

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA
ESCUELA DE COMPUTACION



**PROTOTIPO DE SISTEMA DE INFORMACIÓN GEOGRÁFICA
PARA LA UBICACIÓN DE PROYECTOS DE INVERSIÓN PÚBLICA,
CON APOYO A LA TOMA DE DECISIONES EN EL MUNICIPIO DE SAN
SALVADOR, EN AMBIENTE WEB.**

TRABAJO DE GRADUACIÓN
PARA OPTAR AL GRADO DE INGENIERO
EN CIENCIAS DE LA COMPUTACIÓN

PRESENTADO POR:
JENNIFFER ELIZABETH MARTINEZ ORDOÑEZ
ENRIQUE ALEXANDER SANCHEZ ZALDAÑA
CARMEN ELENA SUAREZ VELASQUEZ

Ciudadela Don Bosco, Febrero del 2004

Agradezco a mis padres con especial amor, por haber creído en mí, por haberme tomado de la mano en mis tropiezos, por todo su esfuerzo, dedicación y preocupación por alcanzar esta meta. A mi mamá por sus palabras siempre a tiempo y apoyo incondicional. A mi papá por enseñarme a través de su ejemplo a luchar, a ser responsable, trabajadora y tenaz. **A mi hijo** por ser el motivo de mi lucha diaria. **A mi esposo** por animarme a seguir siempre adelante y ser mi compañero en la vida. **A mi familia** por estar conmigo siempre y darme su cariño. **A mis compañeros y amigos**, Fran, Zaldi y Jenny, por haber trabajado juntos durante tantos años.

Carmen Elena Suárez Velásquez

Agradezco a Dios por haberme iluminado, por la fuerza otorgada para salir adelante durante todo el proceso y no dejarme caer, agradezco también por los dos grandes regalos entregados para el resto de mi vida mi hijo y mi esposo.

A mis padres por su amor, por todos los consejos expresados, por el incondicional apoyo brindado durante mi formación profesional para que siempre diera lo mejor de mí y mucho más, a **mi padre** por los sacrificios realizados y a **mi madre** por su preocupación, realmente gracias a los dos.

A mis amigos que siempre estuvieron ahí brindándome su amistad y apoyo incondicional y sin faltar el agradecimiento a mis compañeros Carmen y Zaldaña por haber luchado juntos en alcanzar nuestras metas con empeño y dedicación durante toda la carrera.

Jennifer Elizabeth Martínez Ordoñez

Alcanzar esta etapa es una realidad gracias al apoyo y colaboración de muchas personas, las cuales invirtieron y aportaron tiempo, palabras e ilusiones en mi. **Dios** me ha bendecido con ellas y con este logro, y es por ello que para él va mi mayor gratitud. Dedico esta labor **a mis padres**, Félix Sánchez e Hilda Zaldaña, por su constante y profundo apoyo, por tantos esfuerzos que en mi beneficio realizaron. Fueron sus enseñanzas, de una forma u otra, las que han contribuido predominantemente a conseguir este logro. Han sido también **mis compañeras** Carmen Suárez y Jenniffer Martínez, sin quienes hubiera sido muy distinto, difícil y aburrido. Me ayudaron a perseverar, y me halagaron con su compañía y confianza. También por la ayuda de los otros tantos **compañeros y amigos** de la Universidad, que a lo largo de este sendero académico me brindaron su luz y energía. Fueron igualmente útiles el interés y preocupación de **mis familiares y amigos**, quienes con sus preguntas y dudas, me impulsaron tenazmente a continuar. A todos, ¡Gracias!. Este es un logro que llena mi espíritu de alegría y ánimo ante el futuro. Serán muchas otras las metas que se presenten en mi camino. Seguramente no en todas triunfaré. Sin embargo, soy firme al pensar que nunca se deja de aprender. Entonces, aprender a ser mejor tras cada logro o fracaso será siempre mi meta final.

Enrique Alexander Sánchez Zaldaña.

INDICE

Capitulo 1. Definición de Proyecto	1
1.1 Definición de Tema.....	2
1.2 Justificación	3
1.3 Objetivos	4
1.4 Alcances	5
1.5 Limitantes.....	8
Capitulo 2. Análisis del Sistema.....	10
2.1 Recolección de datos.....	11
2.2 Determinación de la situación.....	15
2.3 Diagrama de flujos y procesos	16
2.4 Información Técnica y Operativa.....	28
2.5 Viabilidad de integración de información de otras instituciones con la aplicación a desarrollar	31
Capitulo 3. Investigación de conceptos y tecnologías.....	33
3.1 Generalidades de SIG	34
3.2 Implementación del concepto de SIG	38
3.3 Generalidades sobre tecnologías.....	40
3.4 Herramientas de diseño gráfico.....	49
Capitulo 4. Diseño del Sistema.....	54
4.1 Diseño del Mapa	55
4.2 Diseño de Entrada y Salida de Datos	59

4.3 Diseño y Creación de Base de Datos.....	68
4.4 Diseño Interfaz de usuario.....	80
4.5 Diseño Administración del sistema	83
Capitulo 5. Proceso de Desarrollo	87
5.1 Creación del Mapa.....	88
5.2 Creación Entrada y Salida de Datos	94
5.3 Creación Interfaz de usuario.....	112
5.4 Interactividad entre sistema, mapa y base de datos.....	114
Capitulo 6. Funcionalidad del Sistema	120
6.1 Registro de Obras	121
6.2 Visualización de Información	124
6.3 Navegación en el Mapa	127
Capitulo 7. Presupuesto de implementación	136
7.1 Inversión: hardware y software	137
7.2 Costos de mantenimiento mensual.....	137
7.3 Especificaciones del servidor	137
Capitulo 8. Conclusiones	138
Capitulo 9. Recomendaciones	140
Capitulo 10. Cronograma	143
Capitulo 11. Referencias	145

INDICE DE FIGURAS.

No.	DESCRIPCION	Pag.
2-1	Diagrama de flujo de datos. Nivel 0.	16
2-2	Diagrama de flujo de datos. Nivel 1.	17
2-3	Diagrama de flujo de datos. Nivel 2. proceso 1.1.	21
2-4	Diagrama de flujo de datos. Nivel 2. proceso 1.2.	22
2-5	Diagrama de flujo de datos. Nivel 2. proceso 1.3.	23
2-6	Diagrama de flujo de datos. Nivel 2. proceso 1.4.	24
2-7	Diagrama de flujo de datos. Nivel 2. proceso 1.5.	25
2-8	Diagrama de flujo de datos. Nivel 2. proceso 1.6.	26
2-9	Diagrama de flujo de datos. Nivel 2. proceso 1.7.	27
4-1	Criterios de selección para consulta	60
4-2	Pantalla resultado consulta mapa	61
4-3	Pantalla de captura para registro y modificación por obra.	64
4-4	Pantalla final de captura para el nombre de obra.	64
4-5	Pantalla Necesidades por obra.	65
4-6	Pantalla Proyectos por obra.	66
4-7	Pantalla Fuentes de financiamiento.	66
4-8	Pantalla Seguimiento de proyectos por obra.	67
4-9	Modelo Entidad-Relación.	68
4-10	Tablas utilizadas para el mecanismo de administración	84
5-1	Mapa digitalizado convertido en BMP.	88

5-2	Estructura de calles y manzanas.	89
5-3	Calles identificadas con nombre.	90
5-4	Mapa final y muestra de código SVG generado.	91
5-5	Selección de características para generación de reporte.	94
5-6	Pantalla de mapa con resultados de la consulta.	95
5-7	Reportes consolidados de acuerdo a las características del usuario.	96
5-8	Reportes Listado de proyectos con características específicas.	96
5-9	Pantalla de listado de necesidades.	97
5-10	Pantalla de captura de necesidades.	98
5-11	Pantalla actualización de necesidades	99
5-12	Pantalla consulta de proyectos.	99
5-13	Pantalla de captura de proyectos.	100
5-14	Pantalla actualización de proyecto.	102
5-15	Pantalla consulta fuentes de financiamiento.	102
5-16	Pantalla de captura fuente de financiamiento	103
5-17	Pantalla resultado de actualización fuentes de financiamiento	104
5-18	Pantalla consulta para programación y avances de proyecto	104
5-19	Pantalla captura avance	105
5-20	Pantalla resultado actualización de seguimiento	106
5-21	Pantalla listado de registros de la tabla seleccionada	107
5-22	Pantalla listado de registros de la tabla unidades_medida.	109

5-23	Pantalla Paso 1 para inserción de registros de la tabla unidades_medida.	109
5-24	Pantalla Paso 2 selección de archivo imagen.	110
5-25	Mensaje de confirmación de inserción.	110
5-26	Mensaje de advertencia.	110
5-27	Mensaje de confirmación de eliminación.	111
5-28	Menú generado para el usuario activo.	112
6-1	Mapa con resultados extraídos de la base de datos.	121
6-2	Panel con iconos para captura de obras	122
6-3	Pantalla de captura nombre de obra.	123
6-4	Panel de navegación	134
6-5	Mapa con resultados y disponible para navegación.	135

INDICE DE TABLAS.

No.	DESCRIPCION	Pag.
2-1	Requerimientos mínimos para implementar la aplicación.	28
2-2	Perfil de usuario de la aplicación.	29
3-1	Requerimientos para la instalación de Adobe Illustrator 10.	51
3-2	Resumen de tecnologías a utilizar.	52
4-1	Cantones.	69
4-2	Departamentos.	69
4-3	Estados.	70
4-4	Etapas.	70
4-5	Fuentes de financiamiento.	70
4-6	Instituciones.	71
4-7	Municipios.	71
4-8	Necesidades.	72
4-9	Obras.	72
4-10	Países.	73
4-11	Proyectos.	73
4-12	Fuentes de financiamiento-proyectos.	75
4-13	Responsables.	76
4-14	Seguimiento de proyectos.	77
4-15	Tipos de fuentes de financiamiento.	78
4-16	Tipos de proyectos.	78

4-17	Unidades de medida.	78
4-18	Menú del sistema.	80
4-19	Perfiles del sistema.	81
4-20	Relación perfiles-menu.	81
4-21	Relación perfiles-responsables.	82
4-22	Descripción de tablas	85
4-23	Descripción de campos de tablas	85
5-1	Propiedades modificables de elementos SVG.	115
6-1	Opciones del control de desplazamiento	130

INTRODUCCION.

El Gobierno de El Salvador ha venido impulsando la modernización del Estado, a través de reformas administrativas, legales, simplificación y sistematización de procesos, con el fin de brindar servicios públicos de calidad. Este gobierno es consciente de la necesidad de continuar fortaleciendo la capacidad de ejecución de las instituciones públicas, para proveer de manera más eficiente los servicios a su cargo, a efecto de adecuar su rol y funcionamiento a las necesidades que tiene el país, así como los requerimientos de la sociedad con transparencia, equidad y democracia.

La Secretaría Técnica de la Presidencia, cuya misión es la de fortalecer la toma de decisiones estratégicas, la coordinación de la gestión pública y la propuesta y difusión de normas y políticas de modernización, se le propone institucionalizar un Sistema de Información Geográfica para el Seguimiento de Proyectos, que este basado en delegación de funciones y autoridad, trabajo en equipo y la responsabilidad compartida por objetivos, Con la finalidad de recopilar toda la información de las Instituciones que desarrollan proyectos de inversión pública, para desplegarla geográficamente y hacerla visual a través de mapas, facilitando las consultorías y análisis de las diferentes actividades que se realizan y determinar dónde estaba, dónde está, y hacia dónde van las Instituciones.

El uso de información digital es parte del crecimiento de nuestra vida diaria. Más y más personas en el mundo entero están utilizando informaciones geográficas, sociales, económicas, políticas y ambientales para responder a preguntas prácticas en sus vidas. Para explorar y encontrar soluciones se utilizan una serie de herramientas electrónicas diseñadas para adquirir, presentar e interactuar con la información. Uno de estos se conoce como Sistema de Información Geográfica (SIG o GIS). Un SIG es un sistema diseñado para almacenar, actualizar, analizar, desplegar y manipular datos espaciales, información de lugares, objetos en nuestro planeta. Este sistema utiliza el poder de las computadoras para responder preguntas geográficas manipulando todo tipo de datos en diversas formas, como por ejemplo mapas, gráficas, reportes y tablas.

1.1 DEFINICIÓN DEL TEMA

Prototipo de Sistema de Información Geográfica, que permite ubicar en un mapa digital proyectos de inversión pública que atienden obras específicas, mostrar las necesidades no atendidas, características y seguimiento de ejecución del proyecto y las fuentes de financiamiento involucradas, con el fin de analizar la cobertura de atención a las necesidades de los habitantes de uno o varios sectores geográficos.

Este sistema busca contestar las siguientes preguntas:

- ¿En qué se está invirtiendo?
- ¿Dónde se está invirtiendo?
- ¿Cuánto se está invirtiendo?
- ¿Por qué se está invirtiendo?
- ¿Cuántos se están beneficiando?
- ¿Cuándo se invirtió?
- ¿Quién está invirtiendo?

1.2 JUSTIFICACIÓN.

Estudios recientes demuestran que alrededor del 80% de la información tratada por las empresas e instituciones oficiales tienen relación con localizaciones geográficas o coordenadas espaciales. Las decisiones que toman estos organismos dependen en gran medida de la calidad, exactitud y actualidad de la información, a menudo presentada en forma de mapas. Un SIG está diseñado para la colección, almacenamiento y análisis de objetos y fenómenos donde la localización geográfica es una característica importante o crítica para el análisis.

Cuestiones a las que responde un SIG:

Localización ¿Qué hay en...?

Condición ¿Dónde sucede que...?

Tendencias ¿Qué ha cambiado...?

Pautas ¿Qué pautas existen...?

Modelos ¿Qué ocurriría si...?

La Secretaría Técnica de la Presidencia de El Salvador actualmente cuenta con un Sistema para el Seguimiento y Evaluación de la Inversión Pública a nivel nacional, ubicando por municipio los proyectos de inversión pública realizados por las instituciones que reciben fondos del Estado.

A pesar de que dicho sistema puede proporcionar información de los proyectos de inversión pública que realizan las instituciones para cada municipio, estos no han sido ubicados en un mapa geo-referenciado, es decir, que no pueden ser visualizados en la ubicación exacta en que estos están o han sido desarrollados.

La ubicación visual de dichas obras en un mapa geo-referenciado, permite reforzar el análisis y la toma de decisiones estratégicas.

1.3 OBJETIVOS.

1.3.1 Objetivo General.

Desarrollar un Prototipo de Sistema de Información Geográfica, que proporcione información relacionada con la ubicación y desarrollo de proyectos, presentada a través de un mapa digital.

1.3.2 Objetivos Específicos.

- Ubicar gráficamente sobre un mapa, obras por desarrollarse o en proceso de desarrollo.
- Registrar información general de los proyectos, así como el seguimiento de su ejecución física y financiera, definiendo un modelo de datos que permita registrar dicha información.
- Establecer un proceso que permita integrar fácilmente el mapa de todo el país.
- Delimitar el mapa y su información por zonas de análisis (cantones, municipios, departamentos).
- Generar reportes que faciliten el análisis de estado de los proyectos así como información general y seguimiento.
- Proporcionar un sistema de acceso para las opciones del sistema, haciendo uso de menús personalizados.

1.4 ALCANCES.

Visualización gráfica en pantalla haciendo uso de mapas digitales, sobre la ubicación de obras físicas realizadas o por realizarse.

Esto significa que la ubicación de las obras físicas referenciadas con el sistema de coordenadas es almacenada en la base de datos, a partir de la cual los datos son extraídos para su visualización y posicionamiento al momento de mostrar resultados.

Cada elemento dentro del mapa esta definido de tal manera que ha sido creado como un punto, línea y/o polígonos que se codifica y almacena como una colección de coordenadas x,y. La ubicación de una característica puntual, tal como una perforación, se puede describir con un sólo punto x,y. Las características lineales, tales como calles, se almacenan como un conjunto de puntos de coordenadas x,y. Las características poligonales, tales como municipios y cantones, se almacenan como un circuito cerrado de coordenadas. El modelo vector es extremadamente útil para describir características discretas.

El sistema es configurable de tal forma, que permite la introducción de nuevos cantones, municipios, departamentos e instituciones.

EL mapa del Municipio de San Salvador ha sido creado a partir de herramientas de diseño que soportan el formato SVG, tomando cada uno de sus elementos para ser almacenado en la base de datos, permitiendo así que la información que se tiene del mapa puede crecer con el ingreso de nuevos departamentos, municipios y cantones, dado que la base de datos esta diseñada para esta escalabilidad. Así el funcionamiento del prototipo no es afectado si se hace la analogía entre lo que se maneja en este momento (*municipio-cantón*) a lo que más generalmente puede aplicarse (*departamento-municipio*).

Los mapas digitales cuentan con la opción de realizar acercamientos (zoom) y desplazamientos, para una mejor visualización de la información.

Por muy pequeño que sea nuestro país es difícil lograr visualizar cada parte que es mostrada, y para la finalidad de este prototipo de ubicar obras en un lugar específico ha sido necesario manejar una escala de acercamiento considerable para que sea lo más exacto posible, así como proporcionar al usuario una herramienta que le permita desplazarse para visualizar todo el mapa.

Mostrar información específica de cada obra de forma interactiva en el navegador o browser.

Cada obra ubicada en el mapa cuenta con la posibilidad que el usuario puede acceder a su información, esto puede realizarse dando clic en la obra ubicada en el mapa.

Diferentes usuarios pueden acceder el sistema.

Debido a que el sistema puede ser consultado y accedido por funcionarios de diferentes instituciones, se han establecido mecanismos de seguridad para garantizar la confiabilidad y seguridad de la información, mecanismos en los que cada usuario pueda acceder únicamente a datos restringidos, es decir ninguno de ellos puede manipular o visualizar incluso información perteneciente a otros usuarios.

Se cuenta con una interfaz para llevar el seguimiento de los proyectos (programación y ejecución física y financiera).

Se cuenta con un banco de necesidades por obra, que no cuentan con financiamiento para su realización.

Las necesidades son aquellos requerimientos de funcionamiento aun no presentes en las obras registradas y que son almacenados para cuando puedan ser expuestos en el momento que existan fondos o posibilidades de financiamiento por alguna entidad.

Es posible identificar cada una de las fuentes de financiamiento involucradas en el proyecto, así como el tipo de aporte que estas realizan (donación, préstamo, etc.).

El modelo de datos ha sido apegado a los procesos reales de la institución aunque los datos son ficticios.

El sistema puede ser accedido desde una computadora independientemente del sistema operativo que utilice, gracias a las ventajas de la tecnología multiplataforma que ofrece el lenguaje de programación JAVA y el lenguaje para la generación de páginas Web HTML.

1.5 LIMITACIONES

El prototipo de este sistema se ha hecho únicamente para el Municipio de San Salvador.

Los gráficos son mostrados únicamente en dos dimensiones.

Los SIG pueden tener la capacidad de presentar mediante funciones o métodos elementos espaciales (los datos espaciales incluyen puntos que representan cosas como unidades de salud, hospitales y escuelas, y líneas que representan cosas como calles, autopistas, ríos entre otros. También incluyen áreas políticas o administrativas, tales como límites de países, departamentos, municipios, y cantones para este caso particular). Estos elementos espaciales dependen del tipo de información cartográfica que se quieren mostrar, para este caso el tipo de información que se visualizada no requiere de parámetros minuciosamente calculados debido a que la posición de los elementos puede ser plana y sin volumen, tomando en cuenta también que todos los datos en la base de datos serán utilizados y presentados todos juntos, en el mismo sistema de coordenadas planas X,Y.

El sistema ha sido desarrollado para un óptimo funcionamiento del Internet Explorer, debido a que algunos parámetros de visualización cambian de explorador en explorador.

Para que el prototipo permita una interacción con el usuario ha sido necesario hacer uso de funciones JavaScript pero dicha funcionalidad varia de explorador en explorador, debido a que la compatibilidad aun no es un completo estándar.

Se ubican únicamente obras físicas que puedan ser posicionadas en un lugar específico, como escuelas, hospitales, unidades de salud, pozos, entre otros.

El sistema no hace uso de audio

El sistema no es capaz de realizar cambios estructurales al mapa, es decir, no se pueden incorporar nuevos elementos propios del mismo (dibujos) haciendo uso de la aplicación desarrollada, excepto los proyectos ubicados como obras físicas.

Para modificar el mapa, es necesario utilizar una herramienta de diseño gráfico, la aplicación a desarrollar no esta orientada a la creación y/o modificación de dibujos vectoriales.

2.1 RECOLECCIÓN DE DATOS.

Para comenzar el proceso de levantamiento de requerimientos, se contactó con la Lic. Ivania Avendaño de Garcilazo, de la Unidad de Seguimiento y Evaluación de la Secretaría Técnica de la Presidencia.

Los puntos tratados fueron los siguientes:

2.1.1 Tipo de información que manejan.

La información que esta unidad maneja se basa en la estructura de proyectos de inversión pública desarrollados por todas las instituciones que reciben fondos del estado.

Es importante tener el detalle de cada proyecto, su tipo, etapa, inversión programada, seguimiento mensual de la ejecución, fechas de inicio y finalización; estas últimas son importantes y necesarias para la realización de visitas de campo. Será importante además poder separar las inversiones por institución y por los diferentes rubros, y contar con la información organizada en regiones.

2.1.2 Procesos.

Actualmente la Unidad de Seguimiento y Evaluación cuenta con un sistema para el monitoreo de proyectos, el cual es accedido a través de internet por todas las instituciones, éstas cuentan con su respectivo usuario-contraseña para poder autenticarse en el mismo. Cada institución es responsable de alimentar el sistema con la información que le compete. Se ha diseñado un esquema para el seguimiento de proyectos que se acople a todas las instituciones, ya que, por la naturaleza y actividades de cada una de ellas, la ejecución de los proyectos y procesos internos varían entre si. Las instituciones están obligadas a efectuar informes mensuales.

2.1.3 Debilidades del sistema actual.

Una de las principales debilidades que se tiene en estos momentos es que se vuelve complicado asociar entre sí proyectos a partir de un listado generado, es decir, saber cuantos de ellos están atendiendo una misma obra, o si el mismo proyecto está cubriendo la misma necesidad pero en dos fases distintas; es entonces que se vuelve compendiosa la tarea de contabilizar el número de centros atendidos.

Otro punto que requiere de mucho trabajo es saber en donde exactamente esta ubicada la obra que se está atendiendo, esto es muy importante al momento de analizar la cobertura de servicios, podemos saber por ejemplo que en un municipio específico hay un monto de inversión y un listado de proyectos, pero se vuelve trabajoso contabilizar el número de unidades de salud atendidas o cualquier otro rubro.

Igualmente complejo es establecer por ejemplo, si una carretera se está rehabilitando, si está llegando ayuda oportuna a centros escolares u hospitalarios, de que manera esa calle o camino genera beneficios a la población o si antes de ser rehabilitada las personas tardaban más tiempo en llegar.

Estas debilidades ponen especial énfasis en la necesidad de un mecanismo de asociación de proyectos.

2.1.4 Salidas requeridas por esta unidad.

Una de las principales salidas es la ubicación de proyectos en el mapa de El Salvador para poder hacer un análisis más rápido de la cobertura de servicios. En este tipo de análisis vendría implícito el conteo de los diferentes centros atendidos por zona.

2.1.5 Usuarios.

Se tienen en estos momentos cuatro tipos de usuarios:

1. Usuarios responsables de ingresar la información al sistema. Dentro de cada institución se nombra a una persona responsable de llevar esta información, ninguna otra persona puede modificar los datos referentes a los proyectos que su institución realiza.
2. Usuarios de las instituciones que únicamente realizan consultas, éstos solamente pueden ver información relacionada con su institución.
3. Usuarios de la Unidad de Seguimiento y Evaluación, únicamente consultan, pero pueden ver la información de todas las demás instituciones.
4. El administrador del sistema y base de datos, quien es el encargado de mantener el sistema en operación, dar apoyo a los usuarios y dar respuesta a sus requerimientos.

2.1.6 Tecnología de Secretaría Técnica de la Presidencia.

Se cuenta con un servidor que tiene como sistema operativo Windows 2000, el sistema trabaja con servlets de Java con ServletExec como contenedor y el IIS como Web Server. Microsoft SQL Server 2000 para la base de datos

2.1.7 Integración del sistema actual con las diferentes instituciones.

La Unidad de Seguimiento y Evaluación prefiere manejar este tema a través de consultorías especializada, como se ha hecho mención, cada una de las instituciones trabaja en base a los recursos tecnológicos con los que cuenta, y esto varía en gran

manera entre una y otra, y para llevar a cabo dicha integración es necesario un estudio y análisis profundo de los procesos internos de cada entidad, así como el volumen y características de la información generada.

2.2 DETERMINACIÓN DE LA SITUACIÓN.

2.2.1 Identificación del problema.

Se identificó que el principal problema a resolver es la manipulación de la información para su consolidación y presentación, ya que para generar reportes de proyectos ubicados en el mapa, como para realizar el conteo de las obras, se realiza en estos momentos de manera manual, lo cual consume una gran cantidad de tiempo de las personas que realizan los análisis.

2.2.2 Oportunidad de mejora.

Con la implementación de un sistema de información se evita que manualmente se tenga que estar identificando cada proyecto con su ubicación en el mapa y haciendo el conteo de las obras atendidas por cada sector.

2.2.3 Objetivos del sistema en la Institución.

Se definen entonces los siguientes objetivos del sistema para la institución:

1. Reducir errores y mejorar la precisión de la entrada de datos.
2. Acortar el tiempo de procesamiento de datos.
3. Automatizar los procedimientos manuales para reducir errores, aumentar la velocidad y precisión, disminuir el tiempo requerido por el empleado.

2.3 DIAGRAMA DE FLUJO DE DATOS Y PROCESOS.

2.3.1 Diagrama de Flujo de datos. Nivel 0.

El usuario ingresa al sistema y proporciona a éste la información necesaria para su sesión. De esa forma, puede proporcionar datos de identificación, parámetros para consultas y reportes, solicitud de mapa, cambios o actualizaciones de obras/necesidades, tareas de mantenimiento u otras tareas, según se visualiza en la Figura 2-1.

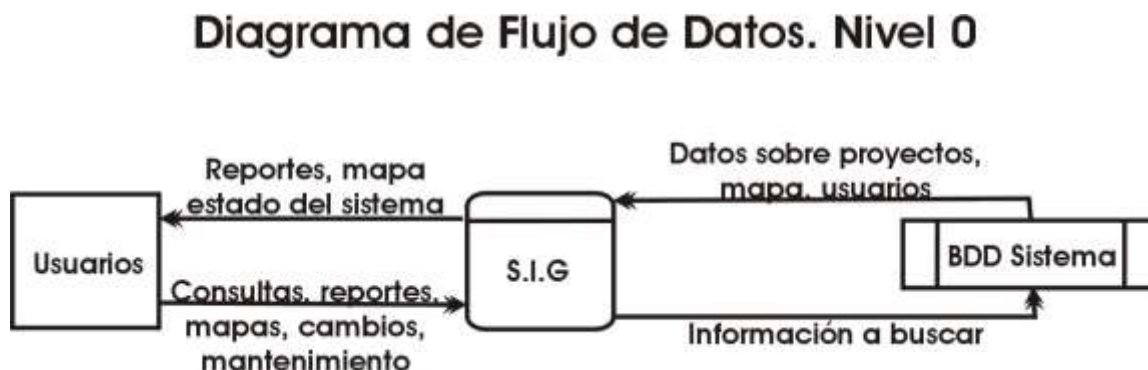


Figura 2-1. Diagrama de Flujo de Datos, nivel 0

Las tareas que el usuario puede realizar dependen del tipo o nivel de acceso correspondiente a la clave con la cual haya ingresado al sistema.

El sistema entonces accede a la base de datos que almacena la información y, a través de sus procesos, obtiene, modifica y/o presenta al usuario lo que éste haya solicitado. Las posibles salidas que el usuario puede obtener serían, por ejemplo: mapa en pantalla, reportes, mensajes de estado, etc.

2.3.2 Diagrama de Flujo de datos. Nivel 1.



Figura 2-2. Diagrama de Flujo de Datos, nivel 1

Procesos.

Proceso 1.1: Verificar usuarios.

Mediante este proceso se determina el nivel de acceso del usuario que desea registrarse. En base a esta información se definen las herramientas y procesos que el usuario puede acceder durante su uso del Sistema. Esencialmente se definen tres tipos de usuarios: de consulta, de actualización y de mantenimiento. Dentro de cada uno de estos tipos (excepto mantenimiento) se distinguen otras clasificaciones,

correspondientes a las instituciones que realizan tareas de consulta o modificación de proyectos de inversión pública.

Proceso 1.2: Interfaz de Consulta.

Diseñada específicamente para el tipo de usuarios de consulta, que solamente tiene autorización de pedir datos sobre los proyectos, y no les compete la actualización o modificación de información de los mismos. Una interfaz de petición de información, que permite la visualización gráfica de resultados en el mapa, así como generación de ciertos reportes.

Proceso 1.3: Interfaz de Actualizar.

Orientada al tipo de usuarios de actualización, quienes cuentan con el nivel de autorización necesario para modificar datos acerca de los proyectos/necesidades registrados en el sistema, así como la inclusión de nuevos proyectos. Permite adicionalmente la generación de reportes acerca de información de los proyectos.

Proceso 1.4: Interfaz de Mantenimiento.

Creada para la persona encargada de la administración del Sistema. Permite por ello, la manipulación de los archivos relacionados al mapa y su configuración, a fin de garantizar el óptimo funcionamiento de la interfaz gráfica del sistema para los usuarios.

Proceso 1.5: Generar Reportes.

Permite definir las características, formato e información relacionada a los reportes solicitados por el usuario. Este proceso accede la base de datos a través del proceso "Servidor de Proyectos", el cual adquiere los datos del proyecto y los pone a disposición del usuario.

Proceso 1.6: Visualizar mapa.

Encargado de las tareas de visualización del mapa. Manipula la forma y estilo de la observación de coordenadas del mapa y accede la base de datos del mismo para adquirir de ella la información necesaria para cubrir las preferencias de visualización del usuario.

Proceso 1.7: Servir Proyectos.

A través de este proceso, se accede la información que se encuentra almacenada acerca de los diferentes aspectos y elementos que conforman los diversos proyectos. Este acceso se restringe de acuerdo al usuario que accede la aplicación.

Usuarios.

1. Usuario Consulta.

Personas autorizadas que desean obtener información acerca de algún proyecto u obra determinada. Cuenta con permisos exclusivamente para solicitud de datos y preferencias de visualización del mapa, y no se le permite el acceso ni modificación de los proyectos/obras, ni del mapa. Existe una sub-clasificación de acceso a los proyectos, de acuerdo a la institución a la que el usuario pertenece.

2. Usuario Actualización.

Se incluyen en esta clasificación los usuarios que cuentan con el nivel de autorización necesario para actualizar la información de los proyectos/obras de las instituciones a las que están igualmente autorizados a acceder. Se permite así a este usuario la modificación y actualización de datos sobre los proyectos que le competen.

3-Usuario Mantenimiento.

Es la persona encargada o autorizada para la administración del Sistema. Por ello, se le permite la modificación y actualización de los aspectos relacionados al mapa, así como a la organización y métodos de acceso a la información de los proyectos/obras.

Bases de Datos.

1. Base de datos sistema.

Acá se almacena la información relacionada a los proyectos/obras, sus características y detalles. Se deposita también en esta base de datos la información acerca de los usuarios y sus niveles de acceso. Esta información no es accedida directamente por los usuarios. El proceso de lectura de datos almacenados se realiza mediante los procesos descritos anteriormente.

2. Base de datos mapa.

Todo lo relacionado al mapa, sus propiedades y características. Cada uno de sus elementos encuentra su representación acá. De igual forma, esta información no puede ser accedida más que a través de los procesos diseñados con esa función.

2.3.3 Diagrama de Flujo de datos. Nivel 2.

Proceso 1.1: Verificar usuario.

Mediante este proceso se valida la sesión del usuario. Se inicia introduciendo las claves correspondientes al usuario y su contraseña. Esta información es recibida por el proceso "Recibir info usuario", el cual valida su sintaxis básica y pasa esos datos a "Identificar usuario", como se observa en la Figura 2-3.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.1: Verificar usuario

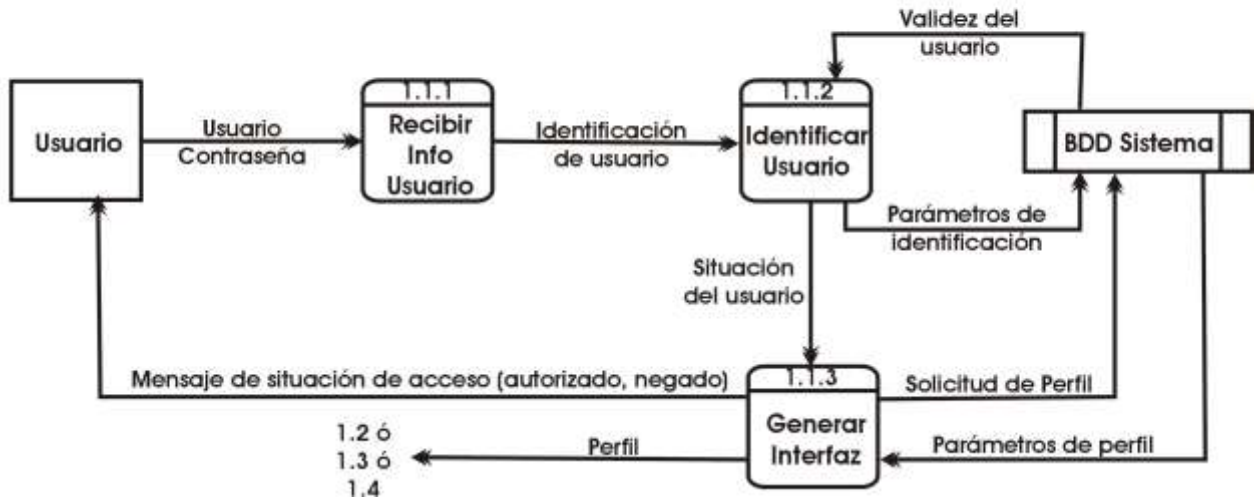


Figura 2-3. Diagrama de Flujo de Datos, nivel 2, proceso 1.1

Este proceso accede la Base de datos en busca del usuario que desea conectarse, y obtiene entonces el estado de validez del mismo. Esta información se transmite a “Generar Interfaz”, que produce un mensaje que refleja el estado de validez de la conexión (autorizado, denegado). De ser autorizada la conexión, se consulta la Base de Datos para obtener la base del perfil correspondiente a ese usuario, se proporcionan estos datos al proceso “Generar Interfaz”, y se proporciona el perfil de interfaz generado al siguiente proceso.

Proceso 1.2: Interfaz usuario consulta.

Este proceso es el que realiza la tarea de interacción con el usuario, es entonces el proceso con el que el usuario verá sus peticiones y modificará algunas de sus características.

Una vez el usuario está dentro del sistema, interactúa con “Enlazar usuario” como se visualiza en la Figura 2-4. A este proceso es al que el usuario solicita reportes, mapas, modificación de formatos, etc.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.2: Interfaz usuario consulta.

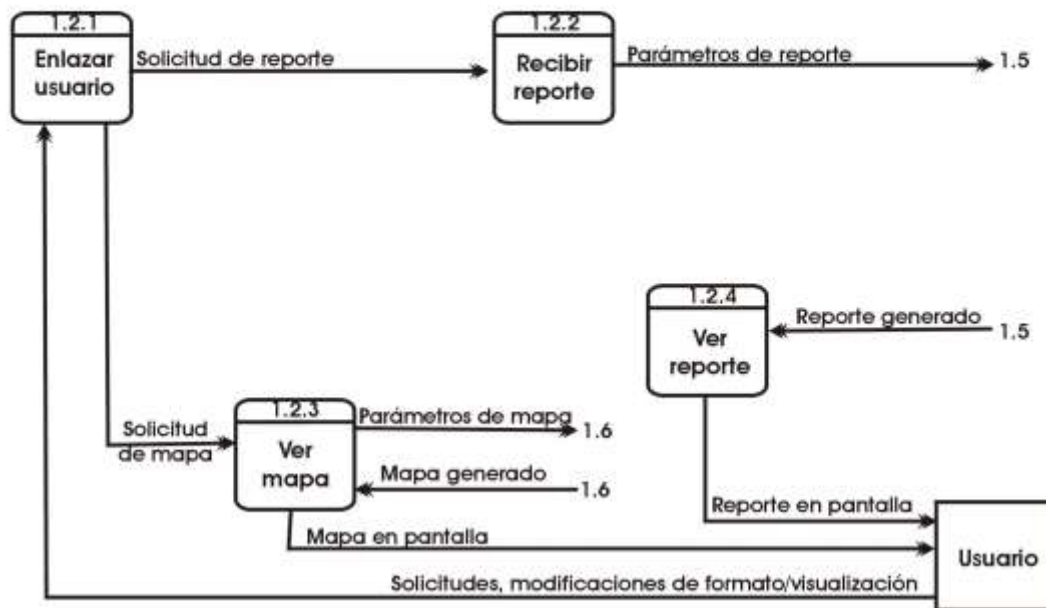


Figura 2-4. Diagrama de Flujo de Datos, nivel 2, proceso 1.2: Interfaz usuario

Si el usuario necesita un reporte, el sub-proceso “Recibir reporte” obtiene los parámetros solicitados por el usuario para el reporte y pasa esa información al proceso 1.5: “Generar reporte”, que lo genera. Una vez creado el reporte, se proporciona al sub-proceso “Ver reporte”, que lo muestra en pantalla al usuario.

Si el usuario solicita ver un mapa, se le muestra el mapa con valores predeterminados. Si se desea cambiar la vista del mapa, se pasan estos datos al proceso 1.6 “Visualizar mapa”, que genera el mapa y lo entrega a “Ver mapa” para su vista en pantalla. Cualquier modificación es proporcionada por el usuario al sub-proceso “Enlazar usuario”, que se encarga de entregarla al proceso correspondiente.

Proceso 1.3: Interfaz usuario actualizar.

Este proceso funciona de la misma manera que la interfaz explicada anteriormente: interfaz usuario consulta. El manejo de solicitud de reportes y mapa se manipula de la misma forma. Sin embargo se adhiere la capacidad de agregar nuevas obras,

proporcionar su información y modificarlas. Para ello, se creó un sub-proceso denominado “Recibir nueva obra”, en el que se reciben los datos de la nueva obra/necesidad/fuente de financiamiento, según se observa en la Figura 2-5. Mediante esta etapa se accede a la Base de datos para almacenar o modificar la información recibida por el sub-proceso. El estado y producto de estas operaciones se informan al usuario mediante mensajes de resultados, que indican si la tarea solicitada fue realizada exitosamente o no. En algunos casos, estas modificaciones pueden implicar un cambio en la vista del mapa, por lo que se brinda al proceso 1.6: “Visualizar mapa”, nuevos parámetros para el mapa.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.3: Interfaz usuario actualizar.

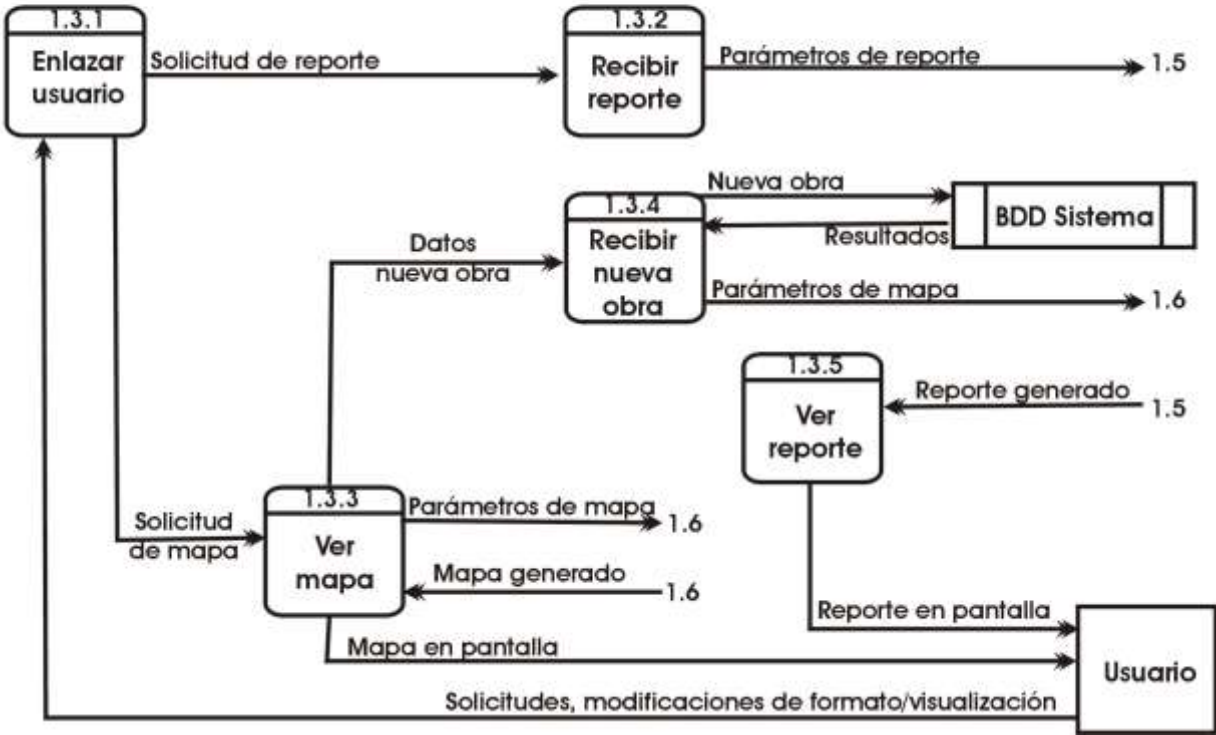


Figura 2-5. Diagrama de Flujo de Datos, nivel 2, proceso 1.3: Interfaz usuario actualizar

Proceso 1.4: Interfaz usuario mantenimiento.

Este proceso es el orientado para el usuario encargado del mantenimiento del sistema. Inicialmente funciona de manera similar a los procesos de interfaz explicados anteriormente, con la adición del acceso a la Base de Datos mediante el sub-proceso “Monitorear/Mantener BDD”, con lo que se le permite a la persona modificar información de la misma, adición de usuarios, perfiles, etc. Ver Figura 2-6.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.4: Interfaz usuario mantenimiento.

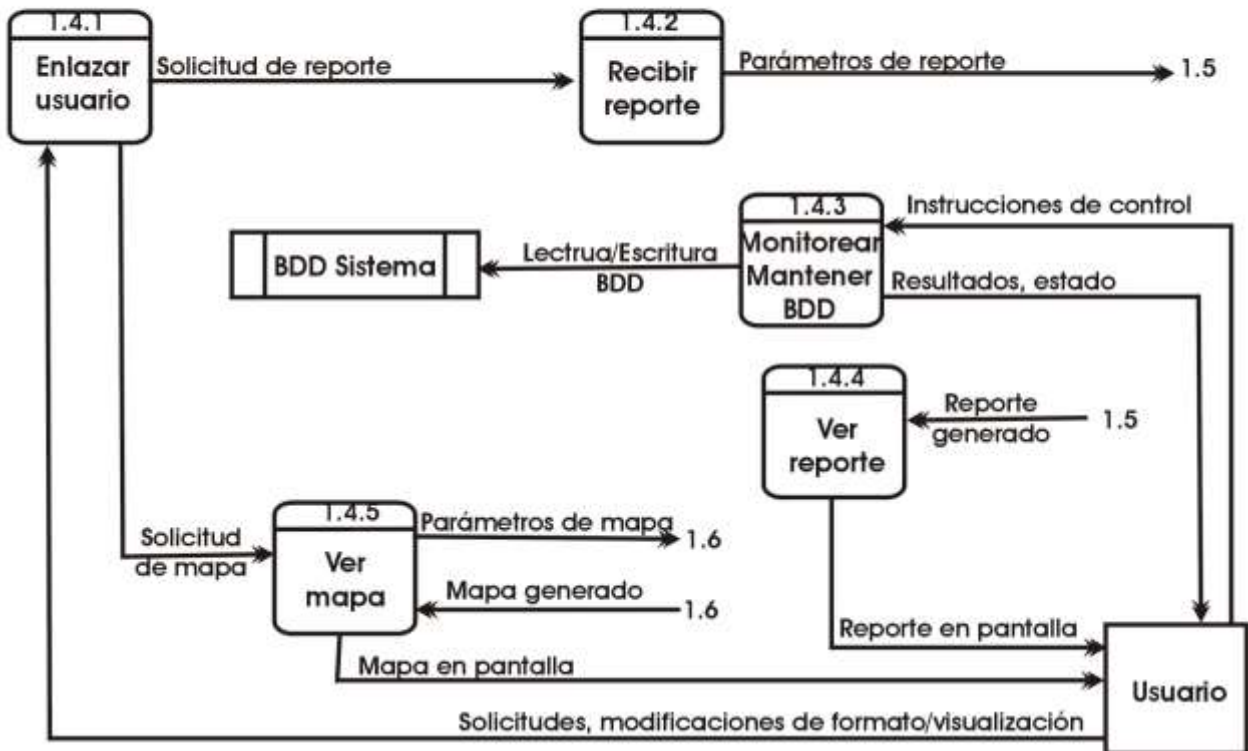


Figura 2-6. Diagrama de Flujo de Datos, nivel 2, proceso 1.4: Interfaz usuario mantenimiento.

Proceso 1.5: Generar reportes.

A este proceso llegan las solicitudes de reportes y consultas que el usuario realice. Se proporcionan entonces al sub-proceso “Ubicar Info”, los parámetros para la generación del reporte. Este obtiene de esos parámetros, la información de los proyectos que

necesita, y lo solicita al proceso 1.7 para que lo devuelva al sub-proceso “Estructurar Info”, que a su vez ha recibido la estructura en base a la cual se organiza la información. “Estructurar Info” organiza los datos recibidos del proceso 1.7 y los ordena de acuerdo a la estructura indicada, generando así el reporte que se entrega para su visualización. Todo este proceso se muestra en la Figura 2-7.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.5: Generar reportes.

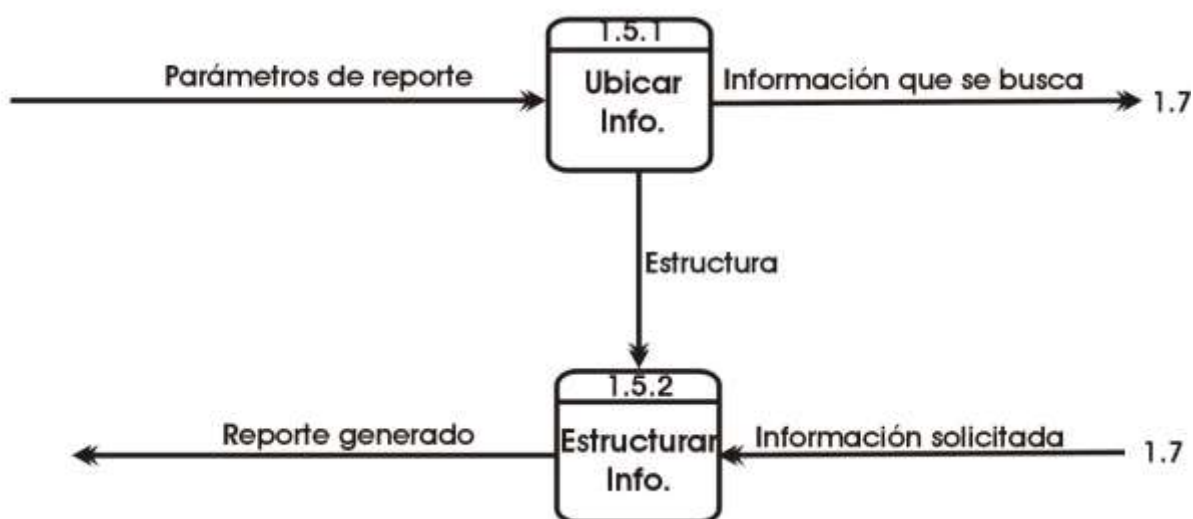


Figura 2-7. Diagrama de Flujo de Datos, nivel 2, proceso 1.5: Generar reportes.

Proceso 1.6: Visualizar mapa.

Este proceso es el encargado de generar la información base para la visualización del mapa. Para ello, se le proporcionan como entrada los parámetros del mapa solicitados por el usuario: nueva ubicación/coordenadas, nivel de zoom, desplazamiento, etc. Estos datos son recibidos por el sub-proceso “Identificar tarea de mapa”, que identifica cual es la tarea que el usuario solicita realizar con el mapa. Una vez definida, se pasan los parámetros de coordenadas al sub-proceso “Verificar mapa actual”, el cual accede la Base de Datos para conocer el estado actual del mapa, es decir su ubicación actual, nivel de zoom, etc. Tomando estos valores, se realizan los cálculos necesarios para

visualizar el mapa según las necesidades que el usuario expresa en sus parámetros de coordenadas. Ver Figura 2-8.

Se pide a su vez, información sobre los proyectos ubicados en el mapa generado. Ello se hace entregando la solicitud de información al proceso 1.7: “Generar proyecto”. Una vez obtenida, y junto con la información generada en “Modificar parámetros”, se ejecuta el sub-proceso “Reubicar mapa”, que integra la información y genera un mapa para su visualización.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.6: Visualizar mapa.

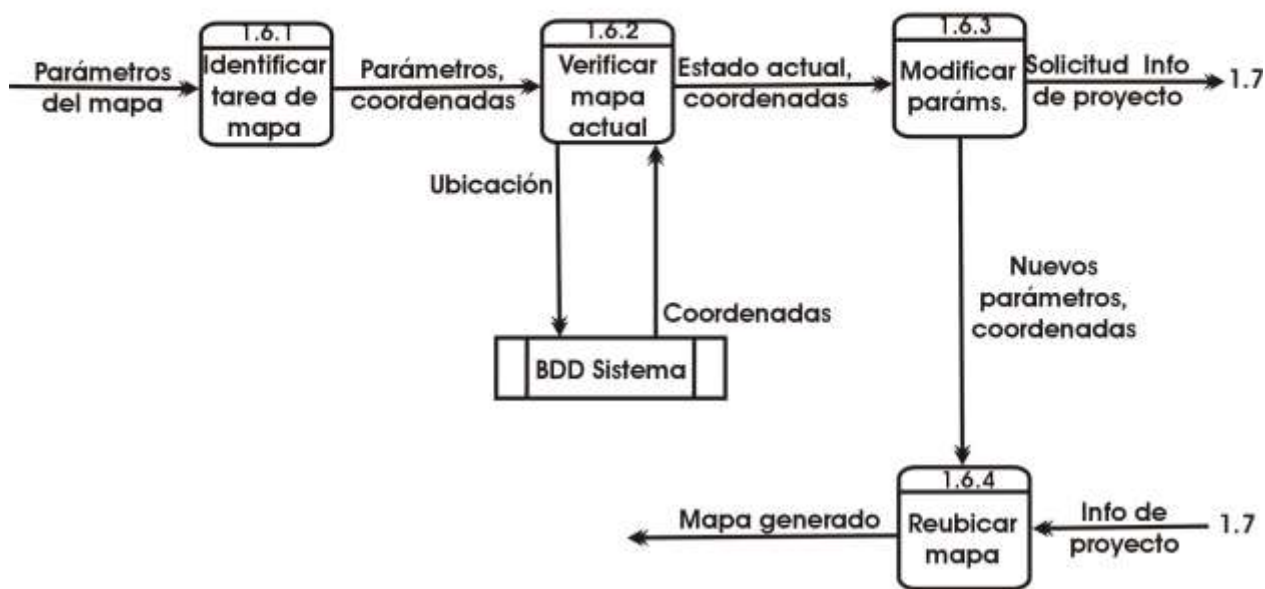


Figura 2-8. Diagrama de Flujo de Datos, nivel 2, proceso 1.6: Visualizar mapa.

Proceso 1.7: Servir proyectos.

A este proceso llegan las solicitudes de información sobre los proyectos. Por ello inicia recibiendo aquello que se busca (nombre de proyecto, estado de ejecución, ubicación/dirección, fuentes de financiamiento, etc.). El sub-proceso “Buscar Info” se encarga de esa tarea y busca la información en la Base de Datos, según se observa en la Figura 2-9. Los resultados, son informados al sub-proceso “Verificar estado info”, el cual identifica si la información es hallada o no. Al ser encontrada, los datos se

piden a la Base de Datos, y una vez obtenidos se entregan al sub-proceso “Estructurar/enrutar info”, en donde se ordenan los datos y se retornan al solicitante como reporte o información de consulta sobre proyectos.

Diagrama de Flujo de Datos. Nivel 2

Proceso 1.7: Servir proyectos.

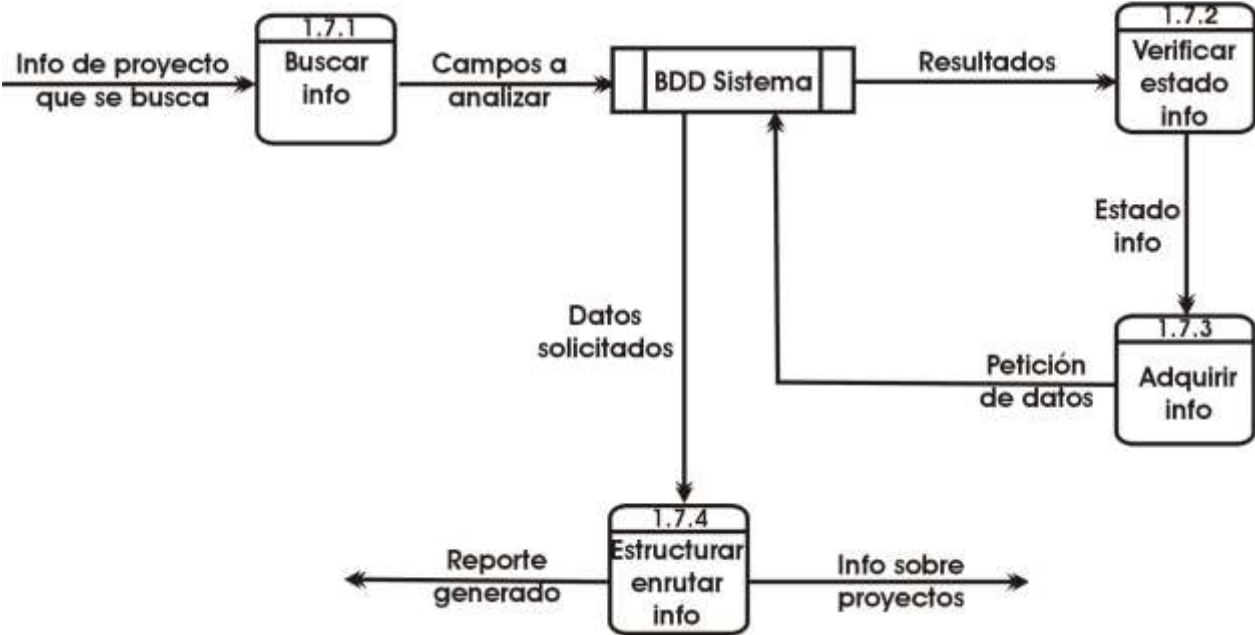


Figura 2-9. Diagrama de Flujo de Datos, nivel 2, proceso 1.7: Servir proyectos.

2.4 INFORMACIÓN TÉCNICA Y OPERATIVA.

2.4.1 Información Técnica.

La Secretaría Técnica de la Presidencia, trabaja su sistema de seguimiento municipalizado, en un servidor Web con las siguientes características:

- Sistema Operativo Windows 2000.
- Servidor Web Internet Information Server.
- Base de datos SQL 2000 Server.
- Programación en Java Servlets.
- ServletExec.

El proyecto ha considerado si los recursos técnicos actuales son suficientes o deben complementarse. En la Tabla 2-1 se detallan los requerimientos mínimos para la implementación de la aplicación tanto en el servidor como en las PC.

Servidor	PC's
Windows 2000 Server	Windows 9x/Me/XP (de preferencia)
ServletExec	Internet Explorer (de preferencia)
Jturbo Single Machine	SVG Viewer plug-in
JTurbo Single Server	Acceso a Internet
SQL Server 2000	
Jasc Web Draw	

Tabla 2-1. Requerimientos mínimos para implementar la aplicación.

Toda la tecnología requerida para la puesta en marcha del Sistema existe y opera actualmente en el lugar propuesto para el uso de la aplicación, la única inversión adicional que debería considerarse es la adquisición de la herramienta Jasc Web Draw. Aunque la falta de esta no afecta el funcionamiento óptimo de la aplicación, debe contarse con esta herramienta (u otra de diseño vectorial en SVG) para la modificación del mapa si en algún momento es requerido.

Dado que el software y equipo requerido para el uso del sistema existe en la Institución, se entiende que es factible técnicamente su desarrollo.

2.4.2 Información Operativa.

Para garantizar el funcionamiento de la aplicación posterior a su implementación, se requiere de recurso humano que participe en la operación del proyecto. Para ello, dicho recurso humano debe contar con el perfil descrito en la Tabla 2-2.

Usuarios del Sistema	Administrador del Sistema y base de datos
Conocimiento y manejo moderado de Internet Explorer.	Manejo avanzado de programación para ambientes Web.
Conocimiento básico de navegación e Internet	Manejo de herramientas para la administración en SQL Server 2000
	Conocimiento básico de aplicaciones de diseño vectorial (Web Draw por ejemplo)

Tabla 2-2. Perfil de usuarios de la aplicación.

La independencia funcional entre interfaces, procesos y datos permite la clasificación de los usuarios del sistema de acuerdo con sus necesidades y conocimientos del mismo. Hay tres grupos claramente identificables:

Técnicos Operativos-Administrativos, son los encargados de la captura cartográfica, el modelado y análisis. Es el usuario más sofisticado y con un profundo conocimiento del tema y el sistema, como mínimo deben cumplir con el perfil de Administrador.

Usuarios, personas que requieren del sistema para consultas, ingreso de obras e información referente a los proyectos. Su interés principal está en la información, su calidad, presentación y accesibilidad.

Visualizadores, esas personas son usuarios de productos finales tales como reportes, mapas o informes que usan para la toma de decisiones.

Debido a que la rama de diseño gráfico es un área muy extensa y minuciosa, se recomienda la contratación de un profesional en esta área para el levantamiento y modificaciones del mapa.

2.5 VIABILIDAD DE INTEGRACIÓN DE INFORMACIÓN DE OTRAS INSTITUCIONES CON LA APLICACIÓN A DESARROLLAR.

La gestión gubernamental engloba la labor de diversas instituciones y organismos que deben velar por el bienestar de la población. Cada institución experimenta necesidades que deben solventarse para poder cumplir óptimamente con las funciones que les corresponden.

Todos estos organismos manejan grandes cantidades de información, cada institución con necesidades de procesamiento de datos específicas y diferentes para el desempeño de sus tareas.

Existe hoy en día la necesidad de compartir información entre instituciones, así, datos de un ente complementan las necesidades de información de otro, y viceversa. Sin embargo, no hay normas gubernamentales establecidas para estandarizar la información que las instituciones comunican entre sí. Establecer esas normas constituye un profundo proceso de investigación de todas las organizaciones involucradas, así como de los formatos, tecnologías y herramientas utilizadas.

Debido a esta necesidad de compartir información se están realizando ya esfuerzos orientados a la integración, sin embargo existen áreas novedosas que recién se agregan a los sistemas de información de las instituciones de gobierno, como los Sistemas de Información Geográficos. Para estos sistemas no hay estándares definidos para su desarrollo de una manera uniforme, y cada institución los trabaja según sus propias necesidades y recursos. Entonces se considera que el objetivo de este proyecto no involucra el diseño de un esquema estandarizado de Sistemas de Información Geográfica a nivel gubernamental, sino más bien la creación de una propuesta de sistema para solventar necesidades de información geográfica de una institución en particular: Secretaría Técnica de la Presidencia.

Cabe mencionar que, dado que gran cantidad de organizaciones implementan sistemas heterogéneos y necesitan unificar su información, existe a nivel mundial un

enfoque orientado a la integración de información y sistemas. Hay por ello, variedad de soluciones de integración, la mayoría de ellas orientadas al *e-integration*, haciendo uso del Internet como medio estándar de tecnologías. Como referencia, pueden consultarse ejemplos de soluciones IBM diseñadas para instituciones específicas, listadas en

<http://www-3.ibm.com/software/success/cssdb.nsf/customerVW?OpenView&Count=15&RestrictToCategory=software> .

3.1 GENERALIDADES DE SIG.

Con la adopción de las nuevas tecnologías de información geográfica, la sociedad contemporánea está dando un gran paso hacia una gestión más eficaz e inteligente de su espacio.

Las nuevas tecnologías comprenden los Sistemas de Información Geográfica (SIG), la Percepción Remota, los Sistemas de Posicionamiento Global (GPS). En conjunto, permiten conocer mejor los fenómenos geográficos o geo-espaciales a todas las escalas, incluyendo los fenómenos naturales (recursos, ecosistemas), las actividades humanas (producción, habitación, movilidad) y los espacios construidos (infraestructuras, asentamientos).

La definición de un SIG dada por el *National Center for Geographic Information and Analysis* es "un sistema de información organizado compuesto por hardware, software, datos geográficos y personal, diseñado con procedimientos para capturar, almacenar, actualizar, manipular, analizar, modelar y desplegar eficientemente todas las formas de datos geo-referenciados, con el objetivo de resolver problemas de planificación, administración y gestión".

El componente más importante de un SIG son los datos. Lograr buenos datos de base frecuentemente absorberá el 60-80% del presupuesto de implementación de un SIG. Asimismo, recolectar buenos datos de base es un proceso largo, que frecuentemente demora el desarrollo de productos que pueden utilizarse para justificar la inversión. Un compromiso a un alto nivel es indispensable para llevar la implementación de un SIG a través de esta fase.

Los datos geográficos y los datos tabulares relacionados pueden obtenerse por recursos propios o adquirirse de un proveedor comercial de datos, para este caso, el mapa será creado a partir de una herramienta de diseño que soporte formato SVG (término que será explicado con detalle en capítulos posteriores) con información de coordenadas en un sistema propio de la herramienta utilizada.

Las ubicaciones de características cartográficas están referenciadas a las ubicaciones de los objetos que representan en el mundo real. Las posiciones de los objetos en la superficie esférica de la tierra se miden en grados de latitud y longitud, también conocidas como coordenadas geográficas. En un mapa plano, las ubicaciones de características cartográficas se miden en un sistema de coordenadas planares bi-dimensional. Las coordenadas planares describen la distancia al origen (0,0) a lo largo de dos ejes separado, un eje horizontal x, y un eje vertical y.

Este proceso de aplanamiento de la tierra crea distorsiones en distancia, área, forma, y dirección. El resultado es que todos los mapas planos están distorsionados en algún grado en estas propiedades espaciales. El impacto de la distorsión causada por una proyección cartográfica sobre los datos espaciales depende de cómo se usará los datos y el tamaño del área que está siendo mapeada, si se trabaja en un área relativamente pequeña tal como una ciudad o un estado; esta distorsión es generalmente insignificante (*según el ESRI en una de sus páginas www.gis.com*), se hará uso de este criterio para este caso en particular que se trabajará con el municipio de San Salvador que no es más grande que un estado dentro de los Estados Unidos, aun cuando este se aplicase para todo el país.

Un SIG permite asociar, en forma interactiva, las bases de datos alfanuméricas de la organización con las correspondientes bases cartográficas digitales. Gracias a estas capacidades, los SIG se están aplicando crecientemente en los más diversos sectores de la actividad pública y privada.

Así mismo un SIG puede mapear cualquier información almacenada en planillas o bases de datos, que tenga un componente geográfico que permita ver patrones, relaciones o tendencias, que no pueden verse en un formato de tabla o lista. Da una perspectiva totalmente nueva y dinámica de la información, y ayuda a tomar mejores decisiones, según el sitio de Geo-tecnologías de Costa Rica .

El proceso de hacer mapas con SIG es mucho más flexible que tradicionales aproximaciones cartográficas, manuales o automáticas. Comienza con la creación de

bases de datos. Pueden digitalizarse mapas existentes en papel y traducirse información computadora-compatible al SIG. La base de datos del mapa basada en el SIG puede ser continua y libre de escala. Los productos cartográficos pueden crearse centrados en cualquier localidad, a cualquier escala y mostrando información seleccionada simbolizada efectivamente para destacar características específicas.

Un nuevo paso en la utilidad de las tecnologías geo-espaciales se produjo gracias a las telecomunicaciones. Ahora se multiplican, mediante el uso de Internet, las posibilidades de intercambiar, compartir, distribuir y acceder a la información geográfica. Esto significará una gran expansión de los usuarios y nuevas oportunidades para los productores de dicha información.

Los Sistemas de Información Geográfica (SIG) están cambiando el terreno de los negocios. Antaño herramienta de geógrafos y cartógrafos, los SIG se han desplazado de los centros de investigación a los círculos corporativos, de la computadora del científico a la del hombre de negocios, de quien produce mapas al gerente.

Sin importar que el tema de estudio sea determinar donde ubicar obras, reorganizar territorios de venta, crear rutas de distribución, identificar nuevos mercados, o publicar mapas en Internet, los usuarios han aprendido la utilidad que representan los SIG para su negocio.

Aplicaciones de los SIG.

Aplicaciones bióticas

Agricultura y usos del suelo

Gestión de los recursos naturales

Forestación

Aplicaciones socioeconómicas

Censos y estadísticas de población

Censos económicos

Análisis de mercados

Logística y distribución

Aplicaciones de administración y gestión

Catastro

Planeamiento urbano y territorial

Defensa y seguridad

Infraestructuras y servicios públicos

3.2 IMPLEMENTACIÓN DEL CONCEPTO DE SIG.

El sistema de información desarrollado es orientado a la administración y gestión del sector público, en infraestructura y servicios de este sector, en el que se pretende mantener un banco de datos identificados y ubicados en un mapa, por departamentos, por municipios y por cantones en la que los datos se encuentran relacionados a las distintas representaciones gráficas de los mismos. Lo que permite visualizar en un lugar un proyecto determinado ya existente en una determinada zona y ayudar en la planificación en cuanto a la localización o ubicación de nuevos centros, confirmar la prioridad del mismo, darle seguimiento a las necesidades de la población, etc.

Para poder ubicar las obras dentro del mapa, el sistema proporciona la interfaz para la captura de dichas obras, donde los usuarios de las diferentes instituciones pueden colocar las figuras de las obras que les corresponden como entidad, en la zona del mapa que ellos deseen (dentro del municipio de San Salvador, para este caso que es un prototipo de Sistema); ya sea para la introducción de una obra existente o una que pueda empezar con su proceso de ejecución, la cual posteriormente será visualizada en el mismo lugar donde fue insertada, así mismo el sistema cuenta con mecanismos que le permiten al usuario ingresar la información respectiva referente a cada una de ellas en el orden adecuado para la funcionalidad de la aplicación y un mejor control de cada obra.

Los datos que allí se recogen permitirán combinar diferentes tipos de información, por ejemplo, centros escolares, centros de salud, centros deportivos y otras.

El sistema cuenta con su propio método de navegación dentro del mapa y con diferentes escalas de visualización dentro del mapa.

Al introducir nuevos datos, permite contar con información precisa para analizar diferentes escenarios y tomar decisiones acertadas, y así poder brindar apoyo a todas las instituciones que requieren información al momento de la toma de decisiones, sin embargo, un SIG no es un sistema automático de toma de decisiones, pero es una herramienta para consultar, analizar y mapear datos como soporte del proceso de

toma de decisiones, permitiendo a los que toman las decisiones enfocarse en los temas reales más que en tratar de entender los datos.

Debido a que el sistema puede ser consultado y accedido por funcionarios de diferentes instituciones, se han establecido mecanismos de seguridad, para garantizar la confiabilidad y seguridad de la información, donde cada usuario pueda acceder únicamente a datos permitidos.

3.3 GENERALIDADES SOBRE TECNOLOGÍAS.

Para llevar a cabo la generación del mapa con todos sus atributos, se utilizó la tecnología SVG, la cual ofrece ventajas sobre las herramientas ArcView y Grass, entre otras herramientas para la creación de SIG. ArcView por ser software propietario, la obtención de una licencia se volvía costosa, Grass por el contrario, perteneciente a software Open Source, se podía obtener sin costo alguno, pero por la complejidad de módulos para su instalación y la poca información, volvió difícil su operatividad.

Para llevar a cabo el desarrollo de la aplicación se ha hecho uso de las siguientes herramientas, que no tienen costo económico para su adquisición.

3.3.1 Lenguaje de Programación JavaScript.

La característica principal de JavaScript, es la de ser un *lenguaje de scripting* (script en inglés significa "guión". De hecho, el uso es exactamente éste: el navegador lee una línea, la interpreta y la ejecuta, después pasa a la sucesiva y hace lo mismo, y así hasta el cierre del script), pero, sobre todo, la de ser el lenguaje de scripting por excelencia y, sin lugar a dudas, el más usado. El código Javascript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido; un script ejecutado en el servidor, sin embargo, sometería a éste a dura prueba y los servidores de capacidades más limitadas podrían resentir de una continua solicitud por un mayor número de usuarios.

3.3.2 Lenguaje y Formato SVG.

La inclusión de imágenes en las páginas web supuso un gran avance en el posterior desarrollo y popularización de la WWW. Con la posibilidad de incluir archivos gráficos, las páginas web pasaron a ser de simples documentos textuales, normalmente destinados a divulgaciones científicas, a verdaderos documentos multimedia capaces de presentar al usuario final un fascinante mundo de color e imagen.

Dos han sido los formatos de archivos gráfico estandarizados en la web, el formato GIF y el JPG. Cada uno presenta sus ventajas e inconvenientes, y han servido durante mucho tiempo (en la escala relativa de Internet y la WWW) a los diseñadores web para construir páginas agradables y operativas.

A pesar de la utilidad demostrada de estos formatos gráficos, con el paso del tiempo empezaron a quedar limitados por su propia naturaleza simple, especialmente por la falta de unas propiedades comunes que permitieran la creación de gráficos compactos y funcionales.

Varios han sido los intentos de encontrar y estandarizar un nuevo formato que supliera estas limitaciones. Cabe destacar el formato BMP, de Microsoft, que aunque presenta una buena calidad en un relativo poco peso, no se ha convertido en estándar, quedando limitado su uso correcto en las diferentes versiones del navegador propietario de esta empresa, Internet Explorer. También hay que mencionar el formato PNG, que intenta conjugar las mejores características de los gráficos GIF y los JPG en un único formato, pero que tampoco ha llegado a alcanzar la popularidad que se esperaba.

Dos son las limitaciones básicas de todos estos formatos, desde el punto de vista del diseñador-programador Web actual:

- Son formatos de mapa de bits, por lo que no pueden ser escalados sin una pérdida de calidad considerable.
- Crean gráficos estáticos, basados en el almacenamiento interno de una tabla de valores de coloración para los diferentes píxeles que forman la imagen final, valores que son fijos e inmutables una vez configurado el archivo.

Con este planteamiento, las investigaciones de empresas y programadores han ido destinadas a la creación de un formato gráfico que reuniera una serie de características básicas:

- Escalabilidad, con lo que la calidad de la imagen final no estaría condicionada por el grado de aumento o reducción que se le diera. Es decir, que guardara gráficos vectoriales, no de mapa de bits.
- Dinamicidad física, con lo que se podrían crear pautas de comportamiento de los gráficos en función del tiempo, permitiendo con ello la construcción de verdaderas animaciones gráficas.
- Dinamicidad lógica, con lo que los gráficos dejarían de ser una entidad con propiedades fijas e inmutables una vez guardado el archivo. Con ello se consigue crear gráficos inteligentes, capaces de adaptarse a cambios en los valores que los definen.

En este sentido empezaron a trabajar las empresas interesadas en el tema, surgiendo diferentes propuestas y alternativas, algunas de las cuales aterrizaron y otras no.

Uno de los formatos gráficos surgidos de esta ola de investigación y que ha llegado a convertirse en un estándar en la web, es el formato Shockware Flash, de la casa Macromedia. Es un formato capaz de crear gráficos vectoriales escalables de alta calidad, con animaciones configurables en el tiempo a las que se pueden aplicar múltiples efectos y filtros. Además, los archivos finales resultan tener un relativo poco peso, en comparación con la calidad obtenida.

Como inconveniente, el formato Flash ha sido creado con un planteamiento de creación general, capaz de crear presentaciones en formato propio, en CD-ROM y, adicionalmente, en cualquier aplicación que admita la ejecución del plugin adecuado. No es pues un formato pensado y creado por y para la Web.

Para llenar este vacío gráfico, el W3C ha desarrollado un nuevo formato de gráficos vectoriales para la Web, el formato SVG (Scalable Vector Graphic).

El trabajo del W3C, a la hora de definir el formato SVG, se ha basado en la total integración del mismo con uno de los lenguajes más potenciados en la actualidad, el XML (eXtensible Markup Language), además de buscar la compatibilidad con otros estándares actuales y futuros, como XSLT, CSS, HTML 4.0, XHTML 1.0 y con el modelo DOM (Document Object model).

Sin entrar en excesivos detalles, se puede decir que XML es un lenguaje de marcas para la creación de documentos, híbrido entre el HTML (al que complementa y amplía) y el lenguaje general SGML. Es un lenguaje extensible, que permite la correcta definición de los datos que forman un documento y la unicidad formal de los mismos, con lo que diferentes aplicaciones pueden compartir los mismos datos sin problemas. También permite la total separación de contenido del contenido de un documento de la presentación del mismo y el acceso a bases de datos.

Con esta base, el Consorcio ha creado el formato SVG con código XML, con lo que este formato hereda muchas de las buenas cualidades del XML. En un futuro cuando el lenguaje XML se estandarice por completo, los gráficos SVG pueden ser una parte más de las páginas web, con su código definido e implementado en el cuerpo del documento html.

Se puede entonces definir el formato SVG como un puente entre diseño gráfico y programación, entre arte y tecnología; debido a su naturaleza puramente matemática, un gráfico SVG es una sucesión de objetos y puntos posicionados y orientados en el lienzo de trabajo, elementos que son definidos por fórmulas matemáticas y que pueden ser accedidos por código de programación para modificar sus propiedades.

Existen herramientas de aspecto tradicional y funcionalidades comunes que permiten generar un gráfico SVG de forma rápida y cómoda.

Las principales características gráficas del formato SVG son:

- Es un formato de gráficos vectoriales, con lo que los gráficos resultan editables, admiten curvas, transparencias, suavizados, rastrillados, etc.

- Produce gráficos sin posibilidades de distorsión, dado que pueden aumentar o disminuir de tamaño sin pérdida de calidad, lo que los hace adaptarse sin problemas a cualquier resolución de pantalla.
- Admite textos editables y seleccionables como elementos independientes de las figuras.
- Admite gestión avanzada del color, manejando 24 bits de profundidad, pudiendo además usarse en su definición cualquiera de los sistemas estándar: RGB, CMYK, etc.
- Se pueden incluir en un gráfico SVG sonidos y etiquetas explicativas.
- Permite la creación de animaciones en escala de tiempo.
- Admite diferentes tipos de filtros, consiguiendo resultados sorprendentes.
- Posiciona el gráfico de acuerdo con el puntero del ratón, con lo que cuando se quiere hacer zoom sobre el gráfico, basta hacer clic sobre la zona en la que se quiere situar y se centrará la ampliación en la misma.
- Es una tecnología de código libre, no propietaria, con las ventajas que eso representa para los desarrolladores.

A estas hay que sumar otra serie de propiedades nacidas de su vinculación al lenguaje XML, entre las que caben destacar:

- El código generador de un gráfico SVG no es binario, sino de simple texto plano, por lo que puede ser editado y modificado con cualquier editor de textos, incluido el Notepad de Windows.
- Como la definición del gráfico está implementada con código y datos tipo XML, se pueden definir los valores de los objetos constituyentes del gráfico en el

servidor Web y con datos provenientes de una base de datos, con la dinamicidad que ello supone.

- Soporta las Hojas de Estilos en Cascada (CSS), permitiendo con ello definir con toda exactitud el formato de presentación que van a tener los elementos del gráfico.
- También permite el acceso al DOM y soporta lenguajes de Script (JavaScript, etc.), con lo que es posible modificar las propiedades de los elementos de un gráfico en tiempo real.
- Es compatible con otro tipo de medios, como los dispositivos inalámbricos.
- Presenta apoyo para multilinguaje.

En resumen, el formato SVG es compatible con los actuales estándares web, proyectando además la compatibilidad de los gráficos con los lenguajes de programación, con lo que llegará un momento en que la integración de gráficos, datos y código de script será total.

Si a esto se suma que el tamaño/peso final de un fichero SVG no es muy alto, se puede deducir que este formato puede llegar a ser capaz de realizar muchas cosas impensables, combinando arte y técnica de una forma espectacular y personalizable.

Al no ser aún un estándar, para visualizar los gráficos SVG en los navegadores Web hace falta instalar un plugin especial en el PC, salvo que ya se tenga instalado un programa de creación SVG. Este plugin no muy pesado (aproximado 3 megas), exige trabajo un poco elevado del CPU de las máquinas. Sin embargo, cuando el lenguaje XML sea un estándar como tal, los gráficos SVG podrán ser integrados perfectamente en los navegadores Web, sin necesidad de software adicional.

Interoperabilidad del SVG.

Es una de las ventajas más importantes de SVG. Estos gráficos pueden importarse, verse, y pueden modificarse en diferentes ambientes.

SVG es un idioma que describe los gráficos bidimensionales en XML [XML10]. SVG permite tres tipos de objetos gráficos: formas gráficas vectoriales, imágenes y texto. Pueden agruparse los objetos gráficos, aplicarles estilo, transformarlos y componerlos en objetos previamente dados.

Los gráficos vectoriales contienen objetos geométricos como líneas y curvas. Esto da mayor flexibilidad comparada con formatos raster o capas (como PNG y JPEG) que tiene que guardar la información para cada píxel del gráfico. Típicamente, los formatos del vector también pueden integrar imágenes del raster y pueden combinarlos con la información del vector como las rutas para producir una ilustración completa; SVG no es ninguna excepción.

Manejo de SVG en páginas web.

Porque SVG es una gramática de XML, pueden generarse fácil y rápidamente los gráficos de SVG en los servidores "web, mientras" usando XML usando herramientas estándar, muchos de los cuales están escritos en el lenguaje de programación Java. Por ejemplo, un servidor web puede generar una calidad alta, bajo ancho de banda que abastecen al gráfico de los datos del mercado.

SVG permite gráficos que pueden ser creados en otros paquetes de software(vea el W3C la SVG aplicación lista) o automáticamente (por ejemplo, usando JAVA Servlets). Con SVG, se pueden manipular los gráficos que usan las herramientas de XML normales fácilmente.

Hay una variedad de formas en el que el contenido de SVG puede ser incluido dentro de una página web.

Una página web estática: En este caso, un documento de SVG (es del tipo image/svg+xml) está directamente cargado en un agente de usuario como un navegador web. El documento de SVG es la página web que se presenta al usuario.

Incluido por referencia: En este caso, una página web padre almacena las referencias separadas de un documento SVG y especifica que dicho documento de SVG debe incluirse como un componente de la página web del padre. Para HTML o XHTML, se presentan tres opciones:

HTML/XHTML el elemento de img es el método más común para usar los gráficos en las páginas de HTML. Para el despliegue más rápido, la anchura y altura de la imagen puede darse como los atributos. Un atributo que se requiere es el alt, daba una alternativa textual para las personas que hojean con las imágenes fuera de la red, o quién no puede ver las imágenes. La red no puede contener ningún encarecimiento.

HTML/XHTML el elemento object puede contener otros elementos anidados dentro de él, diferente img que está vacío. Esto significa que varios formatos diferentes pueden ser ofrecidos, usando anidamiento elementos object, con una alternativa textual final, (incluso el markup, links, etc).

HTML/XHTML elemento applet que puede invocar un applet de Java para ver SVG dentro de la página web dada. Estos applets pueden hacer muchas cosas, pero una tarea común es usarlos para desplegar las imágenes, particularmente algunos en formatos raros o que necesita ser presentado bajo el mando de un programa por alguna otra razón.

Incrustado en línea: En este caso, el volumen de SVG es directamente incluido dentro de página web padre. Un ejemplo es textualmente una página web XHTML con un fragmento de documento SVG incluido.

Un link externo, usando el HTML 'el elemento a': Esto le permite a cualquier visor de SVG ser usado que puede ser (pero no necesario) un programa diferente que desplegaba HTML. Esta opción se usa típicamente para formatos raros de imagen.

Puede generarse dinámicamente en un servidor web como respuesta a instrucciones de Java, JavaScript, Perl o XML.

3.4 HERRAMIENTAS DE DISEÑO GRÁFICO.

Al ser el código interno de un fichero SVG código en texto plano, lo único que hace falta para crear un gráfico en este formato es un editor de textos simple, valiéndose para ello cualquiera de los que existen en el mercado. Pero para poder trabajar en formato de texto es necesario tener conocimientos del lenguaje XML y de la propia sintaxis SVG, por lo que para crear un gráfico SVG de una forma más práctica y rápida resulta mucho más útil y sencillo el uso de alguna herramienta que nos simplifique el proceso.

A la hora de hablar de software SVG hay que distinguir entre aplicaciones que permiten exportar ficheros propios en formato SVG y aplicaciones nativas SVG, es decir, aquellas que directamente trabajan con la sintaxis SVG. Dentro del primer grupo destacan, por ejemplo, Adobe Illustrator en sus versiones 9 y 10 (que además permite también exportar ficheros vectoriales en formatos SWF y PDF), Adobe GoLive 5.0 y Corel Draw en sus versiones 9 y 10. El inconveniente que presentan estos programas es que el código que generan es a veces bastante problemático, por lo que se hace necesario una posterior “limpieza” que permita clarificar su interpretación. En cuanto a programas nativos SVG como WebDraw, para la creación de gráficos en general se dispone en el mercado de diferentes aplicaciones comerciales, pero bajo un costo comercial.

Para llevar a cabo la creación del mapa con sus diferentes atributos y características que se ajusten a las necesidades, se han utilizado las herramientas Adobe Illustrator y Web Draw.

3.4.1 Adobe Illustrator.

El software Adobe Illustrator 10 define el futuro de los gráficos vectoriales con innovadoras opciones de creación y las herramientas potentes para la publicación eficiente del arte final en la web, mediante impresión o como mejor se ajuste a las necesidades. Crea imágenes web soberbias mediante símbolos y opciones de sección

innovadoras. Explora ideas creativas con herramientas de distorsión en vivo. Publica en tiempo récord gráficos dinámicos gestionados por datos y otras prestaciones de productividad. Adobe Illustrator 10 ofrece todo lo que se necesita para crear imágenes de calidad profesional para la web, impresión, PDAs, dispositivos inalámbricos y para cualquier lugar.

Características Especiales:

- Gráficos gestionados por datos para generar variaciones de un mismo diseño con scripts de forma automática.
- Soporte de programación para automatizar tareas repetitivas.
- Soporte de gestión de activos incorporados.
- Soporte de metadatos.
- Paleta de acciones para aplicar series de comandos.
- Herramienta de formato de gran precisión, incluyendo la opción de atraer a la cuadrícula, las Guías inteligentes y la paleta Alinear.
- El Cuentagotas obtiene muestras de atributos de texto y color.
- Paleta muestras para almacenar motivos y colores personalizados.
- Fácil gestión de archivos enlazados.
- Compatibilidad con una amplia gama de formatos de archivo.
- Capas, máscaras, transparencias y formas compuestas compatibles con Adobe Photoshop.
- Edición directa de los archivos de Illustrator en Adobe InDesign.
- Optimización y edición compatible con Adobe GoLive.
- Capas y transparencia para utilizarse en Adobe LiveMotion.
- Capas y transparencia compatibles con Adobe After Effects.
- Integración con Adobe Premiere.

Adobe Illustrator 10 es una de las herramientas más potentes en materia de ilustración vectorial, capaz de integrar con todos los campos creativos actuales: impresión, web, animación... Illustrator es capaz de importar y exportar archivos SVG (animaciones

para la web) y dispone de herramientas de segmentación aptas tanto para imágenes raster o vectoriales. Además, es posible generar rollovers que integran completamente con las características de edición de Adobe Photoshop. Una aplicación ideal para todo tipo de trabajos creativos y de composición.

Aun que dicha herramienta servirá de apoyo para dar un mejor color y forma a cada elemento, se tendrá que realizar una limpieza de código para que funcione como lenguaje nativo de SVG haciendo uso de la herramienta Jasc WebDraw.

Requerimientos del sistema
Win: Windows 98, ME, 2000, XP · 128 Mb RAM
Mac: MacOS 9.1 o sup. MacOS X · 128 Mb RAM

Tabla 3-1. Requerimientos para la instalación de Adobe Illustrator 10

3.4.2 Jasc WebDraw.

Jasc WebDraw 1.0.1 es un programa nativo de SVG (Scalable Vector Graphics), brindando todas las herramientas necesarias para manipular este novedoso formato de gráficos para web. Un formato basado en archivos de texto, SVG permite la utilización de formas, imágenes, texto, eventos de animación e interactividad compatibles con tecnologías Web de uso común como HTML, XML, Javascript and CSS.

WebDraw combina las herramientas estándar de trazado y animación encontradas en los dibujos SVG típicos, utilizando el formato en estado nativo, tal cual es. Eso proporciona un elevado nivel de control sobre los gráficos SVG, garantizando obtener completa ventaja de las características del formato.

WebDraw permite:

- Crear gráficos SVG de alta calidad con herramientas estándar de diseño digital.
- Crear animaciones con la línea de tiempo basada en objetos.
- Editar el código SVG directamente, para máximo control y flexibilidad.
- Asegurar la precisión del código, mediante validación automática de SVG y reporte de errores de sintaxis.
- Agregar vistosidad gráfica con la adición de gradientes, filtros y patrones de relleno.
- Importar y editar archivos SVG creados con otras aplicaciones gráficas.

3.4.3 Resumen de herramientas a utilizar.

En la siguiente tabla se presentan las herramientas de desarrollo utilizadas en el proceso de creación de la aplicación:

Tecnología/herramienta	Versión	Tareas realizadas
SVG	1.1	Especificación sobre la que se rige el diseño y estructura del mapa. Utilización de las funciones y propiedades de sus elementos para manipular el mapa.
Lenguaje Javascript	-	Utilizado para diseñar funciones mediante las que se adquieran, analicen y modifiquen las propiedades y funciones SVG, a fin de permitir al usuario interactuar con el mapa.
Java servlets	2.3	Método de acceso a la Base de Datos, y de escritura de SVG y HTML interactivo-personalizado (según usuario).
Lenguaje HTML	-	Visualización del Sistema en el Navegador.

Jasc WebDraw	4	Diseño gráfico del mapa en formato SVG.
Adobe Illustrator	10	
SQL Server 2000	-	Creación y administración de base datos.

Tabla 3-2. Resumen de tecnologías a utilizar.

4.1 DISEÑO DEL MAPA.

4.1.1 Base para la digitalización del mapa.

1. Selección del tipo de datos de gráficos.

Para llevar a cabo el desarrollo de este sistema era posible hacer uso de datos gráficos de dos tipos:

- a. Datos vectoriales.
- b. Datos raster.

Los datos vectoriales se adquieren mediante el proceso de digitalización. Dicho proceso permite definir clara e individualmente cada elemento del mapa y almacenarlo en un archivo codificado, que será interpretado luego, por el correspondiente software de graficación. La información vectorial esta compuesta por tres clases de datos: puntos, líneas y polígonos.

Según las necesidades del mapa digital, para la representación visual de los diferentes elementos se usan, por ejemplo: puntos para las localidades; líneas para calles, etc.; polígonos para zonas, departamentos, municipios y cantones; o incluso una combinación de ellos.

La diferencia fundamental entre los diferentes tipos de graficación, radica en el modo en que se almacenan y codifican sus coordenadas.

El segundo tipo de información, (Raster), se adquiere mediante el proceso de rasterización de la imagen, esto es, decodificar cada uno de los píxeles que conforman la imagen y almacenar su posición dentro del mismo archivo. Cada píxel en la pantalla será coloreado de acuerdo a su código y cada código representará una característica diferente.

Por ello, se dedujo que convenía más, de acuerdo a los objetivos del sistema, el uso de imágenes vectoriales. Por ello se estableció que dicho método sería utilizado para la creación del mapa.

2. Digitalización.

El proceso de transformar la información gráfica en mapas digitales, se llevó a cabo haciendo uso del software Adobe Illustrator y Jasc Web Draw.

La técnica de digitalización, consiste en trazar los puntos, vértices, líneas o arcos del plano que formarán cada uno de los elementos del mapa. Esta información gráfica (propiedades, coordenadas de ubicación) se almacena en un sistema propio de la herramienta de diseño, basado usualmente en la generación de archivos.

3. Topología.

En esta etapa se logra que todas las entidades del mapa digital (áreas, líneas y puntos) cumplan ciertas condiciones necesarias de relación entre sí, lo que permite identificar a cada polígono, arco o punto como una entidad individual dentro del conjunto. Así se definen las características de los elementos del mapa: si es visible, y cuando no; su color de fondo, grosor del delineado, posición x, posición y, etc. Este paso posibilita, posteriormente, asignarle a cada entidad los atributos descriptivos que le definirán. A cada entidad del mapa digital puede asignársele la cantidad de atributos permitidos por la herramienta; los valores de estos atributos se almacenan en bases de datos alfanuméricas, quedando así relacionada información gráfica (puntos, líneas, polígonos) con información en la base de datos, lo que luego permite identificar un elemento en el espacio gráfico mediante la selección de uno o varios de sus atributos en la base de datos alfanumérica; o bien, indicando un elemento del mapa digital, encontrar inmediatamente sus atributos descriptivos en la base de datos alfanumérica.

4. Definir un sistema de coordenadas.

Se decide implementar un sistema de coordenadas X–Y que permita al usuario definir la posición del elemento (obra física) que se ubicará, y posteriormente hacer posible la visualización, ubicando todos los elementos en las posiciones definidas por el usuario.

Uno de los aspectos principales, es el de relacionar cada objeto representado, con su posición en el mapa.

5. Carga de datos alfanuméricos.

Se hizo necesario diseñar una estructura de base de datos adecuada para los valores del mapa. Los elementos gráficos cuentan con atributos, en los cuales, a su vez, se almacenan valores: posición x, altura, color, etc. Se debe hacer coincidir el tipo de datos de estos valores con su respectivo registro en la base de datos. En esta base de datos pueden consultarse todos los atributos descriptivos de la entidad correspondiente y puede realizarse con un administrador de base de datos comercial (MS SQL Server).

6. Base de datos con información del mapa.

Para poder relacionar información de los elementos del mapa con información del usuario, cada elemento del mapa está relacionado en una base de datos que lo define y almacena sus atributos.

Esta información es accedida para presentar al usuario información que le sea útil, o que haya solicitado. A la vez, puede modificarse esta información para actualizar sus valores. Se necesita así, además de la base de datos, un mecanismo de comunicación entre el Sistema y la base de datos. Se define entonces hacer uso de Java Servlets para estas tareas de enlace Sistema-Base de Datos.

7. Presentación.

Es decir, presentar adecuadamente la información a quienes lo solicitan. Aquí tiene especial relevancia el formato vectorial y la forma en que se mostrará a los usuarios. Existen otros formatos vectoriales, pero no todos son fácilmente compatibles con el entorno web. Se decide utilizar el formato SVG, debido a sus capacidades de formato vectorial y su interactividad web, gracias a su compatibilidad con HTML/JavaScript.

Ello permite crear agradables formas de presentar el mapa, adhiriendo métodos de interacción con quienes usen el sistema.

4.2 DISEÑO DE ENTRADA Y SALIDA DE DATOS.

4.2.1 Salida de Datos.

La salida es la información que se entrega a los usuarios por medio del sistema de información. Debido a que la salida útil es esencial para asegurar el uso y aceptación del sistema de información hay varios objetivos que se han considerado al momento de realizar el diseño.

1. Diseñar la salida para que sirva al propósito deseado.
2. Diseñar la salida para que se ajuste al usuario.
3. Entregar la cantidad adecuada de salida.

El principal propósito de el sistema de información desarrollado es resolver la manipulación de la información para su consolidación y presentación, siendo la mayoría de veces la presentación a través de ubicaciones específicas en el mapa de El Salvador.

Por tanto, se considera que la principal salida de este sistema es la de presentación de información en el mapa, y para ajustar esta salida a los requerimientos de cada usuario, se vuelve útil el uso de una pantalla con listas de selección que muestren diferentes opciones, para lo cual se ha diseñado una herramienta que sirva para la selección de los criterios a utilizar en la consulta de datos a presentarse de forma visual en el mapa.

Las listas de selección (Figura 4-1) definidas poseen información contenida en la base de datos, esto hace referencia a obras con necesidades y proyectos registrados, de tal forma que se realice una extracción de datos que permita la interacción con el usuario.

Criterios de Filtrado	
Institucion	Seleccione ▾
Tipo de Obra	Seleccione ▾
Etapas	Seleccione ▾
Departamento	Seleccione ▾
Municipios	Seleccione ▾
Año	Seleccione ▾
Mostrar	

Figura 4-1. Criterios de selección para consulta

El resultado visual del filtrado de datos se verá reflejado en el mapa Figura 4-2, donde se mostrarán las obras que cumplan con los criterios seleccionados así como información general referente a cada obra ubicada en el mapa, haciendo uso de herramientas visuales que posean características que permitan la interacción con el usuario. El acceso a la información de las obras debe ser únicamente en modo de consulta, desde los botones de la página.

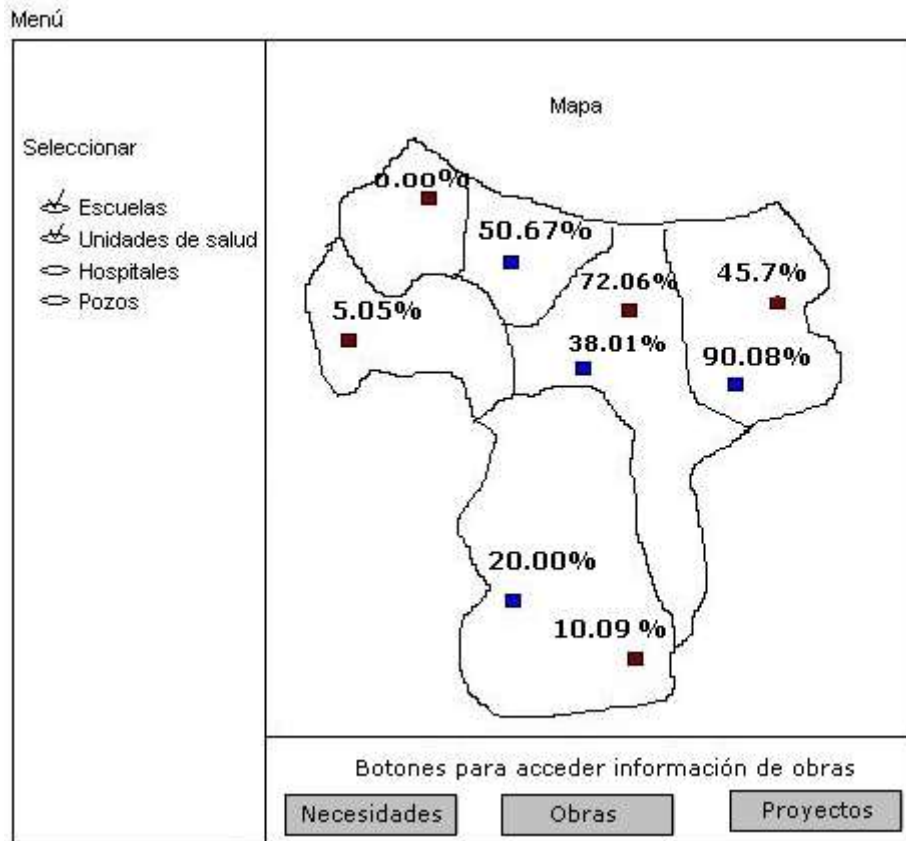


Figura 4-2. Pantalla resultado consulta mapa

Además de las salidas que son presentadas en el mapa, se han considerado salidas en formato de reportes.

No siempre más es mejor, especialmente cuando se refiere a la cantidad de salida, por tanto la cantidad de información debe ser suficiente para realizar análisis, pero clara y precisa, orientada a análisis gerenciales, en los cuales el usuario tenga opción a generar nuevos reportes con mayor precisión de detalles si este así lo deseara.

Se ha diseñado un reporte que incluye información consolidada bajo cinco criterios:

1. Inversión financiera mes a mes.
2. Inversión por tipo de obra.
3. Inversión por fuente de financiamiento.
4. Inversión por institución.

5. Etapas en las que se encuentra la inversión.

Dicha información debe ser presentada en un diseño agradable y sencillo para el usuario, donde no exista saturación y confusión en los datos.

4.2.2 Entrada de Datos.

La calidad de entrada de un sistema determina la calidad de la salida del sistema.

Para ello se han considerado cuatro lineamientos:

1. Que las formas sean fáciles de llenar.

Para facilitar la captura de datos al usuario, se considera el tamaño de las ventanas, un tamaño con el cual el usuario se sienta cómodo. Debe tener una secuencia lógica de la información.

2. Que satisfagan el objetivo para el cual fueron diseñados.

El objetivo es recopilar la información necesaria para obtener las salidas para el cumplir el propósito de la creación de este sistema de información, por tanto, el formulario debe solicitar únicamente dicha información.

3. Asegurar el llenado preciso

Para ello se deben generar mecanismos de interacción con el usuario, los cuales sirvan como guía al momento de realizar el llenado.

Se han definido procesos de validación para la reducción de errores al momento de la captura de datos:

a) Campos alfanuméricos: Si el campo es para capturar información tipo texto, el usuario puede ingresar cualquier carácter.

b) Campos numéricos: Si el campo es para capturar información tipo número, el usuario no puede ingresar ningún otro carácter que no este entre los dígitos del 0 al 9.

- c) Campos fecha: Si el campo es para capturar información tipo fecha, el sistema restringe tanto el formato de la fecha (dd/mm/aaaa), como el rango (que sea una fecha válida).
- d) Campos nulos: Si el campo no puede quedar vacío al momento de hacer la actualización, el sistema obliga al usuario a que introduzca la información.

4. Formas atractivas.

Los colores utilizados deben ser colores llamativos aunque no chillantes ni cansados a la vista del usuario. Para lograr esto se ha contado con el apoyo de un profesional en el diseño gráfico quien orienta a utilizar los colores y matices atractivos.

El diseño de captura de información ha sido basado en pantallas en las que el usuario digite los datos que se deben proporcionar al sistema. De esa forma las pantallas están diseñadas utilizando controles de uso común para el usuario, tales como:

Cuadros de texto: para la introducción de datos desde teclado.

Combos de selección: para seleccionar valores establecidos.

Botones: para realizar acciones.

Todos estos elementos forman parte de formularios, en los que se agrupan de acuerdo a la función que realiza, como almacenar una obra, un nuevo proyecto, información de consulta, etc.

Captura de obras.

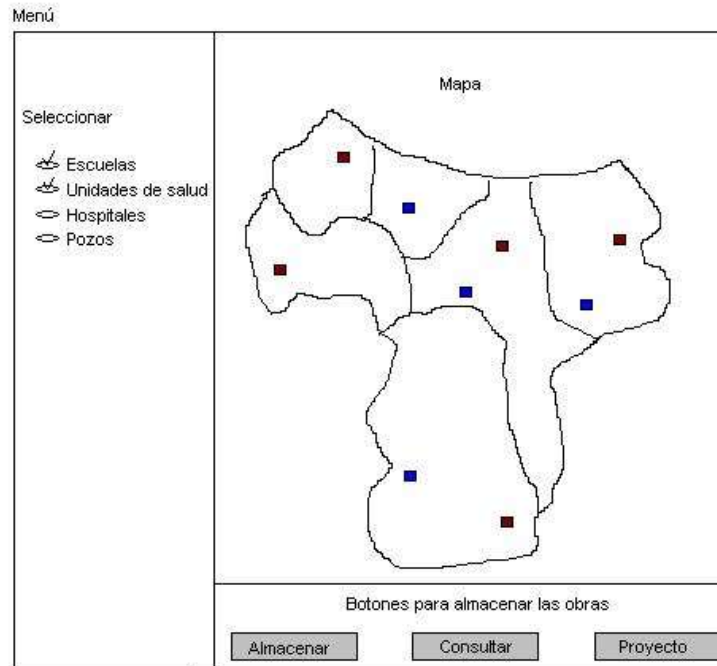


Figura 4-3. Pantalla de captura para registro y modificación de información por obra

Una obra es ingresada directamente desde la pantalla del mapa, para los datos principales que identificarán una obra:

Captura de Posiciones X y Y, cantón al cual será asignado, tipo de obra, tomadas a partir del mapa en pantalla (Figura 4-4).

Introduzca Nombre de Obra
Nombre de Obra
<input type="text"/>
<input type="button" value="Aceptar"/>
<input type="button" value="Cancelar"/>

Figura 4-4. Pantalla final de captura para el nombre de obra.

Constituye la parte de recepción de información a la cual están relacionadas algunas de las pantallas de captura, de modo que debe existir una obra para que Captura de Proyectos, Captura de Necesidades, Captura de Fuentes de Financiamiento tengan un sentido funcional.

Captura de Necesidad.

Corresponde a la información que identifica cada una de las necesidades a ser asignadas a cada una de las obras registradas en la aplicación. (Figura 4-5)

El diagrama muestra una interfaz de usuario para la captura de necesidades. A la izquierda, un recuadro contiene los siguientes campos de texto: 'Código', 'Descripción:', 'Monto:', 'Fecha de Ingreso:' y 'Responsable:'. A la derecha de este recuadro, hay una serie de líneas horizontales que representan los campos de entrada de datos correspondientes. En la parte inferior del formulario, hay un botón rectangular con el texto 'Agregar'.

Figura. 4-5 Pantalla necesidades por obra.

En esta pantalla de captura se realizan los tipos de validaciones correspondientes a cada tipo de dato que debe ser ingresado, como: números para el caso de monto, validación de fecha en campo fecha de ingreso, y el responsable que es obtenido a partir del usuario que ha ingresado al sistema, así como la validación de no permitir campos vacíos en aquellos que son requeridos.

Captura de Proyectos.

Constituye la información relacionada a los proyectos que pueden ser programados, dependen de la existencia de obras, y necesidades asignadas; dado que un proyecto se programa de acuerdo a una necesidad existente para cada obra registrada. (Figura 4-6)

Código

Nombre

Descripción:

Fecha de Inicio:

Fecha de Finalización:

Etapa:

Estado:

No. Beneficiarios:

Responsable:

Agregar

Figura 4-6 Pantalla proyectos por obra

Captura de Fuente de Financiamiento.

Se incluyen en esta sección la asignación de fuentes de financiamiento que se harán responsables por los fondos a invertir en cada uno de los proyectos a programarse. (Figura 4-7)

Código:

Obra:

Proyecto:

Fuente de Fin.:

Tipo de Fin.:

Monto:

Aceptar

Figura 4-7 Pantalla Fuentes de financiamiento

Avances del Proyecto. (Figura 4-8)

Mes	Progreso Financiero	Progreso Físico	Ejecución Fianciaera	Ejecución Física	Observaciones
Enero	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Febrero	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Marzo	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 4-8 Pantalla seguimiento de proyectos por obra

4.3 DISEÑO Y CREACIÓN DE BASE DE DATOS.

4.3.1 Modelo Entidad-Relación.

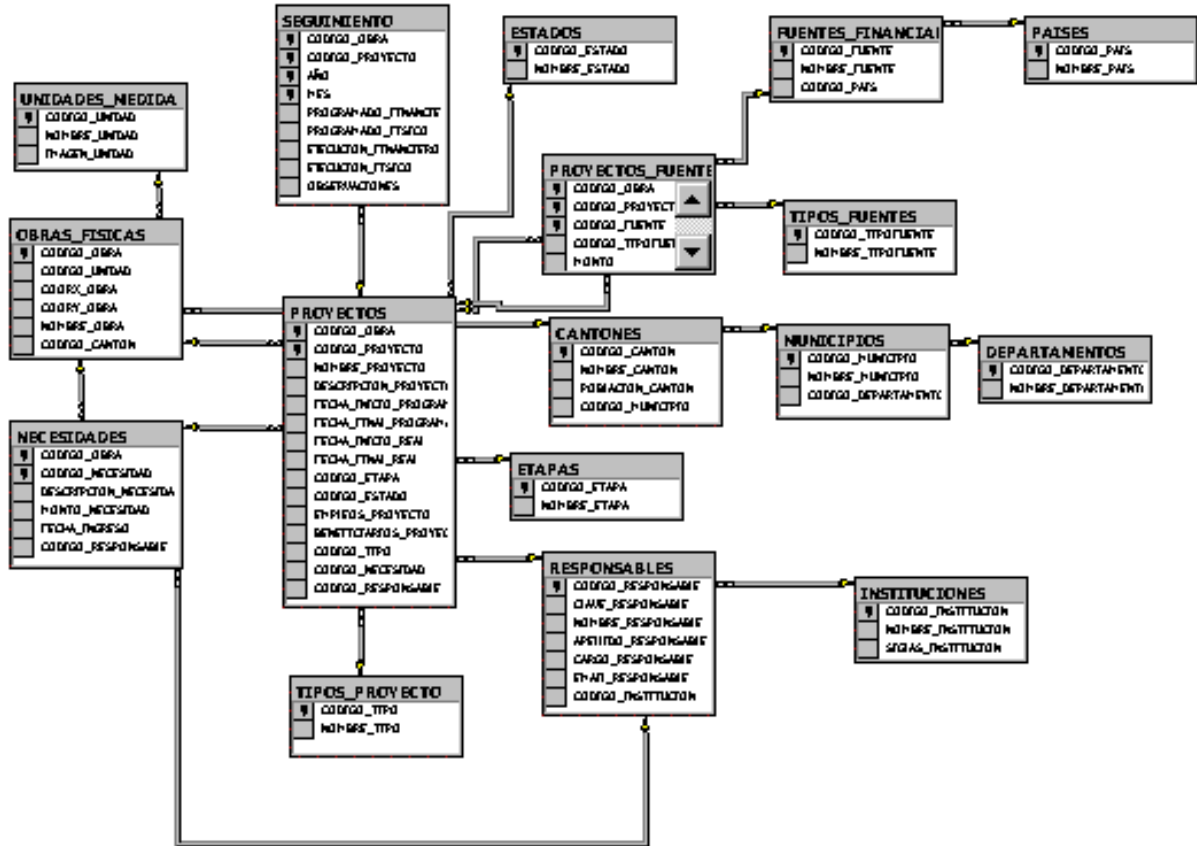


Figura 4-9. Modelo Entidad-Relación.

4.3.2 Definición, detalle de tablas y atributos

Tabla Cantones.

Almacena los cantones y sus atributos.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_CANTON	enteros (int)	Almacena el código de identificación de cada cantón.
NOMBRE_CANTON	alfanuméricos (varchar)	Almacena el nombre del cantón
POBLACION_CANTON	enteros (int)	Almacena el número de habitantes en cada cantón.
CODIGO_MUNICIPIO	enteros (int)	Almacena el código del municipio al cual pertenece el cantón.

Tabla 4-1. Cantones.

Tabla Departamentos.

Almacena los departamentos y sus atributos.

Campos:

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_DEPARTAMENTO	enteros (int)	Almacena el código de identificación de cada departamento.
NOMBRE_DEPARTAMENTO	alfanuméricos (varchar)	Almacena el nombre del departamento.

Tabla 4-2. Departamentos.

Tabla Estados

Almacena los estados en los que puede encontrarse un proyecto, estos pueden ser: Atrasado, A Tiempo, Adelantado, Suspendido, No corresponde Iniciar.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_ESTADO	enteros (int)	Almacena el código que identifica a cada estado.
NOMBRE_ESTADO	alfanuméricos (varchar)	Almacena el nombre del estado.

Tabla 4-3. Estados.

Tabla Etapas

Almacena las etapas por las que puede pasar un proyecto. Por ejemplo, No Iniciado, En Ejecución Física, Terminado Físicamente, etc.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_ETAPA	enteros (int)	Almacena el código que identifica a cada etapa.
NOMBRE_ETAPA	alfanuméricos (varchar)	Almacena el nombre de cada etapa.

Tabla 4-4. Etapas.

Tabla fuentes-financiamiento

Almacena las fuentes de financiamiento que pueden estar participando en un proyecto.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_FUENTE	alfanuméricos (varchar)	Este campo es la llave primaria de la tabla, almacena el código que identifica a cada fuente de financiamiento.

NOMBRE_FUENTE	alfanuméricos (varchar)	Almacena los nombres de las fuentes de financiamiento.
---------------	----------------------------	--

Tabla 4-5. Fuentes de financiamiento.

Tabla Instituciones

Almacena las características de las instituciones que estén involucradas con el Sistema.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_INSTITUCION	enteros (int)	Este campo es la llave primaria de la tabla, almacena el código que identifica a cada institución.
NOMBRE_INSTITUCION	alfanuméricos (varchar)	Almacena el nombre de la institución.

Tabla 4-6. Instituciones.

Tabla Municipios.

Almacena las características de cada municipio.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_MUNICIPIO	enteros (int)	Este campo es la llave primaria de la tabla, almacena el código que identifica a cada municipio.
NOMBRE_MUNICIPIO	Alfanuméricos (varchar)	Almacena el nombre que describe a cada municipio.

Tabla 4-7. Municipios.

Tabla Necesidades.

Almacena las necesidades identificadas para cada obra en particular.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_OBRA	enteros (int)	Este campo junto con el campo CODIGO_NEEDESIDAD forman la llave primaria de la tabla, almacena el código de la obra a la cual esta siendo asignada una necesidad identificada.
CODIGO_NEEDESIDAD	enteros (int)	Almacena el código con el que se identifica la necesidad registrada.
DESCRIPCION_NEEDESIDAD	Alfanuméricos (varchar)	Almacena la descripción de la necesidad para una comprensión de la situación en la obra a la cual esta siendo asignada. Tipo de datos permitidos.
MONTO_NEEDESIDAD	Numéricos, con dos decimales permitidos (decimal).	Almacena el monto estimado para cubrir la necesidad descrita.
FECHA_INGRESO	fecha (datetime)	Almacena la fecha de ingreso al sistema de la necesidad que se esta registrando.
CODIGO_RESPONSABLE	Alfanuméricos (varchar)	Almacena el código de identificación de cada usuario en el Sistema.

Tabla 4-8. Necesidades.

Tabla Obras_Fisicas

Almacena las características de las obras físicas ubicadas en el mapa.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_OBRA	enteros (int)	Este campo es la llave primaria de la tabla, almacena el código de identificación de cada obra
CODIGO_UNIDAD	alfanuméricos (varchar)	Este campo almacena el código del tipo de obra, por ejemplo, si es escuela, unidad de salud, etc.
COORX_OBRA	numéricos con X decimales (decimal).	Almacena el valor de la coordenada X en el mapa.
COORY_OBRA	numéricos con X decimales (decimal).	Almacena el valor de la coordenada Y en el mapa.
NOMBRE_OBRA	alfanuméricos (varchar)	Almacena el nombre de la obra.
CODIGO_CANTON	enteros (int)	Almacena el código del cantón. sobre el cual esta ubicado la obra.

Tabla 4-9. Obras.

Tabla Países.

Almacena los países a los cuales pueden pertenecer las Fuentes de financiamiento.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_PAIS	alfanuméricos (varchar).	Este campo es la llave primaria de la tabla, almacena el código que identifica a cada país.
NOMBRE_PAIS	alfanuméricos (varchar)	Almacena el nombre del país.

Tabla 4-10. Países.

Tabla Proyectos.

Almacena los proyectos que tiene asignada cada obra que se encuentra ubicada en el mapa.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_OBRA	enteros (int)	Este campo junto con el campo CODIGO_PROYECTO, representan la llave primaria de esta tabla. Almacena el código de la obra a la cual se le esta asignando el proyecto a registrar.
CODIGO_PROYECTO	enteros (int).	Almacena el código del proyecto que identifica al mismo.
NOMBRE_PROYECTO	alfanuméricos (varchar)	Almacena el nombre del proyecto.
DESCRIPCION_PROYECTO	alfanumérico (varchar)	Almacena la descripción del proyecto.
FECHA_INICIO_PROGRAMADA	fecha (datetime)	Almacena la fecha de inicio programada que se ha establecido para dar inicio al proyecto.
FECHA_FINAL_PROGRAMADA	fecha (datetime)	Almacena la fecha final programada que ha sido estimada para la finalización del proyecto.
FECHA_INICIO_REAL	fecha (datetime).	Almacena la fecha de inicio real del proyecto, una vez el proyecto ha sido iniciado.
FECHA_FINAL_REAL	fecha (datetime)	Almacena la fecha final real del proyecto, una vez el proyecto ha sido finalizado.
CODIGO_ETAPA	enteros (int)	Almacena el código de la etapa en la cual se encuentra el proyecto. Este campo mantiene relación con la tabla ETAPAS.
CODIGO_ESTADO	enteros (int)	Almacena el código del estado en el cual se encuentra el proyecto. Este campo esta relacionado con la tabla ESTADOS.
EMPLEOS_PROYECTO	enteros (int)	Almacena el número de empleos

		que están siendo generados por este proyecto.
BENEFICIARIOS_PROYECTO	enteros (int)	Almacena el número de personas que serán beneficiadas a través de este proyecto.
CODIGO_TIPO	enteros (int)	Almacena el código del tipo de proyecto. Este campo mantiene relación con la tabla TIPOS-PROYECTO.
CODIGO_NECESIDAD	enteros (int)	Almacena el código de la necesidad que esta siendo cubierta por este proyecto. Este campo mantiene relación con la tabla NECESIDADES.
CODIGO_RESPONSABLE	alfanuméricos (varchar)	Almacena el código del responsable de registro del proyecto en el Sistema. Este campo esta relacionado con la tabla RESPONSABLES.

Tabla 4-11. Proyectos.

Tabla Proyectos-Fuentes.

Un proyecto puede estar siendo financiado por muchas fuentes, y una fuente puede estar financiando muchos proyectos, con lo que nos encontramos con una relación de muchos a muchos. Esta tabla normaliza la relación entre las tablas PROYECTOS y FUENTES_FINANCIAMIENTO, almacena las fuentes de financiamiento que tienen participación en un proyecto.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_OBRA	enteros (int).	Este campo, junto con el campo CODIGO_PROYECTO y CODIGO-FUENTE representan la llave primaria de la tabla. Almacena el código de la

		obra a la que pertenece el proyecto.
CODIGO_PROYECTO	enteros (int).	Este campo almacena el código del proyecto al cual se le están especificando las diferentes fuentes que participan para llevar a cabo la ejecución.
CODIGO_FUENTE	alfanuméricos (varchar)	Almacena el código de la fuente que tiene participación en el proyecto especificado.
CODIGO_TIPOFUENTE	enteros (int)	Almacena el código que representa el tipo de financiamiento que se está dando.

Tabla 4-12. Fuentes de Financiamiento - Proyectos.

Tabla Responsables.

Almacena todas las características de los usuarios del Sistema.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_RESPONSABLE	alfanuméricos (varchar).	Almacena el código del responsable, este código será con el que ingrese al sistema.
CLAVE_RESPONSABLE	alfanuméricos (varchar)	Almacena la contraseña con la cual será admitido el usuario en el sistema.
NOMBRE_RESPONSABLE	alfanuméricos (varchar)	Almacena el nombre del responsable.
APELLIDO_RESPONSABLE	alfanuméricos (varchar)	Almacena el o los apellidos de la persona responsable.
CARGO_RESPONSABLE	alfanumérico (varchar)	Cargo que desempeña la persona en su institución.
EMAIL_RESPONSABLE	alfanuméricos (varchar)	Almacena la dirección de correo electrónico en la cual se puede contactar a este responsable.

Tabla 4-13. Responsables.

Tabla Seguimiento.

Almacena los países a los cuales pueden pertenecer las Fuentes de financiamiento.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_OBRA	enteros (int).	Este campo junto con los campos CODIGO_PROYECTO, AÑO y MES, constituyen la llave primaria de esta tabla. Almacena el código que identifica a la obra.
CODIGO_PROYECTO	enteros (int).	Almacena el código del proyecto al cual se le esta asignando registro de seguimiento. Tipo de datos permitidos: enteros (int). AÑO, almacena el año para el cual se esta registrando el seguimiento.
MES	enteros (int)	Almacena el mes para el cual se esta registrando el seguimiento.
PROGRAMADO_FINANCIERO	numéricos con dos decimales (decimal)	Almacena el monto programado.
EJECUCION_FINANCIERO	numéricos con dos decimales (decimal).	Almacena el monto de ejecución financiera.
PROGRAMADO_FISICO	numéricos con dos decimales (numeric)	Almacena la programación del porcentaje de avance físico.
EJECUCION_FISICO	numéricos con dos decimales (numeric).	Almacena el porcentaje de ejecución física.
OBSERVACIONES	alfanuméricos (varchar).	Almacena las observaciones referentes al seguimiento del proyecto en el período especificado.

Tabla 4-14. Seguimiento de Proyectos.

Tabla Tipos_fuentes.

Almacena los tipos de financiamiento que una fuente puede estar proporcionando, por ejemplo préstamo, donación, fondos propios, etc.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_TIPOFUENTE	enteros (int)	Almacena el código que identifica a cada tipo de financiamiento. Este campo representa la llave primaria de esta tabla.
NOMBRE_TIPOFUENTE	alfanuméricos (varchar)	Almacena el nombre que describe el tipo de financiamiento.

Tabla 4-15. Tipos de fuentes de financiamiento.

Tabla Tipos_Proyecto.

Almacena los tipos de proyecto que pueden estar siendo ejecutados, por ejemplo, si son ampliación, reconstrucción, construcción de nueva infraestructura, etc.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_TIPO	enteros (int).	Almacena el código que identifica el tipo de proyecto. Este campo representa la llave primaria de la tabla.
NOMBRE_TIPO	alfanuméricos (varchar)	Almacena el nombre que describe cada tipo de proyecto.

Tabla 4-16. Tipos de Proyectos.

Tabla Unidades_Medida.

Almacena los tipos de obras que pueden ser ubicados en el mapa, por ejemplo: escuelas, unidades de salud, vivienda, hospitales, etc.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_UNIDAD	alfanuméricos (varchar).	Almacena el código que identifica a cada unidad de medida. Este campo representa la llave primaria de esta tabla.
NOMBRE_UNIDAD	alfanuméricos (varchar)	Almacena el nombre que describe cada unidad de medida.

Tabla 4-17. Unidades de medida (Identifica el tipo de obra).

4.4 DISEÑO INTERFAZ DE USUARIO.

Debido a que el sistema se desarrolla en ambiente web, su acceso es a través de un navegador o browser, por lo que el usuario cuenta con todas las ventajas que este ofrece junto con el lenguaje HTML, tales como barra de navegación (atrás, adelante), hipervínculos (activos y visitados).

Para la interacción sistema-usuario, se ha diseñado un esquema de menú personalizado para cada usuario según su nivel de acceso permitido. Dicho esquema consiste en un proceso de autenticación del usuario el cual consulta la base de datos para verificar:

- a) Existencia del usuario
- b) Contraseña
- c) Nivel de acceso autorizado
- d) Menús disponibles para su nivel

Para desarrollar este esquema se han definido las siguientes tablas:

Tabla Menu.

De esta tabla se obtienen las características de las opciones del menú.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_MENU	alfanuméricos (char).	Almacena el código que identifica a cada menú. Este campo representa la llave primaria de esta tabla.
NOMBRE_MENU	alfanuméricos (varchar)	Almacena el nombre que describe cada elemento del menú.
URL_MENU	Alfanuméricos (varchar)	Almacena las especificaciones del archivo a ejecutar en caso de activar el

		menú.
SUPERIOR_MENU	Alfanumérico (char)	Almacena el código del menú padre del cual depende.
ORDEN_MENU	Entero (int)	Almacena el número que rige el orden de la opción de menú en el listado.

Tabla 4-18. Menú del Sistema.

Tabla PERFILES.

Almacena la descripción de cada uno de los perfiles definidos para los usuarios del sistema.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_PERFIL	alfanuméricos (varchar).	Almacena el código que identifica a cada perfil. Este campo representa la llave primaria de esta tabla.
NOMBRE_PERFIL	alfanuméricos (varchar)	Almacena el nombre que describe a cada perfil.

Tabla 4-19. Perfiles del Sistema.

Tabla PERFILES_MENU.

Para poder asignar los menús que corresponden a cada perfil, es necesario tener una tabla intermedia entre estas dos, ya que, un perfil puede tener asignados muchos menús, y un mismo menú puede estar asignado a diferentes perfiles.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_MENU	alfanuméricos (char).	Almacena el código que identifica a cada menú. Este campo junto con el campo CODIGO_PERFIL representa la llave primaria de esta tabla. Este campo esta relacionado con la tabla

		MENU.
CODIGO_PERFIL	alfanuméricos (char)	Almacena el código del perfil. Este campo esta relacionado con el campo CODIGO_PERFIL de la tabla PERFILES.

Tabla 4-20. Relación Perfiles - Menú.

Tabla PERFILES_RESPONSABLE.

Un perfil puede estar asignado a diferentes usuarios, y un usuario puede tener asignados más de un perfil, por tanto, se vuelve necesaria una tabla intermedia que normalice la relación muchos a muchos entre ambas tablas.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_PERFIL	alfanuméricos (char).	Almacena el código que identifica a cada perfil. Este campo junto con el campo CODIGO_RESPONSABLE representa la llave primaria de esta tabla. Este campo esta relacionado con la tabla PERFILES.
CODIGO_RESPONSABLE	alfanuméricos (char)	Almacena el código del responsable. Este campo esta relacionado con el campo CODIGO_RESPONSABLE de la tabla RESPONSABLES.

Tabla 4-21. Relación Perfiles - Responsables.

4.5 DISEÑO ADMINISTRACION DEL SISTEMA

Es necesario proveer al administrador del sistema de interfaces de entrada y salida especiales para la manipulación de información de tablas de la base de datos que contienen datos propios del sistema, tales como tablas de clasificación, listados, estados, etc.

Las tablas con datos propios del sistema suman una cantidad considerable, y tomando en cuenta que durante el ciclo de vida del sistema pueden darse cambios que pudieran incluir nuevos datos, nuevos campos o incluso nuevas tablas, se ha diseñado un mecanismo de manipulación de datos que incluye la inserción, actualización y eliminación, el cual para generar las interfaces únicamente necesita de ciertos parámetros que describan las características especiales de cada tabla, con esto se evita el desarrollo de una interfaz para cada una de ellas y la modificación en la programación cada vez que alguno de los campos sea alterado.

El mecanismo a desarrollar debe generar automáticamente las interfaces para la manipulación de datos, basándose en la configuración de las tablas creadas para este propósito. Las pantallas que se generen deben ser tres:

- 1) Listado de registros en la tabla.
- 2) Formulario de llenado de datos
- 3) Mensaje de actualización, inserción, eliminación o error.

El listado de registros debe poseer un enlace que genere el formulario del registro seleccionado, y cada campo del formulario debe adecuarse a las características de cada campo de la tabla a modificar, los campos pueden ser:

- a) Numéricos
- b) Carácter
- c) Texto
- d) Selección
- e) Fecha

Para cada uno de estos tipos de campos, el formulario generado debe considerar las características especiales de cada uno de ellos:

- a) *Numéricos*: Si el campo es numérico debe desplegarse un campo de texto con validación para aceptar únicamente números de la longitud permitida.
- b) *Carácter*: Si el campo es de tipo carácter debe desplegarse un campo de texto con validación para aceptar todos los caracteres permitidos, restringiendo la longitud permitida por el campo.
- c) *Texto*: Para este tipo de campo debe generarse un área de texto.
- d) *Selección*: Debe generarse un listado de selección con los datos permitidos para este campo, por tanto debe especificarse de que tabla se toman los datos.
- e) *Fecha*: Si el campo es de tipo fecha, se debe desplegar un campo de texto que permita 10 caracteres máximo y que el valor digitado sea una fecha valida en el formato dd/mm/aaaa.
- f) *Llave de la tabla*: Si el campo es la llave primaria de la tabla el campo no puede ser alterado.

Para llevar a cabo la creación de este mecanismo es necesario almacenar en la base de datos los datos de las tablas a manipular, para lo cual se han diseñado las siguientes tablas:

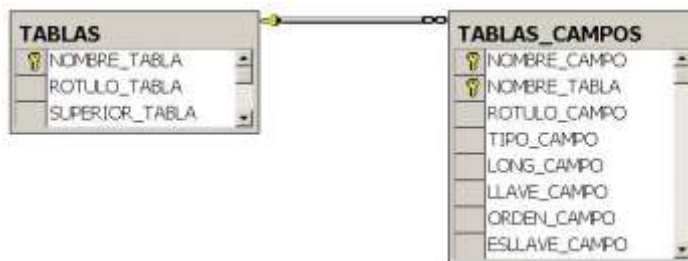


Figura 4-10. Tablas utilizadas para el mecanismo administración

Tabla TABLAS

Almacena el nombre de la tabla en la base de datos, así como el nombre descriptivo, y su relación con tablas padres.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_TABLA	alfanuméricos (varchar).	Este campo es la llave primaria de la tabla, almacena el código que identifica a cada tabla, que es el nombre propio de la tabla en la base de datos.
NOMBRE_TABLA	alfanuméricos (varchar)	Almacena el nombre descriptivo de la tabla.
SUPERIOR_TABLA	Alfanuméricos	Almacena el nombre de la tabla padre, si así fuera el caso.

Tabla 4-22. Descripción de Tablas.

Tabla TABLAS_CAMPOS

Almacena la descripción y características de todos los campos pertenecientes a una tabla en particular.

Nombre de Campo	Tipo de Dato	Descripción
CODIGO_CAMPO	alfanuméricos (varchar).	Este campo, junto con el campo CODIGO_TABLA forman la llave primaria de la tabla. Almacena el código que identifica al campo, es decir, el nombre del campo en la tabla.
CODIGO_TABLA	alfanuméricos (varchar)	Hace referencia a la tabla TABLAS, y almacena el nombre propio de la tabla a la que pertenece el campo.

ROTULO_CAMPO	Alfanuméricos	Almacena el nombre descriptivo (que se mostrará en pantalla) del campo.
TIPO_CAMPO	Alfanumérico	Define que tipo de datos acepta el campo (numérico, fecha, etc.)
LONG_CAMPO	Numérico	Define el tamaño del campo, es decir, el número de caracteres permitidos.
LLAVE_CAMPO	Alfanumérico	Si los datos permitidos en el campo de la base de datos obedecen a un listado previamente definido, es en este campo donde se especifica el nombre de la tabla que guarda dichos valores.
ORDEN_CAMPO	Numérico	El orden del campo en la tabla a la que pertenece.
ESLLAVE_CAMPO	Alfanumérico	Indica si el campo forma parte de la llave primaria de la tabla o no.

Tabla 4-23. Descripción de campos de tablas.

La tabla UNIDADES_MEDIDA que es la tabla que almacena las diferentes unidades de medida tales como hospitales, centros escolares, etc, y su respectiva imagen a mostrar, necesita una interfaz especial, la cual deberá permitir agregar nuevos registros junto con el cargado de la imagen en el directorio correspondiente del servidor. Así mismo debe permitir la eliminación de los mismos junto con la eliminación del archivo imagen que la describe.

Por tanto, se necesitan una serie de pantallas tales como:

- 1) Listado de imágenes con su nombre y figura en miniatura, que incluya además opciones para la inserción y eliminación de los registros.
- 2) Ventanas interactivas que indiquen paso a paso el procedimiento para el cargado de los archivos imágenes y el correspondiente nombre en la base de datos.
- 3) Cuadros de diálogo que interactúen con el usuario para mostrar advertencias de eliminación.

5.1 CREACIÓN DEL MAPA.

5.1.1 Desarrollo de mapa.

Creación del mapa del municipio de San Salvador.

Constituye como parte importante del Proyecto, todos los procesos relacionados con el mapa y su creación. A continuación se describen los detalles esenciales que describen el proceso seguido para incluir el mapa del municipio de San Salvador a la aplicación.

Inicialmente se contó con un mapa en plano del municipio. Dicho mapa se digitalizó en una empresa de servicios de documentos, y es entonces que se contó con un mapa del municipio en formato raster, como una imagen gráfica, en formato *.BMP (Figura 5-1).

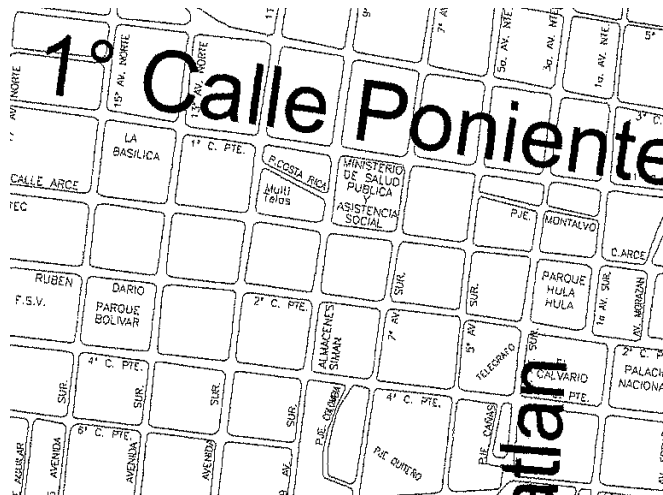


Figura 5-1. Mapa Digitalizado convertido en bmp.
Sección del mapa del municipio de San Salvador.

Se observa una porción del mapa, en formato raster *.BMP. Ninguno de los elementos (manzanas, áreas, etc.) es individualmente diferenciable. Al visualizar a diversas escalas, se percibe pérdida de precisión y claridad.

Luego de haber analizado las ventajas y desventajas existentes entre formatos raster y vectorial, se determino que el formato vectorial cumple adecuadamente las necesidades que la aplicación exige del mapa digital, por lo que se procedió al se trazado vectorial. Para ello, en primer lugar se graficaron todas las manzanas que forman parte del área de trabajo. Ello implicó el trazado a mano de cada una de las áreas formadas por las manzanas, haciendo uso de las herramientas de dibujo proporcionadas por Jasc Web Draw. Al final de este proceso, se tuvo claramente dividido el mapa en manzanas correspondiente a la Figura 5-2.

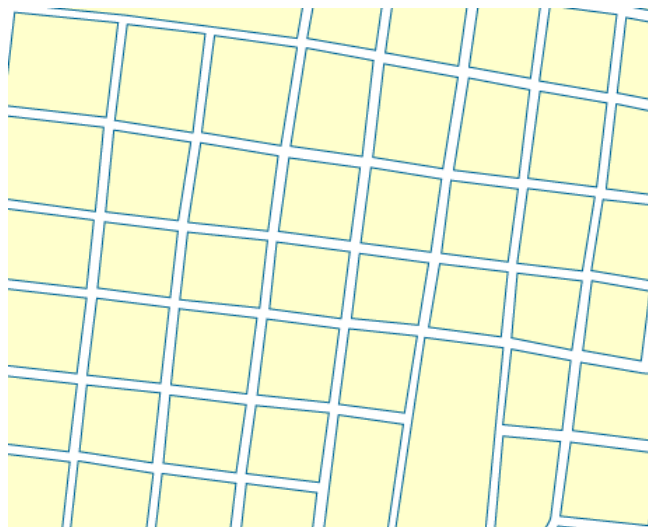


Figura 5-2. Estructura de Calles y Manzanas.

Se muestra la misma sección del mapa reflejado en la Figura 1, pero en formato vectorial *.SVG. Cada elemento es manipulable individualmente.

No hay pérdida de claridad o precisión al visualizar a diversas escalas.

Sin embargo, se necesitaba la inclusión de nombres de referencia que ayudasen a identificar los elementos reflejados en el mapa (Figura 5-3). Se colocaron entonces, nombres a las calles representadas, así como otros elementos fácilmente identificables como parques, estadios, centros comerciales, etc.



Figura 5-3. Calles identificadas con nombre.

La misma sección de mapa, ahora con nombres de referencia.

A partir de este punto, es necesaria la división del mapa en cantones para cumplir con los alcances y objetivos del Proyecto.

Por ello se hicieron las gestiones para la adquisición de un mapa del municipio de San Salvador dividido en cantones. Se da inició con el mapa adecuado, el trazado de los cantones en el mapa de manzanas anteriormente creado. Como siguiente paso, se procedió a relacionar cada manzana y sus nombres con su respectivo cantón. Todos los elementos pertenecientes a un mismo cantón, se agruparon en una sola capa del grafico vectorial. Se realizaron en ese momento, tareas de apariencias: colores de fondo, delineado, características estéticas. Al final de esa etapa, prácticamente se tiene completo el mapa, pero aún falta una etapa de afinamiento del mismo, a fin de acoplarlo a las necesidades del formato vectorial SVG.

Para esta fase, haciendo uso del ambiente proporcionado por Jasc Web Draw 1.0.1, se manipuló el mapa en formato SVG y se le agregaron las características de codificación necesarias para su adición al Proyecto en una página web (Figura 5-4a y Figura 5-4b).



Figura 5-4a



Figura 5-4b

Figura 5-4. Mapa final y muestra de código SVG generado.

Figura 5-4ª. Mapa del municipio de San Salvador dividido en sus cantones.

Figura 5-4b. Sección inicial del código SVG en el que se describen los elementos del mapa.

5.1.2 Etapa de manipulación de mapa.

El mapa se encuentra definido claramente en un archivo formato SVG, donde se encuentran plasmadas sus características y propiedades, es decir, colores, delineados, transparencias, tamaños y ubicación de cada uno de los elementos que en él se visualizan.

Por ello, para cambiar las cualidades del mapa es necesario modificar o crear nuevamente el archivo SVG que define el mapa, por lo que es indispensable conocer las especificaciones del formato SVG, para saber cuales serán las consecuencias de los cambios que se realicen, así como contar con una herramienta propia para el diseño en SVG.

Se definen los elementos del mapa de esta forma:

De poca importancia. Se incluyen acá los nombres de calle, manzanas, etc. Son elementos cuya modificación, inserción o eliminación no influyen en el funcionamiento del Sistema, sino más bien en la visualización del mapa.

De gran importancia. Elementos como los cantones, municipios y departamentos. La modificación de sus propiedades influye nuevamente en la visualización del mapa, pero la modificación de los métodos que utilizan o los eventos a los que responden, tienen severa influencia en el funcionamiento del Sistema.

Es de vital importancia seguir la sintaxis establecida por el formato SVG. Faltar a una de esas reglas puede evitar la óptima y adecuada visualización del mapa, o incluso impedir el funcionamiento del sistema. Por ello, la realización de estas actividades deber ser realizada por una persona autorizada para ese fin, es decir, con los permisos y conocimientos necesarios para acceder a esta parte del sistema.

En esencia, la estructura en formato SVG requerida para el óptimo funcionamiento del sistema es la siguiente:

```

<svg>
  elementos_del_sistema
  <g paneles de funciones>
    <g panel de navegacion>
      barra_de_zoom
      control_de_desplazamiento
      mapa_de_orientación
      seguimiento_de_coordenadas
    </g>
    <g panel de selección de obras>
      cheques para seleccion_de_capas
      (generados dinámicamente mediante servlet)
    </g>
    <g panel de selección de obras>
      iconos para captura de obra/proyecto
      (generados dinámicamente mediante servlet)
    </g>
  </g>
</svg id_pais estilo_propiedades_eventos_metodos_utilizados ">
<Definición de scripts utilizados/>
<g id="id_departamento" estilo_eventos_metodos_utilizados >
  <path path_del_departamento>
    <g id="id_municipio" estilo_eventos_metodos_utilizados >
      <path path_del_municipio>
        <g id="id_canton" estilo_eventos_usados>
          <path path_de_canton>
            elementos_de_canton
            manzanas nombres_calle lugares
          </g>
        <g id="id_canton" estilo_eventos_usados>
          <path path_del_canton>
            elementos del canton
          </g>
        </g>
      </g>
    </g>
  </g>

```

```

                                manzanas nombres_calle lugares
                                </g>
                                .
                                .
                                .
                                </g>
                                .
                                .
                                .
                                </g>
                                .
                                .
                                .
                                <g id="obras">
                                </g>
                                </svg>

                                </svg>

```

A la hora de modificar o crear un nuevo mapa debe seguirse esta misma estructura, poniendo especial cuidado en los estilos, eventos y métodos propios de cada objeto.

5.2 CREACIÓN DE ENTRADA Y SALIDA DE DATOS.

5.2.1 Salida de Datos.

La mayor riqueza de este Sistema es la visualización gráfica de la ubicación de las obras.

La pantalla de selección de los criterios para el filtrado de datos, constituye la información de las obras que se encuentran registradas en la base de datos; por esto se desarrolla una pantalla como la que se muestra en la figura 5-5.



Figura 5-5.

Selección de características para generación de reporte.

Se extraen de la base de datos las instituciones, tipos de obra, etapas, departamentos, municipios y años registrados. Estos datos son cargados inicialmente en la página.

Al seleccionar instituciones, tipos de obras, o departamentos, se lleva a cabo una consulta interna que permite cargar los elementos dentro de la pantalla pero esta vez con datos que coincidieron con el elemento seleccionado de las listas, desde este punto ya se ha realizado un filtrado de datos, que puede proporcionar al usuario una idea de la información que puede ser visualizada o no dentro del mapa.

El botón mostrar permite ver los resultados en el mapa tal como se muestra en la figura 5-6, con las características de navegación, interacción con el usuario, zoom, y acceso a la información en forma de consulta.

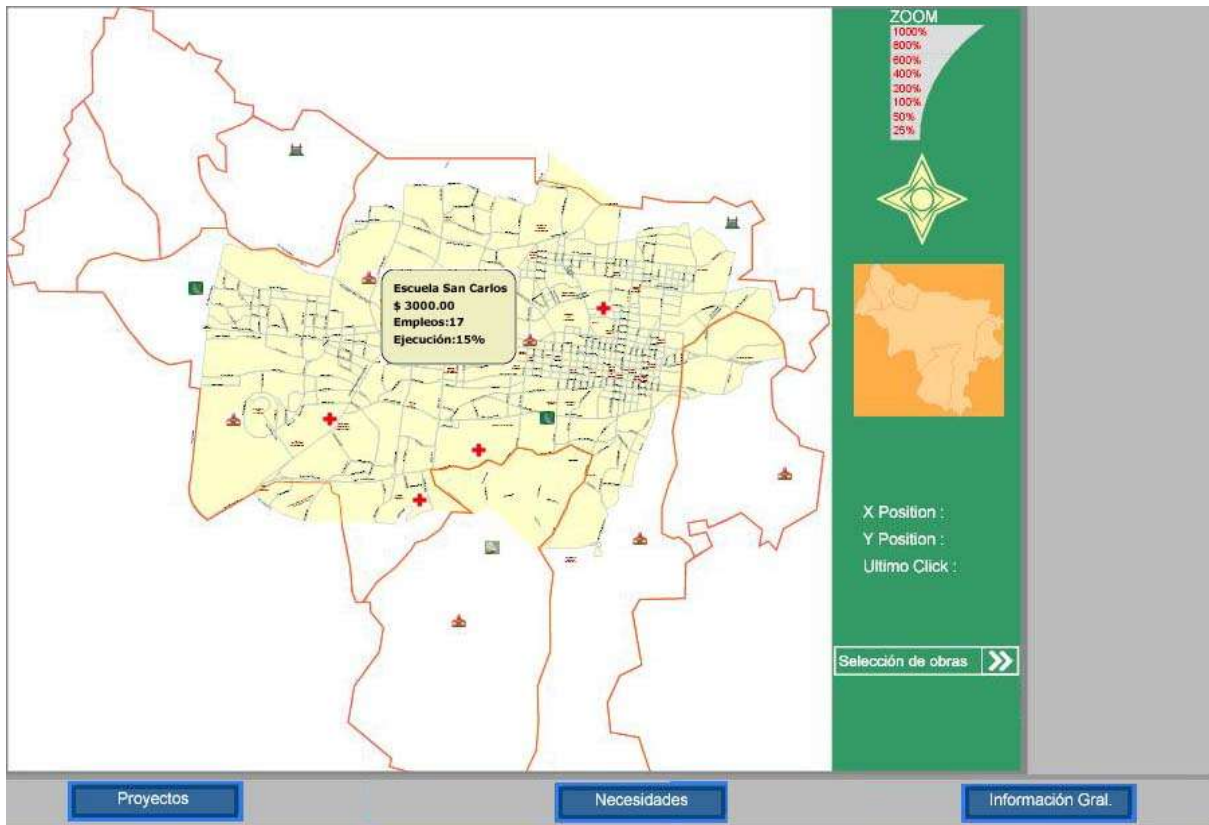


Figura 5-6.

Pantalla de mapa con resultados de la consulta.

Esto permite de igual forma la selección de capas por tipo de obra en el caso no se haya seleccionado un tipo de obra en particular en la pantalla de criterios de selección (Figura 5-5).

Para las salidas de tipo reporte se han generado los procedimientos almacenados en SQL Server, los cuales devuelven los resultados deseados debido a los parametros establecidos que serán definidos por el usuario al momento de la petición e interacción con el sistema.

Para su generación en pantalla se han utilizado los servlets de java, los cuales verifican el usuario que esta haciendo la consulta, para realizar las restricciones necesarias, para mantener el menú que le corresponde y para dibujar los datos devueltos por la base de datos en un diseño amigable para el usuario.

Reportes Consolidados Secretaría Técnica de la Presidencia Año:2003																																																			
<table border="1"> <thead> <tr> <th>Mes</th> <th>Programado</th> <th>Ejecución</th> </tr> </thead> <tbody> <tr><td>Enero</td><td>3000.00</td><td>2000.00</td></tr> <tr><td>Febrero</td><td>3500.00</td><td>.00</td></tr> <tr><td>Marzo</td><td>300.00</td><td>.00</td></tr> <tr><td>Abril</td><td>200.00</td><td>.00</td></tr> <tr><td>Mayo</td><td>1000.00</td><td>.00</td></tr> <tr><td>Junio</td><td>500.00</td><td>.00</td></tr> <tr><td>Julio</td><td>300.00</td><td>.00</td></tr> <tr><td>Agosto</td><td>200.00</td><td>.00</td></tr> <tr><td>Septiembre</td><td>1000.00</td><td>.00</td></tr> <tr><td>Octubre</td><td>500.00</td><td>.00</td></tr> <tr><td>Noviembre</td><td>300.00</td><td>.00</td></tr> <tr><td>Diciembre</td><td>200.00</td><td>.00</td></tr> </tbody> </table>	Mes	Programado	Ejecución	Enero	3000.00	2000.00	Febrero	3500.00	.00	Marzo	300.00	.00	Abril	200.00	.00	Mayo	1000.00	.00	Junio	500.00	.00	Julio	300.00	.00	Agosto	200.00	.00	Septiembre	1000.00	.00	Octubre	500.00	.00	Noviembre	300.00	.00	Diciembre	200.00	.00	<table border="1"> <thead> <tr> <th>Tipos de Obra</th> <th>Programado</th> <th>Ejecutado</th> </tr> </thead> <tbody> <tr> <td>Centro Escolar</td> <td>5500.00</td> <td>1000.00</td> </tr> <tr> <td>Unidad de Salud</td> <td>5500.00</td> <td>1000.00</td> </tr> </tbody> </table>			Tipos de Obra	Programado	Ejecutado	Centro Escolar	5500.00	1000.00	Unidad de Salud	5500.00	1000.00
Mes	Programado	Ejecución																																																	
Enero	3000.00	2000.00																																																	
Febrero	3500.00	.00																																																	
Marzo	300.00	.00																																																	
Abril	200.00	.00																																																	
Mayo	1000.00	.00																																																	
Junio	500.00	.00																																																	
Julio	300.00	.00																																																	
Agosto	200.00	.00																																																	
Septiembre	1000.00	.00																																																	
Octubre	500.00	.00																																																	
Noviembre	300.00	.00																																																	
Diciembre	200.00	.00																																																	
Tipos de Obra	Programado	Ejecutado																																																	
Centro Escolar	5500.00	1000.00																																																	
Unidad de Salud	5500.00	1000.00																																																	
<table border="1"> <thead> <tr> <th>Etapa</th> <th>Programado</th> <th>Ejecutado</th> </tr> </thead> <tbody> <tr> <td>En Ejecución Física</td> <td>11000.00</td> <td>2000.00</td> </tr> </tbody> </table>	Etapa	Programado	Ejecutado	En Ejecución Física	11000.00	2000.00	<table border="1"> <thead> <tr> <th>Fuente de Financiamiento</th> <th>Programación</th> <th>Ejecución</th> </tr> </thead> <tbody> <tr> <td>BID - BANCO INTERAMERICANO DE DESARROLLO</td> <td>22000.00</td> <td>4000.00</td> </tr> </tbody> </table>			Fuente de Financiamiento	Programación	Ejecución	BID - BANCO INTERAMERICANO DE DESARROLLO	22000.00	4000.00																																				
Etapa	Programado	Ejecutado																																																	
En Ejecución Física	11000.00	2000.00																																																	
Fuente de Financiamiento	Programación	Ejecución																																																	
BID - BANCO INTERAMERICANO DE DESARROLLO	22000.00	4000.00																																																	
<table border="1"> <thead> <tr> <th>Institución</th> <th>Programado</th> <th>Ejecutado</th> </tr> </thead> <tbody> <tr> <td>Secretaría Técnica de la Presidencia</td> <td>11000.00</td> <td>2000.00</td> </tr> </tbody> </table>				Institución	Programado	Ejecutado	Secretaría Técnica de la Presidencia	11000.00	2000.00																																										
Institución	Programado	Ejecutado																																																	
Secretaría Técnica de la Presidencia	11000.00	2000.00																																																	

Figura 5-7.

Reportes consolidados de acuerdo a las características del usuario.

Algunos de estos reportes dan opción a activar un reporte de listado de proyectos en un detalle más específico, para el usuario que así lo deseará (figura 5-8).

PROYECTOS DE Centro Escolar España						
Proyecto	Tipo de Proyecto	Etapa	Empleos Generados	Beneficiarios	Monto Programado	Monto Ejecutado
Construcción de cocina	Ampliación Infraestructura	En Ejecución Física	25	50	3000.00	.00

Figura 5-8.

Reportes Listado de proyectos con características específicas.

5.2.2 Entrada de datos.

Captura de Obras.

El levantamiento principal de este tipo de información es recogido a partir del mapa presentado en la página Web como el de la figura (5.2.1 Figura 5), donde el usuario es capaz de ubicar las obras de forma visual haciendo uso del ratón.

Este proceso será explicado en detalle más adelante.

Captura de Necesidades.

A toda obra se le pueden registrar las llamadas “necesidades”. Se llama así a aquellas actividades de necesaria realización, pero que aun no se cuenta con fondos suficientes para llevarlas a cabo.

Se cuenta con un botón “Necesidades”, el cual permite la visualización de un listado de todas las necesidades registradas para dicha obra (figura 5-9).

El servlet ListadoNecesidades, el cual recibe como parámetro el código de la obra y el usuario, realiza la consulta a la base de datos sobre las necesidades registradas por el usuario activo y la obra seleccionada.



Figura 5-9. Pantalla de listado de necesidades

Cada código de necesidades posee un enlace que genera el formulario con la información que describe a la necesidad.

Al inicio de la página posee un cuadro de texto, al cual deberá introducirle el código de una necesidad ha ingresar, y luego presionar el botón de Adicionar.

El botón adicionar o los enlaces en los códigos, ejecutan el servlet CapturaNecesidades, el cual recibe como parámetros el código de la obra y el código de la necesidad, los elementos del registro seleccionado son mostrados en un formulario (figura 5-10).

Obra:	32 - Centro Escolar España
Código:	1
Descripción:	Para poder implementar el programa escuela saludable, es necesario contar con la infraestructura adecuada
Monto:	3000.00
Fecha Ingreso:	01/01/2000
Responsable:	Carmen Suarez
<input type="button" value="Actualizar"/>	

Figura 5-10. Pantalla captura de necesidades

Si se realizan cambios en el formulario de la figura 5-10, puede actualizar o Adicionar según sea el caso. Al presionar el botón se activa el servlet ActualizacionNecesidades, el cual recibe toda la información del formulario y realiza la operación que corresponda: inserción o actualización.

Luego se muestra la información que ha sido modificada en la base de datos (Figura 5-11), en un formato similar, pero sin tener acceso a los cuadros de introducción de datos.

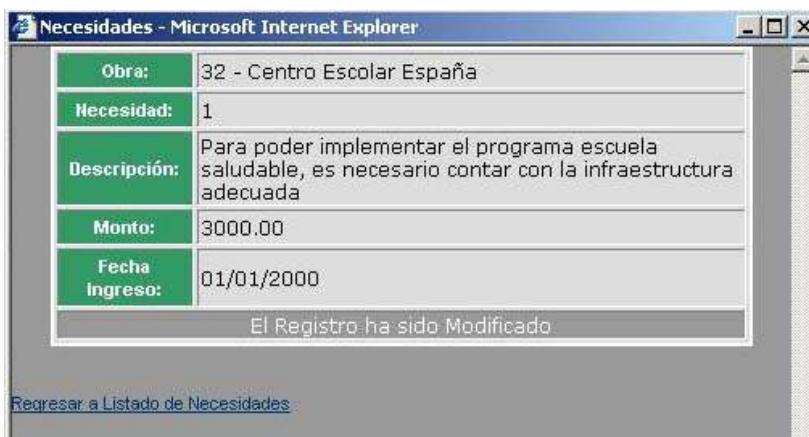


Figura 5-11. Pantalla actualización de necesidades

Captura de Proyectos.

El botón para la captura de proyectos se activa cuando una obra ha sido seleccionada. Este botón permite visualizar un listado de todos los proyectos de la obra seleccionada y que pertenecen al usuario que esta realizando la consulta. Este listado es generado a través de un servlet llamado ListadoProyectos (figura 5-12). El servlet recibe como parámetro el código de la obra seleccionada y el código del usuario, que hace una petición a la base de datos, donde se solicitan todos los proyectos que estén relacionados con la obra indicada y cuyo propietario sea el usuario que realiza la consulta. Este servlet genera una pagina html que devuelve el listado con el código y nombre de cada proyecto encontrado con estas características. Se presenta además la opción de agregar un nuevo proyecto. Cada elemento de la lista contiene un enlace hacia un formulario donde se presentan todos los datos del proyecto con la opción de modificar.



Figura 5-12. Pantalla consulta de proyectos

Cuando un proyecto de la lista es seleccionado, el servlet CapturaProyectos es llamado. Este servlet recibe como parámetro el código de la obra y el código del proyecto. Genera una página html con toda la información del proyecto registrada en la base de datos y presentada en un formulario (Figura 5-13).

Proyectos	
Obra:	32 - Centro Escolar España
Código:	123
Tipo de Proyecto:	Reparación Infraestructura
Nombre:	<input type="text"/>
Descripción:	<input type="text"/>
Fecha Inicio Programada:	<input type="text"/>
Fecha Final Programada:	<input type="text"/>
Fecha Inicio Real:	<input type="text"/>
Fecha Final Real:	<input type="text"/>
Etapa:	Banco de Preinversión
Estado:	A tiempo
Empleos Generados:	<input type="text"/>
Beneficiarios:	<input type="text"/>
Necesidad:	<input type="text"/>
Responsable:	Carmen Suarez
<input type="button" value="Guardar"/>	
Regresar a Listado de Proyectos	
Ver Listado de Fuentes de Financiamiento	
Selección de período para seguimiento	

Figura 5-13. Pantalla captura de proyectos

Como puede verse en la figura 5-13, en la parte superior de la pagina se presenta el código y nombre de la obra a la que pertenece el proyecto, luego el código del proyecto, los campos ya sean numéricos o alfanuméricos que sus datos no excedan de 25 caracteres se presentan en cuadros de textos, si el campo es de texto, entonces la información se muestra en objetos de área de texto, los campos que tienen relación

con otra tabla se presentan en forma de combo o lista de selección, cuyas opciones se obtienen de la tabla a la cual hacen referencia.

El usuario puede realizar todos los cambios que requiera y luego presionar el botón Actualizar. Si se trata de una adición de proyectos, el formulario es el mismo, aunque todos los elementos del formulario aparecen vacíos y al final de este se presenta un botón Agregar.

Al final de la página se presentan dos enlaces, uno que muestra el listado de fuentes que tienen participación en el proyecto, y el enlace para mostrar la selección de programación por año del proyecto.

La actualización o adición la realiza otro servlet llamado ActualizacionProyectos, quien recibe toda la información del formulario, así como el código de la obra y del proyecto, y si es una adición o modificación de proyectos. Luego, dependiendo de la acción que deba ejecutar, se compone la cadena que podrá ser un insert o un update a la tabla proyectos.

Al haber realizado los cambios en la base de datos, se muestran todos los campos que describen este proyecto a manera de confirmación al usuario de los cambios realizados, y se muestra el mensaje “El Proyecto ha sido modificado” o “El proyecto ha sido agregado” (Figura 5-14).

Al final de la página se presentan dos enlaces, uno que muestra el listado de fuentes que tienen participación en el proyecto, y el enlace para mostrar la selección de programación por año del proyecto.

Obra:	6 - Prueba de ingreso
Proyecto:	2
Nombre:	Compra de mobiliario para el area de emergencias
Descripción:	El mobiliario existente se encuentra deteriorado
Fecha Inicio Programada:	2003-01-02
Fecha Final Programada:	2003-12-01
Fecha Inicio Real:	2003-01-02
Fecha Final Real:	2003-12-01
Etapas:	No Iniciado
Estado:	A tiempo
Empleos:	150
Beneficiarios:	500
Tipo de Proyecto:	Mejoramiento Infraestructura

El Registro ha sido Modificado

[Regresar a Listado de Proyectos](#)
[Ver Listado de Fuentes de Financiamiento](#)
[Selección de periodo para seguimiento](#)

Figura 5-14. Pantalla actualización de proyecto

Captura de Fuentes de Financiamiento.

El listado de las fuentes de financiamiento involucradas en un proyecto, es generado por un servlet llamado ListadoFuentes. Este servlet puede ser llamado desde la captura de un proyecto seleccionado para su modificación, o después de haber realizado la inserción o actualización de un proyecto, desde la página de confirmación de actualización. El servlet ListadoFuentes, recibe como parámetro el código de la obra y el código del proyecto, se realiza la consulta a la base de datos sobre las fuentes registradas para este proyecto y se muestra un listado de todos los resultados. También se presenta la opción de agregar una nueva fuente de financiamiento que tenga participación en el proyecto seleccionado (Figura 5-14).

Código Fuente:	20-30SANVI	Adicionar
-----------------------	------------	-----------

Fuente	Monto
BID	3000.00

Figura 5-15. Pantalla consulta fuentes de financiamiento.

Cada elemento de la lista, posee un enlace que al ser seleccionado presenta la pantalla de captura de fuentes de financiamiento (figura 5-16).

Obra:	32 - Centro Escolar España
Proyecto:	1 - Construcción de cocina
Fuente:	BID - BID - BANCO INTERAMERICANO DE DESARROLLO
Tipo de financiamiento:	Préstamo
Monto:	3000.00
<input type="button" value="Actualizar"/>	
Regresar a Listado de Fuentes de Financiamiento	
Regresar a Listado de Proyectos	
Selección de período para seguimiento	
<input type="button" value="Ir a Seguimiento del año:"/> 2003	<input type="button" value="Aceptar"/>

Figura 5-16. Pantalla de captura fuente de financiamiento.

La pantalla de captura de fuentes de financiamiento es generada por un servlet llamado CapturaFuentes.

En la parte superior de la página se muestra el código y nombre de la obra, código y nombre del proyecto, código y nombre de la fuente de financiamiento. A continuación se muestran los campos en forma de formulario en los cuales se pueden hacer los cambios necesarios y presionar el botón Actualizar para que las modificaciones sean almacenadas.

En la parte inferior de la página se muestra el enlace para regresar al listado de fuentes, al listado de proyectos o para ir al seguimiento del proyecto activo.

Si se quiere agregar una nueva fuente de financiamiento, se selecciona una fuente de la lista de selección que aparece en la pagina listado de fuentes de financiamiento y se presiona el botón Agregar. Luego, aparecerá el formulario de captura de fuentes, con los elementos del formulario en blanco, listos para ser llenados por el usuario. En la parte inferior de la pagina se presenta el botón Adicionar. Ya sea que se presione el botón Adicionar o Modificar en cualquiera de los casos, se llamará el servlet ActualizacionFuentes, el cual recibe como parámetro el código de la obra, el código del proyecto y el código de la fuente de financiamiento junto con todos los elementos

del formulario. Se forma la cadena de sentencia según sea el caso: inserción o actualización.

Luego de realizar la operación de modificación de la tabla proyectos_fuentes, se presentan los datos a manera de verificación para el usuario y se envía el mensaje “El Registro ha sido modificado” o “El Registro ha sido adicionado”. (figura 5-17)

Obra:	6 - Prueba de ingreso
Proyecto:	2 - Compra de mobiliario para el area de emergencias
Fuente:	BID - Banco Interamericano de Desarrollo
Tipo de Financiamiento:	Donación
Monto:	2000

El Registro ha sido Modificado

[Regresar a Listado de Proyectos](#)
[Regresar a Listado de Fuentes de Financiamiento](#)
[Selección de período para seguimiento](#)

Figura 5-17. Pantalla resultado de actualización fuentes de financiamiento

En la parte inferior de la página se muestra el enlace para regresar al listado de fuentes, al listado de proyectos o para ir al seguimiento del proyecto activo.

Captura de Programación y avances del proyecto.

Desde la pantalla de captura y actualización de proyectos y fuentes de financiamiento se puede acceder a la pantalla para hacer la selección del periodo a registrar o modificar de la programación y avances del proyecto (figura 5-18).

Selección período de seguimiento:

Año: 2003

Del mes: Enero Al mes: Enero

Aceptar

Captura nuevo seguimiento para el año:

Aceptar

Figura 5-18. Pantalla consulta para Programación y avances de Proyecto

El servlet encargado de generar la pantalla anterior es llamado SeleccionPeriodo, el cual recibe como parámetro el código de la obra y del proyecto. Para seleccionar un periodo de programación y ejecución se debe seleccionar de la lista de selección de año, y definir de que mes a que mes se requiere la visualización de registros.

Si se desea agregar un nuevo registro se ingresa el año para la nueva programación en el espacio que dice “Captura nuevo seguimiento para el año:” Luego se hace clic e el botón Aceptar.

El botón Aceptar llama a ejecución el servlet CapuraSeguimiento, el cual recibe los parámetros código de la obra, código del proyecto, el año y el periodo de meses a desplegar.

Obra: 32 - Centro Escolar España Proyecto: 1 - Construcción de cocina Año: 2003					
Mes	Prog. Finan.	% Prog. Física	Ejec. Finan	% Ejec. Física	Observaciones
Enero	500.00	5.00	.00	.00	
Febrero	250.00	5.00	.00	.00	
Marzo	150.00	10.00	.00	.00	
Abril	100.00	10.00	.00	.00	
Mayo	500.00	10.00	.00	.00	
Junio	250.00	10.00	.00	.00	
Julio	150.00	10.00	.00	.00	
Agosto	100.00	10.00	.00	.00	
Septiembre	500.00	10.00	.00	.00	

Actualizar

Figura 5-19. Pantalla captura avance.

Al inicio de la página se muestra el código y nombre de la obra, el código y nombre del proyecto, el año de programación y a continuación se muestra en un formulario todos los campos correspondientes a la programación y ejecución física tanto física como financiera como las respectivas observaciones que pudieran ser necesarias para los comentarios necesarios. (Figura 5-19)

Cuando se registre una nueva programación, el usuario visualizara en su pantalla el espacio para llenar los 12 meses del año.

Al final de la pantalla se encuentra el botón Actualizar, el cual ejecutará el servlet ActualizacionSeguimiento, que recibe todos los parámetros para poder realizar las modificaciones efectuadas en el formulario.

La pantalla de actualización muestra los datos modificados, y presenta un mensaje “Los registros han sido modificados”

Mes:	Programación Financiera	Programación Física	Ejecución Financiera	Ejecución Física	Observaciones
Enero	500.00	5.00	.00	.00	
Febrero	250.00	5.00	.00	.00	
Marzo	150.00	10.00	.00	.00	
Abril	100.00	10.00	.00	.00	
Mayo	500.00	10.00	.00	.00	
Junio	250.00	10.00	.00	.00	
Julio	150.00	10.00	.00	.00	
Agosto	100.00	10.00	.00	.00	
Septiembre	500.00	10.00	.00	.00	

Los Registros han sido modificados

[Regresar a Listado de Proyectos](#)
[Ver Listado de Fuentes de Financiamiento](#)
[Selección de periodo para seguimiento](#)

Figura 5-20. Pantalla resultado actualización de seguimiento.

Al final de la página se muestran los enlaces para regresar a la información del proyecto, a la información de las fuentes, o a la selección del periodo.

5.2.3 Interfaz de administración

Para la administración de las tablas que contienen los datos del sistema se han creado tres servlets:

- 1) Servlet de listado de registros de la tabla
- 2) Servlet de formulario con los respectivos campos de la tabla
- 3) Servlet de actualización de registros.

Para que estos tres programas funcionen correctamente es necesario llenar los datos de cada tabla, con sus respectivos campos y características de la misma.

El servlet que muestra el listado de los registros existentes en la tabla, toma como parámetro la tabla que se esta modificando y realiza la consulta, aplicando un formato a los resultados obtenidos.



The screenshot displays a web interface titled "Administración de la tabla OBRAS_FISICAS". Below the title is a link labeled "Adicionar Registro". The main content is a table with two columns: "Código" and "Nombre de la obra". The table contains 16 rows of data, with the last row labeled "Nombre Obra".

Código	Nombre de la obra
32	Centro Escolar España
33	Unidad de Salud Montserrat
34	Centro Escolar Juana Lopez
35	Unidad de Salud Central
36	Unidad de Salud San Benito
37	Unidad de Salud San Jacinto
38	Centro Escolar Juan Lopez
39	Pozo El Carmen
40	nueva obra
41	Centro Escolar "ISIDRO MENENDEZ"
42	Instituto Nacional "SAN LORENZO"
43	Hospital Rosales
44	Pozo el abastecedor
45	Puente El Porvenir
46	Puente El Auxiliar
47	Nombre Obra

Figura 5-21. Pantalla listado de registros de la tabla seleccionada.

El campo llave de la tabla se genera con un hipervínculo tal como se muestra en la figura 5-21 que al activarlo llama al servlet que genera el formulario con los campos necesarios para la modificación del registro. La información para la creación del formulario es tomada de la tabla TABLA_CAMPOS creada para este propósito. Este servlet genera en cada campo del formulario las validaciones correspondientes de acuerdo al tipo de campo, estas validaciones son realizadas con JavaScript.

Si es un registro nuevo el que se está agregando el formulario aparece totalmente en blanco listo para ser llenado por el usuario (en este caso el administrador del sistema) donde se realizan todas las validaciones necesarias de tipos de datos. Se propone un código para el campo llave que no se encuentre en la base de datos, con opción del usuario a ser modificado y siempre con validaciones JavaScript no permitiendo la colocación de un código (llave) ya existente en la tabla.

Finalmente, en el formulario aparecen los botones Adicionar, Modificar o Eliminar según sea el caso.

Al ser activado cualquiera de estos botones, se llama al servlet que realiza la acción solicitada (inserción, actualización o eliminación) y se muestra un mensaje de confirmación de la acción.

La administración de la tabla UNIDADES_MEDIDA ha sido creada especialmente para este propósito, ya que dicha tabla tiene la característica de almacenar el detalle de cada una de las unidades de medida que se traducen en las posibles imágenes para la representación de obras en el sistema.

Por tanto, se han creado cuatro programas (servlets):

- 1) Listado de registros existentes en la tabla (UNIDADES_MEDIDA).

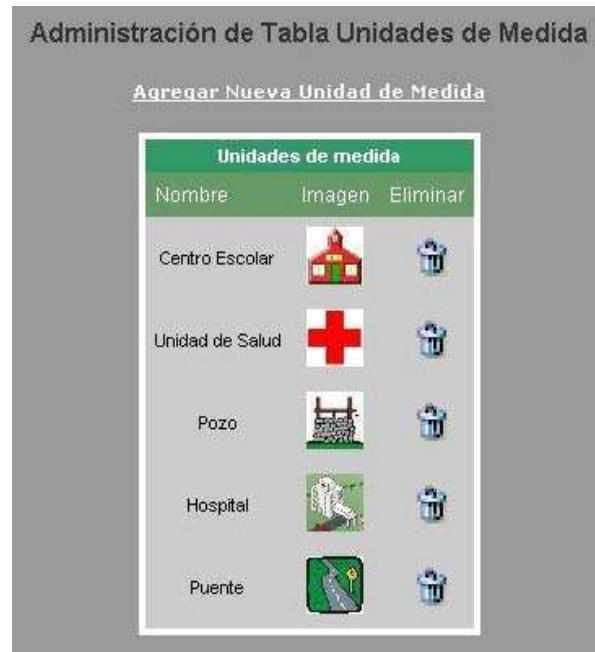


Figura 5-22. Pantalla listado de registros de la tabla unidades_medida.

- Definición de características al momento de agregar un nuevo registro (definición de datos en los campos).



Figura 5-23. Pantalla Paso 1 para inserción de registros de la tabla unidades_medida.

- Copiado de la imagen seleccionada al directorio definido donde se almacenan las unidades de medida.

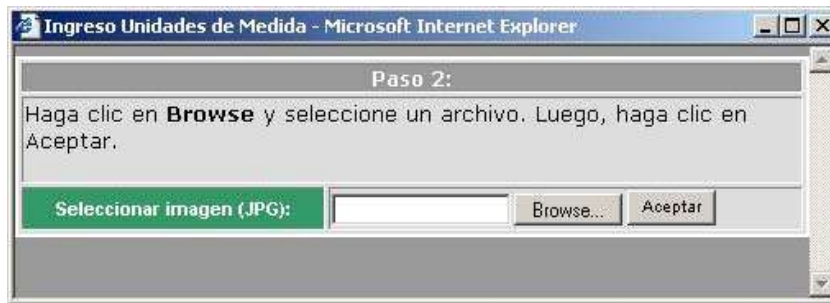


Figura 5-24. Pantalla Paso 2 selección de archivo imagen.

Al momento de presionar el botón Aceptar, el servlet genera un mensaje de confirmación (figura 5-25).

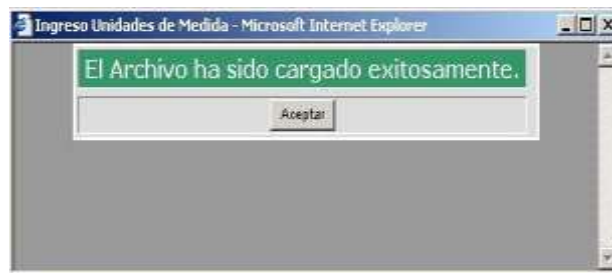


Figura 5-25. Mensaje de confirmación de inserción.

- 4) Eliminación de un registro, que incluye la eliminación del archivo imagen del directorio definido para las imágenes de las unidades de medida.

Antes de realizar la acción de eliminación, advierte al usuario sobre la acción a ejecutar.



Figura 5-26. Mensaje de advertencia.

Cuando el usuario confirma la acción de eliminación, se activa el servlet de eliminación, el cual concluye con un mensaje de confirmación de la acción (Figura 5-27).



Figura 5-27. Mensaje de confirmación de eliminación.

5.3 CREACIÓN INTERFAZ DE USUARIO.

La función principal del menú es la de mostrar las opciones a las que únicamente el usuario que ha ingresado debe tener acceso. Este es generado haciendo uso de servlets extrayendo los datos de la base de datos, y es manipulado mediante funciones JavaScript.

Los items del menú se encuentran almacenados jerárquicamente en la base de datos. Se tienen así menús padre, los cuales despliegan menús hijos. Dicha relación está definida en la base de datos, de acuerdo a la forma en la que el item de menú ha sido almacenado. Esta información es leída haciendo uso de funciones Java Servlets.

Se crean inicialmente los menús padres y son colocados en el documento HTML como elementos `<div>nombre_menu</div>` en una posición inicial, debido a que no todos los nombres de ítem tienen la misma longitud se ha utilizado un método que permite una distancia de separación similar entre ellos dando una imagen de equidistancia, si esta operación no se realiza, todos los menús estarían en la misma posición y no podrían ser visualizados correctamente ni podrían ser accedidos por el usuario.



Figura 5-28. Menú generado para el usuario activo.

Los cuadros que contendrán los submenús de cada menú padre se crean mediante funciones de Java Servlets realizando consultas a la base de datos de la aplicación.

Dichos recuadros inicialmente al ser cargado el Servlet están ocultos ante la vista del usuario y únicamente podrán ser visualizados en el momento que pase el mouse sobre el menú y se ocultarán cuando el puntero no este sobre el menú, para esto se hace uso de las propiedades `OnMouseOver` y `OnMouseOut` de cada elemento `<div></div>` creado (figura 5-28).

Cada una de las propiedades invoca una función `Show(param1, param2, param3, param4)` y `Hide(param1, param2)` para el manejo de los recuadros a mostrar o esconder respectivamente. Los parámetros enviados a las funciones llevan el indicador sobre que menú se esta haciendo la llamada, esto proporcionará la ubicación dentro del documento html de dicho `<div>` principal, que menú se esta llamando y los parámetros que servirán para realizar los cálculos de ubicación, como posición en X, nivel en el que se encuentra el submenú, y la profundidad que son operados en píxeles (unidad básica en la que se manejan los elementos en documento HTML), que posteriormente serán asignados a las propiedades de los elementos para ser mostrados o escondidos.

A cada menú principal, recuadro de submenú, y los enlaces de cada uno, se les han creado estilos de color, forma, tipo de letra y propiedades de visualización propias de elementos HTML que pueden ser manipulados mediante JavaScript.

5.4 INTERACTIVIDAD ENTRE SISTEMA, MAPA Y BASE DE DATOS.

5.4.1 Interactividad del Mapa con JavaScript.

El mapa por sí solo no tiene funcionalidad de interactividad con el usuario, por ello son creadas funciones con instrucciones JavaScript que manipulan los eventos, propiedades y atributos de cada uno de los elementos con los que el mapa está formado.

De acuerdo al nivel de acceso e institución del usuario conectado, se organizan las obras visualizadas en mapa por capas, de acuerdo a la institución a la que pertenecen. Por ello, se accede mediante Servlets, las obras almacenadas por institución en la Base de Datos. Luego, mediante sentencias Javascript y propiedades/atributos SVG, se separan las capas en grupos, y se asocian a un cuadro de selección ('checkbox'), que permite al usuario visualizar las capas que le interesan o necesita y a las que tiene autorización.

De igual forma, la información definida en el mapa se especializa de acuerdo al nivel de acceso del usuario conectado, por lo que se revisa la Base de Datos mediante Servlets para escribir un archivo SVG personalizado, que sirva de base para la sesión que el usuario desea establecer, y se colocan así en el mapa solo los elementos a los que el usuario está autorizado a ver, así como las herramientas que le facilitarán su labor.

SVG y JavaScript.

La tecnología SVG permite un significativo nivel de interactividad con el usuario, ello gracias a su compatibilidad con lenguaje Javascript.

Los elementos gráficos definidos en lenguaje SVG poseen gran variedad de características las mostradas en Tabla 5-1.

Propiedad	Significado	Valor
Fill	Color de relleno	#0000FF blue rgb(0,0,255)
Stroke	Color de delineado	#0000FF blue rgb(0,0,255)
Stroke-width	Grosor de delineado	0~1 0.3, 0.47, 1
Opacity	Opacidad, transparencia	0~1 0.3, 0.47, 1
Visibility	Visibilidad	visible,hidden, collapse,inherit
Width	Ancho	300, 625.45
Height	Alto	300, 625.45
Font-face	Fuente de texto	Arial, Courier

Tabla 5-1 Propiedades modificables de elementos SVG

Así, mediante sentencias Javascript se posibilita modificar propiedades de los elementos del mapa, como visibilidad, opacidad, ubicación y apariencia, dimensiones (zoom), etc.

De igual forma, SVG define métodos específicos para el acceso y modificación a propiedades especializadas de sus elementos, como: coordenadas x, coordenadas y, identificadores, referencia a objetos, nivel de visibilidad, nivel de desplazamiento, etc.

Esto implicó la definición y creación de funciones especializadas para el manejo de los elementos específicos del Sistema: el mapa, las obras, etc.

Estas funciones se han definido de dos formas:

Se ha creado un archivo *.js separado del archivo SVG que es necesario incluir dentro del archivo SVG mediante una línea que indica el lugar donde se encuentran las funciones Javascript utilizadas:

```
<script xlink:href="mapa_svg.js" type="text/javascript" />
```

Definidas en el mismo archivo SVG:

```
<svg>
<script type="text/javascript">
<![CDATA[
variables
funciones(){
    sentencias Javascript
    ...
}
...
// ]]>
</script>
elementos SVG
...
</svg>
```

5.4.2 Almacenamiento y acceso de información: Servlets.

Para que el proceso de almacenamiento y captura de información sea dinámico en cuanto a los datos que son manipulados, se ha creado un servlet que contiene la codificación debidamente separada en partes y que es necesaria para llevar a cabo la generación del mapa final a ser presentado al usuario.

A continuación se da una breve explicación de cómo los servlets de Java generan el mapa, los resultados de la aplicación y el consecuente acceso a los datos almacenados.

Se inicia con la apertura del archivo SVG base, mediante el servlet en el que se ha establecido la estructura base que debe buscar e interpretar línea por línea.

Posteriormente se ha establecido una condición base que permite en ese punto exacto la interacción del servlet con la base de datos, es en este punto antes de finalizar la estructura base del mapa el servlet extrae de la base de datos los tipos de obras registradas en el sistema para ser insertadas en una estructura de capas donde cada capa es identificada mediante propiedades SVG por medio de su identificador de tipo de obra, esto permite el posterior agrupamiento de todas las obras registradas bajo esa categoría de obra.

De igual forma estas obras son extraídas de la base de datos haciendo referencia al identificador de tipo de obra, para ser insertadas dentro de la capa correspondiente; cada una de las obras es identificada con su código de base de datos, más la modificación de un evento propio “onClick” al cual le es asignado una función JavaScript que permite establecer su código como activo para tener acceso a su información desde la pantalla del mapa, con los botones de acceso.

Después de esta sección el servlet acude nuevamente a la base de datos para extraer los íconos de los tipos de obra que son utilizados para el posicionamiento de nuevas obras dentro del mapa. Dichos íconos son incrustados dentro del mapa en una capa independiente al cual se le otorga una función JavaScript que manipula los íconos, cada ícono es colocado dentro de un elemento propio del SVG que es debidamente identificado con el código de tipo de obra para su posterior manipulación.

En este punto es cerrado el tag que encierra la estructura del mapa base.

Seguidamente el servlet que lee el archivo SVG y genera el SVG personalizado creará las herramientas de navegación del mapa, como la barra de Zoom, barra de desplazamiento, etc. (que son elementos estáticos del mapa), así como la barra de íconos que incluye: cuadros de checkbox de código SVG, íconos generados mediante la interacción de instrucciones del servlet con la base de datos, su respectiva identificación, etc., que permiten la interacción dentro del mapa para la visualización de las capas por tipo de obra donde son asignadas las funciones JavaScript que

permiten dicha funcionalidad, y para finalizar la barra de menú de los íconos de tipos de obras con su respectiva función JavaScript para la selección del tipo de obra que se desea registrar.

Al finalizar esta serie de instrucciones el servlet termina la generación del mapa a ser mostrado al usuario.

Otros servlets son invocados dependiendo de la acción del usuario, es decir que estos dependen del tipo de información que deseen ingresar o visualizar, dichos servlet siempre interactúan con la base de datos, de acuerdo a la información solicitada.

5.4.3 Captura de obra.

El proceso inicia desde que el usuario accede al botón agregar nueva obra, en ese momento haciendo uso de javascript se invoca la función `mostrar_icons()` que permite visualizar todos aquellos íconos que el usuario actual puede manipular, desde este momento el usuario puede seleccionar un ícono para el ingreso de una nueva obra, al dar clic sobre el ícono deseado se llama la función `mostrar_simbolos()` la cual leerá de las propiedades del mapa el id correspondiente a dicho ícono seleccionado, permitiendo así mantener activo el identificador único del tipo de obra que se quiere ingresar quedando listas las operaciones involucradas en el proceso de nueva obra.

Desde este punto, el usuario puede recorrer el mapa y seleccionar la ubicación donde desea colocar la obra, para esto deberá dar clic sobre el mapa y en ese momento visualizará el ícono seleccionado en la posición marcada. Para llevar a cabo dicho proceso se manipulan las propiedades disponibles de los elementos dentro del `svg/javascript`, es decir, mientras el usuario se está desplazando dentro del mapa la función `changeText()` es llamada en ese momento, la cual tiene la función de capturar la posición actual del puntero del mouse dentro de la pantalla del mapa, estos datos capturados son operados matemáticamente junto con los propiedades obtenidas del mapa en `svg` como tamaño actual del mapa y las nuevas posiciones X y Y que tiene en el área de visualización, dado que si en ese momento se aplicó un zoom las propiedades del mapa son modificadas para poderlo mostrar con el acercamiento o

alejamiento deseado, así como el desplazamiento utilizado (estas funciones de Zoom y Desplazamiento se explicarán adelante más detalladamente).

En el momento que el usuario decide colocar el ícono de la obra dentro del mapa al dar un clic se invocan las funciones `recordClick(evt)` y `MyrecordClick(evt)` que permiten establecer las coordenadas actuales de la obra, estas toman como parámetros los datos calculados previamente por la función `changeText()`.

A este punto el usuario tiene la opción de dejar permanentemente ubicada la obra o desplazarla a otro lugar, si decide que ese será el lugar definitivo de la ubicación de la obra debe dar Clic en el botón Guardar Obra, en ese momento no podrá cambiar de lugar el ícono que representa el tipo de obra que ha decidido registrar, dado que se ejecuta en ese momento el servlet que modifica la Base de Datos e introduce la nueva obra al registro del Sistema. Se debe asignar un nombre a la obra que está por ingresar. Si no lo ingresa, la obra no será almacenada.

Si la obra ha sido debidamente registrada aparecerá un mensaje con el texto “La obra ha sido almacenada satisfactoriamente.”, ahora el usuario podrá agregar proyectos, necesidades, visualizar información general de la misma.

6.1 REGISTRO DE OBRAS.

Permite el ingreso de nuevas obras. Al ingresar se presenta el mapa con las obras existentes, clasificadas de acuerdo a entidad, siendo mostradas en el lugar donde fueron ubicadas por algún usuario autorizado. (figura 6-1)

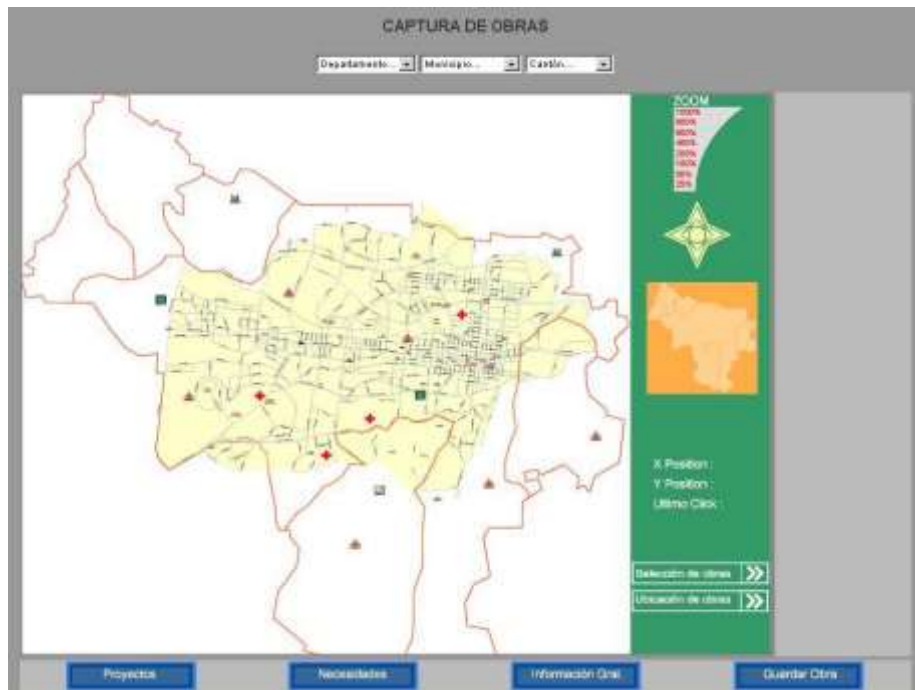


Figura 6-1. Mapa con resultados extraídos de la BDD.

Pasos para el ingreso de nueva obra:

1. Se debe acceder desde el botón “Ubicación de obras”. En ese momento se muestra en la parte derecha del mapa un panel con los íconos de las obras que se pueden ingresar (figura 6-2). Debe darse click sobre la obra de interés. Posteriormente debe darse click en cualquier posición del mapa donde se quiere colocar. En ese momento el ícono aparece en la ubicación marcada.

La posición de las obras es refleja por sus coordenadas dentro del marco del mapa, estas coordenadas x,y serán automáticamente mostradas en el lado

derecho del mapa con las etiquetas **Posición en X:** y **Posición en Y:** (Figura 6-1).

Si se desea cambiar la ubicación de la obra, se da click en otra ubicación dentro del mapa. La obra entonces, se borra de la localidad previa y se dibuja en el último lugar sobre el que el usuario dio click.

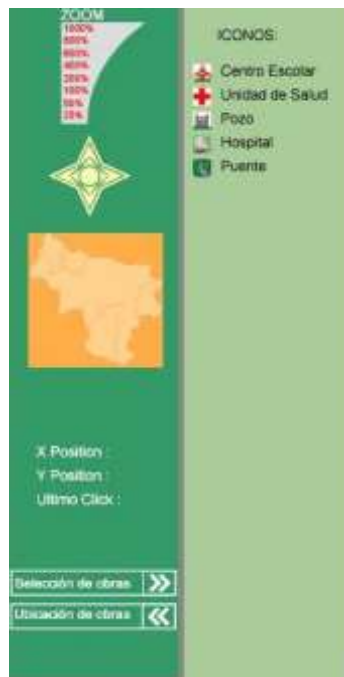


Figura 6-2. Panel con iconos para captura de obras.

2. Si ya se tiene la ubicación definitiva de la obra debe accederse el botón “Guardar Obra”. En este momento se solicita el nombre que se debe asignar a la obra para ser identificada. Para que la obra pueda ser debidamente registrada, debe haberse posicionado la obra en el mapa y habersele asignado el nombre correspondiente en el cuadro de diálogo que es desplegado (Figura 6-3).

La posición final de la ubicación de la obra dentro del mapa será reflejado en el lado derecho con la etiqueta “última posición”.



Figura 6-3. Pantalla de captura nombre de obra.

Luego de proporcionar el nombre de la obra, ésta es almacenada en la base de datos de la aplicación, en este momento se puede tener acceso a los botones Proyectos, Necesidades, Información Gral.

6.2 VISUALIZACIÓN DE INFORMACIÓN.

Las obras que ya estén registradas en la base de datos, son presentadas por la aplicación de manera visual, ubicadas en el mapa, donde fueron colocadas al momento de ser ingresadas.

Dichas obras pueden poseer proyectos o necesidades previamente registradas. Para tener acceso a dicha información debe seleccionarse la obra dando clic sobre ella en la ubicación del mapa, con esto se establece qué obra es la que está activa para mostrar información, en este momento se puede acceder el botón Proyectos, Necesidades o Información Gral.

a) Sección Proyectos.

Abre una ventana independiente mostrando un listado con enlaces a los proyectos registrados, un botón para agregar nuevo proyecto y otro para cerrar ventana. Puede accederse cada nombre de proyecto identificado como link o enlace, se abre entonces una nueva ventana que muestra en detalle la información del proyecto correspondiente a dicha obra.

De forma general se mostrará el nombre de la obra a la cual pertenece el proyecto, esta no podrá ser editada, puesto que no es una característica propia del proyecto. De manera que los usuarios puedan identificar sus propios proyectos deberán ingresar un código único de proyecto, incluir el proyecto dentro de una clasificación, asignarle nombre, ingresar una pequeña descripción del proyecto para una mejor referencia (si el usuario así lo desea), ingresar la fecha de inicio en que se tiene programado comenzar el proyecto así como la fecha estimada para la finalización del mismo; éstas fechas no deben confundirse con las fechas de inicio real y final real, puesto que estas son las fechas en las que el proyecto se puso en marcha y finalizó respectivamente.

Para manejar el estado del proyecto se cuenta con dos características, una es la etapa de trabajo del proyecto, en donde se indica si el proyecto está iniciado, en ejecución o

si está terminado; además, para cada una de las etapas anteriores se puede indicar el estado de la misma, es decir, si la etapa esta a tiempo, adelantada, atrasada, etc.

Como información general también se incluirán el número de empleos generados, el número de beneficiados por el proyecto, la necesidad o justificación por la que se está realizando el proyecto, y principalmente quien es la persona encargada de llevar el control del proyecto.

b) Sección Necesidades.

Abre una ventana independiente mostrando un listado con links de las necesidades registradas, un botón para agregar nueva necesidad y otro para cerrar ventana.

Al dar clic sobre cada nombre de necesidad se abre una nueva ventana que muestra en detalle la información de la necesidad correspondiente a dicha obra, la cual se detalla a continuación:

La necesidad está relacionada con las obras existentes, para esto la pantalla de visualización de información muestra el nombre de la obra a la cual será asignada pero sin ser modificada. Se ingresa: un código de obra único, de modo que el usuario posteriormente lo pueda identificar; una pequeña descripción para explicar la necesidad, de forma que posteriormente sea más fácil entenderla. Se ingresa de igual manera el monto que se ha estimado para llevar a cabo la realización de dicha necesidad, la fecha en la que ha sido ingresada y el responsable a cargo.

c) Selección de Obras.

Al momento de ingresar a la sección del mapa, son cargados los íconos y las obras que han sido ingresadas al sistema y corresponden al usuario que esta haciendo la solicitud de información, dichos elementos son desplegados en el mapa en el lugar en el que fueron colocados al momento de ser registrarlos en la aplicación.

Al lado izquierdo del mapa se muestra un listado por selección de los tipos de obra, cada uno posee un cuadro para ser seleccionado; esto permite mostrar simultáneamente la ubicación de todas las obras contenidas dentro de la clasificación seleccionada. Las diferentes clasificaciones de obras pueden ser mostradas todas al mismo tiempo pero puede resultar que varias obras de diferente clasificación se encuentren ubicadas muy cerca o en el mismo punto de tal forma que no se logren distinguir correctamente, para esto se proporciona la funcionalidad de selección de tipos de obras.

6.3 NAVEGACIÓN EN EL MAPA.

El trabajo con mapas exige una clara visualización de los elementos que se encuentran representados en el mismo. Sin embargo, usualmente, el tamaño de los mapas es mucho mayor al tamaño de las pantallas en las que se muestran al usuario. Entonces se vuelve muy importante proporcionar herramientas que permitan realizar acercamientos y alejamientos (zoom) de la vista del mapa, lo que permite observar el mapa a diferentes niveles. Al modificar la vista de esta manera, es muy probable que algunos elementos del mapa queden fuera de la pantalla de visualización, por lo que es necesario proveer al usuario de mecanismos para desplazarse por el area del mapa (panning). Es decir, cambiar la porción del mapa que se visualiza en un momento en particular, al mover el mapa hacia arriba, abajo, izquierda o derecha. Todo el conjunto de herramientas con este tipo de fin constituyen los medios de navegación por el mapa.

Se ha hecho uso de un visor SVG para presentar el mapa con extensión *.svg, en la pantalla HTML. Este visor incluye herramientas prediseñadas utiles para la navegación en ilustraciones sencillas. Ejemplo de ellas son:

- Aumentar / Disminuir Zoom.
- Desplazamiento (panning).

Sin embargo, su funcionamiento es predefinido, y sus capacidades cumplen inadecuadamente las exigencias de navegación a traves de un mapa.

Para ello, en la aplicación se han diseñado, creado e incluido los siguientes recursos de navegación:

- Barra de Zoom.
- Control de desplazamiento.
- Mapa de orientación.
- Visualización de coordenadas.

- Navegación por combo.
- Selección de obras a visualizar.

6.3.1 Barra de Zoom.

Permite al usuario realizar acercamientos o alejamientos de la vista del mapa. La barra es un elemento diseñado e implementado en SVG, que hace uso de funciones definidas en lenguaje Javascript, para cambiar sus propias características así como las del mapa.

Está constituida de varios objetos rectangulares, cada uno asociado a uno de los siguientes niveles: 25%, 50%, 100%, 200%, 400%, 600%, 800%, 100%. Al hacer clic sobre uno de ellos se modifica la vista del mapa al nivel asociado al elemento seleccionado.

La barra modifica un método SVG llamado Viewport, que almacena la posición (x,y) del mapa, así como su ancho y alto.

Cuando se selecciona un nuevo nivel de visualización, se obtienen los valores actuales del mapa (su actual posición "x,y" y el actual nivel de zoom). Al contar con esos valores se aplica la lógica matemática que recalcula los valores que se aplicaran al mapa para cumplir con la selección del usuario. Estos datos son asignados al método SVG Viewport, y se redibuja el mapa en pantalla.

Algunos aspectos importantes que se deben tomar en cuenta en la lógica matemática del zoom son:

El nivel actual del zoom: Los saltos entre los niveles de zoom modifican la precisión del mapa. Un mapa en un alto nivel de zoom (1000%), proporciona una mayor precisión. Como puede apreciarse, esto tiene una implicación en el manejo de coordenadas, las cuales deben modificarse de acuerdo al nivel de zoom actual y al nivel de zoom al que se desea llegar.

La posición actual del mapa: Al realizar la acción de cambiar el zoom (seleccionando uno de los niveles de la barra de zoom), se modifica la porción del mapa que se visualiza, pues el mapa aumenta o reduce su tamaño, mientras que la pantalla de visualización sigue igual. Este cambio puede desorientar al usuario al reubicar el mapa en una región del mapa no deseada. Por ello, la lógica matemática desarrollada e implementada con las funciones Javascript y los métodos SVG, reubica el mapa en el punto central que el usuario visualiza en el mapa.

6.3.2 Control de desplazamiento.

Permite al usuario desplazarse por el mapa, al mover la vista del mismo en una de las siguientes cuatro direcciones a la vez: izquierda, derecha, arriba o abajo. Este control es un elemento diseñado e implementado en SVG, que utiliza funciones definidas en lenguaje Javascript, para cambiar sus características y las del mapa.

Esta constituido por 13 elementos, organizados de la siguiente manera:

Dirección del desplazamiento	Cantidad de elementos	Incremento en el desplazamiento	Descripción
Izquierda	3	10, 30, 100 unidades	Desplaza el mapa hacia la izquierda, el numero de unidades asociado al elemento.
Derecha	3	10, 30, 100 unidades	Desplaza el mapa hacia la derecha, el numero de unidades asociado al elemento.
Arriba	3	10, 30, 100 unidades	Desplaza el mapa hacia arriba, el numero de unidades asociado al

			elemento.
Abajo	3	10, 30, 100 unidades	Desplaza el mapa hacia abajo, el numero de unidades asociado al elemento.
Centro	1	---	Desplaza el mapa hacia su centro.

Tabla 6.1. Opciones del control de desplazamiento

El control modifica el método SVG denominado Viewport, que almacena la posición (x,y) del mapa.

Cuando el usuario hace clic sobre alguno de los elementos del control de desplazamiento, se realizan las siguientes tareas:

- En primer lugar, se asocia el elemento a un nivel de desplazamiento (10, 30 o 100 unidades, o se calcula en el caso del elemento central).
- Se obtiene la posición actual.
- Se modifica esa variable en base a la dirección y cantidad de unidades a desplazar.
- Se reasignan los valores al método SVG Viewport.
- Se redibuja el mapa.

En todo el proceso hay que tener muy en cuenta el nivel de zoom, dado que este factor modifica la magnitud de las coordenadas.

6.3.3 Mapa de Orientación

Permite al usuario mantener una referencia constante de su actual ubicación en el mapa. Este mapa es una versión altamente reducida del mapa principal. Por lo tanto, su propósito NO es el de mostrar obras o elementos del mapa (calles, manzanas, puntos de referencia, etc.), sino el de mantener actualizada la sección del mapa que

actualmente se visualiza en pantalla. Un cuadro transparente que varía de tamaño y posición, reacciona de acuerdo a los cambios que el usuario realice mediante la barra de zoom y el control de desplazamiento. Si se duplica la vista del mapa, el cuadro de orientación reduce su tamaño a la mitad para acoplarse a la sección vista. Ello permite reflejar en todo instante, la porción del mapa que se muestra en pantalla.

Este mapa adhiere la funcionalidad de desplazar el mapa hacia un punto específico. Al hacer clic sobre el mapa de orientación, se obtienen las coordenadas del punto seleccionado, y se transporta el mapa principal hacia esa ubicación, manteniendo el nivel de zoom con el que el usuario trabaja en ese instante.

A nivel funcional, el mapa es un grupo de elementos SVG, que contiene las formas básicas del mapa: contorno de municipios y cantones, fondos, etc. Un cuadro transparente ubicado sobre éste, representa la pantalla de visualización. De esa forma, cuando el usuario duplica la vista del mapa, el cuadro transparente reduce su tamaño a la mitad, pues la pantalla de visualización contiene ahora la mitad del área del mapa, afectando los siguientes atributos de los elementos SVG del mapa de orientación:

- X: La posición horizontal del cuadro sobre el mapa de orientación.
- Y: La posición vertical del cuadro sobre el mapa de orientación.
- width: El ancho del cuadro sobre el mapa de orientación.
- height: El alto del cuadro sobre el mapa de orientación.

6.3.4 Visualización de coordenadas.

Muestra al usuario las coordenadas del mapa, mediante la ubicación del puntero del ratón sobre la pantalla de visualización. Esto permite conocer las coordenadas de los diversos elementos del mapa al colocar sobre éste el puntero. Estos valores son mostrados en un área de texto del panel de navegación.

Se hace uso de lenguaje Javascript para acceder y modificar métodos SVG que contienen la información necesaria. Esta se actualiza constantemente mientras el

usuario se desplaza con el puntero a lo largo del mapa. Cuando éste hace clic sobre el mismo, se capturan esas coordenadas para realizar alguna tarea relacionada (ubicar o reubicar una obra, por ejemplo).

Contribuye también a mantener una referencia constante de la posición actual del mapa.

6.3.5 Navegación por combo.

Permite al usuario desplazarse en el mapa hacia una ubicación predefinida en un combo de selección. De esa forma, puede llevarse la vista del mapa hacia un área en específico (cantón, municipio o departamento).

Se convierte en una herramienta útil al predefinir destinos de visualización. El usuario selecciona desde un combo una región a la que desea dirigirse, posiblemente una que no es visible actualmente en el mapa en pantalla. El sistema determina la ubicación de ese lugar y reubica el mapa en pantalla para mostrar ese destino. Así, se puede escoger reubicar el mapa en un cantón, municipio o departamento determinado.

Los valores mostrados en el combo de selección no son estáticos, sin embargo, el usuario no puede modificar esos valores directamente. Los destinos son dinámicamente generados al cargar el mapa, es decir los valores presentados en el combo son obtenidos de la base de datos. Así mediante un servlet se incluyen a la lista los nombres de los posibles destinos existentes en el mapa.

El funcionamiento del combo de selección se basa en un script Java que, de acuerdo a la selección de destino del usuario, modifica los métodos y variables de visualización del mapa, y lo reubica en el destino seleccionado.

6.3.6 Selección de obras a visualizar.

Como parte del proceso para especializar el mapa para el usuario, de acuerdo a su nivel de acceso y autoridad, se buscan las obras a las cuales éste tiene acceso, organizadas por proyectos de acuerdo a la institución a la que pertenecen o que las ejecuta, y se procede a mostrarlas en pantalla, en un panel separado.

Se brinda al usuario la opción de visualizar u ocultar los grupo de obras relacionadas (todas las escuelas, por ejemplo). Para ello, dentro del panel de navegación, se ha colocado una acceso para visualizar/ocultar el panel de selección de obras, donde se asocia cada capa o clasificación de obras a un cuadro de selección ('checkbox') con el que el usuario puede decidir si visualizar u ocultar dicha capa de obras.

Mediante un servlet se leen las obras almacenadas en la Base de Datos, y de acuerdo al nivel de acceso de quien este conectado, se obtienen de ella las obras pertinentes. A continuación, haciendo uso nuevamente de servlets, se escriben las obras en un archivo texto/SVG, en una posición determinada, identificada como:

```
<g id="obras">  
codigo_SVG_de_las_obras  
...  
</g>
```

Así, el código SVG correspondiente a las obras adquiridas, queda agrupado e identificado de acuerdo a institución. Se muestran así las obras al usuario.

Mientras el usuario consulta el mapa puede ver o no, las capas de obra que se le permiten, mediante los cuadros de selección , los cuales están gráfica y funcionalmente definidos en SVG. Así, cuando el usuario hace 'click' sobre uno de estos cuadros, el grupo de obras relacionado se oculta o visualiza dependiendo del estado del cuadro (ó). Ello es posible gracias las funciones Javascript que se

han diseñado para manipular las propiedades de visibilidad de los elementos SVG (cuadros de selección y grupos de obra).

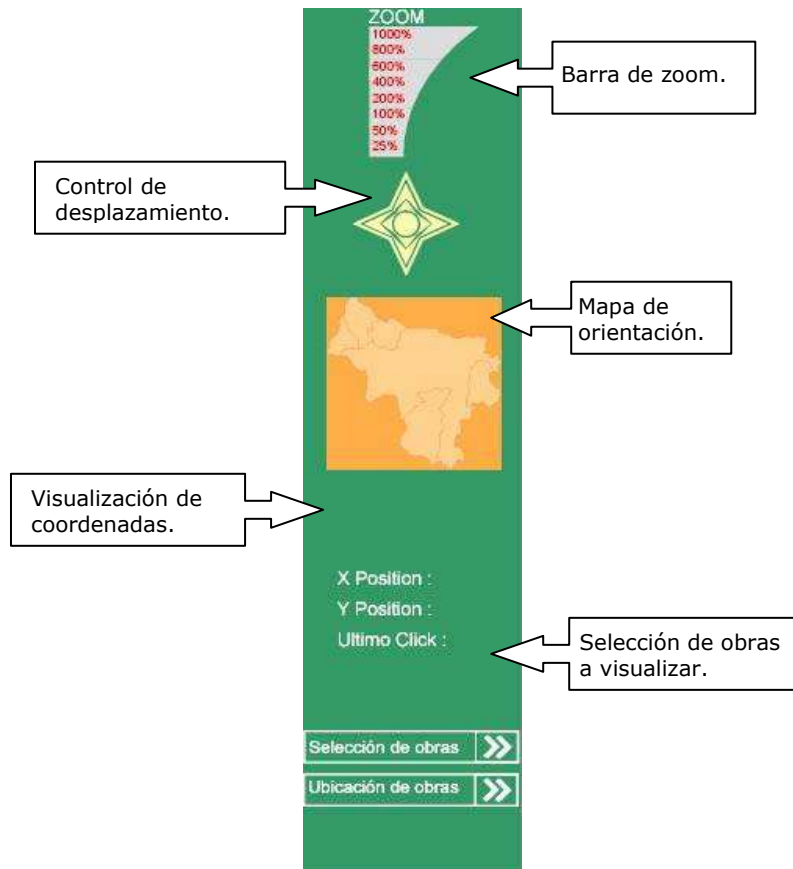


Figura 6-4. Panel de navegación

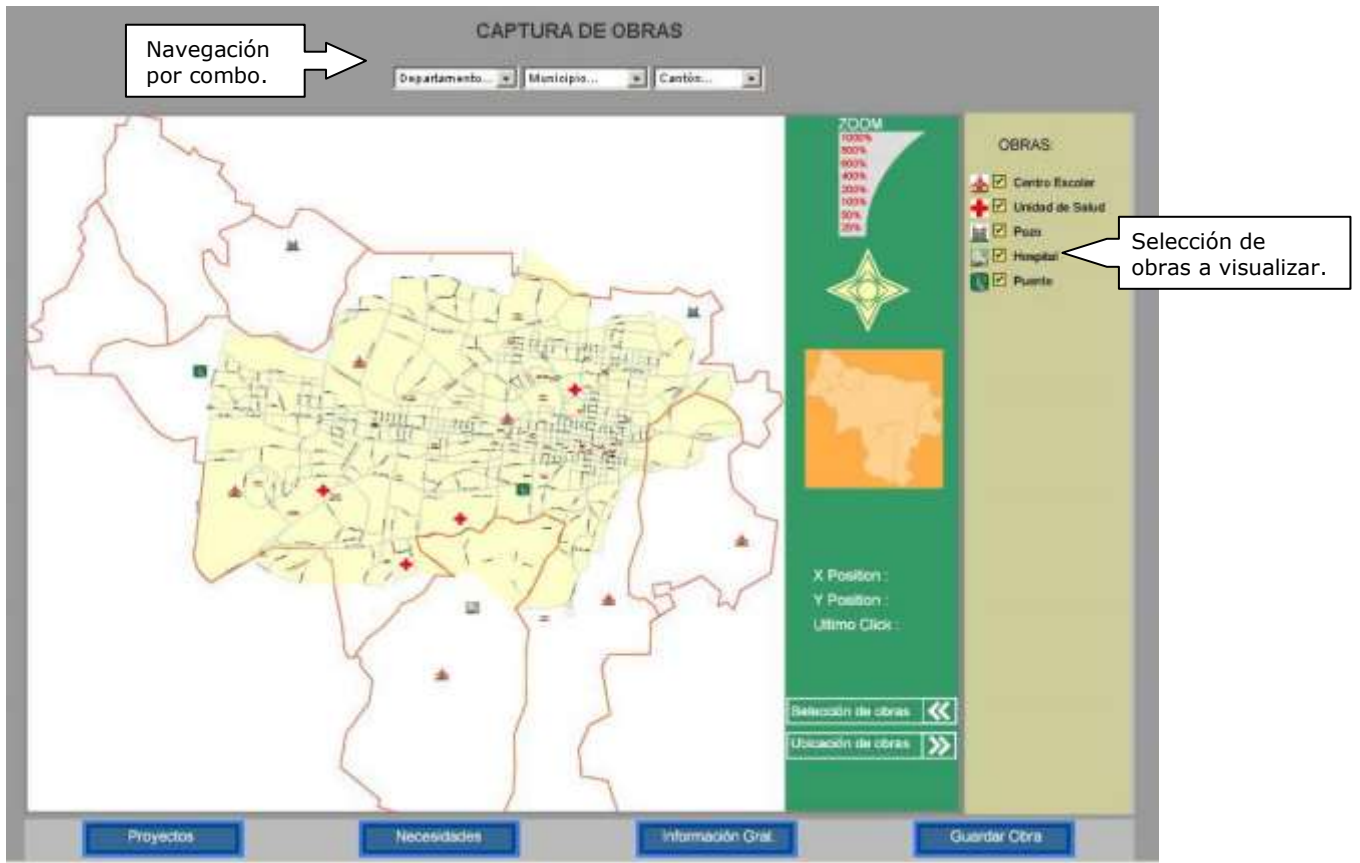


Figura 6-5. Mapa con resultados y disponible para navegación.

7.1 Inversión: Hardware y Software:

1 Servidor	\$ 2,000.00
1 PC	\$ 1,000.00
ServletExec	\$ 695.00
Jturbo Single Machine	\$ 595.00
JTurbo Single Server	\$ 725.00
Windows 2000 Server	\$ 319.00
SQL Server 2000	\$ 667.00

Capacitación

8 hrs. - \$25 por persona	\$ 500.00
---------------------------	-----------

Total \$ 6,501.00

7.2 Costos de Mantenimiento mensual

Honorarios Administrador	\$ 1,000.00
Java Web Hosting	\$ 30.00

7.3 Especificaciones del servidor

- 1 INTEL CAJA P4 3.06GH FSB 533MHZ SK478
- 1 MB BIOSTAR P4 U8668D S/V/L/M ATX
- 1 FLOPPY DISK DRIVE 1.44MB BEIGE
- 1 DISCO DURO CHEETAH LW 10KRPM 36 GB SCSI
- 1 T/CONT ADAPTEC 29160 KIT PCI SCSI
- 1 KVR 512MB PC2700 DDR PC333
- 1 CASE GENERICO P4 MID TWR T313 ATX
- 1 TECLADO Y MOUSE INTERNET MS UND OEM SPA
- 1 U/CD ROM LG OEM 52X IDE INT

El desarrollo de un proyecto de Sistemas de Información comprende varios componentes y pasos llevados a cabo durante la etapa del análisis, el cual ayuda a traducir las necesidades del cliente en un modelo de sistema que vuelva más eficiente los procesos de la institución y que a demás responda a los requerimientos del cliente.

Las salidas en un sistema de información son el resultado de todo el análisis, diseño y desarrollo del mismo. El sistema de información geográfica desarrollado presenta salidas visuales que en conjunto con otras herramientas contribuyen al análisis en la toma de decisiones.

Conjuntamente se ha obtenido un conocimiento profundo sobre la tecnología SVG utilizada para el manejo del mapa, y del alcance aplicativo que este puede tener; como la dinamicidad, adaptación y personalización del mismo, logrando así desarrollar una aplicación que permite interacción amigable y de fácil manejo con el usuario de conocimientos básicos de navegación en Internet, que al mismo tiempo le proporciona una herramienta de trabajo, con resultados dinámicos y de ayuda visual.

Finalmente se generó una lógica de desarrollo que permite reducir el tiempo de mantenimiento de la funcionalidad de la aplicación en cuanto a la parte de desarrollo, y permitirle al administrador realizar mantenimientos al sistema reduciendo la interacción directa con la base de datos, al generar un mantenimiento lo más general posible a aquellas pantallas que no requieren un trato especial, si no que poseen características comunes.

- Diseñar y desarrollar un mecanismo por medio del cual las personas interesadas en obtener un usuario autorizado para el uso del sistema pueda solicitarlo al administrador.

Debido a que actualmente el sistema no cuenta con un mecanismo de comunicación con la administración del sistema o con otras partes que se puedan involucrar en la aplicación, podría crearse una interfaz sencilla de posteo de mensajes; de tal forma que permita dirigirlos a la administración (los cuales solo serán vistos por las personas encargadas de esta sección) o quedar visibles de forma que los demás usuarios puedan leerlos (estos podrían ser como comunicados generales), esto depende de la funcionalidad que se le quiera dar. Este mecanismo podría estar basado en un concepto de navegación abierta a cualquier usuario en la cual se le brinde la opción de solicitar un usuario para poder tener acceso a las demás opciones del sistema.

- Depurar la aplicación para una óptima visualización en cualquier navegador.

Una de las limitantes planteadas es la optimización de la funcionalidad para el Internet Explorer; para que la funcionalidad se extienda a otros navegadores, debe hacerse la investigación correspondiente de la equivalencia de propiedades y atributos de los cuales se hacen uso, tomando especial énfasis en la sección de manipulación del objeto svg que se genera.

- Diseñar y desarrollar mecanismo para la creación de gráficos.

Los reportes son generados únicamente en formato tabla, se recomienda un agregado para generar los resultados de forma gráfica en formato de barra, pastel, etc., que complementen la información de forma visual para proporcionar más apoyo en el análisis de resultados.

- Aumentar las capacidades gráficas del mapa con la adición de un sistema para la introducción de obras no puntuales (ej: sistema de carreteras, energía eléctrica,

sistema de agua potable, etc.), así como los mecanismos necesarios para su administración (ingreso de datos, entradas – salidas de información, reportes).

- Diseñar y desarrollar reportes que permitan un control financiero por parte de los donantes.

Los datos de programación y seguimiento tanto financiero como físico han sido incluidos en el modelo de datos, los cuales se pueden utilizar para obtener reportes consolidados y desagregados de los avances que ha tenido un proyecto y/o proyectos en particular y los cuales desee hacerseles un seguimiento y evaluación por parte de las entidades que lo financian.

- Ampliar la parte de seguimiento de proyectos para poder incorporar procesos como licitación, adjudicación, etc. La parte de las etapas ha sido elaborada de manera global, incluyendo grandes grupos que abarca sub-etapas.
- Diseñar un mecanismo que permita mostrar iconos de las obras sin restricción de cantidad. El sistema permite mostrar hasta 25 iconos de obras.

Para que el crecimiento de obras y que estas no queden restringidas a un número finito de iconos a mostrar en los paneles del mapa que están contenidos dentro del área del objeto svg, podrían generarse de la misma forma dinámica en formato html; siempre en la misma pantalla para facilidad del usuario, teniendo en cuenta las consideraciones del caso para que continúe con la misma funcionalidad que hasta el momento presentan.

- Ampliar la cobertura del mapa, adicionando los mapas de los cantones, municipios y departamentos restantes del país.
- Adherir funciones de interactividad en un ambiente de tres dimensiones.

<http://200.46.127.89/website/proyectos/viewer.htm?Title=Proyectos%20en%20Ejecuci%F3n%20-%20FIS>

<http://www.mef.gob.pa/Programacion%20de%20Inversiones/Contenido/area%20de%20proyectos%20de%20desarrollo.doc>

<http://www.esri.com>

<http://www.adobe.com>

http://www.htmlpoint.com/javascript/corso/js_02.htm

<http://www.ilog.com/products/jviews/conferences/XMLEurope2001.pdf>

<http://www.protocol7.com/svg-wiki/ow.asp?ThePath>

http://www.zvon.org/HowTo/Output/category_SVG.html

<http://www.htmlweb.net/disenosvg/>

http://www.mideplan.go.cr/sinades/Proyecto_SINADES/sostenibilidad/armonizacion/index-10.html

http://www.htmlweb.net/disenosvg/svg_1.html