

## Propuesta de una aplicación móvil para el acceso a geo-información del transporte público de Riobamba

*Proposal for a mobile application for access to geo-information on public transportation in Riobamba*

- <sup>1</sup> Brando Israel Toledo Yanza  <https://orcid.org/0000-0001-9802-0875>  
Escuela Superior Politécnica de Chimborazo (ESPOCH), Riobamba, Ecuador.  
[brando.toledo@esPOCH.edu.ec](mailto:brando.toledo@esPOCH.edu.ec)
- <sup>2</sup> Gisel Katerine Bastidas Guacho  <https://orcid.org/0000-0002-6070-7193>  
Escuela Superior Politécnica de Chimborazo (ESPOCH), Riobamba, Ecuador.  
[gis.bastidas@oesPOCH.edu.ec](mailto:gis.bastidas@oesPOCH.edu.ec)
- <sup>3</sup> Jorge Alfredo Caiza Balseca  <https://orcid.org/0000-0002-8979-0760>  
Instituto Superior Tecnológico Sucúa, Sucúa, Ecuador.  
[jorge.caiza@istsucua.edu.ec](mailto:jorge.caiza@istsucua.edu.ec)
- <sup>4</sup> Patricio Xavier Moreno Vallejo  <https://orcid.org/0000-0002-9317-9884>  
Escuela Superior Politécnica de Chimborazo (ESPOCH), Riobamba, Ecuador.  
[pxmoreno@esPOCH.edu.ec](mailto:pxmoreno@esPOCH.edu.ec)



### Artículo de Investigación Científica y Tecnológica

Enviado: 13/05/2022

Revisado: 22/06/2022

Aceptado: 01/07/2022

Publicado: 20/07/2022

DOI: <https://doi.org/10.33262/concienciadigital.v5i3.1.2237>

Cítese:

Toledo Yanza, B. I., Bastidas Guacho, G. K., Caiza Balseca, J. A., & Moreno Vallejo, P. X. (2022). Propuesta de una aplicación móvil para el acceso a geo-información del transporte público de Riobamba . ConcienciaDigital, 5(3.1), 162-183. <https://doi.org/10.33262/concienciadigital.v5i3.1.2237>



**CONCIENCIA DIGITAL**, es una revista multidisciplinar, **trimestral**, que se publicará en soporte electrónico tiene como **misión** contribuir a la formación de profesionales competentes con visión humanística y crítica que sean capaces de exponer sus resultados investigativos y científicos en la misma medida que se promueva mediante su intervención cambios positivos en la sociedad. <https://concienciadigital.org>

La revista es editada por la Editorial Ciencia Digital (Editorial de prestigio registrada en la Cámara Ecuatoriana de Libro con No de Afiliación 663) [www.celibro.org.ec](http://www.celibro.org.ec)

Esta revista está protegida bajo una licencia Creative Commons Attribution Non Commercial No Derivatives 4.0 International. Copia de la licencia: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

**Palabras claves:**

geolocalización,  
mobile-d,  
transporte, rutas,  
software.

**Keywords:**

geolocalization,  
mobile-d,  
transportation,  
routes, software.

**Resumen**

**Introducción:** El transporte público es de vital importancia para las ciudades debido a que permite una mejor planificación urbana de la ciudad evitando el crecimiento del transporte privado y de esta forma la contaminación ambiental y acústica en las ciudades. Sin embargo, el desconocimiento del funcionamiento del transporte público provoca que los ciudadanos no lo utilicen, optando por el transporte privado. **Objetivos:** Por lo tanto, en el presente estudio se propone desarrollar una aplicación móvil con georreferenciación que permita una fácil disponibilidad de acceso a la información sobre el servicio de transporte público a los usuarios que residen en la ciudad de Riobamba, provincia de Chimborazo, bajo el sistema operativo Android, y con el objetivo de potenciar el uso del transporte público urbano mediante la ayuda de la asistencia GPS en la geolocalización de buses. **Metodología:** La metodología de desarrollo utilizada fue MOBILE-D, se establecieron requerimientos del proyecto mediante técnicas de recolección de datos, se ejecutaron iteraciones planificadas mediante el uso de herramientas como: MySQL como sistema gestor de base de datos, Laravel 8 como framework de desarrollo del back-end, React Native como framework de desarrollo para la aplicación móvil mediante el lenguaje de programación JavaScript, Google Maps y Pubnub como servidores en la presentación de rutas, paradas y buses en un mapa de la ciudad. **Resultados:** Para la evaluación del sistema se realizó el cálculo del porcentaje de disponibilidad de la aplicación y de la distribución de la aplicación, mediante la fórmula de disponibilidad establecida por la *Information Technology Infrastructure Library* (ITIL), la cual impuso niveles de disponibilidad del 100% y 99.767% respectivamente, mismas que fueron evaluadas durante el período diciembre de 2021 a enero de 2022. **Conclusiones:** La aplicación desarrollada logró de manera óptima el objetivo de otorgar un servicio informativo del transporte público a la ciudadanía de Riobamba.

**Abstract**

**Introduction:** Public transportation is vital for cities because it allows better urban planning of the city, avoiding the growth of private transport and thus environmental and noise pollution in cities. However, the lack of knowledge of the operation of public

---

transportation means that citizens do not use it, opting for private transportation. **Objective:** Therefore, in this study, it is proposed to develop a mobile application with georeferencing that allows easy availability of access to information on the public transportation service to users who reside in Riobamba city, province of Chimborazo. The application is under the Android operating system and promotes the use of urban public transportation through the help of GPS assistance in the geolocation of buses. **Methodology:** The development methodology used was MOBILE-D, project requirements were established through data collection techniques, and planned iterations were executed through the use of tools such as MySQL as a database management system, Laravel 8 as the back-end development framework, React Native as the development framework for the mobile application using JavaScript programming language, Google Maps, and Pubnub as servers in the presentation of routes, stops and buses on a city map. **Results:** For the evaluation of the system, the application's percentage of availability and the application's distribution were computed using the availability formula established by the Information Technology Infrastructure Library (ITIL), which imposed availability levels of 100% and 99.767%, respectively. The application was evaluated during the period December 2021 to January 2022. **Conclusions:** The developed application optimally achieved the objective of providing an information service of public transport to Riobamba's citizens.

---

## Introducción

El transporte público urbano ha sido objeto de varios estudios a lo largo del tiempo. Han existido estudios referentes al transporte público en temas de contaminación ambiental (Suárez et al., 2018), y acústica (Jácome et al., 2019), al uso de la tecnología para mejorar la seguridad (Haro et al., 2019), y obtener de forma automática el aforo (Moreno et al., 2020). El aumento de población en las urbes admite nuevos retos en el marco de desarrollo de actividades de cada uno de los individuos, ya sea actividades productivas o de entretenimiento. En dicho argumento, las ciudades deben enfocarse en mejorar servicios como el sistema de transporte público urbano, al cual acceden la mayor parte de la población principalmente por motivos de movilización.

Dentro del área urbana de la ciudad de Riobamba existen diferentes rutas de transporte público, las cuales, por diversas circunstancias, se desconoce hoy en día; ya sea tanto para los moradores actuales o para aquellas personas que están temporalmente la ciudad, como es el caso de los estudiantes universitarios, en donde el desconocer la ruta puede provocar saturación de pasajeros en las líneas más conocidas o aumentar el tráfico vehicular en la ciudad al preferir el transporte particular.

Existen algunos frameworks para el desarrollo de aplicaciones móviles que facilitan y agilizan el desarrollo de manera más limpia y rápida, en el caso de React Native permite crear aplicaciones bien estructuradas con pequeños grupos de desarrollo en tiempos cortos (Danielsson, 2016a, 2018; Hansson & Vidhall, 2016; Söderberg et al., 2018). React Native es un framework desarrollado por Facebook para la construcción de aplicaciones móviles nativas en Javascript que está basado en ReactJS, una librería de Facebook para la construcción de interfaces (Neftali et al., 2019). Una de las ventajas de React-native es que permite generar aplicaciones móviles híbridas para que funcionen en diferentes plataformas (Kaushik et al., 2019). Dado que las diferentes plataformas tienen diferentes aspectos, sensaciones y capacidades, no puede haber una aplicación que sea homogénea en todos los sistemas operativos.

El renderizado que utiliza tanto para ReactJS y React Native es similar, sin embargo es el punto de destino al cual va dirigido lo que lo hace realmente interesante de utilizar, debido a que muchas empresas usan React Native para crear sus aplicaciones móviles, tales como Facebook, Instagram, Uber, Pinterest, por nombrar algunos; y además no se necesita ser un experto desarrollando aplicaciones móviles, sino que basta con solo saber Javascript y React, los cuales son muy usados en el desarrollo web.

En lugar de ejecutar React en el navegador y representar ‘divs’ y ‘texts’, React Native se ejecuta en una instancia incrustada de JavaScriptCore (iOS) o V8 (Android) dentro de las aplicaciones y renderiza en una plataforma específica los componentes de un nivel superior (Danielsson, 2016b).

Por otra parte, las tecnologías de geolocalización se han establecido en nuestra sociedad gracias al masivo incremento en el uso de los dispositivos móviles con GPS (Manoharan, 2009), lo que ha permitido la generación masiva de información georeferenciada (Jeansoulin, 2016). Adicionalmente, los sistemas basados en localización (LBS) son servicios que adquiere una persona cuando posee un dispositivo móvil (junto con alguna aplicación que ofrezca información según la localización actual) y opera utilizando una tecnología de sistemas de información geográfica, una de posicionamiento (como GPS) y una de comunicación de redes (Gaitan & Moreno, 2010; Virrantaus et al., 2001). Los mapas digitales y las aplicaciones basadas en la localización son unas de las tecnologías que más han avanzado en los últimos años. La aparición de diferentes APIs y servicios para la creación o integración de estas

tecnologías en páginas web o aplicaciones, ha provocado una proliferación de productos basados en ellas (Leal et al., 2006). Google Maps, es la más potente y actualizada plataforma de servicios de mapas digitales, ya que además de ofrecer el conocido servicio homónimo a partir de un servidor de aplicaciones de mapas web desplazables, liberó en 2005 su API basada en AJAX y JavaScript, permitiendo a terceros desarrolladores crear nuevos productos basados o incorporando parte de sus servicios. El uso de la API de Google Maps es gratuito, aunque presenta ciertos límites de uso, en donde se debe implementar un método de pago de uso dentro de la plataforma de Google Cloud Platform cuando sobrepase su límite gratuito. A pesar de eso, se trata sin duda, de la más potente y actualizada plataforma que se haya desarrollado en la actualidad.

Por otro lado, Pubnub es una plataforma web que permite la comunicación de datos entre dispositivos en tiempo real por medio de sus claves de API, y puesto que la principal funcionalidad de la aplicación móvil se basa en la geolocalización de buses, se usan dichas claves gratuitas (similares a los de Google Cloud Platform) para la respectiva representación de datos de ubicación de cada dispositivo celular de los operadores de buses, con la principal diferencia de que esta plataforma permite además el manejo de usuarios (operadores/conductores de buses por ser más específico) para cada inicio de sesión dentro de la aplicación. Dicho servicio presenta un límite de gratuidad (1 millón de solicitudes), de tal forma de que, si se sobrepasa el límite de solicitudes mensuales cuando los operadores de buses comparten su ubicación por medio de su dispositivo celular, este servicio queda fuera de servicio. Por ende, para que dicho servicio se encuentre disponible la mayor parte del tiempo hacia la comunidad, es necesario implementar de manera preventiva una forma de pago, como, por ejemplo, una tarjeta de crédito.

Dentro de los parámetros de evaluación del software para el desarrollo de aplicaciones se evalúa la disponibilidad que se describe como la probabilidad de que un sistema o servicio funcione correctamente en una instancia de tiempo específica. Dentro de la disponibilidad de la aplicación se describen dos tipos de disponibilidad para la evaluación de esta métrica:

- *Disponibilidad de la aplicación:* la disponibilidad de la aplicación en su principal funcionalidad de localización aproximada por GPS implica la utilización del tráfico y los errores presentados en la utilización de la API Directions como métricas en el cálculo de la disponibilidad durante el mes de diciembre de 2021.
- *Disponibilidad de la distribución de la aplicación:* hace referencia a la forma en que la aplicación móvil debe estar disponible para las personas que quieran acceder a los servicios que esta posee, independientemente del tiempo y el

momento que lo requieran, por medio de la Google Play Store durante el período diciembre 2021 a enero 2022.

La disponibilidad propiamente dicha es el cociente entre el tiempo disponible para producir y el tiempo total de parada. Para calcularlo, es necesario obtener el tiempo disponible, como resta entre el tiempo total, el tiempo por paradas programada (tiempos de descanso) y el tiempo por parada no programada (errores en solicitudes). Una vez obtenido se divide el resultado entre el tiempo total del periodo considerado. Según la Information Technology Infrastructure Library (ITIL), la fórmula para el cálculo de la disponibilidad viene dada por:

$$\text{Disponibilidad} = \frac{\text{Tiempo de Servicio Acordado} - \text{Tiempo de Inactividad}}{\text{Tiempo de Servicio Acordado}} * 100\%$$

En donde:

- El tiempo de servicio acordado es el tiempo esperado que el servicio está en funcionamiento.
- El tiempo de inactividad es la cantidad de tiempo durante el tiempo de servicio acordado que el servicio no está disponible. Para este trabajo el valor de tiempo de inactividad está dado por el tiempo inactivo del servicio de geolocalización de una unidad de autobús ocasionado por los errores en las solicitudes al servidor que puedan aparecer.

Con la finalidad de poner a disposición la información de las rutas del transporte público a la comunidad de Riobamba, se desarrolla una aplicación móvil nativa, usando el framework React Native y siguiendo la metodología MOBILE-D, capaz de brindar a los ciudadanos, una alternativa de información acerca de las rutas y paradas del transporte público urbano, realizando un respectivo seguimiento de las unidades de transporte mediante la asistencia GPS, de tal forma que le indique al usuario información en tiempo real en el que una unidad de autobús llegue hacia una parada cercana. Todo esto preservando la disponibilidad y seguridad de la información de los respectivos usuarios y operadores.

### Metodología

La metodología de desarrollo usada en el proyecto fue MOBILE-D, aunque no es de las más conocidas dentro del desarrollo de software, es de las más efectivas en el ámbito del desarrollo móvil gracias a su buena planificación y comunicación con el cliente. Por otra parte, facilita adaptarse fácilmente a los cambios que puedan aparecer durante el desarrollo del proyecto. La metodología consta de 5 etapas descritas a continuación:

#### *Fase de exploración*

En esta fase, se define el proyecto a realizar y las partes involucradas del proyecto como sus tareas, roles, requisitos previos, objetivos, etc.

- Roles del proyecto:
  - Líder de Proyecto
  - Equipo de desarrollo
  - Usuarios de la aplicación: Operadores de buses y personas de toda edad que dispongan de un dispositivo móvil en la comunidad de Riobamba.
- Requisitos Previos:
  - Acceso a la información de rutas y paradas en coordenadas de latitud y longitud de todo el transporte público urbano.
  - Acceso a la información de los operadores de buses y sus respectivas unidades.
- Objetivos:
  - Diseñar las API's necesarias para controlar la información de rutas y tiempos de llegada.
  - Diseñar los módulos informativos de la aplicación.
  - Consultar si la cédula del operador corresponde a los registros en la base de datos y permitir el ingreso al módulo de compartir ubicación.
  - Notificar el tiempo aproximado de llegada a una parada cuando el usuario seleccione una de estas.
- Alcance:
  - Prototipo funcional de una aplicación móvil Android que disponga la información necesaria sobre las rutas, paradas y tiempos de llegada del transporte público urbano de la ciudad de Riobamba.
- Limitaciones:
  - Los datos serán almacenados en una base de datos MySQL.
  - La conexión entre la aplicación y la base de datos será por medio de un servicio web.
  - La aplicación será desarrollada únicamente para el sistema operativo Android.
  - La aplicación funciona con conexión a internet.

- Los datos mostrados corresponden únicamente a la zona rural y urbana de la ciudad de Riobamba.

Adicionalmente, se realiza el análisis de riesgos para determinar las amenazas que los objetivos del proyecto pueden sufrir, tomando medidas anticipadas que garanticen que el sistema se desarrolle de manera satisfactoria.

#### *Fase de inicialización*

En esta fase toda la información recabada sobre el proyecto se reúne y se establece el entorno técnico como los recursos físicos, tecnológicos y de comunicaciones.

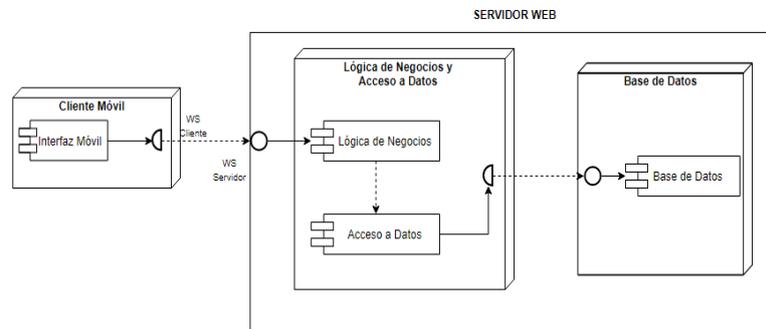
A continuación, se lista el entorno técnico y físico del proyecto:

- Documento de requerimientos iniciales: Documento de Requerimientos Funcionales y No Funcionales.
- Tecnología: Android
- Lenguaje de Programación: JavaScript
- Librerías: Expo Cli (SDK v44.0), Google Maps API for React Native v0.29.0
- Framework: React Native
- IDE: Visual Studio Code v1.57
- Sistema Gestor de Base de Datos: MySQL
- Sistema Operativo: Android v5.0 o superior
- Equipos:
  - 1 servidor XAMPP de Aplicaciones
  - 1 servidor de Base de Datos MySQL
  - 1 laptop con procesador 4 núcleos a más, 4 GB de RAM y con espacio mínimo disponible en Disco de 100GB
- Metodología de Desarrollo: Mobile-D

Se define también la arquitectura de la aplicación, la cual es Modelo, Vista Controlador. En la Figura 1, se presenta la arquitectura de la aplicación móvil RioBuses, la cual tiene como cliente un smartphone con sistema operativo Android versión 5.0 o superior, el teléfono debe contar con acceso a internet pues la aplicación realiza consultas a través de un servicio web en la nube, de disponibilidad 24x7 hacia la base de datos intermedia de la misma.

**Figura 1**

*Diagrama de componentes de la arquitectura de la aplicación*



Adicionalmente, se diseñan los bosquejos de las interfaces de la aplicación mediante la herramienta Balsamiq Wireframes. Finalmente, se establece la planificación de fases con el objetivo de determinar las iteraciones que se va a realizar durante la fase de producción del proyecto, como se muestra en la Tabla 1.

**Tabla 1**

*Planificación de iteraciones por fases*

Fase	Iteración	Descripción
Exploración	Iteración 0	Definición del alcance, establecimiento del proyecto y análisis de riesgos.
Inicialización	Iteración 1	Análisis de requerimientos iniciales, configuración del ambiente de desarrollo, exposición del plan de proyecto, elaboración de prototipos, planificación de fases, obtención de información.
Producción	Iteración Servicios Web	Desarrollo de los servicios web mediante PHP con el uso del framework Laravel que permitan una comunicación con la base de datos.
	Iteración Módulo Informativo	Desarrollo del módulo Informativo que corresponde a la pantalla de Inicio, la pantalla Rutas de visualización de las rutas del transporte público, la pantalla Ayuda y la pantalla Acerca De, refinamiento de interfaces, generación y ejecución de pruebas de aceptación
	Iteración Módulo Encontrar un Bus	Desarrollo del módulo Encontrar un Bus que corresponde a la captura de datos de ubicación recibidos por parte de los operadores para el cálculo de tiempo aproximado de llegada hacia una parada de autobús, refinamiento de interfaces, generación y ejecución de pruebas de aceptación.

**Tabla 1**
*Planificación de iteraciones por fases (continuación)*

Fase	Iteración	Descripción
Producción	Iteración Módulo Operador de Bus	Desarrollo del módulo Operador de Bus el cual consta de los módulos Login y Compartir Ubicación, que corresponden al envío de datos de ubicación hacia el servidor en tiempo real de manera continua, refinamiento de interfaces, generación y ejecución de pruebas de aceptación.
	Iteración Módulo Administrador	Desarrollo del módulo Administrador que corresponde a la gestión de operadores de buses mediante una página web de acceso restringido, refinamiento de interfaces, generación y ejecución de pruebas de aceptación.
Estabilización	Iteración Módulo Informativo	Refactorización del módulo Informativo, refinamiento de interfaces.
	Iteración Módulo Encontrar un Bus	Refactorización del módulo Encontrar un Bus, refinamiento de interfaces.
	Iteración Módulo Operador de Bus	Refactorización del módulo Operador de Bus, refinamiento de interfaces.
	Iteración Módulo Administrador	Refactorización del módulo Administrador, refinamiento de interfaces.
Pruebas del Sistema	Iteración Pruebas del Sistema	Evaluación de Pruebas y se realiza la respectiva reparación del sistema.

*Fase de producción*

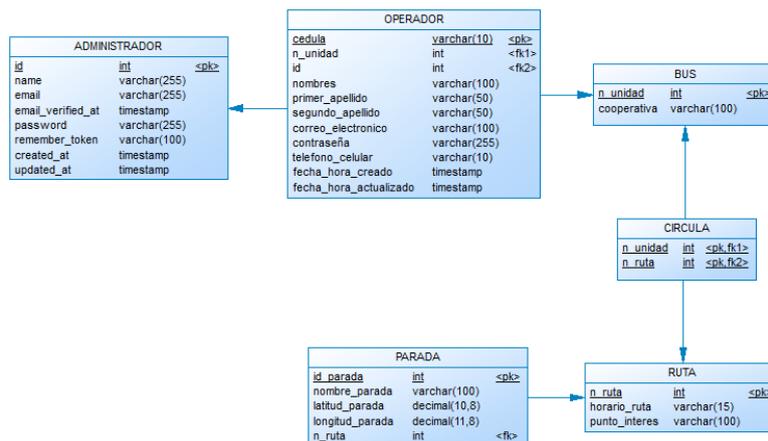
En esta fase del proyecto se presenta las actividades realizadas con la metodología Mobile-D, en base a las iteraciones planificadas en la fase anterior. Con el objetivo de que la codificación generada permita que otros desarrolladores comprendan el código con más rapidez, y a la vez realizar las respectivas correcciones o mantenimiento con la mayor facilidad posible, se ha seleccionado el estándar Snake Case para el diseño de la base de datos y el estándar Pascal Case para el código de la aplicación.

Se establece además el diseño de la base de datos que es donde será almacenada toda la información de forma clara y concisa, cumpliendo los requerimientos establecidos dentro del GADM de Riobamba, de esta manera se refleja las operaciones relacionales para un fácil mantenimiento y actualizaciones cuando se requiera.

Para el respectivo modelado se utiliza el estándar ANSI-SPARC el cual define tres modelos para el proceso de modelado de la base de datos: modelo conceptual, modelo lógico y modelo físico. El modelo de datos lógico que se presenta en la Figura 2, indica cómo se construye el modelo en la base de datos, representando todas las características de una tabla, como los son el nombre de columna, el tipo de datos de columna, las restricciones de columna, la clave primaria, la clave foránea y las relaciones entre las tablas.

Figura 2

Modelo lógico de la base de datos



Por otro lado, con finalidad de describir la navegabilidad y las respectivas conexiones entre las pantallas principales de la aplicación a desarrollar, se estableció un esquema de navegabilidad del sistema.

Finalmente, se describen las Stories Cards, las cuales definen la tarea que se realizó para que se cumpla una determinada funcionalidad, indicando características como el tipo, la dificultad, la prioridad, fecha de creación, etc. Dichas stories cards representan el desarrollo de cada módulo descrito anteriormente en la planificación de fases.

### Fase de estabilización

En esta etapa se procede a integrar las funcionalidades y las características implementadas y de presentarse algún error se realizan las respectivas correcciones. Dichas correcciones se las realiza mediante las tablas de mejora, como se presenta en la Tabla 2.

**Tabla 2**

*Tabla de mejora para el módulo informativo*

Tema de mejora: Diseño de Interfaz			
Responsable: Desarrollador			
Problemas	Acciones	Seguimiento del plan	Realización
El color de la aplicación no se adapta al color del logo.	Establecer un mismo color tanto para la aplicación como para el logo.	Revisar que los nuevos logos estén acordes con el tamaño y colores de la pantalla.	El usuario podrá apreciar un color y logos más atractivos.
Logos principales en la pantalla Inicio y en el menú Lateral no son los adecuados.	Diseñar nuevos logos adecuados a los tonos de color de la aplicación.	Revisar que las imágenes del Slider se adecúen a los colores de la aplicación.	El usuario podrá visualizar unas imágenes acordes al turismo de la ciudad.
Las imágenes del Slider no son atractivas.	Cambiar las imágenes del Slider por otras más nítidas.	Revisar que el cajón deslizable pueda contener toda la información de la línea.	El usuario podrá apreciar una información extra de cada ruta
La pantalla Rutas no contiene la suficiente información de cada línea.	Añadir un cajón deslizable ubicando información específica de cada línea.		

El resultado de las correcciones realizadas en el diseño en la interfaz se muestra en la figura 3.

**Figura 3**

*Módulo informativo corregido*



### *Fase de pruebas y reparación del sistema*

En esta fase se validan las funcionalidades de la aplicación móvil contra los Stories Cards y se corrigen los errores encontrados mediante las pruebas funcionales. Dichas pruebas funcionales permiten verificar el cumplimiento de los requerimientos funcionales de las establecidas en el proyecto, determinando su resultado esperado y comparándolo con el resultado obtenido. En la Tabla 3 se presenta la prueba funcional realizada para comprobar que el usuario puede visualizar el tiempo aproximado de llegada de una unidad de autobús hacia una parada.

**Tabla 3**

#### *Prueba funcional módulo encontrar un Bus*

Código	Nombre
PF04	El usuario podrá realizar una consulta de tiempo aproximado de llegada de unidad de autobús hacia una parada seleccionada
Objetivo	Comprobar que el usuario pueda realizar una consulta de tiempo aproximado de llegada de una unidad hacia una parada seleccionada.
Pasos	<ol style="list-style-type: none"> <li>1. Verificar que tenga conexión a internet</li> <li>2. Acceder a la pantalla 'Encontrar un Bus'</li> <li>3. Seleccionar una línea de autobús</li> <li>4. Seleccionar una parada</li> <li>5. Presionar el botón 'Localizar Bus'</li> </ol>
Resultados Esperados	El usuario podrá visualizar un mensaje con el tiempo aproximado de llegada en minutos de una unidad de autobús.
Resultados Obtenidos	Una alerta indicando el tiempo aproximado de llegada en minutos de una unidad de autobús hacia la parada seleccionada, siempre y cuando exista un operador de bus compartiendo su ubicación.

Por otro lado, se ha puesto a disponibilidad de todos los ciudadanos de Riobamba y a los respectivos operadores del transporte público la aplicación Riobuses en la tienda de Google Play, por lo que se optó por adquirir una cuenta de desarrollador de Google Play Console, la cual sirve como base para el proceso de publicación en dicha tienda de aplicaciones.

### **Resultados**

Para el cálculo de los porcentajes de disponibilidad de la aplicación, se aplica la fórmula establecida por la *Information Technology Infrastructure Library (ITIL)*, misma que estima qué tan bien se desempeña el servicio en el futuro. Por lo tanto, para determinar el porcentaje de tiempo en que la información de una unidad de autobús se encuentra disponible, se realiza un análisis de datos en base a los registros y métricas en cuanto a errores de solicitudes en el uso de mapas durante los horarios laborables del mes de

diciembre de 2021; y a su vez, para determinar el porcentaje de tiempo en que la aplicación se distribuye correctamente en los dispositivos móviles se toma en cuenta la cantidad de fallos y errores de tipo *ANR (Application Not Responding)* que pueden surgir en el arranque de la aplicación durante los meses de diciembre de 2021 y enero de 2022, mediante el acceso a los registros que guarda *Google Cloud Platform*.

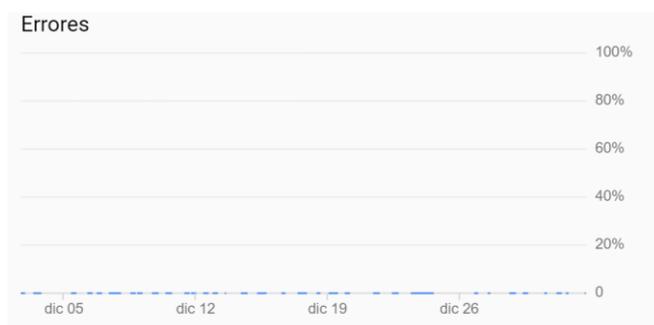
Para la evaluación de la disponibilidad de la aplicación se toma en cuenta la principal funcionalidad en la aplicación desarrollada, la cual es el servicio de ubicación de buses que ofrece la misma, en donde se toma como métrica los errores en solicitudes de mapas puesto que son en donde se muestra la información de las unidades de autobús que están disponibles en el día laborable, y que, de existir algún error en alguna solicitud, este influye directamente en el tiempo en el que el servicio de información de ubicación de una unidad se encuentra disponible.

La métrica de errores tiene como finalidad mostrar la cantidad de errores en las peticiones o solicitudes enviadas hacia el servidor por parte del cliente, los cuales influyen fuertemente en el servicio de disponibilidad de información brindada por la aplicación. El porcentaje de esta métrica es proporcionada por la herramienta *Google Cloud Platform*, y para el cálculo respectivo se ha tomado los valores del mes de diciembre del 2021, mes en el que fue implementada la aplicación, evaluados en los 59 dispositivos que descargaron la aplicación.

En la figura 4 se muestra el diagrama de líneas de los errores en solicitudes correspondientes al mes de diciembre 2021, en donde se puede apreciar la ausencia de errores ocasionados en la recepción de solicitudes por el servidor.

**Figura 4**

*Diagrama de líneas de errores en solicitudes*



Tomado en cuenta que las horas laborables de un operador de bus corresponden entre el lapso de las 06:00 am y las 20:00 pm (14 horas diarias), además de que los días laborables corresponden de lunes a domingo (7 días por semana / 30 días mensuales) y que existen 16 líneas de recorrido para los buses, se obtiene un tiempo de servicio de

6.720 horas mensuales. Por otro lado, existe un tiempo de descanso en donde el operador de bus toma un descanso de 10 minutos al terminar una vuelta, es decir 10 minutos de inactividad por cada vuelta recorrida en la línea de trabajo.

Por lo tanto, el tiempo total de descanso de todas las líneas en un solo día es igual a 1.110 min (18,5 horas), equivalente a 555 horas mensuales, lo que restado de las 6.720 horas mensuales del servicio se obtiene un tiempo de servicio acordado de 6.165 horas mensuales, en los cuales los operadores de buses mantienen su servicio de transporte activo.

Adicionalmente, como se observa en el Gráfico 1, se tiene que durante el mes de diciembre del 2021 tampoco hubo errores en sus solicitudes del servidor, por lo que se tiene un tiempo de inactividad de 0 horas mensuales; en donde, utilizando la fórmula de disponibilidad se puede calcular lo siguiente:

$$\text{Disponibilidad} = \frac{\text{Tiempo de Servicio Acordado} - \text{Tiempo de Inactividad}}{\text{Tiempo de Servicio Acordado}} * 100\%$$

$$\text{Disponibilidad Mensual} = \frac{6165 \text{ horas} - 0 \text{ horas}}{6165 \text{ horas}} * 100\%$$

$$\text{Disponibilidad Mensual} = \frac{6165 \text{ horas}}{6165 \text{ horas}} * 100\%$$

$$\text{Disponibilidad Mensual} = 100\%$$

Por otro lado, se tiene que un solo dispositivo móvil de un operador de bus ocupa una cantidad aproximada de 42.000 solicitudes mensuales hacia el servidor de localización Pubnub mediante una cantidad aproximada de 4.926 Gigabytes mensuales de uso en datos móviles, lo que implica un gran consumo de solicitudes e internet por cada operador de bus.

Por ende, con el cálculo realizado, se puede determinar que el porcentaje de disponibilidad mensual de la aplicación móvil Riobuses evaluado en la funcionalidad de información de ubicación de los dispositivos celulares de las unidades de autobús, mediante un gasto mensual de 4.92 Gigabytes de datos móviles, es del 100%; puesto que el servidor de geolocalización utilizado para este proyecto es de confianza y abarca grandes cantidades de solicitudes a la vez, por lo que brinda riesgo mínimo de errores en sus solicitudes y por ende un tiempo mínimo de inactividad mensual; sin embargo, no se puede llegar a una conclusión general puesto que esta disponibilidad se ha evaluado únicamente durante un mes de prueba, en donde las circunstancias han sido coincidentemente propicias para este estudio.

El nivel de disponibilidad de la distribución de la aplicación está relacionado con el alojamiento de esta para su descarga por parte de los ciudadanos, el cual depende de la

plataforma en la que se aloje, puesto que es aquella la que brinda el servicio de descarga e instalación de las aplicaciones existentes, y a su vez es el encargado de mantener sus servidores trabajando 24/7.

Por lo tanto, se sabe que el tiempo disponible de la aplicación móvil dentro de la Google Play Store es de 24 horas y por 7 días a la semana se tiene un tiempo de servicio acordado de 720 horas mensuales; además de que durante los meses de diciembre de 2021 y enero de 2022 no hubo fallas en sus servicios de acceso y descarga de aplicaciones, se tiene un tiempo de inactividad de 0 horas mensuales, por lo que, utilizando la fórmula de disponibilidad brindada por la *Information Technology Infrastructure Library* (ITIL) se puede calcular lo siguiente:

$$\text{Disponibilidad} = \frac{\text{Tiempo de Servicio Acordado} - \text{Tiempo de Inactividad}}{\text{Tiempo de Servicio Acordado}} * 100\%$$

$$\text{Disponibilidad Mensual} = \frac{720 \text{ horas} - 0 \text{ horas}}{720 \text{ horas}} * 100\%$$

$$\text{Disponibilidad Mensual} = \frac{720 \text{ horas}}{720 \text{ horas}} * 100\%$$

$$\text{Disponibilidad Mensual} = 100\%$$

Por ende, con el cálculo anterior se puede deducir que el porcentaje de disponibilidad de distribución de la aplicación móvil Riobuses, con una cantidad de 118 descargas de la aplicación por medio de la tienda de aplicaciones Google Play Store, es del 100%, puesto que el servicio de alojamiento de esta tienda es altamente competente para que cualquier usuario con un dispositivo Android pueda adquirir dicha aplicación en el momento que lo requiera.

Sin embargo, a pesar de que la aplicación se encuentra 100% disponible ante la ciudadanía por medio de la tienda de aplicaciones Google Play Store, existen otros factores que afectan netamente en el servicio que se ofrece para los dispositivos móviles, como lo son la cantidad de fallos y los ANR (cantidad de veces que la aplicación muestra el error La Aplicación No Responde), los cuales pueden aparecer al momento de ejecución de la aplicación. Y aunque el servicio de la principal funcionalidad de la aplicación (geolocalización de buses) haya estado disponible únicamente por un mes de prueba, esto no ha sido motivo para que aquellas personas que deseen obtener la aplicación no puedan acceder a esta, ya que durante el mes de enero de 2022 existieron registros de adquisición de usuarios nuevos.

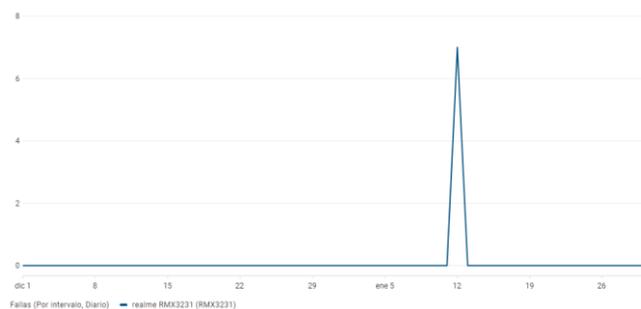
Dicha información referente a los fallos en la interacción de la aplicación con el dispositivo móvil del usuario es brindada por el propio *Google Cloud Platform*, puesto que posee registros de métricas de todas las aplicaciones en su disposición, tales como los antes mencionados Fallos en el arranque de la aplicación y ANR (mensajes de La

Aplicación No Responde), los cuales permiten analizar el estado del sistema en un determinado dispositivo móvil.

En la figura 5 se puede evidenciar que surgieron 7 fallas en un dispositivo en un determinado día, más no ANRs, es decir únicamente no se realizó un correcto arranque de la aplicación en 7 intentos realizados por el usuario del dispositivo móvil perteneciente al modelo ‘Realme RMX3231’.

**Figura 5**

*Diagrama de líneas número de fallas durante diciembre 2021 – enero 2022*



Estos fallos permiten apreciar que no todas las aplicaciones son compatibles a la primera con todos los dispositivos móviles, por lo que *Google Cloud Platform* se da la tarea de calcular el porcentaje promedio de fallos para cada dispositivo, desde el día del mes en el que surgió la falla, o hasta que se suba una nueva versión/actualización con su respectiva corrección y se analice nuevamente. El promedio de fallas se lo puede observar en la figura 6.

**Figura 6**

*Diagrama de líneas promedio móvil de fallas desde 12 enero 2022*



Adicionalmente en la figura 6 se puede observar que solamente surgieron fallas después del mes de prueba de la aplicación, puesto que aún no se evaluaba aquel modelo de dispositivo que registró fallas durante el mes de diciembre 2021, es decir, que a mediados del mes de enero del 2022 aquellas fallas obtenidas otorgan un porcentaje promedio de fallas del 0.233% para aquel modelo de dispositivo; ahora, dicho valor afecta directamente al porcentaje de disponibilidad de distribución de la aplicación del

mes anteriormente calculado, ya que ésta se encuentra relacionada con la adquisición y pérdida de los usuarios durante la finalización del proyecto, por lo que, restado del anterior porcentaje calculado, se obtiene un nuevo valor de 99.767% de disponibilidad de la distribución de la aplicación para el mes de enero del 2022.

### Discusión

Los resultados obtenidos en la presente investigación son favorables debido a que se ha cumplido con el desarrollo de una aplicación web que ayuda a solventar el problema correspondiente a la falta de acceso a la geo información del transporte público. Aunque han existido propuestas de aplicaciones para la geolocalización de rutas de buses en otras ciudades (Almeida & Solís, 2019; Díaz, 2022; Sánchez, 2020; Sandoval, 2020), ninguna de estas ha realizado un análisis profundo referente al rendimiento de la aplicación, ni un análisis de riesgos como lo que se muestra en la presente investigación. Por lo tanto, la presente investigación puede servir de base para el desarrollo de futuras aplicaciones móviles enfocadas a solucionar problemas del transporte. Adicionalmente en estudios similares no se profundiza en la metodología aplicada y no se indica a detalle cada una de las fases de la metodología. Es importante tomar en cuenta que la aplicación puede funcionar para solventar el problema de falta de geo información en otras localizaciones siempre y cuando las entidades encargadas del transporte público urbano dispongan de la información correspondiente que pueda servir para alimentar a la base de datos de la aplicación propuesta.

### Conclusiones

- Se ha desarrollado una aplicación móvil cuyos usuarios son los ciudadanos de la ciudad de Riobamba y que tiene el fin de proveer suficiente información referente al transporte público para facilitar su acceso a todos los usuarios, de tal forma que se aumente su demanda y se disminuya la necesidad de transporte público en la ciudad.
- El sistema gestor de base de datos MySQL se emplea como plataforma base en el uso y manejo de la base de datos debido a la facilidad y confiabilidad que esta ofrece en el salvaguardado de la información personal que cada operador de bus posee, tales como su número de identificación personal, nombres completos, correo electrónico, contraseña de acceso, etc., y con el fin de que esta sea gestionada por un único administrador.
- Para el desarrollo de la aplicación móvil se utilizó la metodología Mobile-D por medio del *framework React Native*, debido a la fácil compatibilidad en el desarrollo de software para dispositivos Android con los servicios de Google *Maps*, todo esto mediante fases y actividades, de tal manera que se construya el software esperado de acuerdo con los requisitos establecidos por la

Municipalidad de Riobamba. Por lo tanto, la aplicación permite hacer un seguimiento por geolocalización de las unidades de autobús de la ciudad con el fin de estimar el tiempo aproximado de llegada hacia una parada.

- La distribución gratuita de la aplicación hacia la comunidad de Riobamba dentro de la tienda de aplicaciones Google Play Store, permite el acceso a la información que se encuentra dentro de la aplicación sobre el servicio de transporte público de la ciudad a todos los usuarios que lo requieran.
- El porcentaje de disponibilidad de la aplicación evaluada de forma mensual durante el mes de diciembre del 2021 en la principal funcionalidad de la aplicación móvil *Riobuses* y mediante la fórmula de disponibilidad brindada por la *Information Technology Infrastructure Library* (ITIL) es del 100%, lo que permite analizar un alto desempeño en el servicio que ofrece la aplicación para mantener una buena disponibilidad de información al momento que se requiera. Por otro lado, durante una evaluación de la disponibilidad de la distribución de la aplicación durante el mes de enero del 2022, se detectaron fallos en el arranque de la aplicación en un determinado dispositivo los cuales redujeron no muy significativamente el valor de la disponibilidad de distribución calculado en el mes de diciembre de 2021, por lo que, una vez corregido dicho valor porcentual, se tiene un nuevo porcentaje de disponibilidad de la distribución de la aplicación con un valor del 99.767%, lo que permite concluir que la aplicación está válidamente apta para ser distribuida a la mayoría de modelos de dispositivos móviles Android con excepción de uno, solamente hasta que se otorgue la debida solución al error.

### Referencias bibliográficas

- Almeida Muñoz, J. F., & Solís Cuñez, S. M. (2019). *Desarrollo de una aplicación móvil Android para la consulta de rutas de una línea de buses urbanos que circulan por la ciudad de Quito referenciando los puntos de partida y destino del usuario*. Universidad Politécnica Salesiana. <https://dspace.ups.edu.ec/handle/123456789/16912>
- Danielsson, W. (2016a). *React Native application development: A comparison between native Android and React Native*. <https://www.diva-portal.org/smash/record.jsf?pid=diva2:998793>
- Danielsson, W. (2016b). *React Native application development: A comparison between native Android and React Native*. *Linköpings Universitet*. <https://www.diva-portal.org/smash/record.jsf?pid=diva2:998793>

- Danielsson, W. (2018). *React Native Performance Evaluation*. <https://aaltodoc.aalto.fi/handle/123456789/32475>
- Díaz Concha, A. S. (2022). *Geolocalización de Rutas de Buses en tiempo Real: Caso de estudio Ciudad de Otavalo*. Universidad Politécnica Salesiana. <https://dspace.ups.edu.ec/handle/123456789/22102>
- Gaitan, A., & Moreno, J. (2010). Construcción de una arquitectura que provea servicios de información y/o publicidad a dispositivos móviles basados en su ubicación georeferenciada - CORE Reader. In *Pontificia Universidad Javeriana*. <https://core.ac.uk/reader/71418960>
- Hansson, N., & Vidhall, T. (2016). *Effects on performance and usability for cross-platform application development using React Native*. <https://www.diva-portal.org/smash/record.jsf?pid=diva2:946127>
- Haro, C. A. V., Torres, S. C. V., & Haro, V. E. V. (2019). Diseño de un sistema electrónico para el bloqueo de vehículos de transporte público basados en la medida de alcohol por litro de sangre en los conductores. *Ciencia Digital*, 3(3.3), 220–230. <https://doi.org/10.33262/cienciadigital.v3i3.3.801>
- Jácome, M. G. M., Jarro, J. A. P., Armijo, L. A. U., & Mora, J. C. M. (2019). Evaluación de la contaminación acústica en el terminal terrestre del cantón Morona, ciudad Macas mediante la identificación de niveles de presión sonora. *Ciencia Digital*, 3(3.1), 253–269. <https://doi.org/10.33262/cienciadigital.v3i3.1.699>
- Jeansoulin, R. (2016). Review of Forty Years of Technological Changes in Geomatics toward the Big Data Paradigm. *ISPRS International Journal of Geo-Information* 2016, Vol. 5, Page 155, 5(9), 155. <https://doi.org/10.3390/IJGI5090155>
- Kaushik, V., Gupta, K., & Gupta, D. (2019). React native application development. *International Journal of Advanced Studies of Scientific Research*. [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3330011](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3330011)
- Leal, A., Gómez, J. D., & Castañeda, H. A. (2006). Proveedor de Servicios Basados en Localización para Dispositivos Móviles. *Revista Avances en Sistemas e Informática*, 3, 13–19. <https://www.redalyc.org/articulo.oa?id=133114991003>
- Manoharan, S. (2009). On GPS tracking of mobile devices. *Proceedings of the 5th International Conference on Networking and Services, ICNS 2009*, 415–418. <https://doi.org/10.1109/ICNS.2009.103>
- Neftali, R., Calixto, L., Ángel, L., González, V., Esteban, D., Díaz, B., & Venegas Guzmán, R. (2019). React Native: acortando las distancias entre desarrollo y

diseño móvil multiplataforma. *Revista Digital Universitaria UNAM*.  
<https://doi.org/10.22201/codeic.16076079e.2019.v20n5.a5>

Sánchez Vera, I. (2020). *Desarrollo de sistema de mapeo y visualización de rutas de buses urbanos de la provincia de Santa Elena para la Agencia Nacional de Tránsito. Módulo: Gestor web*. Universidad Estatal Península de Santa Elena.  
<https://repositorio.upse.edu.ec/handle/46000/5322>

Sandoval Royero, O. J. (2020). *Prototipo de aplicación móvil para consultar rutas de buses urbanos en Villavicencio*. Repositorio institucional.  
<https://repository.uniminuto.edu/handle/10656/11570>

Söderberg, J., Johansson, E., Persson, D., & Foo, J. (2018). *Evaluating performance of a React Native feature set*. <https://www.diva-portal.org/smash/record.jsf?pid=diva2:1215795>

Suárez, M. A. C., Cunalata, C. P. M., Robalino, O. E. R., Pérez, J. G. V., Cárdenas, J. C. S., & Almeida, E. L. S. (2018). Estimación cuantitativa y cálculo de emisiones ambientales (huella de carbono), en el terminal terrestre de la ciudad de Guaranda. *Ciencia Digital*, 2(4), 283–293. <https://doi.org/10.33262/cienciadigital.v2i4.229>

Virrantaus, K., Markkula, J., Garmash, A., Terziyan, V., Veijalainen, J., Katanosov, A., & Tirri, H. (2001). Developing GIS-supported location-based services. *Proceedings of the 2nd International Conference on Web Information Systems Engineering, WISE 2001*, 2, 66–75. <https://doi.org/10.1109/WISE.2001.996708>

Moreno, P., Bastidas, G., & Moreno Costales, P. (2020). Estudio de factibilidad del uso de modelos de redes neuronales artificiales en la automatización del aforo y clasificación vehicular del transporte público. *ConcienciaDigital*, 3(3), 528–540. <https://doi.org/10.33262/concienciadigital.v3i3.1355>

### Conflicto de intereses

Los autores deben declarar si existe o no conflicto de intereses en relación con el artículo presentado.

El artículo que se publica es de exclusiva responsabilidad de los autores y no necesariamente reflejan el pensamiento de la **Revista Conciencia Digital**.



El artículo queda en propiedad de la revista y, por tanto, su publicación parcial y/o total en otro medio tiene que ser autorizado por el director de la **Revista Conciencia Digital**.



#### Indexaciones

