

SISTEMA DE LOCALIZACIÓN EN TIEMPO REAL MEDIANTE UN SERVIDOR WEB Y APLICACIONES MÓVILES

Ignacio Contreras Contreras

Universidad de Guanajuato

i.contrerascontreras@ugto.mx

Geovanni Hernández Gómez

Universidad de Guanajuato

geov.hernandez@ugto.mx

Resumen

El sistema de transporte público del municipio de Yuriria es el método de transporte más utilizado por los estudiantes del Departamento de Estudios Multidisciplinario de la Universidad de Guanajuato. El sistema de transporte tiene un horario aleatorio y no existe una manera de conocer su ubicación. Los estudiantes deben de estar a tiempo para realizar sus actividades académicas es por eso por lo que es importante conocer la ubicación de los camiones de transporte. Con esta información los estudiantes pueden decidir si esperar al camión o buscar otro método de transporte. Detectado este problema se propone un sistema de localización de bajo costo en tiempo real mediante un servidor web y aplicaciones móviles. Este sistema utilizará el sistema de posicionamiento global (GPS por sus siglas en inglés) para obtener las coordenadas, latitud y longitud, del autobús. Utilizando el módulo GPS de un dispositivo móvil se puede crear una aplicación de rastreo de bajo costo. El sistema propuesto tiene como objetivo proveer una herramienta que pueda ser utilizada por los estudiantes de la Universidad y la población en general del municipio de Yuriria.

Palabra(s) Clave: Android, Aplicaciones móviles, GPS, Sistemas de información.

Abstract

The public transportation system of the municipality of Yuriria is the transportation method most used by students of the Department of Multidisciplinary Studies of the University of Guanajuato. The transportation system has a random schedule and there is no way to know its location. Students should be on time to perform their academic activities which is why it is important to know the location of the bus. With this information students can decide whether to wait for the bus or look for another method of transportation. Having detected this problem a low-cost GPS based tracking system with mobile applications and a web server is proposed. This system will use the Global Positioning System (GPS) to obtain the coordinates, latitude and longitude, of the bus. The proposed system aims to provide a tool that can be used by students of the University and the general population of the municipality of Yuriria.

Keywords: *Android, GPS, Information systems, Mobile applications.*

1. Introducción

El servicio de transporte público del municipio de Yuriria cuenta con siete rutas para brindar una cobertura amplia a una población de aproximadamente 70,000 personas [Municipio de Yurira, 2017] [INEGI, 2017]. Cada ruta cuenta con un horario, el cual los choferes de los camiones deben respetar. Los camiones no siempre pueden pasar en los horarios establecidos ya que se pueden encontrar con problemas de tráfico o mecánicos, los cuales pueden impedir el funcionamiento correcto del camión. Los choferes de los camiones no cuentan con un servicio para poder informarles a sus usuarios que el camión tiene problemas ya sea mecánicos o retraso de horario. Al no tener esta información importante los usuarios no pueden decidir con tranquilidad si esperar o utilizar otro método de transporte.

Al principio el uso del GPS era restringido para uso militar por el Departamento de Defensa de los Estados Unidos. En la década de 1980 se liberó para uso civil, aunque la señal era limitada y la exactitud no era la más óptima. El GPS se liberó por completo al público hasta el año 1995 al alcanzar la capacidad operacional

total con 24 satélites en órbita al mismo tiempo [Lee, 2017]. Al ser liberado al público, la tecnología del GPS fue utilizada de manera creativa para resolver varios problemas. Gracias a esto, el público identificó varias aplicaciones de esta tecnología e incrementaron su valor comercial. Con el incremento de vehículos en el mundo, el robo de autos aumentó, por lo cual fue necesario desarrollar distintos tipos de sistemas de localización vía GPS. Estos sistemas también sirven para tener un control de vehículos que transportan mercancía y optimizar las rutas que utilicen. Los sistemas de localización en tiempo real comerciales actualmente son muy caros, pueden estar restringidos a ciertas áreas comerciales, o en algunos otros casos, son muy complejos para utilizar. Los servicios de Fleetistics [Fleetistics, 2017], Advanced Tracking Technologies, Inc. [Advantrack, 2017], y Fleetmatics [Fleetmatics, 2017] se ajustan más a empresas que buscan optimizar y agilizar sus propios servicios de logística.

Los sistemas basados en Open Source tienen la desventaja de utilizar varios componentes como microcontroladores [Mistary, 2015] para poder alcanzar la funcionalidad propuesta, estos componentes pueden ser complejos y difíciles de utilizar para cualquier persona sin conocimientos en el área y requieren de un operador que este interactuando constantemente con el sistema [Al Rashed, 2013], otros no tienen la forma de compartir la ubicación de los vehículos con otras personas [Chadil, 2008] [Gaikwad, 2015], generando una desventaja al momento de querer optimizar los recursos.

Como justificación a este proyecto cabe mencionar que el sistema de transporte público utilizado por los alumnos, el personal académico y administrativo de la Universidad de Guanajuato Sede Yuriria, se encuentra regulada dentro de un horario aleatorio. Esto genera una problemática para los usuarios que necesitan cumplir con un horario de entrada y salida definido tanto para clases de los alumnos como para las horas laborales de los trabajadores. Para muchos usuarios, el sistema de transporte público ofrecido por el Ayuntamiento del municipio de Yuriria es el único método de transporte disponible para cumplir con sus obligaciones.

En la actualidad, las personas viven un ritmo de vida muy apresurado, especialmente los estudiantes que se trasladan de sus hogares a sus respectivos centros educativos. El beneficio de poder saber la ubicación del autobús que deben abordar les brindaría un sentido de seguridad y tranquilidad para administrar mejor sus tiempos y sus actividades. Actualmente, no se encuentra implementado ningún sistema de ubicación del autobús y los sistemas modernos establecidos en otros estados, se encuentran fuera de un alcance comercial para ser implementados en el municipio de Yuriria. Debido a esto se observa que es necesario realizar un sistema que permita monitorear la ubicación del transporte público a manera de optimizar el servicio en transporte público. Con esto se pretende lograr alcanzar un servicio que sea útil tanto para toda la comunidad universitaria que labora en la Sede Yuriria como los usuarios de la misma ciudad. El sistema propuesto en este trabajo fue diseñado, desarrollado y al ponerse en marcha puede tener un impacto positivo que será discutido en las siguientes secciones.

2. Metodología

Aplicación de rastreo (primera versión)

La primera versión de la aplicación de rastreo se desarrolló en MIT App Inventor, una plataforma para crear aplicaciones móviles en un navegador web [MIT App Inventor, 2017]. Esta versión se desarrolló como una prueba de concepto y para identificar un dispositivo móvil con alta precisión del módulo GPS. La aplicación desarrollada en MIT App Inventor consiste en obtener las coordenadas, latitud y longitud, del autobús mediante el sistema de posicionamiento global y guardarlas en una lista. El desarrollo de aplicaciones móviles en MIT App Inventor consiste en dos partes, una parte donde se eligen los componentes a utilizar y se define el diseño de la aplicación, y la otra parte denominada como el editor de bloques es donde se define el comportamiento de los componentes. Después de cada recorrido se visualizan las coordenadas guardadas en un mapa y se analiza el rendimiento del módulo GPS del dispositivo móvil:

- Diseñador de la aplicación. En esta parte se eligen los componentes esenciales para que la aplicación cumpla con el objetivo. En la figura 1 se identifican los componentes elegidos para crear la aplicación:
 - ✓ Etiquetas: Se seleccionan tres etiquetas de la sección de interfaz de usuario. Estas etiquetas son utilizadas para identificar la información que se obtiene del sensor de ubicación correspondientes a la latitud, la longitud, y la exactitud de estas coordenadas.
 - ✓ Cajas de texto: Se seleccionan tres cajas de texto de la sección de interfaz de usuario. Las cajas de texto son utilizadas para mostrar la información obtenida del sensor de ubicación. Esta información corresponde a las coordenadas, latitud y longitud, y su exactitud. Estas cajas de textos son identificadas con una etiqueta que las precede.

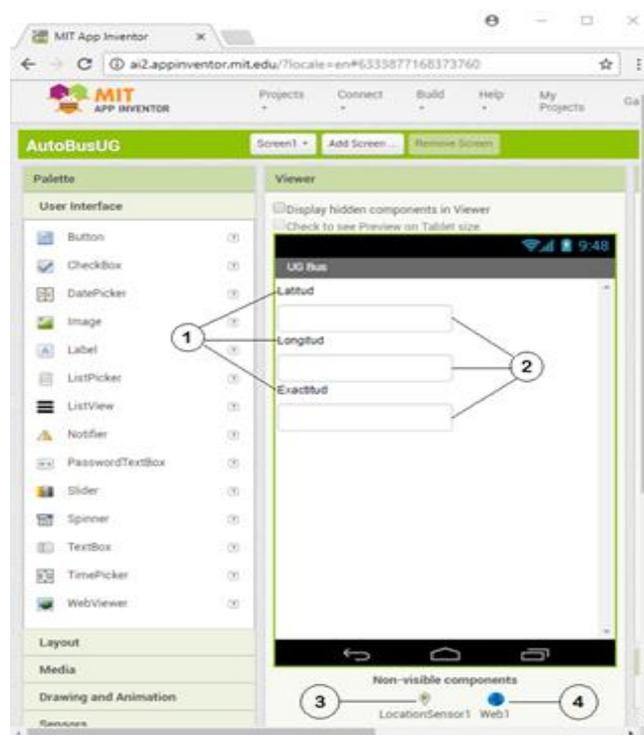


Figura 1 Pestaña “Diseñador” donde se muestran los componentes utilizados.

- ✓ Sensor de ubicación: Se selecciona un sensor de ubicación de la categoría de sensores. Este sensor se utiliza para que la aplicación se pueda comunicar con el receptor de GPS que se encuentra en el

dispositivo móvil. Una vez que el sensor de ubicación se comunica con el receptor de GPS, este puede la ubicación del dispositivo. El comportamiento de este sensor se define en el editor de bloques.

- ✓ Web: Este componente provee las funciones para realizar peticiones HTTP como GET, POST, PUT y DELETE. Se utiliza este componente dentro de la aplicación para comunicarse con un servidor y así guardar las coordenadas, latitud y longitud, obtenidas del autobús durante un recorrido. Estas coordenadas se guardan en una lista para posteriormente ser analizadas.
- Editor de bloques. Al tener todos los componentes seleccionados y la interfaz de usuario definida se pasa al editor de bloques para definir su comportamiento dentro de la aplicación. El funcionamiento se divide en dos bloques de código esenciales. El Bloque A de código que verifica si el sensor de ubicación está habilitado y el Bloque B de código que controla el comportamiento del sensor de ubicación y envía las coordenadas (latitud y longitud) al servidor web se muestran en figuras 2 y 3, respectivamente.

```
if LocationSensor1.Enabled then
  set LatitudValor.Text to LocationSensor1.Latitude
  set LongitudValor.Text to LocationSensor1.Longitude
  set ExactitudValor.Text to LocationSensor1.Accuracy
  set LocationSensor1.DistanceInterval to 10
  set Screen1.BackgroundColor to blue
```

Figura 2 Bloque A.

```
when LocationSensor1.LocationChanged
  do
    if LocationSensor1.Accuracy > 1
    then
      set Web1.Url to join "http://figurug.ddns.net/gps/get.php?lat="
        LocationSensor1.Latitude
        &long=
        LocationSensor1.Longitude
      set LatitudValor.Text to get latitude
      set LongitudValor.Text to get longitude
      set ExactitudValor.Text to LocationSensor1.Accuracy
      set Screen1.BackgroundColor to green
      call Web1.Get
    else
      set LatitudValor.Text to get latitude
      set LongitudValor.Text to get longitude
      set ExactitudValor.Text to LocationSensor1.Accuracy
      set Screen1.BackgroundColor to red
```

Figura 3 Bloque B.

A continuación, se define la lógica de cada bloque:

- ✓ Bloque A: Este bloque de código se utiliza para comprobar que el sensor de ubicación está habilitado. Si el sensor de ubicación está habilitado entonces el contenido de las cajas de texto correspondientes a la latitud, longitud, y exactitud es actualizada. El contenido de las cajas ahora corresponde a los datos obtenidos del sensor de ubicación los cuales son las coordenadas, latitud y longitud, y la exactitud. Posteriormente se define el parámetro de intervalo de distancia del sensor de ubicación con el número 10. Este parámetro define una distancia en metros, la cual se debe recorrer para obtener una nueva ubicación. Se decidió utilizar la distancia de 10 metros para obtener una mayor cantidad de coordenadas, ubicaciones dentro del recorrido del autobús. Finalmente se cambia el color de la pantalla al color azul para señalar al usuario que el sensor de ubicación está habilitado.
- ✓ Bloque B: Este bloque de código se utiliza para definir el comportamiento de los componentes cuando el sensor de ubicación obtiene una nueva ubicación. Este bloque de código se activa después del bloque A, donde se comprueba si el sensor de ubicación está habilitado. Una vez que el sensor de ubicación está habilitado, la aplicación comienza a obtener la ubicación del dispositivo móvil. El bloque B funciona de la siguiente manera. Cuando el sensor de ubicación detecta que la ubicación del dispositivo móvil ha cambiado entonces ejecuta el código dentro del bloque B. Si la exactitud del sensor de ubicación es mayor a 1 entonces se manda la ubicación al servidor. Para mandar la ubicación al servidor la propiedad URL del componente Web es actualizado con una dirección de red, en este caso la dirección del servidor. Este URL contiene como parámetros la latitud y longitud de la nueva ubicación. Se actualizan las cajas de texto con sus datos correspondientes y el color de la pantalla se cambia a verde, símbolo de que la obtención de una nueva ubicación

fue correcta. Finalmente, para que se guarden las coordenadas en el servidor se hace una llamada al método GET del componente Web, el cual hace una petición GET a la dirección de red previamente definida. Se utilizó el método GET ya que en este punto del desarrollo no se contaba con una base de datos para guardar las coordenadas con el método POST. El método GET permite obtener las coordenadas de la URL fácilmente y posteriormente guardarlas en un archivo de texto. En caso de que la exactitud obtenida del sensor de ubicación sea menor que 1 entonces sucede lo siguiente. Las cajas de texto correspondientes a las coordenadas, y la exactitud son actualizadas con la nueva ubicación obtenida. En este caso la ubicación no se envía al servidor y el color de la pantalla se cambia a rojo.

Aplicación de rastreo (segunda versión)

La segunda versión de la aplicación de rastreo se desarrolló en Android Studio, un entorno de desarrollo integrado para crear aplicaciones móviles para dispositivos Android [Android Studio, 2017]. MIT App Inventor al ser un programa donde la creación de aplicaciones móviles que consiste en encajar bloques para controlar el comportamiento de los componentes limita la personalización de los componentes y en consecuencia el funcionamiento de la aplicación. Con Android Studio se logra crear aplicaciones móviles a un nivel de programación más bajo, con código y no bloques, lo que significa que se puede modificar el comportamiento de los componentes y así aumentar el rendimiento de la aplicación.

La versión desarrollada en Android Studio está compuesta por tres actividades:

- En la actividad principal (figura 4a), el chofer puede elegir una ruta, que recorre un camión, para rastrear, agregar una nueva ruta para rastrear, o actualizar las rutas disponibles en el sistema.

- La segunda actividad (figura 4b), permite agregar una nueva ruta al sistema.
- En la tercera y cuarta actividad, (figuras 4c y 4d), el chofer inicia y detiene el servicio de rastreo.

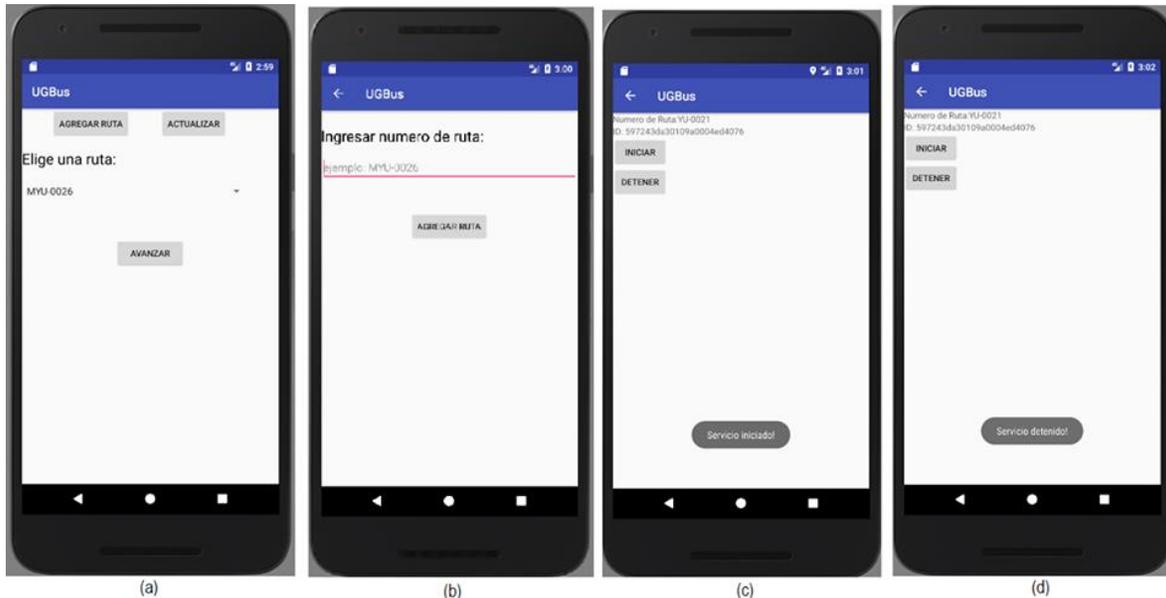


Figura 4 (a) Actividad principal. (b) Agregar ruta. (c) Iniciar servicio. (d) Detener servicio.

Descripción del funcionamiento de cada actividad:

- Actividad Seleccionar Ruta: Esta actividad se muestra en la figura 4a, funciona como la actividad principal de la aplicación de rastreo. El propósito principal de esta actividad es seleccionar una ruta del sistema, para pasar a la actividad donde se inicia el rastreo dando clic en el botón Avanzar. En esta actividad también se puede acceder a la actividad de agregar una ruta dando clic en el botón Agregar Ruta. Con el botón Actualizar el chofer hace una petición al servidor para obtener de nuevo la lista de rutas disponibles.
- Actividad Agregar Ruta: La actividad para agregar una ruta se muestra en la figura 4b. Esta actividad le permite al chofer agregar una nueva ruta que para poder seleccionarla e iniciar su rastreo en caso de que no exista en el sistema. El chofer ingresa un número de ruta en la caja de texto y procede

a dar clic en el botón de Agregar Ruta. La aplicación manda una petición al servidor y la agrega al sistema.

- Actividad Iniciar Rastreo: La actividad para iniciar y detener el rastreo de la ruta se muestra en la figura 4c. Al dar clic al botón de Iniciar el chofer inicia el servicio de rastreo. Se decidió utilizar un servicio de Android para mandar la ubicación al servidor web, en intervalos de tiempo de ocho segundos, ya que este tiene la ventaja de no destruirse o parar al cerrar la aplicación o bloquear el dispositivo móvil. Si el chofer da clic en el botón Detener, el servicio se detiene y deja de enviar las coordenadas, latitud y longitud, al servidor web correspondientes a la ruta que se esté rastreando.

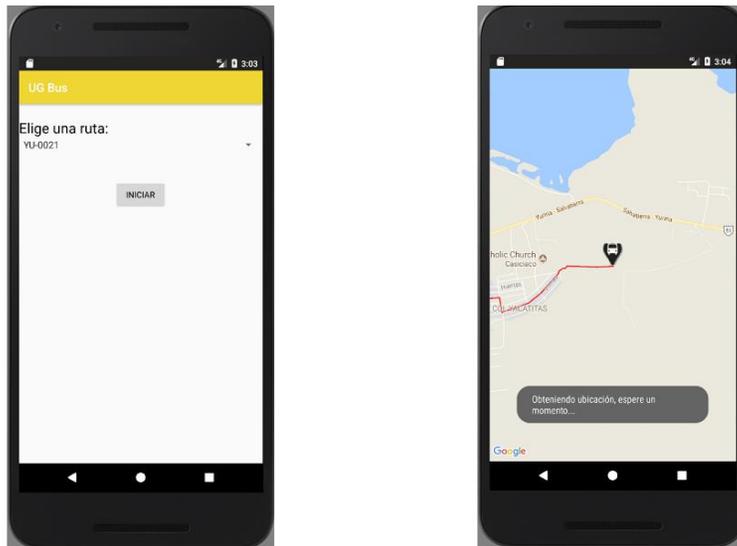
Consideraciones adicionales

Una vez que se verifico que es posible crear una aplicación de rastreo de bajo costo, reutilizando dispositivos móviles descartados se pasa a la siguiente etapa, la construcción de un servidor web [Mozilla, 2017]. Se decidió desarrollar el servidor web con la pila de desarrollo MongoDB, Express, AngularJS, y NodeJS. Se utilizó un servidor web para guardar las coordenadas del camión, enviadas por la aplicación de rastreo y posteriormente enviarlas a los pasajeros que deseen conocer la ubicación.

Para que los pasajeros puedan ver la ubicación del autobús en tiempo real se desarrolló una segunda aplicación móvil. Esta aplicación le permite al pasajero elegir una ruta para consultar su ubicación. En la figura 5 se muestra la interfaz de esta aplicación y su funcionamiento es el siguiente:

- Una vez que el pasajero selecciona una ruta (figura 5a), la aplicación se conecta con el servidor web y hace una petición para obtener los datos correspondientes a la ruta. El objeto spinner se llena con las rutas disponibles y le permite al pasajero seleccionar una. Al seleccionar una ruta el pasajero da clic en el botón de Avanzar y se envía el id del autobús el cual se utiliza para hacer peticiones al servidor web.
- Al obtener los datos de la ruta (figura 5b), la aplicación muestra un mapa, obtenido de Google Maps, con el recorrido en rojo y un marcador en las

coordenadas de la ubicación del autobús. La aplicación hace peticiones al servidor web en intervalos de 10 segundos para así tener la ubicación del autobús más reciente.



a) Actividad principal.

b) Ubicación del dispositivo rastreado.

Figura 5 Interfaz de la aplicación.

3. Resultados

Resultados: Primera versión MIT App Inventor

En la figura 6 se muestra un mapa con las coordenadas obtenidas con la primera versión de la aplicación de rastreo desarrollada con MIT App Inventor. Como se puede observar los resultados no son los más óptimos ya que aún se pueden observar partes de la ruta donde no se obtuvo una coordenada, donde no hay un marcador. La velocidad del autobús es un factor que impacta el rendimiento de esta versión. En los tramos donde el autobús acelera y alcanza velocidades altas, la aplicación no logra obtener una ubicación nueva hasta que el autobús desacelere la velocidad. Esta versión al momento de bloquear el dispositivo móvil, la conexión al servidor web se pierde para que el dispositivo móvil ahorre batería. Es por esto por lo que se decidió continuar el desarrollo en Android Studio. En Android Studio se desarrolla una segunda versión que resuelve estos problemas.



Figura 6 Resultados de la primera versión desarrollada en MIT App Inventor.

Resultados: Servidor web

Uno de los beneficios de utilizar un servidor web es que se puede desarrollar una aplicación web en conjunto. En la figura 7 se muestra la interfaz de esta aplicación web. En la aplicación web se muestran las rutas disponibles y los pasajeros eligen la ruta que deseen saber su ubicación. Al elegir una ruta se direcciona a un mapa de Google Maps con su ubicación. En la figura 8 se muestra un ejemplo de este mapa. Con el botón de refrescar se muestra en la figura 8 el usuario obtiene la ubicación más reciente. La aplicación web permite una cobertura mayor ya que por el momento sólo se cuenta una aplicación para el sistema operativo Android. Con la aplicación web los pasajeros que no cuenten con la aplicación móvil pueden acceder a través del navegador web de Google Chrome.

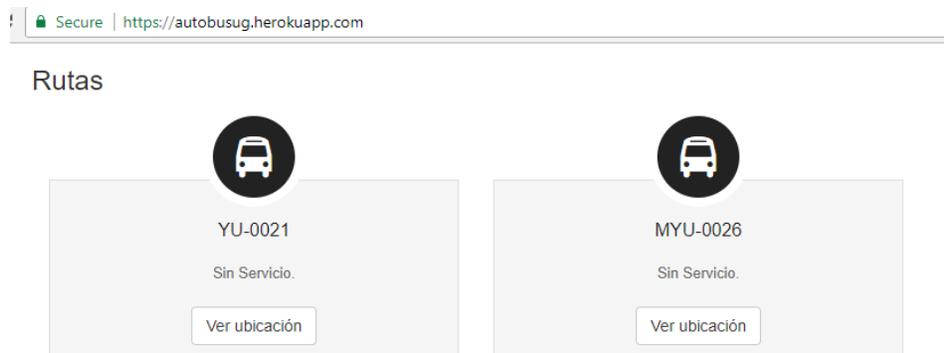


Figura 7 Interfaz de la aplicación web.

Resultados: Sistema completo

En las figuras 9 y 10 se muestra el recorrido de un autobús que va desde el Departamento de Estudios Multidisciplinarios hacia la colonia "La Aldea". Este

recorrido tiene un tiempo total de 30 minutos así que para mostrar todo el recorrido completo se dividió en intervalos de 2 minutos.



Figura 8 Mapa que muestra la ubicación del dispositivo rastreado en la aplicación web.



Figura 9 Primera parte de un recorrido del autobús.

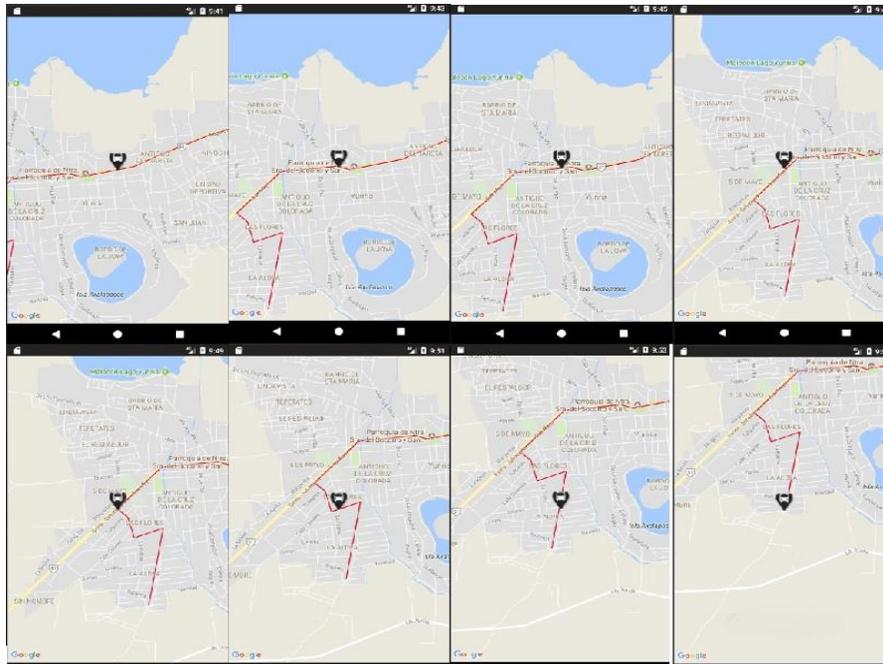


Figura 10 Segunda parte de un recorrido del autobús.

El recorrido inicia del cuadro superior izquierdo de la figura 9 y avanza hacia la derecha, y termina en el cuadro inferior derecho de la figura 10. Este recorrido se logró con las tres partes del sistema trabajando en conjunto: la aplicación de rastreo que envía las coordenadas al servidor web, y la aplicación que visualiza estas coordenadas en un mapa.

4. Discusión

Se logró el objetivo de crear un sistema de localización para el transporte público del municipio de Yuriria. Este sistema de localización está compuesto de dos aplicaciones móviles y un servidor web. La primera aplicación móvil utiliza el módulo GPS del dispositivo móvil donde está instalada para obtener las coordenadas, latitud y longitud, del autobús y enviarlas al servidor web donde se guardan. La otra aplicación móvil la utilizan los pasajeros para consultar la ubicación de un autobús y visualizarla en un mapa.

Es importante mencionar que la aplicación de visualización sólo puede mostrar un autobús a la vez. El servidor web aparte de guardar y enviar los datos de los camiones también sirvió para crear una aplicación web. Esta aplicación web tiene

la misma funcionalidad de la aplicación móvil que visualiza la ubicación del autobús. Con la aplicación web los pasajeros que no cuenten con un dispositivo móvil Android pueden consultar la ubicación del autobús a través de Google Chrome. Con la aplicación web se incrementa el número de personas que se pueden beneficiar del sistema. Se obtuvo una aplicación de rastreo con consumo de datos considerablemente bajo ya que consume aproximadamente de 4 a 6 megabytes en una jornada laboral de 8 horas. El gasto de los datos será cubierto por los choferes que opten por utilizar el servicio. En cuanto al consumo de energía la aplicación de rastreo consume aproximadamente un 30% de una batería completamente recargada.

El sistema propuesto funciona y muestra la ubicación del camión a los pasajeros, pero aún no se tiene una medida para comprobar que tan útil es. Como trabajo a futuro se realizarán encuestas a los pasajeros para ver qué impacto ha generado el sistema con respecto a la espera del transporte público. También se presentará el sistema y se aplicará una encuesta a los dueños y choferes de los camiones para ver si están de acuerdo con que sus unidades sean rastreadas.

5. Bibliografía y Referencias

- [1] Municipio de Yuriria. (2017). Tránsito y Transporte: <https://goo.gl/mKG5bJ>.
- [2] INEGI. (2017). México en Cifras, Áreas geográficas e indicadores del municipio de Yuriria, Guanajuato: <https://goo.gl/tFwzE1>.
- [3] Lee, D. (2017). Garmin | What is GPS? Obtenido de <https://goo.gl/LcwM5r>
- [4] Fleetistics. (2017). GPS Vehicle Tracking: <https://goo.gl/fwsLqt>.
- [5] Advantrack. (2017). GPS Vehicle Tracking System: <https://goo.gl/K1XMGv>.
- [6] Fleetmatics. (2017). GPS Tracking Solutions: <https://goo.gl/r7VrcA>.
- [7] Mistary, P. V & Chile, R. H. (2015). Real time Vehicle tracking system based on ARM7 GPS and GSM technology. India Conference 1-6: <https://goo.gl/FSD6QF>
- [8] Al Rashed. (2013). A real time GSM/GPS based tracking system based on GSM mobile phone. Future Generation Communication Technology. 65-68.
- [9] MIT App Inventor. (2017): <https://goo.gl/dxiYhd>.

- [10] Chadil. (2008). Real-time tracking management system using GPS, GPRS and Google earth. ECTI-Con 2008. 393-396
- [11] Gaikwad. (2015). Implementation of real-time GPS receiver system for providing navigation based services and SMS tracking. *Industrial Instrumentation and Control*. 630--634
- [12] Android Studio The Official IDE for Android (2017): <https://goo.gl/j4PqwR>.
- [13] Mozilla. (2017). What is a web server? de Mozilla: <https://goo.gl/TjPonr>.
- [14] GPS.gov. (2017). What is GPS?: <https://goo.gl/nygp2x>.