

Universidad Autónoma Metropolitana

Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

**Reporte Final del Proyecto de Integración
Licenciatura en Ingeniería en Computación**

Modalidad de Estancia Profesional

Sistema web para gestión de zonas en la distribución de gas L.P.

Elaboró:

Alejandro Castro Romero

2112000470

Beatriz Adriana González Beltrán

Carmen Daniel García Domínguez

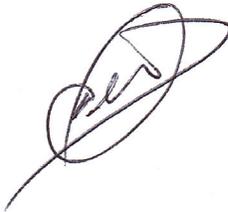
15-P

02/09/2015

Yo, Beatriz Adriana González Beltrán, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Yo, Carmen Daniel García Domínguez, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Yo, Alejandro Castro Romero, doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.

Alejandro Castro R.

Resumen

La industria gasera tiene como objetivo suministrar gas alrededor del país. México está dividido por áreas de distribución y las empresas de distribución de gas tienen asignadas una o varias áreas.

Gas de Pluviosilla S.A. de C.V. requiere tener control sobre las áreas que le han asignado. Estas áreas están divididas en zonas que pueden tener modificaciones debido a factores como: enfermedades, despidos o permisos de los operadores; bloqueos de caminos, etc.

La problemática que se presenta es la que se menciona a continuación. Cuando el encargado de planeación modifica una zona debe reimprimir el mapa con la nueva distribución y este mapa debe ser entregado al personal encargado del área de ventas. El personal de ventas se basa en el mapa para saber qué zona está cubriendo un operador, así en caso de recibir una llamada solicitando el suministro de gas el personal de ventas sabrá que operador es el adecuado para realizar dicho suministro. Este proceso puede automatizarse de manera que los actores involucrados conozcan las zonas en el momento en que hayan sido modificadas.

Para automatizar el proceso, la propuesta de proyecto de integración consistió en el desarrollo de un sistema para gestionar zonas. El encargado de la planificación de las zonas es el responsable de actualizarlas a través del mapa digital y de dibujar sobre este mapa las zonas.

El presente documento muestra la construcción de un sistema para la gestión de rutas en la distribución de Gas L.P. En este documento se incluyen los antecedentes, justificación, marco teórico, desarrollo del sistema así como los resultados incluyendo su análisis y perspectiva a futuro.

Índice

1	Introducción.....	7
2	Antecedentes.....	8
2.1	Referencias externas.....	8
2.2	Referencias internas.....	8
3	Justificación.....	9
4	Objetivos.....	10
4.1	Objetivo general.....	10
4.2	Objetivos particulares.....	10
5	Marco teórico.....	10
5.1	Conceptos.....	11
5.2	Servicios web.....	11
5.2.1	Representational State Transfer.....	11
5.3	API Google Maps.....	13
6	Desarrollo del proyecto.....	14
6.1	Metodología empleada para el desarrollo del proyecto.....	14
6.2	Diseño del sistema.....	14
6.2.1	Diagramas de caso de uso.....	15
6.2.1.1	GestionPluviosilla (SIGEZ).....	15
6.2.1.2	MapsZone.....	15
6.2.2	Casos de uso de texto.....	16
6.2.2.1	GestionPluviosilla.....	16
6.2.2.1.1	Agregar Zona.....	16
6.2.2.1.2	Modificar Zona.....	17
6.2.2.1.3	Eliminar Zona.....	18

6.2.2.2	MapsZone.....	19
6.2.2.2.1	Agregar Zona	19
6.2.2.2.2	Modificar Zona	20
6.2.2.2.3	Eliminar Zona	21
6.2.3	Diagrama de clases	22
6.2.4	Arquitectura del sistema	23
6.2.5	Estructura de la base de datos.....	26
6.2.5.1	GestionPluviosilla	26
6.2.5.2	MapsZone.....	26
6.3	Uso del sistema.....	27
6.3.1	Agregar Zona	27
6.3.2	Modificar Zona	30
6.3.3	Eliminar Zona	32
7	Hardware y software necesario.....	34
7.1	Tecnología usada para el desarrollo	34
7.1.1	Hardware	34
7.1.2	Software.....	34
7.2	Tecnología y puesta en marcha de la aplicación	35
7.2.1	Hardware	35
7.2.2	Software.....	35
8	Resultados.....	35
9	Análisis y discusión de resultados	36
10	Conclusiones.....	37
11	Perspectivas del proyecto	38
12	Referencias bibliográficas	38

Apéndice A. Entregable: Listado del API del código fuente desarrollada	39
Apéndice B. Entregable: Memoria de diseño	39
Apéndice C. Entregable: Manual técnico.....	39
Apéndice D. Entregable: Manual de usuario.....	39

1 Introducción

La industria gasera tiene como objetivo suministrar gas alrededor del país. México está dividido por áreas de distribución y las empresas de distribución de gas tienen asignadas una o varias áreas.

Gas de Pluviosilla S.A. de C.V. requiere tener control sobre las áreas que le han asignado. Estas áreas están divididas en zonas que pueden tener modificaciones debido a factores como: enfermedades, despidos o permisos de los operadores; bloqueos de caminos, etc.

La problemática que se presenta es la que se menciona a continuación. Cuando el encargado de planeación modifica una zona debe reimprimir el mapa con la nueva distribución y este mapa debe ser entregado al personal encargado del área de ventas. El personal de ventas se basa en el mapa para saber qué zona está cubriendo un operador, así en caso de recibir una llamada solicitando el suministro de gas el personal de ventas sabrá que operador es el adecuado para realizar dicho suministro. Este proceso puede automatizarse de manera que los actores involucrados conozcan las zonas en el momento en que hayan sido modificadas.

Para automatizar el proceso, la propuesta de proyecto de integración consistió en el desarrollo de un sistema para gestionar zonas. El encargado de la planificación de las zonas es el responsable de actualizarlas a través del mapa digital y de dibujar sobre este mapa las zonas.

El presente documento muestra la construcción de un sistema para la gestión de rutas en la distribución de Gas L.P. En este documento se incluyen los antecedentes, justificación, marco teórico, desarrollo del sistema así como los resultados incluyendo su análisis y perspectiva a futuro.

2 Antecedentes

Esta sección contiene referencias a trabajos tanto internos como externos, que tienen características similares al trabajo propuesto.

2.1 Referencias externas

A method for building thematic map of GIS based on Google Maps API [1].

Este trabajo propone un método para crear mapas temáticos utilizando el API de Google Maps. El método permite generar un archivo para las coordenadas de un polígono, convertirlo en un archivo XML y generar el mapa temático. Se propone una aplicación para mostrar en un mapa la incidencia y severidad con que fueron atacadas 18 provincias de Beijing en 2004 por el virus de la influenza aviar. Este artículo sirvió de base para implementar el sistema propuesto.

Disaster management system based on GIS web services [2].

Este trabajo propone crear un sistema de gestión de desastres basada en servicios Web. Este sistema consta de tres importantes servicios: policía, bomberos y médico. Estos se desarrollaron usando web services y AJAX para cargar de manera más eficiente los datos espaciales en el navegador. Este trabajo tiene en común con este proyecto el uso de servicios Web.

2.2 Referencias internas

Gestión de calificaciones de cursos mediante servicios Web [3].

Este trabajo propone desarrollar un sistema de gestión de calificaciones de cursos mediante servicios Web. En este sistema se manejan servicios Web basados en REST, los cuales se invocan desde una aplicación cliente. Este trabajo tiene en común con este proyecto el uso de REST para la construcción de servicios Web, ya que los servicios de Google Maps están basados en REST.

Calendarios compartidos y eventos grupales (cceg): aplicación de escritorio para la gestión de eventos individuales y grupales en calendarios de Google [4].

Este proyecto tiene como objetivo crear una aplicación que permita la gestión de eventos individuales y grupales en calendarios de Google. En este proyecto se hace uso del API de

Google Calendar, para ello indica las bibliotecas de Google que se utilizaron. Este trabajo tiene en común con este proyecto el uso de una API de Google.

Aplicación de tecnologías de cómputo móvil y sistemas de información para la mejora en el proceso de negocio dentro de un restaurante [5].

Este proyecto plantea un sistema para la gestión de carta digital para restaurantes. En este proyecto se propone crear servicios web encargados de la autenticación y autorización. Este trabajo tiene en común con este proyecto la creación y uso de servicios web.

3 Justificación

Este proyecto fue solicitado por Vel A Gas de Pluviosilla S.A de C.V. ya que permitirá automatizar el proceso de gestión de las zonas. Actualmente, no existe una aplicación comercial que permita realizar esta gestión.

El proyecto permite al encargado de planeación editar las zonas sobre un mapa digital, la edición permite dibujar las zonas sobre el mapa. Con el uso de este sistema se busca facilitar la tarea del encargado de diseñar las zonas de ventas, y que el personal encargado de ventas pueda visualizar los mapas cuando así lo requiera.

4 Objetivos

4.1 Objetivo general

Desarrollar un sistema web que permita la gestión de zonas de reparto de gas L.P.

4.2 Objetivos particulares

- Diseñar e implementar un módulo para poder gestionar la comunicación con Google Maps.
- Diseñar e implementar un módulo para poder dibujar sobre un mapa.
- Diseñar e implementar un módulo para almacenar y recuperar las coordenadas de un mapa.
- Diseñar e implementar un módulo para poder visualizar las zonas definidas en un mapa.
- Integrar los módulos en una aplicación web.
- Instalar el sistema en el servidor que destine la empresa.
- Realizar pruebas en el ambiente de producción.

5 Marco teórico

Esta sección describe los principales temas involucrados en la concepción del sistema. Con esta sección se busca que el lector tenga un contexto más claro sobre la solución que se elaboró.

Esta sección se divide en tres temas, el primero tiene que ver con los conceptos del negocio, los dos temas restantes se centraran más en la tecnología que se empleó para el desarrollo del sistema. Para que el texto sea más claro cada que se haga una referencia a Vel A Gas de Pluviosilla se hablará de esta como la “**Gasera**”.

5.1 Conceptos

Zona de distribución

Se considera una zona de distribución al área designada para que un operador lleve a cabo la distribución de Gas L.P. En particular, la gasera maneja el concepto como rutas de distribución, es por ello que cualquier referencia a zona o ruta es indistinto.

Tipos de distribución

La gasera actualmente tiene dos tipos de distribución, estacionario y portátil. La distribución de gas estacionario es la que se hace mediante pipas, estas pipas se encargan de suministrar gas directamente en los tanques estacionarios de los clientes, los tanques estacionarios son aquellos que permanecen inamovibles. El otro tipo de distribución es portátil, en esta modalidad la gasera le deja al cliente un cilindro con gas, y el cliente le entrega a la gasera un cilindro vacío y el importe del gas.

Debido a que se necesitó distinguir entre las rutas de distribución de gas estacionario y gas portátil es que surgió el término capas.

Capa

Una capa es aquella que agrupa a un conjunto de zonas. Particularmente la gasera maneja dos capas, estacionario y portátil.

5.2 Servicios web

Actualmente existen diferentes tipos de servicios web. En particular, para la elaboración de este proyecto se ocuparon servicios web basados en RESTful.

5.2.1 Representational State Transfer

Según la definición puramente teórica, REST es un estilo de arquitectura que abstrae los elementos de dicha arquitectura dentro de un sistema hipermedia distribuido [6]. En términos más coloquiales, REST es un conjunto de principios, o maneras de hacer ciertas cosas, define las reglas que distintos componentes deben seguir. Debido a esas características es que se utiliza el protocolo HTTP.

Bajo esta definición toda aplicación web bajo el protocolo HTTP es una aplicación REST. Pero esto no quiere decir que todas las aplicaciones web sean servicios web RESTful, para tal fin se deben de cumplir ciertos requisitos.

Reglas de la arquitectura REST:

- **Arquitectura cliente-servidor:** Debe de tener una separación entre el cliente y el servidor. Se debe de tener independencia, lo cual nos da una gran flexibilidad.
- **Stateless:** Esto nos habla de que el servidor no debe de almacenar datos del cliente para mantener un estado del mismo.
- **Uso de memoria caché:** Implica que el servidor tome las peticiones del cliente y para ello se debe de definir un método que permita tomarlas y controlar las respuestas.
- **Sistema por capas:** El cliente no debe de saber cómo se tramita su información.
- **Interfaz uniforme:** Esto nos habla de que al servidor no debe importarle quien sea el cliente, sino que la petición que se le haga sea correcta, es por ello que la interfaz debe de estar bien definida.

Principios básicos:

Para que podamos considerar a nuestra aplicación REST como buena. Tendremos que tomar en cuenta cuatro principios básicos.

- **Identificación de recursos:** Esto se lleva mediante la identificación de URIs (Uniform Resource Identifier).
- **Recursos y representaciones:** Ya que tenemos la identificación, REST define como podemos interactuar con la representación del mismo. La aplicación puede mandar la petición en el formato que se quiera, siempre y cuando el contenido contenga la información necesaria para poder operar con el mismo.
- **Mensajes autodescriptivos:** Cuando se hace una petición a un servidor, este deberá de devolver una respuesta que permita entender lo que sucedió con nuestra operación. HTTP implementa esto mediante las cabeceras

- **Hypermedia as the Engine of Application State (HATEOAS).** Los servicios web de incluir en las respuestas del servidor toda aquella información que necesita el cliente para seguir operando con este servicio web. En otras palabras, el cliente no tiene porqué saber que cuando obtenemos, por ejemplo, un objeto cualquiera, tenemos además las opciones de modificarlo, o eliminarlo. El servidor debe enlazar a estas operaciones en la respuesta a dicha petición.

Servicios web RESTful

Relacionando los conceptos anteriores con el protocolo HTTP. Podemos decir que un servicio web RESTful hace referencia a un servicio web que implementa la arquitectura REST.

Un servicio web RESTful debe contener lo siguiente:

- URI del recurso
- El tipo de la representación de dicho recurso, comúnmente JSON, XML o TXT.
- Operaciones soportadas. HTTP define algunas como GET, PUT, POST, etc.
- Hipervínculos. La respuesta puede incluir hipervínculos hacia otras acciones que se puedan realizar sobre los recursos. Normalmente se incluyen en el mismo contenido de la respuesta, así por ejemplo, si nuestra respuesta es un objeto en JSON, podemos añadir una propiedad más con los hipervínculos a las acciones que admite el objeto.

5.3 API Google Maps

Actualmente la compañía Google cuenta con un servidor de aplicaciones de mapas en la web. En esta se ofrecen imágenes de satélite, rutas, etc.

El API Google Maps, nos permite interactuar con sus mapas mediante el uso de código en JavaScript. Lo que se hace es definir un mapa y sobre este con métodos ya definidos por el API, se pueden realizar diversas acciones sobre el mismo.

El API se podría dividir en cuatro conceptos:

- **Creación del mapa**

La creación del mapa es la primera parte en esta se define las características principales del mismo, tamaño, tipo de mapa, estilos, etc.

- **Dibujado sobre el mapa**

En esta se describe todo lo relacionado con el dibujado sobre el mapa.

- **Visualización de datos**

Esta parte permite ver detalles en el mapa como tráfico, tránsito, mapas de calor, camino en bicicleta, etc.

- **Servicios**

Esta sección nos permite hacer uso de servicios que tiene el API tales como: direcciones, elevación, vista de calle, geocodificación, etc. En esta sección se describe lo que el sistema debe hacer.

6 Desarrollo del proyecto

Esta sección describe la metodología empleada, el diseño y uso del sistema, los componentes de hardware y software empleados.

6.1 Metodología empleada para el desarrollo del proyecto

Este proyecto empleó el proceso unificado como metodología de base. Esta metodología tiene las siguientes características: es iterativo e incremental, dirigido por caso de uso, centrado en arquitectura y enfocado en los riesgos. Cuenta con cuatro fases, inicio, elaboración, construcción y transición.

6.2 Diseño del sistema

Esta sección contiene los diagramas principales de caso de uso y casos de uso de texto, tanto para el servicio web como para la aplicación. También contiene la arquitectura del sistema en conjunto y la estructura de la base de datos.

6.2.1 Diagramas de caso de uso

En esta sección se tiene los diagramas generales de caso de uso, pero si se quiere ver más a detalle se puede consultar el Apéndice B, en él se describen a detalle los diagramas de casos de uso.

6.2.1.1 GestionPluviosilla (SIGEZ)

La Figura 1 describe el caso de uso general del SIGUEZ.

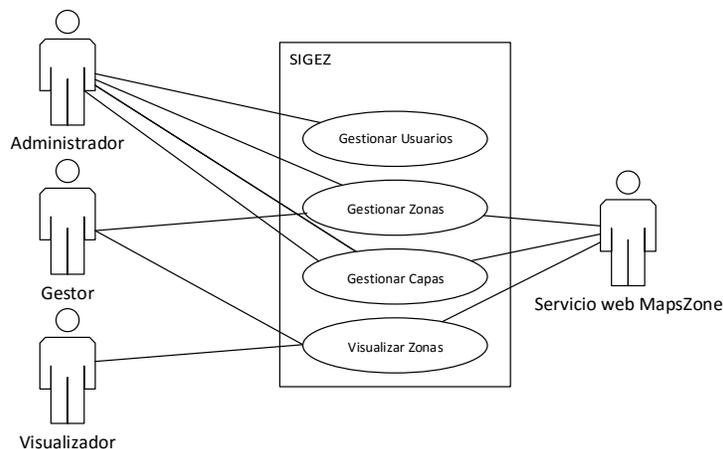


Figura 1. Diagrama de casos de uso general de SIGUEZ.

6.2.1.2 MapsZone

La Figura 2 describe el caso de uso general del servicio web MapsZone.

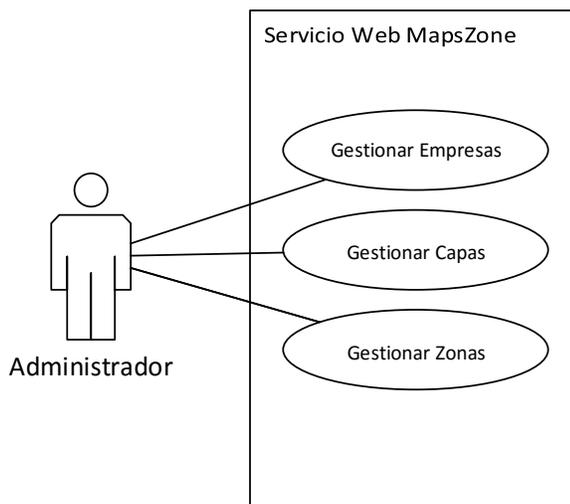


Figura 2. Diagrama de casos de uso general de MapsZone.

6.2.2 Casos de uso de texto

En esta sección se tiene algunos casos de uso de texto, pero si se quiere ver más a detalle se puede consultar el Apéndice B, en él se describen a detalle los casos de uso de texto.

6.2.2.1 GestionPluviosilla

6.2.2.1.1 Agregar Zona

Descripción

Este caso de uso describe lo que pasa en el sistema cuando se agrega una zona.

Actores

Actor	Descripción
Gestor	Este usuario es el encargado de agregar zonas.
Administrador	Este usuario no es el encargado de agregar zonas, pero puede hacerlo.

Precondiciones

El gestor ya inicio sesión en el sistema.

Postcondiciones

El sistema contará con una nueva zona.

Flujo básico

- 1) El gestor elige la opción de zonas.
- 2) El gestor elige la capa en la que quiere agregar la zona.
- 3) El sistema regresa una interfaz para el dibujado de la zona que se agregará y un formulario.
- 4) El gestor dibuja la zona, llena el formulario y lo envía.
- 5) El sistema agrega la zona.
- 6) El sistema refresca la interfaz

Flujos alternativos

- 1) El sistema falla.
 - a) El gestor debe de iniciar sesión nuevamente y verificar si se agregó la zona.
- 2) El gestor agrega la zona con datos incorrectos.
 - a) El gestor modifica la información de la zona.
 - b) El gestor eliminará la zona.

6.2.2.1.2 Modificar Zona

Descripción

Este caso de uso describe lo que pasa en el sistema cuando se modifica una zona.

Actores

Actor	Descripción
Administrador	Este usuario es el encargado de agregar usuarios.
Gestor	Este usuario es el encargado de modificar zonas.

Precondiciones

El gestor ya inicio sesión en el sistema.

Postcondiciones

El sistema tendrá una zona actualizada.

Flujo básico

- 1) El gestor elige la opción de zonas.
- 2) El gestor elige la capa en la que quiere modificar la zona.
- 3) El sistema regresa una interfaz para el modificado de la zona.

- 4) El gestor selecciona la zona.
- 5) El gestor modifica la información de la zona, y lo envía.
- 6) El sistema modifica la zona.
- 7) El sistema refresca la interfaz.

Flujos alternativos

- 1) El sistema falla.
 - a) El gestor debe de iniciar sesión nuevamente y verificar si se agregó la zona.
- 2) El gestor modificó la zona con datos incorrectos.
 - a) El gestor modifica la información de la zona.
 - b) El gestor eliminará la zona.

6.2.2.1.3 Eliminar Zona

Descripción

Este caso de uso describe lo que pasa en el sistema cuando se elimina una zona.

Actores

Actor	Descripción
Gestor	Este usuario es el encargado de eliminar zonas.
Administrador	Este usuario no es el encargado de eliminar zonas, pero puede hacerlo.

Precondiciones

El gestor ya inicio sesión en el sistema.

Postcondiciones

El sistema ya no tendrá la zona que se borró.

Flujo básico

- 1) El gestor elige la opción de zonas.
- 2) El gestor elige la capa en la que quiere eliminar la zona.
- 3) El sistema regresa una interfaz para el eliminado de la zona.
- 4) El gestor selecciona la zona.
- 5) El gestor elige la opción eliminar.
- 6) El sistema elimina la zona.
- 7) El sistema refresca la interfaz.

Flujos alternativos

- 1) El sistema falla.
 - a) El gestor debe de iniciar sesión nuevamente y verificar si se agregó la zona.
- 2) El gestor eliminó la zona incorrecta.
 - a) El gestor deberá agregar la zona.

6.2.2.2 MapsZone

6.2.2.2.1 Agregar Zona

Descripción

Este caso de uso describe lo que pasa en el servicio web cuando se agrega una zona.

Actores

Actor	Descripción
Cliente	Este usuario es el encargado de agregar zonas.

Precondiciones

N/A

Postcondiciones

El servicio web contará con una nueva zona.

Flujo básico

- 1) El cliente envía la petición de agregar capa correctamente formada al servicio web.
- 2) El servicio web recibe la petición.
- 3) El servicio web procesa la petición
- 4) El servicio web regresa la respuesta de la petición.

Flujos alternativos

- 1) El servicio web falla.
 - a) El cliente debe mandar la petición nuevamente.
- 2) El cliente ingresa mal los datos.
 - a) El cliente debe mandar la petición de modificación.

6.2.2.2 Modificar Zona

Descripción

Este caso de uso describe lo que pasa en el servicio web cuando se modifica una zona.

Actores

Actor	Descripción
Cliente	Este usuario es el encargado de modificar zonas.

Precondiciones

N/A

Postcondiciones

El servicio web tendrá una zona actualizada.

Flujo básico

- 1) El cliente envía la petición de modificar zona correctamente formada al servicio web.
- 2) El servicio web recibe la petición.
- 3) El servicio web procesa la petición
- 4) El servicio web regresa la respuesta de la petición.

Flujos alternativos

- 1) El servicio web falla.
 - a) El cliente debe mandar la petición nuevamente.
- 2) El cliente ingresa mal los datos.
 - a) El cliente debe mandar la petición de modificación.

6.2.2.2.3 Eliminar Zona

Descripción

Este caso de uso describe lo que pasa en el servicio web cuando se elimina una zona.

Actores

Actor	Descripción
Cliente	Este usuario es el encargado de eliminar zonas.

Precondiciones

N/A

Postcondiciones

El sistema ya no tendrá la zona que se borró.

Flujo básico

- 1) El cliente envía la petición de eliminar zona correctamente formada al servicio web.

- 2) El servicio web recibe la petición.
- 3) El servicio web procesa la petición
- 4) El servicio web regresa la respuesta de la petición.

Flujos alternativos

- 1) El servicio web falla.
 - a) El cliente debe mandar la petición nuevamente.
- 2) El cliente ingresa mal los datos.
 - a) El cliente debe mandar la petición de eliminar nuevamente.

6.2.3 Diagrama de clases

La figura 3 muestra el diagrama de clases de SIGEZ, este diagrama indica cómo es que las clases interactúan para la inserción de cuentas de usuario.

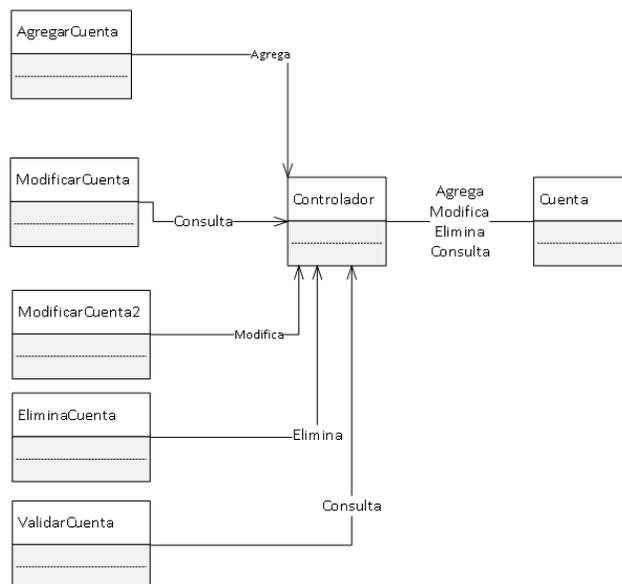


Figura 3. Diagrama de clases de GestionPluviosilla (SIGEZ).

La figura 4 muestra las clases del servidor web, WsResource se encarga de enviar al gestor la información necesaria este a su vez ocupa el controlador para realizar operaciones CRUD (Creación, lectura, actualización y escritura).

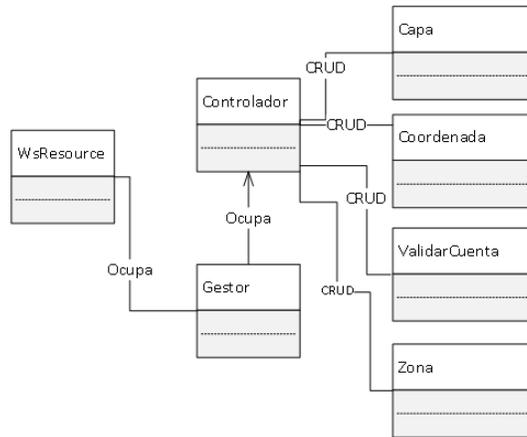


Figura 4. Diagrama de clases de MapsZone.

6.2.4 Arquitectura del sistema

En la figura 5 se tiene una arquitectura a tres capas, este diagrama muestra lo relacionado con la gestión de los usuarios, esta gestión se da únicamente en la aplicación y no en el servicio web.

Cada jsp se comunica con su servlet correspondiente. Los servlets se comunican con el controlador el cual tiene comunicación con la capa de datos.

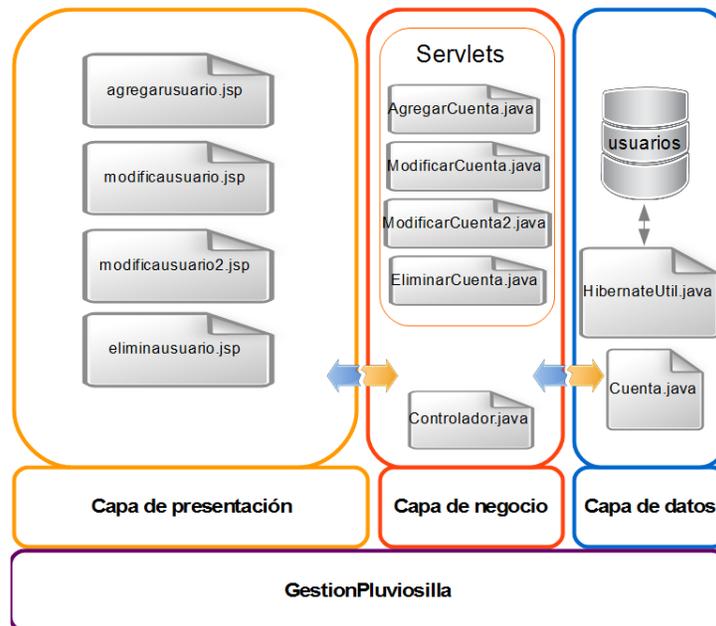


Figura 5. Arquitectura de GestionPluviosilla.

aplicación web. Una vez que se envía una petición el servicio web MapsZone se encargará de responderle a la aplicación. Toda esta comunicación se da en XML.

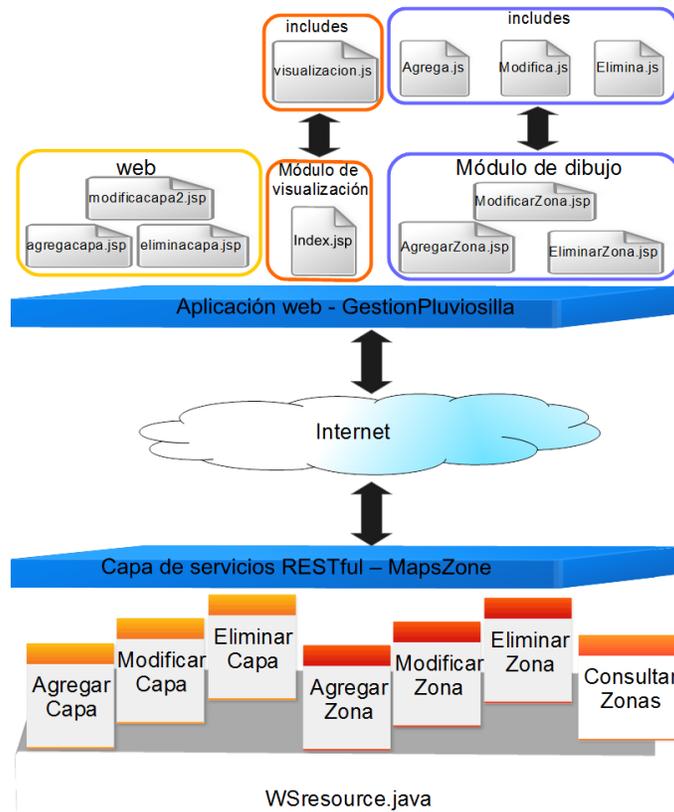


Figura 7. Arquitectura general del sistema.

La figura 8 muestra la arquitectura del sistema de forma general, en ella se ven las bases de datos involucradas y los módulos.

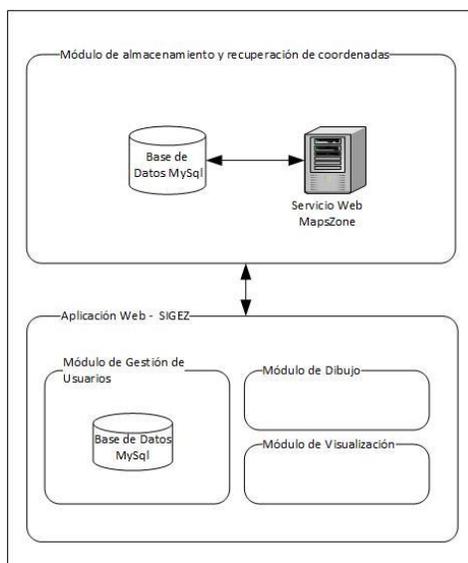


Figura 8. Arquitectura general del sistema en módulos.

6.2.5 Estructura de la base de datos

Esta sección cuenta con las dos bases de datos, la de GestionPluviosilla y MapsZone.

6.2.5.1 GestionPluviosilla

La figura 9 muestra la base de datos de GestionPluviosilla. En la tabla cuenta se almacena un identificador de usuario, el nombre, contraseña y privilegio en el sistema.

cuenta	
idCuenta	INT
user	VARCHAR(30)
pass	VARCHAR(30)
priv	INT
Indexes	

Figura 9. Base de datos de GestionPluviosilla.

6.2.5.2 MapsZone

La figura 10 muestra la base de datos del servidor web MapsZone. La empresa contiene capas, las capas contienen zonas y las zonas se componen de coordenadas.

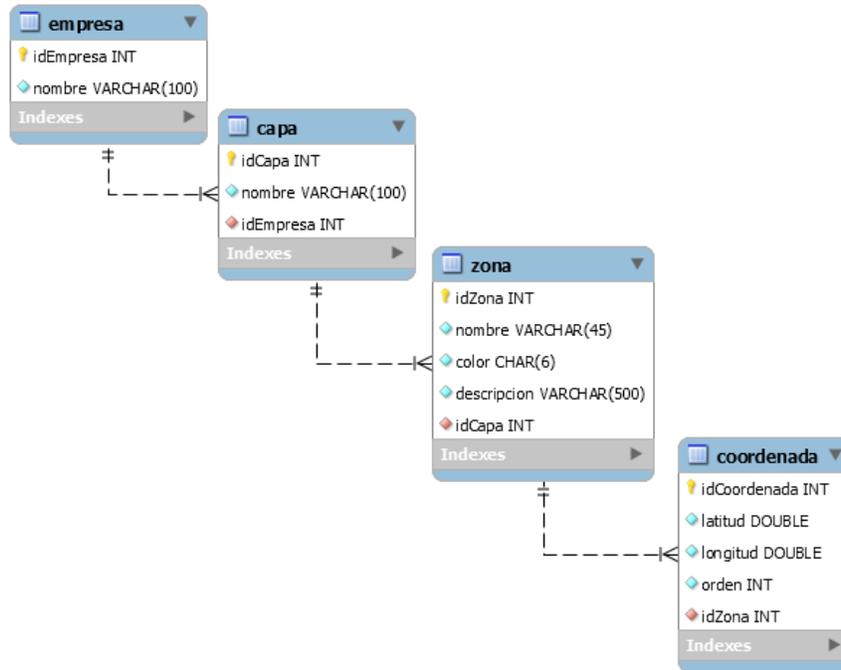


Figura 10. Base de datos de MapsZone.

6.3 Uso del sistema

Esta sección muestra la implementación de los principales casos de uso.

6.3.1 Agregar Zona

Para agregar una ruta debe de dirigirte al menú principal, en el debe elegir la opción de rutas/dibujo, esto despliega un menú, debe elegir la opción agregar, en el submenú sale una lista con las capas sobre las cuales puede agregar una ruta, debe de seleccionar la capa en la que se quiera la inserción. Se muestra en la figura 18 un ejemplo de esto.



Figura 11. Agregar ruta.

Esto lleva a una ventana donde debe de ingresar los datos correspondientes a la ruta que se desea insertar. Esta ventana se muestra en la figura 19.

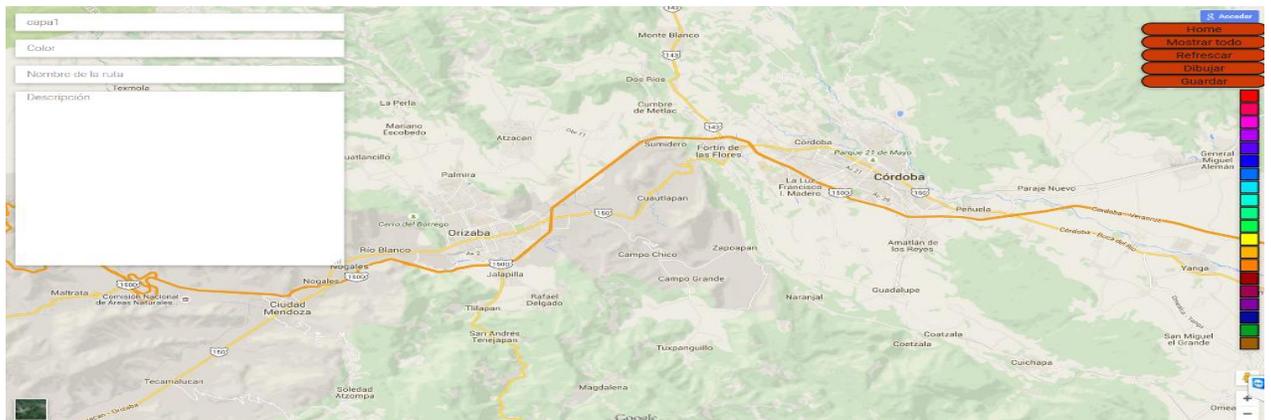


Figura 12. Interfaz para agregar ruta.

En la figura 20 se muestra un ejemplo del llenado, los pasos que debe seguir son: elegir el color de la zona de la barra de colores que se encuentra del lado derecho, ingresa el nombre de la ruta y su descripción en el formulario que se encuentra del lado izquierdo y dar clic en dibujar.

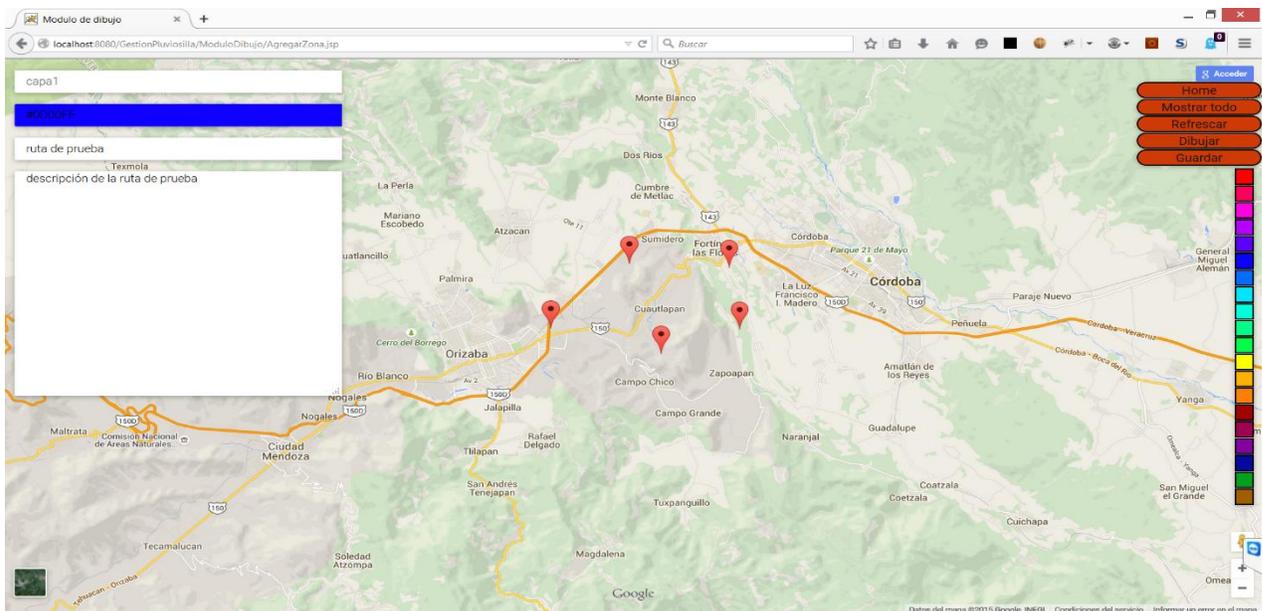


Figura 13. Captura de datos de la ruta nueva.

En la figura 21 se muestra la zona que se dibujó, aun se pueden modificar los puntos que la componen solo se deben de arrastrar los círculos blancos que limitan la ruta. Una vez que esté lista se debe de dar clic en guardar.

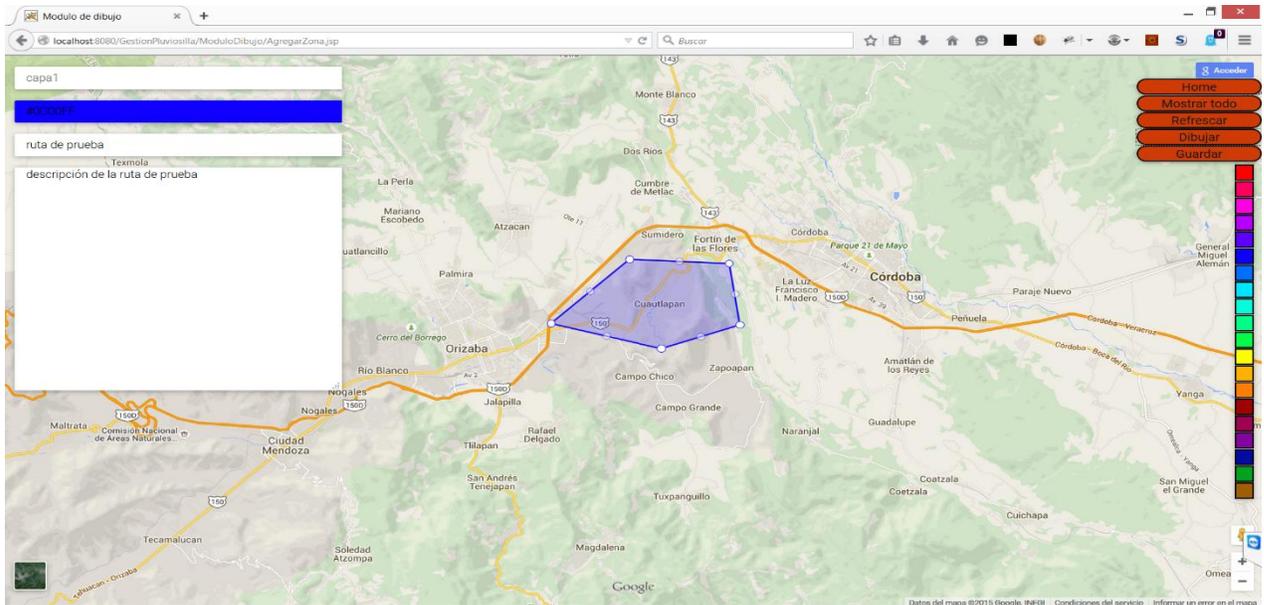


Figura 14. Capa dibujada.

Si la capa se dibujó correctamente se desplegará un mensaje confirmándolo como se muestra en la figura 22.

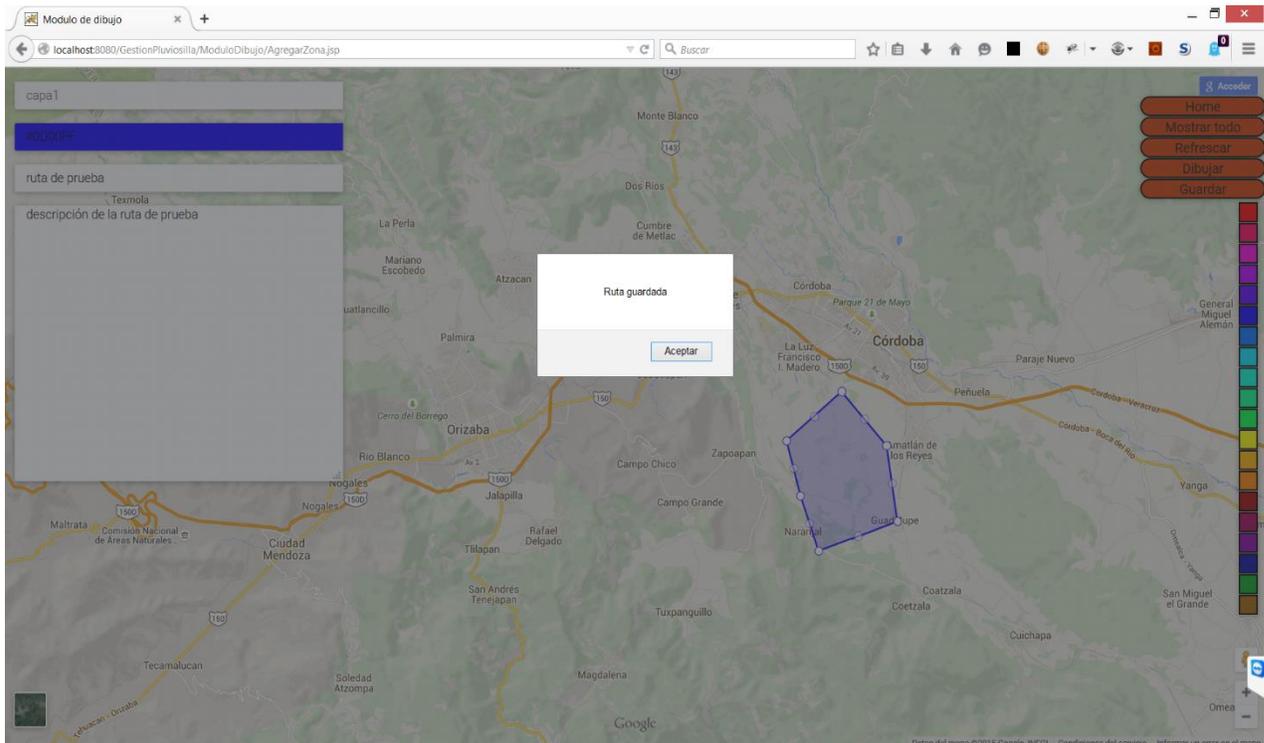


Figura 15. Ruta guardada.

6.3.2 Modificar Zona

Para modificar una ruta debe de dirigirse al menú principal, en él debe elegir la opción de rutas/dibujo, esto te despliega un menú, debe elegir la opción modificar, esto muestra un submenú en el cual debe de elegir la capa en la cual está la ruta que se quiere modificar. En la figura 23 se muestra el ejemplo.



Figura 16. Modificar capa.

Esto te lleva a una ventana donde se visualizan todas las rutas que tienen esa capa. Esta ventana se muestra en la figura 24.

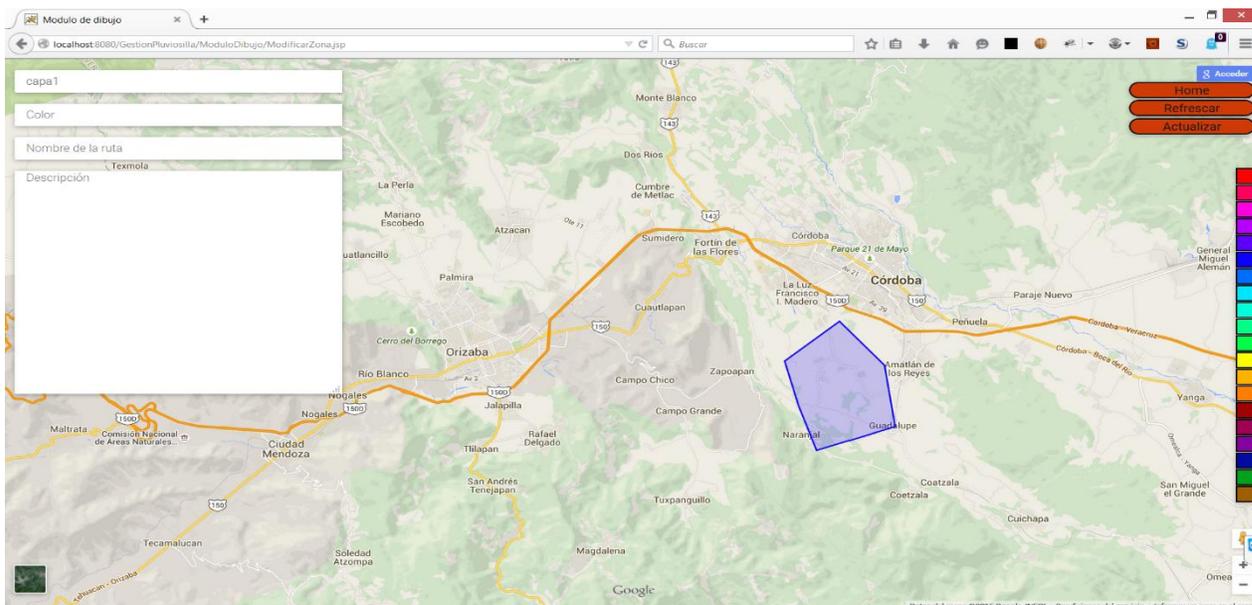


Figura 17. Rutas que se pueden editar.

Para modificar la ruta debe de darle clic, esto hará que se rellene el campo de color, nombre de la ruta y descripción. En la ruta que seleccionaste aparecerán los puntos que la limitan. Se puede editar el color, el nombre, la descripción y los puntos que la limitan. Para modificar

los puntos que la limitan basta con arrastrar los círculos blancos al lugar deseado. La figura 25 muestra un ejemplo de una ruta seleccionada.

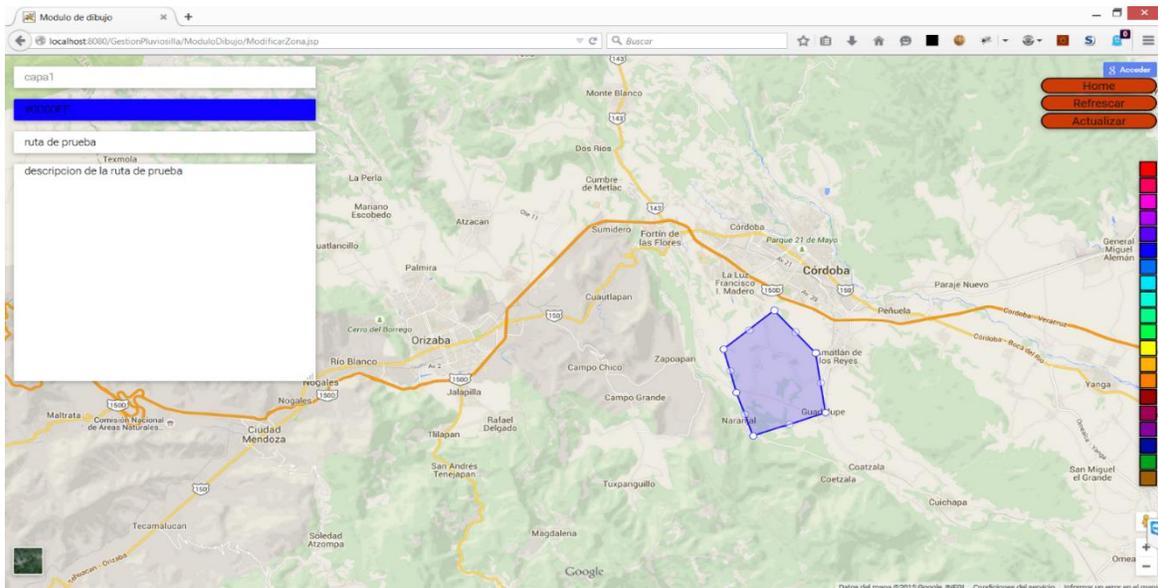


Figura 18. Ruta seleccionada.

Por último, una vez que esté lista la ruta se debe de dar clic en el botón de actualizar. Si la operación se lleva sin fallas se mostrará un mensaje que indica que la ruta se actualizó, la figura 26 muestra el mensaje.

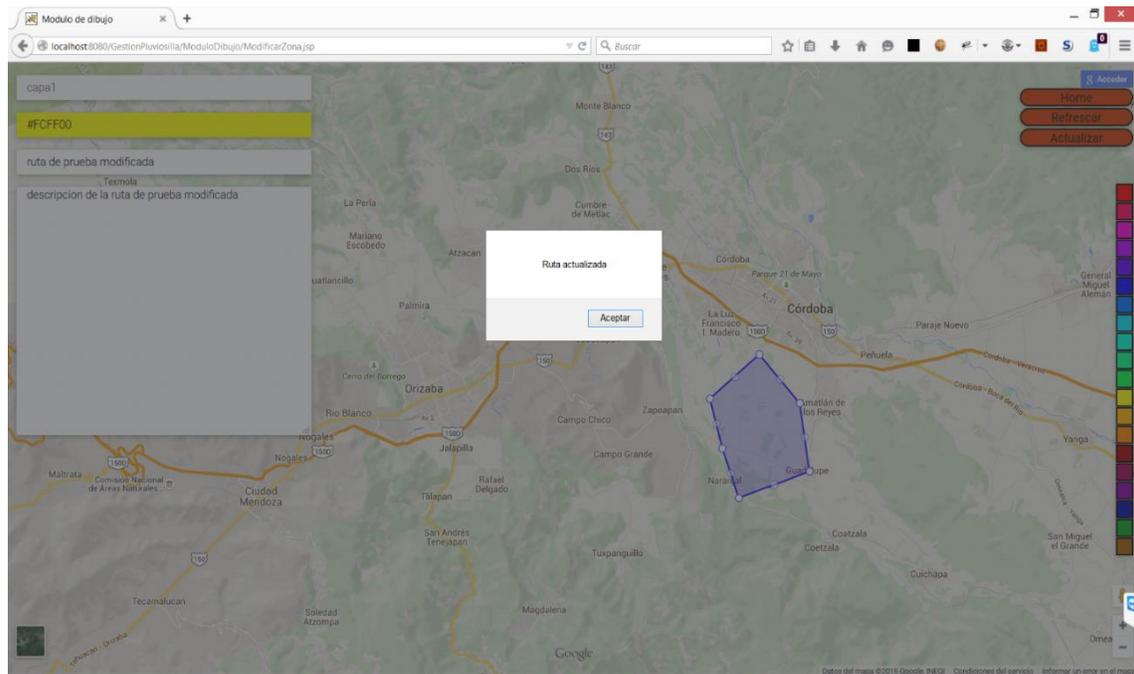


Figura 19. Confirmación de modificación de ruta.

6.3.3 Eliminar Zona

Para eliminar una ruta debe de dirigirse al menú principal, en el debe elegir la opción de rutas/dibujo, esto despliega un menú, debe elegir la opción eliminar, aparecerá un submenú con las capas que contienen las rutas, debe de elegir la capa donde esta la ruta que se eliminará. En la figura 27 se muestra el ejemplo.



Figura 20. Eliminar ruta.

Esto lleva a una ventana donde se visualizan todas las rutas que tienen esa capa. Esta ventana se muestra en la figura 28.

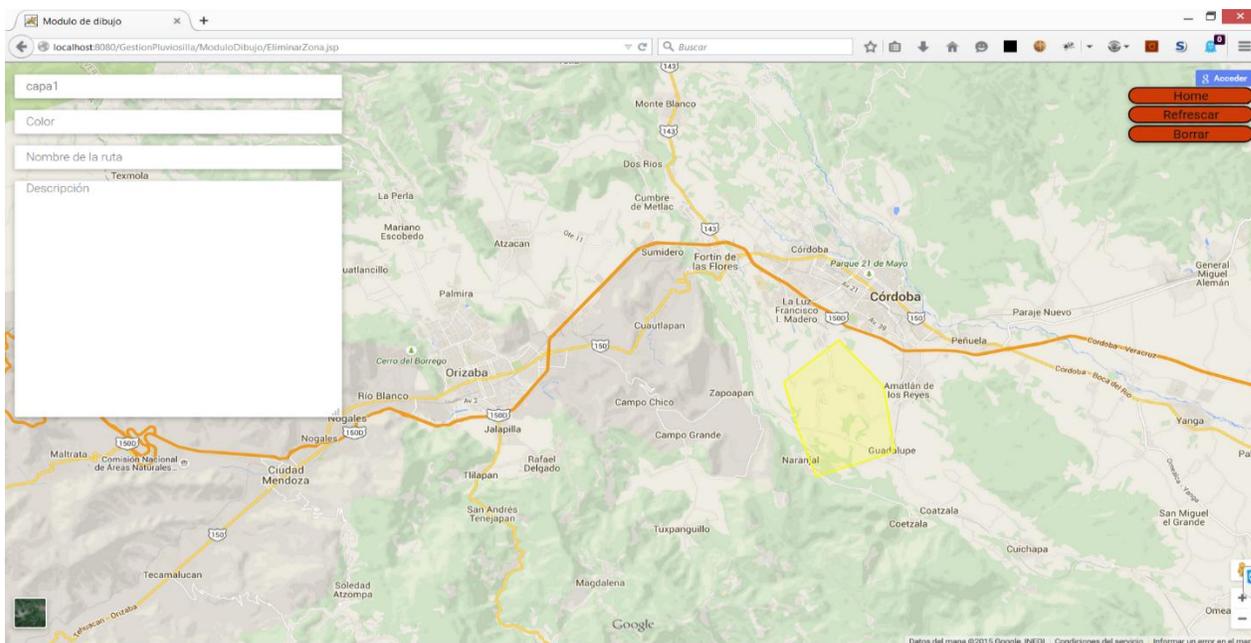


Figura 21. Interfaz de eliminar ruta.

Para eliminar la ruta debe de seleccionarla, una vez que la selecciono los datos de la ruta aparecerán en pantalla. Si está seguro de que es la ruta puedes dar clic en borrar, en caso de

que no sea la ruta adecuada puedes seleccionar otra. La figura 29 muestra la selección de la ruta a eliminar.

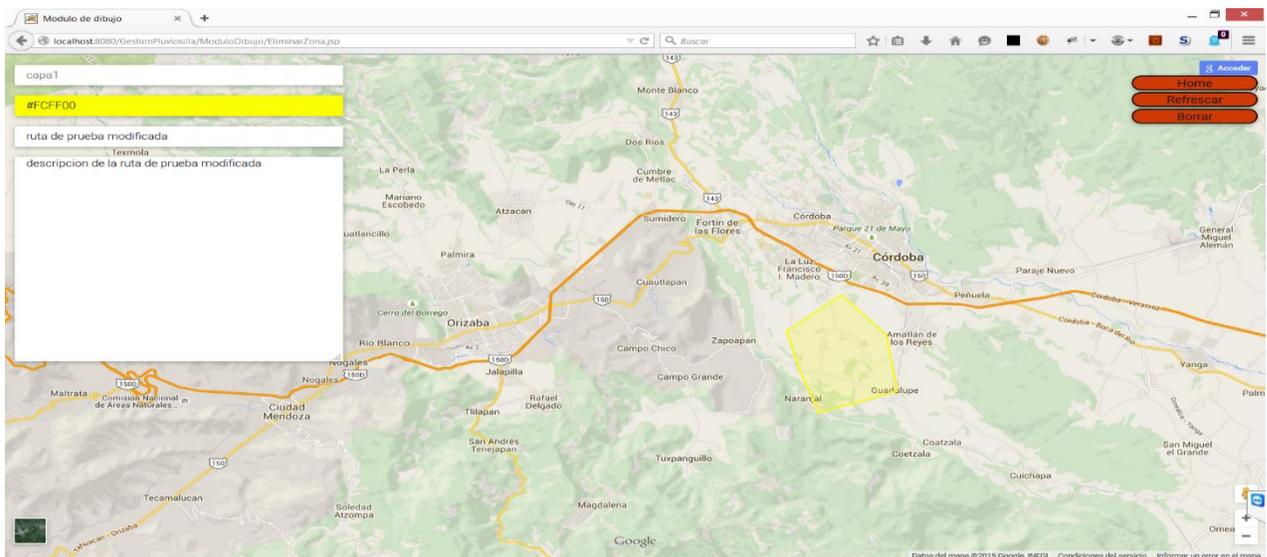


Figura 22. Ruta a eliminar seleccionada.

Si la eliminación fue correcta se mostrará en pantalla un mensaje que lo indica. Este mensaje se muestra en la figura 30.

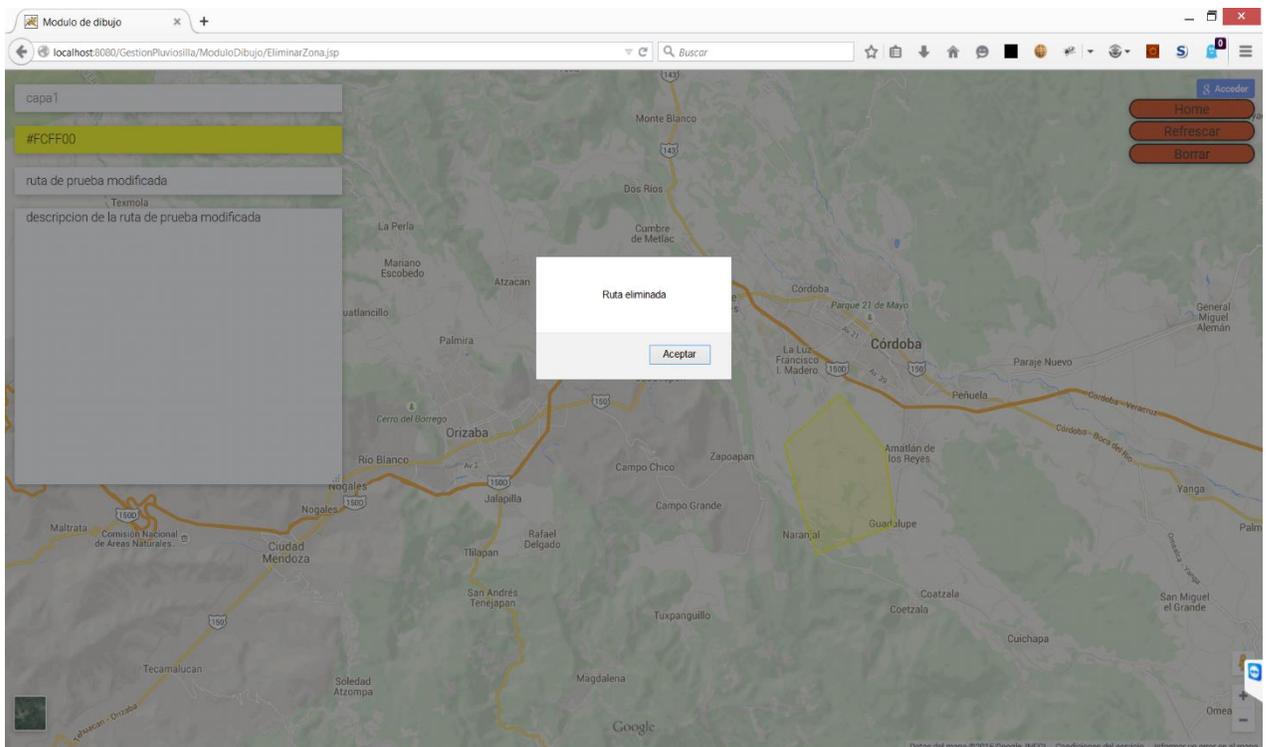


Figura 23. Confirmación de eliminación de ruta.

7 Hardware y software necesario

Esta sección indica el hardware y software que se ocupó en el desarrollo y puesta en marcha del sistema.

7.1 Tecnología usada para el desarrollo

7.1.1 Hardware

El proyecto se elaboró con un computador con las siguientes características.

- Procesador: Intel Core i7-4500u.
- Memoria ram: 8 Gb.
- Disco duro: 1 TB.

7.1.2 Software

Sistema operativo

- Ubuntu 14.0x4 LTS

Lenguajes de programación

- Java
- Javascript

IDE

- NetBeans
- WebStorm

Frameworks, API's

- Hibernate 4.

Servidor de aplicaciones

- Apache Tomcat 8.0.21

Manejador de base de datos

- MySql 5.5.43

7.2 Tecnología y puesta en marcha de la aplicación

7.2.1 Hardware

El proyecto se aloja en un computador con las siguientes características.

- Procesador: 1 CORE.
- Memoria ram: 1024 Mb.
- Disco duro: 25 GB.

7.2.2 Software

Se puede ocupar cualquier sistema operativo que pueda soportar Apache Tomcat 8.0.21 , MySql 5.5.43 y Java 8.

Sistema operativo

- Ubuntu 14.04-64 Minimal for VSI

Servidor de aplicaciones

- Apache Tomcat 8.0.21

Manejador de base de datos

- MySql 5.5.43

8 Resultados

Esta sección describe los resultados obtenidos, a partir del análisis y diseño de los requerimientos del usuario.

El sistema visto como un conjunto se compone de la aplicación web (SIGEZ) y de las solicitudes que hace al servicio web (MapsZone).

- SIGEZ (GestionPluviosilla)

Esta aplicación web contiene el módulo de visualización y el de dibujo, adicionalmente a esto por requerimientos del cliente se incluye un módulo encargado de la gestión de usuarios.

- MapsZone

Este servicio es el encargado de gestionar las peticiones que le hace SIGEZ, estas peticiones únicamente involucran el manejo de coordenadas, rutas y capas. Por lo tanto el módulo de almacenamiento y recuperación de coordenadas está representado por MapsZone.

El módulo de visualización contiene opciones que le permiten al usuario distinguir entre capas, visualizar todas las capas, buscar direcciones o lugares.

El módulo de dibujo se dividió en tres secciones: agregar, modificar y eliminar rutas.

Dentro de SIGEZ se implantó un módulo encargado de la gestión de los usuarios.

9 Análisis y discusión de resultados

Inicialmente en el levantamiento de requerimientos nunca se mencionó el manejo de usuarios, esta necesidad se evidenció cuando se estaba modelando la aplicación. Si bien la aplicación podría funcionar sin este manejo, la Gasera requería tener un control de los usuarios.

Cuando se realizó la propuesta de proyecto de integración se concibieron tres módulos. Uno de ellos no se elaboró, ese módulo se iba a encargar de la comunicación con Google Maps. Esta decisión se tomó por el manejo tan especial que módulo requería en cuanto a la interacción con los mapas.

El sistema se dividió en SIGEZ y MapsZone, esta decisión se tomó ya que se quería cierta independencia entre los sistemas. SIGEZ funciona como una especie de cliente ligero, el cual solo recibe los datos de MapsZone. Esto le dio mucha flexibilidad al sistema, por esta razón se pensó en que se podía expandir a otras gaseras de la misma empresa para el cual fue elaborado. Sin embargo estos cambios no podían llevarse a cabo ya que la base de datos no lo permitía, esto derivó en un cambio en la base de datos.

A pesar de que el servidor web puede gestionar otras gaseras u empresas, tiene una limitante ya que no existe una interfaz que permita adicionar a la base de datos estas empresas, por esta razón se tendría que ir directamente a la base de datos y agregarla, después, crear una

aplicación que genere algo parecido a SIGEZ cambiando todo el entorno a una nueva empresa.

10 Conclusiones

El objetivo del proyecto se cumplió completamente, actualmente la gasera cuenta con un sistema web que permite la gestión de sus rutas.

Durante el diseño del sistema se vio la falta de un mayor análisis del problema.

La tabla 1 muestra los objetivos particulares inicialmente propuestos y su alcance.

Objetivo particular	Alcance
Diseñar e implementar un módulo para poder gestionar la comunicación entre con Google Maps.	Aunque este módulo no se implementó de manera independiente. La comunicación con Google Maps se maneja en cada módulo que lo necesita.
Diseñar e implementar un módulo para poder dibujar sobre un mapa.	Se modeló y elaboró un módulo que gestiona agregar, modificar y eliminar rutas.
Diseñar e implementar un módulo para almacenar y recuperar las coordenadas de un mapa.	Se modeló y elaboró un módulo de servicios web que se encarga de gestionar todas las solicitudes relacionadas con las rutas.
Diseñar e implementar un módulo para poder visualizar las zonas definidas en un mapa.	Se modeló y elaboró un módulo que permite la visualización de zonas.
Integrar los módulos en una aplicación web.	Los módulos se integraron exitosamente.
Instalar el sistema en el servidor que destine la empresa.	El sistema se ejecuta actualmente en un servidor que se proporcionó para dicho proyecto.
Realizar pruebas en el ambiente de producción.	Las pruebas en producción se realizaron por las personas que fueron capacitadas para usar el sistema, partiendo de ellas se hicieron cambios mínimos en la interfaz encargada de la visualización.

11 Perspectivas del proyecto

El sistema está preparado para una ampliación, durante la fase de diseño y construcción se dio un cambio en el diseño de la base de datos, con este cambio MapsZone puede gestionar más de una empresa.

El trabajo futuro consiste en agregar una interfaz que permita agregar nuevas empresas, y que a su vez genere un entorno similar a SIGEZ.

Para que SIGEZ mantenga la esencia de cliente ligero se tendría que extraer el módulo de gestión de usuarios, el cual podría vivir en otro servidor, este servidor podría proporcionar los accesos.

SIGEZ debe de ser reestructurado, se debe hacer uso de interfaces para tener una mayor flexibilidad.

12 Referencias bibliográficas

[1] Li Su, “A method for building thematic map of GIS based on Google Maps API.,” presented at the 19th International Conference on Geoinformatics, China, Beijing., 2011, vol. 32, p. 4.

[2] S. Jeberson Retna Raj, R. T., “Disaster management system based on GIS web services,” presented at the Recent Advances in Space Technology Services and Climate Change (RSTSCC), Chennai, India, 2010, pp. 448 – 451.

[3] Avendaño Méndez Sergio Enrique, “Gestión de calificaciones de cursos mediante servicios Web,” Universidad Autónoma Metropolitana Unidad Azcapotzalco, México, D.F., 2010.

[4] Echenique Álvarez Lucía, “Calendarios compartidos y eventos grupales (cceg): aplicación de escritorio para la gestión de eventos individuales y grupales en calendarios de google.,” Universidad Autónoma Metropolitana Unidad Azcapotzalco, México, D.F., 2013.

[5] Rubidio Ugalde Sergio, Díaz Alegría Tonatihu, “Aplicación de tecnologías de cómputo móvil y sistemas de información para la mejora en el proceso de negocio dentro de un restaurante.” Universidad Autónoma Metropolitana Unidad Azcapotzalco, México, D.F., 2009.

[6] Dosideas(2015,Julio), “Introducción a los servicios web RESTful,” Disponible: <http://www.dosideas.com/noticias/java/314-introduccion-a-los-servicios-web-restful.html>

Apéndice A. Entregable: Listado del API del código fuente desarrollada

[Enlace a código fuente GestionPluviosilla](#)

[Enlace a código fuente MapsZone](#)

Apéndice B. Entregable: Memoria de diseño

[Enlace a memoria de diseño](#)

Apéndice C. Entregable: Manual técnico

[Enlace a manual técnico](#)

Apéndice D. Entregable: Manual de usuario

[Enlace a manual de usuario](#)