



Enfoque UTE

ISSN: 1390-9363

ISSN: 1390-6542

Universidad Tecnológica Equinoccial

Rojas, Juan Pablo; Bustos, Julio César; Ordóñez Camacho, Diego
Transporte público inteligente al alcance de sus manos
Enfoque UTE, vol. 8, núm. 1, Suppl, 2017, pp. 122-134
Universidad Tecnológica Equinoccial

DOI: <https://doi.org/10.29019/enfoqueute.v8n1.143>

Disponible en: <https://www.redalyc.org/articulo.oa?id=572262176009>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

UAEH redalyc.org

Sistema de Información Científica Redalyc
Red de Revistas Científicas de América Latina y el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso
abierto

Transporte público inteligente al alcance de sus manos

(Smart public transportation at your fingertips)

Juan Pablo Rojas¹, Julio César Bustos¹, Diego Ordóñez Camacho¹

Resumen:

El sistema QBus permite ubicar las rutas de buses que circulan en la ciudad de Quito. Está conformado por una aplicación móvil la cual utiliza tecnología Bluetooth Low Energy o códigos Quick Response para identificar cada una de las paradas. Se basa en la ubicación del usuario para mostrar las paradas más cercanas y buscar las rutas de buses disponibles. La aplicación muestra información relevante de cada ruta y, mediante un mapa digital, dibuja su trayecto y las paradas que realiza. Además de esto, ofrece la funcionalidad "¿Cómo Llego?" que permite al usuario buscar la ruta de buses disponibles para desplazarse de un punto a otro dentro de la ciudad. Para cumplir con este proceso, QBus realiza una búsqueda de las rutas de buses que tengan paradas cercanas a los puntos de origen y destino; esta funcionalidad indica al usuario la parada en la cual debe tomar el bus y en cual debe bajarse. Por otro lado el sistema consta de un portal web el cual permite gestionar toda la información de la aplicación, es decir, cooperativas, paradas y rutas de buses de una manera gráfica mediante el uso de la API V3 de Google Maps.

Palabras clave: transporte público; Bluetooth Low Energy; mapas digitales; ciudades inteligentes; aplicaciones ubicuas

Abstract:

QBus is a system that locates the bus routes circulating in the city of Quito. It consists of a mobile application which uses either Bluetooth Low Energy technology or Quick Response codes to identify the stops. It is based on the location of the user to show the nearest stops and search the available bus routes. The application shows relevant information of each route and, using a digital map, draws its path and the stops it performs. In addition to this, it offers the "How do I get there" functionality that allows the user to search the available bus routes to move from one point to another within the city. In order to comply with this process, QBus performs a search of the bus routes that have stops near the points of origin and destination; this functionality indicates to the user the stop in which he must take the bus and in which it must get off. On the other hand the system consists of a web portal which allows to manage all the information of the application, ie cooperatives, stops and bus routes in a graphical way through the use of the Google Maps V3 API.

Keywords: public transportation; Bluetooth Low Energy; digital maps; smart cities; ubiquitous apps

¹ Universidad Tecnológica Equinoccial, Quito – Ecuador ({rhjp1023573, bnjc1023650, dordonez}@ute.edu.ec)

1. Introducción

El sistema QBus permite a los usuarios de transporte público conocer la ubicación de las paradas, las líneas de transporte que pasan por ellas, y el recorrido de dichas líneas, permitiendo también graficar su trayecto en el mapa digital de Google Maps. Permite usar la aplicación tanto en línea, mediante un browser como desde un Smartphone, sin la necesidad de tener conexión a internet. Para el reconocimiento de las paradas hace uso de tecnología Bluetooth Low Energy, BLE, y de códigos Quick Response, QR. La aplicación web permite además de lo mencionado, administrar el contenido de la aplicación, como lo es crear paradas y dibujar rutas nuevas utilizando la API de Google Maps para JavaScript.

Para una correcta movilidad en las urbes actuales, maximizando el uso del tiempo, los usuarios del transporte público necesitan obtener información precisa sobre las posibilidades de desplazamiento entre los puntos de origen y destino. Esto implica un conocimiento apropiado de las rutas y paradas, lo cual, dado el tamaño tanto de las urbes como de las redes de transporte implementadas en ellas, se beneficia de algún tipo de ayuda automatizada (Sungur, Babaoglu, & Sungur, 2015).

Los sistemas de transporte en general, y con respecto a sus usuarios más particularmente, se encuentran constantemente exigiendo mejoras tecnológicas que permitan aprovechar de manera más eficiente sus servicios. Estas mejoras requieren de la interacción de una variada gama de elementos íntimamente relacionados con las tecnologías de la información, y especialmente con las facilidades al alcance del usuario típico a través de su Smartphone (Nkoro & Vershinin, 2014).

El problema de ubicarse en relación a la red de transporte público es muy común y en ocasiones dramático, sobre todo para el turista. Verma et al (Verma et al., 2016) describen de manera muy acertada las peripecias a las que se puede ver enfrentada una persona que sin estar familiarizada con una ciudad mal señalizada, decide utilizar el transporte público, principalmente: ¿Dónde están las paradas más cercanas y qué líneas de bus pasan por ellas? ¿Cómo llegar hasta el lugar de destino? ¿Es necesario caminar? ¿Cuánto tiempo puede tomar el trayecto? Estas y muchas otras dudas y dificultades forman parte del día a día de un gran número de personas al enfrentarse al transporte público.

Enfrentar esta problemática, como ya se ha mencionado, requiere del concurso de una diversidad de elementos que deben ser correctamente conjugados e integrados en una arquitectura distribuida capaz de consolidar distintos sistemas de transporte, estructuras de datos y dispositivos heterogéneos tanto en lo que se refiere al componente emisor como al sensorial, para proporcionar al usuario una experiencia eficiente con un razonable uso de recursos (Nasim & Kassler, 2012).

1.1 Trabajos relacionados

Actualmente es común encontrar en ciudades sistemas de transporte inteligente (Santos, Pereira, & Leal, 2012). Estos sistemas son generalmente de tipo web, y permiten al usuario conectado al internet buscar una ruta entre dos puntos, con las respectivas relaciones de transporte público. Ejemplos interesantes de este tipo de sistema pueden ser SMARTBUS (Ram et al., 2016), el cual es una iniciativa internacional de investigación entre universidades e instituciones gubernamentales, puesto en práctica en Fortaleza, Brasil, que entre otras cosas busca un posicionamiento más adecuado de las paradas, de acuerdo a datos recolectados de los usuarios. Más cerca del gran público, uno de los sistemas web más desarrollados es Moovit (Moovit Inc., 2016), que ofrece una red de varias ciudades a nivel mundial, y que integra una aplicación móvil que no solo facilita el uso del browser en el dispositivo, sino que integra ya el GPS para el posicionamiento, siempre mientras el usuario está en línea.

Más allá del típico uso del GPS, SubwayPS (Stockx, Hecht, & Schöning, 2014) toma en cuenta que bajo techo dicha señal se pierde, e implementa un sistema basado principalmente en el acelerómetro para brindar ayuda al posicionamiento en las redes de metro. Más cerca de los intereses del presente proyecto, Palumbo et al (Palumbo, Barsocchi, Chessa, & Augusto, 2015) investigan el uso de Bluetooth Low Energy, BLE, y Received Signal Strength, RSS. Asimismo, tanto Kobayashi (Kobayashi, 2015), como Eken y Sayar (Eken & Sayar, 2014) utilizan códigos Quick Response, QR, para resolver el problema de la ubicación en las paradas. Couto et al (Couto, Leal, Costa, & Galvão, 2015) juntan BLE, RSS y Near Field Communication, NFC, para explorar el problema del tiquetaje, siempre en relación con el transporte público.

A continuación el artículo se encuentra dividido de la siguiente manera: la sección 2 detalla los métodos, tecnologías y herramientas utilizadas; en la sección 3 se presentan los detalles del sistema resultante, considerando funcionalidad, diseño y pruebas; la sección 4 entabla una discusión que intenta comparar el sistema presentado con aquellos existentes destacando los puntos fuertes y débiles relevantes; finalmente la sección 5 presenta las conclusiones más importantes así como ciertas recomendaciones de trabajo a futuro.

2. Metodología

Gracias al análisis de los emprendimientos realizados a nivel de la investigación en el área, así como del sondeo de la problemática y necesidades de los usuarios se pudo delinear un sistema que facilite conocer la ubicación de las paradas, las líneas de transporte que pasan por ellas, y el recorrido de dichas líneas, permitiendo también graficar un trayecto. Uno de los requisitos de base también fue, además de poder usar la aplicación en línea, mediante un browser y conexión a internet, también poder hacerlo desde un Smartphone, sin conexión a internet, y en la parada misma.

Posterior a la ejecución de un plan rápido inicial que consistió en identificar los requerimientos funcionales y no funcionales, y un diseño del prototipo que incluyó el desarrollo de diagramas de secuencia, diagramas de flujo, diagramas de clase y el diseño de la base de datos; se decidió implementar un sistema distribuido con un servidor centralizado en la nube, el cual contiene la información de cooperativas de buses, paradas, líneas de transporte y rutas, y responde a las consultas distribuyendo la información tanto en HTML para su uso en un navegador, como en mensajes JSON personalizados, para su uso en aplicaciones cliente móviles, mediante un servicio web. Este componente servidor se desarrolló utilizando PHP y una base de datos relacional MySQL.

El lado cliente consideró dos alternativas, su uso en un browser estándar, para lo cual se trabajó con HTML5, CSS3 y JavaScript sobre el CMS Drupal, y su uso en un dispositivo móvil tipo Smartphone, para la cual se decidió realizar una aplicación nativa para iOS utilizando el lenguaje de programación Objective-C.

Para el primer caso QBus cuenta con dos elementos, un portal de administración de información y una página web orientada netamente a los usuarios del transporte público. Dentro del portal de administración se puede agregar todos los datos concernientes a las cooperativas de buses, paradas, rutas y graficar los trayectos en un mapa digital implementando el API de Google Maps. Mientras que en la página web los usuarios pueden obtener sugerencias de rutas para desplazarse de un lugar a otro dentro de la ciudad.

En el caso de la aplicación móvil se concibió sobre todo para un uso fuera de línea, por lo cual al encontrarse conectada a internet sincroniza la información del servidor con una base de datos interna.

Para el posicionamiento, al estar en línea se puede hacer uso del GPS en conjunto con el framework nativo de mapas de iOS, Maps. Para aquellos casos en que se está fuera de línea, y muy probablemente fuera del alcance GPS, como sucedería en una parada subterránea de metro, se implementó un sistema doble que puede utilizar BLE y QR.

BLE permite, mediante un beacon instalado en la parada, que el dispositivo conozca su ubicación exacta. Dado que esta tecnología es relativamente nueva, mediante códigos QR y la cámara del dispositivo, la aplicación puede también conocer en qué parada se encuentra el usuario.

3. Resultados

El resultado final de la investigación fue el desarrollo de un prototipo funcional del sistema QBus, el cual permite brindar un apoyo para los usuarios de transporte público de Quito al proporcionar información acerca de las distintas líneas de buses que circulan por la ciudad.

Entre las funcionalidades más importantes del sistema está el poder dar servicio al usuario fuera de línea, a través de la aplicación móvil. En la *Figura 1* se esquematiza el proceso de toma de decisiones que permite manejar los dos estados posibles: online y offline. En caso de encontrarse conexión a internet, la aplicación compara, a través de hashing por rapidez, la versión de los datos en el servidor con la versión guardada, actualizando los datos si es del caso.

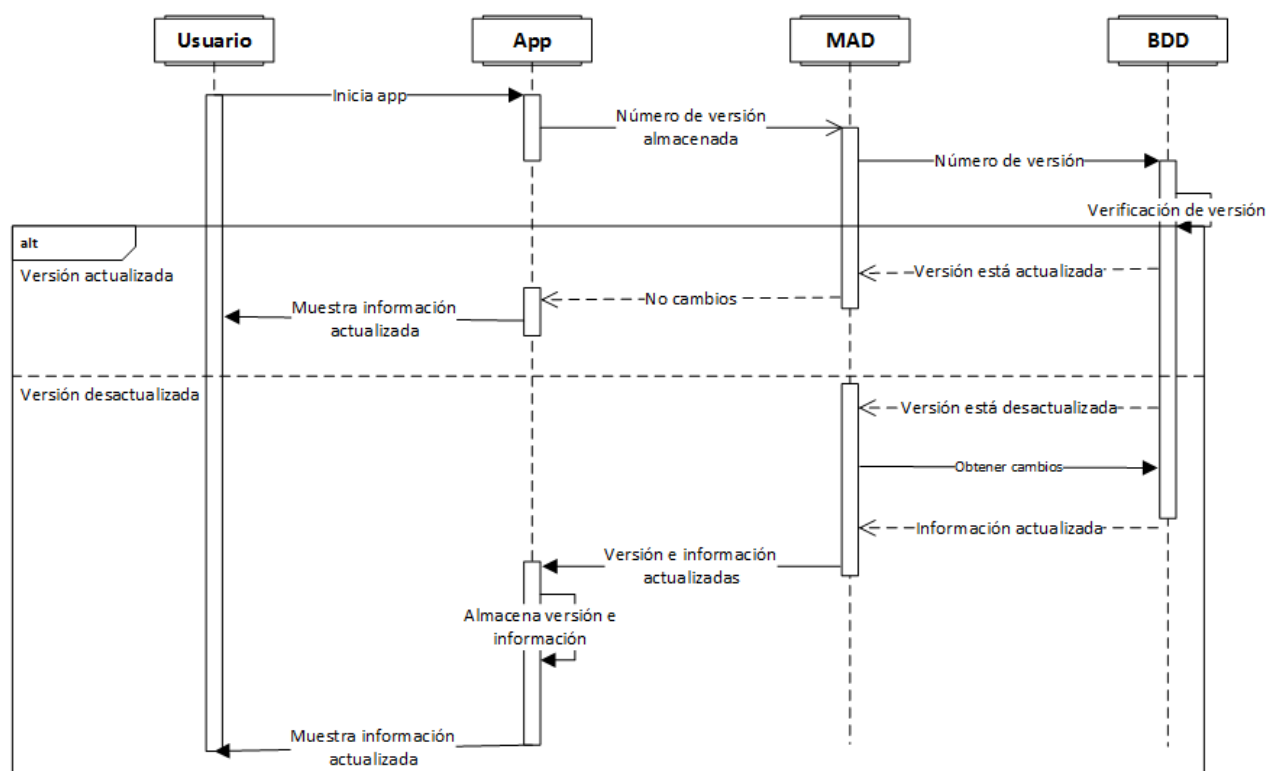


Figura 1. Detección del estado online y offline

Clave entre las decisiones de diseño del sistema se encuentra la posibilidad de utilizar BLE y QR mediante la aplicación móvil para conocer en qué parada se encuentra el usuario, y así poder efectuar búsquedas específicas con dicho origen. Ambas alternativas son funcionalmente iguales y en la *Figura 2* se presenta la alternativa BLE.

Cuando el usuario ingresa a la funcionalidad del buscador de beacons, la aplicación inicia el monitoreo para encontrar uno cercano. En el caso de que sea detectado, la aplicación recibe el UUID y en base al identificador realiza una búsqueda en los datos que tiene almacenados internamente el dispositivo, para presentar la información de la ruta correspondiente. Si no existe ningún registro con dicho identificador se muestra un mensaje al usuario indicando lo sucedido. El mismo proceso sigue la funcionalidad de captura de códigos QR.

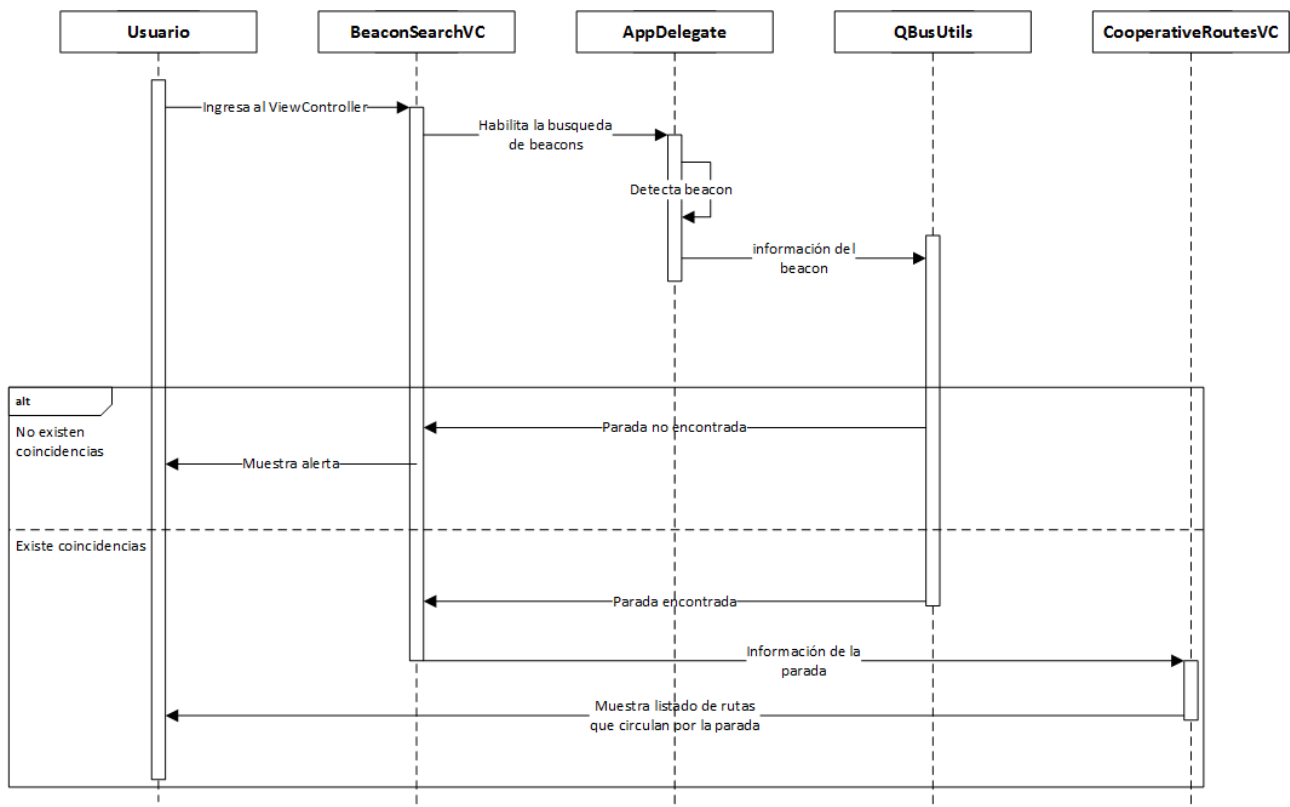


Figura 2. Proceso BLE para detección de paradas

Una de las principales necesidades de los usuarios es conocer qué ruta tomar para poder llegar a un destino, Generalmente esta acción considera como origen la parada actual, sin embargo el sistema es suficientemente genérico como para tanto desde un browser, como desde la aplicación móvil, poder seleccionar el punto de partida y el de llegada. El proceso general se esquematiza en la *Figura 3*, donde se presenta la secuencia que sigue la aplicación para mostrar toda la información disponible llegando hasta la presentación de sugerencias para el desplazamiento, que se puede apreciar en la aplicación, en la *Figura 4*, y en la página web en la *Figura 5*.

Las pruebas de funcionamiento del sistema fueron de caja negra principalmente, con mayor énfasis en la aplicación móvil, y se organizaron en tres fases, requerimientos, funcionalidad, y resistencia y rendimiento, finalizando con la verificación del protocolo de aseguramiento de calidad, con resultados positivos en todos sus ítems, lo cual se presenta en la *Tabla 1*.

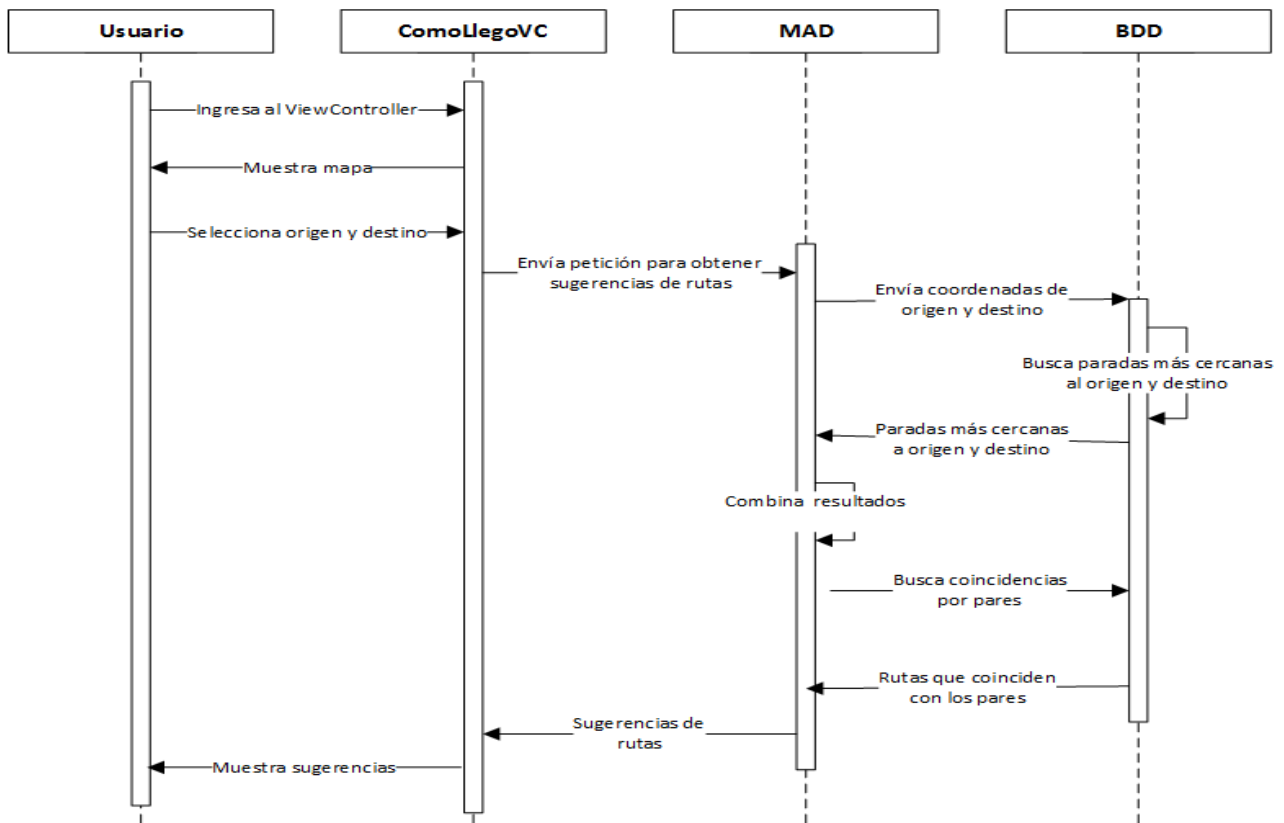


Figura 3. Buscador de rutas

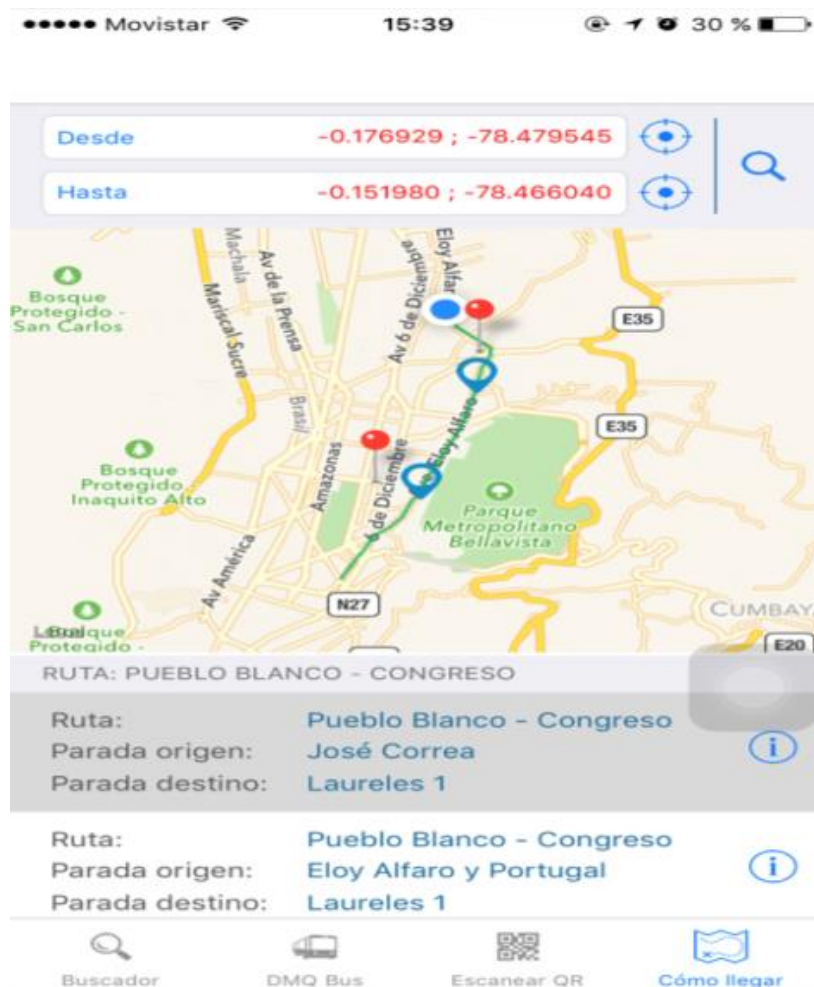


Figure 4. Sugerecias de ruta

Q-Bus

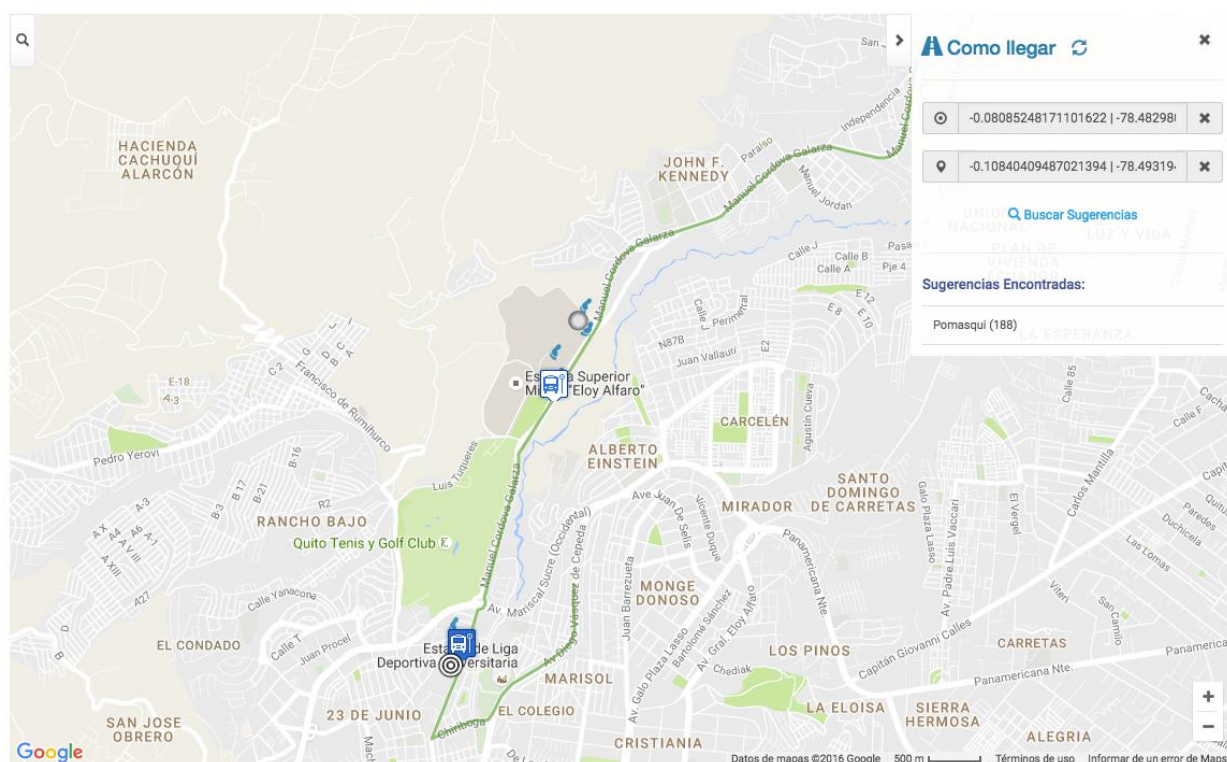


Figura 1. Funcionalidad Cómo Llego en la página web

Tabla 1. Pruebas y protocolo de aseguramiento de calidad

Pruebas y actividades validadas satisfactoriamente
Instalación en equipos
Obtener información vía WiFi
Obtener información vía EDGE, 3G, LTE
Verificar el funcionamiento del Bluetooth
Activar la localización del usuario
Detectar beacon y mostrar información asociada
Capturar código QR y mostrar información asociada
Buscar una cooperativa, ver sus rutas y detalles
Ver el recorrido de una ruta y las paradas por las cuales circula
Utilizar la funcionalidad ¿Cómo llego? Para detectar un punto en el mapa y mostrar sugerencias de buses para llegar hasta el mismo
Verificar la posición del usuario en el mapa digital.
Salir y liberar recursos del aplicativo

3. Discusión

El sistema QBus se encuentra implementado con todas las funcionalidades mencionadas, y para poder identificar las fortalezas y debilidades del producto como tal se hizo una comparativa con otras aplicaciones que ofrezcan información sobre el transporte público de una o varias ciudades.

Las aplicaciones utilizadas para realizar la comparación fueron seleccionadas a través de una búsqueda en AppStore, siendo los parámetros de idoneidad las descargas realizadas, calificaciones obtenidas, costo y características de las apps. Resultado de esta búsqueda fueron elegidas Moovit (Moovit Inc., 2016), Google Maps (Google Inc., 2017), BA Cómo Llego (Buenos Aires Ciudad, 2017) y Transit App (Transit App Inc., 2017).

Entre las principales fortalezas con las que cuenta QBus es el uso de tecnologías como BLE y códigos QR para la identificación de paradas, funcionalidad que ninguna de las apps analizadas ofrece. Es importante mencionar también que QBus permite presentar no únicamente información correspondiente a buses, sino también rutas de los distintos corredores como Trolebús, Ecovía, Metrobus, Metro y sus alimentadores, lo cual permite a la aplicación brindar mayor información al usuario. Para obtener sugerencias de buses para llegar desde un lugar a otro, tanto QBus como Google Maps y Transit App utilizan el método de seleccionar puntos en el mapa, lo cual dependiendo de las circunstancias puede ser más sencillo para el usuario que recordar el nombre de una calle o intersecciones.

QBus no presenta información en tiempo real de la ubicación de los distintos buses debido a que esta funcionalidad requeriría del uso de un GPS en cada bus lo que supondría una inversión costosa. Sin embargo, podría ser también una tentativa para una próxima investigación ya que aportaría tanto a QBus como al control de la ruta que sigue el bus y el cumplimiento de los horarios establecidos. Aplicaciones como Moovit y Transit App ofrecen esta funcionalidad, pero no hacen uso de dispositivos GPS; calculan la posición actual de un bus basados en el tiempo de recorrido, el horario de salida y la distancia que deben recorrer. El problema con este método es que no se toman en cuenta variables externas como la densidad del tráfico que puede existir, accidentes o cualquier otro factor que pueda afectar al recorrido del bus.

Cabe recalcar que QBus todavía se la considera un prototipo dado que al momento no se ha puesto a disposición del público general, principalmente por que aún se busca incluir algunas funcionalidades secundarias, que podrían ayudarla a ser más competitiva en el mercado y atractiva para los usuarios. Tal es el caso del soporte para entregar sugerencias para llegar desde un lugar a otro teniendo en cuenta los trasbordos de los usuarios. Sin embargo, presenta también otras funcionalidades como el uso de BLE y códigos QR, para presentar información completa de las rutas de buses que circulan por determinada parada, atractivo que ninguna otra app de las analizadas presenta.

La *Tabla 2* muestra los resultados completos obtenidos durante el proceso de comparación.

Tabla 2. Comparación de funcionalidad

Funcionalidad	Aplicaciones				
	Q-Bus	Moovit	Google Maps	Cómo BA Llego	Transit App
DISTRIBUCIÓN					
Distribución a través de tiendas virtuales	n/a	X	X	X	X
Aplicación gratuita	n/a	X	X	X	X
ÁMBITO					
Soporte para múltiples ciudades	-	X	X	-	X
Soporta una o más ciudades de Ecuador	X	-	-	-	-
Información proveniente de la comunidad	-	X	X	-	X
Permite visualizar información de la ubicación de distintos buses, metro, tranvía, etc. En tiempo real	-	X	-	-	X
Soporta múltiples medios de transporte público (bus, tranvía, metro, subterráneo)	X	-	X	X	X
SUGERENCIAS PARA LLEGAR DESDE UN LUGAR A OTRO					
Presenta sugerencias de rutas para llegar desde un punto hasta otro	X	X	X	X	X
Es posible seleccionar los puntos mediante la selección de puntos en el mapa	X	-	X	-	X
Permite la identificación de paradas cercanas	X	X	-	-	X
Presenta sugerencias de rutas tomando en cuenta trasbordos	-	-	X	X	X
Permite obtener información detallada de las rutas que circulan por las paradas	X	-	-	-	-
PARTICULARIDADES					
Presenta mapas temáticos e información turística	-	-	X	X	X
Implementa notificaciones	-	X	-	-	X

Funciona sin conexión a internet	X	X	-	X	X
Utiliza tecnología Bluetooth para la detección de paradas	X	-	-	-	-
Utiliza códigos QR para la detección de paradas	X	-	-	-	-

4. Conclusiones y Recomendaciones

QBus es un sistema que engloba una amplia gama de características útiles para el usuario de transporte público, resolviendo especialmente el problema de obtener información cuando no hay conexión internet, y el usuario se encuentra ya en búsqueda de una buena alternativa de transporte, mientras se desplaza en la ciudad.

El sistema hace una correcta integración de tecnologías clásicas y de punta, consolidando GPS, QR y BLE en una aplicación móvil bien integrada al sistema web.

Las herramientas y librerías seleccionadas, descritas en la *Sección 2*, permitieron acortar los tiempos de desarrollo del proyecto y brindar una interfaz visual mejorada al usuario.

A futuro se planea desarrollar una versión de QBus para la plataforma Android, la cual tiene mayor porcentaje de usuarios en el mercado.

En próximas investigaciones se implementarán nuevas funcionalidades como las sugerencias de rutas implementando trasbordos, manejo del sentido de las rutas e implementación de protocolos seguros de transmisión de datos como los es HTTPS para evitar el robo de la información que fluye en la aplicación así como la ubicación.

Un cuello de botella es la recolección de información de paradas y líneas, para lo cual se pretende implementar las recomendaciones de (Santos et al., 2012) automatizando la recolección de esta información con la participación del mismo usuario.

Además, mediante las innovaciones que han tenido los beacons (Kontakt.io, s/f), se planea incorporar funcionalidades que permitan recolectar información sobre aspectos como la temperatura o grado de contaminación del ambiente en las distintas paradas de la ciudad.

Bibliografía

- Buenos Aires Ciudad. (2017). CómoLlego. Recuperado el 2 de septiembre de 2017, a partir de <http://comollego.ba.gob.ar/>
- Couto, R., Leal, J., Costa, P. M., & Galvão, T. (2015). Exploring Ticketing Approaches Using Mobile Technologies: QR Codes, NFC and BLE. En 2015 IEEE 18th International Conference on Intelligent Transportation Systems (pp. 7–12). <https://doi.org/10.1109/ITSC.2015.9>
- Eken, S., & Sayar, A. (2014). A smart bus tracking system based on location-aware services and QR codes. En 2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings (pp. 299–303). <https://doi.org/10.1109/INISTA.2014.6873634>
- Google Inc. (2017). Google Maps. Recuperado el 2 de septiembre de 2017, a partir de <https://www.google.com/maps/>
- Kobayashi, H. (2015). A personal localization system using self-contained 2D barcode landmarks. En 2015 IEEE 24th International Symposium on Industrial Electronics (ISIE) (pp. 882–887). <https://doi.org/10.1109/ISIE.2015.7281587>
- Kontakt.io. (s/f). iBeacon vs Eddystone. Recuperado el 19 de septiembre de 2016, a partir de <https://kontakt.io/beacon-basics/ibeacon-and-eddystone/>
- Moovit Inc. (2016). Moovit: Your Local Transit App – iOS, Android & Windows. Recuperado el 2 de septiembre de 2016, a partir de <http://moovitapp.com/>
- Nasim, R., & Kassler, A. (2012). Distributed Architectures for Intelligent Transport Systems: A Survey. En 2012 Second Symposium on Network Cloud Computing and Applications (NCCA) (pp. 130–136). <https://doi.org/10.1109/NCCA.2012.15>
- Nkoro, A. B., & Vershinin, Y. A. (2014). Current and future trends in applications of Intelligent Transport Systems on cars and infrastructure. En 17th International IEEE Conference on Intelligent Transportation Systems (ITSC) (pp. 514–519). <https://doi.org/10.1109/ITSC.2014.6957741>
- Palumbo, F., Barsocchi, P., Chessa, S., & Augusto, J. C. (2015). A stigmergic approach to indoor localization using Bluetooth Low Energy beacons. En 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (pp. 1–6). <https://doi.org/10.1109/AVSS.2015.7301734>

- Ram, S., Wang, Y., Currim, F., Dong, F., Dantas, E., & Sabóia, L. A. (2016). SMARTBUS: A Web Application for Smart Urban Mobility and Transportation. En Proceedings of the 25th International Conference Companion on World Wide Web (pp. 363–368). Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. <https://doi.org/10.1145/2872518.2888613>
- Santos, M., Pereira, R. L., & Leal, A. B. (2012). GBUS - Route GeoTracer. En 2012 First International Workshop on Vehicular Traffic Management for Smart Cities (VTM) (pp. 1–6). <https://doi.org/10.1109/VTM.2012.6398697>
- Stockx, T., Hecht, B., & Schöning, J. (2014). SubwayPS: Towards Smartphone Positioning in Underground Public Transportation Systems. En Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (pp. 93–102). New York, NY, USA: ACM. <https://doi.org/10.1145/2666310.2666396>
- Sungur, C., Babaoglu, I., & Sungur, A. (2015). Smart Bus Station-Passenger Information System. En 2015 2nd International Conference on Information Science and Control Engineering (ICISCE) (pp. 921–925). <https://doi.org/10.1109/ICISCE.2015.209>
- Transit App, Inc. (2016). Transit. Recuperado el 7 de febrero de 2017, a partir de <http://transitapp.com/>
- Verma, R., Shrivastava, A., Mitra, B., Saha, S., Ganguly, N., Nandi, S., & Chakraborty, S. (2016). UrbanEye: An outdoor localization system for public transport. En IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications (pp. 1–9). <https://doi.org/10.1109/INFOCOM.2016.7524393>