Intelligent Public Transportation Monitoring

Public Challenge #1
Technical Report

GovTech Portoviejo 2050
Municipal Policy for GovTech Collaborative Ecosystem

1. Problem Definition

In the field of urban public transportation, users frequently face the lack of real-time information regarding routes and arrival times of the units. This situation becomes especially critical when a bus is missed and it is unknown whether another vehicle with a similar route will arrive within a reasonable time.

The uncertainty generated by these types of events hinders efficient decision-making regarding travel, causing loss of time, stress, and a poor perception of public transportation service.

This project arises as a response to this problem, proposing a technological solution based on Artificial Intelligence (AI) that provides timely, accurate, and accessible information about urban transport routes and frequencies. The aim is to improve the user experience, optimize waiting times, and promote more efficient urban mobility.

2. Project Description

This project proposes the development of an intelligent assistance system for urban mobility, which uses artificial intelligence algorithms to recommend public transportation routes in real time. The core of the system will be a predictive analysis model that processes geolocation data, traffic, and bus frequency to provide the user with the best transfer option at any given time.

The application will be accessible from any mobile device through a QR code strategically placed at stops and transport units. Once scanned, the user will access an optimized website (PWA) where they can enter their destination. From this point, the system will perform the following operations:

- Automatic geolocation of the user through the device's GPS.

- Query of active routes in real time, using open data APIs from transport systems or GPS sensors installed in buses.

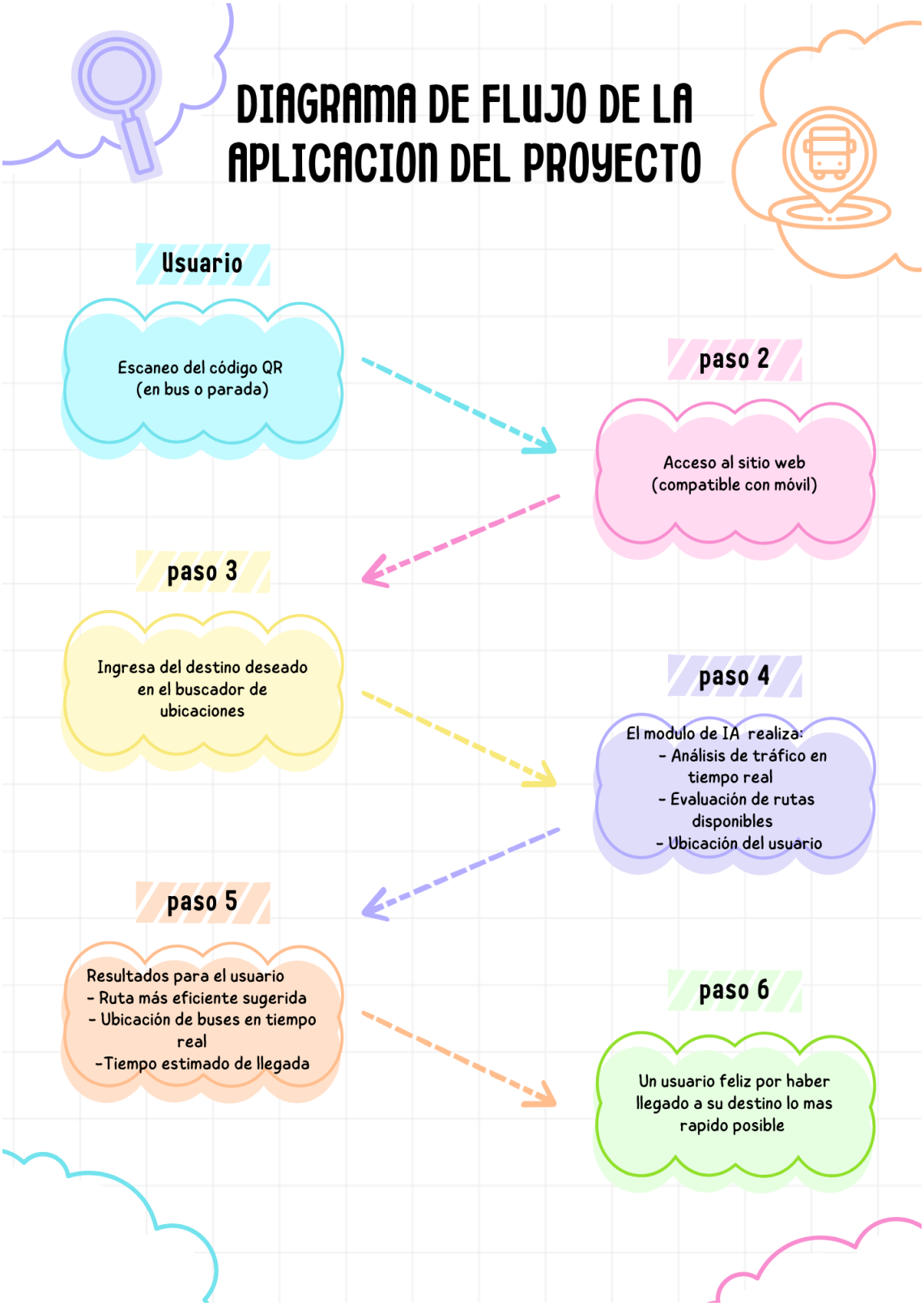
- Application of machine learning models to predict estimated arrival times based on traffic and historical frequency.

- Visualization of available routes, current location of the selected bus, and estimated arrival times at the destination.

The user interface will be developed using modern web technologies (React.js or similar) with dynamic maps integrated through services such as Mapbox or Google Maps API. The artificial intelligence will be implemented in Python using libraries such as Scikit-learn or TensorFlow, hosted on cloud servers (AWS or Azure), allowing agile and scalable responses.

With this system, the goal is to optimize citizens' travel time, increase the reliability of public transportation, and contribute to intelligent urban planning.

3. Diagram or graphic scheme of the prototype



4. Project Implementation Flowchart

5. Requirements

This section details the resources needed for the implementation of the intelligent public transportation route recommendation system. They are divided into human talent, materials, and technological equipment.

5.1. Human Talent Requirements:

The system will be mainly developed by the project team, composed of students who will assume key responsibilities in design, programming, testing, and documentation. To complement specific technical areas, the support of experienced individuals will be required in the following areas:

Área de Apoyo Técnico	Participación requerida
Diseño y desarrollo web (HTML, CSS, JavaScript)	Se contará con el acompañamiento de un programador con experiencia en diseño de interfaces web responsivas y dinámicas, quien apoyará al equipo en la construcción del sitio web del sistema.
Desarrollo de aplicaciones móviles o PWA	Un desarrollador con conocimientos en tecnologías como React.js o similar podrá asesorar en la implementación de una versión adaptable a móviles, así como en la integración con servicios de geolocalización y mapas.
Equipo del proyecto (nosotros)	Los integrantes del proyecto participamos activamente en todas las fases del desarrollo, incluyendo diseño, programación, documentación, pruebas funcionales, análisis de datos y gestión de requerimientos.

This collaborative approach allows students to be directly involved in the development process, while ensuring adequate technical standards through the guidance of experts or mentors in each area.

5.2. Material Requirements:

The system will include an operations center (based on a Raspberry Pi) and physical modules installed in buses for monitoring and data transmission. The necessary materials are divided into two categories:

Operations Center:

Componente	Cantidad	Costo Unitario ($)	Costo Total ($)	Observaciones
Raspberry Pi 5 (8 GB RAM)	1	170.00	170.00	Incluye fuente de alimentación y refrigeración. Ya existente.

Bus Module (Monitoring Board):

Componente	Cantidad	Costo Unitario ($)	Costo Total ($)	Observaciones
Módulo I2C 0.91" OLED 128x32	1	1.96	1.96	Pantalla para mostrar información en el bus.
Módulo GPS GYONE06MV2	1	3.23	3.23	Para geolocalización del bus.
Teclado matricial 3x4	1	2.03	2.03	Entrada de datos manual desde el bus.
Placa de desarrollo ESP32-C3	1	1.70	1.70	Controlador principal del sistema.
Módulo inalámbrico LoRa	1	2.57	2.57	Comunicación de largo alcance.
Cable de alimentación USB tipo C	1	2.00	2.00	Fuente de energía para el módulo.
Carcasa del módulo impresa en 3D	1	2.50	2.50	Protección física del módulo.
Ventosas para fijación	2	0.50	1.00	Fijación del módulo en el bus.

Production Cost (per unit):

Concepto	Cantidad	Costo Unitario ($)	Costo Total ($)
Producción y ensamblaje de módulo	1	8.00	8.00

5.3. Equipment Requirements:

Basic computer equipment will be required to carry out the development, testing, and presentation of the system. The following technological resources are necessary:

Equipo / Herramienta	Cantidad	Especificaciones mínimas
Computadora personal (portátil o de escritorio)	2 – 3	Procesador dual-core o superior, mínimo 8 GB RAM, navegador actualizado, acceso a Internet.
Smartphones para prueba	2	Al menos un dispositivo Android y uno iOS. Deben tener GPS activo y acceso a cámara para escaneo de códigos QR.
Conexión a Internet	1	Para desarrollo, pruebas y despliegue del sitio web. Velocidad mínima recomendada: 10 Mbps.
Navegadores modernos	N/A	Google Chrome, Firefox o Safari actualizados para probar compatibilidad.
Herramientas online (gratuitas)	N/A	Visual Studio Code, GitHub, Netlify/Vercel (para despliegue), Google Maps API (freemium), entre otros.

6. Expected Scenario Description

The system will be implemented in an urban environment, specifically in the context of land-based public transport through buses. Its deployment includes both a fixed operations center and physical modules installed in transport vehicles, which interact with users through a web application accessible via QR codes.

Physical Environment

- *Module location*: Installed in urban buses, preferably in visible and accessible areas for users.

- *Environmental conditions*: Exposure to vibrations, temperature changes, humidity, and public handling must be considered.

- *Fixation*: Suction cups will be used to facilitate temporary installation during field tests.

Operations Center

- *Location*: It will operate in a closed environment such as an office, laboratory, or data center.

- *Functionality*: Receives and processes information sent from the modules and synchronizes it with the web system database. A Raspberry Pi 5 (8 GB) will be used as a low-consumption local server.

- C*onnectivity*: Stable internet access for data reception, route analysis, and remote monitoring.

User Interaction

- *System access*: Users will scan a QR code located on the transport unit or stop, accessing a progressive web interface (PWA) optimized for mobile devices.

- *Expected functionality*: Upon entering the destination, the system will display available routes, estimated arrival time of the next bus, and real-time location of the nearest vehicle.

Necessary Technological Conditions

Recurso	Condición mínima
Internet móvil en buses	Al menos cobertura 3G o 4G para transmisión de datos desde los módulos.
Dispositivos móviles de usuarios	Compatible con navegación web y lectura de códigos QR.
Acceso a servicios de mapas	Uso de Google Maps API o Mapbox para geolocalización e interfaz.
Base de datos y backend	Alojamiento en la Raspberry Pi local o servidor en la nube (escalable según necesidad).

7. References:

CEPAL. (2021). Sustainable Urban Mobility in Latin America and the Caribbean. Economic Commission for Latin America and the Caribbean. https://www.cepal.org/es/publicaciones/46630-movilidad-urbana-sostenible-america-latina-caribe

UITP. (2019). Digitalisation and Mobility as a Service (MaaS). International Association of Public Transport. https://www.uitp.org/publications/digitalisation-and-mobility-as-a-service-maas/