

DEFINICIÓN DEL CASO DE ESTUDIO

CONTEXTO

En la actualidad, las estaciones del sistema urbano de transporte público de la ciudad de Bogotá (SITP), solo constan de un panel metálico que indica las rutas que tienen como paradero dicha estación, y no brindan más información de primera a mano. Esto trae consigo problemas en la experiencia del usuario, debido a que no conoce los tiempos de espera del bus, retrasos de este, o las horas estimadas de llegada.

Es cierto que actualmente se ofrece el servicio de la aplicación de *TransmiApp* que permite conocer los tiempos de llegada de cada articulado a la estación con ciertos márgenes de error. No obstante, es importante destacar que para que el usuario de uso de esta requiere de su instalación en sus dispositivos móviles y contar con conexión internet, ya que su funcionamiento se da por medio del sistema de posicionamiento global GPS. Mediante el cual los buses comunican su ubicación a una base central, y esta información se distribuye en la aplicación.

Con esto en mente, la aplicación deja de un lado una gran parte de la población que usa el sistema (SITP), como personas sin dispositivos móviles, o personas sin disponibilidad de servicio de internet/datos en sus dispositivos móviles.

Así, nuestra propuesta consiste en implementar una red de comunicación para establecer una conexión eficiente entre los paraderos y los buses, dando lugar a un sistema de transporte público inteligente. De tal forma que los usuarios puedan obtener de manera precisa los tiempos estimados de llegada de cada ruta que tienen parada en una estación en particular. Esta propuesta busca transformar el sistema en una experiencia orientada al usuario, por medio de una arquitectura orientada hacia el internet de las cosas, mejorando la eficiencia y la conveniencia mediante la optimización de la gestión de datos y la disponibilidad de los datos en tiempo real, en cuanto a los tiempos de espera de las rutas.

PROBLEMA A ABORDAR

El problema consta en crear un sistema de comunicación entre paraderos y buses, para informar en tiempo real el tiempo estimado de llegada de cada bus a una estación a los usuarios, de los respectivos articulados que tienen como parada cada estación. Para ello primero es esencial conocer el sistema de funcionamiento del SITP.

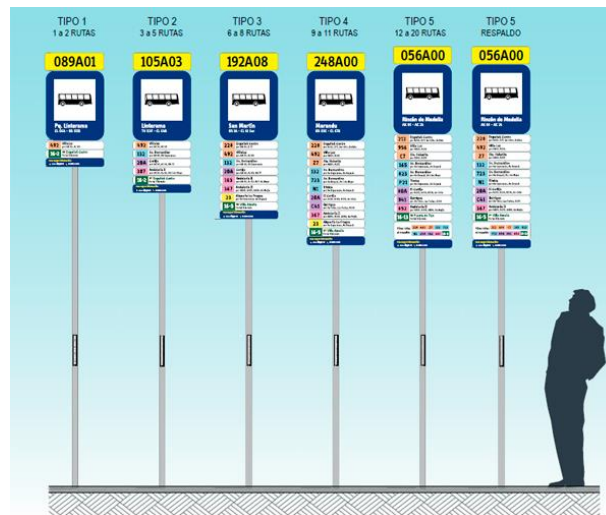


Figura 1. Tipos paraderos del SITP

En la *Figura 1*, se muestra una ilustración de los tipos de paraderos del SITP, los cuales representan la cantidad de rutas que tienen como parada dicho paradero, variando estos de 1 a 20 rutas, la sección amarilla denominada *cenefa* es el identificador único del paradero, la sección azul es la señalización y el nombre del paradero, y la última sección son las rutas que paran en la estación. Asimismo, cada ruta tiene un identificador único, y un “rutero” que describe las vías que toma la ruta y su destino. En la figura 2, se muestra la descripción del rutero el identificador es el que se ve en la sección azul 193B, en este caso, luego se ve el nombre del lugar al que se dirige el articulado, y finalmente los corredores por los que pasa el mismo, por otra parte, en la figura 3, se muestra el cambio del rutero al cambiar de dirección, se mantiene el identificador único, el cambio más importante es el nombre del lugar a donde se dirige el articulado.

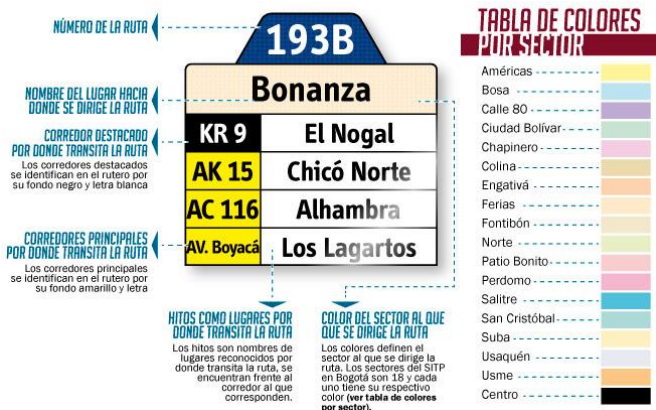
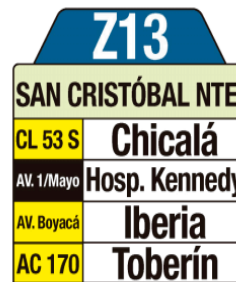


Figura 2. Descripción rutero SITP.

Cuando el bus va hacia San Cristóbal Norte:



Cuando el bus va hacia Metrovivienda:

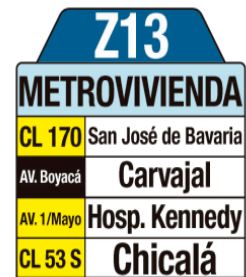


Figura 3. Dirección rutero SITP

Por otra parte, en la *RESOLUCIÓN No. 264 de 2016* se expone que la distancia promedio entre paraderos/estaciones siendo esta entre 300 y 400 metros. Así, tenemos definidas las principales variables para la construcción del sistema de comunicación, siendo estas los identificadores únicos de paradero y bus (ruta), y la distancia promedio entre estaciones.

Por lo que, el problema a abordar consiste en crear un sistema de comunicaciones entre paraderos y buses, para dar de primera mano el tiempo estimado de llegada de los articulados a los paraderos, a los usuarios del sistema

SITP, y así mejorar la experiencia del usuario. Con el fin, de generar un sistema inteligente de transporte, capaz de brindar información en tiempo real a los usuarios y con ello mejorar la experiencia del usuario en el uso del mismo.

Finalmente, cabe aclarar que este sistema por el momento se propone para las estaciones que lucen como la figura 4, debido a que su estructura permite la implementación de distintos sistemas que se presentarán en las siguientes secciones, para las estaciones que lucen como la figura 5, se dificulta dicha implementación, puesto que solo se tiene el poste que indica las rutas que paran en la estación, y así no se tiene espacio para la inclusión de nuevos elementos, por lo cual no se consideran en esta propuesta en su totalidad, trabajarán como nodos intermedios de comunicación para conocer la posición de los buses y así transmitir información hacia el resto de estaciones, para el despliegue de información en las mismas.



Figura 2 . Paradero apto para la implementación del sistema de comunicaciones



Figura 3 . Paradero no apto para la implementación del sistema de comunicaciones

JUSTIFICACIÓN

El sistema que proponemos incluye una pantalla para visualizar los tiempos de espera de los buses y una fuente de alimentación independiente para cada pantalla, como se ilustra en la figura 6. Al llegar un bus, se actualizará el estado de la ruta y se transmitirá esta actualización a las estaciones y/o a los nodos intermedios, para reflejar con precisión los tiempos estimados de llegada en cada estación.

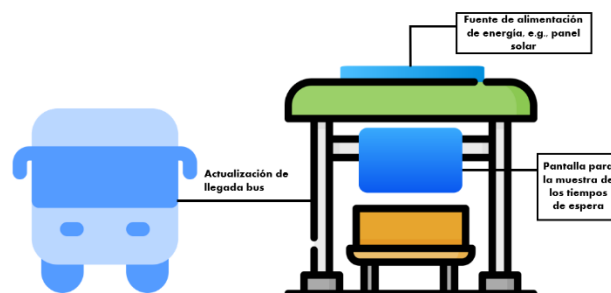


Figura 4. Diagrama base

Este sistema será de gran utilidad ya que permite a los usuarios del SITP conocer el tiempo de llegada de los buses a su paradero y facilita el seguimiento del tiempo que tarda un bus en recorrer el trayecto entre paradas consecutivas. Esto posibilita realizar estimaciones de tiempo de viaje más exactas.

A diferencia de la aplicación *TransmiApp*, nuestra propuesta emplea una red del Internet de las Cosas (IoT) que prescinde del uso del GPS, eliminando la necesidad de conexión a Internet en los paraderos. Sugerimos un sistema de comunicación entre paraderos y entre estos y los buses, permitiendo que, mediante retransmisiones, cada paradero acumule datos actualizados sobre la ubicación y el tiempo de llegada de los buses.

Este enfoque integral no solo mejora la distribución de información en tiempo real en las estaciones, sino que también enriquece la recolección de datos para el análisis de rutas. Proporciona un servicio equitativo al considerar a todos los usuarios del SITP, mejorando la accesibilidad de la información y la planificación de viajes para aquellos sin acceso a dispositivos móviles con Internet, así como perfeccionando la captura y estimación de datos para usuarios de aplicaciones como *TransmiApp*.

En resumen, implementa la integración de un sistema inteligente al SITP, para mejorar la experiencia de usuario al estimar en tiempo real los tiempos de espera de cada una de las rutas en sus respectivos paraderos, y así informando al usuario de dicha información teniéndola de primera mano. Para agregar, este sistema también permitirá identificar deficiencias en cuanto a las rutas, al calcular el tiempo de espera de las mismas, y así se obtendrán deficiencias del sistema de transporte público de manera rápida y eficaz, para con ello realizar análisis y soluciones a las mismas.

LEVANTAMIENTO DE REQUERIMIENTOS

En esta fase inicial del desarrollo del proyecto, es imprescindible realizar un correcto levantamiento de requerimientos. Este paso esencial definirá las bases para construir la infraestructura del sistema IoT, estableciendo claramente los objetivos, funcionalidades y criterios de calidad necesarios para el éxito del proyecto.

Durante esta fase, será central en comprender en profundidad las necesidades y expectativas de todos los actores implicados, que van desde los usuarios finales del SITP hasta los operadores del sistema y las autoridades gubernamentales. Nuestro objetivo es capturar con precisión y exhaustividad los requisitos funcionales y de calidad que orientarán el diseño e implementación del sistema.

Contamos con la colaboración de estudiantes y usuarios habituales del SITP para entender desde su perspectiva las funcionalidades esenciales para el proyecto. Además, se han identificado requisitos de calidad mediante comparaciones con otros sistemas IoT existentes.

IDENTIFICACIÓN DE LOS INTERESADOS DEL SISTEMA

El proyecto IoT tiene una variedad de stakeholders dada su amplitud. A continuación, se enumeran los interesados y cómo se beneficiarían con el sistema:

1. **Usuarios del SITP:** Los usuarios del Sistema Integrado de Transporte Público (SITP) se beneficiarían al recibir información en tiempo real sobre los tiempos de llegada de los autobuses en los paraderos. Esto

les permitiría planificar sus viajes de manera más eficiente, reducir el tiempo de espera en las paradas y mejorar la experiencia general de viaje.

2. **Alcaldía de Bogotá y el Ministerio de Transporte:** La Alcaldía de Bogotá y el Ministerio de Transporte podrían beneficiarse al mejorar la calidad del transporte público en la ciudad. El sistema proporcionaría datos valiosos sobre la operación de autobuses, la demanda de transporte en diferentes rutas y las condiciones del tráfico. Esta información puede ayudar a tomar decisiones informadas para optimizar la infraestructura de transporte y mejorar la movilidad en la ciudad. Además, al tener un sistema organizado, implicaría mayor ganancia por cada ruta de SITP que opere.
1. **Desarrolladores de aplicaciones móviles:** Los desarrolladores de aplicaciones móviles podrían utilizar los datos proporcionados por el sistema para crear aplicaciones que ofrezcan a los usuarios funciones adicionales, como notificaciones en tiempo real, sugerencias de rutas y la posibilidad de recargar tarjetas de transporte desde sus dispositivos móviles. Esto aumentaría la utilidad de las aplicaciones y podría generar oportunidades de negocio, al habilitar una API con la información relevante para los desarrolladores.
2. **Investigadores y planificadores urbanos:** Los investigadores y planificadores urbanos pueden utilizar los datos recopilados por el sistema para analizar patrones de movilidad, evaluar la eficiencia del transporte público y diseñar estrategias para mejorar la planificación urbana. Esto podría conducir a un mejor diseño de rutas y del número de buses que se envían, así como otras medidas para mejorar el transporte público en Bogotá.

ID	Nombre del interesado	Rol u Organización	Intereses u Expectativas	Nivel de poder	Nivel de interés	Implicación en el proyecto	Impacto en el proyecto
1	Usuarios del SITP	Pasajeros	Mejorar la experiencia de viaje, reducir tiempos de espera.	Bajo	Alto	Principales beneficiarios	Alto
2	Alcaldía de Bogotá	Gobierno Local, duseño del SITP	Mejorar la movilidad en la ciudad, optimizar la infraestructura de transporte.	Muy Alto	Alto	Influyente en la toma de decisiones	Alto
3	Desarrolladores de Aplicaciones	Empresas de Desarrollo o sesarrolladores independientes	Crear aplicaciones con datos en tiempo real, oportunidades de negocio.	Medio	Medio	Integración de datos	Medio
4	Investigadores y Planificadores	Instituciones de Investigación	Analizar patrones de movilidad, mejorar la planificación urbana.	Medio	Alto	Datos para análisis y planificación	Alto

ESPECIFICACIÓN Y PRIORIZACIÓN DE LOS REQUISITOS FUNCIONALES

Para el proyecto IoT planteado, existen requisitos funcionales a nivel de Hardware y a nivel de Software; son los siguientes:

1. Sistemas de los buses:

- Los buses deben tener un sensor de proximidad integrado dentro de ellos. Este sensor trabajara con el protocolo Thread para poder comunicarse con las estaciones que estén en su rango y su ruta.
- Los buses deben tener un módulo de comunicación. Como se mencionó anteriormente estos módulos trabajan con Thread y permiten la transmisión de datos sobre el estado del autobús a la infraestructura central de las estaciones.

2. Sistemas de las estaciones intermedias:

- Cada estación contará con un sensor de proximidad. Este sensor igualmente trabaja con el protocolo Thread para poder comunicarse con los buses que pasen dentro de su rango y con las demás estaciones intermedias al igual que la estación core que recolecta la información.
- Cada estación tendrá un sistema de procesamiento. Este sistema debe ser capaz de identificar el bus que pase, y retransmitir la información a las demás estaciones (ya sean intermedias o core). Igualmente, debe almacenar momentáneamente la información en dado caso de pérdida de conexión para hacer la recuperación.
- Cada estación contará con un panel solar. Esta es la fuente principal de alimentación para los sistemas en las estaciones, hay que tener en cuenta el uso energético por parte del sistema para utilizar un panel solar de acuerdo con los requerimientos.
- Para todo el sistema se hará la implementación del protocolo Thread. Como se mencionó previamente, se utilizará este protocolo para las comunicaciones entre las estaciones y los buses. Se debe asegurar que la implementación del protocolo sea coherente y cumpla con los estándares para facilitar la comunicación eficiente.
- Cada estación debe estar en la capacidad de identificarse a si misma con un ID e identificar a las estaciones cercanas correspondientes según una tabla de enrutamiento.
- Cada estación debe ser capaz de poder enrutar y descubrir las demás estaciones que la rodean siempre que están sean de su misma ruta.
- Debe existir un código que permita a la estación no solo recolectar la información sino agruparla y retransmitirla a las demás estaciones correspondientes.

3. Sistemas de las estaciones core:

- Adicionalmente a lo que ya tienen las estaciones intermedias, las estaciones core deberán tener una pantalla LCD. Esta es la forma de interactuar con el usuario, a través de la pantalla se van a mostrar los tiempos estimados para cada una de las rutas correspondientes a esa estación.
- De igual forma, las estaciones core contarán con procesos de identificación, enrutamiento y procesamiento, como lo tienen las estaciones intermedias.
- Cada estación debe ser capaz de implementar un algoritmo de predicción de tiempos de espera. Se debe desarrollar un algoritmo predictivo que utilice datos de los sensores y autobuses para estimar los tiempos de espera de manera precisa. El software debe ser capaz de procesar la información y actualizar las predicciones en tiempo real.
- Cada estación debe mostrar una interfaz de usuario en las pantallas. Es necesario desarrollar una interfaz de usuario clara y comprensible en los dispositivos de pantalla de las estaciones. La interfaz debe mostrar de manera efectiva los tiempos de espera y cualquier otra información relevante.

4. El sistema en general debe:

- Proveer un API con la información actualizada de la ubicación de los buses y el tiempo de cada paradero.
- Proveer una base de datos con la información de los buses y paraderos.
- Proveer una interfaz donde se puedan realizar análisis de los datos almacenados.

- Detectar e informar sobre fallos en paraderos o buses, debido a la falta de recepción.

A continuación, se explicará con más detalle el proceso para predecir los tiempos de llegada de los autobuses y presentar la información en las estaciones core:

1. Detección de autobuses en las estaciones intermedias:

Las estaciones intermedias, que actúan como nodos de detección, utilizan sensores de proximidad para reconocer la llegada y salida de autobuses. Cuando un autobús pasa por una estación intermedia, esta registra la información relevante, el identificador del bus, su ruta, y la velocidad con la que iba, y transmite estos datos al sistema central a través del protocolo Thread. Esta información se envía a la estación core designada como estación central receptora dentro de la ruta correspondiente. Cabe aclarar que dentro de una misma ruta pueden haber varias estaciones core, y los datos siempre serán enviados hacia el sentido en el que va el bus, pues no tendría sentido avisar a una estación anterior.

2. Procesamiento de datos en la estación core receptora:

La estación core receptora, que actúa como nodo central para recibir y procesar datos, recibe la información enviada por las estaciones intermedias (un JSON). Un software centralizado en esta estación procesa los datos brutos, identifica los autobuses que vienen en camino y actualiza la base de datos con las rutas de los buses que vienen y sus tiempos.

3. Algoritmo Predictivo en la Estación Core:

En la estación core, se implementa un algoritmo predictivo que utiliza los datos procesados del paso anterior para calcular un tiempo aproximado utilizando la velocidad y una distancia base entre estaciones. De esta forma sabiendo que un bus paso por en frente de una estación con cierta velocidad se puede estimar el tiempo de llegada a la siguiente estación. Este mismo procedimiento se puede efectuar independientemente de la cantidad de estaciones intermedias previas a la estación core, pues los paquetes enviados contienen la cantidad de saltos hechos. Es decir, si un bus viene cuatro estaciones atrás, el paquete informa la ruta del bus, la velocidad y que va cuatro estaciones atrás, entonces el algoritmo es capaz de hacer los cálculos aproximados para cada tramo entre estaciones y obtener un resultado.

4. Actualización de Pantallas en Estaciones Core:

Con base en las predicciones generadas por el algoritmo, la estación core actualiza la información en la pantalla disponible en esa estación. Esta actualización se realiza en tiempo real, mostrando a los usuarios los tiempos estimados de llegada de los autobuses próximos. El software en la estación core también envía esta información a otras estaciones core conectadas en la red para mantener la coherencia en la presentación de datos.

ESPECIFICACIÓN Y PRIORIZACIÓN DE LOS REQUISITOS DE CALIDAD

El sistema IoT propuesto presenta requisitos de calidad en base a diferentes atributos de calidad:

- Precisión en la Predicción de Tiempos de Espera:

- Garantizar que la estimación de los tiempos de llegada de los autobuses sea altamente precisa es fundamental para la utilidad y confiabilidad del sistema. La precisión en la predicción es crucial para que los usuarios puedan planificar sus viajes de manera efectiva.
- Prioridad alta. La exactitud en la predicción de tiempos de espera es esencial para cumplir con las expectativas de los usuarios y mejorar la eficiencia del sistema de transporte público.

- Disponibilidad del Sistema:
 - Asegurar que el sistema esté disponible y funcional en todo momento es esencial para mantener la confiabilidad del servicio. La información precisa de tiempos de espera debe estar accesible para los usuarios en cualquier momento que la necesiten.
 - De igual forma en cuanto a la disponibilidad, se debe tener en cuenta los factores climáticos que pueden afectar al sistema. De esta forma debe existir una clase de protección contra el medio ambiente para que independientemente de las condiciones climáticas el usuario tenga acceso al dispositivo.
 - Prioridad alta. La disponibilidad constante del sistema es crítica para la satisfacción del usuario y la eficiencia operativa del transporte público. Es crucial que cuando un usuario llegue a una estación, esta esté funcionando adecuadamente y pueda indicarle al usuario los tiempos en los que llegará su bus.
- Desempeño:
 - La latencia máxima permitida debe estar definida claramente para garantizar que los usuarios obtengan información actualizada en tiempo real muy cercano a tiempo real. Por ejemplo, la latencia podría estar en el rango de máximo segundos para proporcionar datos precisos.
 - El sistema debe ser capaz de manejar múltiples solicitudes de lectura y transmisión simultáneamente, asegurando que la información se actualice en tiempo real para todos los paraderos y autobuses.
 - Prioridad media. Puesto que, si bien es necesario que los datos estén cercanos a tiempo real, al usuario no le afecta si el sistema está demorado por un par de segundos, siempre y cuando haga una aproximación considerablemente cercana en minutos al tiempo de llegada de su bus, estará satisfecho.
- Seguridad de Datos:
 - Implementar medidas robustas de seguridad para proteger la integridad y confidencialidad de los datos del sistema. Debe haber un control estricto de acceso para garantizar que solo las partes autorizadas puedan acceder al sistema y realizar cambios en la información.
 - Prioridad baja. La seguridad de los datos no es tan relevante en el contexto planteado, ya que la información transmitida no es sensible, sin embargo, hay que asegurar que solo las partes autorizadas puedan enviar y recibir la información.
- Escalabilidad:
 - Diseñar el sistema para manejar un aumento en el número de sensores y usuarios sin pérdida significativa de rendimiento. Esto asegurará que el sistema pueda crecer y adaptarse a las necesidades cambiantes del transporte público. Es decir, si se desean agregar nuevas rutas, nuevos buses o paraderos, que esto no implique un costo adicional al sistema ni un cambio en la arquitectura. Para ello más adelante especificamos la utilización de protocolos, como MQTT, que nos permiten lograr este objetivo.
 - Prioridad media. A medida que el sistema se expande, la escalabilidad se vuelve más crucial, pero se puede abordar de manera progresiva a medida que crece la demanda. De igual forma, desde un inicio se tiene un conocimiento casi completo de todos los paraderos y rutas, y la agregación de nuevos elementos es poco frecuente en un ambiente real.

DISEÑO DEL SISTEMA A NIVEL DE HARDWARE DE SENSADO, ACTUACIÓN Y GATEWAY IOT

DETERMINAR Y CARACTERIZAR LAS VARIABLES DE MONITOREO

Para el proyecto propuesto, es importante tener en cuenta la definición de las variables de monitoreo y su caracterización. Estas variables son esenciales para la recopilación de datos precisos, lo que a su vez es crucial para la mejora continua del sistema de transporte y la experiencia del usuario. La correcta identificación y uso de estas variables permitirán optimizar los tiempos de viaje, aumentar la eficiencia de la flota y proporcionar información actualizada y confiable a los pasajeros.

Las siguientes son las variables fundamentales de monitoreo y su justificación en el contexto del proyecto:

1. Velocidad del Bus:

- La medición de la velocidad es crucial para calcular los tiempos de desplazamiento de un bus entre paradas.
- Ayuda a predecir los tiempos de llegada, lo que permite ajustes en tiempo real en la programación y mejora la fiabilidad del servicio.
- Proporciona datos para análisis de flujo de tráfico y planificación de rutas, identificando posibles retrasos debido a congestiones o desvíos.

2. Tiempo de Aproximación a la Estación:

- Importante para que las estaciones y los sistemas de monitoreo registren cuando se acerca un bus.
- Facilita la creación de un sistema de priorización en tiempo real para la secuenciación de los buses en la ruta.

3. Número de la Ruta:

- Es fundamental para que las estaciones identifiquen qué rutas están siendo servidas.
- Permite tomar decisiones informadas sobre la retransmisión de la información o cálculos relacionados con la logística y la asignación de recursos.
- Cada una de estas variables contribuye a un sistema integrado de monitoreo que, al ser analizado y gestionado correctamente, puede conducir a una prestación de servicios más eficiente y a una mejor experiencia para el usuario del transporte público.

4. Dirección de la Ruta (sentido de ida y vuelta):

- Los buses de SITP funcionan en rutas de ida y vuelta; tras completar un trayecto, continúan en dirección opuesta.
- Monitorear el sentido de la ruta es vital, ya que algunas rutas mantienen su nombre, pero cambian la dirección del servicio que ofrecen.

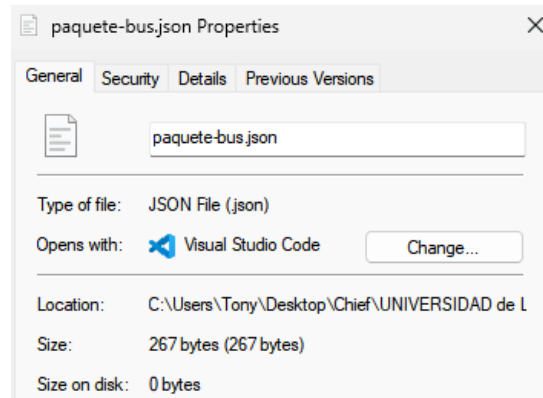
Teniendo en cuenta estas variables, se construyó el siguiente paquete de datos en formato JSON, que sería el paquete que se transmitiría desde los buses hacia las estaciones de bus:

```
{
  "type": "info",
  "bus": {
    "id": "58361192-0081-4aff-9f32-15a265f613e7",
    "velocity": 35.2
  },
  "route": {
    "id": "9adfd734-0087-4fcb-9f39-e0e3420b36e9",
    "name": "97",
    "direction": 1
  },
  "timestamp": "2023-11-10T12:15:21.000Z"
}
```



paquete-bus.json

Nótese que, en este paquete de datos, se contiene toda la información relevante y todas las variables mencionadas. Adicionalmente, se contiene un “id” tanto para cada uno de los buses y para cada una de las rutas. Al tratarse de un paquete tan condensado, el peso de este en bytes corresponde exactamente al número de bytes que contiene. Para el caso mostrado anteriormente, es:



Se asumirá que el peso promedio del paquete es de 300 bytes. Además, es importante considerar también el tamaño del encabezado del protocolo que se seleccione para obtener el tamaño total del paquete.

REALIZAR UN DIAGRAMA DE BLOQUES DEL FUNCIONAMIENTO PROPUESTO PARA LA CAPA DE DISPOSITIVOS DEL SISTEMA IOT

En la figura 1, se presenta el funcionamiento propuesto para el dispositivo IoT, por medio de un diagrama de bloques, en donde se considera cada tipo de estación, las denominadas nodo/estación intermedia, serán usadas solo para la transmisión de información hacia las estaciones que tienen los atributos para la implementación completa. En dichas estaciones se encuentra la fuente de energía, panel solar, para la alimentación de la pantalla LCD en cada estación, esta muestra el tiempo de espera de cada ruta.

Por otra parte, la identificación del bus en cada estación se realizará por medio de *Thread* en donde se almacenará, el código de la ruta, y su destino, e igualmente la velocidad y dirección de la misma. Finalmente, se tiene un *nodo central* el cual debe estar conectado a una red confiable y con ello conectarse a una base de datos con el fin de guardar información para propósitos de analítica de datos.

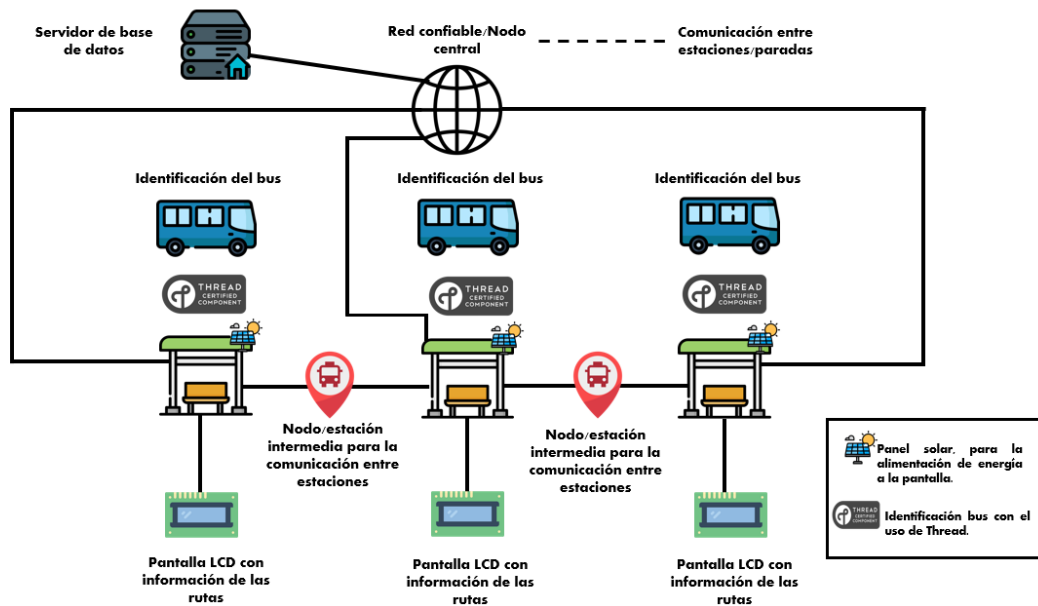


Figura 5. Diagrama de Bloques estructura IoT para el monitoreo de estaciones de bus y buses

SELECCIONAR LOS DISPOSITIVOS DE A UTILIZAR

SENSADO

Para el sensado de las variables del sistema, únicamente se requiere de un sensor que permita calcular la velocidad lineal del bus en un momento específico. Para esto se propone el uso de un sensor de velocidad del vehículo (VSS), porque este componente es robusto, confiable y ya está integrado en la mayoría de los vehículos modernos como parte del sistema de gestión del motor y control de tracción. Este sensor funciona mediante la detección de cambios en el movimiento de una pieza interna del vehículo, típicamente la transmisión o el eje de salida, para producir una señal eléctrica que varía con la velocidad de rotación. La señal generada es proporcional a la velocidad del vehículo y puede ser fácilmente convertida en velocidad lineal utilizando el radio de la rueda y datos de la transmisión como referencias. Esto permite un monitoreo continuo y en tiempo real de la velocidad del autobús, lo cual es fundamental para calcular el tiempo estimado de llegada (ETA) a las siguientes paradas y actualizar la información de espera para los usuarios en las estaciones del SITP.

Para especificar aún más las características del sensor VSS para su implementación en los autobuses del SITP, se podrían definir los valores de la siguiente manera:

- **Tipo de Sensor:**
Hall Effect: debido a su capacidad de proporcionar lecturas precisas y su resistencia a la interferencia en ambientes con mucho ruido eléctrico.
- **Rango de Medición:**
0 a 120 km/h: para cubrir desde paradas completas hasta la velocidad máxima esperada en un entorno urbano.
- **Resolución:**
1 pulso por milímetro de movimiento del vehículo: gran precisión para la conversión a velocidad lineal.

- **Salida de Señal:**
Señal Digital: con un nivel de señal de salida de 0-5V o 0-12V compatible con la mayoría de las ECU de vehículos.
- **Confiabilidad:**
Rango de temperatura operativa de -40°C a +125°C: adecuado para enfrentar las variadas condiciones meteorológicas.
Resistente a vibraciones: con especificaciones que cumplan o excedan la norma ISO16750-3 para la vibración del vehículo.
- **Alimentación:**
5V DC: estándar de alimentación para sensores de efecto Hall en vehículos.
- **Interfaz de Sistema:**
CAN Bus o LIN Bus: para comunicaciones en vehículos, permitiendo integración directa con la telemática del bus y paneles de visualización.
- **Frecuencia de Actualización:**
10 Hz o mayor: suficiente para actualizar la posición y velocidad del vehículo cada 0.1 segundos, lo cual es adecuado para un sistema de seguimiento en tiempo real.
- **Durabilidad y Mantenimiento:**
Vida útil: esperada de más de 10 años o 1 millón de kilómetros de recorrido, para minimizar el mantenimiento y los reemplazos.

Estos valores garantizarán que el sistema sea capaz de proporcionar datos fiables y actualizados con la frecuencia necesaria para la efectiva operación del sistema de transporte y la información proporcionada a los pasajeros. Es importante aclarar que para cada vehículo sobre el cual se instale este sensor, se debe caracterizar la función de velocidad lineal con respecto a los pulsos por minuto (RPM) que se generen en el sensor, para poder obtener correctamente la velocidad lineal y obtener un valor preciso, y así obtener una estimación precisa del tiempo que tardará el bus.

ACTUACIÓN

Como dispositivos de actuación se considera la Pantalla LCD. Para comunicar de manera efectiva los tiempos restantes estimados de llegada de los buses, se implementarán pantallas LCD que se montarán en las estaciones del SITP utilizando el espacio proporcionado por las placas de metal existentes. Para maximizar la visibilidad y el uso del espacio, la pantalla se ubicará al lado derecho del panel de la estación, de forma que armonice con la estética y estructura actual, similar a la disposición en las estaciones de Transmilenio.

El tamaño de la pantalla LCD dependerá del volumen de información a mostrar y del flujo de rutas de cada paradero. Por ejemplo, para paraderos con una afluencia alta y una variedad más grande de rutas, se necesitará una pantalla más grande, posiblemente extendiéndose a modelos que puedan mostrar varias rutas simultáneamente en una interfaz de desplazamiento o de varias páginas. Un tamaño inicial sugerido podría ser de 24 cm x 8 cm para estaciones con tráfico moderado, pero en paraderos con más de 8 rutas, se recomendaría una pantalla de mayor tamaño, que podría ser de 48 cm x 8 cm, que permite mostrar más rutas y detalles al mismo tiempo. La pantalla debería tener la capacidad de ciclar a través de las rutas en intervalos definidos, asegurando que toda la información relevante sea presentada en un loop continuo.

La selección de pantallas LCD debe considerar la legibilidad en condiciones de luz solar directa, lo que implica pantallas con alto brillo y contraste. Además, se deben usar pantallas LED-backlit para optimizar el consumo de energía. La resolución de las pantallas deberá ser suficiente para mostrar textos y números claramente legibles a la distancia promedio a la que los usuarios esperan en el paradero.

Para la selección de este actuador y sus especificaciones, se tomó en cuenta el caso de Transmilenio, tal como se muestra en la siguiente imagen:



Figura 6. Ejemplo actuador pantalla LCD para la muestra de los tiempos de demora de los buses.

Es importante tener en cuenta que esta pantalla LCD únicamente se implementará en las estaciones que sean aptas para su implementación, como se puede ver en la *Figura 4*; en otras estaciones, como la presentada en la *Figura 5*, no se dispondrá de esta pantalla; es decir que únicamente se utilizarán como nodos estación intermedios que retransmiten la información a las demás estaciones.

ALIMENTACIÓN

Para la alimentación del sistema, se debe pensar en ambos casos de aplicación, tanto en el dispositivo que se tendrá en el bus, como el que se tendrá en las estaciones; adicionalmente, del montaje de la alimentación en los diferentes tipos de estaciones.

Para esto, se propone el uso de paneles solares, similar al caso de aplicación de *Tembici*, cuyos dispositivos centrales se alimentan de paneles solares instalados a la par con las estaciones. Siguiendo este esquema de alimentación, y teniendo en cuenta que los dispositivos que se van a emplear en el proyecto propuesto están altamente optimizados en cuanto al consumo de energía y no requieren de un procesamiento muy complejo, estos paneles solares requerirían menores dimensiones. Para el nodo bus, dado que los dispositivos de transmisión no requieren mayor energía para funcionar, se tomó la decisión de conectarlos a la fuente de poder del vehículo para no incurrir en mayores artefactos.



Figura 9. Sistema de alimentación para los nodos estación

Para las estaciones intermedias y finales, donde el tamaño y el peso no son una limitante tan grande, podrían instalarse paneles de mayor tamaño, por ejemplo, de 100cm x 100cm con una potencia de 200W a 400W, para soportar el consumo de las pantallas de visualización y la electrónica de comunicación. Estos paneles alimentarían directamente a los dispositivos durante el día y cargarían baterías de reserva para su operación durante la noche o en periodos de baja luminosidad solar.

Adicionalmente, es crucial que todos los componentes electrónicos estén diseñados para operar con la máxima eficiencia energética posible. Por ejemplo, las pantallas LCD deberían ser de tipo LED-backlit para un consumo mínimo de energía con la máxima visibilidad. Los sistemas de comunicación deben ser igualmente eficientes, utilizando protocolos de transmisión de datos de baja potencia como los que se mencionarán más adelante, para la comunicación entre nodos.

DETERMINAR LOS PROCESOS A REALIZAR EN EL GATEWAY IOT

El gateway desempeñará varias funciones clave que incluyen el reenvío, procesamiento, visualización y gestión de datos. A continuación, se presentan en detalle sus características de funcionamiento:

- **Reenvío de datos:** El gateway presente en los paraderos debe retransmitir los datos de los buses a los paraderos cercanos, con el fin de obtener el tiempo de llegada de cada ruta, al realizar el procesamiento de datos respectivo sobre los datos que recibe. Cada paradero transmitirá información en tiempo real sobre el paso de un autobús a todos los paraderos en la ruta de ese autobús para mantenerlos informados sobre el tiempo que falta para llegar a cada paradero.
- **Gestión:** La gestión de actualizaciones, en lo que respecta a cambios de ruta dinámicos, es esencial para mantener actualizados tanto los paraderos como la información de los autobuses. La gestión remota desempeñará un papel fundamental en este proceso, dada la gran cantidad de paraderos disponibles. Los gateways también serán responsables de comunicar métricas relevantes a bases de datos centrales para procesos de análisis y control de tiempos en los autobuses.

Por otra parte, las pantallas LCD estarán directamente conectadas al gateway, lo que significa que la información que se muestra en las pantallas se generará mediante un procesamiento de datos disponible. Este procesamiento incluirá el uso de métricas numéricas, como promedios de tiempo, que se relacionarán con los paraderos cercanos a un autobús específico.

SELECCIONAR EL GATEWAY IOT A UTILIZAR

Para el diseño del gateway IoT en el sistema de transporte público de Bogotá, se adoptará el kit Wi-Fi HaLow, específicamente el modelo STK-AH9066, que sigue el estándar 802.11ah de Wi-Fi HaLow. Este gateway se caracteriza por su capacidad de extender la cobertura de la red Ethernet hasta varios cientos de metros, mucho más allá de los repetidores Wi-Fi convencionales, aunque con un bitrate menor.

El STK-AH9066 opera en bandas de frecuencia de 902-930MHz o 760-790MHz y puede alcanzar un alcance de hasta 1 km para video 1080p con línea de visión directa, o hasta 150 metros a través de 3-4 paredes de concreto. Además, cuenta con un puerto Ethernet 10/100M, una antena de 2dBi, y un suministro de energía de 5V/1A

Dada su arquitectura basada en un microcontrolador Arm Cortex-M0/M3 con firmware en lenguaje C y su capacidad de funcionar como punto de acceso o cliente (STA), este gateway es ideal para integrarse en las estaciones de transporte público. Su capacidad para penetrar obstáculos como paredes y su configuración flexible

lo hacen adecuado para establecer una red dinámica y resiliente en el sistema de transporte, facilitando la comunicación en tiempo real entre buses y estaciones sin depender de GPS o internet constante

Esta implementación permitirá una red de comunicación altamente eficiente entre buses y estaciones, recolectando y transmitiendo datos clave para la gestión del tráfico y la experiencia del usuario, como tiempos de llegada y ubicaciones en tiempo real. La combinación de estas tecnologías mejorará la coordinación y planificación de rutas, reduciendo tiempos de espera y optimizando la gestión de flotas.

Se espera que su comunicación sea basada en un router sencillo donde se conectará el Access Point, y tanto los paraderos sencillos como los buses tendrán las antenas que serán STA que harán la retransmisión de los datos entre los paraderos para que su respectivo panel LCD que se encargue de mostrar la información en pantalla.

ARQUITECTURA DE COMUNICACIÓN

PROTOCOLOS DE CAPA FÍSICA Y DE ENLACE

El protocolo elegido para la comunicación entre dispositivos es *WiFi Halow*, y *Thread*, tanto para comunicar los gateways entre las estaciones como para la comunicación entre vehículo y estación para los datos y variables propias del bus.

WiFi Halow es un protocolo de comunicación de baja potencia y largo alcance que ofrece una solución WiFi IoT para aplicaciones de automatización del hogar, industria inteligente, ciudades inteligentes, entre otras. El cual opera en el espectro por debajo de un gigahercio para ofrecer un mayor alcance de hasta 1 km y una conectividad de menor potencia. Este protocolo trabaja en la banda de 900 MHz y puede tener una tasa de transferencia de hasta 347 kbps, el cual trabaja con redes de malla (*mesh*) que son autoorganizativas y auto-reparables.

Por lo cual, es ideal para el envío de nuestros paquetes y para la comunicación entre gateways y estaciones, debido a que la distancia promedio entre estaciones esta entre 300 y 500 metros, y *WiFi Halow* tiene un alcance de hasta 1 km. También, es apropiado en cuanto al consumo de energía al ser orientado hacia IoT, lo cual es ideal para el entorno operativo de los autobuses, que pueden tener limitaciones en cuanto a la capacidad de la batería y el acceso a fuentes de carga.

Para agregar *WiFi Halow* está bajo el estándar IEEE 802.11ah, siendo esto una extensión del estándar de *WiFi* por lo cual facilita la conexión directa de los dispositivos a redes IP. Lo cual permite que cada autobús y estación tengan su propia dirección IPv6. Esto elimina los problemas de direccionamiento y hace que la red sea más fácil de integrar y escalar dentro del ecosistema de Internet actual.

La capacidad de *Wifi Halow* para operar eficientemente con paquetes de datos de pequeña y mediana escala, específicamente con un tamaño promedio de 248 bytes, se ajusta adecuadamente al volumen de datos generado por las estaciones de autobuses. Este ajuste es congruente con las necesidades de transmisión de información relacionada con los tiempos estimados de llegada de los autobuses, una funcionalidad crítica para la mejora de la experiencia del usuario en el sistema de transporte. Además, la eficiencia en el tamaño de los paquetes promueve una transmisión rápida y confiable de la información, fundamental en un entorno donde la prontitud de la información es esencial para la toma de decisiones tanto de los usuarios como de los operadores del sistema.

WiFi Halow al ser compatible con dispositivos que funcionan con energía limitada, no requiere una inversión significativa en mantenimiento y es altamente compatible con los dispositivos que funcionan con energía limitada, como es el caso de las estaciones de autobuses que utilizan paneles solares como fuente de energía. Estas características hacen de *WiFi Halow* una opción sostenible y económicamente viable para el proyecto.

Asimismo, la arquitectura de red de malla que ofrece *WiFi Halow* y al ser compatible con IPv6, es particularmente beneficiosa para el dinámico entorno de transporte público. Al proporcionar direcciones IPv6 únicas a cada autobús, lo cual elimina la complejidad del direccionamiento en una red donde los nodos, es decir, los autobuses, están en constante movimiento. Esta característica facilita la identificación precisa y el rastreo de los vehículos en tiempo real, una funcionalidad clave para la actualización de tiempos de llegada y para la gestión eficiente de la flota.

Además, la capacidad de *WiFi Halow* para manejar de forma segura y eficiente la transmisión de datos en una red de malla robusta es crucial. Dada la naturaleza de las rutas de los autobuses, que incluye constantes paradas y arranques, así como variaciones en las rutas, la red debe ser capaz de adaptarse y reconfigurarse en tiempo real. Para agregar este protocolo es escalable en la inclusión de nuevos dispositivos siendo esto fundamental para el proyecto. A medida que la ciudad de Bogotá expande su flota de autobuses y red de estaciones, la infraestructura de comunicación debe poder crecer sin problemas. *WiFi Halow* es capaz de integrar nuevos dispositivos sin la necesidad de una reconfiguración extensiva de la red, lo cual es esencial para un sistema de transporte que busca evolucionar y mejorar continuamente.

Finalmente, *WiFi Halow* es el protocolo ideal gracias a las fortalezas ya mencionadas, se acopla a las necesidades de la red, gracias a su capacidad de formar redes de malla extensas y su bajo consumo de energía, además proporciona medidas de seguridad que garantizan la integridad de los datos al momento de su envío, siendo esto fundamental a la hora de estimar los tiempos de llegada de cada bus.

PROTOCOLO DE ENRUTAMIENTO

El protocolo RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) se distingue como el más adecuado para nuestra red de estaciones energizadas por paneles solares debido a su diseño especializado para optimizar el rendimiento en entornos de energía limitada. Esta optimización se logra a través de una gestión eficiente del tráfico y una topología de red adaptativa, donde el protocolo ajusta dinámicamente las rutas para mantener la eficiencia energética y la fiabilidad de la comunicación. RPL organiza los nodos en una estructura de DAG (*Directed Acyclic Graph*), permitiendo una transmisión de datos coherente y estructurada hacia el gateway, y utiliza el *Trickle Algorithm* para minimizar la congestión y el uso excesivo de energía. Además, su capacidad para dirigir tanto el tráfico multicast como unicast favorece la eficacia en la entrega de paquetes. Estas características lo hacen ideal para nuestras estaciones de transporte público, asegurando una red robusta y sostenible.

Además, RPL es versátil en la gestión de tráfico multicast y unicast, lo que garantiza una entrega eficiente de paquetes en diversas situaciones de comunicación, desde anuncios generales hasta mensajes específicos dirigidos a un solo nodo. Para afrontar problemas comunes de enrutamiento como el reenvío infinito de paquetes, el protocolo emplea el campo TTL, que descarta los paquetes cuando el contador alcanza cero, eliminando los bucles de retransmisión y garantizando una red más estable y predecible. Estas ventajas hacen de RPL una solución robusta y sostenible para las necesidades de comunicación de los gateways entre estaciones.

Principio del formulario

A continuación, se contempla un ejemplo de cómo se vería el paquete de datos al momento de enrutar entre los gateways de las estaciones de buses:

```
{
  "type": "routing",
  "station": {
    "id": "0c6ac0dc-9c59-4e47-89d0-fde05fb08165"
  },
  "message": {
```



```

    "bus": {
      "id": "58361192-0081-4aff-9f32-15a265f613e7",
      "velocity": 35.2
    },
    "route": {
      "id": "9adfd734-0087-4fcb-9f39-e0e3420b36e9",
      "name": "97",
      "direction": 1
    },
    "timestamp": "2023-11-10T12:15:21.000Z"
  }
}

```



paquete-enrutamiento.json

PROTOCOLO DE ENCAPSULAMIENTO

El protocolo 6LoWPAN ha definido mecanismos de encapsulación y compresión de cabecera que permiten enviar y recibir paquetes IPv6 a través de redes basadas en IEEE 802.15.4, su diseño facilita la transmisión eficiente de paquetes IPv6, para redes inalámbricas de baja potencia y velocidad, i.e., redes en donde los dispositivos puedan tener limitaciones de energía, memoria y/o procesamiento. Igualmente, presenta técnicas de compresión y fragmentación, que permiten optimizar el uso del ancho de banda (compresión), supera restricciones de enlaces de baja capacidad (fragmentación), y garantiza una transmisión efectiva de los datos.

6LoWPAN también ofrece compatibilidad con el protocolo de enrutamiento RPL, el cual es considerado en el sistema, por otra parte, está basado en el estándar IEEE 802.15.4 como Thread, que es usado como protocolo en la capa física y de enlace. Generando así una estandarización en los protocolos considerados, al no considerar una gran cantidad de estándares solo el 802.15.4 y el 802.11.ah de WiFi Halow.

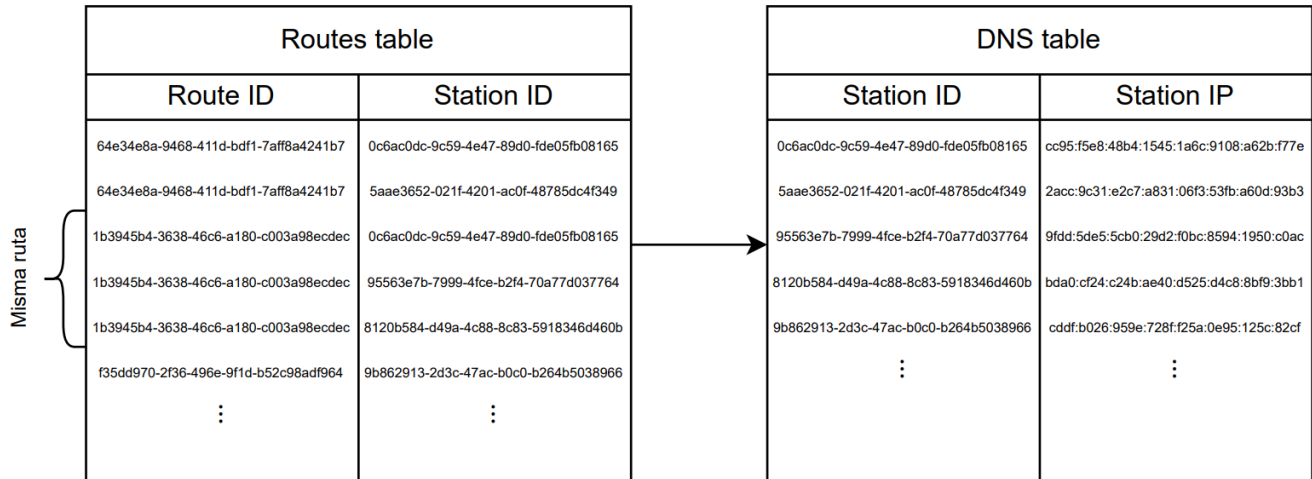
Además, 6LoWPAN al estar diseñado para funcionar con IPv6, ofrece un amplio espacio de direcciones, dejando de un lado problemas para el ingreso de un nuevo dispositivo a la red (bus), debido a que se pueden presentar adiciones de nuevos articulados al sistema de transporte. Por otra, parte su eficiencia energética es apta para nuestra red de estaciones energizadas por paneles solares, al ser apto a las limitaciones de energía que pueda presentar el sistema.

PROTOCOLO DE DESCUBRIMIENTO

El proceso de descubrimiento en nuestro sistema de transporte es crucial, especialmente durante la fase de implementación de todo el sistema y cuando se añade una nueva estación de autobús. Cada estación recién instalada debe identificar automáticamente las estaciones vecinas para establecer una comunicación eficaz, determinando cuáles serán fuentes de información y cuáles destinatarias de datos retransmitidos.

Para facilitar este proceso entre estaciones, para el proyecto se adoptará el protocolo Multicast DNS (mDNS) con mDNS se podrá limitar la comunicación a estaciones específicas mediante el almacenamiento de una tabla de enrutamiento, evitando así la ineficiencia y el gasto excesivo de energía que resultaría de un broadcast general. El mDNS nos permite mantener la red libre de tráfico superfluo, enviando datos solo a los puntos relevantes.

No obstante, para adaptar mDNS al proyecto, se debe realizar una implementación personalizada. Cada estación requiere conozca los autobuses que pasan por ella para dirigir adecuadamente el tráfico de información. Con este fin, los gateways de cada estación almacenan dos tablas esenciales durante el descubrimiento: una tabla de rutas y una tabla DNS. La tabla de rutas asocia cada estación vecina con las rutas de autobús pertinentes, mientras que la tabla DNS vincula cada estación con su respectiva dirección IP.



La eficiencia se maximiza retransmitiendo paquetes de datos únicamente a las estaciones necesarios, identificados por la ruta y número de autobús. Es importante recalcar que una misma ruta puede figurar en múltiples entradas dentro de la tabla de rutas, reflejando todas las estaciones pertinentes para la retransmisión de mensajes. Este enfoque asegura un uso energético óptimo y un proceso de enrutamiento preciso y automatizado.

A continuación, se puede observar un ejemplo de cómo sería el paquete de datos de respuesta cuando se realiza una solicitud de descubrimiento por parte de una estación nueva a las demás estaciones:

```
{
  "type": "discover",
  "station": {
    "id": "0c6ac0dc-9c59-4e47-89d0-fde05fb08165",
    "ip": "bda0:cf24:c24b:ae40:d525:d4c8:8bf9:3bb1",
    "routes": [
      "9adfd734-0087-4fcb-9f39-e0e3420b36e9",
      "fb35d1b4-c29f-4adb-a26c-d378509634a1",
      "23ccb391-d42e-42b6-9d09-662841909184"
    ]
  }
}
```



paquete-descubrimiento.json

En cuanto al protocolo de aplicación, decidimos que lo mejor es emplear MQTT. Principalmente debido a su patrón de mensajes publicación-suscripción, ya que este patrón proporciona una arquitectura de comunicación distribuida que se adapta de manera óptima a la naturaleza descentralizada y dinámica de nuestro sistema. De esta forma, la implementación de un publicador-suscriptor implica que los nodos, como los sensores en las estaciones y los autobuses, pueden comunicar información de manera eficiente y selectiva. Cuando un nodo (publicador) tiene datos relevantes, simplemente publica un mensaje en un "tema" específico. Otros nodos (suscriptores) interesados en esos datos se suscriben a ese tema y reciben automáticamente la información relevante cuando está disponible. En otras palabras, las estaciones únicamente estarán al tanto de la información de las estaciones previas que corresponden a las rutas que transitan por ellas, evitando así un broadcast de información que inundaría el sistema.

Por otro lado, se tuvo en cuenta que este protocolo utiliza un MQTT broker, el cual actúa como intermediario entre los nodos que publican información (sensores, autobuses) y aquellos que la reciben (pantallas en estaciones), de tal forma que se consideró lo siguiente. Primero, el uso de este broker simplifica la lógica de comunicación entre los nodos, ya que no necesitan conocer directamente la ubicación o estado de cada otro nodo. De igual manera, facilita el desacoplamiento entre los nodos en la red. Los nodos publicadores y suscriptores no necesitan conocerse directamente, lo que mejora la flexibilidad del sistema. Esto es esencial en un entorno de transporte público donde los autobuses y las estaciones pueden cambiar con el tiempo, y nuevos nodos pueden agregarse sin afectar la lógica de comunicación existente. Asimismo, el broker es responsable de gestionar la entrega de mensajes. Garantiza que los mensajes se enruten correctamente a los nodos suscritos, incluso si estos están temporalmente desconectados. Además, el broker puede manejar la retención de mensajes, permitiendo que nuevos suscriptores reciban la última información disponible. Finalmente, se ofrece niveles de calidad de servicio (QoS) que permiten garantizar la entrega de mensajes, lo que es crítico para aplicaciones en las que la precisión y la puntualidad de la información son fundamentales.

Como complemento a lo anteriormente dicho, otra de las ventajas del uso de este protocolo de aplicación es la baja sobrecarga del ancho de banda. En un entorno de transporte público donde la disponibilidad de ancho de banda puede ser limitada, la baja sobrecarga de ancho de banda de MQTT es esencial. La eficiencia en la transmisión de datos significa que la cantidad de información intercambiada entre los nodos es mínima, reduciendo la carga en la red. Esto es especialmente beneficioso para garantizar una comunicación fluida y rápida en un sistema donde múltiples estaciones, autobuses y pantallas pueden estar transmitiendo y recibiendo datos simultáneamente. De igual forma, la adaptabilidad de MQTT a redes de baja potencia permite que los sensores, las pantallas y otros dispositivos en el proyecto puedan operar con fuentes de energía limitadas. En nuestro caso dado que estamos utilizando paneles solares es crucial maximizar la eficiencia energética, ya que las condiciones climáticas de Bogotá no siempre son ideales. También, MQTT permite que los dispositivos entren en estados de baja potencia cuando no están activamente transmitiendo o recibiendo datos, lo que contribuye significativamente a la conservación de la energía y prolonga la vida útil de las baterías.

Bajo este esquema MQTT, el gateway de cada estación dispondría un tópico al cual estaría suscrito para recibir los mensajes de los buses; por otra parte, los buses serían nodos publicadores que estarían constantemente buscando los tópicos disponibles y publicando con el mensaje que tengan disponible en el momento preciso.

INTEGRACIÓN FOG Y CLOUD

CLOUD

En la capa cloud, se llevará a cabo un proceso integral para gestionar y analizar los datos de los autobuses que transitan por los paraderos. Para propósitos de analítica de datos básica, con el fin de analizar la efectividad de las rutas en cuanto a tiempo de llegada, y conocer así problemas en las rutas en cuanto a número de buses disponibles y tiempos de espera adecuados para los usuarios. Así, el nodo base actuará como la estación central encargada

de enviar la información a la nube, estos nodos serán las estaciones core definidas anteriormente. A continuación, se detallan los componentes y funcionalidades clave de cloud para el proyecto:

- **Base de datos:** Tendrá una base de datos central la cual en primer lugar almacenará la información de las rutas y paraderos del sistema. Asimismo, recolectará el *timestamp* cuando un bus pase por un paradero para con ello realizar estadísticas de tiempos de espera y eficacia del algoritmo de tiempo de espera para con ello realizar mejoras de este.
- **Interfaz para analítica de datos:** Cloud proveerá una interfaz con el fin de visualizar los resultados en cuanto a efectividad del sistema, y realizar análisis básicos sobre la eficacia de este. En cuanto a la efectividad de los tiempos de espera estimados y eficacia de cada una de las rutas.
- **Detección de paraderos y/o buses con fallas:** Se registrará información sobre paraderos y/o buses que presenten fallas, ya sean de sistema operativo, hardware, software, o seguridad, permitiendo así una gestión proactiva de posibles incidencias.
- **Sistema de Backup integral:** Se implementará un sistema de respaldo para asegurar la integridad de la información crítica, como datos de paraderos y rutas de autobuses.
- **Actualizaciones:** Cloud proporcionará las actualizaciones necesarias de software hacia los nodos, para con ello llevar constancia de los distintos parches y actualizaciones que se llevan al sistema y con esto asegurar que sean verificadas y autorizadas.

Este sistema en la nube permitirá realizar análisis profundos sobre los tiempos de tránsito, la frecuencia de autobuses por ruta y evaluar la eficacia del algoritmo de cálculo de tiempos de espera. La infraestructura propuesta busca optimizar la gestión del transporte público y mejorar la experiencia de los usuarios mediante un análisis continuo y actualizado de los datos operativos. Asimismo, cloud permitirá llevar seguridad de los datos y del mismo sistema al almacenar un backup y ser el único que proporcione actualizaciones a los dispositivos conectados a la red, con ello se evita que terceros intenten realizar modificaciones a esta, e igualmente se lleve registro de la información vital para el sistema con sus medidas de seguridad respectivas.

FOG

En la capa fog, se llevará a cabo el procesamiento eficiente de datos antes de su envío a la nube y el almacenamiento temporal de datos como el tiempo de espera de un bus que no ha pasado por la estación. Esto también sirve como medida de seguridad en dado caso de una desconexión temporal de un nodo, al momento de su reingreso a la red se contará con los datos almacenados anteriormente y con ello se recuperará la información disponible antes de su desconexión. Así, cada nodo fog se encontrará en una estación, sin embargo, solo se enviará información hacia la nube desde las estaciones core, para con ello disminuir la cantidad de información que se manda a la nube y realizar el preprocesamiento de datos adecuado. A continuación, se detallan los componentes y funcionalidades clave de fog para el proyecto:

- **Segmentación de información por rutas:** La información se envía de manera segmentada por rutas, facilitando un procesamiento más efectivo en la nube. Además, se realiza un preprocesamiento de datos que incluye los tiempos y paraderos por ruta. Es decir, el algoritmo de tiempo de espera se realizará antes del envío de datos hacia cloud, para con ello obtener dicha información en tiempo real para con ello informar a los usuarios del tiempo de espera de cada ruta en las estaciones core.
- **Almacenamiento de datos temporales:** Almacena temporalmente datos de tiempo, como el tiempo estimado de llegada de un autobús en relación con su ID y/o nombre, hasta que el autobús llega al paradero. También registra la velocidad del autobús para informar al resto de las rutas.
- **Filtrado y Preprocesamiento de datos:** Realiza filtros y preprocesamiento de datos antes de enviarlos a la nube, en donde la nube recibirá el id del paradero junto con las rutas y el tiempo de llegada de estas, y si

llego o no en el tiempo estimado. Esta información la enviará cada cierto tiempo, para así disminuir el tráfico en la red y evitar sobrecargas en la misma.

- **Canal de Comunicación con Dispositivos:** Establece un canal de comunicación efectivo entre los nodos fog y los dispositivos para la actualización constante de datos, asegurando que la información esté siempre actualizada. Se usará MQTT debido a su baja latencia y confiabilidad en la transmisión de datos, lo que resulta fundamental para un intercambio de información eficiente.
- **Detección e informe de fallos en paraderos:** Realizará análisis periódicos con el fin de detectar posibles fallos en el paradero, con respecto a la lectura de datos, disponibilidad de la información, seguridad, entre otros. Al detectar un fallo informará sobre este inmediatamente a la nube, para así tener una gestión proactiva de posibles problemas operativos.

En resumen, la capa fog desempeña un papel estratégico al optimizar la información antes de su envío a la nube, mejorando así la eficiencia y la precisión en la gestión del transporte público. Al disminuir la latencia y el flujo de datos entre fog y cloud.

REQUERIMIENTOS DE SEGURIDAD

A NIVEL DE INFORMACION

Los requerimientos a nivel de información para este proyecto son cruciales para garantizar la seguridad, integridad y privacidad de los datos transmitidos y procesados dentro del sistema. Este nivel se enfoca en cómo se manejan los datos dentro del sistema, incluyendo su captura y transmisión. La importancia de estos requerimientos se centra en varios aspectos clave:

- Confidencialidad: Esta se logra mediante el envío de datos cifrados, lo cual no solo mantiene la integridad de la información sino que también garantiza el acceso exclusivo a dispositivos autorizados, siendo estos los únicos capaces de descifrar el mensaje. Es decir, que solo los buses y los paraderos estén en la capacidad de efectuar las comunicaciones. En el contexto de los protocolos utilizados, WiFi aplica el (Wi-Fi Protected Access (WPA2) mientras que MQTT emplea protocolos de seguridad como TLS. Esta diferenciación en los métodos de cifrado subraya la adaptabilidad de las estrategias de seguridad a las especificidades de cada protocolo, asegurando que la transmisión de datos se realice en un entorno seguro y controlado.
- Integridad: Otro aspecto crucial en los sistemas IoT, se asegura mediante la implementación de firmas digitales, como es el caso de TLS (Transport Layer Security). Esta medida tiene como objetivo principal garantizar que los datos no sean alterados durante su transmisión. Es fundamental mantener la confiabilidad y la precisión de los datos transmitidos, aspectos vitales en este sistema donde la exactitud de la información es crítica para su funcionamiento eficiente.
- Actualización: Se emplea un mecanismo seguro que garantiza la instalación exclusiva de actualizaciones auténticas y verificadas. Este proceso se complementa con una gestión eficaz de parches de seguridad y un monitoreo continuo, diseñado para detectar cualquier intento de acceso no autorizado o manipulación de la información. Este enfoque integral hacia la actualización no solo salvaguarda la seguridad del sistema sino que también asegura la integridad de los datos, reforzando la premisa de que la información recibida es exactamente la que se envió, libre de alteraciones o corrupciones no autorizadas.

En general, es importante asegurar el sistema a nivel de información puesto que así se previenen ataques de Man In The Middle en los cuales se podrían estar modificando los datos afectando los tiempos que aparecen en las pantallas. Y no solo eso, la analítica de datos se vería perjudicada de igual forma puesto que si se tienen datos erróneos los resultados extraídos no evidenciarían la verdad. Un ejemplo de un ataque que perjudicaría el sistema

sería la alteración de las velocidades con las que pasa el bus, en este sentido la siguiente parada mostraría tiempos inexactos que desorientarían a los usuarios del transporte. Otro caso sería un atacante que se lograra hacer pasar por un bus y le mande una señal al paradero, luego las personas esperando en la siguiente parada se quedarían esperando un bus que nunca llegará porque nunca existió.

A NIVEL DE ACCESO

Los requerimientos a nivel de acceso para este caso van más enfocados a los sistemas implementados hacia la nube. Puesto que realmente nuestros nodos no tienen mayor interacción ni acceso a los mismos, lo importante son los procesos en la nube los que tienen que tener un acceso restringido para garantizar la seguridad. De los aspectos importantes a resaltar, están los siguientes:

- Autorización: La implementación de un sistema de roles es fundamental para administrar eficientemente los derechos de acceso de los usuarios. Este enfoque permite definir y controlar de manera precisa quién tiene acceso a la nube y qué nivel de permisos posee cada usuario. Esta estratificación de acceso asegura que los usuarios puedan interactuar con el sistema solo en los niveles permitidos, lo cual es vital para mantener la integridad y seguridad del sistema. En sistemas complejos, como aquellos utilizados en redes de transporte público, la adecuada asignación de roles y permisos no solo facilita una gestión de acceso eficiente sino que también reduce significativamente el riesgo de accesos no autorizados o inapropiados a datos sensibles.
- Identificación y autenticidad: Para nuestro sistema es esencial asignar identificadores únicos a cada bus y paradero. Esta asignación única facilita el seguimiento y la gestión eficaz de cada entidad dentro del sistema. Además, el uso de certificados digitales para la autenticación asegura que solo los dispositivos autorizados puedan conectarse y comunicarse dentro de la red. Esta capa de seguridad es imperativa para prevenir la intrusión de dispositivos no autorizados, lo que podría comprometer la integridad y funcionamiento del sistema. Por tanto, la autenticación robusta y la identificación precisa son esenciales para mantener la seguridad y la eficiencia operativa de la red.

A NIVEL FUNCIONAL

Los requerimientos a nivel funcional son relevantes puesto que ayudan a garantizar la calidad del sistema comprometiéndose con la seguridad. Estos están enfocados en garantizar un sistema confiable que pueda recuperarse de fallas y este en pie. Algunos de los aspectos que vale la pena mencionar son:

- Disponibilidad: Esta se centra en mantener el servicio operativo y accesible en todo momento. Para garantizar una alta disponibilidad, se implementan respaldos periódicos de datos. Esta estrategia es fundamental para mitigar el riesgo de pérdida de datos, que podría ocurrir debido a fallos del sistema, errores humanos o ataques cibernéticos. Los respaldos regulares aseguran que, en caso de una interrupción del servicio o degradación del sistema, se pueda restaurar rápidamente la información vital, manteniendo así la continuidad del servicio y la confiabilidad del sistema.
- Manejo de Excepciones y Resiliencia: Estos aspectos se refieren a la capacidad del sistema para recuperarse de interrupciones o fallos. En nuestro contexto, se implementan mecanismos de recuperación para restablecer conexiones perdidas. Un enfoque común es el uso de memoria temporal en la arquitectura FOG, que permite almacenar temporalmente los datos hasta que se pueda restablecer la conexión. Esto asegura que, en caso de una desconexión temporal, no se pierda información importante y que el sistema pueda recuperarse rápidamente y continuar operando de manera efectiva. Por ejemplo, si un bus pierde la conexión con la red, la información sobre su ubicación y horarios se almacena

localmente y se sincroniza con el sistema central una vez que se restablece la conexión. O en caso de que un paradero se caiga, este preserva la información y apenas pueda se reconecta con la red y se actualiza.

- **Auto-organización:** Es una característica avanzada de la seguridad a nivel funcional que implica la capacidad del sistema para adaptarse y responder automáticamente a fallos o cambios en la red. En nuestro caso incluye la implementación de medidas que detecten fallas en la comunicación entre dispositivos o posibles caídas de los nodos. Un aspecto crucial de la auto-organización es la gestión de nodos. Si un nodo en la red falla, el sistema está diseñado para identificar automáticamente esta falla y redirigir la información a través de otros nodos funcionales. La verificación periódica de conexión en toda la red es otra estrategia importante para identificar y responder a la pérdida o fallo de un nodo. Esta capacidad de auto-organización asegura que el sistema sea robusto y pueda mantener su funcionalidad incluso en situaciones adversas, como la caída de un nodo crucial en la red.

BIBLIOGRAFÍA

1. *Infraestructura de Datos Espaciales de Bogotá.* (n.d.). Ideca. <https://www.ideca.gov.co/recursos/mapas/paraderos-zonales-del-sitp>
2. Secretaría Distrital de Movilidad. (2016). RESOLUCIÓN No. 264 de 2016. <https://www.movilidadbogota.gov.co/web/sites/default/files/Resolucion%20total.pdf>
3. Sitp. (2019, Julio 30). *Identifica los paraderos.* SITP. <https://www.sitp.gov.co/publicaciones/40572/paraderos-del-sitp/>
4. Sitp. (2020, septiembre 4). *El Rutero o Tabla de Ruta.* SITP. <https://www.sitp.gov.co/publicaciones/40082/el-rutero/>
5. CNX Software. (2021, January 11). MKR SharkyPro BLE, Zigbee, OpenThread development board follows Arduino MKR form factor. CNX Software. <https://www.cnx-software.com/2021/01/11/mkr-sharkypro-ble-zigbee-openthread-development-board-follows-arduino-mkr-form-factor/>
6. Thread Group. (n.d.). Developers: The nRF52840 is fully multiprotocol capable with full protocol concurrency of Thread and Bluetooth LE. Thread Group. <https://www.threadgroup.org/what-is-thread/developers>
7. Nordic Semiconductor. (n.d.). nRF52840 DK. Nordic Semiconductor. <https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dk>
8. Tembici. (2021). Informe de impacto ESG 2021. <https://www.tembici.com.br/wp-content/uploads/2022/12/Tembici-ESG-ESP-DIGITAL.pdf>

ANEXOS

En el siguiente [enlace de OneDrive](#), se pueden encontrar todos los archivos JSON de los paquetes ejemplificados en este documento. De igual manera, el ejemplo de las tablas de enrutamiento definidas en la etapa de descubrimiento.