OpenBus Navigator

P. Hernandez,S. Leal, G. Marin
Escuela de Ingeniería Eléctrica
Universidad de Costa Rica, San José, Costa Rica
maria.hernandezgochez@ucr.ac.cr,stuart.leal@ucr.ac.cr, giancarlo.marin@ucr.ac.cr

***Resumen*—The following article presents the development of an application that improves the experience of traveling by bus, using an innovative algorithm to automatically determine bus stops and routes with data acquired through information provided by the user's cellphone sensors, making traveling by bus a flexible and dynamic system by avoiding direct communication with any bus company or the Public Transport Council. The information generated through the platform can be useful for numerous tasks, from evaluating the bus routes quality, to facilitate the most convenient route for an user.**

***Palabras clave*—Algorithm, Bus Routes, Bus Stops, GPS, Open Source.**

I. Introducción

En Costa Rica se cuenta con múltiples formas de transporte público, como lo son taxis, trenes y buses. El transporte mediante bus es ampliamente usado por la población, ya que se estima que alrededor de 1,1 millones de personas ingresan a la capital a diario utilizando este medio.[5]
Sin embargo, la deficiencia de información con respecto a rutas y paradas del transporte público provocan molestias entre los usuarios por la pérdida de tiempo y dinero que esto representa, teniendo en ocasiones que tomar otros tipos de transporte o desplazarse en rutas sin el conocimiento de la ubicación de las paradas.
Una de las maneras en que se acostumbra solventar estas dudas por parte de los usuarios es consultándole al chofer del bus u a otras personas, viendose expuestos a que esta información pueda ser errónea. Con esto surge la pregunta: ¿De qué manera se podría obtener los datos de las rutas y paradas de los buses de transporte público en tiempo real y desde cualquier sitio con el uso de internet?
Es por esto que se plantea una solución en el cuál se tiene como fin mejorar la información proporcionada al usuario final sobre las paradas oficiales, extraoficiales y rutas proporcionadas según su posición y destino, así como la ruta del bus en tiempo real, con una proyección a determinar los horarios por los que cierto bus se localiza en un punto de la ruta sin la utilización de equipos físicos adicionales alojados en el bus.

II. Estado de la cuestión

En Costa Rica los intentos por solventar el problema descrito han sido abarcados parcialmente, por la aplicaciones como Çazadora UCRz "Busmaps", ambas disponibles en el App Store de Android. Sin embargo, estas aplicaciones poseen horarios y paradas fijas, ya que toman los datos directamente de los servicios de autobuses, pero estos datos suelen no ser fidedignos ya que no toman en cuenta la variabilidad de horas que se da, por el tráfico en horas de alta demanda. Junto con esto está su alta dependencia de las empresas de transporte ya que la base de datos de ambas está basado en lo proporcionnado por las respectivas empresas de buses.

Además, la obtención de esta información del gobierno presenta múltiples desventajas debido a la ineficiencia en brindar esta información en ocasiones pasadas, la rigidez que esto provoca en la aplicación al necesitar ajustes si existe un cambio en la ruta del bus y la incapacidad de brindar información de lo que realmente sucede con los horarios de los buses.

III. Objetivos

Objetivo general

- Desarrollar un algoritmo que determine donde se encuentran las paradas y cuál es la ruta de manera automática, para poder mostrar estos datos a los usuarios mediante una aplicación.

Objetivos específicos

- Desarollar un algoritmo para la determinación de paradas y rutas
- Implementar un servidor para la interacción del usuario con los datos
- Programar una aplicación en Android que sea un cliente básico del algoritmo alojado en el servidor

IV. Descripción del Proyecto

El proyecto consiste en mejorar la experiencia del uso de los buses de transporte público en Costa Rica mediante el desarrollo de un algoritmo para la determinación automatizada de paradas y rutas; utilizando una aplicación, con el fin de que los usuarios alimenten una base de datos encontrada en un servidor y puedan recibir los datos correspondientes a las paradas y desplegarlos en un mapa real donde observen la ruta y sus paradas.

V. Metodología

Para el desarrollo de este proyecto, se dividirá en las siguientes etapas :

Etapa I. Desarrollo e implementación del algorítmo de localización de paradas:

Se desarrollará un algoritmo para analizar cada uno de los puntos que se encuentran dentro de cada archivo gpx generado con información tomada del sensor GPS de un teléfono android. Si una persona se sube al bus en una parada determinada, o se baja en otra, en general hay un cambio en la cantidad de puntos antes y después de cada parada. Este cambio es determinado por el algoritmo, y define un parada en el punto donde existe dicho cambio.
Al finalizar, con la información de la ubicación de cada parada, se genera un archivo gpx de vuelta, que contiene la información de cada parada. Esta información puede ser muy amplia, por ejemplo los horarios promedio de los buses (basados en el tiempo en el que se tomo cada dato). El código de este algoritmo será construido alrededor de la plataforma Github con dos objetivos: el primero tener un control de versiones adecuado, y el segundo poder compartir este código y promover el trabajo de nuevos miembros a la aplicación.
Se utilizará Python en el desarrollo del algorítmo por la simpleza de la modelación en este lenguage, y las librerías ya existentes.
Python [3]

Etapa II. Comunicación cliente/servidor
Se desarrollará un servidor en Python, utilizando un protocolo de comunicación TCP/IP mediante sockets. El servidor va a contar con dos tipos de solicitudes, una en la cual el cliente va a solicitar las paradas que desea y otro en el que se le indica al servidor que proporcione información de la base de datos.
Al recibir la solicitud de paradas el servidor busca la carpeta generada por el algoritmo en la cual se encuentren los archivos de las con la información de cada punto de las paradas y las envía al cliente.
Por el contrario, si el cliente indica que desea alimentar la base de datos el servidor, mantiene la conexión abierta, recibiendo datos hasta que encuentre una indiación de escape en los datos recibidos.

Etapa III. Desarrollo de la aplicación
Se desarrollará una aplicación en android que posee dos funciones principales: Primero la recolección de los datos mediante el uso del GPS del teléfono apenas se inicializa la aplicación hasta que el cliente lo detenga o por defecto ella decida no tomar más datos y luego de esto envía un archivo GPX al servidor donde se aloja el algoritmo con todos los datos de "*waypoints*"tomados. Segundo, la aplicación solicitará los datos para una ruta determinada y recibirá del servidor un archivo GPX con la ruta con sus respectivas paradas, para así desplegarla sobre un mapa utilizando una conexión entre el app y OpenStreetMaps. El desarrollo será en python mediante las librerías de kivy [2], esto en la fase de prototipado, y se migrará parte de la aplicación a JAVA para agregar capacidades gráficas que permitan una manera más sencilla de interactuar para los usuarios.

VI. Resultados

Se logró concretar el algorítmo y realizar pruebas suficientes en una ruta para determinar que la base del algorítmo funciona. En la figura 1 de los anexos se ven las rutas de entrada entregadas al algoritmo y en la figura 2 se muestra el resultado posterior al aplicar el algoritmo.

La aplicación logra tomar exitosamente los archivos con extensión gpx, procesarlos y convertirlos en texto plano, luego utilizar la información para realizar una comparación entre los puntos de cada ruta tomada. En esta comparación, si hay una distancia menor a 75 metros (variable) crea una lista para los dos puntos que se están comparando en el momento, y los enlista a ambos en esta lista. Si encuentra un nuevo punto que tambíen se encuentra dentro de esta distancia de 75 metros lo agrega a la lista.

Al final de este proceso, se hace una comparación entre la cantidad de elementos que tiene cada lista. Si la cantidad cambia de un punto a otro, esto quiere decir que entre los puntos alguna persona se subió al bus y generó algún otro punto, o por otro lado una persona abandonó el bus y no generó un punto. Por lo tanto, en cada punto donde esta cantidad varía, el algoritmo generá un punto en coordenadas geográficas, y posteriormente lo guarda en un nuevo archivo que puede ser leído por el usuario. La **figura 3** ejemplifíca el proceso del algoritmo.

El código se logro implementar a través de dos dependencias principales: gpxpy y geopy, librerías para el manejo de datos gpx y manejo de información latitudinal y longitudinal, y comparaciones entre puntos de coordenadas.

El proyecto de la aplicación se manejó correctamente a través de la plataforma Github, permitiendo involucrar a cualquier interesado en el proyecto de manera ordenada. El proyecto se llama **openbus-navigator**, donde se encuentra todo el código.

VII. Conclusiones

- Se diseñó un algoritmo que determina las rutas y paradas basado en datos obtenidos a través de una aplicación tipo *GPS tracker*.
- Se implementó el algoritmo en python el cual genera un archivo .gpx a partir de todas las entradas a una misma ruta, que contiene la ruta y las paradas establecidas para la misma. Esta ruta es totalmente independiente de datos externos de entes públicos y se basa en la información obtenida de usuario a usuario.
- Se desarrolló un sistema cliente/servidor para compartir los archivos .gpx creados desde la aplicación y estos a su vez procesados en el mismo servidor y compartidos de vuelta a solicitud del cliente.
- Se creó una aplicación con la funcionalidad de ser tracker y solicitar resultados de una ruta determinada, esta se despliega en un mapa con la información de las paradas (ver Fig 1 y 2).

Referencias

[1] Jeske, T. (2013). Floating car data from smartphones: What google and waze know about you and how hackers can control traffic. Proc. of the BlackHat Europe.
[2] "Kivy: Cross-platform Python Framework for NUI", Kivy.org, 2016. [Online]. Available: https://kivy.org/. [Accessed: 02- Jun- 2016].
[3] "Welcome to Python.org", Python.org, 2016. [Online]. Available: https://www.python.org/. [Accessed: 02- Jun- 2016].

[4] “OpenStreetMap”, OpenStreetMap, 2016. [Online]. Available: https://www.openstreetmap.org/#map=5/51.500/-0.100. [Accessed: 02- Jun- 2016].
[5] Bosque D, 2016. Grupo de buseros ignoró pedido para opinar sobre plan de sectorización. La Nacion. San Jose, Costa Rica

VIII. ANEXOS

Figura 1: *Ruta utilizada para la prueba de la aplicación. Las líneas representan cada una de las mediciones tomadas (Autobus interno de la UCR).*

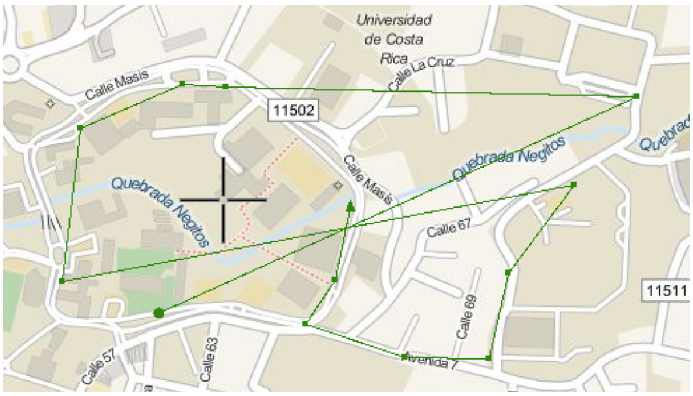

Figura 2: *Respuesta generada por el algorítmo después de procesar los datos.*

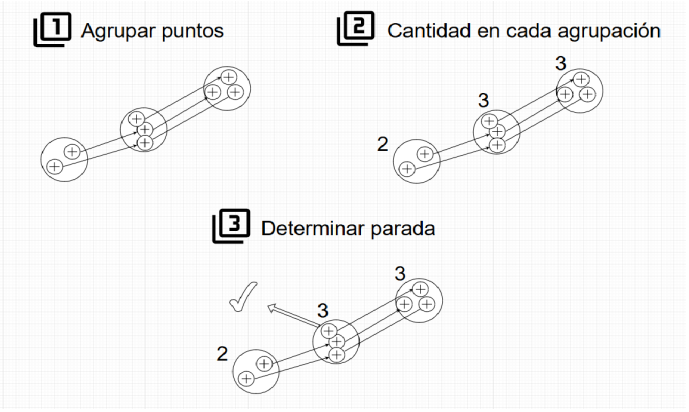

Figura 3: *Lógica detrás del algoritmo utilizado.*