

# **MIGRACIÓN VISOR SIG A JAVASCRIPT**

**AUTOR, WILLIAM PÉREZ CARDONA**



**UNIVERSIDAD DE MANIZALES  
FACULTAD DE CIENCIAS E INGENIERÍA  
ESPECIALIZACIÓN EN SISTEMAS DE INFORMACIÓN GEOGRÁFICA  
MANIZALES  
2016**

# **MIGRACIÓN VISOR SIG A JAVASCRIPT**

**AUTOR, WILLIAM PÉREZ CARDONA**

Trabajo de Grado presentado como opción parcial para optar  
al título de Especialista en Sistemas de Información Geográfica

**UNIVERSIDAD DE MANIZALES  
FACULTAD DE CIENCIAS E INGENIERÍA  
ESPECIALIZACIÓN EN SISTEMAS DE INFORMACIÓN GEOGRÁFICA  
MANIZALES  
2016**

## **AGRADECIMIENTOS**

El autor expresa sus agradecimientos a:

Wilson Andrés Rudas, Jefe equipo de desarrollo Aguas de Manizales, agradecimientos por haber gestionado, impulsado y ayudar a dar enfoque al proyecto

Víctor Hugo Vargas, Ingeniero de desarrollo SIG Aguas de Manizales, agradecimientos por su valioso aporte en la orientación y conocimientos del sistema suministrados

## CONTENIDO

	Pág.
INTRODUCCIÓN	10
1. ÁREA PROBLEMÁTICA	11
2. OBJETIVOS	13
2.1 OBJETIVO GENERAL	13
2.2 OBJETIVOS ESPECÍFICOS	13
3. JUSTIFICACIÓN	14
4. MARCO TEÓRICO	15
4.4 ANTECEDENTES	17
5. METODOLOGÍA	22
5.1 TIPO DE TRABAJO	22
5.2 PROCEDIMIENTO	22
5.2.1 Fase 1 Levantamiento de información	22
5.2.2 Fase 2 Implementación.	22
6. RESULTADOS	25
6.1.2 COMPARATIVO DE RESULTADOS	27
7. CONCLUSIONES	31
8. RECOMENDACIONES	32
BIBLIOGRAFÍA	33

## LISTA DE FIGURAS

	<b>Pág.</b>
Figura 1. Esquema Visor	11
Figura 2. Visor Segovia – Silverlight	12
Figura 3. Panel derecho	12
Figura 4. Panel inferior	13
Figura 5. Arquitectura Segovia	16
Figura 6. Visor asignación en JavaScript Versión 1	25
Figura 7. Visor asignación en JavaScript Versión 2	26
Figura 8. Visor asignación en JavaScript Versión Final	27
Figura 9. Interfaz gráfica del aplicativo restitución de tierras	18
Figura 10. Modelo vista controlador Segovia	22
Figura 11. Modelo de Casos de Uso Visor Asignación	23
Figura 12. Descripción elementos visor Silverlight	27
Figura 13. Descripción elementos visor JavaScript	28
Figura 14. Selector de capas visor Silverlight	28
Figura 15. Selector de capas visor JavaScript	28
Figura 16. Barra comandos visor Silverlight	29
Figura 17. Barra comandos visor JavaScript	29
Figura 18. maptip visor Silverlight	30
Figura 19. infotemplate visor JavaScript	30
Figura 20. Esquema de Arquitectura Visor Geográfico de Parqueaderos	19

## GLOSARIO

**API:** Interfaz de programación de aplicaciones, es el conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Una API representa la capacidad de comunicación entre componentes de software. Se trata del conjunto de llamadas a ciertas bibliotecas que ofrecen acceso a ciertos servicios desde los procesos y representa un método para conseguir abstracción en la programación.

**Servicio REST:** Servicio de mapas de resultado, es un servicio de mapas dinámico y admite todas las operaciones y recursos secundarios de un servicio de mapas dinámico típico, haciendo uso del protocolo cliente/servidor sin estado: cada mensaje HTTP contiene toda la información necesaria para comprender la petición. Como resultado, ni el cliente ni el servidor necesitan recordar ningún estado de las comunicaciones entre mensajes.

**Silverlight:** Aplicaciones Silverlight son aplicaciones del lado del cliente que se pueden incrustar en una página web. Las aplicaciones Silverlight pueden proporcionar funcionalidad compleja para interactuar con los medios de comunicación, presentaciones y servicios web que no pueden ser proporcionados de otro modo por las tecnologías basadas en la web. Para su uso es necesario descargar un plugin en el cliente el cual se encarga de interpretar lo que es el código XAML(eXtensible Application Markup Language).

**JavaScript:** JavaScript es un lenguaje de scripting multiplataforma y orientado a objetos. Es un lenguaje pequeño y liviano que se puede utilizar para construir sitios Web y para hacerlos más interactivos. Java es un lenguaje de programación basado en clases, diseñado para lograr seguridad de tipos, y ejecución rápida. Seguridad de tipos significa, por ejemplo, que no es posible acceder a memoria protegida mediante la alteración del bytecode de una clase. El núcleo de JavaScript puede extenderse para varios propósitos, complementándolo con objetos adicionales, por ejemplo:

**Client-side JavaScript:** Extiende el núcleo del lenguaje proporcionando objetos para controlar un navegador y su modelo de objetos (o DOM, por las iniciales de Document Object Model). Por ejemplo, las extensiones del lado del cliente permiten que una aplicación coloque elementos en un formulario HTML y responda a eventos del usuario, tales como clicks del ratón, ingreso de datos al formulario y navegación de páginas.

**Server-side JavaScript.** Extiende el núcleo del lenguaje proporcionando objetos relevantes a la ejecución de JavaScript en un servidor. Por ejemplo, las extensiones del lado del servidor permiten que una aplicación se comuniquen con una base de datos, proporcionar continuidad de la información de una invocación de la aplicación a otra, o efectuar manipulación de archivos en un servidor.

**Web Map Service:** El servicio Web Map Service (WMS) definido por el OGC (Open Geospatial Consortium) produce mapas de datos referenciados espacialmente, de forma dinámica a partir de información geográfica. Este estándar internacional define un "mapa" como una representación de la información geográfica en forma de un archivo de imagen digital conveniente para la exhibición en una pantalla de ordenador. Un mapa no consiste en los propios datos. Los mapas producidos por WMS se generan normalmente en un formato de imagen como PNG, GIF o JPEG, y opcionalmente como gráficos vectoriales en formato SVG (Scalable Vector Graphics) o WebCGM (Web Computer Graphics Metafile).

El estándar define tres operaciones:

1. Devolver metadatos del nivel de servicio.
2. Devolver un mapa cuyos parámetros geográficos y dimensionales han sido bien definidos.
3. Devolver información de características particulares mostradas en el mapa (opcionales).

Las operaciones WMS pueden ser invocadas usando un navegador estándar realizando peticiones en la forma de URLs (Uniform Resource Locators). El contenido de tales URLs depende de la operación solicitada. Concretamente, al solicitar un mapa, la URL indica qué información debe ser mostrada en el mapa, qué porción de la tierra debe dibujar, el sistema de coordenadas de referencia, y la anchura y la altura de la imagen de salida.

El servicio WMS permite así la creación de una red de servidores distribuidos de mapas, a partir de los cuales los clientes pueden construir mapas a medida.

## RESUMEN

Este texto representa los principales resultados de la migración de un visor desarrollado en la tecnología Silverlight, la cual se encuentra obsoleta y sin soporte. Como mejor alternativa pensando en su necesaria integración con un servidor ArcGIS Server, se encontró la migración al lenguaje JavaScript y como API se elige la proporcionada por esri, ArcGIS API for JavaScript versión 3.17. En el proceso se abordaron aspectos como la carga de capas haciendo uso de servicios REST, en el nuevo visor se crean componentes que ayudan a interactuar con las capas y además se realiza la integración con librerías gráficas como DOJO y JQuery.

**PALABRAS CLAVES:** Sistema de información geográfico, servidor de mapas, ArcGIS API for JavaScript, mapeo web, Servicios REST



## ABSTRACT

This paper presents the main results of the necessary migration of a Silverlight technology visor which is obsolete and without support. Like best alternative thinking in its necessary with a ArcGIS server, was found migration of javascript language and like API its elected the created by esri. ArGIS API for JavaScript version 3.17 . In the process were incorporated aspects as the load layers making use of REST service, in the new visor are created tools that help to interact with the layers and do the integration with graphics libraries like DOJO and JQuery.

**KEY WORDS:** geographic information system, map server, ArcGIS API for JavaScript, web mapping, REST services

## INTRODUCCIÓN

En las últimas décadas los Sistemas de Información Geográfica, están evolucionando gracias al desarrollo tecnológico de la informática e internet desarrollando aplicaciones que permiten desplegar los mapas en la web e incluso permiten al público realizar análisis. La década de los noventa fue la protagonista del comienzo del despliegue de los visores cartográficos los cuales comenzaron poco después de la aparición de la “World Wide Web” (WWW) con la introducción del visualizador de Mapas Xerox PARC en 1993.

El año 2005 fue un punto de inflexión para la creación de nuevas aplicaciones y tipos de software para el desarrollo e implementación del SIG en la web desde el lanzamiento de las “Application Programming Interfaces” (API) de Mapas, un ejemplo importante de ello es la API de Google Maps, luego en julio de 2008 entraría ArcGIS API for Javascript, la respuesta de esri a la competencia de las herramientas para realizar web mapping.

Por otro lado en cuanto a los desarrollos la mayor parte de ellas se materializaban como desarrollos ex profeso y por tanto se resolvían los mismos problemas una y otra vez. Esta situación derivó como es natural hacia proyectos que intentan proporcionar un conjunto de componentes comunes en general en forma de documentos HTML y aplicaciones escritas en JavaScript que proporcionan al desarrollador una base sobre la que realizar su aplicación específica.

Paralelamente, se fueron desarrollando una serie de visores web para la consulta de datos específicos: mapas de progreso, catálogo de vuelos, o señales geodésicas. Los visores web dependían de la tecnología de cada fabricante y las tecnologías eran difícilmente integrables entre sí o algunas presentaron una corta vida como en el caso de Silverlight. Más tarde nacieron los geoservicios, que ofrecían todas las capas de información mencionadas mediante el protocolo WMS de OGC, con lo que pudieron consultarse conjuntamente desde diferentes aplicaciones remotas.

## 1. ÁREA PROBLEMÁTICA

La empresa Aguas de Manizales, encargada de operar las redes de acueducto y alcantarillado de la ciudad de Manizales, para soportar su operación cuenta con un software desarrollado a medida llamado Segovia, el cual tiene un componente espacial compuesto por un visor SIG para la gestión de las ordenes de trabajo ejecutadas por la empresa, con el esquema definido en la Figura (1)

Este visor de Asignación de órdenes de trabajo Figura (2), compuesto por un visor de mapas, un panel lateral derecho figura (3) con una grilla correspondiente al listado de las ordenes de trabajo y unas pestañas de agrupación de elementos, y en la parte inferior figura (4) el visor cuenta con un panel inferior conformado por un formulario y pestañas de agrupación de elementos. Este visor fue desarrollado haciendo uso de la tecnología Silverlight, la cual tiene actualmente una problemática de obsolescencia, junto con Java y Flash son tres plugins que son los más explotados por los cibercriminales para tomar el control de los navegadores, por tales razones estas tecnologías son bloqueadas por defecto en navegadores como Chrome y Firefox, además su creador Microsoft ha anunciado el abandono de esta tecnología, la cual no tendrá soporte en su nuevo navegador, Edge, y Microsoft ya ha animado a las compañías a que dejen de utilizar Silverlight en sus páginas web y que comiencen la transición a otras alternativas con mejor futuro.

Por las razones anteriormente descritas, es necesaria la búsqueda de alternativas para la migración del visor de Asignación de Ordenes de trabajo a una tecnología más universal y con una alta compatibilidad con el servidor ArcGIS Server y ArcGIS Online, con los cuales cuenta la empresa para el soporte y gestión de los servicios SIG.

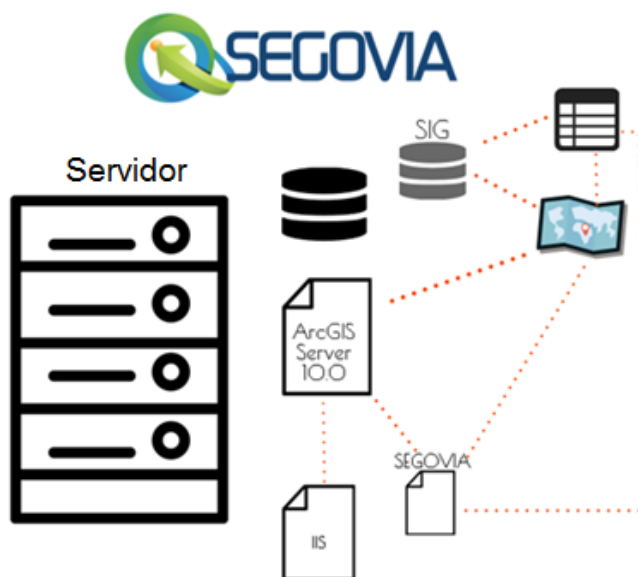


Figura 1. Esquema Visor

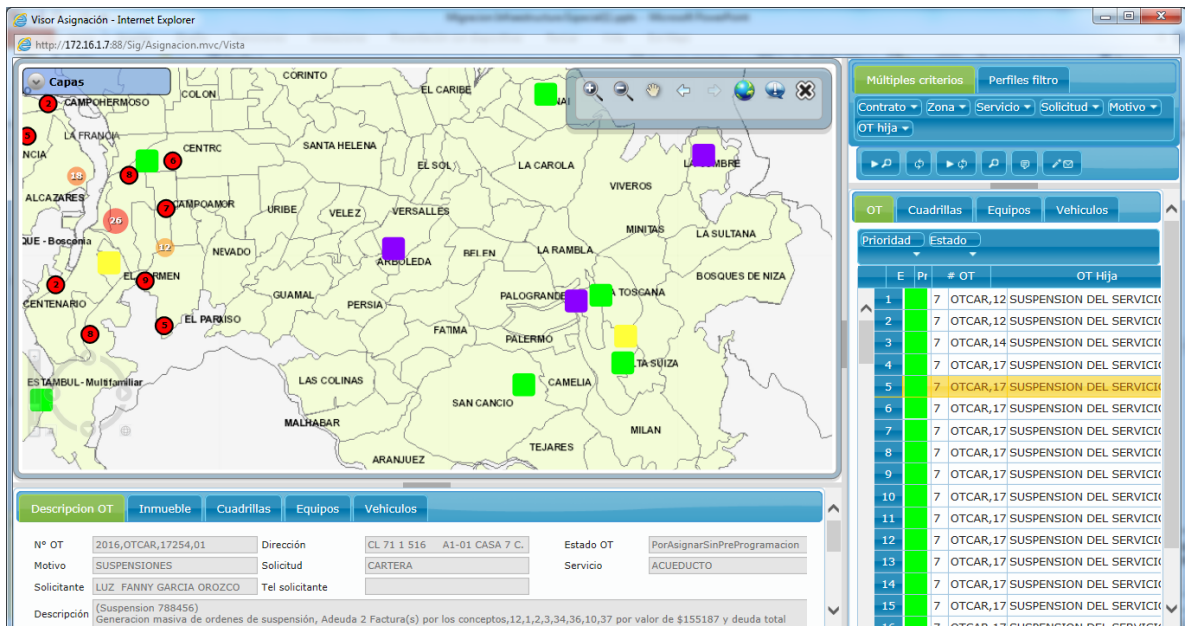


Figura 2. Visor Segovia – Silverlight



Figura 3. Panel derecho

Descripción OT	Inmueble	Cuadrillas	Equipos	Vehiculos	
N° OT	2015,OT,42207,01	Dirección	CR 32A 46 18	Estado OT	PorAsignar
Motivo	DESPEJAR MEDIDOR	Solicitud	MEDIDORES Y LLAVES	Servicio	ACUEDUCT
Solicitante	LIDA P - ED	Tel solicitante			
Descripción	FAVOR RETIRAR MEDIDOR MUY PROFUNDO Y CON TUBERIA GALVANIZADA PARA PRUEBAS, INSTALAR MED PROVISIONAL. CONTACTO: FRANCISCO RINCON.				
Fecha OT	10-09-2015 11:45:22	Tiempo OT	319d:5h:31m	Tiempo ejecución	
Zona	ZONA 2 URBANA CRA 23	Diámetro		Material	
Fec preprog		Suspensión			
Obs. suspensión					

**Figura 4. Panel inferior**

## 2. OBJETIVOS

### 2.1 OBJETIVO GENERAL

Implementar un visor SIG en la tecnología JavaScript, haciendo uso de la api ArcGIS API For JavaScript para sustituir el visor de asignación de ordenes de trabajo existente, el cual fue desarrollado en Silverlight.

### 2.2 OBJETIVOS ESPECÍFICOS

- Realizar conexión a las bases de datos espaciales y transaccionales de Segovia, para realizar las consultas de la información de las órdenes de trabajo y el consumo de los servicios REST.
- Cargar en el visor el mapa base de la ciudad de Manizales y capas de acueducto, alcantarillado, vehículos, equipos, cuadrillas y ordenes de trabajo.
- Crear herramientas de navegación e interacción con las capas y mapa base.

### 3. JUSTIFICACIÓN

Las aplicaciones desarrolladas con Silverlight necesitan utilizar plugins para que la aplicación pueda ejecutarse en el navegador del cliente, estos plugins corresponden a software adicional que debe ser instalado. Con el uso de Javascript toda la funcionalidad la ejecutan los navegadores de serie lo que constituye una ventaja en eficiencia y agilidad de uso.

El hecho de tener que utilizar plugins, como en el caso de Silverlight, puede ser un inconveniente si se requiere que la aplicación se pueda ver en todo tipo de dispositivos sin problemas, esta es quizás una de las mayores bondades de la tecnología multiplataforma como Javascript, la cual permite crear mapas responsive (adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarlas), para su cómoda visualización en diferentes dispositivos. JavaScript es el lenguaje líder de interfaces web para crear aplicaciones que se ejecutan en el navegador. Utilizado tanto por Google Maps como por los principales clientes de mapas web open source (OpenLayers o Leaflet).

Considerado como el pionero en los SIG, ESRI ha creado la Api Arcgis API for JavaScript, para hacer webmapping, la cual cuenta con muchas funcionalidades y además es compatible con lenguajes de uso masivo como son JQuery, ExtJS y una alta integración con ArcGIS Online o ArcGIS Server. Por estas razones es conveniente y necesario realizar la migración de tecnología del visor de asignación de ordenes de trabajo a la tecnología JavaScript, haciendo uso de la Api de Esri con el fin de lograr una mayor integración con el servidor de ArcGIS Server de la empresa.

## 4. MARCO TEÓRICO

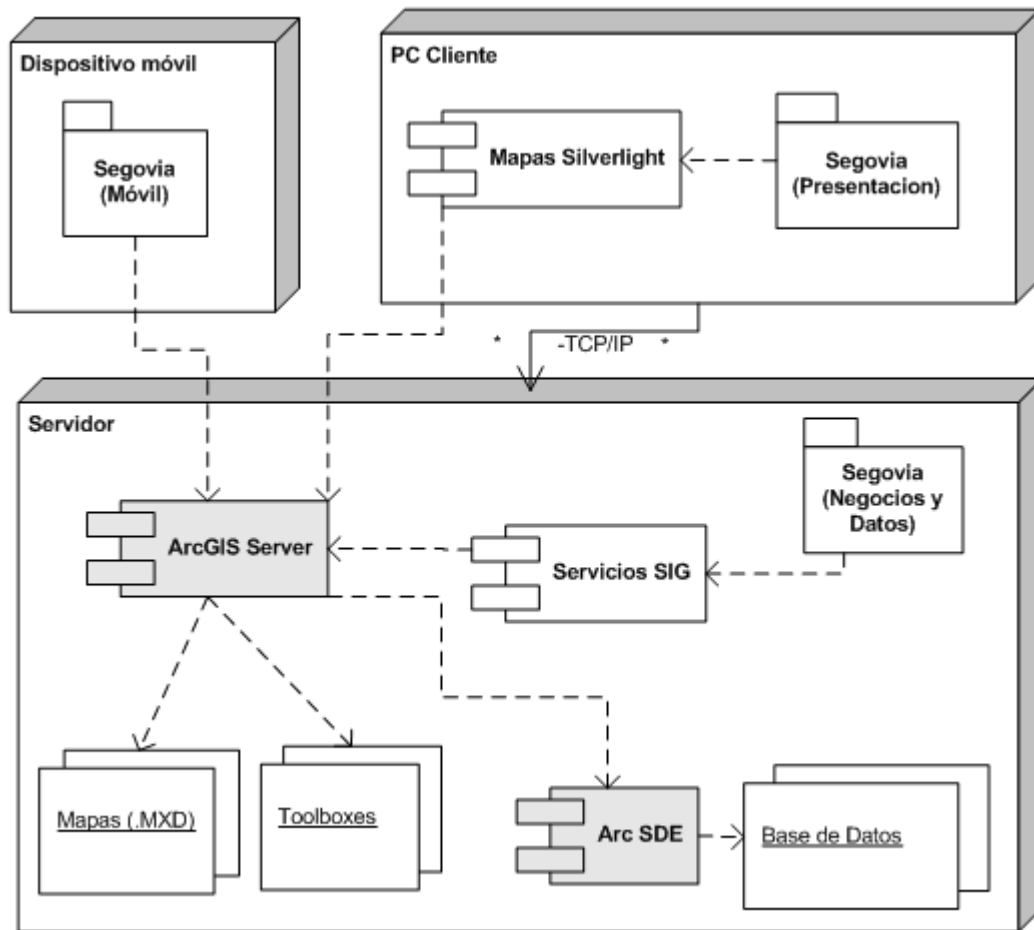
### 4.1 Segovia.

Segovia es un sistema de información desarrollado a medida para la empresa Aguas de Manizales, empresa de acueducto y alcantarillado de la ciudad de Manizales, este software es de gran envergadura y posee componentes web para la gestión desde oficinas, un módulo móvil para interacción con el sistema desde campo y módulo de Sistema de Información Geográfica (SIG) para ayudar a la toma de decisiones en oficina y ayudar a la ubicación geográfica en campo Figura 5.

Segovia cumple con la función de soportar toda la operación de la empresa, con este software se pueden captar todas las solicitudes reportadas por la comunidad al Call Center de atención al usuario, las cuales son consultadas y analizadas en la oficina donde se tiene el centro de control de la empresa, para determinar que solicitudes o grupo de solicitudes deben dar origen a ordenes de trabajo. Las ordenes de trabajo son generadas con un atributo espacial de tipo punto con la información espacial del sitio del daño o de la labor a realizar, estas ordenes pueden ser de diferentes tipos como tubo roto, toma de presión, reinstalación del servicio, visita de fontanero, detección de fugas, etc.

Las ordenes de trabajo son asignadas por el centro de control a cuadrillas que se encuentran en campo, las cuales pueden realizar la consulta del trabajo asignado haciendo uso de un dispositivo móvil, así mismo como de poder consultar la imagen SIG como ayuda a la ubicación del sitio. El centro de control hace uso del visor de asignación de ordenes de trabajo, para visualizar espacialmente las ubicaciones de las cuadrillas, las ordenes en ejecución y las que están pendientes, de esta forma se toma la mejor decisión de que ordenes asignar a las cuadrillas de acuerdo a su ubicación espacial. Además de las capas de ordenes de trabajo, cuadrillas y mapa base, el visor de asignación también da la posibilidad de interactuar con las capas de las redes de acueducto y alcantarillado, para de esta forma saber sobre qué red se va a operar.

Desde la Creación de Segovia en el año 2012, se han generado a el año 2016 más de 470 mil ordenes de trabajo y cada vez son más los procesos de la empresa que se apoyan en este software para soportar sus procesos.



**Figura 5. Arquitectura Segovia**

- **Mapas (.MXD):** Son archivos creados y mantenidos en ArcMap, en donde se define la forma de presentar la información espacial a un usuario (generalmente mapas), junto con la ubicación de tales datos (Geo-Databases).
- **Toolboxes:** Son elementos de Geoprocesamiento de ArcGIS que se utilizan para realizar diversos procesos en el SIG. En el caso de Segovia, se utilizan en la georreferenciación manual (desde el mapa) de OTs y Solicitudes.
- **ArcSDE:** Es el componente de ArcGIS que permite el almacenamiento, manejo de datos espaciales por medio de una Base de Datos empresarial.
- **ArcGIS Server:** Es el componente de ArcGIS que permite la publicación de mapas (archivos MXD), Locators y otros elementos SIG para ser consumidos en forma de Web Services por diversos clientes sobre una red.



- **Mapas Silverlight:** Son las mapas que se embeben dentro de la aplicación principal de SEGOVIA en una página Web.

## **4.2 ANTECEDENTES**

### **4.2.1 Aplicación Web Basada en SIG Para la Administración de la Información Catastral de las Solicitudes Recibidas Por la Unidad de Restitución de Tierras en El Valle del Cauca.**

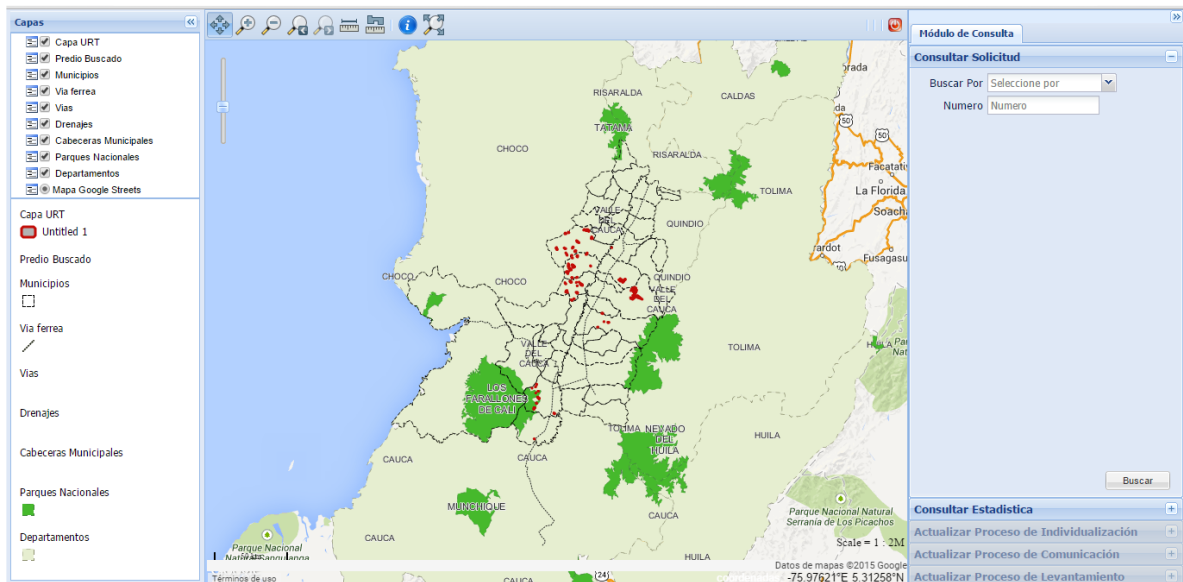
El objetivo de este trabajo fue el desarrollar una herramienta (Aplicación WEB basada en SIG) que facilite la administración eficiente de la información generada por el área catastral de la unidad de restitución de tierras URT del departamento del Valle del Cauca, esta entidad es la entidad encargada de restituir los predios que fueron abandonados por sus propietarios o poseedores a causa del conflicto armado del país.

La función principal de la aplicación WEB basada en SIG era suministrar información espacial y alfanumérica de los procesos que se desarrollan en el área catastral (Individualización, comunicación y levantamiento) de acuerdo a las solicitudes recibidas en la Unidad de Restitución de Tierras territorial Valle del Cauca.

El aplicativo cuenta con una página de inicio desde la cual se puede acceder a las consultas por medio del ingreso de una clave y contraseña definida, una vez logueado se puede consultar una solicitud específica, al filtrar por el número identificador de la solicitud deseada (id\_solicitud), o número de identificación del solicitante se obtendrá en una tabla de tipo acordeón la información alfanumérica específica de cada solicitud. La aplicación también da la posibilidad de navegar por el mapa con herramientas como Zoom (in, out, previous, next y extent), Pan, medidor de distancias y áreas. La herramienta Zoom Extent visualiza el área del departamento del Valle del Cauca.

El sistema cuenta con paneles para la visualización de la información a consultar, un primer panel sirve para la visualización de la información espacial que se desea sobreponer a la consulta que se realice, otro panel se usará para la visualización de la información espacial de acuerdo a la consulta realizada y un último panel se utilizará para establecer las consultas requeridas a realizar.

Con respecto a la estructura de la interfaz, está compuesta de diferentes paneles de información, los cuales se despliegan u ocultan, y es posible fijar los paneles haciendo clic en la parte superior-derecha de un panel abierto.



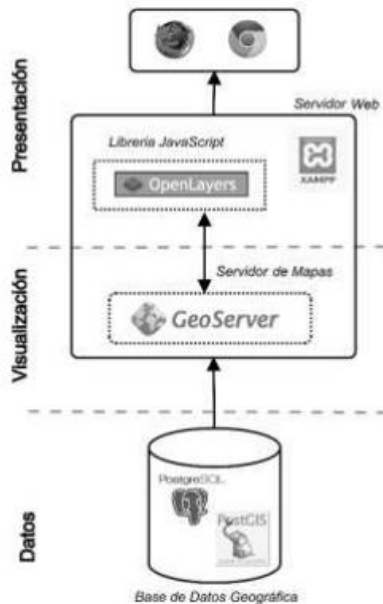
**Figura 9. Interfaz gráfica del aplicativo restitución de tierras**

En este desarrollo también se encuentra presente el uso de JavaScript para realizar parte de su creación. Como resultado de los scripts del lado del cliente crearon: formularios para las consultas, las cuales están estructuradas en forma de acordeón; un formulario para el registro del editor; paneles para la visualización de las capas y leyenda que permiten apreciar la forma y la simbología que contienen el visor geográfico; y una barra de herramientas, la cual permite una interacción con la información desplegada en el visor geográfico. Las herramientas presentes son: Máxima Extensión, Pan, Zoom In, Zoom out, Zoom previo, Zoom siguiente, cálculo de distancia, cálculo de área e información relacionada con la capa seleccionada. Todas estas ventanas se crearon a partir de JavaScript con las librerías GeoExt, Ext y OpenLayers.

#### **4.2.2 Implementación de visor geográfico que permita ubicar parqueaderos.**

Este trabajo se enfocó en la implementación de una herramienta web que utilizando un servicio de mapas ubique y muestre los parqueaderos cercanos a un punto específico en la localidad de Chapinero. La metodología aplicada en la implementación consistió en: obtener de los niveles de loteo, uso y placa domiciliaria del Mapa de referencia de Bogotá la ubicación de los parqueaderos públicos; configurar y publicar los puntos de parqueaderos por medio de un servicio web de mapas utilizando GeoServer; y desarrollar el visor geográfico con la librería OpenLayers. Como resultado se obtiene un aplicativo geográfico que permite visualizar los parqueaderos cercanos a un punto y consultar sus atributos (precio, dirección y uso).

El objetivo principal del proyecto fue crear un visor geográfico en donde se combinan el internet y algunas características de los Sistemas de Información Geográfica (SIG). Esta herramienta funciona en distintos navegadores y permite realizar funciones básicas de visualización como acercar, alejar, desplazar y consultar información geográfica almacenada en una base de datos. Basado físicamente en una máquina servidor, con una serie de programas para servicios Web encargado de alojar las páginas y servir la información espacial, recibir las peticiones de los clientes que quieran acceder a la información y dar respuesta a dichas peticiones bajo un esquema sencillo de consulta cliente-servidor a través del Internet. Es decir, un usuario solicita un mapa al Servidor Web y el Servidor pasa la solicitud al Servidor de mapas, quien entrega conjuntamente todos los datos como un mapa resultante que es devuelto al navegador Web del usuario. Una de las principales ventajas de un visor Web Geográfico como el que se propone, es la facilidad de su uso. A diferencia de un SIG, en donde el usuario requiere de un conjunto de conocimientos especializados, el usuario del visualizador no los requiere, ya que las bases de datos geográficas y de atributos ya están construidas para que el usuario las vea y las consulte.



**Figura 20. Esquema de Arquitectura Visor Geográfico de Parqueaderos**

**4.2.3 Sistema de información geográfica (SIG) para el diagnóstico integral de la actividad volcánica utilizando software libre.** El proyecto desarrolla un sistema de información geográfico (SIG), con el fin de proporcionar las herramientas necesarias para realizar el diagnóstico adecuado de la actividad volcánica en Colombia, en convenio con el Instituto Colombiano de Geología y Minas: INGEOMINAS, sede Manizales. Sin embargo, a pesar de existir un proyecto similar, éste presenta limitantes como: la necesidad de adquisición de software licenciado para su utilización, su valor elevado y su rendimiento regular, las cuales son superadas en este nuevo proyecto.

El hecho de tener un SIG con software libre, abre las puertas a la adquisición de información (de este tipo) y a la manipulación de la misma, además de realizar las modificaciones y novedades adecuadas para la topografía colombiana. Por tanto, el sistema presenta varios módulos de aplicación acordes para la evaluación de la actividad volcánica. Los módulos proporcionan interfaces y herramientas para el ingreso de la información al sistema, como la creación de reglas de diagnóstico en diferentes niveles, por ende, existe una interactividad total con la información, controlando así el acceso de cada uno de los usuarios, que sin duda alguna es uno de los aspectos más indispensables para el manejo actual de la información.

El sistema tiene como base de datos Postgresql en su versión 8.2, con la extensión Postgis que permite el manejo de sistemas de información. Como entorno gráfico, el sistema utiliza a MapServer; una de las novedades del sistema es la creación de un ítem para la generación de archivos. Esta interface permite la estructuración de gráficos de líneas y de barras, eligiendo dos parámetros por tabla, acotados por meses o por años. Otro punto importante es la inserción de datos al sistema, en la cual un archivo puede ser adjuntado al sistema para su utilización. El funcionamiento del sistema se llevó a cabo en la zona de Caldas (Nevado del Ruiz), pero perfectamente se puede almacenar información de otras zonas para su pertinente estudio. Sin embargo, el SIG se encuentra en sus primeras versiones, debido a que el ideal es la conectividad total del sistema con las máquinas recolectoras de información, implementando así la automatización de la información, siendo el usuario tan sólo un visitante al sistema.

#### **4.2.4 Diseño de una IDE, geoportal y aplicación web de denuncias adaptada a dispositivos móviles**

Este proyecto tiene varios objetivos como por ejemplo implementar una aplicación web, adaptada a dispositivos móviles que consiste en una pequeña red social para los ciudadanos, pero para los intereses del trabajo actual nos centraremos en el objetivo de desarrollar para la ciudad de Torrent de la comunidad de Valencia España, una infraestructura de datos espaciales (IDE) conforme a los estándares especificados por el Open Geospatial Consortium (OGC) y cuyo producto final y visible para el usuario consiste en un Geoportal que ofrece unos servicios y funcionalidades mediante los cuales el usuario puede interactuar con la Información Geográfica. La aplicación está concebida como una aplicación colaborativa, donde los propios usuarios alimentan la información haciendo uso la aplicación móvil.

Otro de los aspectos importantes de este trabajo fue el análisis de las tecnologías existentes para realizar el desarrollo, como es el caso de NodeJS, que corresponde a una plataforma diseñada para ejecutar programas escritos en JavaScript. Utiliza el motor V8 de JavaScript de Chrome, posee una arquitectura orientada a eventos con la utilización de un único hilo de ejecución (Thread) no bloqueante, lo cual le convierte en una plataforma ligera y eficiente. Además NodeJS posee un manejador de paquetes (Node Package Manager o NPM) que alberga el mayor ecosistema de desarrolladores y librerías de código abierto. Incluye en su núcleo una gran variedad de módulos (correspondientes al manejo de capas de bajo nivel) como los módulos http y https, que en concreto ayudan a realizar servidores HTTP y HTTPS.

Se destaca como principal característica y valor agregado de NodeJS su repositorio NPM, ya que en este se alojan módulos de otros desarrolladores que generalmente facilitan el trabajo y nos van a evitar escribir código que ya está escrito y testeado. Alguno de los módulos más importantes y famosos dentro de NPM es ExpressJS, el cual ayuda en la puesta en marcha y la creación de aplicaciones web.

Otro aspecto importante de NodeJS es que este se programa del lado del servidor, lo que resuelve el problema de "Cross Browser", que indica la necesidad en el lado del cliente de hacer código que se interprete bien en todos los navegadores. Cuando se trabaja con Node solamente es necesario preocuparse de que el código que se escriba se ejecute correctamente en el servidor.

NodeJS es por tanto un entorno realmente potente para la realización de aplicaciones web que generalmente consumen recursos del propio servidor (imágenes, contenido multimedia, etc.) o de otros servicios como una base datos local, servicios externos, etc.

## 5. METODOLOGÍA

### 5.1 TIPO DE TRABAJO

El trabajo corresponde a un desarrollo tecnológico, en el cual se integran tecnologías como HTML, ArcGIS Server, ArcGIS API for JavaScript, ArcGIS Server, Oracle y JQuery.

### 5.2 PROCEDIMIENTO

**5.2.1 Fase 1. Levantamiento de información.** Conocimiento del estado actual del sistema de información

- **Revisión del código fuente y documentación.** Se solicitó acceso al código fuente del proyecto de desarrollo Segovia, para conocer la funcionalidad actual de la aplicación. En esta revisión se encontró que desde el año 2012 hasta 2016, el modulo espacial no había sido evolucionado, sólo en el presente año se habían realizado algunas modificaciones necesarias para la migración de la versión del ArcGIS Server a la versión 10.3.1. En cuanto al código fuente se encontró que Segovia opera bajo el modelo MVC (Modelo Vista Controlador) figura (10), codificada en el lenguaje C# y haciendo uso de JavaScript y JQuery para el manejo de las interfaces y eventos, para construcción de los visores se identificó el uso de Silverlight.

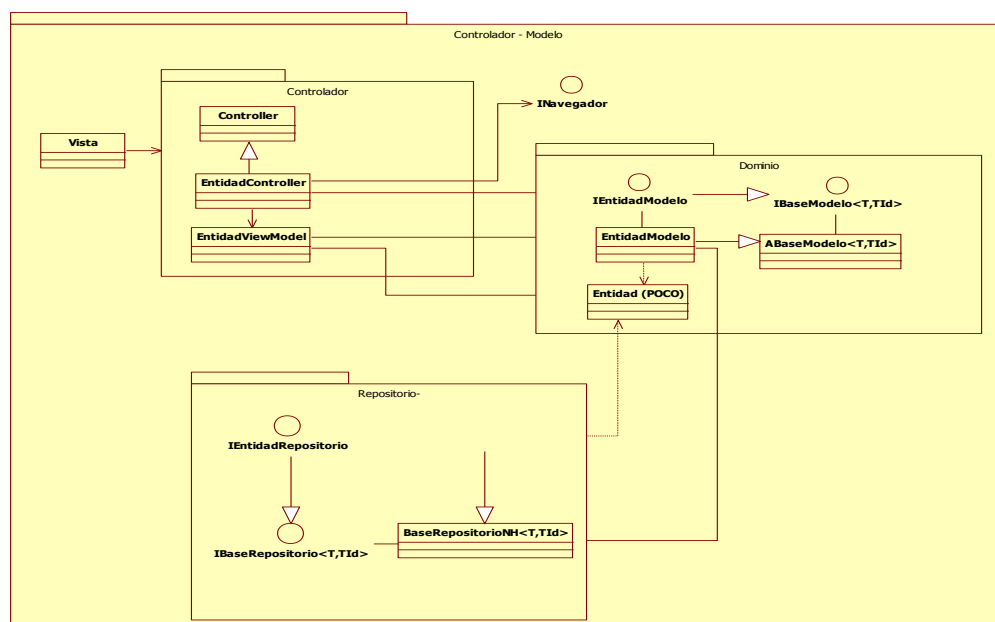
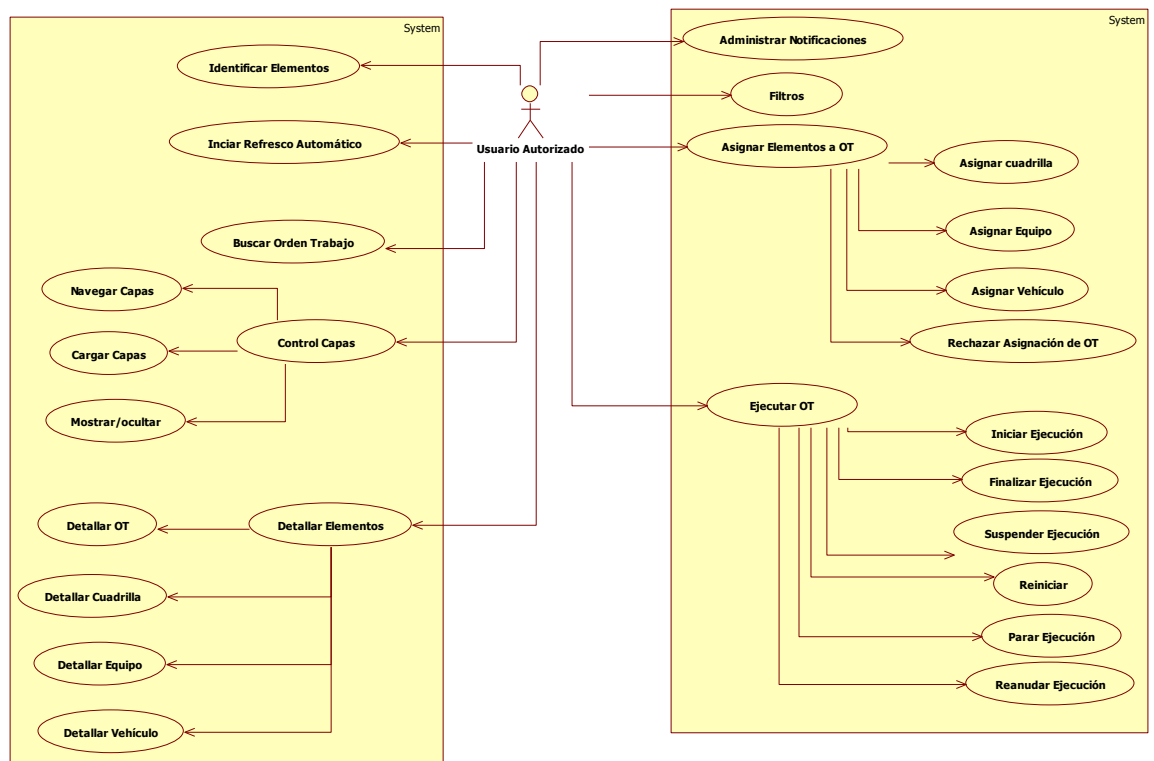


Figura 10. Modelo vista controlador Segovia

- **Revisión de conexiones y servicios.** Se revisaron las conexiones a bases de datos existentes y los diferentes servicios suministrados por ArcGIS Server. Con respecto a la información transaccional, Segovia realiza la conexión a la base de datos haciendo uso del ORM NHibernate, con el cual se conecta al motor Oracle. En cuando a la información espacial, hace uso del componente ArcSDE para realizar la conexión con la base de datos de Oracle.
- **Casos de Uso.** Se identifican los casos de uso pertenecientes al visor de asignación. Al lado izquierdo de la figura (11) se ubican los casos de uso correspondientes al alcance de este proyecto, los casos de uso ubicados al lado derecho de la imagen, son complementarios y corresponden a una futura fase de la migración.



**Figura 11. Modelo de Casos de Uso Visor Asignación**

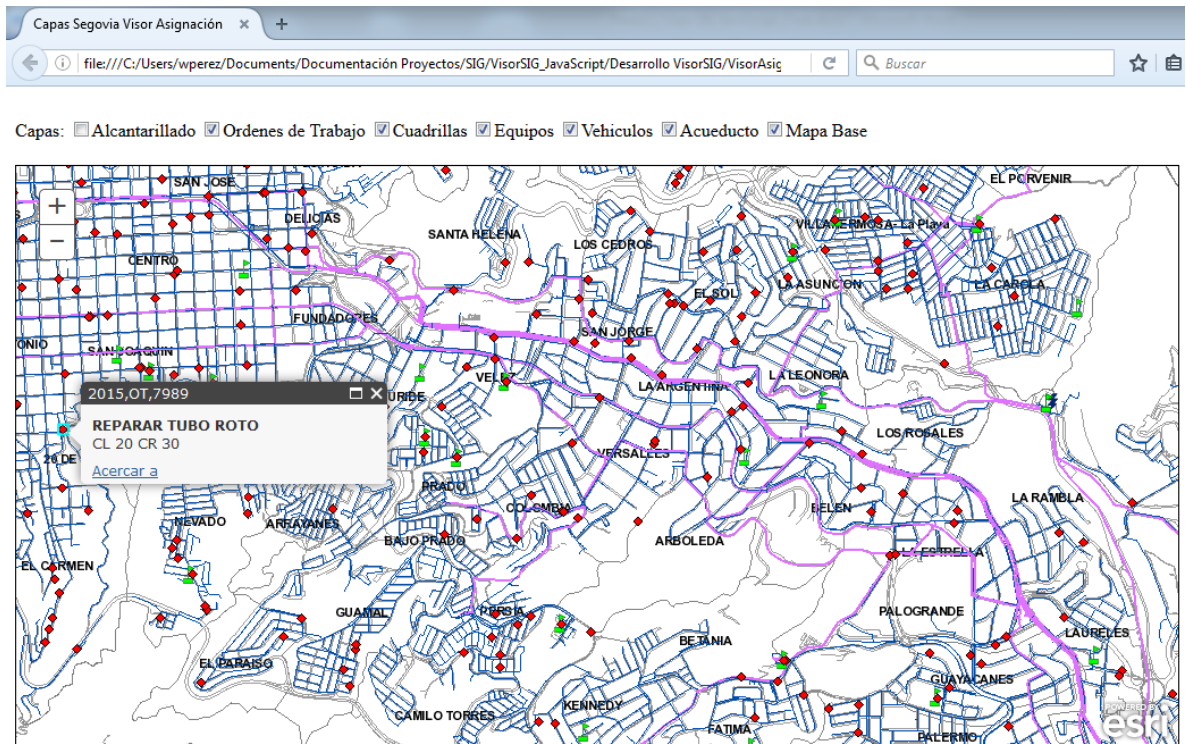
### 5.2.2 Fase 2 Implementación.

- **Documentación en el lenguaje JavaScript.** Revisión bibliográfica de uso de la herramienta Esri ArcGIS API for JavaScript. En esta etapa se hizo uso de revisión de foros y tutoriales en al web, sin embargo se obtuvo un mayor provecho revisando la documentación de la API que proporciona esri en su portal web, en dicho portal se encuentra la documentación oficial de cada uno de los componentes de la API y además cuenta con ejemplos prácticos de su uso, de los cuales se puede ver su código fuente y la simulación de su funcionamiento. Sin embargo no constituyen una metodología completa de implementación de un visor, corresponden a ejemplos aislados los cuales sirvieron como base o punto de inicio para el desarrollo.
- **Montaje del entorno de desarrollo.** Instalación y configuración de los lenguajes de programación como el editor de texto sublime, para la codificación, instalación del plugin Firebug para realizar seguimiento al código cuando se realizaban pruebas en el navegador web. Para la interacción con los datos, se hizo uso de un sitio de pruebas el cual se encontraba previamente configurado por el ingeniero SIG de la empresa.
- **Desarrollo del visor.** Se generó el código JavaScript y HTML en la API de ArcGIS.
- **Pruebas.** En esta etapa se realizaron pruebas en compañía del ingeniero SIG de la empresa, estableciendo comparativos con respecto al visor existente y operativo actualmente en producción, se revisan tiempos de respuesta, saturación del servidor y la calidad de la data.





que me permitiera levantar la ventana de información al picar sobre una orden de trabajo o sobre una cuadrilla y por ultimo crear el control que permitiera apagar o encender las capas de acuerdo a las necesidades del usuario, para esto se hizo uso de la librería DOJO y aunque no se aplicaron estilos, si se consiguió el objetivo de lograr la funcionalidad esperada Figura (7).



**Figura 7. Visor asignación en JavaScript Versión 2**

La versión final correspondió a la resolución del problema para construir los marcos del visor figura (2) y figura (3), para lo cual se contaban con dos opciones, la primera correspondía a crear los multis y formularios desde cero, haciendo uso de la librería DOJO, pero al realizar la revisión del código fuente se encontró un desarrollo complejo y grande, por tal razón se optó por la segunda alternativa la cual consistió en realizar la integración del nuevo visor con los marcos existentes figura (8).

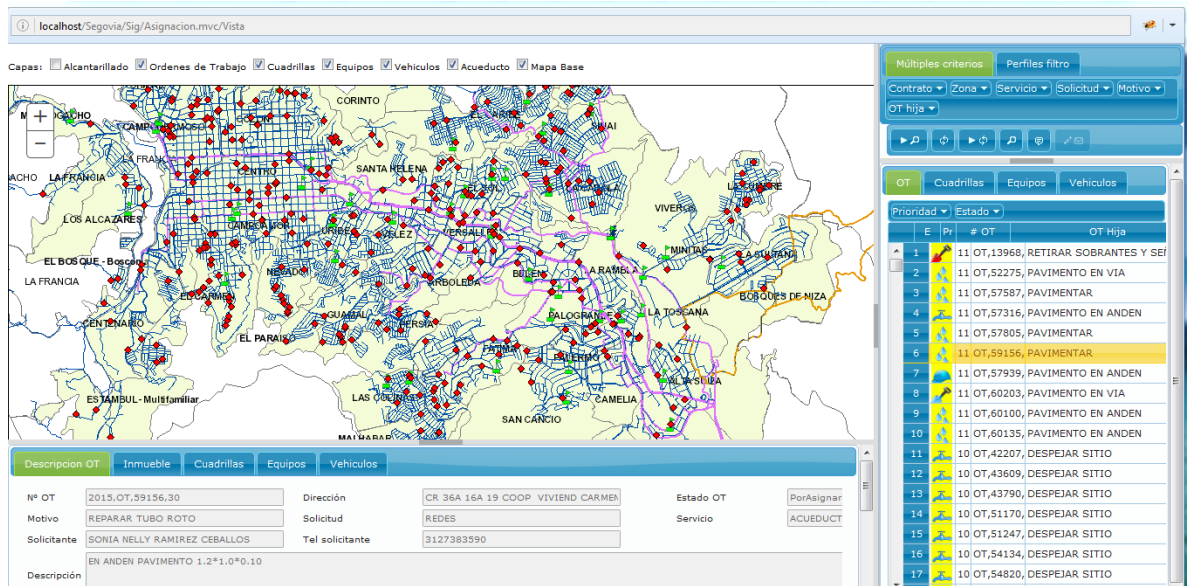


Figura 8. Visor asignación en JavaScript Versión Final

### 6.1.2 COMPARATIVO DE RESULTADOS

A continuación se realiza el comparativo de los resultados obtenidos con JavaScript versus el visor existente en Silverlight.

**Visor Completo:** El visor desarrollado en JavaScript tiene igual número de componentes, más una herramienta adicional para realizar consulta de ordenes de trabajo (Ot)

- **Silverlight**

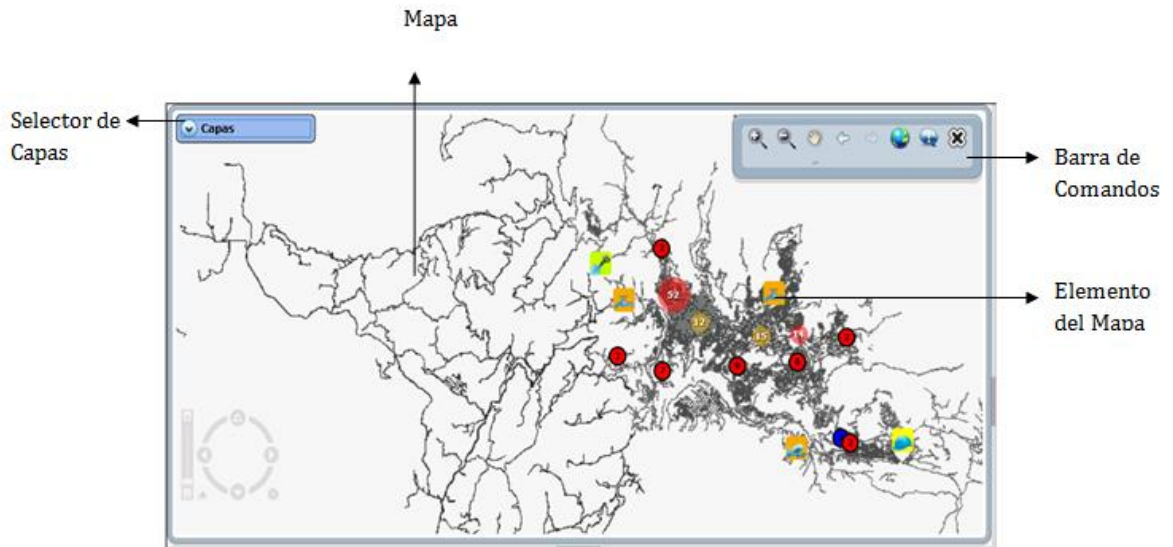


Figura 12. Descripción elementos visor Silverlight

- JavaScript

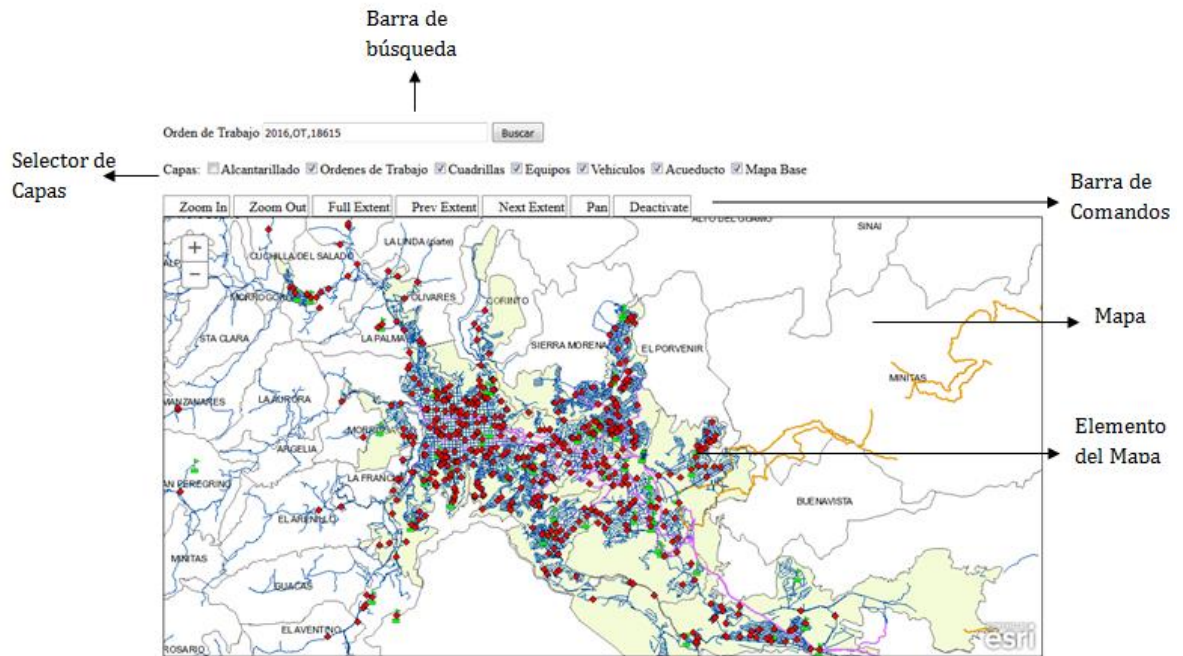


Figura 13. Descripción elementos visor JavaScript

**Selector de Capas:** Este componente permite mostrar u ocultar capas. En el comparativo se aprecia diferencias en diseño, pero con igual funcionalidad

- Silverlight

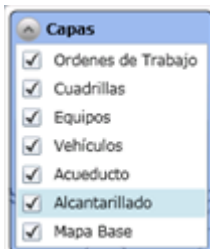


Figura 14. Selector de capas visor Silverlight

- JavaScript

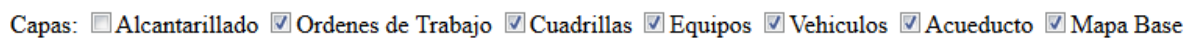


Figura 15. Selector de capas visor JavaScript

Capa	Descripción
Órdenes de trabajo	Si esta capa se encuentra seleccionada, se visualizarán los órdenes de trabajo que se encuentran georeferenciados.
Cuadrillas	Si ésta capa se encuentra seleccionada, se visualizarán las cuadrillas que se encuentran georeferenciadas.
Equipos	Si ésta capa se encuentra seleccionada, se visualizarán los equipos que se encuentran georeferenciados.
Vehículos	Si esta capa se encuentra seleccionada, se visualizarán los vehículos que se encuentran georeferenciados.
Acueducto	Si esta capa se encuentra seleccionada, se visualizará la red de acueducto de la ciudad de Manizales.
Alcantarillado	Si esta capa se encuentra seleccionada, se visualizará la red de alcantarillado de la ciudad de Manizales.

**Barra de comandos:** Este componente permite interactuar con el mapa y realizar operaciones sobre sus elementos.

- **Silverlight**

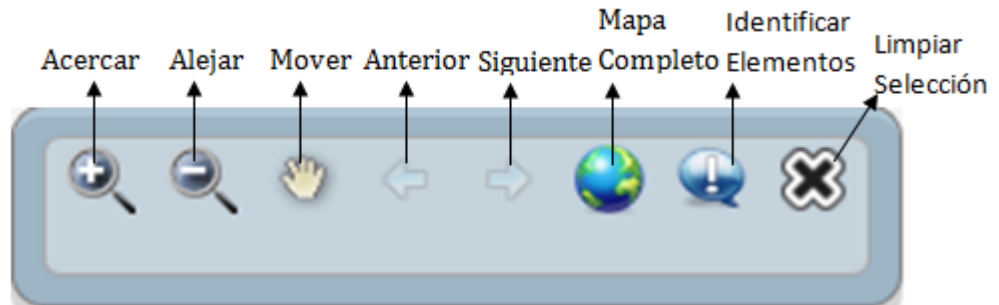


Figura 16. Barra comandos visor Silverlight

- **JavaScript**

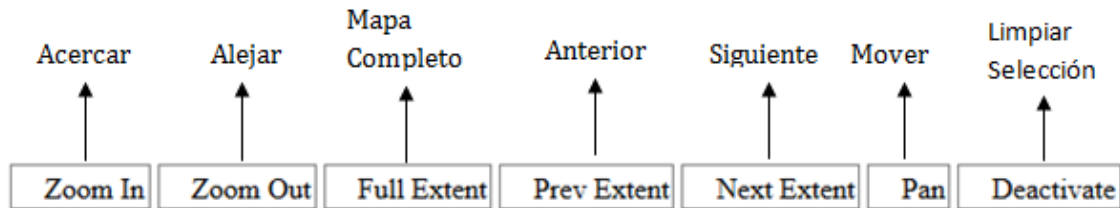
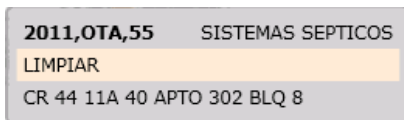


Figura 17. Barra comandos visor JavaScript

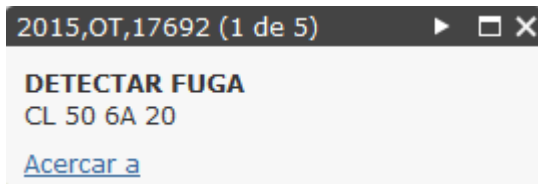
**Ventana emergente:** Al dar clic sobre un elemento ya sea en representación individual o en representación cluster, se despliega un cuadro con información de identificación de cada elemento, este cuadro es conocido en Silverlight como maptip y en ArcGIS API for JavaScript como infotemplate. Para ambos controles se tiene la información de: Número de la orden de trabajo, Tipo de solicitud asociada a la orden de trabajo y dirección.  
Dirección de la orden de trabajo

- **Silverlight**



**Figura 18. maptip visor Silverlight**

- **JavaScript**



**Figura 19. infotemplate visor JavaScript**

Con los resultados obtenidos se considera que al permitir la conexión a las bases de datos espaciales y transaccionales de Segovia, y realizar las consultas de la información así como el consumo de los servicios REST, se da cumplimiento de un 100% en el primer objetivo específico.

Con respecto al segundo objetivo específico, con la carga de las capas de acueducto, alcantarillado, vehículos, equipos, cuadrillas y ordenes de trabajo se da cumplimiento en un 100% a este objetivo.

Con respecto al tercer objetivo específico, se logra crear las diferentes herramientas de navegación e interacción con las capas, y se presenta un elemento adicional, el cual permite realizar consultas por orden de trabajo, sin embargo la funcionalidad de clustering sobre las capas no se ha implementado en la nueva versión del visor en JavaScript, lo que podría disminuir el porcentaje de cumplimiento a un 95% para este objetivo específico.

## 7. CONCLUSIONES

- El desarrollo de esta herramienta haciendo uso de la Api JavaScript de ESRI, permitió una fácil integración con los servicios REST(REpresentational State Transfer) proporcionados por el ArcGIS Server de la empresa Aguas de Manizales, así como con la tecnología JQuery, en la cual están contruidos los formularios y grillas de visualización.
- El visor de asignación de ordenes de trabajo es fundamental en el proceso de gestión de las labores de la empresa Aguas de Manizales en la ciudad, por lo tanto su migración a una tecnología como JavaScript es imprescindible para garantizar su continuidad y buen funcionamiento.
- En la migración se encontraron otras tecnologías coexistentes como Jquery la cual es utilizada para la creación y el manejo de las grillas y formularios que interactúan con el mapa, su acople es tan grande que se llegó a la conclusión que la mejor opción fue realizar la integración con ArcGIS API for JavaScript, en vez de su reconstrucción con otra herramienta orientada a interfaz gráfica.
- La empresa Aguas de Manizales cuenta con un volumen muy importante de información de ordenes de trabajo georreferenciadas, las cuales se han generado a partir del año 2012, esta información constituye un insumo valioso para la realización de análisis enfocados a los intereses de la empresa, pero que en la actualidad la funcionalidad con la que cuenta el visor no permite sacar mayor provecho.

## 8. RECOMENDACIONES

- Con la migración de el visor de asignación de órdenes de trabajo se tiene un parte de tranquilidad en cuanto a su funcionalidad y uso, sin embargo se encontraron versiones muy antiguas de las librerías JQuery, las cuales se recomienda actualizar o migrar a otras librerías más compatibles con ArcGIS API for JavaScript, como son DOJO o AngularJS.
- Se recomienda la revisión de los controles existentes en el visor, para determinar su verdadera utilidad, debido a que cada control tiene un consumo significativo en los servicios y por ende un impacto en el rendimiento de la aplicación.
- Para continuar con la evolución del proyecto a una nueva fase, se considera necesario el montaje de un entorno completo de desarrollo, el cual permita el manejo de versionamiento.
- Se encontró un gran potencial en el uso de arcGIS API for JavaScript, el cual puede ser aprovechado en la construcción de un visor con más funcionalidades, razón por la cual se recomienda un acercamiento con los usuarios finales para identificar posibles falencias que puedan ser suplidas ampliando el alcance del desarrollo actual.



## BIBLIOGRAFÍA

Cadena Enciso, J. Construcción de un modelo que determine la ruta óptima para el servicio de entrega de productos de la empresa “La Lechería”, en el municipio de Ubaté, Cundinamarca. Bogotá, 2015. Trabajo (Especialista en Geomática). Universidad Militar Nueva Granada, Facultad Ingeniería, Especialización en Geomática.

Castellanos Giraldo, L. and López López, M. Estimación de rutas y tiempos de respuesta de los organismos de socorro en la ciudad de Manizales, apoyado en herramientas SIG. Manizales, 2015. Tesis (Especialización en Sistema de Información Geográfica). Universidad de Manizales, Facultad de Ciencias e Ingeniería, Especialización en Sistemas de Información Geográfica.

Cerón González Frank Erwin, Liévano Idrobo Katherin. Aplicación web basada en SIG para la administración de la información catastral de las solicitudes recibidas por la unidad de restitución de tierras en el Valle del Cauca. Manizales, 2015. Trabajo de grado (Especialista en Sistema de Información Geográfica). Universidad de Manizales, Facultad de Ciencias e Ingeniería, Especialización en Sistema de Información Geográfica.

González Parra, D. Implementación de visor geográfico que permita ubicar parqueaderos. Bogotá, 2015. Trabajo de grado (Especialista en Geomática). Universidad Militar Nueva Granada, Facultad de Ingeniería, Especialización en Geomática.

Jose Ángel Herмосilla Rodrigo. Diseño de una IDE, geoportal y aplicación web de denuncias adaptada a dispositivos móviles. Valencia, 2016. Grado en Ingeniería Geomática y Topografía. Universidad Politécnica de Valencia, Facultad de Ingeniería, Máster En Ingeniería Geomática Y Geoinformación.

Quintero Valencia, A. and Restrepo Gómez, G. (2009). Sistema de información geográfica (SIG) para el diagnóstico integral de la actividad volcánica utilizando software libre. Manizales, 2009. Trabajo de grado (Ingeniería de Sistemas y Telecomunicaciones). Universidad de Manizales. Facultad de Ciencias e Ingeniería, Especialización en Sistemas de Información Geográfica

esri. (2016). ArcGIS API For JavaScript versión 317. Consultado 10 junio, 2016, from <https://developers.arcgis.com/javascript/3/> Disponible en: (Arcgiscom, 2016)

Mozillaorg. (2016). *Mozilla Developer Network*. Consultado 27 June, 2016, Disponible en <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Introducción> JavaScript: (Mozillaorg, 2016)

Microsoftcom. (2016). *Silverlight*. Consultado 27 June, 2016, Disponible en <https://msdn.microsoft.com/es-es/silverlight/bb187401.aspx> In-text citation: (Microsoft, 2016)

GEOWARN. Geospatial Warning System. ETH, Zurich. [En línea]. <http://www.geowarn.ethz.ch> , 2003.

VALENZUELA Carlos R. Curso SIG: Introducción a los Sistemas de Información Geográficos. COLOMBIA, MINISTERIO DE HACIENDA Y CRÉDITO PÚBLICO. INSTITUTO GEOGRÁFICO AGUSTÍN CODAZZI.; Bogotá, febrero de 1989. p. 65

Velarde Gutiérrez, Víctor. Desarrollo de aplicaciones geográficas web: OpenLayers, Curso de experto en desarrollo y gestión de sistemas de información geográfica (31-EX-75). Universidad de Cantabria. 2011.