registro en la tabla.
- *updateSym()*: actualiza un registro en la tabla.
- *deleteSym()*: elimina un registro de la tabla.

*Convenciones*: clase encargada de las operaciones sobre la tabla que almacena los datos de las convenciones correspondientes a cada una de las capas de los mapas de la aplicación. Contiene los siguientes métodos:

- *getAll()*: recupera los registros correspondientes a un mapa y una capa en particular.
- *toArray()*: serializa las propiedades del objeto en un array.
- *add()*: almacena un nuevo registro en la tabla.
- *modify()*: actualiza un registro en la tabla.
- *remove()*: elimina un registro de la tabla.

*Perimetro*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes al perímetro urbano del municipio de San Juan de Pasto. Contiene el siguiente método:

- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*Zonas*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a las zonas en las que se divide el municipio de San Juan de Pasto. Contiene el siguiente método:

- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*AreasHomogeneas*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a las áreas morfológicas homogéneas en las que se divide el municipio de San Juan de Pasto. Contiene el siguiente método:

- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*Comunas*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a las comunas en las que se divide el municipio de San Juan de Pasto. Contiene el siguiente método:

- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*Predios*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a los predios existentes en el municipio de San Juan de Pasto.

*Ríos*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a los principales ríos del municipio de San Juan de Pasto.

*UsosSuelos*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes al uso de suelo de los predios del municipio de San Juan de Pasto.

Contiene los siguientes métodos:

- *getAreaActividad()*: recupera el nombre del área de actividad (uso del suelo) de un determinado predio.
- *getSiglaArea()*: recupera la sigla que representa el área de actividad.
- *getTotalAreaByActividad()*: calcula mediante análisis espacial el área total en metros cuadrados ( $m^2$ ) correspondiente a una área de actividad específica.
- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*Amenazas*: clase encargada de las operaciones sobre la tabla que almacena los datos espaciales correspondientes a las amenazas presentes en el municipio de San Juan de Pasto. Contiene el siguiente método:

- *getInfoXY()*: recupera la información mediante una consulta espacial en un punto específico.

*SII\_AmenazasPredios*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las amenazas que afectan cada uno de los predios del municipio de San Juan de Pasto.

*SII\_EstadosLotes*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes al estado de loteo de los predios.

*SII\_Estratos*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a la clasificación por estratos de los predios del municipio de San Juan de Pasto.

*SII\_Predios*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a los predios del municipio de San Juan de Pasto.

*SII\_Personas*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a los propietarios de los predios, tanto como a los funcionarios de la Subsecretaría de Planeación del municipio.

*SII\_Propietarios*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes los predios de los cuales una persona es propietario.

*SII\_PotAmenazas*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las amenazas que pueden afectar a un predio determinado.

*SII\_PotAreasActividad*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a la clasificación de los predios del municipio de San Juan de Pasto según la actividad que realizan.

*SII\_PotClaseSuelos*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las diferentes clasificaciones del suelo del municipio de San Juan de Pasto.

*SII\_PotDemarcaciones*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las demarcaciones que se realizan sobre los predios del municipio de San Juan de Pasto.

*SII\_PotPiezasUrbanas*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las piezas urbanas del municipio de San Juan de Pasto.

*SII\_PotSectores*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a los sectores en los que se encuentra dividido el municipio de San Juan de Pasto.

*SII\_PotSubSectores*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a los subsectores en los que se encuentran divididos los sectores del municipio de San Juan de Pasto.

*SII\_PotTipologiaVias*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a las diferentes tipologías de vías que circundan a un determinado predio del municipio de San Juan de Pasto.

*SII\_PotTratamientos*: clase encargada de las operaciones sobre la tabla que almacena los datos correspondientes a los diferentes tratamientos de una zona del municipio que más adelante influyen en su desarrollo y planificación.

**1.2.3 Paquete UI.** Dentro de este paquete se encuentran las siguientes clases:

*ConfigUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente a las variables de configuración del sistema.

*MenuUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente al editor del menú de la aplicación.

*UsuariosUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente a la administración de usuarios del sistema. Contiene el siguiente método:

- *displayLogin()*: despliega la interfaz que el usuario utiliza para el proceso de logeo antes de ingresar al sistema.

*PermisosUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente a la administración de los privilegios de acceso de cada uno de los usuarios del sistema.

*SimbolosUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente a la administración de la biblioteca de símbolos utilizada para definir convenciones en los mapas de la aplicación.

*ConvencionesUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente a la administración de las convenciones de cada uno de los mapas de la aplicación.

*MapaUsoSuelosUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente al mapa de uso de suelos del municipio. Contiene los siguientes métodos:

- *createLayout()*: crea los componentes necesarios para la interacción del usuario con el mapa.
- *doQuery()*: delega la ejecución de una determinada consulta realizada sobre el mapa y posteriormente despliega su resultado.

*MapaAmenazasUI*: clase encargada de la construcción de la interfaz grafica de usuario correspondiente al mapa de amenazas presentes en el municipio. Contiene los siguientes métodos:

- *createLayout()*: crea los componentes necesarios para la interacción del usuario con el mapa.
- *doQuery()*: delega la ejecución de una determinada consulta realizada sobre el mapa y posteriormente despliega su resultado.

**1.2.4 Paquete Util.** Dentro de este paquete se encuentran las siguientes clases:

*msMap*: clase encargada de encapsular el manejo del API de MapServer la cual permite la manipulación de los mapas a través de código de programación. Contiene los siguientes métodos:

- *setWidth()*: asigna el valor del ancho del mapa.
- *setHeight()*: asigna el valor de la altura del mapa.
- *getName()*: retorna el nombre que identifica el mapa.
- *getMsObj()*: retorna el objeto que manipula el mapa directamente del API.
- *getLegend()*: retorna el objeto que representa la leyenda del mapa.
- *getLayer()*: retorna un objeto layer o capa que componen el mapa.
- *getNumLayers()*: retorna el numero de capa que componen el mapa.
- *getMapWidth()*: retorna el ancho del mapa.
- *getMapHeight()*: retorna el alto del mapa.
- *getMapScale()*: retorna la escala actual en el mapa.
- *getWebImagePath()*: retorna la ruta donde son almacenadas la imágenes que son producidas por MapServer.
- *getExtent()*: retorna los valores del rectángulo que determina el área actual de visualización en el mapa.
- *drawMap()*: dibuja el mapa, el cual se almacena en un archivo de imagen y retorna la ruta a esta.
- *drawLegend()*: dibuja la leyenda correspondiente a las capas del mapa que se encuentran activas en un determinado momento y retorna la ruta al archivo de imagen correspondiente.
- *toggleLayer()*: intercambia el estado de una determinada capa entre encendida o apagada.
- *toggleLayerClass()*: intercambia el estado de un ítem de una capa entre encendido o apagado.
- *toggleAllLayerClasses()*: intercambia el estado de todos los ítems de una capa entre encendidos o apagados.
- *processAction()*: realiza todas los cálculos pertinentes para permitir la navegación por el mapa.
- *getAllLayers()*: retorna todas la capas que componen el mapa.
- *getActiveLayers()*: retorna todas las capas que se encuentren visibles en el mapa.
- *getLayerIcons()*: retorna todos los nombres de los ítems que componen una capa y la ruta del archivo de imagen de cada uno de ellos.
- *pixelToGeo()*: convierte una coordenada en pixeles a una coordenada geográfica.
- *getTmpFile()*: retorna la ruta al archivo temporal donde se almacena el estado actual del mapa.
- *saveMapState()*: almacena el estado actual del mapa en un archivo temporal.

*msMapLayout*: clase encargada de construir todos los componentes que utiliza la interfaz grafica de cada mapa y responder a las acciones que realiza el usuario sobre estos. Contiene los siguientes métodos:

- *createLayout()*: construye el marco de trabajo necesario para los diferentes controles del mapa.
- *getTempMap()*: retorna el objeto asociado al archivo temporal del estado actual del mapa.
- *saveTempMap()*: delega el almacenamiento del estado actual del mapa a un archivo temporal.
- *resizeMap()*: responde a la acción de cambio de tamaño del mapa.
- *doAction()*: responde a las acciones de desplazamiento, acercamiento, alejamiento, cambio de estado de las capas y cambio de estado de los ítems de las capas.
- *getLayers()*: construye panel utilizado para la manipulación de las capas del mapa.
- *quickSearch()*: delega la ejecución de búsquedas de acuerdo con la capa y posteriormente despliega su resultado.
- *addSymbols()*: procesa la información de las convenciones correspondientes al mapa y consecuentemente son adicionadas a este.
- *cleanText()*: reemplaza caracteres especiales presentes en los nombres de las capas y sus ítems, particularmente acentos del español.

*PdfMapReport*: clase encargada de exportar el área visualizada actualmente en el mapa a un archivo PDF para su posterior impresión. Contiene los siguientes métodos:

- *setMap()*: asigna el objeto mapa asociado al reporte.
- *getMapObj()*: retorna el objeto mapa asociado al reporte.
- *drawFrame()*: dibuja el marco del formato de impresión.
- *parseImage()*: convierte los archivos de imágenes que representan el mapa, a archivos compatibles con el formato del reporte.
- *addImages()*: adjunta las imágenes correspondientes al logo de la Subsecretaria de Planeación, el mapa y sus respectivas convenciones.

## 2. DISEÑO DE GEOPASTO

### 2.1 DIAGRAMA DE PAQUETES

#### 2.1.1 Diagrama de Paquetes.

Cualquier sistema se debe dividir en unidades más pequeñas, de modo que las personas puedan trabajar con una cantidad de información limitada, a la vez y de modo que los equipos de trabajo no interfieran con el trabajo de los otros. Un paquete es una parte de un modelo. Cada parte del modelo debe pertenecer a un paquete. Pero para ser funcional, la asignación debe seguir un cierto principio racional, tal como funcionalidad común, implementación relacionada y punto de vista común. UML no impone una regla para componer los paquetes. Los paquetes ofrecen un mecanismo general para la organización de los modelos/subsistemas agrupando elementos de modelado. Cada paquete corresponde a un sub-modelo (subsistema) del modelo (sistema).

Los siguientes diagramas representan la estructura de paquetes utilizada en GEOPASTO.

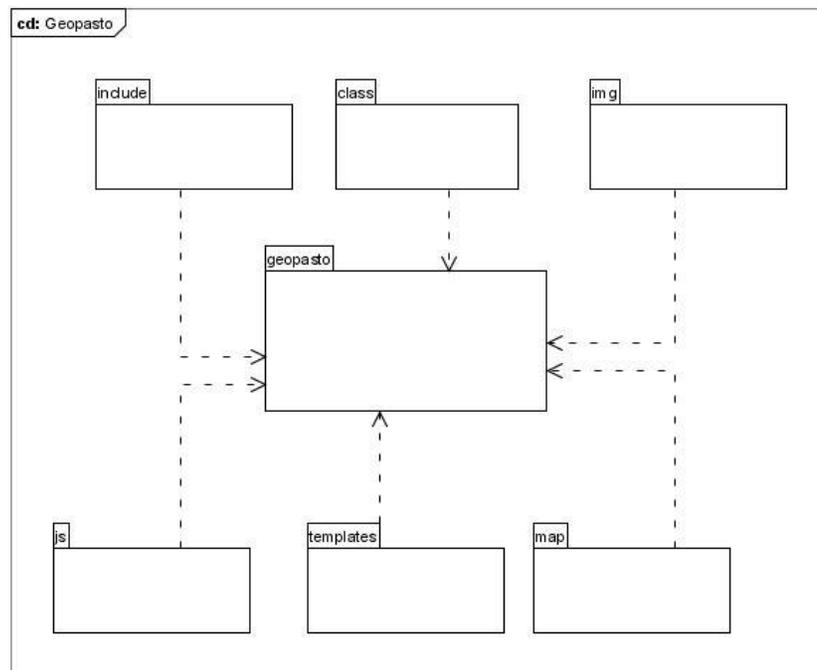


Figura 2. Diagrama Paquete Principal

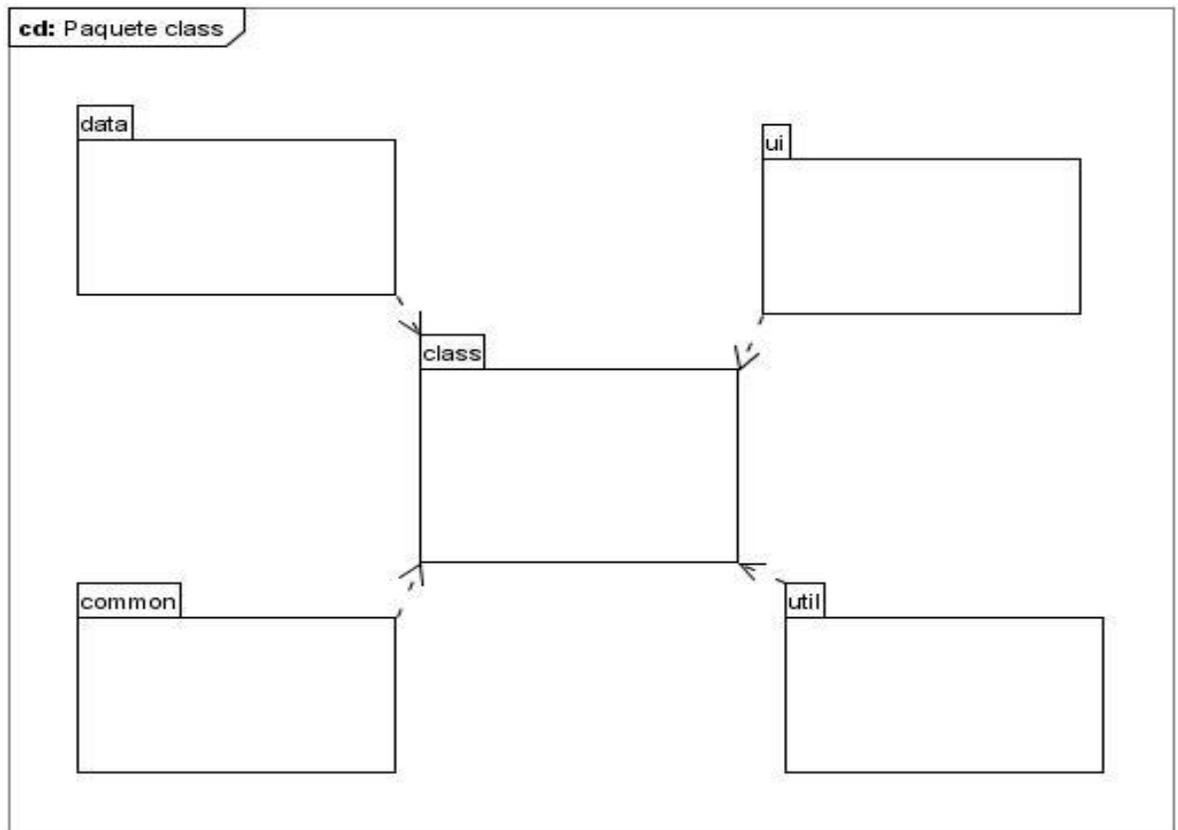


Figura 3. Diagrama Paquete Class

### 2.1.2 Diagramas de Clases.

Las clases representan los bloques de construcción más importantes de cualquier sistema orientado a objetos. Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica. Los Diagramas de Clases son utilizados durante el proceso de Análisis y Diseño de los sistemas informáticos donde se crea el diseño conceptual de la información que se manejará en el sistema, los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

Los siguientes diagramas representan las clases utilizadas en GEOPASTO.

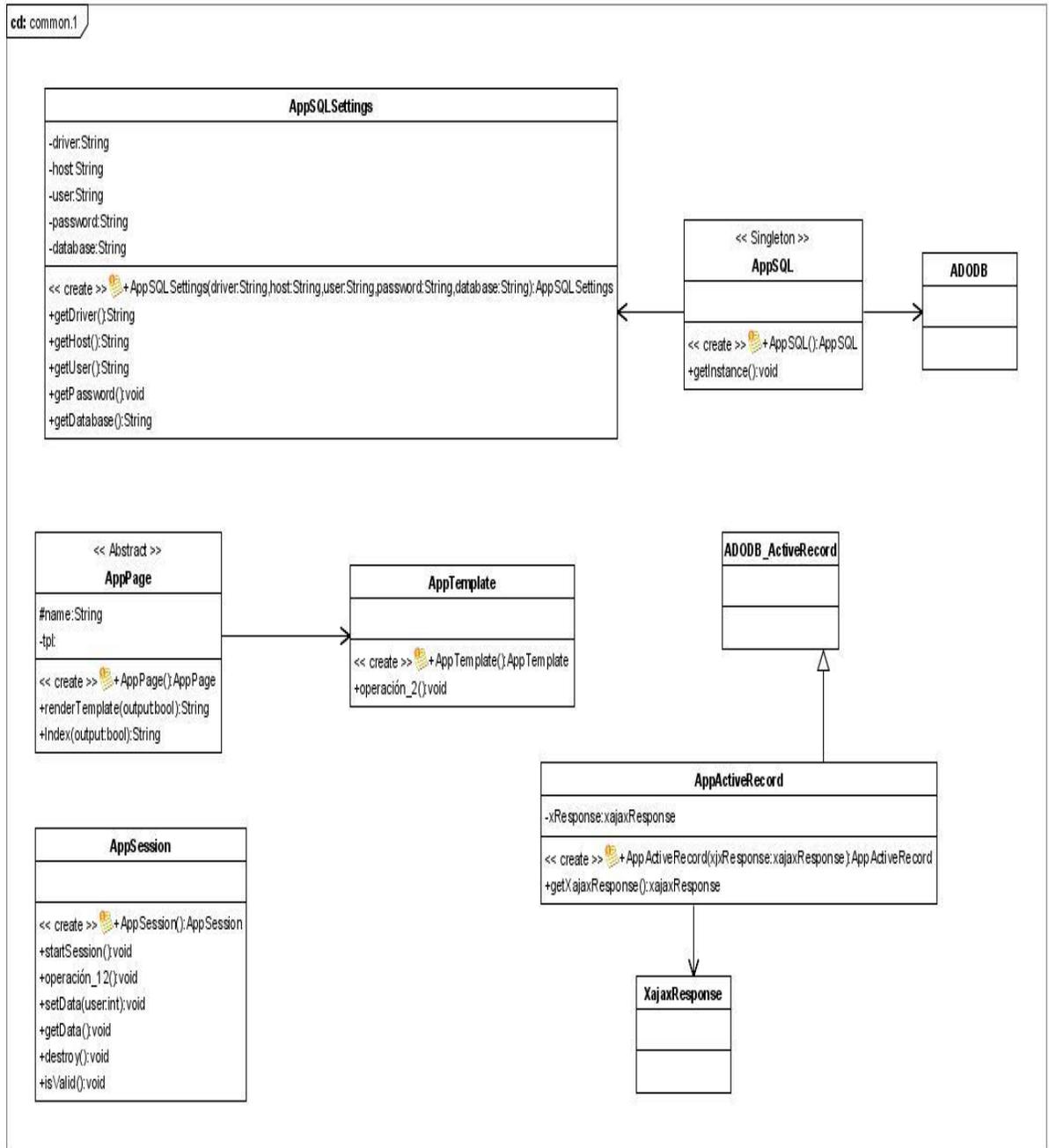


Figura 4. Diagrama de Clases Paquete Common

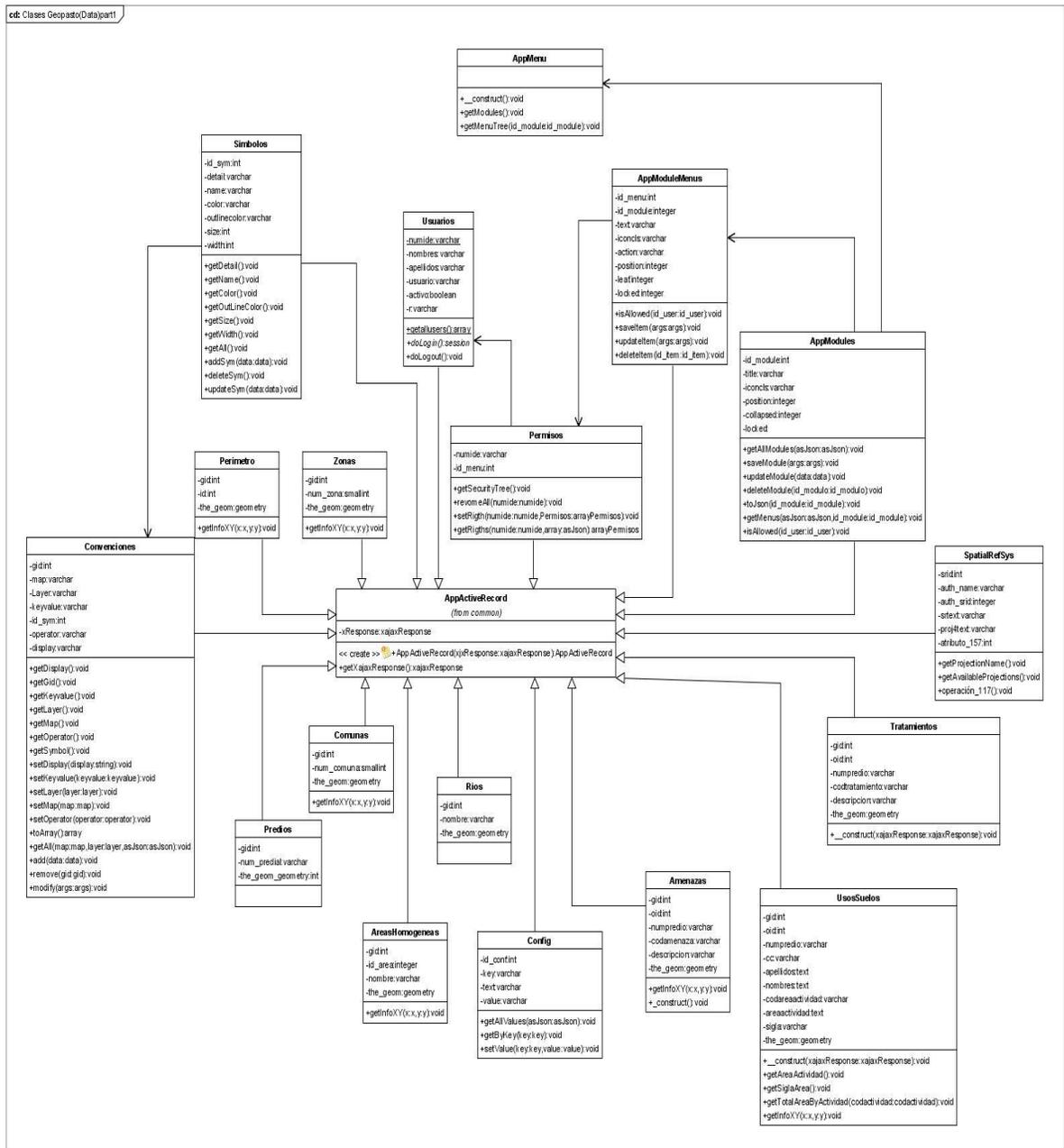


Figura 5. Diagrama de Clases Paquete Data

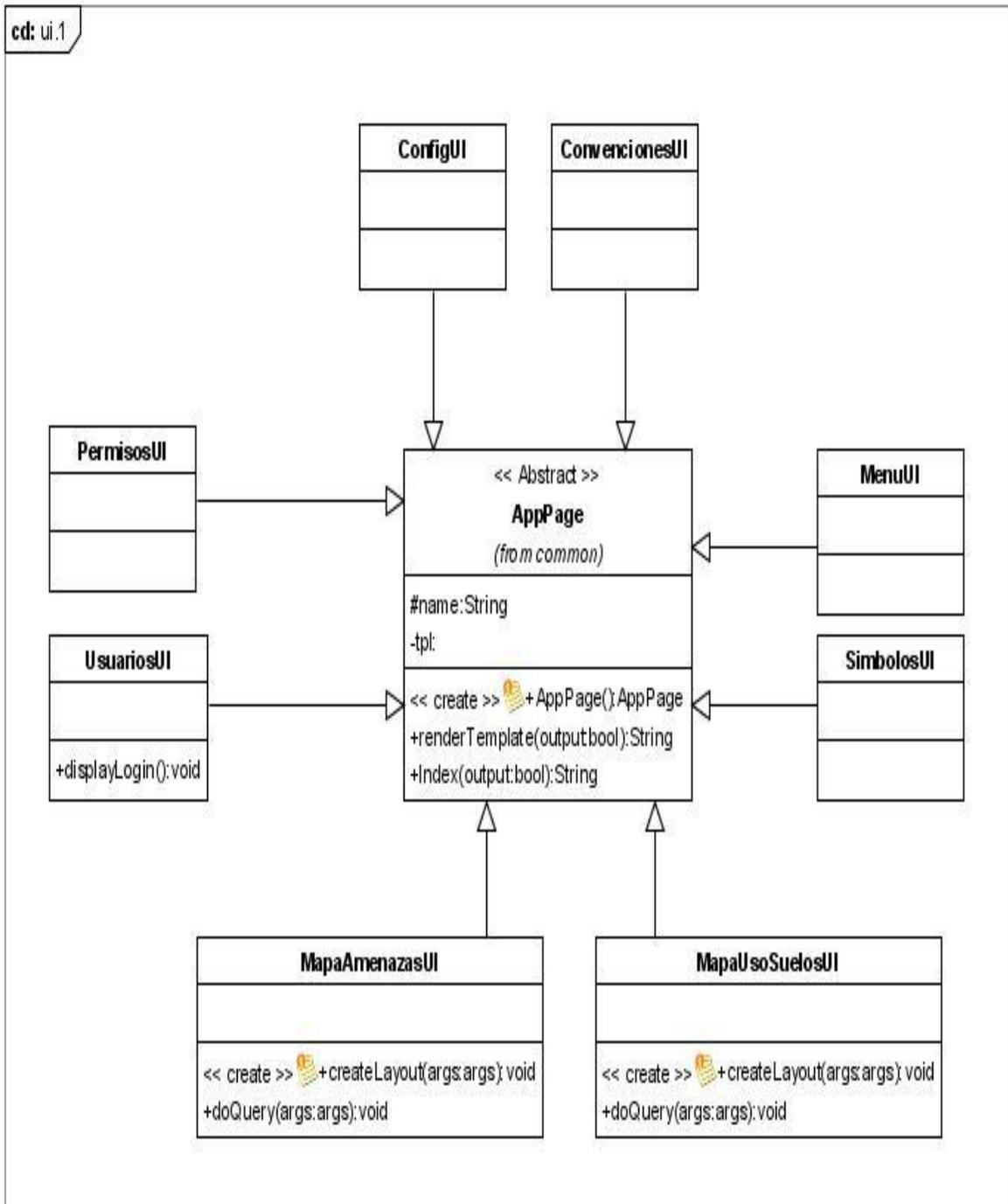


Figura 6. Diagrama de Clases Paquete UI

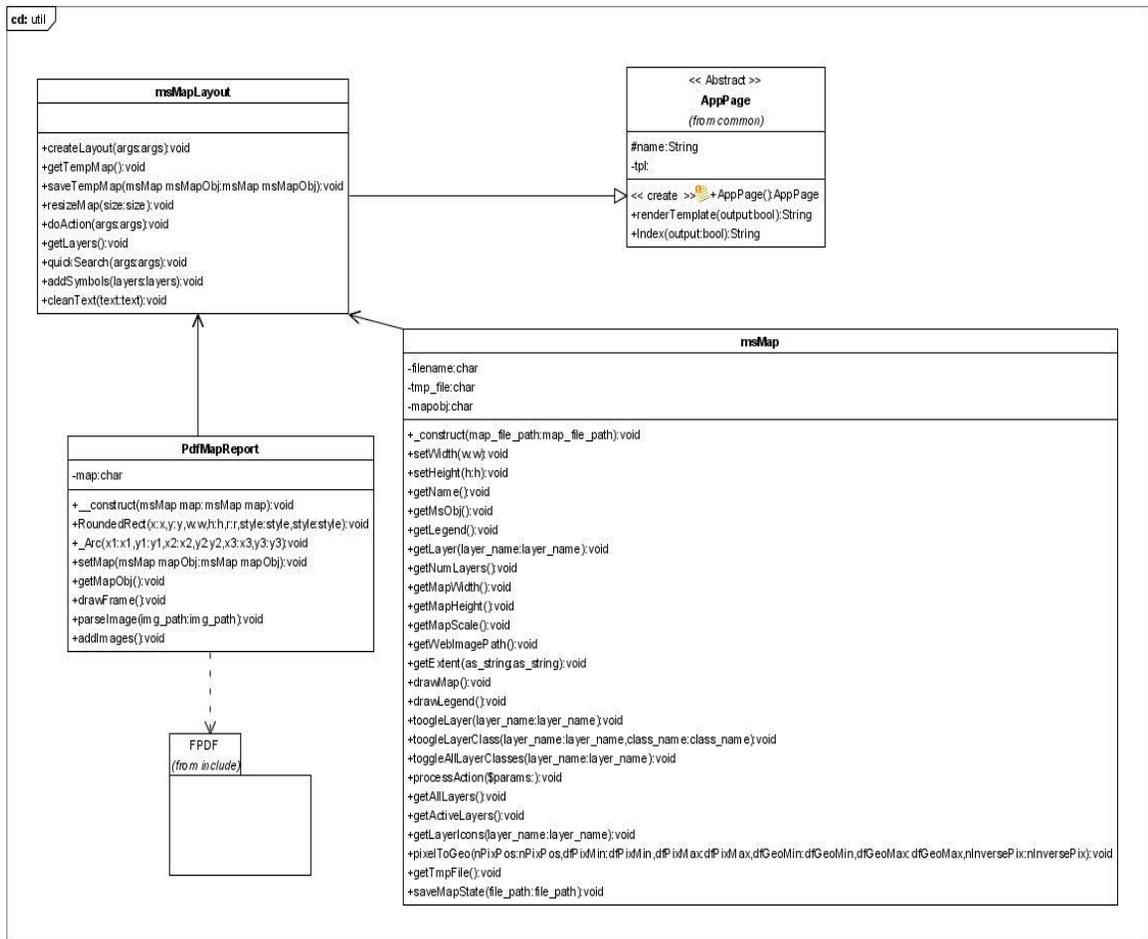


Figura 7. Diagrama de Clases Paquete UTIL

## 2.2 DISEÑO DE LA BASE DE DATOS

El sistema de información geográfica GEOPASTO involucra en su desarrollo el uso de tablas propias y de las existentes en el Sistema de Información Integral de la alcaldía del municipio de Pasto.

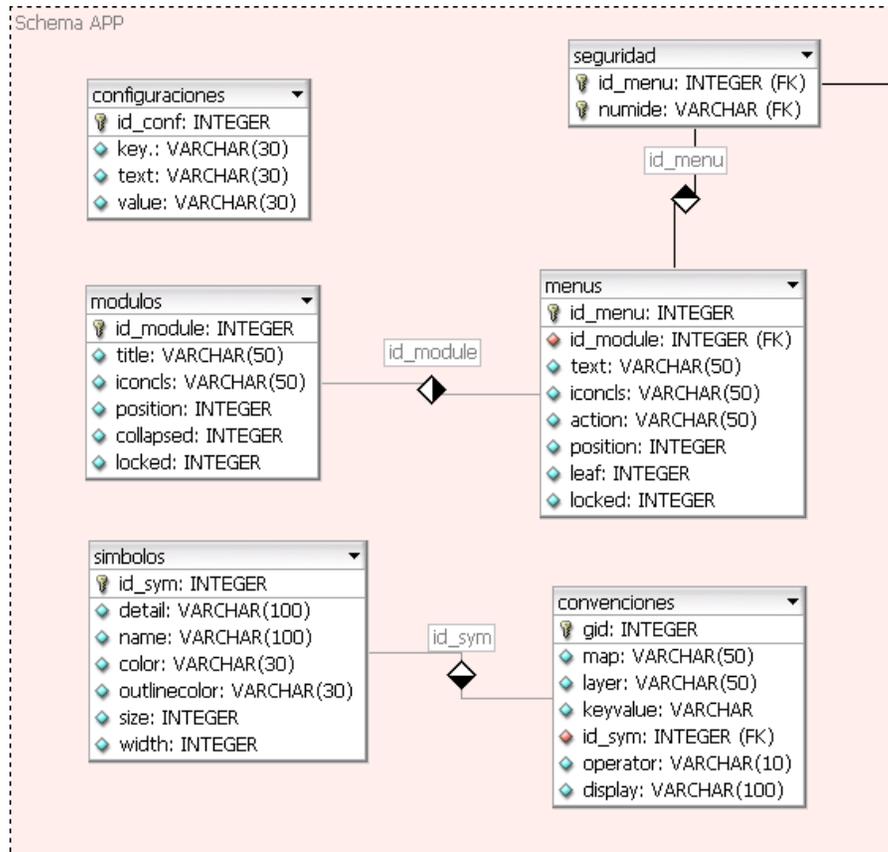
La base de datos de GEOPASTO está organizada en tres esquemas que se describen a continuación.

**2.2.1 Esquema APP.** En este esquema se encuentran las tablas relacionadas directamente con la configuración del sistema, los cuales permiten la configuración de la

presentación de la herramienta. El diagrama entidad-relación de este esquema se muestra en la figura 8.

Este esquema está compuesto de las siguientes tablas:

- *configuraciones*: En esta tabla se encuentra una relación concreta sobre la presentación de la herramienta, depurado y control de sesión; con un identificador de configuración (id\_conf), una identificación (key), un ítem de presentación (text) y un valor correspondiente (value).
- *convenciones*: En esta tabla se encuentra los datos correspondientes a las diferentes características de la información de las capas 'layers' del mapa que se presenta en el aplicativo; con un identificador de convenciones (gid), un mapa al cual pertenece (map), una capa relacionada (layer), un valor para condicionar el uso de símbolo (keyvalue), un identificador de símbolo (id\_sym) y un operador para la condición. Se relaciona con la tabla símbolos.
- *menus*: Se relaciona los datos concernientes a los menús que forman parte del sistema de información geográfica GEOPASTO, con un identificador de menú (id\_menu), identificador de un modulo (id\_module), un título (text), un icono que lo caracteriza (iconcls), un archivo con el cual se relaciona (action), la posición ocupada dentro del modulo (position), un identificador de nodo padre (leaf) y un campo de seguridad para el borrado (locked). Se relaciona con la tabla modulos.
- *modulos*: se relaciona los datos concernientes al agrupamiento de los menús según su característica con un identificador de modulo (id\_module), un título (title), un icono que lo caracterice (iconcls), una posición dentro del panel de la herramienta (position), característica que permita el verlo abierto o cerrado (collapsed) y un campo de seguridad para el borrado (locked). Se relaciona con la tabla menus.
- *seguridad*: En esta tabla se maneja los datos concernientes a los 'privilegios' que tiene cada uno de los usuarios que interactuó con el sistema de información geográfica GEOPASTO con un numero de identificación (numide) y un identificador de menú al cual tiene acceso (id\_menu). Se relaciona con las tablas menus y public.personas.
- *simbolos*: En esta tabla se encuentran los datos que corresponden a los símbolos a utilizar en la presentación del mapa, con un identificador de simbolos (id\_sym), una pequeña descripción de donde se lo usa (detail), el nombre propio del símbolo (name), el color que va a ser utilizado en formato RGB y separado con un espacio en cada uno de sus campos (color), el color de la línea que le corresponde en el formato RGB y separado con un espacio en cada uno de sus campos, un tamaño del símbolo (size) y un ancho (width) del mismo.



**Figura 8. Diagrama entidad relación esquema APP**

**2.2.2 Esquema GIS.** En este esquema se encuentran las tablas relacionadas con la presentación del mapa, cada una de estas tablas tiene dentro de si los campos de información de la geometría necesaria para la representación en el mapa. La característica de este esquema es que no hay relación alguna entre las tablas que lo conforman, como se puede ver en la figura 9.

- *areas\_homogeneas*: Esta tabla contiene los datos de las diferentes áreas homogéneas en las cuales se encuentra dividida el municipio de San Juan de Pasto con un identificador (gid), una numeración correspondiente (id\_area), un nombre (name) y la geometría correspondiente (the\_geom).
- *comunas*: Esta tabla contiene la información de las diferentes Comunas en las cuales se encuentra dividida el municipio de San Juan de Pasto con un identificador (gid), una numeración correspondiente para su identificación (num\_comuna) y la geometría correspondiente (the\_geom).

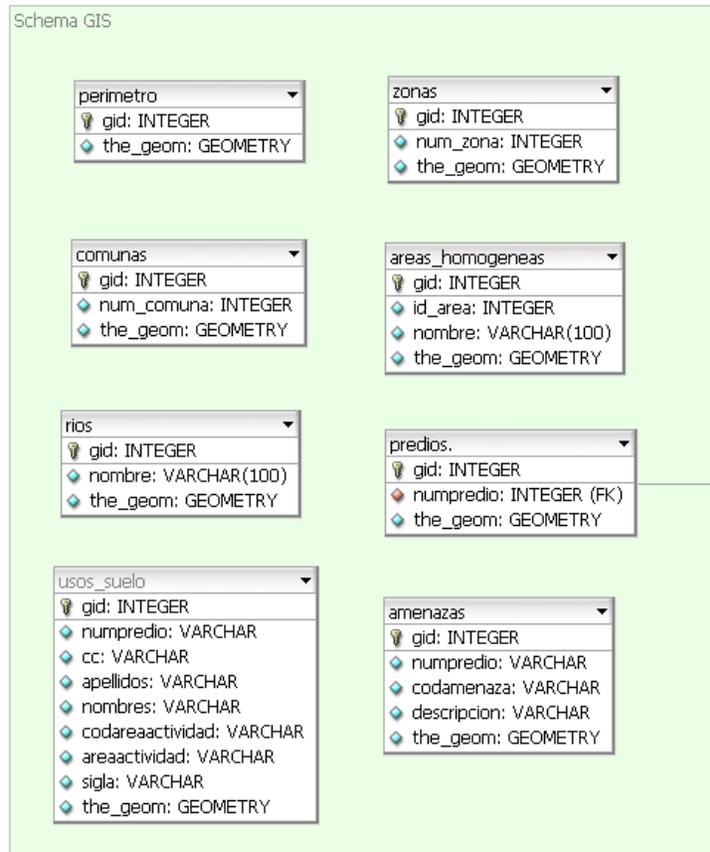


Figura 9. Diagrama entidad relación esquema GIS

- *construcciones*: Esta tabla tiene los datos de las diferentes construcciones que se encuentran en el municipio de San Juan de Pasto; con un identificador (gid), una numeración correspondiente (id) y la geometría correspondiente (the\_geom).
- *perimetro*: Esta tabla contiene los datos del perímetro correspondiente al municipio de San Juan de Pasto; con un identificador (gid), identificador propio (id) y la geometría correspondiente (the\_geom).
- *predios*: Esta tabla contiene los datos de los predios correspondientes al municipio de San Juan de Pasto; con un identificador (gid), el numero predial (num\_predial) y la geometría correspondiente (the\_geom).
- *rios*: Esta tabla contiene los datos de los ríos correspondientes al municipio de San Juan de Pasto; con un identificador (gid), el nombre (nombre) y la geometría correspondiente (the\_geom).
- *zonas*: Esta tabla contiene los datos de las zonas que dividen al municipio de San Juan de Pasto; con un identificador (gid), una numeración correspondiente (num\_zona) y la geometría correspondiente (the\_geom).

**2.2.3 Esquema PUBLIC.** En este esquema se encuentran agrupada toda la información que contiene el sistema de Información Integral (SII) del municipio de Pasto (ver figura 10):

- *personas*: Se relaciona los datos de las personas que hacen uso del SII. Con un número de identificación (numide), nombres y apellidos de las personas (nombres, apellidos).
- *p\_amenaza*: En esta tabla se relaciona la información del predio con el tipo de amenaza (codamenaza) que le esté afectando. Relacionada con las tablas p\_pot\_amenazas.
- *p\_pot\_amenazas*: En esta tabla se relaciona el tipo de amenaza (codamenaza) con una descripción y denominación de la misma (descripción y amenaza).
- *p\_pot\_areasactividad*: en esta tabla se relaciona la información correspondiente a la actividad que se presenta en el predio, teniendo un código de área de actividad (codareactividad), denominación del área de actividad (areaactividad), descripción de la misma (descripción), sigla propia del POT (sigla).
- *p\_pot\_clasesuelos*: Se relaciona la información con un código (codclasesuelo) y una denominación (clasesuelo).
- *p\_pot\_sectores*: Se relaciona la información con un código (codsector), denominación (sector) y un código de una pieza urbana (codpiezaurbana), se relaciona con las tablas p\_pot\_piezasurbanas, p\_pot\_subsectores.
- *p\_pot\_piezasurbanas*: se relaciona la información con un código de pieza urbana (codpiezaurbana) y una denominación de la misma (piezasurbana). Se relaciona con la tabla p\_pot\_sectores.
- *p\_pot\_subsectores*: Se relaciona la información con un código de subsector (codsubsector), una denominación del mismo (subsector), un código de sector (codsector), índice de construcción (iconstruccion), índice de ocupación (iocupacion), índice de cesión (icesion). Se relaciona con la tabla p\_pot\_sectores.
- *propietariospredios*: Se relaciona los datos de un predio (numpredio) con el número de identificación del dueño (numide), se relación con las tablas predios, personas.
- *predios*: Se relaciona los datos correspondientes a un predio con su ubicación en el municipio de San Juan de Pasto, con el número de predio (numpredio), la dirección(direccion), el código de barrio (codbarrio), código área de actividad (codareaactividad), sector (sector), manzana (manzana). Se relaciona con las tablas p\_pot\_sector, p\_pot\_areasactividad, p\_div\_barrio.
- *p\_div\_barrios*: Se relaciona los datos correspondientes a la ubicación del predio en el municipio de San Juan de Pasto con un código de Barrio (codbarrio), el nombre del mismo (barrio) e identificador de comuna (codcomuna).

- *p\_pot\_tratamientos*: Se relacionan los datos correspondientes a los diferentes tratamientos que se tiene sobre el suelo con un código de tratamiento (codtratamiento) y una descripción del mismo (tratamiento).
- *p\_pot\_tipologiavias*: Se relacionan los datos correspondientes a los diferentes distintivos de los corredores viales del municipio de Dan Juan de Pasto con un código de tipología (codtipologia), una denominación (tipología), una descripción de la misma (descripcion) y una sigla correspondiente a cada tipo de vía propia del POT (sigla).
- *p\_dem\_estrato*: Se relacionan los datos correspondiente a una solicitud realizada a la oficina de planeación por medio del SII, con un código de solicitud (codsolicitud), el numero de la manzana radicado por el dane (manzanadane), el numero de manzana radicado por el instituto Geográfico Agustín Codazzi IGAC (manzanaigac) y una dirección (dirección), se relaciona con la tabla *p\_pot\_solicitudes*.
- *p\_pot\_solicitudes*: Se relacionan los datos de todas las solicitudes que se presenten ante la oficina de planeación; con un código de solicitud (codsolicitud), número de identificación de la persona que hace la solicitud (numide), número del predio (numpredio), código de barrio al que pertenece el predio (codbarrio). Se relaciona con las tablas predios, *p\_pot\_demarcaciones*, *p\_div\_barrios*, personas, propietariospredios, *p\_dem\_estrato*.
- *p\_pot\_demarcaciones*: Se relacionan los datos de los predios de una demarcación relacionados con un número de predio (numpredio), tipología de vía (codtipologiavia), clase de suelo (codclasesuelos), area de actividad (codareaactividad), subsector al que pertenece (codsubsector), Tratamiento de suelo (codtratamiento), Amenazas (codamenaza), índice de ocupación (iocupacion), índice de construcción (iconstruccion), índice de sesión (icesion). Se relaciona con las tablas *p\_pot\_subsectores*, *p\_pot\_tipologiavias*, *p\_pot\_tratamientos*, *p\_pot\_amenazas*, *p\_pot\_clasesuelos*.



### 3. MANEJO DE GEOPASTO

#### 3.1 INICIO

Acción del Usuario	Respuesta del sistema
2. El Usuario escribe su nombre de Usuario en el campo 'Usuario' (A). 3. El Usuario escribe la contraseña en el Campo 'Contraseña' (B). 4. El Usuario hace clic en el botón Iniciar Sesión (C)	1. Se muestra la pantalla Iniciar Sesión (figura 11).  5. Si la verificación de los datos 'Usuario' y 'Contraseña' es correcta, entonces se muestra la ventana principal de la aplicación, de lo contrario se muestra un mensaje de error (figura 12).



Figura 11. Pantalla Iniciar Sesión

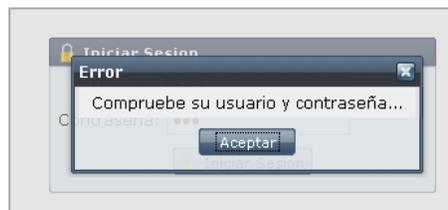


Figura 12. Mensaje de Error

### 3.2 MENU PRINCIPAL

Acción del Usuario	Respuesta del sistema
1. El Usuario Ingresa a la Aplicación	2. Se Muestra la interfaz gráfica de la aplicación, como se muestra en la Figura 13. (A): Árbol de Módulos, En esta parte se encuentran los diferentes Módulos que forman la aplicación, entre los cuales se encuentran el Modulo de configuración, Modulo de Seguridad y Modulo de Cartografía. Por defecto todas las pestañas se presentan en forma cerrada. (B): Área de Trabajo, Lugar donde se presentaran al usuario todas las pantallas que habrá en el uso de la aplicación.

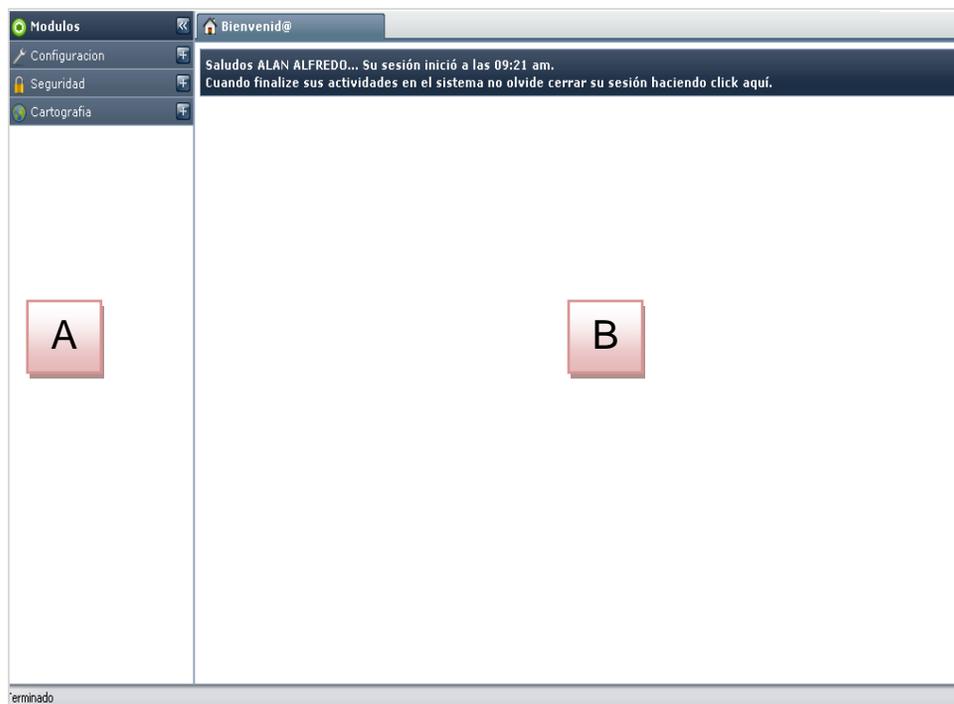


Figura 13. Ventana Principal de la Aplicación

### 3.3 MÓDULO CONFIGURACIÓN

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la pestaña de Configuración (A)	2. Se despliegan las opciones que están registradas en el Módulo de Configuración (B) (Figura 14).

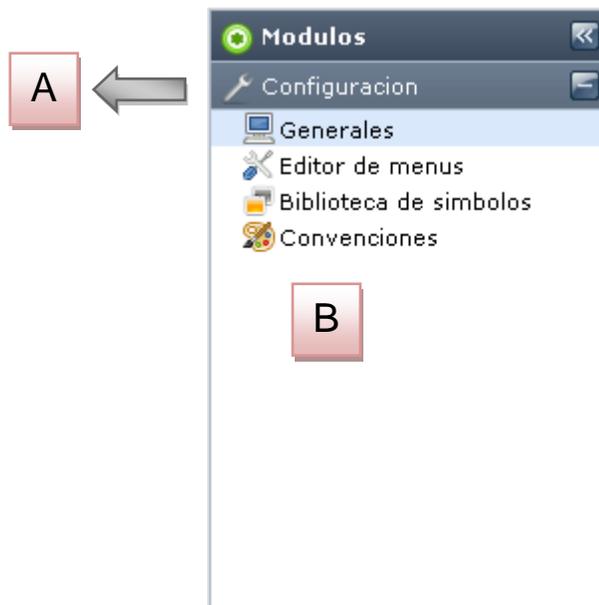


Figura 14. Módulo de Configuración

### 3.3.1. Opción Generales

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la Opción de generales (A)	2. Se abre la ventana correspondiente. Figura 15.

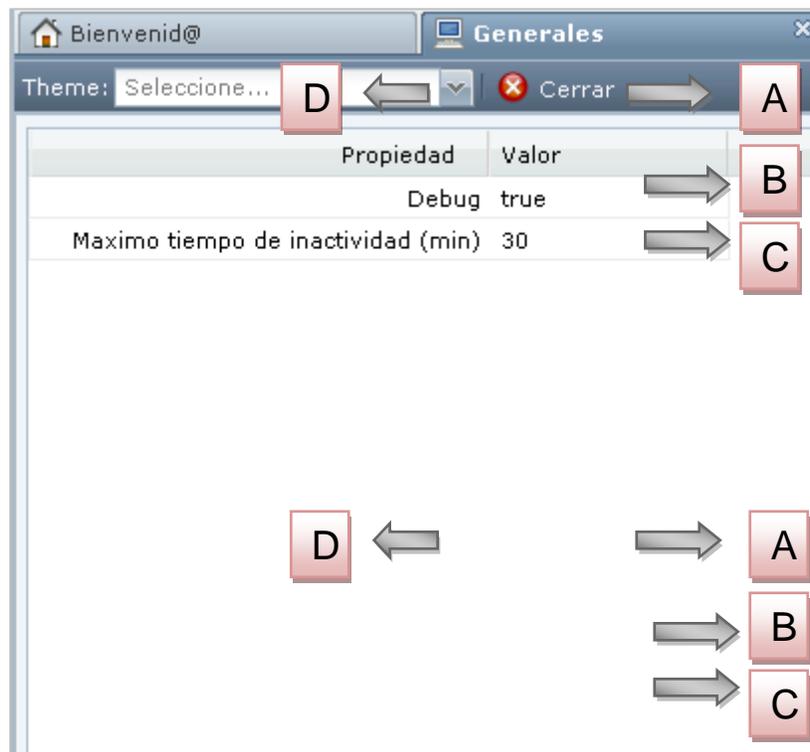


Figura 15. Pantalla Generales

Acción del Usuario	Respuesta del Sistema
<p>1. El Usuario hace clic en la Opción Cerrar (A)</p> <p>3. El usuario puede Cambiar el valor de Debug (B) en true o False.</p> <p>4. El usuario puede cambiar el valor del tiempo de inactividad (C).</p> <p>5. El usuario puede escoger el tema de trabajo (D).</p>	<p>2. La pantalla se cierra.</p> <p>6. Se despliega las opciones de los temas. Figura 16.</p>

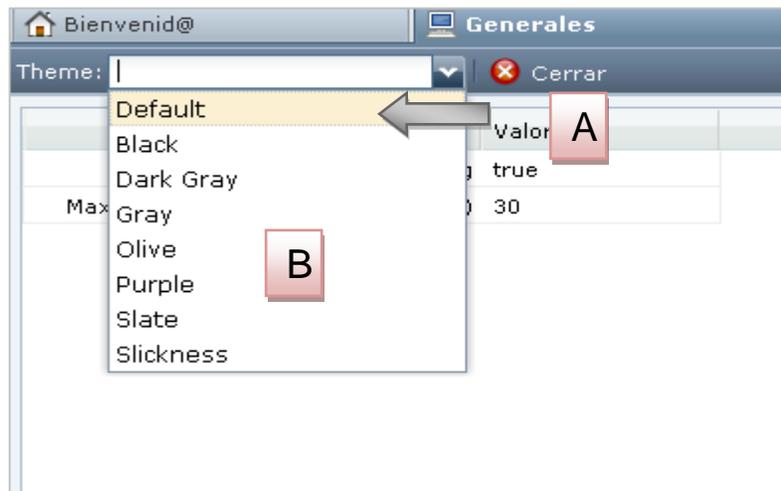


Figura 16. Temas

Acción del Usuario	Respuesta del Sistema
<p>1. El Usuario hace clic en la Pestaña de despliegue (A).</p> <p>3. El usuario puede escoger el tema que guste.</p>	<p>2. Se presentan los diferentes temas (B).</p>

### 3.3.2. Editor de Menús

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la Opción Editor de menús (A).	2. Se presenta la pantalla correspondiente. Figura 17.

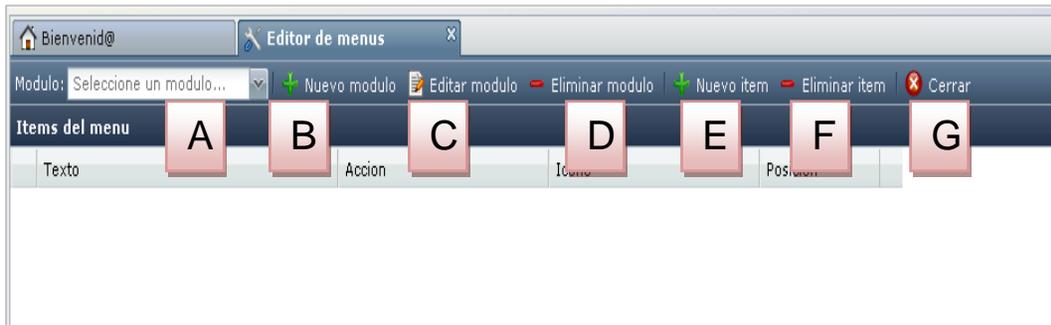
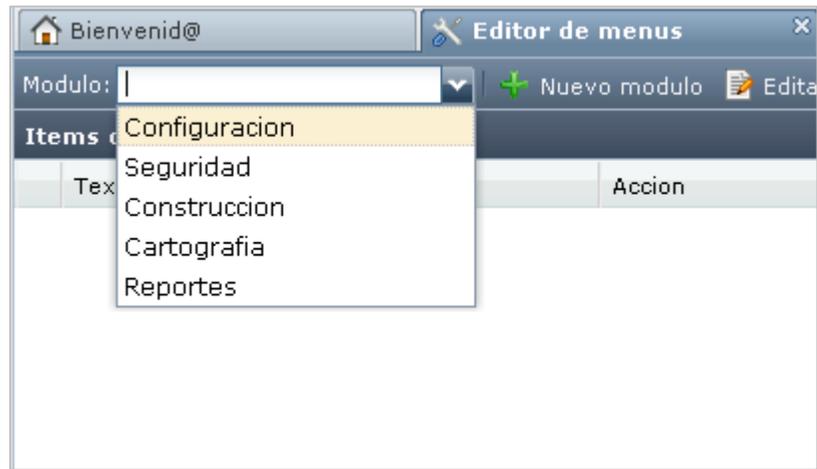


Figura 17. Pantalla Editor de Menús

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la pestaña para desplegar los diferentes Módulos que conforman el Sistema (A).	2. Se presentan las opciones correspondientes. Figura 18.
3. El Usuario Selecciona el módulo sobre el cual trabajara.	4. Se listan los diferentes ítems correspondientes al módulo seleccionado. Figura 19.
5. El Usuario Selecciona la opción 'Nuevo Módulo' (B)	6. El nuevo Modulo se Añade en el combo.
7. El Usuario Selecciona 'Editar Módulo'	8. Se presentan las opciones que son Editables para el módulo seleccionado. Figura 20.
9. El Usuario selecciona 'Eliminar Módulo'	10. Se presenta un mensaje de confirmación de borrado. Figura 21.
11. El Usuario selecciona 'Nuevo Item'	12. Se añade una fila en la lista del modulo escogido. Figura 22.

- |   |   |
|---|---|
| 13. El Usuario selecciona 'Eliminar Ítem' | 14. Se presenta un mensaje de confirmación de borrado. Figura 23. |
|---|---|

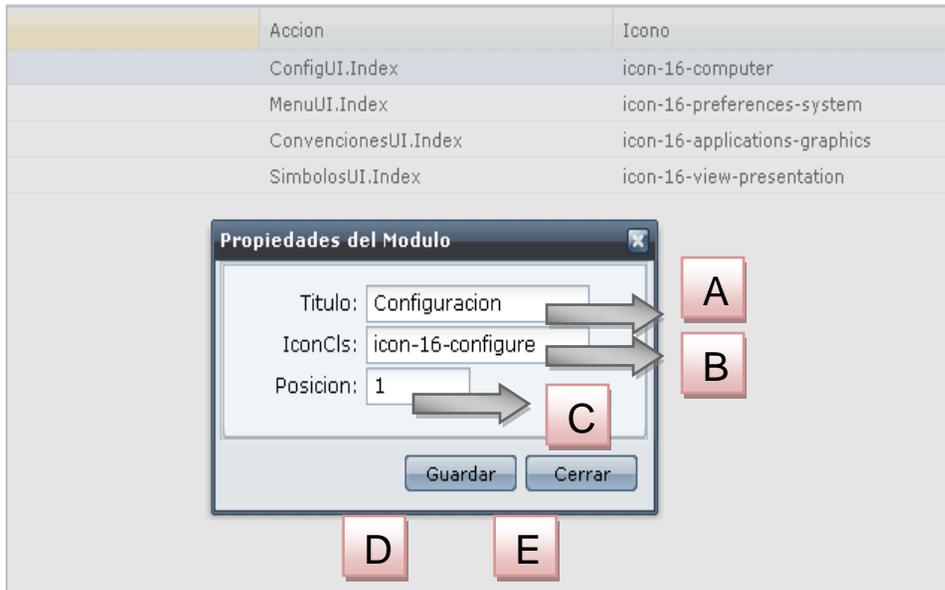


**Figura 18. Selección de Módulo**

The screenshot shows the 'Editor de menus' window with the 'Modulo:' dropdown set to 'Configuracion'. The 'Items del menu' section displays a table with the following data:

	Texto	Accion	Icono	Posicion
1	Generales	ConfigUI.Index	icon-16-computer	1
2	Editor de menus	MenuUI.Index	icon-16-preferences-system	2
3	Convenciones	ConvencionesUI.Index	icon-16-applications-graphics	4
4	Biblioteca de simbolos	SimbolosUI.Index	icon-16-view-presentation	3

**Figura 19. Listado de Módulos**



**Figura 20. Edición del Módulo**

Acción del Usuario	Respuesta del Sistema
<p>1.El Usuario Puede cambiar el nombre del módulo en el campo de texto 'Titulo'(A).</p> <p>2. El Usuario Determina el icono con el cual se distingue el módulo en el campo de texto d 'IconCls' (B).</p> <p>3. El Usuario puede determinar la posición en la cual se va a ubicar el módulo.(C)</p> <p>4. El Usuario guarda los cambios realizados al dar clic en el botón 'Guardar'</p> <p>7. El usuario cierra la Ventana de dialogo (E)</p>	<p>6. Se cierra la ventana de dialogo y se guardan los cambios</p> <p>8. Se cierra la ventana de dialogo sin guardar los cambios</p>



Figura 21. Eliminar Módulo

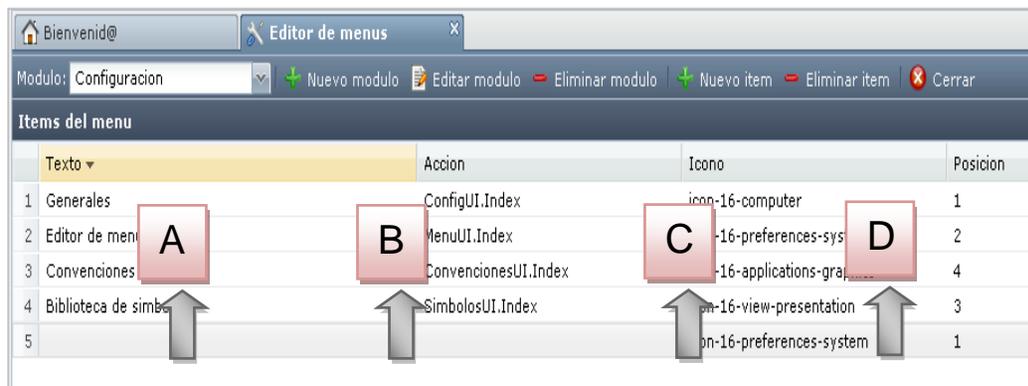


Figura 22. Insertar Ítem

Acción del Usuario	Respuesta del Sistema
1. El Usuario Nombra el ítem en el campo 'Texto'(A).	
2. El Usuario Determina el archivo que servirá como enlace a este nuevo ítem en el campo 'Acción' (B).	
3. El Usuario puede determinar el icono que caracteriza a este ítem en el campo 'Icono'.(C)	

4. El Usuario ubica la posición del ítem dentro del módulo en el campo 'Posición' (D)	5. Se subrayan los cambios realizados.
---	--



Figura 23. Eliminar Ítem

### 3.3.3. Biblioteca de Símbolos

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la opción Biblioteca de Símbolos (A).Figura 24	2. Se presenta la pantalla correspondiente. Figura 25

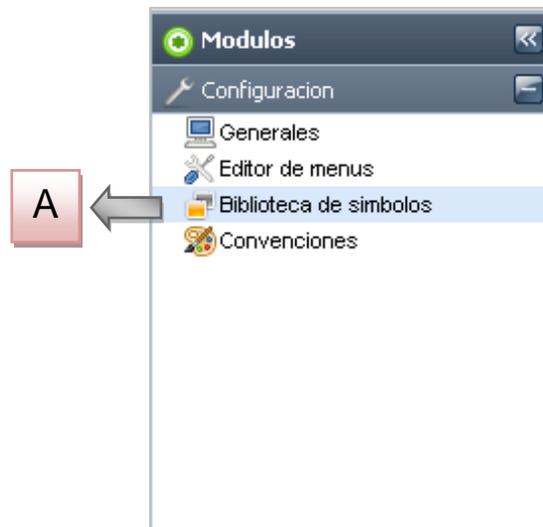


Figura 24. Opción Biblioteca de Símbolos



Figura 25. Pantalla biblioteca de símbolos

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la opción 'Nuevo Símbolo'. (A)	2. Se agrega una nueva fila en la pantalla. Figura 26
3. El usuario hace clic en la opción 'Eliminar Símbolo' (B)	4. Se presenta un cuadro informativo sobre el borrado del símbolo. Figura 27
5. El usuario hace clic en la opción 'Cerrar'. (C)	6. Se cierra la ventana de 'Biblioteca de Símbolos'



Figura 26. Nuevo Símbolo

Acción del Usuario	Respuesta del Sistema
1. El Usuario introduce el detalle del nuevo símbolo en el campo 'Detalle'. (A)	
2. El usuario introduce el nombre del símbolo (Existente en el archivo de	

<p>configuración Symbols.sym) correspondiente en el campo 'Nombre' (B)</p> <p>3. El usuario introduce el color que le corresponde en formato RGB, cada campo separado por un espacio. (C)</p> <p>4. EL usuario puede cambiar el color de la línea, van en formato RGB, cada valor separado por un espacio. El color por defecto es negro. (D)</p> <p>5. El usuario puede cambiar el tamaño del símbolo. El valor por defecto es 5. (E)</p> <p>6. El usuario puede cambiar el tamaño del calibre de la línea. El valor por defecto es 1.(F)</p>	<p>7. Los cambios son registrados.</p>
--	--

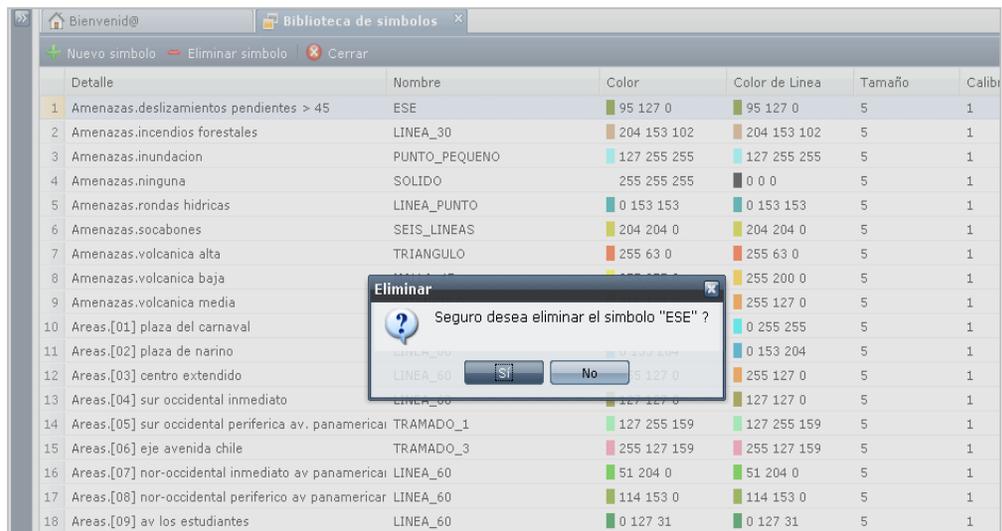


Figura 27. Eliminar Símbolo

### 3.3.4. Convenciones

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la opción Convenciones (A).Figura 28	2. Se presenta la pantalla correspondiente. Figura 29



Figura 28. Opción Convenciones

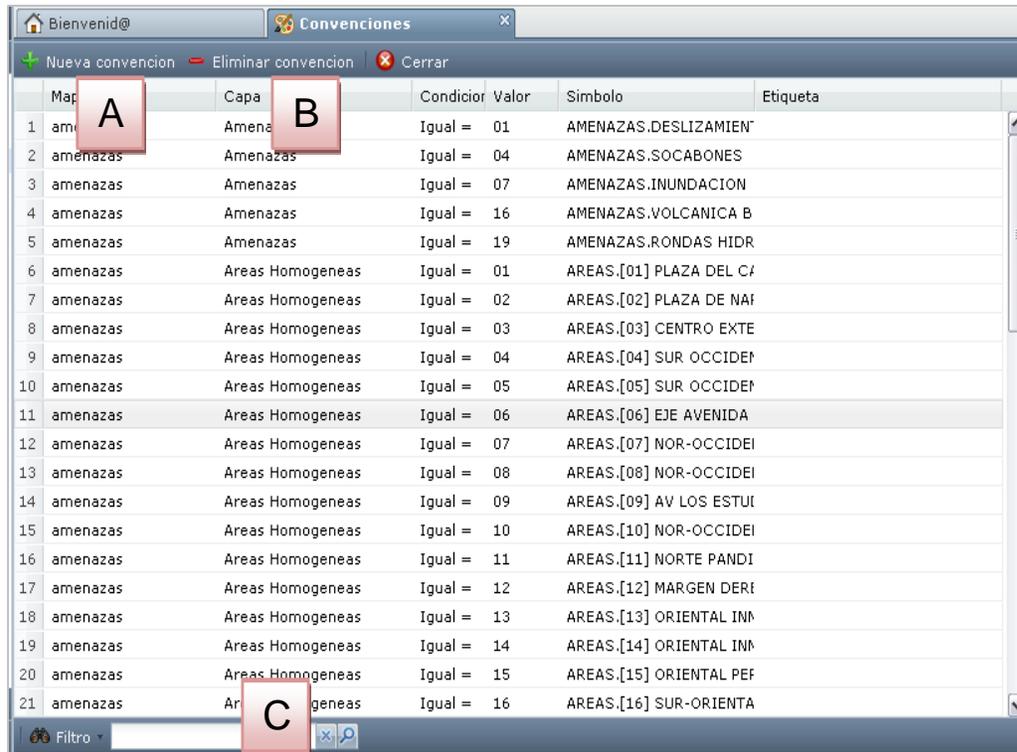


Figura 29. Pantalla Convenciones

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la opción 'Nueva convención' (A).	2. Se añade una nueva fila para la inserción de los datos correspondientes. Figura 30.
3. El usuario hace clic en la opción 'Eliminar Convención' (B).	4. Se presenta un mensaje de confirmación. Figura 31.
5. El usuario utiliza la herramienta de búsqueda rápida (C).	6. Se presenta los resultados obtenidos.

	Mapa	Capa	Condicion	Valor	Simbolo	Etiqueta
1					SOS.RESIDENCIAL	
2	amenazas	Amenazas		01	AMENAZAS.SOCABONES	ZAMIENTOS PE
3	amenazas	Amenazas	Igual =	04	AMENAZAS.SOCABONES	
4	amenazas	Amenazas	Igual =	07	AMENAZAS.INUNDACION	

**Figura 30. Nueva Convención**

Acción del Usuario	Respuesta del Sistema
<p>1. El Usuario Introduce el nombre del mapa al cual se relaciona la convención creada(A).</p> <p>2. El usuario introduce el nombre de la capa en la cual se indicara (B).</p> <p>3. El usuario Establece la condición con la cual se usara esta convención. Las opciones son: <i>igual =, mayor &gt;, menor&lt;, diferente !=</i>.(C )</p> <p>4. El usuario establece el valor par que la condición se cumpla (D).</p> <p>5. el usuario determina el Símbolo que usara con esta convención. El sistema por defecto presenta <i>USOS.RESIDENCIAL</i> (E)</p>	<p>6. Se registran los cambios realizados.</p>

Categoría	Condición	Valor	Símbolo
Amenazas	Igual =	01	AMENAZAS.DESLIZAMIENTOS PE
Amenazas	Igual =	04	AMENAZAS.SOCABONES
Amenazas	Igual =	07	AMENAZAS.INUNDACION
Amenazas	Igual =	16	AMENAZAS.VOLCANICA BAJA
Amenazas	Igual =	19	AMENAZAS.RONDAS HIDRICAS
Areas Homogeneas	Igual =	01	AREAS.[01] PLAZA DEL CARNAV
Areas Homogeneas	Igual =	02	AREAS.[02] PLAZA DE NARINO
Areas Homogeneas	Igual =	03	AREAS.[03] PLAZA DE O EXTENDID
Areas Homogeneas	Igual =	04	AREAS.[04] NOR-OCCIDENTAL II
Areas Homogeneas	Igual =	05	AREAS.[05] NOR-OCCIDENTAL P
Areas Homogeneas	Igual =	06	AREAS.[06] NOR-OCCIDENTAL I
Areas Homogeneas	Igual =	07	AREAS.[07] NOR-OCCIDENTAL I
Areas Homogeneas	Igual =	08	AREAS.[08] NOR-OCCIDENTAL F
Areas Homogeneas	Igual =	09	AREAS.[09] AV LOS ESTUDIANTI
Areas Homogeneas	Igual =	10	AREAS.[10] NOR-OCCIDENTAL T
Areas Homogeneas	Igual =	11	AREAS.[11] NORTE PANDIACO M
Areas Homogeneas	Igual =	12	AREAS.[12] MARGEN DERECHA I

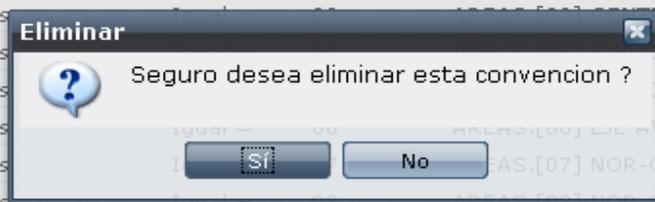


Figura 31. Eliminar Convención

### 3.4 MÓDULO SEGURIDAD

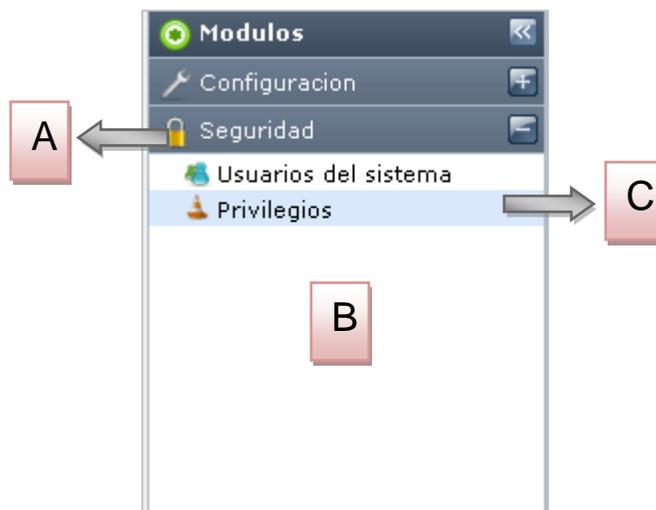


Figura 32. Modulo Seguridad

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la pestaña de Seguridad (A). Figura 32.	2. Se despliegan las opciones que están registradas en el Módulo de Seguridad (B).
3. El Usuario hace clic en la opción Usuarios del sistema (B).	4. Se presenta la pantalla correspondiente, listando a los usuarios que se encuentran Activos en el sistema. Figura 33.
5. El usuario hace clic en la opción privilegios(C).	6. Se presenta la pantalla correspondiente. Figura 34.

Acción del Usuario	Respuesta del Sistema
1. Se selecciona el usuario al cual se le van asignar los privilegios (A). Figura 34.	4. Se presenta la pantalla correspondiente, listando los ítems registrados en el sistema. Figura 35.  6. Se presenta el resultado de la búsqueda.
3. El usuario hace clic en la opción 'Asignar Privilegios' (B). Figura 34.	
5. El usuario utiliza la herramienta de búsqueda rápida (C).	



Identificación	Apellidos	Nombres	Usuario
1 98398972	ALBORNOZ ROMO	ALAN ALFREDO	alan
2 51580959	AMEZQUITA AMEZQUITA	PAULINA	pau
3 12991957	ASTORQUIZA MORENO	MAURICIO	mauricio
4 12966770	BENAVIDES L	MARCO ANTONIO	marco
5 52717747	BURBANO MU	NATHALY	netabm
6 12975292	CABRERA PANTOJA	ALVARO ENRIQUE	0
7 27087471	CALLE VELASCO	CAROLINA	carocalle
8 999999	CAMPA	SIXTO	siger
9 98392398	CAMPA	SIXTO ENRIQUE	
10 24326431	CANO ARIAS	AMPARO CLARET	claret
11 12989547	CHAMORRO MORENO	GERMAN	gerchamo
12 11307918	CHAVEZ URREA	JULIO CESAR	jchavez
13 37012747	CLAUDIA XIMENA	RODRIGUEZ ORTEGA	claux

**Figura 33. Lista de Usuarios del Sistema**

	Apellidos	Nombres	Usuario
1	ALBERNOZ ROMO	ALAN ALFREDO	alan
2	ALMEIDA AMEZQUITA	PAULINA	pau
3	ASCONZA MORENO	MAURICIO	mauricio
4	BELTRAN S L	MARCO ANTONIO	marco
5	BURBANO MU	NATHALY	natabm
6	CABRERA PANTOJA	ALVARO ENRIQUE	0
7	CALLE VELASCO	CAROLINA	carocalle
8	CAMPA	SIXTO	siger
9	CAMPA	SIXTO ENRIQUE	
10	CANO ARIAS	AMPARO CLARET	claret
11	CHAMORRO MORENO	GERMAN	gerchamo
12	CHAVEZ URREA	JULIO CESAR	jchavez
13	CLAUDIA XIMENA	RODRIGUEZ ORTEGA	claux
14	DE CONSTRUCCI	LIC	
15	DE CONSTRUCCI	LICENCIAS	licencias
16	DE PROYECTO	BANCO	banco
17	DE PROYECTOS	ADMINISTRADOR	adminproye
18	DE PRUEBA	USUARIO	prueba
19	DELGADO GUERRERO	LUIS CARLOS	lukas
20	DELGADO SAMUDIO	DIANA CAROLINA	diana
21	ENRIQUEZ MARTINEZ	JAIME ENRIQUE	jaimee

Figura 34. Pantalla de Privilegios

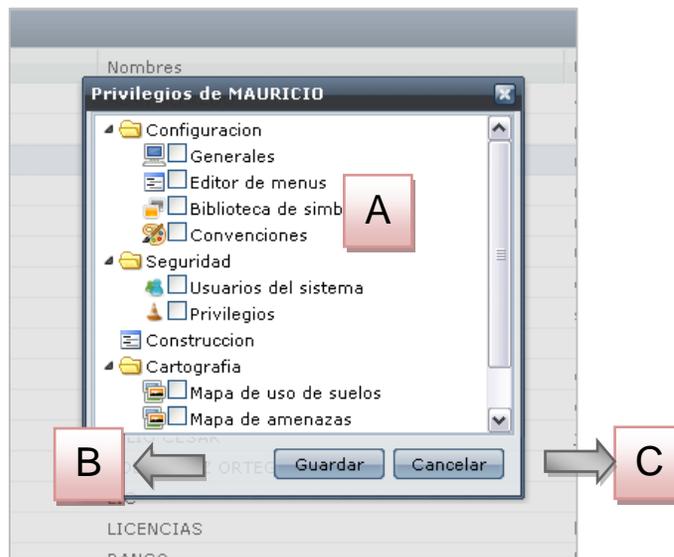


Figura 35. Asignación de Privilegios

Acción del Usuario	Respuesta del Sistema
1. Se Marca cada uno de los ítems a los cuales se le quiere dar acceso(A). Figura 35.	
2. El usuario hace clic en el botón Guardar (B).	3. Se registran los cambios realizados.
4. El usuario hace clic en el botón Cancelar (C).	5. Se cierra la ventana y no se guardan los cambios realizados.

### 3.5 MÓDULO CARTOGRAFIA

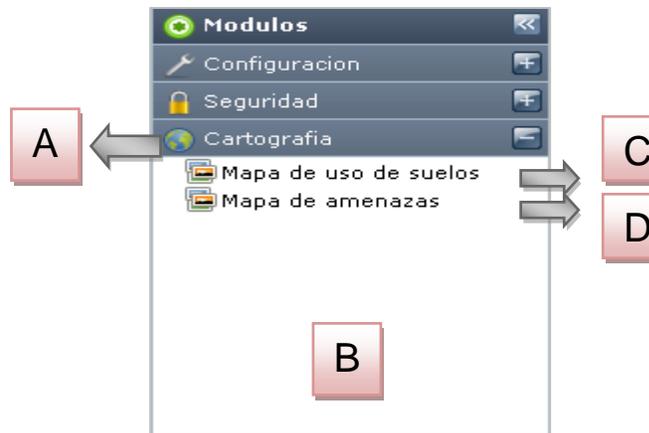


Figura 36. Módulo Cartografía

Acción del Usuario	Respuesta del Sistema
1. El Usuario hace clic en la pestaña Cartografía (A). Figura 36.	2. Se despliegan las opciones que están registradas en el Módulo cartografía (B).
3. se escoge la opción <i>Mapas de uso de Suelos</i> (C).	4. Se presenta en pantalla el mapa correspondiente. Figura 37.
5. Se escoge la opción <i>mapa de amenazas</i> (D).	6. se presenta en pantalla el mapa correspondiente. Figura 38.
<b>NOTA:</b> Las herramientas de navegación y consulta son las mismas en los dos tipos de mapas, salvo unas excepciones.	

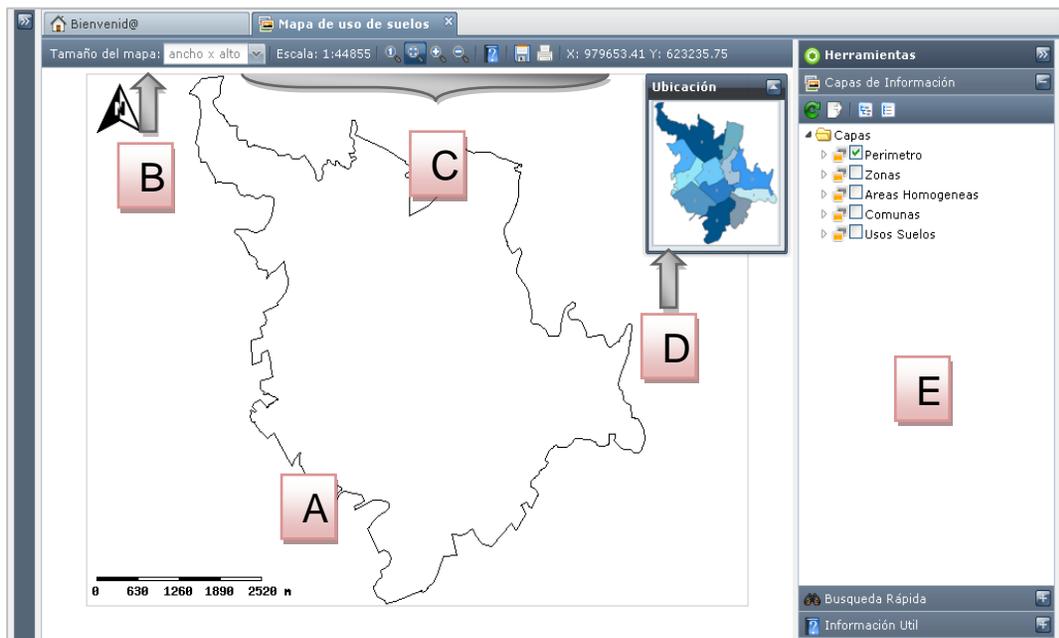


Figura 37. Mapa Uso de Suelos

Acción del Usuario	Respuesta del Sistema
<p>2. El Usuario hace clic en el área del mapa (A). Figura 37.</p> <p>4. Se escoge el tamaño de presentación del mapa. Existen las opciones de 640x480; 800x600; 1024x768 (B).</p> <p>6. Se usan las herramientas de navegación (C). Figura 39.</p>	<p>1. Se presentan el modulo de Herramientas. Por defecto se presenta abierta la pestaña <i>Capas de información</i> y activa la capa de perímetro (E). Figura 37.</p> <p>3. Se realiza un zoom de entrada o de salida del mismo, según este configurado.</p> <p>5. Se presenta en pantalla el mapa en las dimensiones seleccionadas.</p> <p>7. Se presenta una ubicación a nivel general del lugar de navegación actual (D).</p>

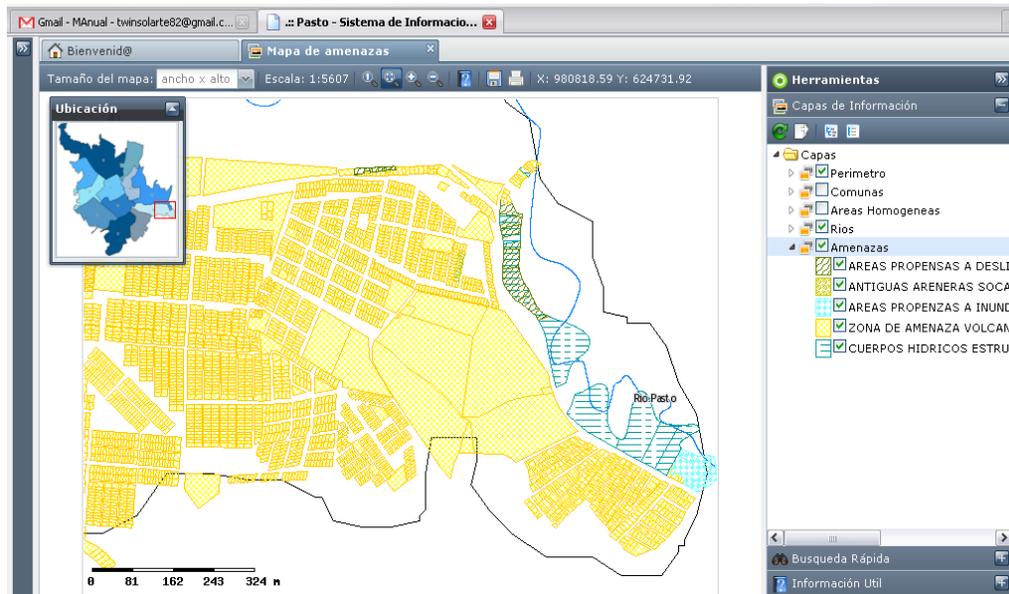


Figura 38. Mapa de Amenazas

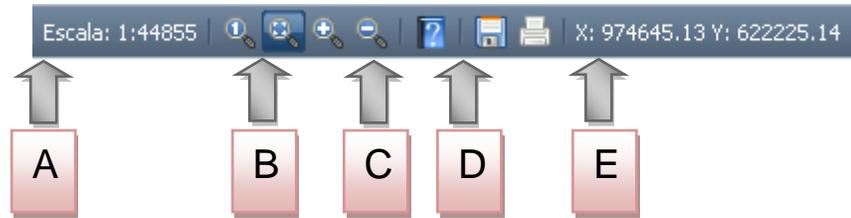


Figura 39. Barra de herramientas

Acción del Usuario	Respuesta del Sistema
<p>3. El usuario puede escoger las herramientas de navegación; entre las cuales se encuentra <i>Herramienta vista completa</i>, <i>Herramienta Navegar</i>, <i>Herramienta acercar</i>, <i>Herramienta alejar</i> (B).</p> <p>4. Se puede escoger la herramienta de Consulta (C)</p> <p>6. Se escogen las opciones de guardado <i>Guardar imagen</i>; <i>Exportar a Pdf</i> (D).</p>	<p>1. Se presenta la Escala en la cual se está navegando el mapa (A).</p> <p>2. Se presenta las coordenadas del punto en el cual se encuentra (E)</p> <p>5. Se despliega la pestaña de <i>Información útil</i>, del panel de <i>Herramientas</i>. Figura 44.</p> <p>7. Se presentan los diálogos de Acción pertenecientes al sistema Operativo.</p>

### 3.6 MÓDULO HERRAMIENTAS

#### 3.6.1. Capas de Información.

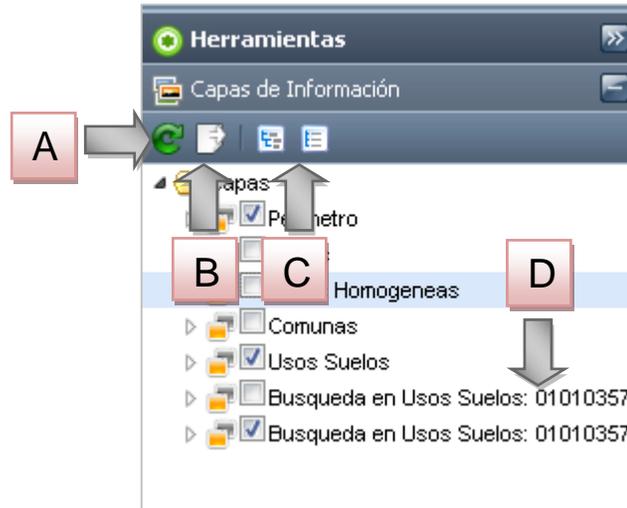


Figura 40. Pestaña Capas de Información

Acción del Usuario	Respuesta del Sistema
1. El usuario puede refrescar las capas del sistema (A). Figura 40.	2. Se presentan las capas en su estado original.
3. El usuario puede poner en estado inicial al mapa (B).	4. Se presentan las capas originales del sistema en su estado original, borrando las añadidas por las búsquedas (D).
5. El usuario puede expandir o <i>contraer</i> todos los árboles de las capas existentes(C).	6. Se Registran las capas de las búsquedas realizadas.

### 3.6.2 Búsqueda Rápida



Figura 41. Búsqueda Rápida

Acción del Usuario	Respuesta del Sistema
1. El usuario da clic en la pestaña <i>Búsqueda Rápida</i> (A). Figura 41.	2. Se presentan las opciones de <i>Capa</i> (B) y <i>Texto</i> (C).
3. El usuario Despliega las opciones de la capa. Figura 42	4. Se Presenta un cuadro informativo referente a la búsqueda ( Figura 43) y se añade en la pestaña de <i>Capas de Información</i> como una capa más, como puede observarse en la Figura 40, item (D)
5. El usuario puede <i>expandir</i> o <i>contraer</i> todos los árboles de las capas existentes(C).	



Figura 42. Datos de Búsqueda

numpredio	cc	apellidos
010104440019	41105218	HERNANDEZ QUIRO

Filtro

Figura 43. Resultado de la Búsqueda

### 3.6.3 Información Útil

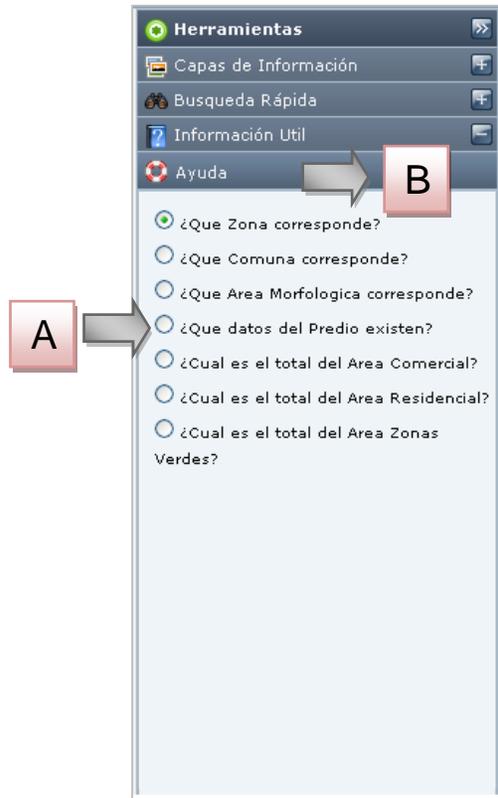


Figura 44. Información Útil

Acción del Usuario	Respuesta del Sistema
1. Se escoge el tipo de consulta que se va a realizar (A). Figura 44.	2. Se presenta una pantalla con el resultado de la consulta solicitada. Figura 45.
3. Se da clic en el botón de ayuda	3. Se presenta un cuadro explicativo. Figura 46.

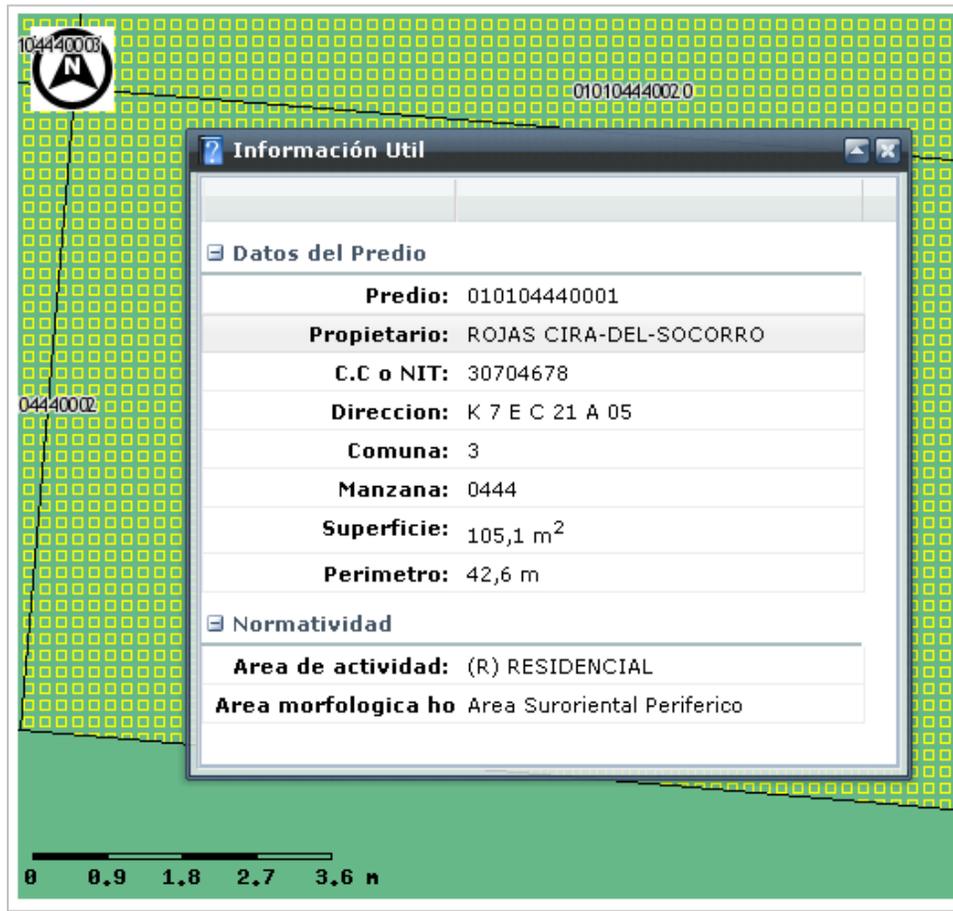


Figura 45. Resultado de Consulta

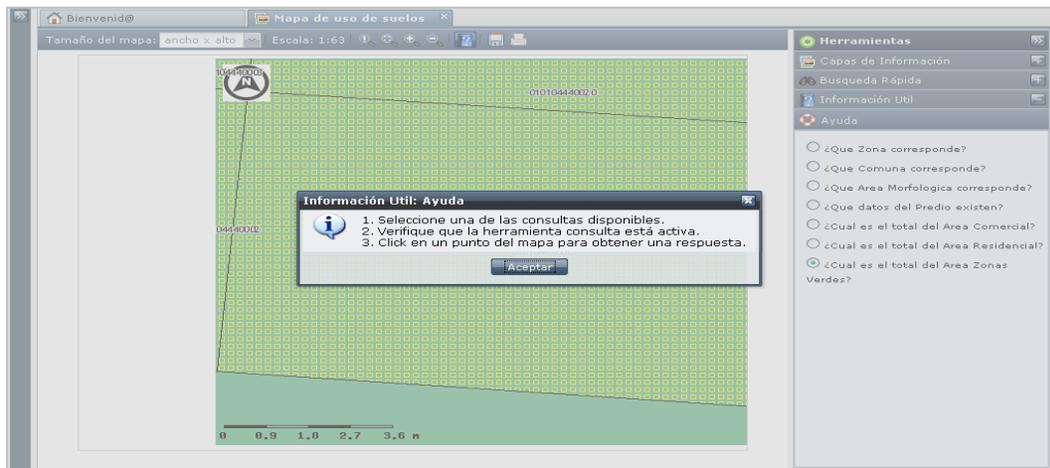


Figura 46. Cuadro explicativo de Ayuda

### 3.6.4 Opciones sobre el mapa



Figura 47. Opciones sobre el mapa

Acción del Usuario	Respuesta del Sistema
1. Se da clic derecho sobre cualquier parte del área del mapa	2. Se presenta un cuadro de opciones de herramientas Figura 47.
<b>NOTA:</b> Las herramientas que se presentan tienen las mismas funciones que las de la Figura A31	

## 4. PRUEBAS DE FUNCIONALIDAD

Para la realización de las pruebas de funcionalidad se utilizó el repositorio de datos de la información alfanumérica de la comuna 3 del municipio de Pasto, el cual fue cargado en la base de datos de GEOPASTO. Además se hizo un levantamiento de la información geográfica a partir de los mapas digitales suministrados por la Subsecretaría de Planeación los cuales se transformaron en datos espaciales para ser utilizados posteriormente a través de la extensión de objetos espaciales PostGIS.

Durante las pruebas de funcionalidad fueron evaluadas las principales características de GEOPASTO como sistema de información geográfica.

### 4.1 Pruebas de Visualización

El objetivo de estas pruebas es evaluar las capacidades de visualización de mapas, mediante las distintas herramientas o características que se encuentran presentes en la aplicación.

**4.1.1 Herramientas de navegación.** Se evaluaron las herramientas de navegación que permiten al usuario la interacción con el mapa, dentro de estas herramientas están: desplazamiento, acercamiento, alejamiento, vista completa y mapa de referencia.

En la figura 48 se observa la interfaz inicial sobre la cual comienzan las tareas del usuario.

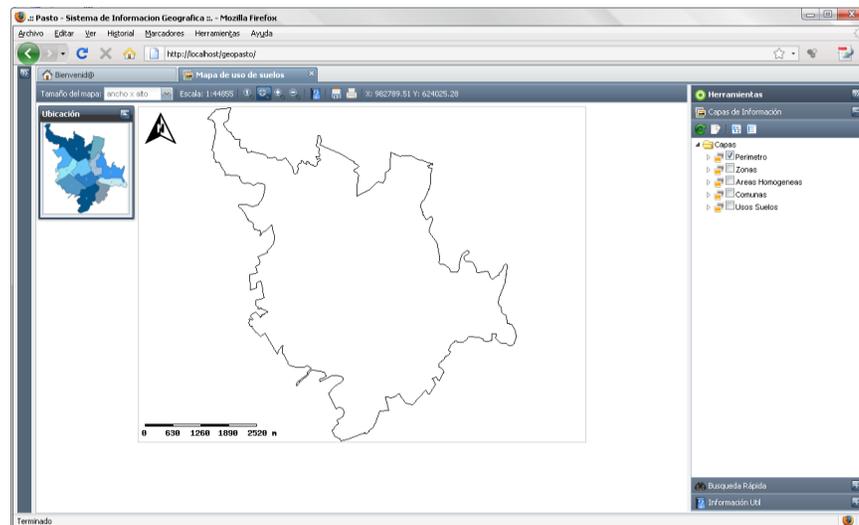
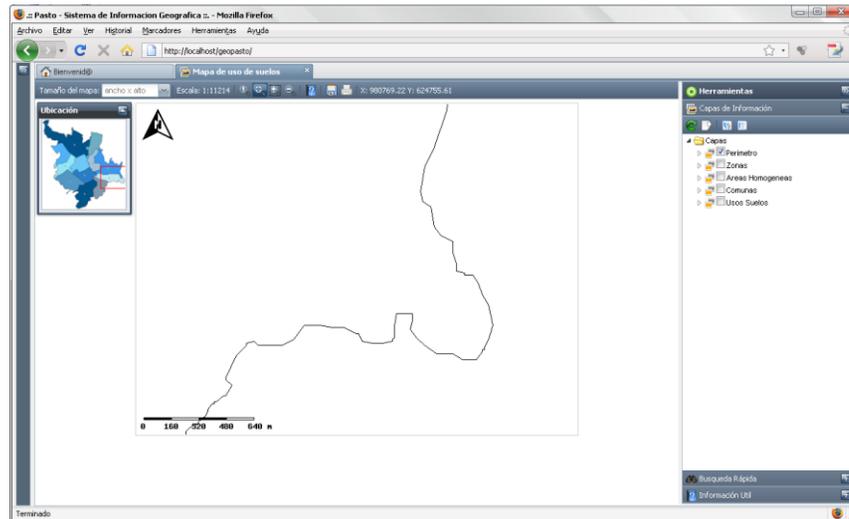


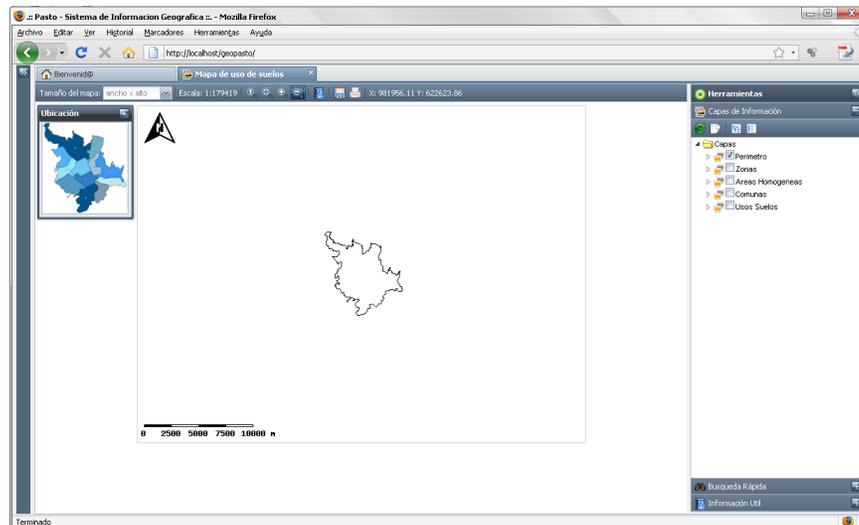
Figura 48. Estado inicial del mapa

A continuación, en la figura 49 se refleja el cambio luego de utilizar la herramienta de acercamiento. Podemos tener un mejor conocimiento de la ubicación actual observando el mapa de referencia.



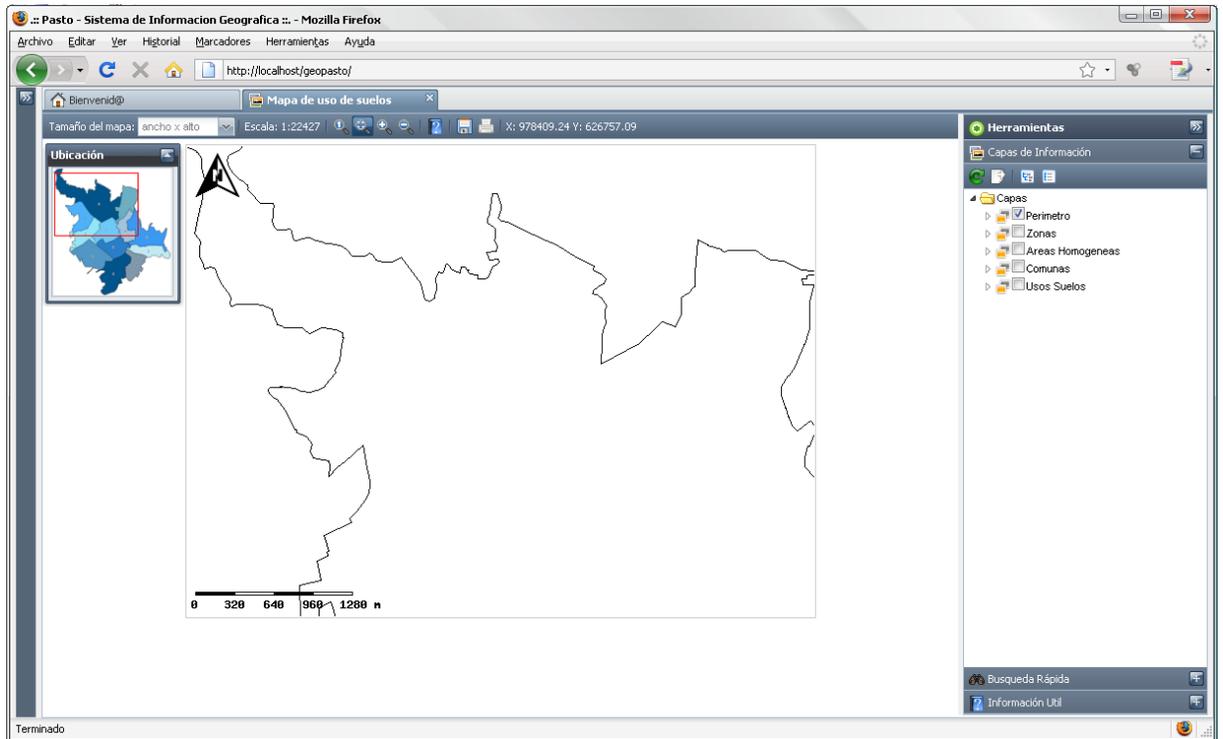
**Figura 49. Resultado de la herramienta acercar**

En la figura 50 se muestra el resultado después de utilizar la herramienta de alejamiento.



**Figura 50. Resultado de la herramienta alejar**

En las figuras 51 y 52 se ilustra el resultado después de aplicar la herramienta de desplazamiento.



**Figura 51. Resultado de la herramienta alejar**

En la figura 52 se visualiza un nuevo desplazamiento realizado sobre el mapa. Se observa claramente en el mapa de referencia el rectángulo de color rojo que corresponde a la sección del mapa visualizada actualmente.

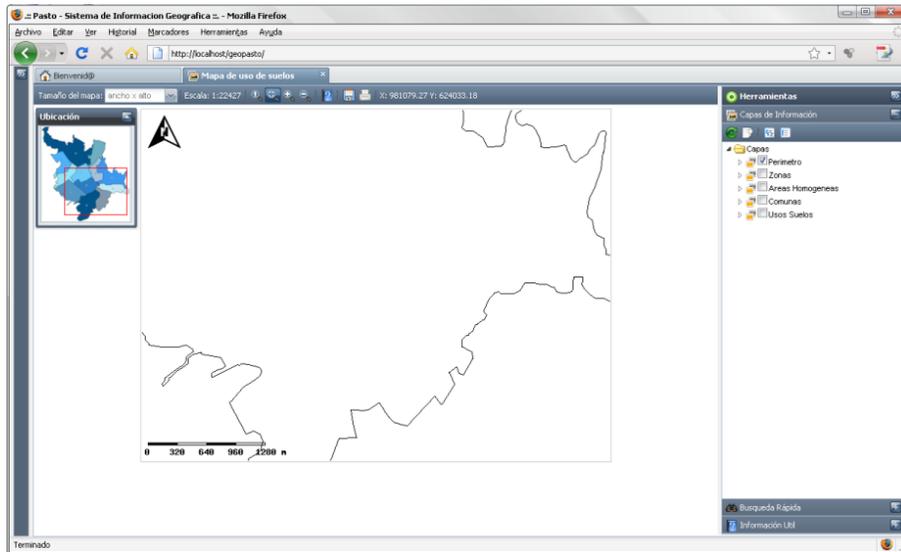


Figura 52. Resultado herramienta desplazamiento

**4.1.2 Herramienta Capas de información.** Fueron evaluadas las diferentes características del panel de capas de información, el cual permite al usuario interactuar con las distintas capas de datos que conforman el mapa.

La figura 53 corresponde al resultado de activar la capa de comunas del mapa.

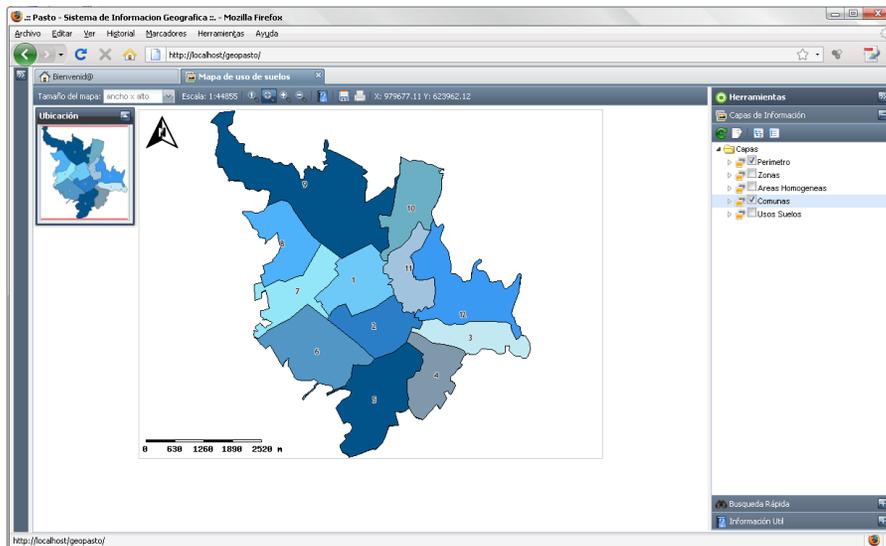


Figura 53. Resultado de la activación de la capa comunas

Otra de las características presentes en el panel de capas de información es, la posibilidad de habilitar o deshabilitar los ítems que componen una determinada capa. En las figuras 54 y 55 se muestran los resultados.

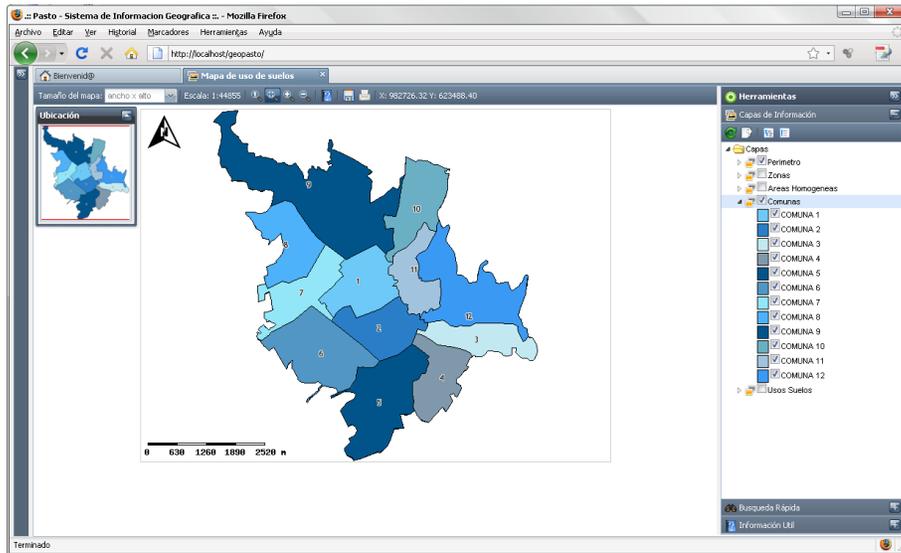


Figura 54. Visualización de ítems de la capa comunas

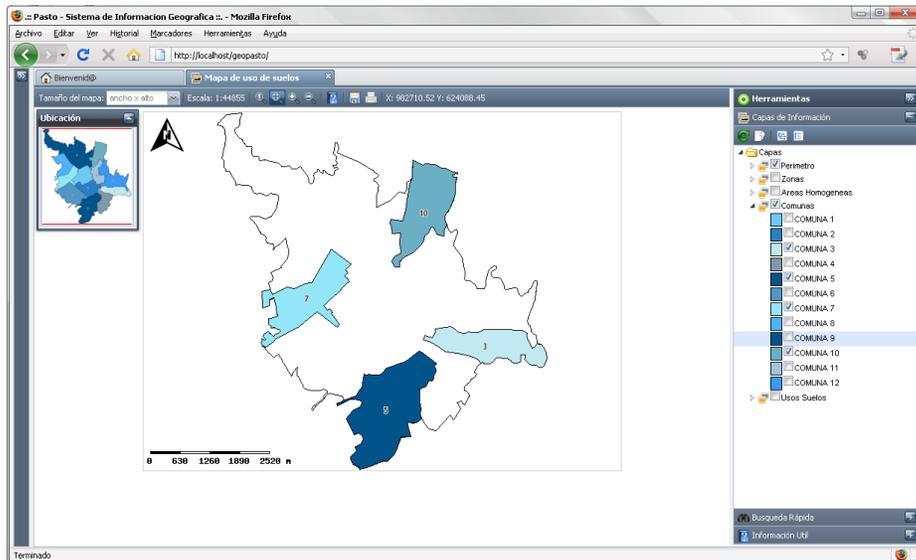


Figura 55. Resultado visualización de ítems activos de la capa comunas

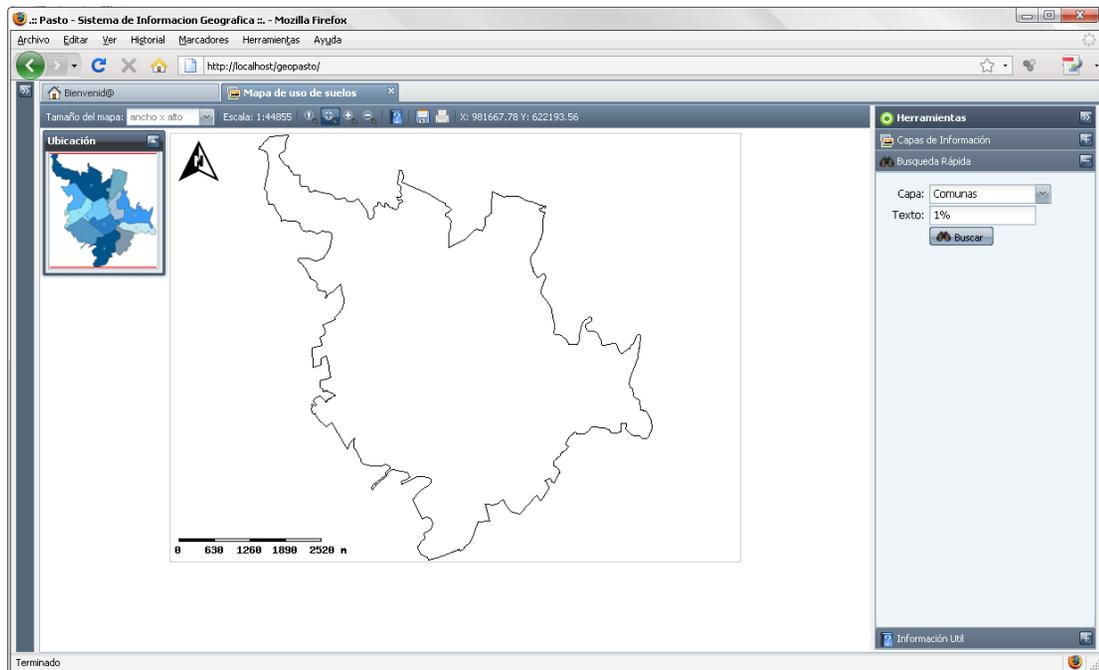
## 4.2 Pruebas de Consulta.

El objetivo de estas pruebas es evaluar las capacidades de consultar información correspondiente al mapa, mediante las herramientas disponibles para tal fin.

### 4.2.1 Herramienta Búsqueda Rápida.

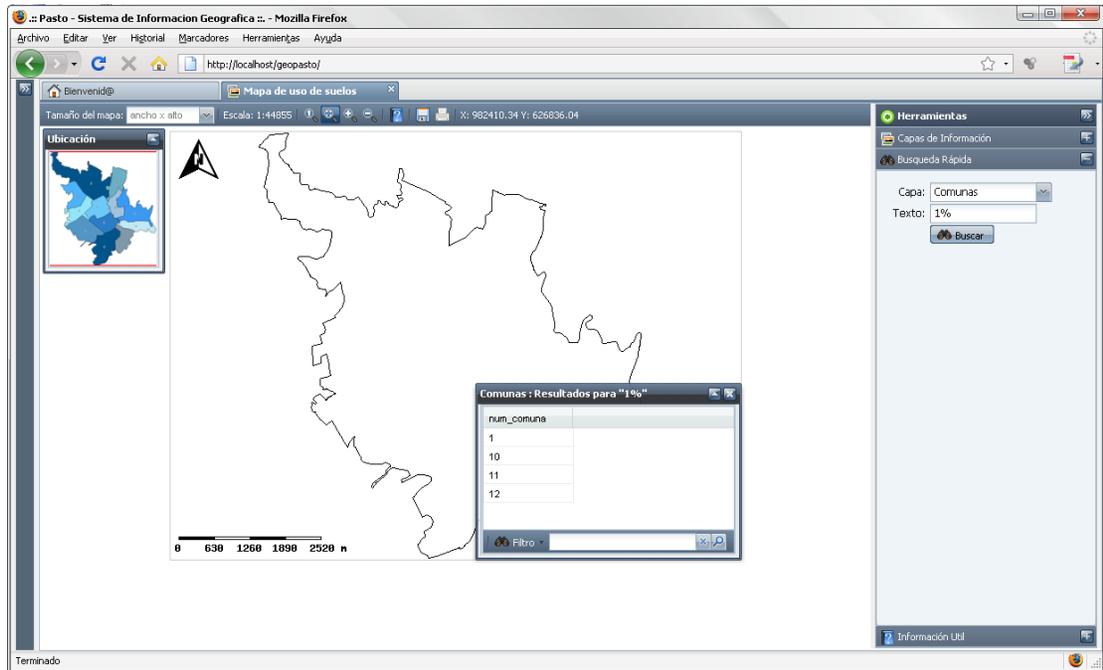
Se realizaron una serie de búsquedas sobre algunas de las capas disponibles en el mapa.

Las figuras 56, 57 y 58 muestran el panel de búsqueda rápida y su resultado tras realizar una búsqueda en la capa de comunas.

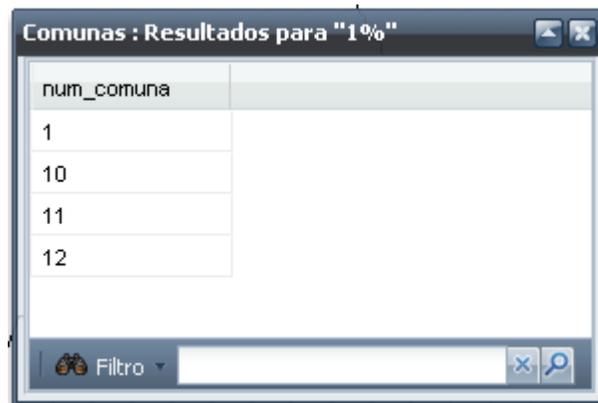


**Figura 56. Panel de búsqueda rápida**

Una vez proporcionada la información requerida, el sistema realiza la consulta en base a los atributos disponibles en la capa seleccionada. Se observa un resultado como el de la figura 57.



**Figura 57. Resultado ejecución de la búsqueda**



**Figura 58. Ampliación ventana con los resultados de la búsqueda**

Para ubicar algún resultado de la búsqueda, el usuario debe hacer doble click sobre el registro que desea ubicar en el mapa. El resultado de la consulta se muestra en las figuras 59 y 60.

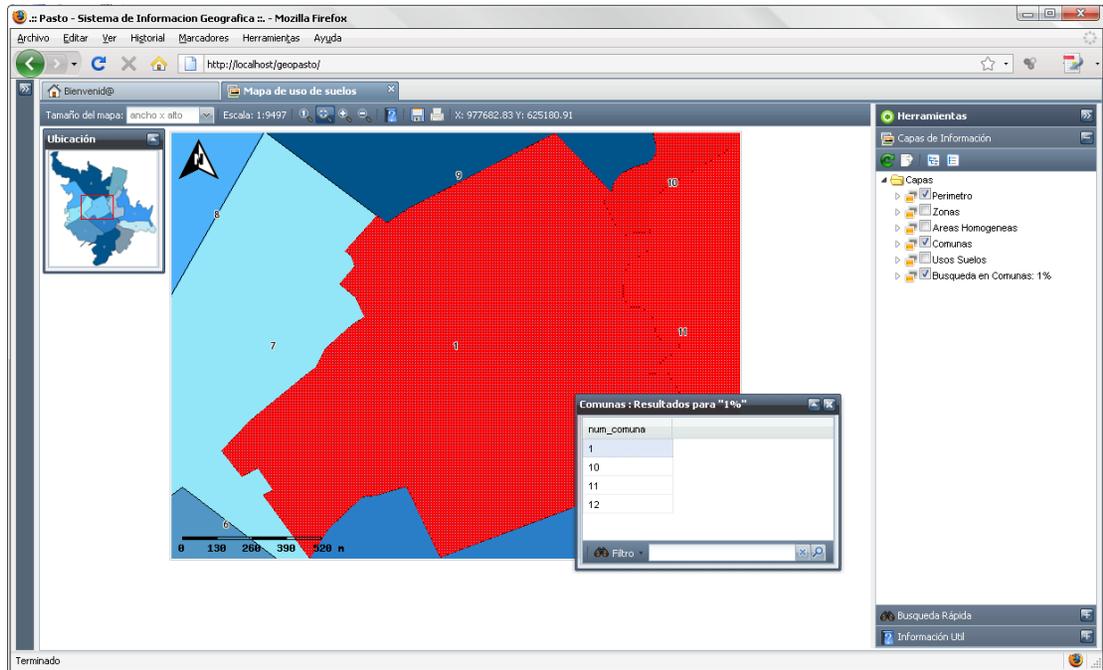


Figura 59. Ubicación de un registro del resultado de la consulta en el mapa

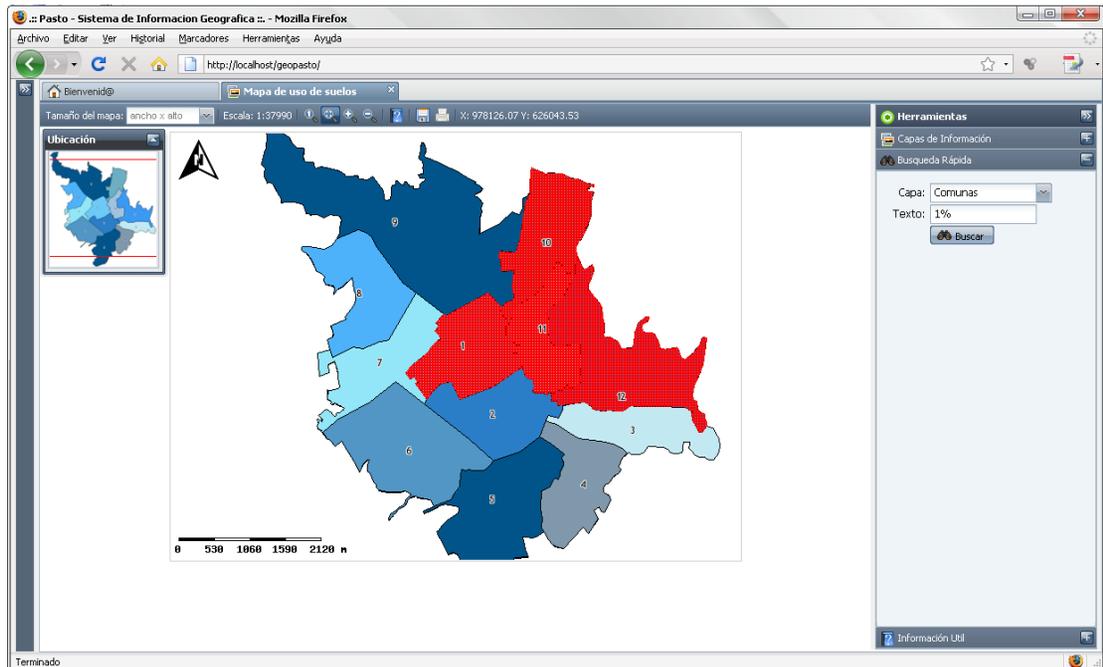


Figura 60. Visualización de todos los resultados de la consulta

Cada búsqueda realizada en el mapa, produce una nueva capa de información en el panel de capas. De esta manera el usuario logra tener un historial de todas las búsquedas realizadas (ver figuras 61, 62 y 63).

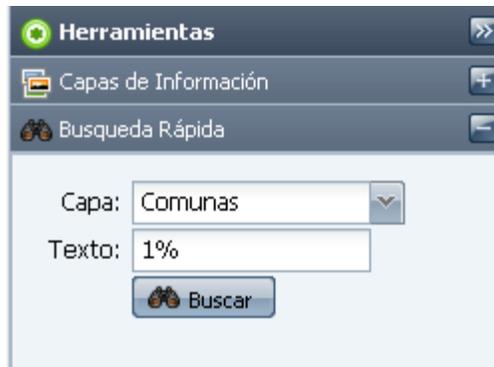


Figura 61. Ampliación del panel de búsqueda rápida

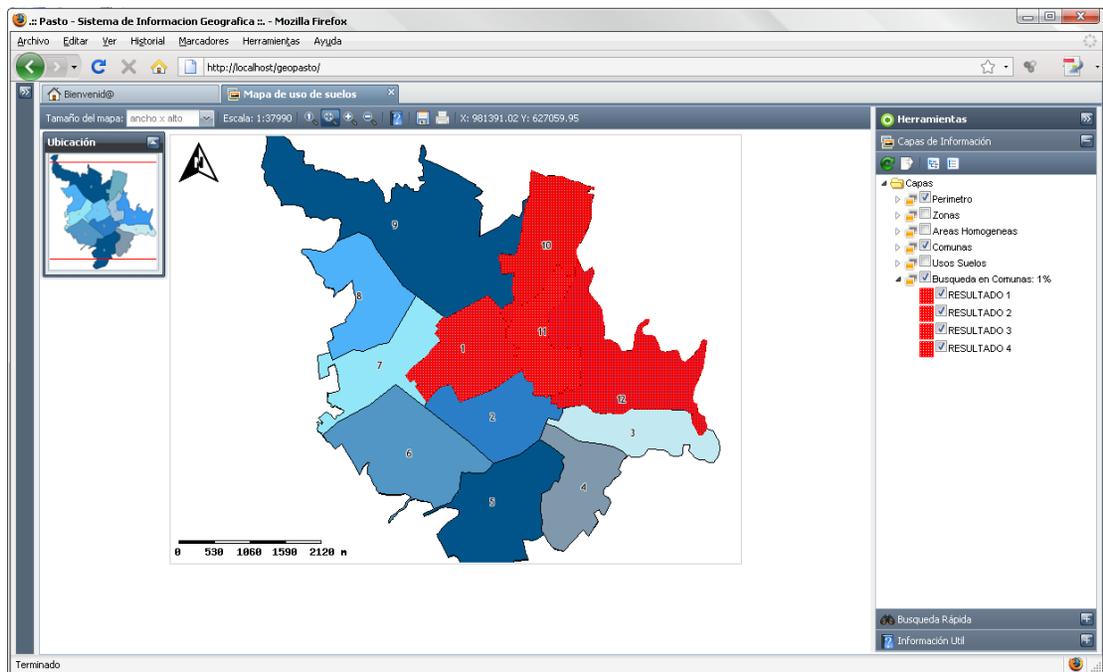
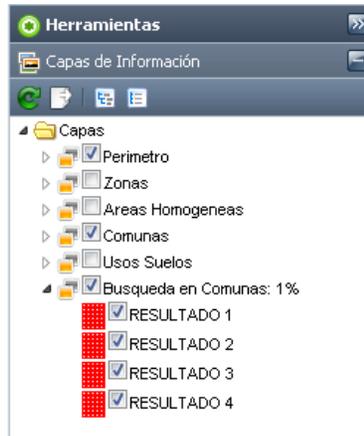


Figura 62. Representación de los resultados de la búsqueda como una nueva capa de datos

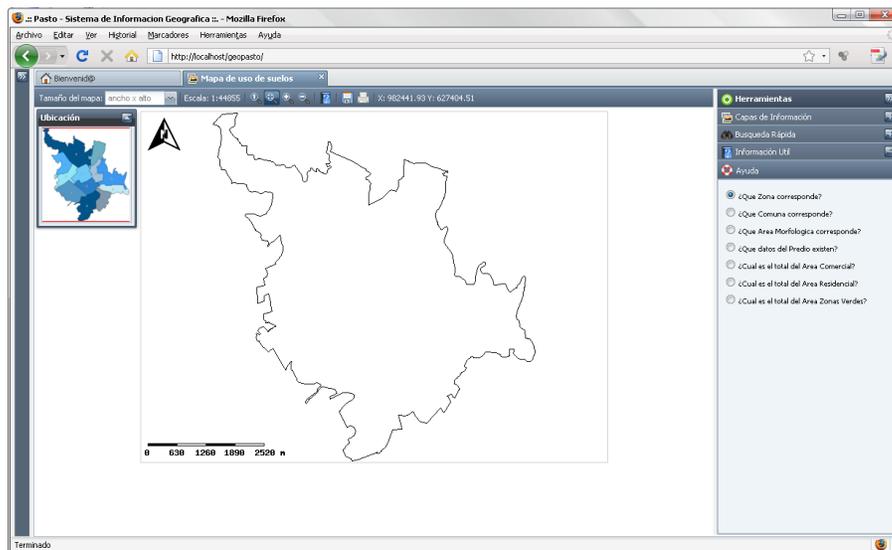


**Figura 63. Ampliación del panel de capas de información**

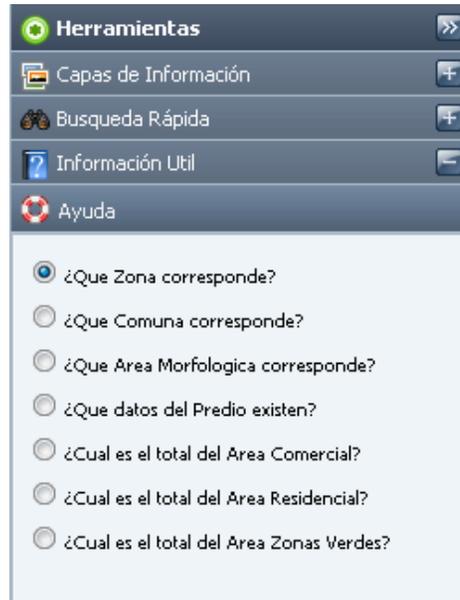
#### 4.2.2 Herramienta Información Útil

Se ejecutaron consultas especializadas correspondientes al un mapa en específico.

En la figura 64 se muestra el panel de información útil, el cual contiene las consultas especializadas representadas mediante preguntas (figura 65).

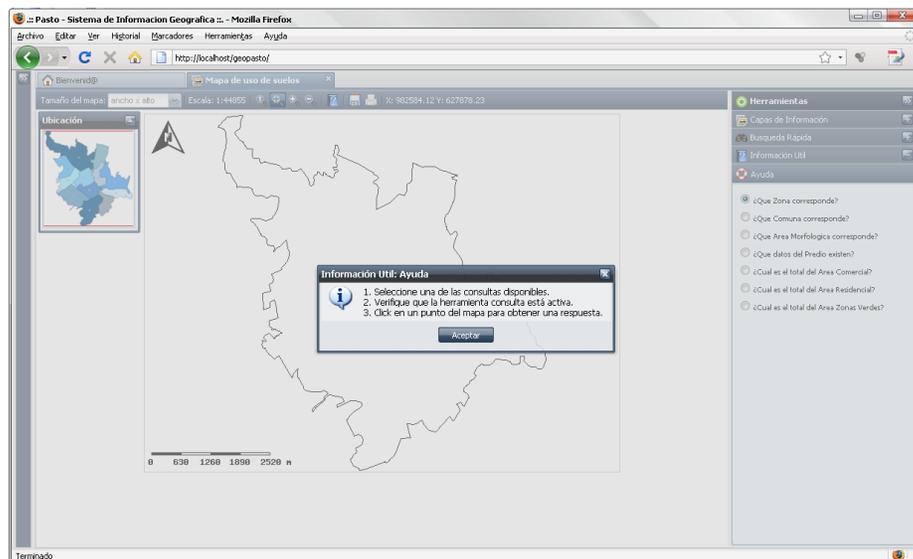


**Figura 64. Panel de información útil**

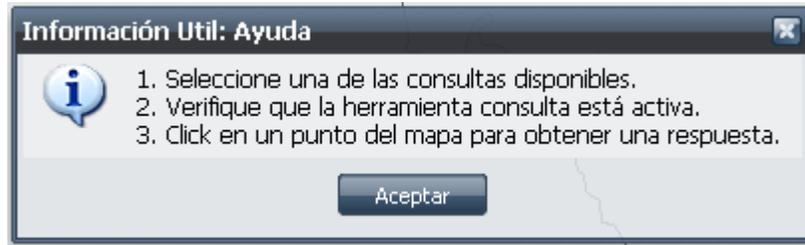


**Figura 65. Consultas especializadas disponibles en el mapa**

Presionando el botón ayuda del panel de información útil, el usuario recibe las indicaciones respectivas para poder ejecutar alguna de las consultas disponibles (figuras 66 y 67)..



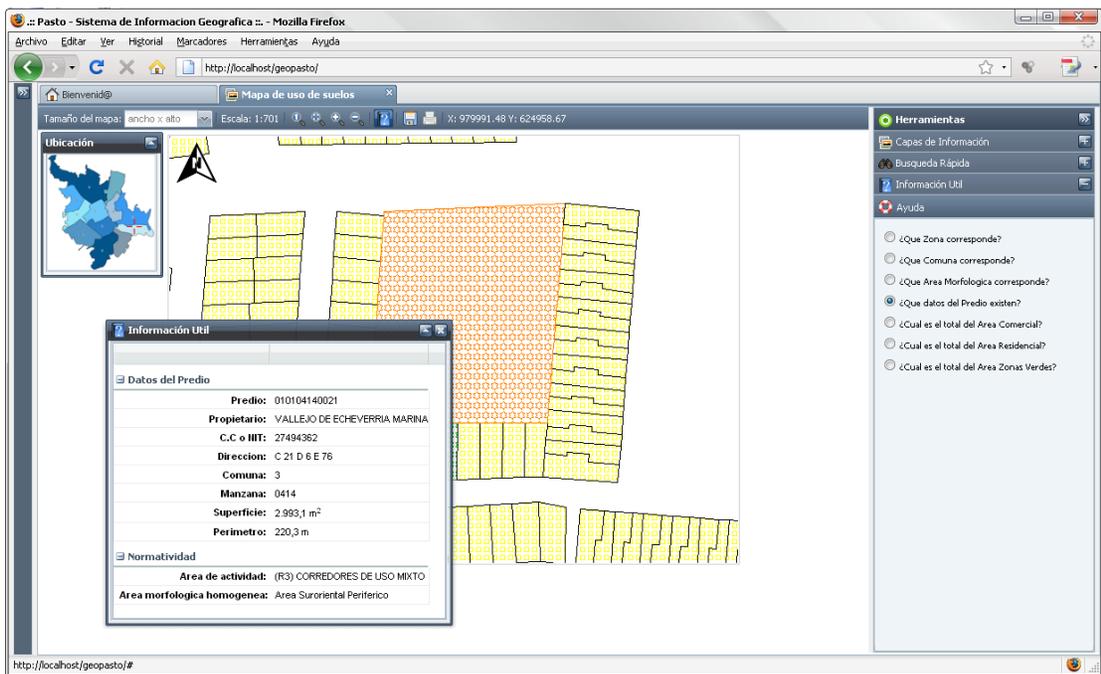
**Figura 66. Ventana de ayuda para la ejecución de consultas especializadas**



**Figura 67. Ampliación de la ventana de ayuda**

Para visualizar el resultado de las consultas realizadas durante las pruebas, se ha activado la capa de usos suelos. Ya que algunas de las consultas tienen relación con la información consignada en esta capa.

Las figuras 67 y 68 muestran el resultado de la ejecución de la consulta ¿Qué datos del predio existen?



**Figura 68. Resultado de la ejecución de la consulta en el predio de mayor tamaño**

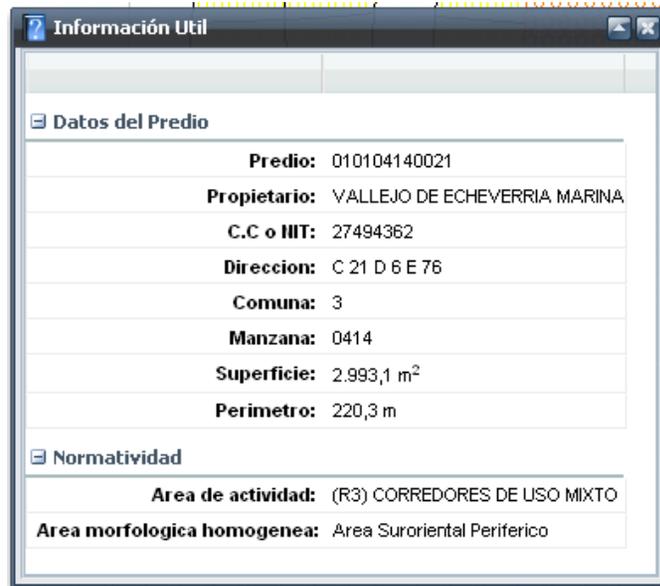


Figura 69. Ampliación de la ventana con la respuesta a la consulta realizada

Las figuras 70 y 71 muestran el resultado de la ejecución de la consulta ¿Cuál es el total del área residencial?

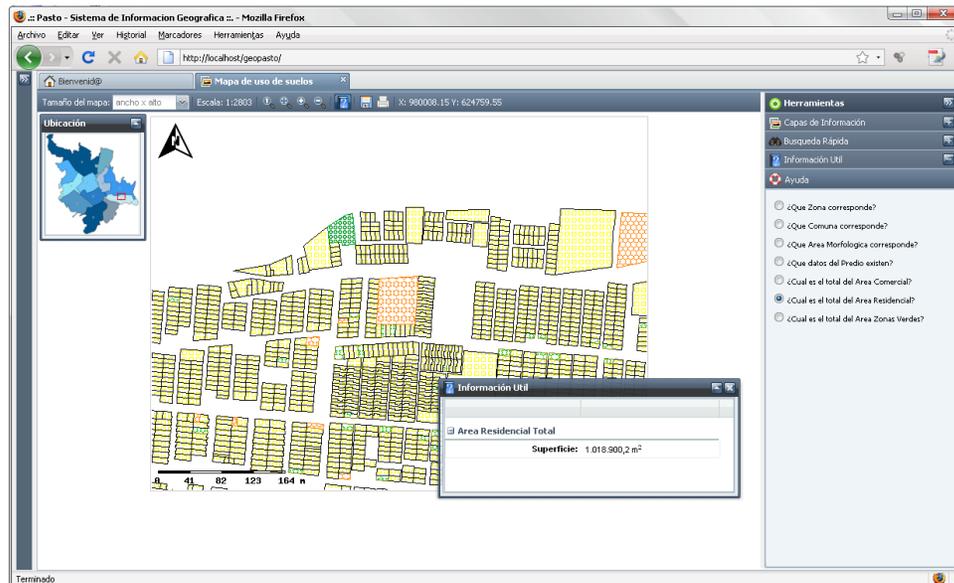


Figura 70. Resultado de la consulta que no depende de un punto específico



Figura 71. Ampliación de la ventana con el resultado de la consulta