

Design of an urban vehicle tracking prototype for the Zacatenco Smart Campus under the GTFS standard

Jorge-Diego Robles-Morales BsC, Federico Felipe-Duran, Eng², and Miguel Sánchez-Meraz MsC³,
¹Instituto Politécnico Nacional, México, jproblem1400@alumno.ipn.mx, ffelipe@ipn.mx², mmeraz@ipn.mx³

Resumen– Este trabajo presenta un prototipo IoT para el seguimiento vehicular y utiliza parte del estándar GTFS para presentar a usuarios de una página Web la información en tiempo real de vehículos de transporte público que realizan recorridos en el Campus universitario Zacatenco del Instituto Politécnico Nacional de México. El nodo IoT se comunica utilizando tecnología inalámbrica LoRa con un servidor en donde se guarda la información del estándar GTFS, así como la de las rutas de los vehículos. El nodo IoT está implementado con una placa Arduino MKR WAN 1300 y un módulo GPS MKR Shell

Palabras clave- GTFS, IoT, LoRa, Arduino MKR WAN 1300.

Abstract - This work presents an IoT prototype for vehicle tracking and uses part of the GTFS standard to present to users of a Web page the real-time information of public transport vehicles that make routes in the Zacatenco University Campus of the National Polytechnic Institute of Mexico. The IoT node communicates using LoRa wireless technology with a server where the GTFS standard information is stored, as well as the routes of the vehicles. The IoT node is implemented with an Arduino MKR WAN 1300 board and a MKR Shell GPS module.

Keywords- GTFS, IoT, LoRa, Arduino MKR WAN 1300.

Prototipo de seguimiento vehicular urbano para el Smart Campus Zacatenco bajo el estándar GTFS

Jorge-Diego Robles-Morales BsC, Federico Felipe-Duran, Eng², and Miguel Sánchez-Meraz MsC³, ³

¹Instituto Politécnico Nacional, México, jroblem1400¹@alumno.ipn.mx, ffelipe@ipn.mx², mmeraz@ipn.mx³

Resumen– Este trabajo presenta un prototipo IoT para el seguimiento vehicular y utiliza parte del estándar GTFS para presentar a usuarios de una página Web la información en tiempo real de vehículos de transporte público que realizan recorridos en el Campus universitario Zacatenco del Instituto Politécnico Nacional de México. El nodo IoT se comunica utilizando tecnología inalámbrica LoRa con un servidor en donde se guarda la información del estándar GTFS, así como la de las rutas de los vehículos. El nodo IoT está implementado con una placa Arduino MKR WAN 1300 y un módulo GPS MKR Shell

Palabras clave- GTFS, IoT, LoRa, Arduino MKR WAN 1300.

Abstract - This work presents an IoT prototype for vehicle tracking and uses part of the GTFS standard to present to users of a Web page the real-time information of public transport vehicles that make routes in the Zacatenco University Campus of the National Polytechnic Institute of Mexico. The IoT node communicates using LoRa wireless technology with a server where the GTFS standard information is stored, as well as the routes of the vehicles. The IoT node is implemented with an Arduino MKR WAN 1300 board and a MKR Shell GPS module.

Keywords- GTFS, IoT, LoRa, Arduino MKR WAN 1300.

I. INTRODUCCIÓN

Las ciudades modernas cuentan con sistemas de transporte masivo como lo son camiones, trolebuses o metros, incluso trenes suburbanos. Si la gente utiliza estos medios de forma adecuada, podrá disminuir sus tiempos de traslado y en consecuencia tendrá más tiempo disponible para su trabajo, descanso, sueño o bien diversiones. Esto sería más sencillo si la gente contará con información precisa sobre las rutas de tales medios de transporte, sus horarios de llegada a los puntos de parada estaciones donde descienden y abordan los pasajeros. Otra información importante es el estatus del desplazamiento de las unidades, ubicación en la ruta, tiempo de llegada al punto de espera del pasajero, ocupación de la unidad entre otros datos. Los gobiernos de las Ciudades, los usuarios de los sistemas colectivos de transporte, organizaciones civiles sin afán de lucro y organizaciones de profesionistas relacionados con este problema realizan esfuerzos de forma continua para lograr desplazamientos más eficientes lo que redundará en comodidad para los usuarios, el uso más eficiente de los sistemas colectivos, en consecuencia, menor consumo de combustible y menor contaminación atmosférica y evitar molestias como la aglomeración de gente esperando por medios de transporte que tal vez demoren en llegar o bien no tengan lugares disponibles, como se puede ver en la Figura 1.



Figura 1. Aglomeración de usuarios de transporte colectivo

Un estándar desarrollado por la organización sin fines de lucro Mobility Data es el General Transit Feed Specification, este estándar fue diseñado para compartir datos de los medios de transporte colectivo en una ciudad, consta de datos que pueden ser explotados por sistemas de software que los utilicen y tengan como salida información utilizada por las partes interesadas en la movilidad urbana.

Otras herramientas tecnológicas se utilizan en una ciudad actual para mejorar la movilidad urbana, El Internet de las Cosas (IoT por sus siglas en inglés), las Tecnologías de la Información las Comunicaciones (TIC's) también se conjugan para mejorar la movilidad, con la información descrita en GTFS, con nodos IoT en las unidades de transporte colectivo del Campus Zacatenco y las TIC's se desarrolló el presente trabajo.

II. MARCO TEÓRICO

El estándar GTFS se originó en los Estados Unidos, en la ciudad de Portland, en donde su organismo público de transporte TriMet en colaboración con Google trataron de resolver la programación de viajes de tránsito utilizando un conjunto de datos abiertos que se pudieran consultar en línea. La ciudad de Portland proporcionaba los datos, Google los procesaba de forma que los usuarios del sistema los pudieran ver y utilizar El estándar se utiliza por ciudades alrededor del mundo, más de cien y cerca de diez mil empresas de transporte, tanto de gobierno como privadas las utilizan [1].

El estándar GTFS describe tablas de datos planas en forma de texto, esta característica permite que los datos que se definen y comparten sean muy sencillos de integrar en cualquier manejador de bases de datos o aplicación que desarrolle con ellos con un tratamiento bastante simple. GTFS

está organizado en dos secciones: GTFS Schedule orientado a la planeación y programación del transporte y el GTFS Realtime orientado a compartir las actualizaciones del desplazamiento del transporte.

GTFS Schedule recomienda la creación de veintitrés archivos que van desde describir las rutas y paradas del transporte público hasta las empresas o agencias propietarias del medio de transporte o puntos de transferencia de diferentes rutas. En cambio, GTFS Realtime está más orientado al estatus del desplazamiento de los vehículos y su posición en el tiempo y posición geográfica. Está compuesto por tres entidades: Actualización o fluctuación de viajes (Trip updates) en su programa propuesto; Alertas de servicio (Service Alerts), describe problemas de un nivel complejo, por ejemplo, si una estación o ruta no estará en servicio por periodos de tiempo extensos y ubicación o posición de los vehículos (Vehicle positions) describe en tiempo real la ubicación de los vehículos en sus rutas establecidas [2][3].

Las Tecnologías de la Información y Comunicaciones abarcan varias que ayudan a una mejor movilidad urbana, las comunicaciones inalámbricas, el desarrollo Web, las bases de datos, las redes 5G de telefonía celular, las aplicaciones para dispositivos móviles principalmente. Es necesario establecer comunicación entre los usuarios de transporte público y las unidades de transporte y las tecnologías mencionadas son parte de todo el sistema. A continuación, se describen:

Redes Inalámbricas. Hoy en día existen funcionando varios sistemas de redes inalámbricas que dan soporte a dispositivos electrónicos instalados, para este trabajo en medios de transporte y que pueden comunicarlos con servidores para manejar información. Una red desarrollada en Francia con este fin es LORAWAN que tiene varias características que simplifican la conexión de nodos de Internet de las Cosas. Estas características son: consumo muy bajo de energía, comunicación punto a punto, un alto alcance, hasta quince kilómetros en áreas rurales, seguridad implementada en sus transmisiones y facilidad para conectar varios nodos a su red. El consorcio de LoRaWAN define un sistema que incluye servidores de red y de aplicaciones, gateways y los dispositivos que cuentan con un nodo IoT. Sus frecuencias de trabajo son 868 MHz en Europa, 433 Mhz en Así o bien 915 en el continente americano [4]

La red SigFox es un protocolo de red, también de bajo consumo de energía, una red de área amplia, cuya propietaria es el consorcio UnaBiz. Este consorcio tiene presencia en más de setenta países y proporciona servicios e integración de productos para redes de dispositivos IoT. Entre sus clientes se encuentran empresas de logística, empresas de control y administración de edificios, de control de suministros entre otros. Esta red trabaja en banda sin licencia disponible públicamente de 200 kHz, puede ser entre 868 y 869 Mhz o

bien en 902 a 920 Mhz, de acuerdo a la región del planeta en que se vaya a utilizar [5].

Existen otras redes inalámbricas que también pueden utilizarse en nodos IoT, como redes de telefonía celular, en especial 5G diseñada con tal fin, redes de Internet tipo WiFi, sin embargo, debe considerarse que cada tipo de red inalámbrica tiene sus propias características y en algunas situaciones cada red puede ser la más adecuada y en otras situaciones no serlo [6][7].

IoT. La tecnología de Internet de las Cosas se utiliza en muy diversas situaciones y lugares, es más notoria su utilidad en las ciudades, pero sus aplicaciones pueden verse en áreas rurales y en alta mar. Un nodo IoT está instalado en algún dispositivo o máquina del cual toma información por medio de sensores de temperatura, de aceleración o inclinación, también parámetros físicos de las personas, las señales de los sensores son recolectados por medio de una computadora y ésta puede realizar acciones de control sobre el dispositivo con lo que se logra independencia de operación local. La computadora, utilizando algún puerto de comunicaciones establece comunicación con servidor que realiza varias operaciones como registrar información del nodo IoT realizar análisis de datos, albergar sistemas web para con información de ecosistemas IoT e incluso enviar órdenes a los nodos por medio de la red inalámbrica y ejecutados por la computadora en el nodo. En la figura 2 se puede ver una placa Arduino con antena LoRa para IoT [8].



Figura 2 Placa Arduino con antena LoRa

Computadora de placa reducida: El desarrollo de la industria electrónica ha logrado reducir el tamaño de los circuitos de este tipo, una consecuencia es que el tamaño de las computadoras y microcontroladores se ha reducido y ha aumentado su capacidad de cómputo y de almacenamiento de

datos. También se ha logrado que estos equipos ya tengan integrados varios puertos que simplifican su interacción al exterior de la computadora. Tienen puertos seriales, puertos para tarjetas de red o incluso ya están integrados a la computadora. Esto le agrega a la computadora independencia de operación además del poder de procesamiento.

Hoy en día las computadoras de placa reducida más populares son Raspberry, Arduino y Orange Pi. Cada una tiene ventajas sobre la otra, pero cada nuevo modelo va integrando más capacidades lo que hace que los ingenieros de IoT cada día desarrollen sistemas más completos, más potentes y más eficientes. En el caso de la Raspberry el modelo más reciente es el Pi 5, esta versión tiene un procesador Arm que trabaja a 2.4 GHz, cuatro puertos paralelos, puerto WiFi 802.11ac, salida HDMI dual de 4Kp60 entre otras características, desde luego existen variantes de este modelo, ya sea en tamaño o en velocidades de procesamiento. Por otra parte, Arduino es la principal competencia de Raspberry, tiene varios modelos con la cualidad de atacar diferentes nichos de mercado, se tienen las familias Nano, MKR, la Classic y la Mega, cada una con sus nichos ya mencionados, otra ventaja es su amplia biblioteca de funciones que le ahorran muchas horas de programación a los desarrolladores de aplicaciones IoT y acelera el proceso de implementación, con este propósito también cuentan con IDE's, esto es medios ambientes de desarrollo integrados que facilitan el trabajo a los programadores haciendo que se obtengan sistemas completos en tiempos menores [9] [10].

Programación Web. La programación Web consiste en procesar información almacenada en un servidor y hacerla accesible vía Internet para usuarios tanto locales como ubicados en otros lugares. La programación Web está basada en la arquitectura cliente-servidor. Esta arquitectura está basada en diferentes computadoras de una red en la que una de ellas lanza una petición o requerimiento a otra computadora y ésta le responde de alguna forma, casi siempre compartiendo datos o información. La computadora que hace la petición se le conoce como cliente y la que responde como servidor. Para implementar este modelo se requiere una conexión entre computadoras, para este trabajo Internet, una computadora que funcione como servidor, un manejador de base de datos almacenado en el servidor y varios bloques de software que gestionen tanto las peticiones como las respuestas entre computadoras, también que administren las consultas a la base de datos y que administren la comunicación. A todo esto se le conoce como Programación Web o aplicaciones Web y es la base de como funciona Internet desde hace ya varios años [11].

Lenguajes de Internet. Para desarrollar una aplicación Web se utilizan varios lenguajes de programación enfocados en diferentes tareas en el modelo cliente-servidor, el primero de ellos es HTML (Hyper Text Markup Lenguaje) cuyo objetivo principal es definir una página web, esto es los

elementos que van a conformar la página, dicho de otra forma la arquitectura que tendrá la página, HTML por medio de marcas o indicaciones le indica a los navegadores de Internet como manejar la página web, desde el título de la misma, sus secciones y los enlaces a otras páginas entre otras marcas. Otro lenguaje es Hojas de Estilo en Cascada (Cascade Style Sheets) que permite manejar y controlar el aspecto de documentos electrónicos de una página web. El organismo World Wide Web Consortium ha dictado recomendaciones con respecto a este lenguaje, cada versión se reconoce por un número, la más usada por el momento es CSS3, pero las versiones seguirán cambiando a través del tiempo. La principal ventaja de usar CSS es que se hace una definición genérica de la presentación de la página web y ya no tiene que volverse a definir, por ejemplo, en HTML se puede especificar el tipo de letra, pero se tiene que hacer cada vez que se usa, pero con CSS basta una sola vez para que se respete tal definición. Otro ejemplo es si se define un área en la página web se pueden definir colores, bordes, tamaño. Si bien en un principio puede parecer engorroso el trabajo, una vez que se ha hecho ya no se tiene que rehacer continuamente. Javascript es un lenguaje de programación bastante sencillo enfocado en desarrollar páginas web dinámicas, una página desarrollada con HTML es bastante sencilla y estática, esto es, que la interacción con el usuario es limitada, en cambio una página web mejorada con Javascript le permite mejor interacción al usuario de la misma. El otro lenguaje utilizado en páginas web es PHP, al igual que Javascript utiliza scripts, pero del lado del servidor, operaciones como conexiones con bases de datos, enviar o recibir cookies o generar páginas con contenidos dinámicos son algunos ejemplos de tareas que se pueden implementar con PHP, el uso combinado de estos lenguajes es lo que facilita la Programación Web [12].

Bases de datos. La información que se almacena en los servidores se maneja con algún manejador de bases de datos que facilita la organización de la información, la búsqueda y ordenamiento de la misma. Los principales manejadores de bases de datos utilizados en la actualidad son MySQL, MariaDB o MongoDB. Existe debate entre las ventajas de usar MySQL y MongoDB, porque el primero exige una definición de tablas de información más rígida, en cambio MongoDB es más flexible para definir las relaciones entre sus tablas, desde luego ambos enfoques tienen ventajas y desventajas, si los datos son cambiantes, como en muchas situaciones del día de hoy, la flexibilidad que proporciona MongoDB es preferible, si en cambio se requiere un mayor control y la información es más estable en el tiempo MySQL es más adecuado. Cada dueño de información debe decidir cual es el criterio que más conviene a su información [13].

III. TRABAJOS RELACIONADOS

La ciudad de Chikuma, en Japón, es una ciudad de tamaño medio, cerca de 120 kilómetros cuadrados, 59, 380

habitantes en el año 2019. Ahí se instaló un sistema de monitoreo en el sistema de transporte de la ciudad. Se instalaron tres estaciones de monitoreo con comunicación LoRa en tres puntos de la ciudad, como si fueran faros de navegación y transmisores, de la misma tecnología, en los autobuses del servicio urbano. La ciudad cuenta con nueve rutas de transporte urbano de autobuses y de acuerdo a los experimentos que realizaron investigadores japoneses se tuvo éxito en el monitoreo de los autobuses en el 62.3% del monitoreo, esta información se almacenaba en tiempo real en una nube de datos. Los investigadores concluyeron que instalando mas puntos de monitoreo, los mencionados faros, sería posible monitorear en tiempo real a todo el sistema de autobuses de Chikuma [14].

Un proyecto similar se realizó en las Islas Filipinas en el año 2022, pero ellos monitorearon trenes del metro para brindar información a los usuarios del metro (Passenger Information Display System), este sistema despliega información de los trenes como velocidad, posición relativa en las estaciones, y el tiempo estimado de llegada a la estación. Para ello usaron sensores de efecto Hall para detectar el movimiento de los trenes, el sensor está controlado por una computadora Arduino modelo MKR1300, un GPS modelo NEO6M. EL sistema fue probado utilizando dos automóviles simulando al sistema de trenes [15].

También se han desarrollado trabajos que mezcla la información que se puede obtener de redes LoRa que monitorean el tráfico de una ciudad y la inteligencia artificial con técnicas de Aprendizaje profundo para predecir el grado de congestión vehicular en una ciudad, con esas predicciones se logra tener información para mejorar la movilidad urbana al evitar avenidas congestionadas y así reducir los retrasos, también se logra, de manera indirecta mejorar la seguridad en los desplazamientos vehiculares. Para lograrlo se utilizó el modelo LSTM (Long Short Term Memory) a bancos de datos disponibles en línea para entrenar y evaluar el modelo propuesto. El sistema desarrollado mostró buenos resultados con las bases de datos que se utilizaron, de acuerdo a sus resultados se puede predecir el comportamiento del tráfico en una ciudad [16].

La Ciudad de México, el Campus Zacatenco se encuentra en esta ciudad, ha hecho avances en el estándar GTFS estático, tiene disponibles en una base de datos la información más relevante de su sistema de transporte urbano, en el año 2022 inició un proyecto para desarrollar el GTFS dinámico, pero el tamaño de la ciudad y los recursos necesarios para desarrollarlo abarcando todas las unidades de transporte de la ciudad. Existe otra situación que complica el problema, varias rutas de transporte son concesionadas lo que obliga a establecer conversaciones con los dueños de las unidades para avanzar en la implementación del estándar porque los dueños de esas unidades deberían realizar una inversión extraordinaria

en sus unidades para montar el nodo IoT de monitoreo de sus unidades.

III. ANÁLISIS Y DESARROLLO

Para el desarrollo de este prototipo se siguió el siguiente proceso o metodología, primero se inició con el análisis del problema y su magnitud, a continuación, se revisaron las opciones de implementación y herramientas de ingeniería disponibles, posteriormente se procedió al diseño de la solución, el desarrollo de la solución se dividió en dos módulos, en uno de ellos se implementó el hardware del nodo IoT y la comunicación que éste tendría hacia su exterior, posteriormente se desarrolló el software para interactuar con los usuarios del sistema. La interacción tiene dos principales funciones, la de consulta de datos de rutas y la información del avance de las unidades para concluir se efectuaron las pruebas del prototipo, se realizaron en el Smart Campus Zacatenco, con un automóvil y una bicicleta por las rutas que siguen los transportes públicos en la Unidad.

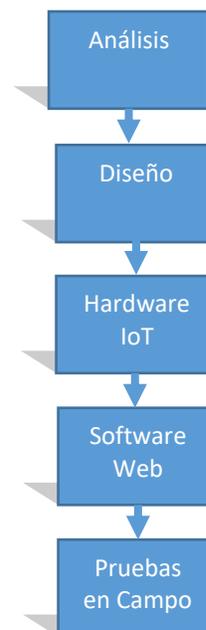


Figura 4 Metodología del prototipo

El Smart Campus Zacatenco del Instituto Politécnico Nacional se encuentra en la Ciudad de México, la Figura 3 muestra una toma aérea de parte de él. En este Campus, se encuentran seis Unidades Académicas que imparten educación a nivel licenciatura, maestría y doctorado, también se imparte educación continua de idiomas y de cursos de diversos tópicos para el público en general, dos Centros de Investigación, la administración central del Instituto Politécnico Nacional e instalaciones complementaria como campos deportivos, una unidad cultural y un planetario. En el campus confluyen alrededor de cuarenta mil estudiantes, cerca de tres mil empleados administrativos y una cantidad de visitantes diarios

que van a realizar entrega de bienes y mercancías, visitan el área cultural o bien acompañan a sus familiares, estudiantes o trabajadores por algún motivo, la cantidad de visitantes que nunca ha sido cuantificada adecuadamente, pero también utilizan el sistema de transporte. La Figura cinco muestra una imagen aérea del Campus Zacatenco y se puede observar una de las avenidas que la atraviesa.

El campus cuenta con tres avenidas internas y está rodeada por cinco grandes avenidas. Existen dos líneas de transporte que atraviesan el Campus, una línea de trolebuses y una de autobuses. La línea de trolebuses realiza un circuito, en las avenidas internas y externas, el circuito llega a una estación terminal del metro de la ciudad, la estación Politécnico de la línea amarilla y con una estación del sistema de metrobus de la Ciudad de México, la línea 6. La otra línea, la de autobuses parte de una estación del metro de la Ciudad de México, la estación la Raza que a su vez pertenece a rutas de metro, la ruta tres y la ruta cinco. Los autobuses también tienen una parada en la estación del metro Lindavista.

Las dos rutas de transporte público del Campus Zacatenco además cruzan o circulan por avenidas importantes de la Ciudad de México, la avenida Insurgentes, la avenida Montevideo, la avenida Politécnico, Eje Central y avenida Las Torres.

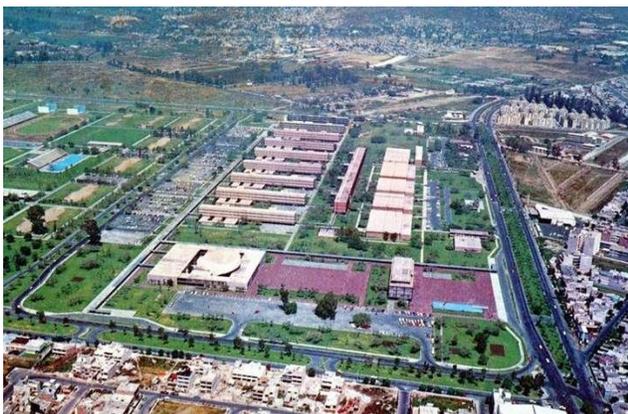


Figura 5 Imagen aérea del Campus Zacatenco

La ruta de autobuses tiene un circuito diferente, pero tiene la ventaja de cruzarse con estaciones de tres líneas del metro de la ciudad y también con tres estaciones de tres rutas de metrobus. Internamente, el Campus cuenta con once estaciones de abordaje, ambos sistemas comparten la mayoría de las estaciones. La frecuencia de servicio de las rutas es variable durante el día, los horarios de mayor demanda de servicio son por las mañanas, entre siete y ocho de la mañana, por las tardes entre una y cuatro de la tarde y el último pico es entre ocho y nueve de la noche. Para ambos sistemas no existe información sobre los sistemas de transporte, es en base a experiencia e información de boca en boca que se conocen detalles de ambos sistemas.

El sistema/diseño propuesto se enfoca en mostrar información en tiempo real del desplazamiento de las unidades de transporte público utilizando el estándar GTFS para que los usuarios del transporte público tengan información oportuna para que optimice sus desplazamientos y pueda hacer una planeación adecuada de sus desplazamientos.

Para desarrollar el sistema decidió implementar un nodo de Internet de las Cosas para ser instalado en las unidades de transporte colectivo, este nodo debe reportar la posición de la unidad en tiempo real a un servidor/Gateway de forma que ahí se pueda consultar esta información. Un servidor de datos y de aplicaciones que permita almacenar la información de las unidades y alojar una aplicación web para que los usuarios conozcan la información del transporte público, en la Figura 6 se puede ver una unidad de la ruta de trolebuses.



Figura 6 Unidad de trolebus

El nodo fue construido una placa Arduino MKR WAN 1300, la placa contó con un módulo CMCMWX1ZZABZ de Murata que permite la conectividad LoRa y un dispositivo de criptoautenticación ATECC508, también cuenta con una memoria Flas SPI de 2MB con lo que cuenta con suficiente memoria para las tareas que como nodo IoT.

Por otra parte, la placa MKR WAN 1300 tiene la capacidad de conectarse con la nube de Arduino The Things Network (TTN), esta nube tiene varias herramientas para construir ecosistemas para IoT, por ejemplo tiene un amplio conjunto de sensores de medio ambiente, humedad, presión temperatura y muchos más. También tiene contadores de personas y de proximidad, así como un conjunto de dispositivos para crear redes de nodos IoT, tal vez la herramienta más útil sea el seguimiento de los nodos IoT construidos con Arduino, para ello es necesario crear una cuenta en TTN relacionada con el dispositivo Arduino utilizado. Se deben utilizar las bibliotecas de software de Arduino y del MKRWAN para configurar las comunicaciones del nodo IoT con LoRa y LoRaWAN. La principal función del nodo IoT es la geolocalización, para ello se utiliza un módulo

GPS, el MKR GPS Shell, el módulo se conecta directamente a la placa Arduino, pero tiene un conector Eslov por lo que se puede conectar a otras placas que cuenten con un puerto de ese tipo.

En este prototipo la información más relevante es la información de las dos rutas de transporte público que atraviesan el Smart Campus Zacatenco, esta información son las paradas (stops.txt en el estándar GTFS) y las rutas, los datos de acuerdo al GTFS Schedule. Para el caso del GTFS Realtime se requiere la posición de los vehículos y el tiempo estimado de llegada, con esa información es más sencillo para el usuario planear sus desplazamientos, tanto dentro como fuera de la Unidad.

La información para los usuarios del transporte colectivo del Campus Zacatenco se consulta por medio de una página Web, la página tiene como objetivos ser lo más simple posible, no tener restricciones en su uso y desplegar la información arriba descrita.

IV. RESULTADOS

El prototipo ha sido probado por módulos, el nodo IoT en las rutas de transporte público y muestra la posición con buena resolución para las necesidades planteadas, se ha probado con automóviles y bicicletas, pero no ha sido posible montarlos en las unidades de transporte público debido a los problemas de ser una empresa de gobierno, pero que no depende del Instituto Politécnico Nacional. La comunicación con la plataforma TTN ha sido eficiente. También se ha implementado comunicación punto a punto con antenas LoRa y un servidor. La página Web ha sido desarrollada y se está en la etapa de ser probada por usuarios del transporte público vía Internet para que ellos proporcionen retroalimentación respecto a su facilidad de uso.

IV. CONCLUSIONES

El diseño del prototipo está completo y se han probado varios módulos por separado, todos ellos adecuadamente, es recomendable hacer las gestiones necesarias ante autoridades gubernamentales para probarlo en las unidades del transporte público, esta última situación es complicada por los costos de montar los nodos IoT en las unidades de transporte y por la red LoRa que se debe utilizar y cuya operación requiere de personal que la mantenga operando de forma continua.

AGRADECIMIENTOS

Los autores expresan en esta sección su agradecimiento a los Sistemas de Becas por Exclusividad de la Comisión Organizadora de Fomento a las Actividades Académicas, al Sistema de Estímulo al Desempeño Docente de la Secretaría

Académica y al Sistema de Estímulos a la Investigación de la Secretaría de Investigación y Postgrado, ambos Sistemas soportados por el Instituto Politécnico Nacional de México. También a la Secretaría de Investigación y Postgrado porque este trabajo se deriva de las actividades del proyecto de investigación “Micro movilidad urbana en el Smart Campus Zacatenco”, clave 20240757.

REFERENCIAS

- [1] StreetsblogSF, “How Google and Portland’s TriMet Set the Standard for Open Transit Data” <https://sf.streetsblog.org/2010/01/05/how-google-and-portlands-trimet-set-the-standard-for-open-transit-data>, consultado 28/04/2024
- [2] GTFS, “GTFS Schedule”, <https://gtfs.org/es/schedule/reference/>, consultado 04/04/2024.
- [3] GTFS, “GTFS Realtime”, <https://gtfs.org/es/realtime/feed-entities/>, consultado 04/04/2024.
- [4] LoRa Alliance, “What is LoRaWAN®?”, <https://loro-alliance.org/about-lorawan/>. Consultado 04/04/2024.
- [5] UnaBiz, “What is SigFox”, <https://www.sigfox.com/what-is-sigfox/>. Consultado 04/04/2024.
- [6] Instituto Federal de Telecomunicaciones, “¿Qué es 5G?”, https://sensor5g.ift.org.mx/que_es_5g. Consultado 04/04/2024.
- [7] Unión Internacional de Telecomunicaciones, “WiFi 6 y sus dispositivos”. https://www.itu.int/en/ITU-R/seminars/rrs/2021-Americas/Forum/Session%206_Spectrum%20Licensing%20models/ACCES%20PARTNERSHIP_WiFi6%20%26%20dispositivos.pdf Consultado 04/04/2024.
- [8] A. Rayes, S. Salam, “Internet of Things From Hype to Reality”, Second Edition, 2019, Springer Natures, Switzerland, pp 2-22.
- [9] S. Monk, “Programming Arduino Getting Started with Sketches”, 2nd Edition, Union Kingdom, 2016.
- [10] S. Monk, “Programming the Raspberry Pi: Getting Started with Python”, 1st Edition, Union Kingdom, 2013.
- [11] C. Parraga, M. J. Boné Andrade, A. P. Mora Olivero, “Programación Web del Frontend al Backend”, Editorial AEA 1era Edición, Santo Domingo, Ecuador, 2024.
- [12] C. R. Jaimez González, “Programación del Web Dinámico”, Editorial UAM Cuajimalpa, Ciudad de México, México, 2015.
- [13] P. Lake, P. Crowther, “Concise Guide to Databases”, Springer – Verlag, London, England, 2013.
- [14] K. T. Murata et al., “LoRa Communication Maps of Medium-Sized Rural City in Japan via Community Bus Services,” 2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCCom/CyberSciTech), Fukuoka, Japan, 2019, pp. 1054-1059, doi: 10.1109/DASC/PiCom/CBDCCom/CyberSciTech.2019.00189
- [15] S. A. Pasia et al., “MRT Trains Mobility Tracking Using LoRa Network,” 2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Boracay Island, Philippines, 2022, pp. 1-6, doi: 10.1109/HNICEM57413.2022.10109545.
- [16] H. Benadji, L. Zitoun and V. Vèque, “Predictive Modeling of Loss Ratio for Congestion Control in IoT Networks Using Deep Learning,” GLOBECOM 2023 - 2023 IEEE Global Communications Conference, Kuala Lumpur, Malaysia, 2023, pp. 2814-2819, doi: 10.1109/GLOBECOM54140.2023.10437769.