



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Diseño de prototipo IoT para el
monitoreo del conteo de personas
en el sistema de transporte
Pumabús utilizando la red 0G-Sigfox**

TESIS

Que para obtener el título de
Ingeniero Mecatrónico

P R E S E N T A

Edgar Omar Hernández Sosa

DIRECTOR DE TESIS

Ing. Carlos Omar Calieca Romero



Ciudad Universitaria, Cd. Mx., 2025



**PROTESTA UNIVERSITARIA DE INTEGRIDAD Y
HONESTIDAD ACADÉMICA Y PROFESIONAL
(Titulación con trabajo escrito)**



De conformidad con lo dispuesto en los artículos 87, fracción V, del Estatuto General, 68, primer párrafo, del Reglamento General de Estudios Universitarios y 26, fracción I, y 35 del Reglamento General de Exámenes, me comprometo en todo tiempo a honrar a la institución y a cumplir con los principios establecidos en el Código de Ética de la Universidad Nacional Autónoma de México, especialmente con los de integridad y honestidad académica.

De acuerdo con lo anterior, manifiesto que el trabajo escrito titulado DISEÑO DE PROTOTIPO IOT PARA EL MONITOREO DEL CONTEO DE PERSONAS EN EL SISTEMA DE TRANSPORTE PUMABUS UTILIZANDO LA RED 0G-SIGFOX que presenté para obtener el título de INGENIERO MECATRÓNICO es original, de mi autoría y lo realicé con el rigor metodológico exigido por mi Entidad Académica, citando las fuentes de ideas, textos, imágenes, gráficos u otro tipo de obras empleadas para su desarrollo.

En consecuencia, acepto que la falta de cumplimiento de las disposiciones reglamentarias y normativas de la Universidad, en particular las ya referidas en el Código de Ética, llevará a la nulidad de los actos de carácter académico administrativo del proceso de titulación.

EDGAR OMAR HERNANDEZ SOSA
Número de cuenta: 314074038

Agradecimientos

A mi asesor, Carlos Calieca, por aceptarme como primer tesista y buscar calidad de trabajo. A mis amigos de Ninzu, por el empeño y motivación que pusieron en un proyecto que nos ilusionaba a todos. Al coordinador de la Dirección General de Servicios Generales y Movilidad, Agustín Valdés y a Lourdes M., por la atención e información brindada.

Deseo expresar mi gratitud a mi familia por su amor, apoyo incondicional en cada decisión que he tomado, por inculcarme valores y hábitos buscando mi bienestar. Por sus sacrificios y esfuerzo en todo lo que hacen. Por enseñarme a ser agradecido y a no rendirme sin luchar por lo que anhelo hasta conseguirlo. A mis padres, Enrique Hernández y Patricia Sosa, como seres humanos y profesionales de la salud. A mis abuelos: mi referente humano absoluto.

Agradezco a la Universidad Nacional Autónoma de México (UNAM), mi casa, por brindarme tantas oportunidades, en lo académico, lo cultural y lo deportivo.

A la Facultad de Ingeniería y a los excelentes profesores que me compartieron su conocimiento, su guía y me ayudaron a seguir forjando la visión de cómo quiero ser. A Juan Ocariz y Nydia Lara, por su énfasis en la correcta expresión del pensamiento. A Raúl Puente, por contagiar su pasión y refrescar la mente con *pláticas de viernes*. A Yair Bautista, por la claridad, sensatez e innovación en cada curso; por su tutela durante mi servicio social, y por presidir mi sínodo.

A Miguel Serrano, que me maravilló con la calidad de sus cursos, despertó en mí fascinación por la ciencia biomédica y me inspiró a orientar mis estudios hacia este campo.

A mis amigos de toda la vida: Kevin y Mau, por acompañarme en tantas etapas de mi vida y recordarme que los lazos fuertes perduran a pesar del tiempo y la distancia. Igualmente a Caro. A Mario, por ser un soporte desde el inicio e, incluso más, a la distancia.

A mis inseparables Oscar y Angel por tanto desde la prepa. Al entrenador Víctor Angoa y mis compañeros de Halcones Gimnasia, por marcar un antes y un después en mi persona. De ellos, a mi primer amigo: Ricardo, por siempre estar. A Adrián, por su amistad desde la prepa y a todos quienes me acompañaron en mi paso por la facultad.

A Bety, cuyo apoyo fue invaluable durante los momentos más complicados. Mi compañera en todo momento, que tanto dio y me enseñó.

A Tona, mi entrenador, amigo y segundo padre, por el equipo que ha formado con personas maravillosas, a quienes tengo el privilegio de llamar amigos: Norman, Dante, Marko, Andrés, Armando, Iván, Carlo, Néstor, Kevin, Raúl, Durango, Marty, Alex, Axel, Irvin, etc. A Frida, fiel a sus principios, ejemplo de congruencia y autenticidad. A Rebe, por los ánimos y calidez siempre. A Nat, por ser, estar y hacer posible que llegara al final de este proceso.

A Dora Fuentes, de la Dirección General de Orientación y Atención Educativa, por ser el mejor ejemplo de eficacia con calidez humana, por tratar a cada estudiante con dignidad y buscando siempre la resolución pronta y sensata. Y por haberme apoyado como becario en el programa de Prácticas Profesionales Ejercicio Fiscal 2024.

Extiendo mis agradecimientos al Programa de Vinculación con los Egresados de la UNAM, por el apoyo económico *Becas de Titulación Egresados Alto Rendimiento* para culminar mi titulación.

A los mencionados y a quienes se me haya pasado, gracias por formar parte de quien soy.

Índice general

INTRODUCCIÓN	1
1. MARCO TEÓRICO	3
1.1. Internet de las cosas	3
1.1.1. Cadena de valor del IoT	4
1.1.2. Hardware IoT	5
1.1.3. Redes de comunicación IoT	5
1.1.4. Software IoT	7
2. PLANTEAMIENTO DEL PROBLEMA	9
2.1. Representación gráfica del problema	13
3. ETAPA DE PLANEACIÓN DE DISEÑO	17
3.1. Entrevistas con usuarios	17
3.2. Identificación de requerimientos	24
3.3. Especificaciones	28
3.4. Evaluación comparativa en el mercado	30
3.4.1. Evaluación comparativa de soluciones - Software	31
3.4.2. Evaluación comparativa de soluciones - Dispositivos IoT	34
4. ETAPA DE DESARROLLO DE CONCEPTO	39
4.1. Evaluación comparativa en el mercado - Componentes para la propuesta de solución	39
4.1.1. Sensores para conteo de pasajeros	40
4.1.2. Módulos para obtención de ubicación	45
4.1.3. Red de comunicación	47
4.1.4. Comparación de servicios de software IoT en la nube: Ubidots y Losant	48
4.2. Definición del concepto	50
4.3. Cadena de valor IoT de la solución propuesta	51
4.4. Representación gráfica de la solución propuesta	53
5. DESARROLLO DE CONCEPTOS	56
5.1. Bocetos y Representaciones Visuales	56
5.1.1. Diagramas de concepto	60
5.1.2. Diagramas de caja negra	60
5.2. Arquitectura del sistema	60
5.2.1. Subsistemas	62
5.2.2. Diagrama de flujo de operación	66
5.3. Representación gráfica de la detección de ascensos y descensos por el sistema de conteo de pasajeros.	67

6. DESARROLLO DE CONCEPTO EN HARDWARE	71
6.1. Componentes del hardware	71
6.2. Esquemas eléctricos	73
6.3. Diseño de PCB	73
6.4. Evolución del diseño del dispositivo	74
7. PROCESAMIENTO DE INFORMACIÓN (FIRMWARE) Y TRANSMISIÓN A TRAVÉS DE LA RED DE COMUNICACIÓN SIGFOX	86
7.1. Creación del paquete de datos para envío a través de la red de comunicación Sigfox	95
8. DESARROLLO A NIVEL SOFTWARE	98
8.1. Aplicativo de servicios de software en la nube: Ubidots	98
8.2. Aplicativo de servicios de software en la nube: Losant	109
9. VALIDACIÓN DE FUNCIONAMIENTO DE LA SOLUCIÓN	124
10. ANÁLISIS DE RESULTADOS	131
10.1. Geolocalización	131
10.2. Conteo de pasajeros	134
11. CONCLUSIONES	138
11.1. Etapas siguientes del proyecto	141
Bibliografía	143

Índice de figuras

1.	Mapa de movilidad en Ciudad Universitaria.	2
1.1.	Cadena de valor IoT.	4
1.2.	Hardware IoT.	5
1.3.	Redes de comunicación.	6
1.4.	Software IoT.	8
2.1.	Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (a) y (b).	13
2.2.	Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (c) y (d).	14
2.3.	Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (e).	15
2.4.	Diagrama de fases del diseño.	16
2.5.	Diagrama de etapas que construyen el proceso de diseño en el presente proyecto.	16
4.1.	Componentes elegidos para conteo de pasajeros y geolocalización y PCB como herramienta de integración de diseño.	50
4.2.	Red Sigfox elegida y Kit de desarrollo Devkit 2.0.	51
4.3.	Cadena de valor IoT de la solución propuesta.	52
4.4.	Ejemplo de uso práctico de la solución propuesta (a) y (b).	53
4.5.	Ejemplo de uso práctico de la solución propuesta (c), (d) y (e).	54
5.1.	Boceto a mano alzada	56
5.2.	Boceto realizado con bolígrafo	57
5.3.	Vista frontal. Creado con Google Jamboard.	58
5.4.	Sujetador removible realizado con Google Jamboard.	58
5.5.	Vista trasera e interior con componentes. Creado con Google Jamboard.	58
5.6.	Vista frontal 3D. Realizado en Canva.	59
5.7.	Diagrama de caja negra genérico.	60
5.8.	Diagrama de caja negra. Sistema general.	61
5.9.	Diagrama de caja negra.Sistema de alimentación.	63
5.10.	Diagrama de caja negra. Sistema de recopilación de datos/sensado.	63
5.11.	Diagrama de caja negra. Sistema de geolocalización.	64
5.12.	Diagrama de caja negra. Sistema de conteo de pasajeros.	64
5.13.	Diagrama de caja negra. Unidad de procesamiento.	65
5.14.	Diagrama de caja negra. Sistema de comunicación.	65
5.15.	Diagrama de caja negra. Interfaz de usuario.	66
5.16.	Diagrama de flujo de operación.	67
5.17.	Secuencia de activación de infrarrojos a la entrada de un pasajero.	68
5.18.	Secuencia de activación de infrarrojos a la bajada de un pasajero.	70
6.1.	Esquema de flujo eléctrico del sistema.	73

6.2. Diseño de PCB en Eagle.	73
6.3. Vista de las pistas de conexión.	74
6.4. Vista final de la PCB.	74
6.5. Fotografías del Dispositivo en elaboración y distribución interna de componentes.	75
6.6. Interior de unidad Pumabús.	76
6.7. Fotografías del dispositivo.	77
6.8. Fotografías del Dispositivo. Distancia entre infrarrojos ajustable.	78
6.9. Fotografías del Dispositivo. Antena sobresaliente. Dimensiones reducidas.	78
6.10. Fotografías del Dispositivo. Antena sobresaliente. Dimensiones finales.	79
6.11. CAD de carcasa con dimensiones finales.	79
6.12. Vistas del prototipo desde las seis caras principales: frontal, trasera, superior, inferior, lateral derecha y lateral izquierda.	80
6.13. Vistas isométricas superiores e inferiores del CAD de la carcasa.	81
6.14. Vistas isométricas superiores e inferiores del CAD de la carcasa.	82
6.15. Planos y explosionado isométrico del CAD de la carcasa. Dimensiones finales.	83
6.16. Carcasa impresa en 3D con componentes ensamblados.	84
6.17. Carcasa impresa en 3D. Vista trasera con tapa puesta.	84
6.18. Carcasa impresa en 3D. Vista trasera de componentes internos.	85
6.19. Carcasa impresa en 3D montada en unidad Pumabús.	85
7.1. Diagrama de flujo del código del sistema.	86
7.2. Callback en Sigfox para Losant.	97
8.1. Variables en Ubidots.	98
8.2. Íconos o widgets.	99
8.3. Opción para desarrollo con HTML en Ubidots.	99
8.4. Vista de tablero de desarrollador.	100
8.5. Registro de valores en Ubidots.	101
8.6. Gráficas de registro histórico de valores en Ubidots.	101
8.7. Vista de pasajeros.	102
8.8. Vista de desarrollador.	103
8.9. Vista de coordinador.	103
8.10. Ruta.	104
8.11. Placas.	104
8.12. Ocupación (a) Poca gente (b) Algo de gente.	105
8.13. Texto con la ocupación.	105
8.14. Siguiente estación.	105
8.15. Subidas en la última estación.	106
8.16. Bajadas en la última estación.	106
8.17. Recorrido de pumabús en el mapa.	107
8.18. Gráfica subidas y bajadas.	107
8.19. Gráfica de pasajeros.	108
8.20. Registro histórico de valores.	108
8.21. Variables en Losant.	109
8.22. Bloques para tableros en losant.	110

8.23. Registro de valores en Losant. Tablero de Desarrollador	111
8.24. Vista de desarrollador.	112
8.25. Vista de pasajeros.	112
8.26. Vista de coordinador.	113
8.27. Bloque de información de unidad.	114
8.28. Bloque Estación previa.	114
8.29. Bloque Siguiente estación.	114
8.30. Bloque Personas en el pumabús.	115
8.31. Bloque Mapa.	115
8.32. Bloque Ascensos en la última estación.	116
8.33. Bloque Descensos en la última estación.	116
8.34. Bloque Tabla de registro histórico.	117
8.35. Bloque Gráficas de registro histórico.	117
8.36. Diagrama de bloques del flujo de trabajo en Losant. Obtención de datos de sigfox.	119
8.37. Webhook con enlace generado.	119
8.38. Código del bloque de Función Formato de hora.	120
8.39. Bloque Device:State de salida del flujo de trabajo.	120
8.40. Diagrama de bloques del flujo de trabajo en Losant. Agregación a tabla de registro histórico.	121
8.41. Bloque Device:State de inicio de flujo de trabajo.	121
8.42. Bloque de inserción de datos a la tabla de registro y despliegue.	122
10.1. Detecciones correctas por estación.	133
10.2. Consistencia en la identificación de estaciones.	133
10.3. Registro de mensajes recibidos en Losant.	136
10.4. Graficas de los datos recibidos en Losant durante las pruebas.	137

Índice de tablas

1.1.	Comparación de cuatro tipos de redes de comunicación.	7
1.2.	Principales plataformas de servicios IoT en la nube.	8
2.1.	Distribución de unidades Pumabús en 13 rutas.	9
2.2.	5W 2H.	13
3.1.	Entrevista Coordinador de Pumabús	19
3.2.	Entrevista Pasajero.	21
3.3.	Entrevista personal de monitoreo.	23
3.4.	Tabla de requerimientos.	27
3.5.	Tabla de especificaciones.	29
3.6.	Benchmarking Plataformas.	33
3.7.	Benchmarking Dispositivos.	37
4.1.	Benchmarking sensores para conteo de pasajeros.	44
4.2.	Benchmarking módulos para obtención de ubicación.	46
4.3.	Comparación de redes de comunicación IoT.	48
4.4.	Comparación entre plataformas IoT: Ubidots y Losant	49
6.1.	Componentes y materiales utilizados en el proyecto.	72
6.2.	Características finales del dispositivo.	85
7.1.	Estructura del paquete de datos (payload) enviado a través de la red Sigfox	95
7.2.	Ej. de valores contenidos por byte en la estructura del paquete de datos construido.	96
7.3.	Datos manejados en el backend de Sigfox	96
8.1.	Resumen de decisiones técnicas y su justificación.	123
9.1.	Metodología para realización de pruebas de funcionamiento de la solución.	126
9.2.	Pruebas de funcionamiento general. Geolocalización	127
9.3.	Pruebas de funcionamiento general. Conteo de pasajeros. Primera vuelta.	128
9.4.	Pruebas de funcionamiento general. Conteo de pasajeros. Segunda vuelta.	129
9.5.	Pruebas de funcionamiento general. Conteo de pasajeros. Tercera vuelta, inmediatamente después de la segunda.	130
10.1.	Tasa de identificación correcta de estaciones en 3 vueltas	132

INTRODUCCIÓN

El sistema de transporte requerido por una comunidad depende directamente del tamaño de su población. De la misma manera, los cambios o adaptaciones necesarios para su correcto funcionamiento están dados por la forma en que evoluciona la población que lo usa.

Si la población se mantiene o decrece, el sistema puede no requerir más que mantenimiento regular, pero si se incrementa significativamente, llega a ser necesario implementar estrategias que pueden incluir el aumento en el número de unidades de transporte, el incremento en la frecuencia de paso, la modificación o implementación de nuevas estaciones, rutas o vialidades; la creación de medios de transporte alternativos, así como la utilización de herramientas tecnológicas para la coordinación y comunicación entre personal operativo y usuarios.

Algunos síntomas de que la capacidad de un sistema de transporte está por ser rebasada son las cantidades de tiempo requeridas para desplazarse de un sitio a otro mayores a las esperadas y una incertidumbre de los usuarios acerca del estado de los medios de transporte en un momento cualquiera, es decir, un desconocimiento de la disponibilidad de unidades de transporte, de los tiempos de espera requeridos para poder tener acceso a una unidad y, de ser el caso, si resultaría conveniente cambiar la forma de desplazamiento.

En Ciudad Universitaria, la Universidad Nacional Autónoma de México (UNAM) ofrece, de manera gratuita, a la comunidad universitaria y a los visitantes el servicio de autobuses Pumabús para trasladarse hacia las escuelas, facultades, institutos, centros, dependencias administrativas e instalaciones culturales dentro del campus. El sistema de transporte interno de pasajeros está compuesto principalmente por unidades de Pumabús y bicicletas conocidas como Bicipuma, distribuidas en distintas estaciones.

A lo largo de los años en los que se ha contado con el servicio de Pumabús, han habido cambios, como la modificación de rutas existentes, la creación de nuevas rutas, nuevas estaciones o paraderos y hasta la desaparición tanto de estaciones como de rutas completas. Estos cambios responden a la variación en la demanda por parte de los usuarios, que puede concentrarse en ciertos puntos del campus por encima de otros.

Actualmente hay 13 rutas de Pumabús en operación, distribuidas como en el mapa de la Figura 1, con hasta 60 unidades en funcionamiento diario a través de carriles confinados con 93 parabuses. Existen centros de dirección, coordinación y monitoreo que dependen de la Dirección General de Servicios Generales y Movilidad (DGSGM) y gracias a los cuales se cuenta en todo momento con información actualizada de las condiciones y eventualidades que puedan surgir concernientes al sistema de movilidad de toda Ciudad Universitaria.

En este contexto, el presente trabajo tiene como propósito desarrollar un prototipo de sistema capaz de identificar las estaciones de una ruta del servicio Pumabús, realizar el conteo del número de pasajeros a bordo y transmitir esta información actualizada mediante tecnología IoT utilizando la red 0G-Sigfox. Se busca evaluar la viabilidad técnica de la solución propuesta, su integración con plataformas de monitoreo en la nube, así como su capacidad para brindar información útil tanto para usuarios como para coordinadores del sistema. La implementación y validación del prototipo permitirán valorar su potencial como herramienta de apoyo en la mejora de la eficiencia del transporte dentro del campus universitario.

MAPA INTEGRAL DE RUTAS PUMABÚS Y CICLOPISTA



- Ruta 1 Metro Universidad
- Ruta 2 Metro Universidad
- Ruta 3 Metro Universidad
- Ruta 4 Metro Universidad
- Ruta 5 Metro Universidad
- Ruta 6 Estadio Olímpico
- Ruta 7 Estadio Olímpico
- Ruta 8 Estadio Olímpico
- Ruta 9 Metrobús CU
- Ruta 10 Metrobús CU
- Ruta 11 Metrobús CU
- Ruta 12 E.N.A.L.L.T. Anexo de Filosofía
- Ruta 13 Zona Cultural

SIMBOLOGÍA



HORARIO

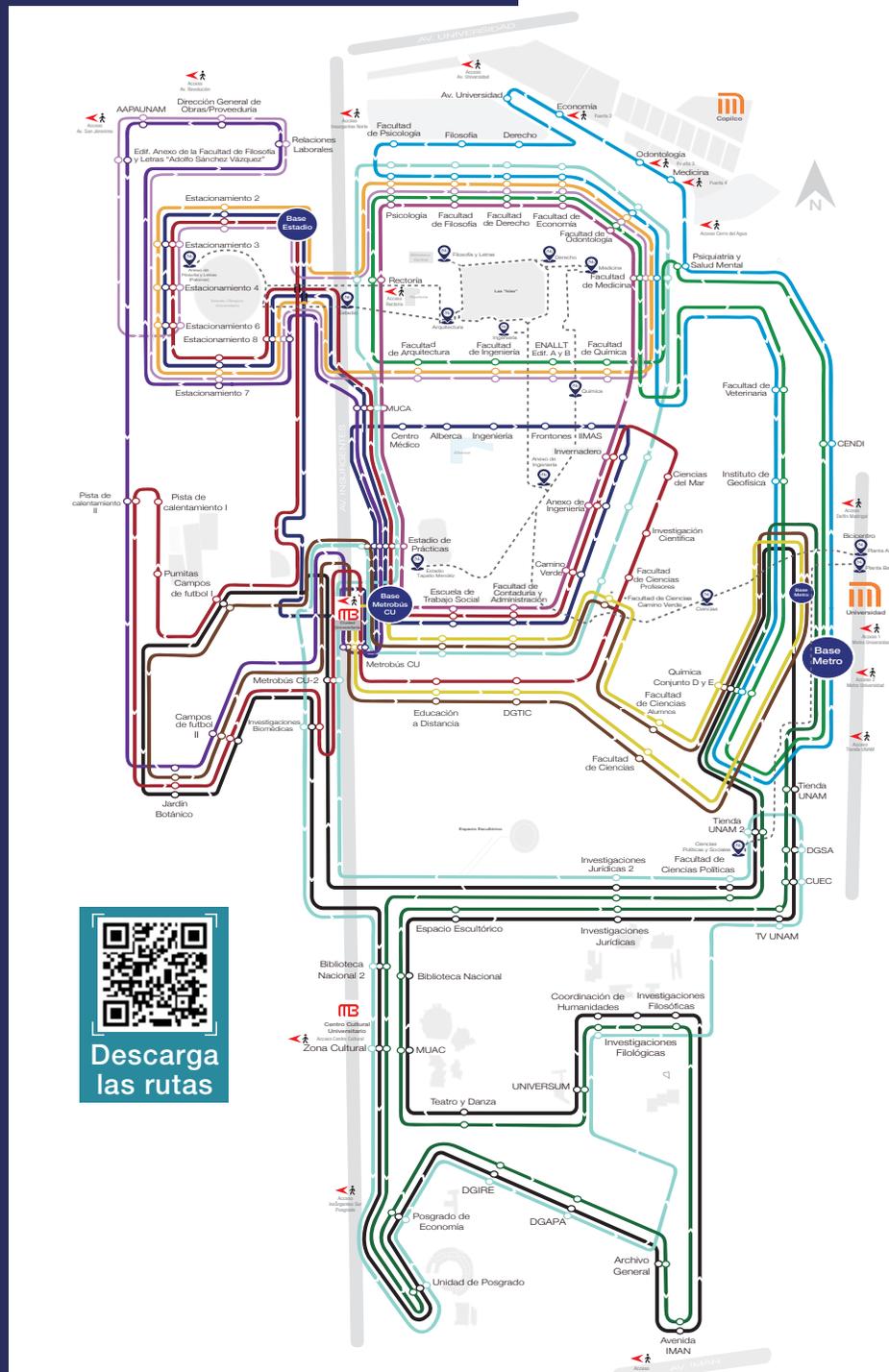
Lunes a viernes
Todas las rutas
De 6:00 a 22:00 horas.

Sábado
Rutas 1, 2, 4, 5 y 9
De 6:00 a 15:00 horas.

Domingo
Rutas 3 y 10
De 6:00 a 23:00 horas.

Solicita el Transporte Especial para personas con discapacidad al:

55 56 22 6151.



Servicio gratuito



Figura 1: Mapa de movilidad en Ciudad Universitaria.

1 MARCO TEÓRICO

1.1 Internet de las cosas

Este concepto es central para el desarrollo del presente proyecto, ya que el dispositivo propuesto forma parte de la categoría de objetos conectados que adquieren, procesan y transmiten datos a plataformas remotas.

El término “Internet de las cosas” (IoT por las siglas en inglés de Internet of Things) apareció poco antes del inicio del siglo XXI y describe a los sistemas constituidos por tres elementos fundamentales: hardware, software y redes de comunicación. Estos componentes están interconectados con fines específicos y, a su vez, están conformados por sensores, microcontroladores, microprocesadores, antenas y módulos de comunicación. Algunas definiciones de IoT que servirán para entender el concepto de manera clara se presentan a continuación:

- La Internet de las cosas (IoT) describe la red de objetos físicos (“cosas”) que llevan incorporados sensores, software y otras tecnologías con el fin de conectarse e intercambiar datos con otros dispositivos y sistemas a través de Internet [1].
- Una red de dispositivos físicos, vehículos, electrodomésticos y otros objetos físicos que están integrados con sensores, software y conectividad de red que les permite recopilar y compartir datos. Estos dispositivos, también conocidos como “objetos inteligentes”, pueden variar desde simples dispositivos “domésticos inteligentes”, como termostatos inteligentes, hasta dispositivos portátiles como relojes inteligentes y ropa con RFID, hasta complejas maquinarias industriales y sistemas de transporte. Los tecnólogos incluso están imaginando “ciudades inteligentes” enteras basadas en tecnologías de IoT.[2].
- Una infraestructura global para la sociedad de la información que habilita servicios avanzados mediante la interconexión (física y virtual) de cosas basadas en tecnologías de la información y comunicaciones interoperables ya existentes y en evolución; y en donde la ‘cosa’ es un objeto del mundo físico (cosas físicas) o del mundo de la información (cosas virtuales) que puede identificarse e integrarse en redes de comunicación. Unión Internacional de Telecomunicaciones (ITU, por sus siglas en inglés) [3].
- La red de objetos físicos: dispositivos, instrumentos, vehículos, edificios y otros elementos integrados con electrónica, circuitos, software, sensores y conectividad de red que permite a estos objetos recopilar e intercambiar datos. Sin embargo, la definición exacta de IoT aún se encuentra en proceso de formación y está sujeta a las perspectivas adoptadas. En general, se ha definido a IoT como una “infraestructura dinámica de red global con capacidades de autoconfiguración basadas en estándares y protocolos de comunicación”. [4]

Una característica de los sistemas IoT es el manejo de la información, ya que, para ser considerados como tales, es necesario que los datos obtenidos a través de las etapas de sensado y procesamiento estén disponibles, pues con las comunicaciones máquina a máquina, máquina

a persona y persona a persona, a través de las redes y dispositivos, es posible llevar a cabo un análisis de la información que entra al sistema.

Los sistemas IoT tienen una enorme cantidad de posibles aplicaciones y, según el caso, dependiendo de los objetivos que se tengan, así como de las limitaciones presentes, habrá tecnologías más adecuadas que otras. La elección de una tecnología de transmisión de información sobre otra depende de factores como las áreas a cubrir y la cantidad de información que se envía y se recibe, así como el tiempo que esto implica, considerando la frecuencia de reporte requerida.

1.1.1 Cadena de valor del IoT

Los tres pilares tecnológicos que constituyen la cadena de valor del IoT son el hardware IoT, las redes de comunicación IoT y el software IoT. Como elementos complementarios, la identificación del caso de uso y del usuario final terminan de configurar este ecosistema, tal como se ilustra en el diagrama de la Figura 1.1. Esta conceptualización es relevante, ya que determina las arquitecturas, protocolos y herramientas empleadas a lo largo del desarrollo.



Figura 1.1: Cadena de valor IoT.

1.1.2 Hardware IoT

El primer eslabón de la cadena de valor del IoT, correspondiente al hardware, está formado por elementos físicos tangibles que permiten la obtención, procesamiento y envío de datos. El hardware propio de los sistemas IoT, o hardware IoT, incluye sensores, ya sean analógicos o digitales, que recopilan datos; microprocesadores y/o microcontroladores, que dan soporte para convertir los datos en información valiosa; módulos de comunicación, que hacen posible la recepción y transmisión de la información obtenida de los sensores, a través de redes y protocolos de comunicación correspondientes; y antenas, que amplían la capacidad de comunicación inalámbrica.

Asimismo, las baterías y/o conexiones eléctricas desempeñan un papel fundamental al suministrar la energía necesaria para el funcionamiento continuo de todo el sistema, garantizando así su autonomía y eficacia en entornos IoT.

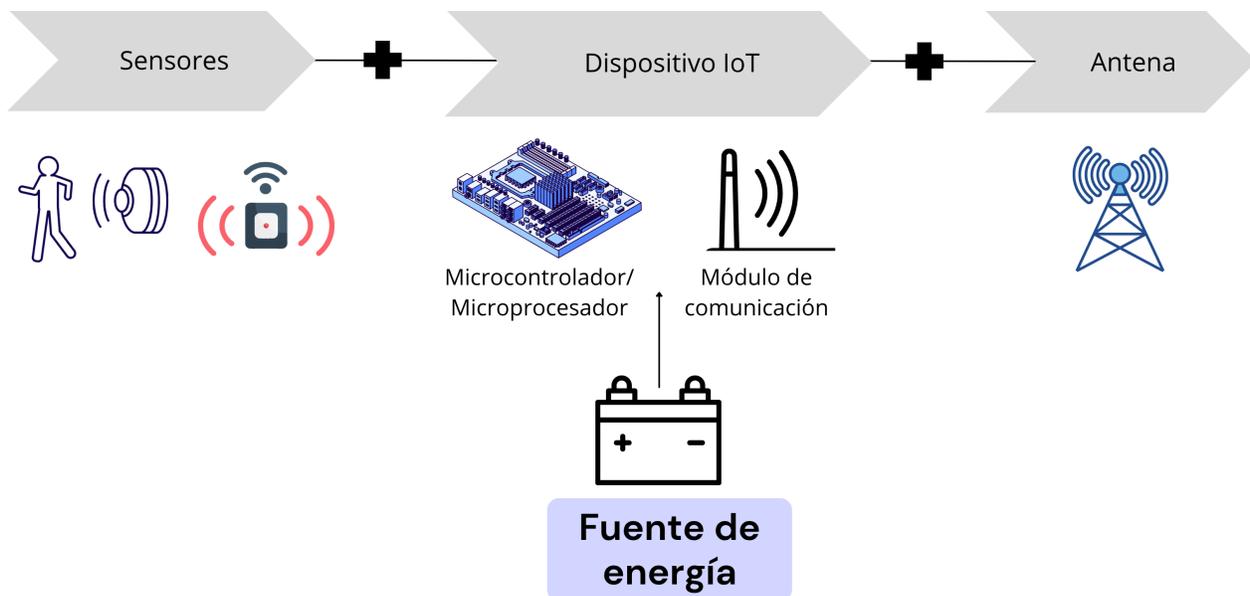


Figura 1.2: Hardware IoT.

1.1.3 Redes de comunicación IoT

Si entendemos una red de comunicación como un conjunto de elementos interconectados de modo que puedan compartir información entre ellos, podemos clasificar las redes de comunicación, según el medio físico de conexión, en alámbricas e inalámbricas.

También existe la tecnología de red mesh, que forma una malla de nodos que pueden comunicarse entre sí de manera inalámbrica, propagando la información por distintos caminos posibles hasta llegar al destino. Este tipo de red es especialmente útil en entornos con obstáculos o cuando se busca mayor resiliencia ante la caída de nodos individuales, ya que permite rutas alternativas de comunicación. En el contexto del IoT, las redes mesh se implementan casi exclusivamente mediante tecnologías inalámbricas, por lo que así se considera en este trabajo.

Las redes inalámbricas pueden ser clasificadas según sean de corto o largo alcance (Figura 1.3). Entre los protocolos de comunicación inalámbrica de corto alcance (Local Area Network - LAN) se encuentran Bluetooth, Wifi, NFC, RFID y RF-Modulado directo. Entre los protocolos de comunicación inalámbrica de largo alcance (Wide Area Network - WAN) se encuentran comunicación celular 2G, 3G y 4G, y WAN de bajo consumo energético (Low Power Wide Area Network protocols), como LTE Cat-M, NB-IoT, Sigfox y LoRa-WAN. Cada protocolo de comunicación posee diferentes características, como ancho de banda y consumo de energía.

Dentro del espectro electromagnético hay bandas de frecuencias atribuidas a servicios específicos de radiocomunicación, como las designadas para las Telecomunicaciones Móviles Internacionales (IMT por sus siglas en inglés) y hay otras bandas clasificadas como espacio libre. Tal es el caso de la banda de 902 a 928 MHz, en la que se hallan Sigfox y LoRaWan.

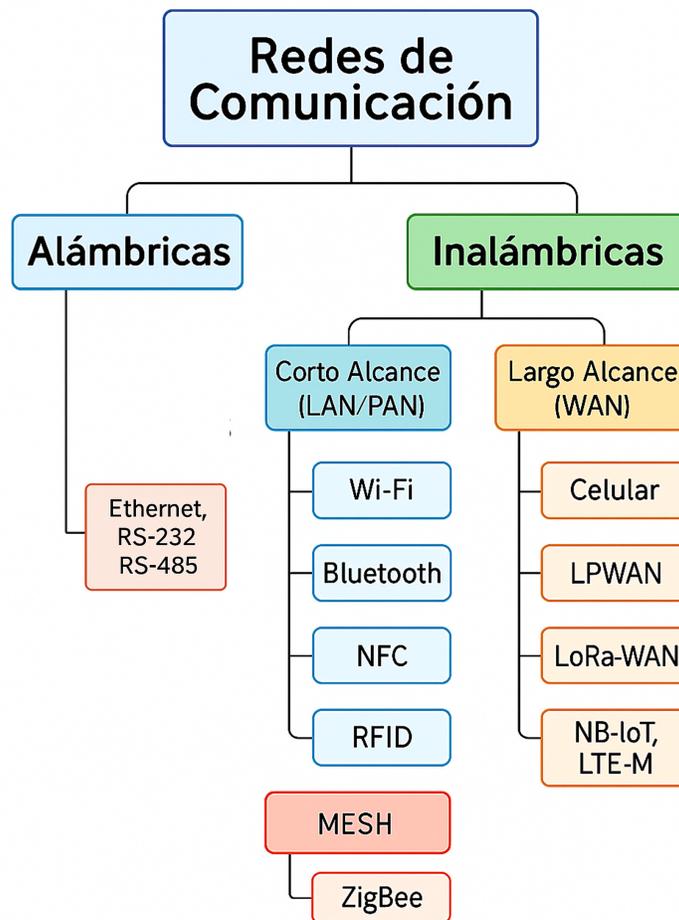


Figura 1.3: Redes de comunicación.

En la Tabla 1.1 se presenta una comparación entre cuatro tipos principales de redes utilizadas en el ámbito del Internet de las Cosas: redes celulares, redes de área local/personal (LAN/-PAN), redes de área amplia de baja potencia (LPWAN) y redes de tipo malla (MESH). Esta comparación permite visualizar de manera clara las ventajas y limitaciones de cada tipo de red en función de su aplicación.

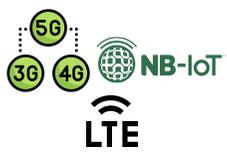
Tipo de red / Parámetro	Celular	Red de área local (LAN/PAN)	Red de área amplia de baja potencia (LPWAN)	Protocolo de malla (MESH)
Nombre asignado a la red	NB-IoT*, 3G, 4G, LTE-M	Bluetooth (BLE), WiFi	LoRaWAN, Sigfox	Zigbee, RFID
Tasa de transferencia de datos	~100 kbps - 100 mbps	~100 kbps - 100 mbps	~10 kbps	~250 kbps
Alcance	Largo	Corto	Largo	Corto
Duración de la batería (uso moderado)	Mediana 1 - 2 días	Mediana 5 - 7 días	Larga 6 meses - 1 año	Larga 1 - 2 años
Casos de uso	Tradicional Máquina a máquina (M2M)	Edificios/ Construcciones y en el hogar	Monitoreo ambiental (temperatura, humedad, calidad del aire)	Iluminación inteligente, sistemas de riego
Ejemplos				

Tabla 1.1: Comparación de cuatro tipos de redes de comunicación.

*NB-IoT, aunque se basa en infraestructura de red celular, se considera una tecnología LPWAN por su bajo consumo de energía y gran cobertura, pensada específicamente para aplicaciones IoT.

1.1.4 Software IoT

Además del hardware, un sistema IoT requiere software especializado que permita interpretar y visualizar los datos recopilados. Esto incluye el firmware de los microcontroladores, plataformas en la nube, protocolos de comunicación (MQTT, HTTP), servicios para almacenamiento y análisis, y herramientas de visualización de tableros. Estas soluciones permiten tomar decisiones informadas y gestionar dispositivos de manera remota.

En particular, los servicios en la nube desempeñan un papel muy importante en los sistemas IoT, ya que permiten escalar sistemas, almacenar grandes cantidades de datos, aplicar modelos de análisis en tiempo real y facilitar la interacción con dispositivos en distintos pun-

tos geográficos. Existen diversas plataformas IoT comerciales que ofrecen soluciones integrales para gestionar dispositivos, recolectar datos, analizarlos y generar alertas o automatizaciones. Entre las más utilizadas destacan las incluidas en la tabla 1.2 a continuación:

Plataforma	Descripción
	Amazon Web Services (AWS IoT Core): permite conectar miles de dispositivos de forma segura, manejar flujos de datos, ejecutar lógica de negocio y visualizar métricas.
	Microsoft Azure IoT Hub: brinda capacidades para integrar dispositivos IoT con otros servicios de Azure, como almacenamiento, bases de datos, inteligencia artificial y automatización.
	Google Cloud IoT Core: permite gestionar dispositivos a gran escala y aplicar herramientas de análisis como BigQuery, Cloud Functions y AI Platform.
	IBM Watson IoT: orientada al análisis de datos industriales con integración a servicios cognitivos de IBM.

Tabla 1.2: Principales plataformas de servicios IoT en la nube.

Losant y Ubidots son considerados **aplicativos** o plataformas de desarrollo IoT, orientadas a soluciones de prototipado y despliegue rápido que utilizan infraestructuras subyacentes de los ervcios en la nube incluidos en la tabla. Losant destaca por su enfoque en flujos visuales de lógica (workflow engine), soporte a múltiples protocolos (MQTT, HTTP, Webhooks) y dashboards interactivos. Por su parte, Ubidots ofrece una interfaz intuitiva para crear dashboards, herramientas de visualización en tiempo real y funcionalidades para la automatización de eventos, siendo especialmente popular en entornos educativos y pruebas piloto.

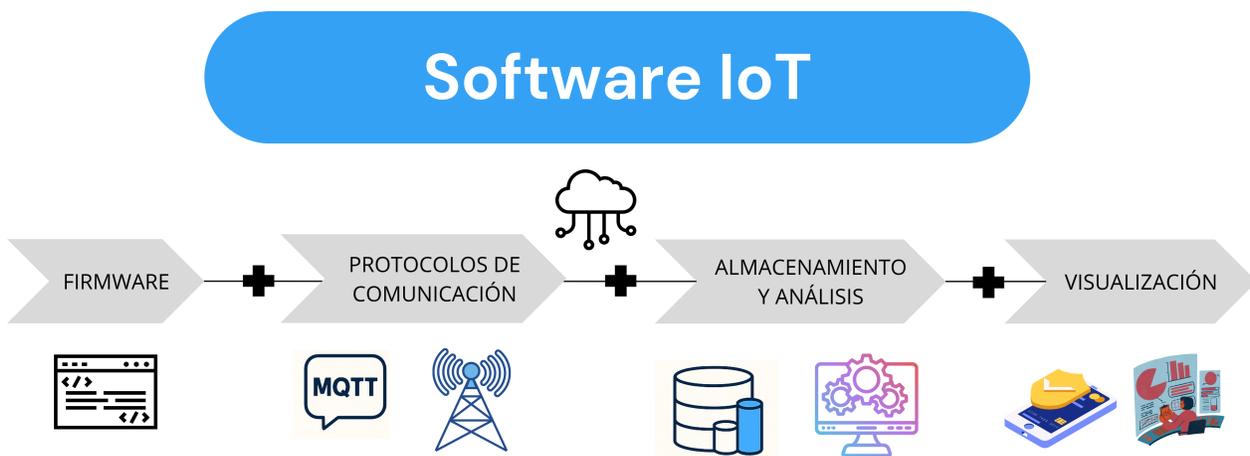


Figura 1.4: Software IoT.

2 PLANTEAMIENTO DEL PROBLEMA

La DGSGM, como lo ha establecido en su página de internet, promueve el desarrollo óptimo de la movilidad, a través de los servicios brindados por las Coordinaciones de Pumabús, Bicipuma, Estacionamientos Controlados y Archivo General. Busca ampliar la cobertura y mejorar la calidad de estos servicios con tecnología sustentable y de vanguardia. Prueba de ello es que se mantiene comunicación entre la brigada vial, operadores de pumabús, coordinadores, personal de seguridad, entre otros encargados de labores diversas, sumado a la utilización de herramientas de visualización de los recorridos de las unidades y registro histórico de parámetros de las mismas y reportes de eventualidades [5].

Ello es posible por el uso de sensores con los que están equipadas las unidades, como módulos GPS. Además, se cuenta con cámaras de videovigilancia a bordo para la seguridad de los usuarios y operadores. **Sin embargo, aunque se cuenta con módulos GPS y herramientas de visualización, el sistema no permite detectar las condiciones dentro de las unidades de forma actualizada, como la cantidad de usuarios transportados, la congestión en ciertas rutas o el uso efectivo de los autobuses según la demanda.**

Esta limitación impide llegar a una asignación óptima de recursos, como el número de autobuses en circulación, especialmente en horarios pico. Además, la falta de monitoreo en aspectos como ascensos y descensos impide tomar decisiones basadas en datos para la mejora del servicio y la planificación operativa.

El sistema de transporte actualiza su flota y depura o integra nuevas unidades con el tiempo, buscando que las adquisiciones sean las más adecuadas para las necesidades en el campus. Prevalecen algunos modelos de autobús con distintas características unos de los otros. En 2007 llegaron unidades Mercedes Benz Citaro, que cuentan con rampa para silla de ruedas, 33 asientos, capacidad total para, aproximadamente, 95 pasajeros y principalmente dan servicio a las rutas con base en el estadio; en 2019 se adquirieron las unidades Volkswagen 15.190 OD, que cuentan con 28 asientos, en una disposición que otorga capacidad de aproximadamente 90 pasajeros. Las unidades Pumabús están asignadas de la siguiente manera para las 13 rutas:

Ruta	1	2	3	4	5	6	7	8	9	10	11	12	13
Unidades	5	5	5	4	5	2	3	3	4	2	3	5	4
Modelos	2019	2019	2019	2019	2019	2019	2007	2007	2007	2019	2019	2019	2019
Tiempo por vuelta (min)	23-30	9-15	22-35	15-24	23-35	19-40	19-36	15-38	23-30	22-45	21-30	25-40	30-50

Tabla 2.1: Distribución de unidades Pumabús en 13 rutas.

Sin embargo, cuando se observa gran diferencia en la demanda de una ruta sobre otra, se toman unidades de la menos demandada y se destinan para servicio de la que más lo requiera. La DGSGM intenta volver al manejo de unidades asignadas de manera fija a una sola ruta para tener un mejor control y conocimiento de las rutas durante su monitoreo para facilitar la disposición de ese tipo de información a los usuarios.

En el último trimestre de 2023, el cálculo del total aproximado de pasajeros que trasladó el sistema Pumabús fue realizado por personal de la DGSGM, tomando en cuenta la capacidad de pasajeros de cada unidad, el número de unidades operantes y el número de vueltas diarias que da cada una. El resultado fue de alrededor de 6,670,000 a lo largo de los 520,326 *km* en 63,668 vueltas que se dieron en ese periodo. Ese es el dato más reciente que se tiene de manera oficial sobre los servicios en un periodo determinado de tiempo.

En realidad, no se lleva un conteo regular del número de pasajeros que asciende y desciende de las unidades. El único acercamiento directo a estas cifras ocurre de manera ocasional, cuando personal autorizado aborda una unidad para realizar un conteo manual con base en su observación, y a partir de ello estima la cantidad de usuarios transportados, considerando las unidades que se encuentren en circulación.

La DGSGM reconoce que, año con año, la matrícula de la UNAM aumenta y, por lo tanto, la afluencia de gente en los campus también se incrementa significativamente. Al día se han llegado a estimar 150,000 servicios diariamente (se considera un servicio cada que un usuario aborda un pumabús).

Un día normal, entre lunes y viernes las pruebas de arranque de las unidades, para verificar su estado adecuado para circular, comienzan a las 4:00 am. Anteriormente se tenía conocimiento empírico de que hay horas del día a las que la capacidad del sistema de transporte está rebasada completamente por la demanda. Una manera en la que la DGSGM realizaba la estimación de usuarios que debía transportar estaba basada en la capacidad de los vagones del metro y la frecuencia de llegada de los trenes a la terminal de metro Universidad. Este es el caso de las rutas 1, 2, 3, 4 y 5, que son las responsables de la tarea de desplazar a gran parte de esas personas.

Podría decirse que a los usuarios que identificaban algunas horas pico les resultaba más sencillo tomar una decisión acerca de su forma de traslado, pero, actualmente, resulta muy difícil identificar momentos particulares con demanda extraordinaria debida al aumento en la población en movimiento. Es por esto que el usuario puede no contar con los criterios suficientes para la elección que más le convenga. Tampoco hay una plataforma que muestre a la comunidad en general en qué punto de una ruta se halla una unidad.

Existen, entonces, elementos que ayudan a recabar información importante del sistema de movilidad, pero no hay un vínculo mediante el cual el usuario pueda acceder a ello de manera directa para aprovechar realmente las alternativas que están a su alcance.

Es en este punto cuando se vuelve evidente que una herramienta que provea de la información más actualizada y certera sobre el estado de ocupación y la ubicación de las unidades de Pumabús facilitaría mucho la dinámica del transporte para los usuarios al permitirles tomar una decisión razonada según sus necesidades. Esto traería consigo la posibilidad de una distribución de usuarios entre diferentes medios de transporte o caminos para peatones y un desahogo en los puntos de espera. Con el fin de abordar esta problemática, el proyecto se estructura en torno a un objetivo general y una serie de objetivos específicos que guían el desarrollo del sistema propuesto, en cada etapa, y se describen a continuación:

Objetivo general:

Desarrollar un prototipo de sistema capaz de identificar las estaciones de una ruta del servicio Pumabús, durante sus trayectos, realizar el conteo del número de pasajeros a bordo y transmitir esta información actualizada mediante tecnología IoT utilizando la red 0G-Sigfox, con una interfaz accesible para su consulta y análisis.

Objetivos específicos:

- Obtener las coordenadas geográficas (latitud y longitud) de la ubicación de una unidad Pumabús a lo largo de su recorrido por la ruta 3 en Ciudad Universitaria.
- Contar el número de pasajeros que ascienden y descienden de la unidad Pumabús en las estaciones que conforman la ruta 3.
- Procesar los datos recopilados para calcular el número total de pasajeros a bordo y asociar esta información con la estación correspondiente.
- Transmitir los datos procesados mediante la red 0G-Sigfox cada vez que la unidad abandone una estación.
- Desplegar la información en una plataforma IoT con visualizaciones diferenciadas según el tipo de usuario (pasajero, coordinador del servicio o desarrollador), actualizadas conforme se reciban nuevos datos.

Una herramienta útil para terminar de plantear la situación del problema y llegar a una solución viable es la metodología 5W + 2H, cuyo nombre se debe a las iniciales de una serie de palabras en inglés correspondientes a preguntas decisivas en el análisis de diferentes aspectos estratégicos para un proyecto. Esta herramienta ayuda a comprender y visualizar correcta y claramente el caso de estudio y el camino a trazar hacia la solución deseada. Esta herramienta está en forma de la tabla 2.2 a continuación:

Pregunta	PROYECTO: CONTEO DE USUARIOS Y MONITOREO DE UBICACIÓN DE UNIDADES PUMABÚS
What? ¿Qué oCuál?	Falta de un sistema que monitoree el estado de ocupación de las unidades de transporte Pumabús en Ciudad Universitaria. Falta de conocimiento de usuarios sobre la ubicación de unidades de Pumabús. Falta de una herramienta que registre el número de personas que suben o bajan de una unidad Pumabús y en qué estación lo hacen.
When? ¿Cuándo?	El problema se presenta principalmente durante las horas pico de los días más concurridos en Ciudad Universitaria: de lunes a viernes a primera hora (7:00 - 9:00), por la tarde, cuando se da el cambio de turno matutino- vespertino (13:00 – 15:00 H) y por la tarde-noche (19:00 – 21:00 H).
Why? ¿Por qué?	Actualmente las unidades de Pumabús no cuentan con un sistema que indique el nivel de ocupación de una unidad de Pumabús, lo que impide a los encargados distribuir los autobuses tener un panorama completo para poder decidir si enviar otra unidad a la ruta que se encuentra saturada y, menos, a qué estación enviarla para cubrir con la demanda. El usuario final (pasajero) no tiene acceso a la ubicación de las unidades, con lo que no puede tomar otras medidas para llegar a tiempo a su destino.
Where? ¿Dónde?	El problema ocurre en todas las unidades y rutas de Pumabús en Ciudad Universitaria, ya que ninguna de ellas cuenta aún con un sistema automatizado de detección y registro de ascensos y descensos de pasajeros. La ruta más conveniente para la planificación de un proyecto de solución IoT y realización de pruebas es la 3 porque tiene base de llegada en donde se desaloja la unidad y llega vacía a la base de salida, donde siempre el total de pasajeros deberá ser 0 y, de las rutas con esta característica, la 3 es la de menor afluencia.
Who? ¿Quién?	En la ruta 3, seleccionada por su conveniencia, los afectados más comunes son estudiantes de licenciatura, a partir de la mayoría de edad, que desean llegar a la Facultad de Ciencias Políticas y Sociales (FCPyS), miembros de posgrado en algún instituto de investigación o gente en general que asiste a la zona cultural, entre otros puntos del recorrido. Desconocer la ocupación de los autobuses que van a abordar les impide tomar la decisión de cambiar de ruta o elegir un medio de transporte alternativo y, por ende, pueden perder tiempo y llegar tarde a su destino.
How? ¿Cómo?	Se ha observado que el tiempo que pasa entre la llegada de un autobús y otro varía bastante. Llega a tardar desde 5 minutos hasta más de 20 minutos. Incluso cuando la espera no es tan prolongada, puede ya no haber espacio para más ascensos a la unidad.

Pregunta	PROYECTO: CONTEO DE USUARIOS Y MONITOREO DE UBICACIÓN DE UNIDADES PUMABÚS
How much? ¿Cuánto cuesta?	Recorridos de reconocimiento en unidades de Pumabús en CU muestran que el tiempo para completar un recorrido completo es de hasta 35 minutos. En el peor de los casos, si solo se cuenta con un autobús para una ruta y un usuario llega cuando la unidad acaba de partir de la estación, dicho usuario tendría que esperar a que la unidad regrese al completar un recorrido, es decir hasta 35 minutos para abordar la unidad.

Tabla 2.2: 5W 2H.

2.1 Representación gráfica del problema

Para complementar el análisis de la problemática identificada a través de la tabla de 5W 2H, a continuación en la figura 2.3, se muestra una representación visual del problema con situaciones específicas que podrían mejorar. Este recurso gráfico, también conocido por su nombre en inglés de Storyboard, presenta diferentes etapas o momentos clave que caracterizan la problemática, brindando una visión más dinámica y detallada de los desafíos enfrentados en el contexto de interés. A través de estos elementos, se busca ayudar a que la comprensión de la situación sea más completa.



(a) Usuarios desconocen ubicación del pumabús. (b) Llegan unidades cuando usuarios se han ido.

Figura 2.1: Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (a) y (b).

En la situación de las viñetas (a) y (b) un par de amigos tiene la posibilidad de llegar en Pumabús o caminando a su destino. Al no contar con criterios suficientes para elegir, optan por ir caminando al no ver unidades aproximarse. Poco tiempo después, no solamente llegan unidades, sino que lo hacen una muy seguida de la otra y esto pasa incluso en paraderos de una sola ruta. La situación indica una distribución inadecuada de las unidades.



(c) Arribo de unidad con capacidad al límite.

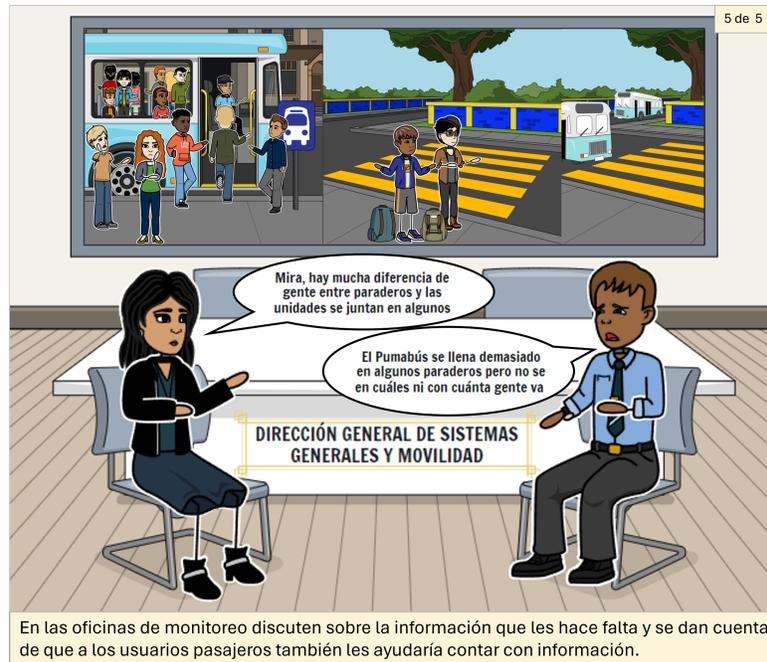


(d) Toma de decisiones confiadas al azar.

Figura 2.2: Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (c) y (d).

La viñeta (c) muestra a una usuaria que optó por aguardar pacientemente la llegada del pumabús durante un extenso periodo de tiempo, a pesar de tener alternativas para llegar a su destino. Sin embargo, cuando la unidad finalmente arribó, se encontraba tan abarrotada que ella y otros usuarios no pudieron subir, lo que resultó en una experiencia incómoda y una pérdida de tiempo para los afectados.

En el escenario de (d) se observa el momento en el que dos decisiones contrarias son tomadas, cuando, por una parte, dos amigos aguardan en un paradero contemplando únicamente la posibilidad de que el pumabús llegue en poco tiempo y sin considerar las implicaciones de un posible retraso prolongado, y por otro lado hay dos amigas que optan por comenzar a caminar como precaución ante cualquier demora en la llegada de unidades, pero también renuncian a la oportunidad de abordar el pumabús si llega en breve.



(e) Falta de información para mejor distribución de unidades.

Figura 2.3: Situaciones comunes por falta de información actualizada del estado de ocupación y ubicación de pumabús (e).

La situación representada en e) muestra a personal de la DGSGM en la oficina de monitoreo, observando las cámaras de algunos paraderos y teniendo conocimiento de que no hay una buena distribución de unidades. Sin embargo, poco pueden hacer con los usuarios que ya sufrieron los estragos de tomar la decisión de esperar o no esperar sin tener criterios suficientes, como pérdidas de tiempo, ni con los que están por vivir lo mismo.

Ahora que se ha identificado un espacio de oportunidad y se ha planteado un problema concreto, se abre la puerta para abordarlo mediante el diseño de una solución IoT. Esta tarea implica describir las distintas etapas que conforman el proceso de diseño, utilizando métodos estructurados que guíen cada paso. Es esencial comprender la razón fundamental de cada decisión en este proceso iterativo. Recordemos que el diseño es un proceso dinámico y adaptable, donde algo importante es poder regresar entre etapas según sea necesario.

Como establece Roger D. Ulrich en su libro *Diseño y desarrollo de productos*[8] y se ilustra en el diagrama de fases del diseño de la figura 2.4, el proceso de diseño puede representarse mediante diferentes enfoques, dependiendo del tipo de sistema o producto que se desarrolla. En este trabajo se opta por el proceso genérico de desarrollo del producto (a), que permite una estructura clara y secuencial, ya que el proyecto implica la construcción de un dispositivo físico con funcionalidades específicas, sin la complejidad de múltiples subsistemas altamente especializados ni la necesidad de múltiples ciclos de iteración como en los modelos (b) o (c).

La primera etapa en el proceso de diseño es la planeación, donde se evalúan las oportunidades para el proyecto, basadas en las necesidades y restricciones, recabando y tomando en cuenta información relevante sobre el mercado, usuarios y competencia.

Lo anterior da pie al desarrollo conceptual, en el que se busca la propuesta y posterior selección de las ideas más factibles y prometedoras para, luego, profundizar en los sistemas

y subsistemas que las conformen. Al definir y estructurar los sistemas está implicada una elección preliminar de componentes clave y la identificación de posibles proveedores.

Después, logrado un modelo funcional, se entra en el diseño de detalle, que involucra definir la geometría de las piezas, materiales y tolerancias. Con esto, es posible crear modelos y representaciones visuales para evaluar el diseño y, así, llegar a un prototipo y realizar pruebas y evaluaciones para volver a iterar si es necesario y rehacer las etapas que requieran modificaciones para obtener una mejor versión. La etapa de pruebas y validación aporta conocimiento sobre la funcionalidad general del resultado del proyecto y su rendimiento, lo que facilita lanzar e implementar el producto. Una vez implementado, se dará el servicio y mantenimiento requerido, lo que aportará información de su desempeño en operación.

Este proyecto tiene como objetivo presentar una propuesta de un modelo funcional y su alcance se limita a la presentación de dicha propuesta, sin alcanzar las etapas finales del proceso, que involucran adentrarse en la fase de producción. Así, el enfoque se centra en el diseño y desarrollo conceptual del modelo funcional, sin llegar necesariamente a la implementación y lo que esta involucra.

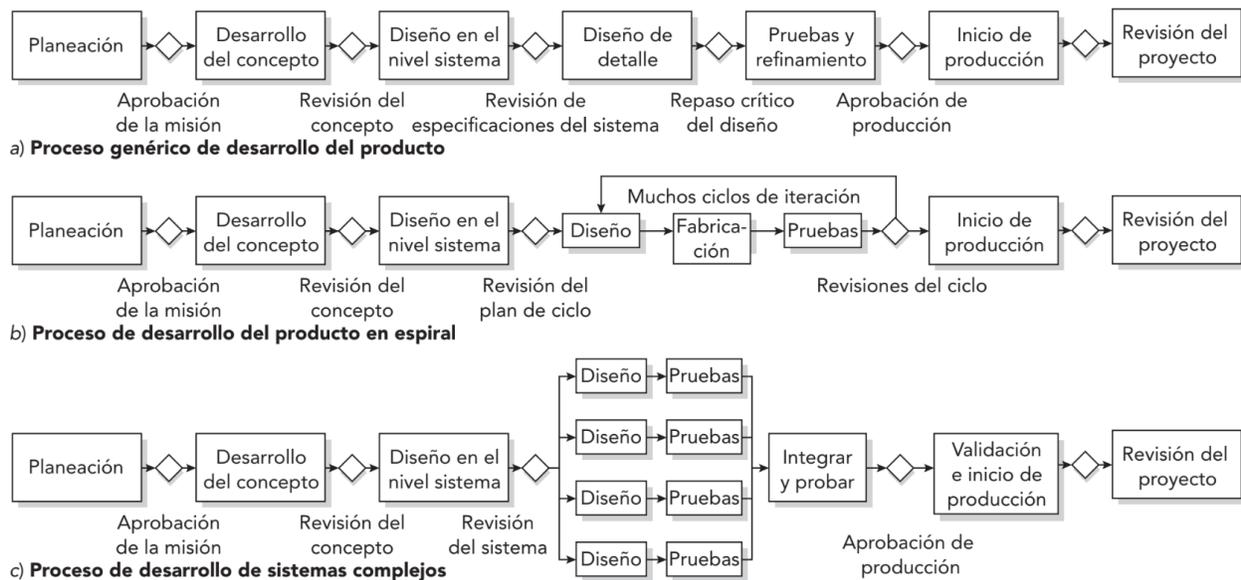


Figura 2.4: Diagrama de fases del diseño.

Este proyecto se guiará por un proceso simplificado de tres etapas principales: **planeación de diseño**, **desarrollo del concepto** y **pruebas de concepto**. Este esquema permite una estructura clara, coherente y manejable para avanzar progresivamente desde la identificación del problema hasta la validación de una solución funcional. Cada etapa se desarrolla con métodos específicos que ayudan a la toma de decisiones, con flexibilidad para retroceder y reiterar cuando sea necesario. A continuación, se ilustra esta secuencia de trabajo:

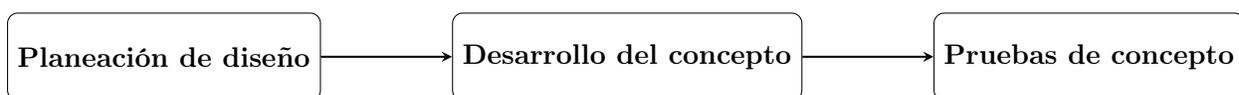


Figura 2.5: Diagrama de etapas que construyen el proceso de diseño en el presente proyecto.

3 ETAPA DE PLANEACIÓN DE DISEÑO

Con el propósito de comprender mejor las necesidades y requerimientos a satisfacer en el proyecto, se realizaron un total de tres entrevistas durante una serie de visitas a las oficinas de la coordinación y dirección del Pumabús, de la DGSGM, así como a las oficinas de monitoreo. En estas entrevistas participaron Agustín Oscar Valdés Barrón, responsable de la coordinación del Pumabús; Viridiana, encargada del área de monitoreo, que contestó junto con María Lourdes M. Aguilar Sánchez, quien proporcionó información valiosa acerca del funcionamiento del sistema de movilidad y su experiencia en la resolución de problemáticas relacionadas.

Durante estas visitas se recabó información sobre los desafíos operativos actuales, las necesidades específicas para planes futuros, así como las expectativas del personal involucrado. Adicionalmente, se entrevistó a un usuario frecuente del servicio, con el fin de conocer la perspectiva de quienes hacen uso cotidiano del transporte.

A continuación, en las tablas 3 y 4, se presenta la información recopilada durante las entrevistas.

3.1 Entrevistas con usuarios

Usuario coordinador del sistema Pumabús: Agustín Oscar Valdés Barrón.

Ocupación: Coordinador del sistema Pumabús en la DGSGM.

No.	Pregunta	Respuesta
1	¿Cuál es tu trabajo en el sistema de transporte en la UNAM? ¿Desde hace cuánto trabajas en este rubro?	Desde hace 20 años, coordinar todas las actividades relacionadas al servicio de transporte interno, incluyendo brigada vial, taller mecánico, centro de monitoreo, etc.
2	Dentro del área de movilidad y transporte en la UNAM, ¿Sabes cuántas personas trabajan? ¿Existe alguna clase de monitoreo de las unidades de Pumabús?	Al rededor de 250. Todas las unidades cuentan con seguimiento GPS y radio. De las unidades se tiene conocimiento: su velocidad, tiempo que pasa detenido, ubicación en tiempo real.
3	Dentro de CU, ¿Dirías que hay falta de movilidad? ¿Consideras que, actualmente, existe un problema de sobrecupo en el Pumabús? En caso de que sí, ¿Crees que existe forma de resolverlo? ¿Cuál?	Hay un gran sobrecupo, el sistema está rebasado y se va incrementando esto. No es tan fácil como decir “Vamos a meter más unidades” porque hay puntos críticos que se congestionan y resultaría contraproducente. El objetivo ahorita es reducir los tiempos de espera/aumentar la frecuencia de paso.

No.	Pregunta	Respuesta
4	¿Qué ves en esta situación? ¿Se ha incrementado el sobrecupo a lo largo del tiempo? Por ejemplo, después de la cuarentena con el regreso a clases presenciales. ¿Consideras que se trata de un problema urgente?	Que se tienen fortalezas y debilidades. Hay también factores externos que afectan mucho. CU es un distribuidor vial que se congestiona. Se incrementó de manera importante. Según el encargado de vialidad, de que se regresó de pandemia, los vehículos incrementaron en un 50 % y lo mismo con los peatones. Previamente se tenían identificadas ciertas horas pico que presentaban mucho movimiento de usuarios, ahora ya no son ciertas horas, es más constante.
5	¿Cómo te sientes acerca del Pumabús? ¿Te parece un transporte seguro y efectivo? ¿Dirías que los usuarios se encuentran satisfechos con el servicio brindado?	La cantidad de usuarios rebasa la capacidad. Nos preocupa bastante la seguridad, tengo una hija que está por entrar a la universidad. Tenemos protocolos para cada situación que se presente y se requiere actuar con prontitud, las unidades se pueden bloquear de manera remota desde el centro de control, se manda personal a resolver la situación, para casos de acoso, de robo, de amenaza, se lleva un registro de incidencias, se dan cursos de todo tipo al personal.
6	¿Qué es lo que haces en el área del Pumabús? ¿Has buscado alguna manera de disminuir o eliminar el problema del sobrecupo?	Se ha acercado a pedir información gente de ciencias, de ingeniería, de diversas áreas, se han intentado hacer varios proyectos de su parte y nuestra parte: aplicaciones, sistemas con pantallas en la parada donde se ve en cuanto tiempo llega una ruta. A veces lo que pasa es que alguien de la administración cambia y desecha un Proyecto.
7	¿Cómo operas actualmente en el sistema de movilidad/Pumabús? Tenemos planeado contabilizar las personas que suben y bajan de las unidades, ¿crees que eso podría ser útil para el sistema?	Casi siempre estamos en campo, yendo a atender los reportes que se nos hacen por radio, hemos realizado conteo de personas directamente con personal que se queda en la unidad para llevar un conteo. Claro, y cuentan con mi apoyo, con nuestra mejor disposición para conseguirlo.
8	¿Qué es lo más complicado en esta área? ¿Cuáles son las situaciones que les causan más complicaciones?	Mantener el servicio eficiente cuando hay factores externos que afectan, eventos, marchas, que cada elemento del personal es diferente y también tienen sindicatos, con los que llega a haber problemas.

No.	Pregunta	Respuesta
9	Si pudieras enlistar las cosas más difíciles de manejar y monitorear el sistema de transporte en CU, ¿cuáles dirías que son las 5 mayores?	<p>Que cada chofer es diferente y se apoya de su sindicato cuando no cumple con el comportamiento deseado. Serían:</p> <ul style="list-style-type: none"> -Ausentismo de operadores, faltas del personal. -El volumen de gente a transportar nos rebasa. A un mismo tiempo salen del turno nocturno los alumnos de una facultad. -Vandalismo, lastiman las cámaras. -Hay que dar tiempos de comida, baño, lavar unidades, cargar combustible, dar mantenimiento. -CU se congestiona, avenidas y calles que dan a CU sin semáforos coordinados suficientes, incremento gradual de población que entra y se mueve al interior. La unidad de apoyo para personas con discapacidad también está rebasada.
10	De manera general, ¿cuáles son las 5 cosas que te gustaría mejorar?	<ul style="list-style-type: none"> -La frecuencia de paso de las unidades. -La educación vial también por parte de ciclistas, patinadores, motociclistas, paseantes. -Coordinación de semáforos de acuerdo con cómo se mueve la comunidad. -Mayor seguridad al interior de las unidades El sobre cupo en general.

Tabla 3.1: Entrevista Coordinador de Pumabús

Usuario pasajero: Abraham L.
 Ocupación: Estudiante de Ingeniería Civil.

No.	Pregunta	Respuesta
1	¿Cuál es el rol que desempeña dentro del sistema de Pumabús de CU?	Usuario.
2	¿Qué nos puede decir acerca de problemas o imprevistos que pueden llegar a presentarse dentro del sistema de Pumabús?	El pumabús se tarda mucho en salir de la base o en pasar por la estación. El cupo se ve limitado debido a que los pasajeros no se recorren. Recientemente, alguien se desmayó dentro de la unidad de Pumabús y el operador no sabía qué hacer. Los pasajeros tuvieron que llamar a una ambulancia.

No.	Pregunta	Respuesta
3	<p>¿Qué piensa usted sobre estos problemas en el sistema de Pumabús? ¿Se pueden evitar?</p>	<p>Pienso que la mayoría de los problemas son generados por los usuarios. El tiempo de espera podría reducirse si hubiera más pumabuses por ruta, sin embargo, creo que esto se trata de un problema menor para el tipo de servicio que se brinda (es gratis).</p>
4	<p>¿Qué observa usted cuando ocurren problemas en el sistema de Pumabús? ¿Cómo se comportan los usuarios? ¿Cómo se comportan los operadores?</p>	<p>He observado que existen protocolos a seguir cuando se presentan situaciones específicas. Por ejemplo, el día del sismo de 19 de septiembre de 2022, los operadores recibieron instrucciones por radio de detener las unidades de Pumabús en cuanto empezó a sonar la alerta sísmica. Sin embargo, para otro tipo de situaciones, los operadores no están bien capacitados, como en el caso del desmayo de un usuario dentro de la unidad, donde fueron otros usuarios quienes llamaron una ambulancia y no el chofer.</p>
5	<p>¿Cómo se siente o cuál es su opinión en cuanto a los comportamientos de usuario y operadores cuando ocurren problemas en el sistema de Pumabús?</p>	<p>Siento que aun con los problemas que presenta el sistema de Pumabús, sigue siendo un mejor servicio de transporte que el colectivo urbano de la Ciudad de México. Hay más gente imprudente en los camiones y peceros de la ciudad que en el Pumabús. Por otra parte, el enojo de los usuarios del Pumabús cuando éstos se retrasan está justificado, aunque también hay que entender que en CU somos demasiadas personas y se hace lo mejor que se puede con lo que se tiene. Como ya mencioné, tenemos que recordar que el servicio es completamente gratuito.</p>
6	<p>Cuando se presentan problemas en el sistema de Pumabús, ¿qué es lo que hace? ¿Existe algún protocolo?</p>	<p>Si el pumabús va muy lleno o ya tardó demasiado, me voy caminando hacia mi facultad (Facultad de Ingeniería) o a donde tenga que llegar, a menos que tenga un examen o compromiso importante, en esos casos sí opto por utilizar otro medio de transporte como la bicipuma o taxi.</p>

No.	Pregunta	Respuesta
7	Tenemos conocimiento de que la frecuencia con la que pasan los pumabuses de una cierta ruta no es siempre la misma, ¿esto es correcto? ¿A qué cree que se deba esto?	Es correcto. Creo que se debe a que la demanda no es la misma durante todo el día del servicio. En ciertas horas hay más gente que utiliza el Pumabús que en otras. Sin embargo, creo que existe una cierta rutina diaria de flujo de gente; los encargados de la distribución de pumabuses podrían poner más de su parte y contemplar esto para evitar las grandes aglomeraciones de gente en las paradas de Pumabús.
8	¿Qué es lo que más le molesta cuando se presenta algún problema en el sistema de Pumabús?	Llegar tarde a mis clases por imprevistos que están fuera de mi control. Aun cuando llegue al campus a la misma hora todos los días, al final todo depende de si el pumabús pasa rápido o no y esto ha provocado que, en ocasiones, llegue temprano a mis clases y otras no
9	¿Cuáles son las 5 cosas más difíciles que debe enfrentar en cuanto a la frecuencia con la que pasa el pumabús, la distribución de pumabuses por ruta o algún otro problema relacionado?	1) Llegar tarde a mis clases. 2) Tener que esperar mucho tiempo una unidad de Pumabús. 3) Que el pumabús vaya demasiado lleno (que en parte es derivado de que el pumabús tarde en pasar por la estación, ya que se va juntando más y más gente conforme pasa el tiempo). 4) Desesperación por llegar a mi destino. 5) La frecuencia con la que pasan los pumabuses de algunas rutas es muy baja comparadas con otras.
10	¿Cuáles son las 5 cosas que le gustaría mejorar respecto a la frecuencia con la que pasa el pumabús, la distribución de pumabuses por ruta o algún otro problema relacionado?	1) Que, en todas las rutas, la salida de los pumabuses esté regida por la demanda observada y no que salgan cada cierto tiempo. 2) Mejor identificación en los pumabuses, porque en ocasiones veo a lo lejos que viene un pumabús, me quedo esperando y resulta que no es de la ruta que quería. Muchos pumabuses no tienen encendido el cartel de identificación qué ruta son. 3) Yendo un poco más lejos, me gustaría que en las paradas se indique el tiempo estimado de llegada del pumabús en una pantalla (ejemplo, el metro de Japón). 4) Que exista un cartel dentro de cada pumabús, donde se muestren todas las estaciones para que las personas sepan qué estación sigue (como en el metro), específicamente para los nuevos usuarios. 5) Señalización para saber dónde formarte para esperar el pumabús (igualmente como en el metro), ya que la gente se mete a la fuerza.

Tabla 3.2: Entrevista Pasajero.

Usuario coordinador del sistema Pumabús: María Lourdes M. Aguilar Sánchez y Viridiana.

Ocupación: Personal de Monitoreo y de Operación del sistema Pumabús.

No.	Pregunta	Respuesta
1	¿Cuál es el rol que desempeña dentro del sistema de Pumabús de CU?	Jefa de Departamento de Operación de Pumabús. Oficina de Monitoreo
2	¿Qué nos puede decir acerca de problemas o imprevistos que pueden llegar a presentarse dentro del sistema de Pumabús?	Anteriormente había casos muy frecuentes de robo y acoso dentro de la unidad. Ahora ya es menos con los protocolos implementados. En cuanto a demanda y saturación, antes podíamos distinguir horas pico en distintos momentos del día. Ahora es como si todo el día fuera hora pico.
3	¿Qué piensa usted sobre estos problemas en el sistema de Pumabús? ¿Se pueden evitar?	Constantemente estamos trabajando en resolver eventualidades, modificamos e implementamos nuevas rutas, distribuimos las unidades conforme a las necesidades. Para tener un mejor control estamos buscando tener unidades asignadas de manera fija para cada ruta. Estamos volviendo a dar seguimiento a herramientas como Pumamovil para atender mejor a los usuarios y actualizamos plataformas de servicio.
4	¿Qué observa usted cuando ocurren problemas en el sistema de Pumabús? ¿Cómo se comportan los usuarios? ¿Cómo se comportan los operadores?	Contamos con protocolos para diversas situaciones, prevemos marchas y sus implicaciones, los choferes se comunican por radio y limitamos los recorridos para resguardar las unidades y al personal, porque ya han tratado de tomar las unidades los manifestantes. Si un empleado tiene, por ejemplo, problemas de salud o accidente. Se ausentará y buscaremos que los cubra otro trabajador.
5	¿Cómo se siente o cuál es su opinión en cuanto a los comportamientos de usuario y operadores cuando ocurren problemas en el sistema de Pumabús?	Es realmente complicado que siempre hayan condiciones idóneas para el funcionamiento óptimo de todo el sistema, porque hay muchos factores como los que ya mencionamos, aunados a los servicios requeridos por las unidades, como carga de combustible, lavado, reparación, etc. Que todo el personal esté presente, sano, capacitado, es algo que hay que procurar todos los días.

No.	Pregunta	Respuesta
6	Cuando se presentan problemas en el sistema de Pumabús, ¿qué es lo que hace? ¿Existe algún protocolo?	Si una ruta tenía mucha más demanda que otra, se buscaba tomar una unidad de las rutas menos demandadas, pero ahora se buscará restablecerlas unidades asignadas exclusivas a cada ruta para llevar mejor control. En situaciones de robo o acoso, se bloquea la unidad y se cierran las puertas mientras llega vigilancia.
7	Tenemos conocimiento de que la frecuencia con la que pasan los pumabuses de una cierta ruta no es siempre la misma, ¿esto es correcto? ¿A qué cree que se deba esto?	Así es. A veces se descansan unidades cuando no hay tanta demanda para poder hacerles servicio. Cuando una ruta cedía una unidad a otra, cambiaba la frecuencia de paso temporalmente en ambas rutas, disminuyendo en la que cedió y aumentando en la que recibió.
8	¿Qué es lo que más le molesta cuando se presenta algún problema en el sistema de Pumabús?	Que empieza a haber mucho movimiento, muchos reportes y por más que se hagan movimientos, ya todo el tiempo es hora pico y no se da abasto. Cuando hay cambio de unidades a otra ruta, la gente no sabe de qué ruta será una unidad o con qué frecuencia pasará la ruta que espera.
9	¿Cómo funcionan Pumamovil y las plataformas mencionadas?	PumaMóvil tiene la finalidad de encontrar la ruta más eficiente para los traslados de la comunidad universitaria, así como brindar información sobre eventos, actividades y exposiciones que puedes visitar a su paso. Estuvo deshabilitada un tiempo y recientemente, de nuevo estuvo en periodo de pruebas. Proporciona la ubicación de los pumabuses en tiempo real, disponibilidad de Bicipumas y viajes compartidos con ayuda de Waze. La plataforma GPS & Cloud it, independiente de Pumamovil, nos permite visualizar en un mapa la ubicación de las unidades e información de las mismas. Con ella llevamos un registro histórico de los datos recabados.

Tabla 3.3: Entrevista personal de monitoreo.

Las entrevistas realizadas a los distintos tipos de usuarios (pasajeros y coordinadores del sistema Pumabús) reflejan una preocupación común por el **sobrecupo**, la **frecuencia irregular de paso de las unidades** y la **saturación generalizada del servicio**.

Coordinadores del sistema Pumabús: el coordinador Agustín Valdés y el personal operativo señalaron que el sistema se encuentra rebasado por la demanda actual, lo cual ha

sido agravado desde el regreso a clases presenciales tras la pandemia. Esto se manifiesta en congestiones constantes, incluso fuera de los horarios pico previamente detectados.

Desde la perspectiva operativa, se destaca la existencia de protocolos de seguridad y monitoreo con GPS y radio, así como el esfuerzo continuo por implementar mejoras, como nuevas rutas, plataformas digitales y coordinación en semáforos. Sin embargo, se enfrentan congestiones viales dentro y fuera de CU, ausentismo del personal, y factores externos como marchas y eventos masivos.

Pasajeros: por su parte, los usuarios pasajeros coinciden en que el servicio, a pesar de ser gratuito, presenta **retrasos significativos, aglomeraciones en estaciones y unidades**, y en algunos casos una **falta de capacitación del personal ante emergencias**. También expresaron ideas claras sobre lo que debería mejorarse: mayor frecuencia en función de la demanda, mejor identificación de rutas, señalización clara en paradas y unidades, y mayor difusión de protocolos ante eventualidades.

En conjunto, las entrevistas evidencian una carga operativa considerable y una intención institucional de mejorar el servicio, frente a una demanda persistente por mayor eficiencia, seguridad y accesibilidad en el sistema.

3.2 Identificación de requerimientos

Durante el proceso de investigación acerca de la situación del transporte de pasajeros y del funcionamiento general del sistema, en el que se realizaron entrevistas a los usuarios, con el objetivo de comprender sus necesidades y expectativas, se llevó a cabo un análisis de las respuestas de los entrevistados que permitió identificar una serie de requerimientos de usuario que servirían como base para el diseño y desarrollo del sistema.

Estos requerimientos, que abarcan desde funcionalidades específicas hasta aspectos de usabilidad y rendimiento, fueron organizados en una tabla (tabla 3.4) para garantizar que el sistema satisfaga las necesidades reales de los usuarios.

Es importante explicar a qué se refieren los conceptos a los que se hará referencia en lo posterior:

Necesidad: Condición problemática o carencia experimentada por los usuarios, que justifica el desarrollo de una solución.

Deseo: Aspiración o preferencia expresada por los usuarios, que complementa la funcionalidad básica, pero no es crítica para su operación.

Expectativa: Las anticipaciones o esperanzas de los usuarios sobre el desempeño o resultados del sistema, que pueden influir en su percepción general y satisfacción.

Requerimiento: Las funcionalidades, características o condiciones que son esenciales para el correcto funcionamiento del sistema y cumplimiento de las necesidades de los usuarios.

Especificación: La descripción detallada y precisa de algún requerimiento, expresada en términos técnicos y funcionales que guían el diseño e implementación del sistema.

Con estas definiciones en mente, la tabla de requerimientos permitió crear una nueva tabla (Tabla 7) que contiene especificaciones y resulta en una representación concreta y traducción de los requerimientos en términos de características técnicas y funcionales que se espera que el sistema debe cumplir. De esta forma sirve como guía en el proceso de diseño e implementación, pues relaciona cada especificación con los requerimientos que satisface.

No.	Requerimiento	Usuario	Origen
1	Contar con información suficiente para distribuir unidades de Pumabús de manera eficiente.	Coordinador	Necesidad
2	Saber si el pumabús va muy lleno para decidir si esperar o no.	Pasajero	Necesidad
3	Saber si el pumabús tardará mucho para decidir si esperar o no.	Pasajero	Necesidad
4	No llegar tarde por retrasos imprevistos del Pumabús.	Pasajero	Necesidad
5	No tener que esperar mucho tiempo una unidad de Pumabús.	Pasajero	Necesidad
6	Que no se acumule demasiada gente por los tiempos de espera prolongados.	Pasajero	Necesidad
7	No sentir desesperación por la incertidumbre de los tiempos de espera.	Pasajero	Necesidad
8	Que en todas las rutas la salida de los pumabuses esté regida por la demanda observada y no que salgan cada cierto tiempo.	Pasajero	Necesidad
9	Mejor identificación de los pumabuses. Que el saber a qué ruta corresponde una unidad no dependa de que la unidad esté tan cerca como para ver el identificador, pues muchos pumabuses no tienen identificador claramente visible.	Pasajero	Necesidad
10	Que en las paradas se indique el tiempo estimado de llegada del pumabús en una pantalla.	Pasajero	Necesidad
11	Que exista un cartel dentro de cada pumabús, donde se muestren todas las estaciones para que las personas sepan qué estación sigue, específicamente para los nuevos usuarios.	Pasajero	Necesidad

No.	Requerimiento	Usuario	Origen
12	Señalización para saber dónde formarte para esperar el pumabús (igualmente como en el metro), ya que la gente se mete a la fuerza.	Pasajero	Necesidad
13	Mantener el servicio eficiente cuando factores externos que afectan, como eventos, marchas, problemas con sindicatos, ausentismo de operadores, faltas del personal.	Coordinador	Expectativa
14	Reducir los tiempos de espera/aumentar la frecuencia de paso.	Coordinador	Expectativa
15	Que un proyecto no se deseche por cambio de administración.	Coordinador	Expectativa
16	Ya no realizar manualmente el conteo de personas en las unidades de Pumabús.	Coordinador	Expectativa
17	Poder consultar de qué estación parte el pumabús y hacia cuál se dirige.	Desarrollador	Deseo
18	Ubicación del dispositivo en un lugar cerca de la entrada del pumabús donde no estorbe ni pueda ser arrollado por los pasajeros.	Desarrollador	Deseo
19	Poder consultar de manera remota, actualizada y a través de dispositivos móviles la información recabada por el dispositivo y procesada tanto en él como en la plataforma.	Desarrollador	Deseo
20	Dispositivo ligero y compacto con componentes fijados en su interior para evitar desconexiones por el movimiento propio del pumabús.	Desarrollador	Deseo
21	Batería de larga duración y recarga rápida.	Desarrollador	Deseo
22	Contar con elementos gráficos que faciliten el análisis de los datos con los que se cuente, tales como horas pico y estaciones más concurridas.	Desarrollador	Deseo

No.	Requerimiento	Usuario	Origen
23	Poder conocer la ubicación del pumabús constantemente.	Desarrollador	Deseo
24	Tener un registro histórico del número de personas que ascienden y descienden del pumabús por estación en el tiempo.	Desarrollador	Deseo
25	Llevar el conteo de las personas a bordo del pumabús durante todo el recorrido.	Desarrollador	Deseo
26	Que el dispositivo esté ubicado en un espacio en el que los usuarios no tengan contacto físico con él involuntariamente.	Caso de uso/ entorno	Expectativa
27	Que el dispositivo esté ubicado en un espacio en el que no interfiera con la funcionalidad de los barandales o apoyos para usuarios.	Caso de uso/ entorno	Expectativa
28	Que el dispositivo no ocupe más espacio del requerido por sus componentes.	Caso de uso/ entorno	Expectativa
29	Que sea posible ajustar la posición de ciertos componentes para que las mediciones sean las mejores posibles.	Caso de uso/ entorno	Expectativa
30	Que las antena de recepción y envío de mensajes tengan una salida o la carcasa del dispositivo sea de un material que permita el paso de las señales.	Caso de uso/ entorno	Expectativa

Tabla 3.4: Tabla de requerimientos.

Con base en el enfoque propuesto por Ulrich y Eppinger [8], se realizó una interpretación de las necesidades identificadas en entrevistas y observación directa, las cuales fueron jerarquizadas de acuerdo con su impacto en la experiencia del usuario y viabilidad técnica. Si bien no se aplicó una ponderación cuantitativa formal, sí se consideró la importancia relativa de cada necesidad para transformarla en un requerimiento de diseño claro.

De los requerimientos recopilados, se identificaron los que serán directamente atendidos por el sistema propuesto. Entre ellos, se priorizaron los relacionados con la recopilación de datos actualizada sobre la ocupación de las unidades (**requerimientos 1, 2, 4, 5, 6, 14, 16, 22, 24 y 25**).

También se atienden, de forma adicional, requerimientos relacionados con la geolocaliza-

ción y visualización remota de datos (**requerimientos 3, 17, 19, 23 y 24**). El uso de un módulo GPS y su integración con una plataforma IoT permite conocer la ubicación actual de las unidades, visualizar rutas y mantener un registro histórico de los trayectos.

Finalmente, se atienden requerimientos vinculados con el diseño físico y operativo del dispositivo (requerimientos 18, 20, 21, 26, 27, 28, 29 y 30), garantizando una instalación segura, discreta y funcional, sin interferir con el espacio del usuario ni con la operación del pumabús. Se consideraron aspectos como el tamaño compacto, el consumo energético optimizado, el uso de baterías recargables, la fijación adecuada de componentes y el uso de materiales que no bloqueen las señales de transmisión, para asegurar el desempeño eficiente y sostenido del sistema.

Es importante señalar que no todos los requerimientos recopilados serán atendidos en esta etapa del proyecto. La razón principal es que algunos de ellos implican desarrollos complementarios o de mayor complejidad técnica, logística o presupuestal que escapan al alcance actual. Por ejemplo, aspectos como la señalización física en paradas (requerimiento 12), la implementación de pantallas informativas (requerimiento 10), o la mejora de identificación visual de unidades (requerimiento 9) requieren coordinación directa con las autoridades operativas del sistema de transporte y modificaciones a la infraestructura física.

Los requerimientos seleccionados para ser atendidos fueron priorizados con base en su posible impacto en la eficiencia operativa y en la experiencia del usuario, así como en su alineación con los objetivos del proyecto. Se espera que este primer paso sienta las bases para futuras ampliaciones e integraciones del sistema.

3.3 Especificaciones

No.	Especificación	Métrica	Unidades	Valor marginal	Valor ideal	Requerimiento
1	Geolocalización	Coordenadas (Latitud, Longitud)	Decimales	0.00001	0.000001	1, 3, 4, 5, 9, 10, 17, 23
2	Conteo de personas	Exactitud Precisión	Porcentaje	70 %	95 %	1, 2, 5, 8, 16, 25
3	Envío de datos al backend del sistema	Error al enviar datos	Porcentaje	20 %	5 %	1, 2, 3, 4, 5, 6, 7, 8, 9, 17, 18, 23, 24, 25
4	Consulta de información actualizada	Retraso en la información	Estaciones	1	0	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 16, 17, 19, 22, 23, 24
5	Resistente a golpes y movimiento	Presencia de abolladuras y/o desconexiones	Binario (Sí / No)	Presenta	No presenta	20

No.	Especificación	Métrica	Unidades	Valor marginal	Valor ideal	Requerimiento
6	Tamaño	Dimensiones (Largo x Ancho x Alto)	cm	30x30x30	20x10x10	18, 20, 26, 27, 28, 29
7	Duración de la batería en operación	Tiempo	Horas	2	16	21
8	Ligero	Masa	kg	3	1	20
9	Indicador de batería	Resolución	Porcentaje de incremento (%)	25 %	10 %	21
10	Tiempo de recarga de la batería	Tiempo	horas	4	2	21
11	Intuitivo	Tiempo del usuario para aprender a manejarlo	horas	5	0.5	11
12	Configuración remota	Distancia mínima de configuración	km	1	∞	1, 8, 13, 16
13	Notificar errores al operar	Notificación	Binario (Sí / No)	No	Sí	13, 16, 19
14	Registro histórico	Tiempo de almacenamiento de datos	Días	7	30	1, 8, 14, 22, 24, 25
15	Ubicación	Distancia a los pasajeros	cm	10	30	18, 26, 27
16	Gráficas de visualización	Frecuencia	Intervalo e tiempo	Hora, día o estación	Todas, según se elija	1, 8, 14, 19, 22, 24, 25
17	Señal sin obstrucción	Material	Material	MDF o foam-board	Plástico	30

Tabla 3.5: Tabla de especificaciones.

3.4 Evaluación comparativa en el mercado

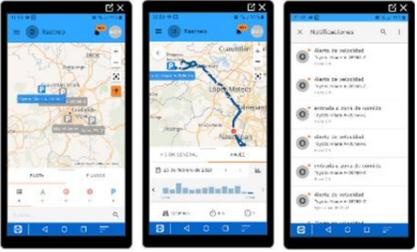
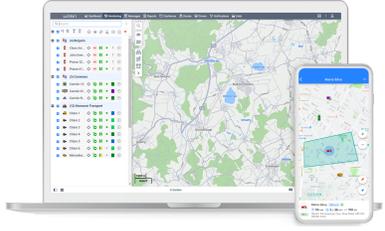
Como siguiente paso en el proceso de diseño del presente proyecto, se empleará una técnica conocida como ‘Análisis Comparativo’ o ‘Evaluación Comparativa’ de lo que se puede hallar en el mercado actual y también comúnmente llamada ‘Benchmarking’, en inglés. Esta práctica consiste en investigar, estudiar y comparar el desempeño, procesos o productos de organizaciones, empresas o entidades de la misma área de mercado o una similar a la que pertenece el proyecto en cuestión.

El objetivo principal de este análisis es descubrir, primero, si ya existen soluciones que realicen o se aproximen a lo que se busca lograr con la propuesta de proyecto y, de ser el caso, identificar las mejores prácticas, tendencias y áreas de mejora en el sector. A través del análisis comparativo, se puede obtener información valiosa que permitirá optimizar procesos, productos y estrategias, y así asegurar la viabilidad y la conclusión del proceso de diseño de un producto.

En este caso particular, el análisis se enfocará en tres ámbitos clave: productos de software, hardware y redes de comunicación. Esta revisión permitirá conocer cómo distintos enfoques tecnológicos abordan retos similares y qué elementos pueden ser tomados como referencia o adaptados para el desarrollo del presente proyecto.

A través del benchmarking, se puede obtener información valiosa sobre cómo empresas, organizaciones o entidades abordan problemas similares, qué estrategias utilizan y cómo logran resultados satisfactorios.

3.4.1 Evaluación comparativa de soluciones - Software

Nombre de la plataforma	Imagen de referencia	Enlace de información	Precio
GPS & cloud it		https://www.gpscloudit.com/	No disponible
Samsara		https://www.samsara.com/mx/products/platform#impact0	\$80 - 120 USD al mes por vehículo
Wialon		https://wialon.com/es	No disponible
Pumamovil		https://pumamovil.c3.unam.mx/index.html	No disponible
GPScontrol		https://gpscontrol.com.mx/beneficios-gpscontrol/	\$80 - 120 USD al mes por vehículo

Nombre de la plataforma	Descripción	Especificaciones
GPS & cloud it	División de Localización Vehicular y Logística de Locant Innovations enfocada en rastreo de Vehículos automotores, personal en campo, carga en tránsito, cadena de frío, etc.	<ul style="list-style-type: none"> -Precisión (ubicación): hasta 5 m - Actualizaciones: 15 s (configurable) -Acceso multiplataforma: versión web y app móvil (Android / iOS) -Historial de rutas: hasta 12 meses -Protocolos de comunicación: TCP/IP, UDP, HTTP -Visualización: tablas, gráficas, y alertas personalizables en dashboard.
Samsara	Plataforma en la nube de monitoreo de flotas con reportes en dashboards. Cuenta con alertas de audio en la cabina.	<ul style="list-style-type: none"> -Cumple con normativa ELD (Electronic Logging Devices) para registro electrónico de horas de servicio -Precisión GPS: > 2.5 metros -Red celular 4G LTE -Visualización de datos con gráficas -Compatibilidad con APIs para integración de datos
Wialon	Solución multifuncional de seguimiento de flotas por GPS. Fácil de integrar y adaptar a todo tipo de proyectos, incluyendo complejos controles de procesos en diferentes industrias.	<ul style="list-style-type: none"> -Compatible con más de 2,500 modelos de dispositivos GPS -Visualización de sensores (temperatura, peso, inclinación, mecanismos) -Plataforma basada en la nube con acceso 24/7 -Intervalo de actualización: desde 5s -Exportación de datos en formatos CSV, XLSX, PDF -Disponibilidad de marca blanca para personalización -APIs REST para integración con sistemas externos
Pumamovil	App movil que brinda información en tiempo real sobre Bicipuma y Puma-bús, la ruta más eficiente para los traslados de la comunidad universitaria y muestra eventos, actividades y exposiciones.	<ul style="list-style-type: none"> -Visualización de rutas georreferenciadas del servicio Pumabús -Información actualizada sobre disponibilidad de bicicletas (Bicipuma) por estación -Interfaz gráfica optimizada para dispositivos móviles Android -Acceso gratuito sin registro previo -Actualización de datos de vehículos y bicicletas cada 60 s (estimado) -Tecnología base: WebApp desarrollada por el C3, UNAM
GPScontrol	Plataforma web y app móvil para rastreo GPS de unidades con 3G o 4G, involucrando geocercas.	<ul style="list-style-type: none"> -SIM multicarrier. (TELCEL, TELEFÓNICA, AT&T) -Reporte a plataforma, de 3 a 8 veces por minuto -Batería de respaldo por 8 horas -Reportes históricos de ubicación, rutas y alerta de 6 meses -Geocercas Ilimitadas poligonales, circulares y de ruta

Nombre de la plataforma	Beneficios	Adicionales
GPS & cloud it	Administrar costos, gestionar eventos, tener control de manejo, seguimiento de mercancías, validación de rutas, optimización de tiempos de entrega y traslado.	NA
Samsara	Métricas de consumo y eficiencia de las rutas en la plataforma con manejo intuitivo.	Lectura de temperatura y presión ambiental. Voltaje de batería cada 2 minutos. Velocidad de vehículo cada 30 s o cuando la velocidad cambia en más de 3 mph. Nivel de combustible y condiciones del motor.
Wialon	Elementos de interfaz personalizados según la marca del cliente. Control de flota con análisis de consumo de combustible, optimización de rutas, control con geocercas.	NA
Pumamovil	Conocer trayectoria en tiempo real del pumabús, consulta de bicicletas disponibles, cálculo de la mejor ruta.	NA
GPScontrol	Módulo de Puntos de interés ilimitados Street View dinámico, Tráfico y Mapas Google Set de reportes parametrizables. App para dispositivos móviles.	Conteo de pasajeros con sensores de presencia en la entrada de la unidad Telemetría Sensor de combustible Audio de dos vías en cabina.

Tabla 3.6: Benchmarking Plataformas.

3.4.2 Evaluación comparativa de soluciones - Dispositivos IoT

Además de las plataformas evaluadas, es importante considerar los sistemas de monitoreo GPS convencionales que se utilizan comúnmente en el transporte público o flotillas privadas. Estos dispositivos suelen estar configurados para enviar información mediante redes celulares como 3G, 4G o LTE, utilizando tarjetas SIM y contratos de datos. Si bien pueden ofrecer buena calidad de servicio y altas tasas de transferencia, presentan diversas desventajas para su aplicación en el presente proyecto. Entre ellas se encuentran: mayor consumo energético, necesidad de cobertura celular constante, dependencia de servicios de telefonía móvil con suscripciones mensuales, y costo de implementación y operación más elevado.

Estas características (Tabla 3.7) los hacen menos adecuados para entornos donde se busca eficiencia energética, autonomía prolongada y bajos costos operativos, como en el caso del sistema de monitoreo propuesto para las unidades de Pumabús.

Nombre del dispositivo	Imagen de referencia	Enlace de información	Precio
TK905-4G		https://www.tk-star.com/product/TK905-4G.html	\$25 USD
ST4315		http://www.suntechint.com/board/products/board_view.php?num=36	No disponible
ST4945		https://www.suntechint.com/board/products/board_view.php?num=58	No disponible
T4 Tracker		https://partners.sigfox.com/products/zenit-t4 http://www.amosenseiot.com/products/t4	No disponible
T2C Tracker		https://partners.sigfox.com/products/t2c-rc1 http://www.amosenseiot.com/products/t2	No disponible

Nombre del dispositivo	Descripción	Dimensiones	Peso
TK905-4G	Dispositivo rastreador GPS portátil con batería recargable y carcasa imantada. Diseñado para vehículos o motocicletas. Transmite datos a través de redes móviles 2G/GSM mediante tarjeta SIM.	90 × 72 × 22 mm.	168 g.
ST4315	Rastreador con características esenciales LTE Cat M1/NB-IoT/2G 6 conexiones por cable.	94 x 55 x 23 mm.	103 g.
ST4945	Rastreador con características esenciales LTE Cat M1/NB-IoT/2G.	50.5 x 75 x 32.5 mm.	113 g.
T4 Tracker	Solución todo en uno para monitorear y rastrear activos fijos y/o móviles, recuperar coches, bicicletas, motos, camiones, vehículos utilitarios, etc. Múltiples opciones de soporte de montaje (tubo, brida, tipo tornillo). Con mecanismo de bloqueo antirrobo. Clasificación IP65 ideal para la industria minorista y sistemas de distribución. Transmisión mundial basada en Sigfox Monarch sin complejos acuerdos de roaming de datos. Batería de larga duración y valla virtual (geovalla).	110 × 26 × 29 mm.	104 g.
T2C Tracker	Transmite datos de ubicación de flotas, así como información de golpes y temperatura. Optimiza rutas, previene pérdidas y extravíos. Ideal para prevención de robos, detección de vibraciones. Clasificación IP67/IK10. Transmisión mundial basada en Sigfox Monarch sin complejos acuerdos de roaming de datos y permite el seguimiento de la carga entre países. Batería de larga duración y valla virtual (geovalla).	110 × 26 × 29 mm.	163 g.

Nombre del dispositivo	Variable de medición	Principio de funcionamiento	Exactitud	Temperatura de operación
TK905-4G	Latitud, longitud y altitud, aceleración, campo magnético y temperatura.	Sistema de posicionamiento global (GPS) + red GSM (2G).	5 m.	-20°C a 55°C.
ST4315	Latitud, longitud, altitud, aceleración en tres ejes y temperatura.	Sistema de posicionamiento global (GPS). Geocerca circular/poligonal.	±3 m CEP*50 %, 2D RMS, -130dBm, > 6 Stats.	-30°C a 80°C
ST4945	Latitud, longitud y altitud, aceleración, campo magnético y temperatura.	Sistema de posicionamiento global (GPS).	±3 m CEP*50 %, 2D RMS, -130dBm, > 6 Stats.	-20°C a 60°C
T4 Tracker	Latitud, longitud, altitud, aceleración, humedad, presión, campo magnético y temperatura.	Sistema de posicionamiento global (GPS).	±0.2°C, ±3.5 %RH, ±1 hPa.	-40°C a 80°C
T2C Tracker	Latitud, longitud y altitud, aceleración, campo magnético, presión, humedad y temperatura.	Sistema de posicionamiento global (GPS).	±0.2°C, ±3.5 %RH, ±1 hPa.	-40°C a 80°C

Nombre del dispositivo	Voltaje de operación	Corriente de operación	Especificaciones	Beneficios
TK905-4G	3.7 V (batería interna).	5V entrada para carga, 50 mA promedio.	Batería de 5000 mAh Comunicación por red celular (requiere SIM) Imán para fijación en vehículos Alarma de movimiento, exceso de velocidad y retiro.	Batería de larga duración (hasta 30 días en reposo) Compatible con app móvil y seguimiento en tiempo real.
ST4315	12 V.	40-50 mA.	Bluetooth 4.2, protocolos de transmisión TCP, SMS, Tipo de receptor GPS & Glonass /SBAS: WAAS, EGNOS, MASA. Protocolos LTE Cat M1//NB-IoT/2G, NFC.	Monitoreo constante de ubicación, optimización de rutas, duración de batería de hasta 3 días.
ST4945	3.7 V.	40-50 mA	Bluetooth 4.2, protocolos de transmisión TCP, SMS, Tipo de receptor GPS & Glonass /SBAS: WAAS, EGNOS, MASA. Protocolos LTE Cat M1//NB-IoT/2G, NFC.	Capacidad de batería de 3000 mAh, monitoreo constante de ubicación, bloqueo de unidades. Alerta de baja batería.
T4 Tracker	3.6 V.	-	Protocolos Sigfox, Wifi, Bluetooth Low Energy (BLE), NFC. Celda de batería Li-SOCl ₂ .	Capacidad de batería de 3650 mAh con vida de 10 años, monitoreo constante de ubicación, optimización de rutas y recuperación de vehículos robados.
T2C Tracker	3.6 V.	-	Protocolos Sigfox, Wifi, Bluetooth Low Energy (BLE), NFC. Celda de batería Li-SOCl ₂ .	Capacidad de batería de 5000 mAh con vida de 10 años, monitoreo constante de ubicación y optimización de rutas.

Tabla 3.7: Benchmarking Dispositivos.

La revisión de soluciones disponibles en el mercado, tanto a nivel de plataformas como de dispositivos, permitió identificar características como funcionalidad y escalabilidad. Sin embargo, también se observaron limitantes para su implementación en el contexto de Ciudad Universitaria, entre ellas: el elevado costo de implementación por unidad, la necesidad de infraestructura específica como redes móviles 4G/LTE o gateways propios, así como la falta de flexibilidad para adaptar las soluciones a las necesidades operativas concretas del sistema Pumabús.

Las plataformas analizadas ofrecen interesantes herramientas de visualización y análisis, pero muchas de ellas están diseñadas para flotillas comerciales y requieren licencias o suscripciones costosas. En cuanto a los dispositivos, a pesar de contar con precisión aceptable y características adicionales como sensores ambientales o alertas inteligentes, el hecho de que dependan de redes celulares y su diseño cerrado dificultan su integración personalizada con otros sensores o plataformas.

Por lo anterior, se optó por el desarrollo de un sistema propio, adaptable, de bajo consumo energético, que le permita alcanzar autonomía prolongada y cobertura sin requerir infraestructura local adicional. Este enfoque no sólo resulta económico, sino que permite mayor personalización del hardware, firmware e interfaz, alineándose con las necesidades específicas del contexto universitario.

4 ETAPA DE DESARROLLO DE CONCEPTO

4.1 Evaluación comparativa en el mercado - Componentes para la propuesta de solución

En la etapa de selección de los componentes esenciales para el diseño del dispositivo de solución, es conveniente realizar un análisis de los sensores disponibles en el mercado que podrían satisfacer los requisitos para el funcionamiento del sistema. Así se evalúan y comparan diversas opciones en términos de especificaciones técnicas, rendimiento, costo y compatibilidad con el diseño propuesto.

La elección adecuada de sensores no solo influye en la precisión y eficiencia del dispositivo, sino que también determina su fiabilidad y durabilidad en condiciones operativas reales. Además, la evaluación considera los requerimientos específicos de los distintos tipos de usuarios del sistema, tales como la necesidad de una detección precisa de ascensos y descensos, la entrega oportuna de información, y la integración con una interfaz intuitiva.

A continuación, en las tablas 4.1 y 4.2, referentes a las tareas de conteo de pasajeros y obtención de ubicación respectivamente, se presenta la comparación de los sensores hallados, considerando parámetros como el rango de medición, exactitud, consumo de energía y facilidad de integración, para asegurar que la selección final se alinee con los objetivos del proyecto y las expectativas de quienes harán uso de la solución.

4.1.1 Sensores para conteo de pasajeros

Nombre del dispositivo	Imagen de referencia	Enlace de información	Precio
Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK		https://uelectronics.com/producto/e18-d80nk-sensor-de-proximidad-npn-3-80cm/ https://naylampmechatronics.com/sensores-proximidad/236-sensor-de-proximidad-fotoelectrico-infrarrojo-e18-d80nk-npn-no.html	\$52-\$100 MXN
Sensor de Proximidad Inductivo LJ12A3-4-Z/BX		https://uelectronics.com/producto/lj12a3-4-z-bx-npn-sensor-de-proximidad-inductivo/	\$58-\$295 MXN
Sensor Fotoeléctrico E3F-20DNL-Z		https://www.amazon.com.mx/Taiss-fotoel%C3%A9ctrico-interruptor-infrarrojos-E3F-20DNL-Z/dp/B07VPSSLBR	\$663 MXN
Sensor de proximidad capacitivo LJC18A3-H-Z/BX		https://naylampmechatronics.com/sensores-proximidad/296-sensor-de-proximidad-capacitivo-ljc18a3-h-zbx-npn-no.html	\$196 MXN
Sensor Ultrasonico US-100		https://uelectronics.com/producto/sensor-ultrasonico-us-100-compensacion-de-temperatura/	\$76 MXN
Sensor Ultrasonico HC-SR04		https://uelectronics.com/producto/sensor-ultrasonico-hc-sr04/	\$28 MXN

Nombre del dispositivo	Descripción	Principio de funcionamiento
Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK	Sensor de proximidad fotoeléctrico(cambio de luz infrarroja) que permite detectar objetos sin necesidad de contacto. LED rojo indicador de detección.	Se basa en la detección del cambio de intensidad de luz. Se emite un haz de luz infrarroja que al impactar sobre un objeto es reflejado, este reflejo es detectado por el receptor fotoeléctrico que está integrado dentro (tipo difuso) y se emite una señal de detección. El sensor no emite una señal con la distancia, únicamente muestra estados on/off.
Sensor de Proximidad Inductivo LJ12A3-4-Z/BX	Detectan la presencia o ausencia de objetos metálicos. Cuando el sensor detecta un objeto metálico su salida eléctrica cambia de estado (ON/OFF), de forma similar a un interruptor.	Un circuito oscilador alimenta una bobina inductora que genera un campo electromagnético frente a la superficie activa del sensor. Si un objeto metálico interactúa con el campo, una corriente parásita es inducida y ocasiona que el circuito oscilador pierda energía. Un circuito comparador detecta el cambio y emite una señal de activación/desactivación a la salida en el sensor.
Sensor Fotoeléctrico E3F-20DNL-Z	Sensor de proximidad fotoeléctrico. Permite detectar objetos pequeños a largas distancias. Para uso en interiores y exteriores. Sensor de alarma.	Se basa en la detección del cambio de intensidad de luz. Al pasar un objeto entre el emisor y receptor se interrumpe la luz y se reduce la cantidad de luz que le llega al receptor, esto lo considera como el paso de un objeto.
Sensor de proximidad capacitivo LJC18A3-H-Z/BX	Permite detectar objetos metálicos y no metálicos en un rango de hasta 5 mm mandando un ON o un OFF.	Emite un campo eléctrico que se ve alterado en presencia de algún objeto cercano, el sensor detecta el cambio en la capacitancia y emite una salida eléctrica de activación/desactivación.
Sensor Ultrasonico US-100	Módulo que permite detectar distancias sin contacto desde 2cm hasta 4.5 m. Incorpora un sensor de temperatura para corregir los resultados de medición.	El emisor piezoeléctrico emite 8 pulsos de ultrasonido(40KHz) luego de recibir la orden en el pin TRIG, las ondas de sonido viajan en el aire y rebotan al encontrar un objeto, el sonido de rebote es detectado por el receptor piezoeléctrico, luego el pin ECHO cambia a Alto (5V) por un tiempo igual al que demoró la onda desde que fue emitida hasta que fue detectada, el tiempo del pulso ECO es medido por el microcontrolador y así se puede calcular la distancia al objeto.
Sensor Ultrasonico HC-SR04	Módulo que sirve para medir distancias por medio de sus dos transductores: un micrófono y altavoz.	Genera pulsos de alta frecuencia(no perceptible por el ser humano) que rebota en los objetos cercanos y es reflejado hacia el sensor, que es captado por un micrófono. Son sensores económicos y fáciles de usar.

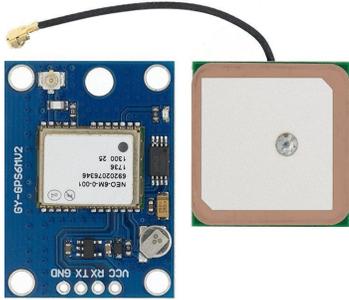
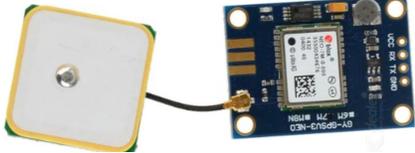
Nombre del dispositivo	Salida analógica o digital	Rango de detección	Voltaje de alimentación	Voltaje de operación	Corriente de operación	Temperatura de operación
Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK	Digital	3-80 cm	5 V	5 V	20 mA	-25°C a 70°C
Sensor de Proximidad Inductivo LJ12A3-4-Z/BX	Digital	4 mm ± 10 %	6-36 V	6-36 V	Máximo 300 mA	-25°C a 55°C
Sensor Fotoeléctrico E3F-20DNL-Z	Analógica	0-20 m	6-36 V	6-36 V	300 mA	No especificada
Sensor de proximidad capacitivo LJC18A3-H-Z/BX	Digital	1-5 mm	5 V	6-36 V	20 mA	-25°C a 70°C
Sensor Ultrasonico US-100	Digital	2-450 cm	2.4 - 5.5 V	2.4 - 5.5 V	15 mA	-20°C a 70°C
Sensor Ultrasonico HC-SR04	Digital	2-450 cm	5 V	5 V	15 mA	-15°C a 70°C

Nombre del dispositivo	Precisión	Especificaciones	Beneficios
Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK	Errores linealidad de 0.15 % (Muy preciso)	-Peso: 42 g -Longitud cable: 40 cm -D: 18 mm, L: 50 mm -Salida: NPN(Normalmente Abierto) -Tiempo de respuesta <2 ms	-Ajuste de distancia de detección. -Fácil integración con diferentes tarjetas (Arduino,ESP 32). -El más económico -Alcance suficiente para entrada del pumabús. -LED indicador de detección. - Es modo difuso. Receptor y emisor en un mismo dispositivo. -Detecta tanto objetos no metálicos como metálicos.
Sensor de Proximidad Inductivo LJ12A3-4-Z/BX	No especificada	-Peso:45 g -L: 64 mm, D: 12 mm -Frecuencia de respuesta: 100 HZ -Normalmente Abierto	-No se ve afectado por características como la luz, el polvo o líquido.
Sensor Fotoeléctrico E3F-20DNL-Z	No especificada	-Peso:70 g -Longitud Cable: 120 cm -D: 12 mm, L: 66 mm Construcción resistente a la intemperie (IP66)	-Alcanza grandes distancias de medición.
Sensor de proximidad capacitivo LJC18A3-H-Z/BX	No especificada	-Peso:108 g -Frecuencia de respuesta: 0.5 Hz -Longitud del cable: 1.80 m -D: 18, L: 70 mm -Normalmente Abierto	-Detecta objetos metálicos y no metálicos. -Polvo, líquidos no lo afectan tanto. . -LED indicador de detección.
Sensor Ultrasonico US-100	3 mm +1 %	-Dimensiones: 45 x 20 x 15 mm -Frecuencia de trabajo: 40 KHz -Ángulo de detección: 15 ° -Duración mínima del pulso: 10 μS -Comunicación UART TRIG/ECHO -Perro guardián en el interior	- Detecta cualquier tipo de objeto.
Sensor Ultrasonico HC-SR04	± 3 mm	-Peso:9 g -Dimensiones: 45 x 20 x 15 mm -Frecuencia de trabajo: 40 KHz -Duración mínima del pulso: 10 μS	- Detecta cualquier tipo de objeto.

Nombre del dispositivo	Inconvenientes
Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK	-Puede verse afectado por la luz, el polvo o líquido.
Sensor de Proximidad Inductivo LJ12A3-4-Z/BX	-Solo detecta materiales metálicos. -Distancias muy cortas de detección.
Sensor Fotoeléctrico E3F-20DNL-Z	-De las opciones más costosas. -Ocupa más espacio.
Sensor de proximidad capacitivo LJC18A3-H-Z/BX	-Rango de medición muy pequeño.
Sensor Ultrasonico US-100	-Tiene un ángulo de detección de 15° y si la superficie a detectar no es uniforme se complica su detección.
Sensor Ultrasónico HC-SR04	-A más de 3 m la sensibilidad del dispositivo es muy poca, debe tener un ángulo de medición de 15° a 30° . -Tiempo de espera mínimo: $20 [\mu S]$.

Tabla 4.1: Benchmarking sensores para conteo de pasajeros.

4.1.2 Módulos para obtención de ubicación

<p>Imagen de referencia</p>		
<p>Nombre del dispositivo</p>	<p>Módulo GPS NEO-6MV2</p>	<p>Módulo GPS Ublox NEO-7M</p>
<p>Enlaces de información</p>	<p>https://uelectronics.com/producto/neo6m-v2-gps/</p>	<p>https://techmake.com/products/gpsmod00563</p>
<p>Precio</p>	<p>\$116 MXN</p>	<p>\$470 MXN</p>
<p>Descripción</p>	<p>Módulo receptor para aplicaciones de geolocalización con una antena, memoria EEPROM para guardar datos y una batería para respaldar la configuración del módulo.</p>	<p>Módulo receptor para aplicaciones de geolocalización con una antena. Tiene una batería de respaldo que permite conservar los datos recolectados si le deja de llegar alimentación por parte de la fuente de poder.</p>
<p>Variable de medición</p>	<p>Latitud, longitud y altitud</p>	<p>Latitud, longitud y altitud</p>
<p>Principio de funcionamiento</p>	<p>Sistema de posicionamiento global (GPS).</p>	<p>Sistema de posicionamiento global (GPS).</p>
<p>Salida analógica o digital</p>	<p>Digital</p>	<p>Digital</p>
<p>Rango de detección</p>	<p>-Altura máxima: 18000 -Velocidad máxima: 515 m/s</p>	<p>-Altura máxima: 18000 -Velocidad máxima: 515 m/s</p>
<p>Voltaje de alimentación</p>	<p>3.3-5 V</p>	<p>2.7-5 V</p>

Módulos para obtención de ubicación

Nombre del dispositivo	Módulo GPS NEO-6MV2	Módulo GPS Ublox NEO-7M
Voltaje de operación	3.3-5 V	3.3/5 V
Corriente de operación	45 mA	35 mA
Temperatura de operación	-40 a 85[°C]	-40 a 85[°C]
Precisión	2.5 m	2.5 m
Exactitud	1 [μ S]	1 [μ S]
Especificaciones	<ul style="list-style-type: none"> -Peso:18 g -Antena: 25 x 25 mm -Módulo: 36 x 25.9 mm -Comunicación: UART -Sistema de coordenadas: WGS-84 -Sensibilidad: Captura 148[dBm], Rastreo: 161 [dBm] -Frecuencia receptora: L1 (1575.42 Mhz) -Tiempo de conexión: Primera vez: 5-10 min. Segunda vez: 40-60 s 	-Receptor Ublox NEO-7M
Beneficios	<ul style="list-style-type: none"> -Compatible con cualquier microcontrolador -Dimensiones reducidas -Librerías de Arduino existentes -(Tiny GPS) 	<ul style="list-style-type: none"> -Antena activa de alta ganancia. -Interfaz IPX, para conectar diferentes antenas activas. -Batería de respaldo recargable -Mantiene los datos recolectados cuando se apaga -Admite arranques en caliente.
Inconvenientes	<ul style="list-style-type: none"> -No funciona en espacios cerrados. -Con poca corriente no funciona correctamente y demora el tiempo de conexión. 	<ul style="list-style-type: none"> -No funciona en espacios cerrados. -Tiempo de adquisición largo al frío (cold start): hasta 30 s, si se enciende después de larga inactividad. -Tamaño y peso mayores que otras alternativas como el NEO-6MV2.

Tabla 4.2: Benchmarking módulos para obtención de ubicación.

Tras analizar las alternativas en términos de precisión, integración eléctrica, costo y compatibilidad con el sistema, se eligió el **sensor infrarrojo E18-D80NK** como componente principal para el conteo de pasajeros, por su capacidad de detección precisa en un rango

ajustable de hasta **80 cm** y su salida digital directa compatible con microcontroladores de **3.3 V y 5 V**; y el **Módulo GPS NEO-6MV2** para la geolocalización, por su rápida adquisición de señal [**TTF** (tiempo para primera ubicación) **<27 s** en condiciones frías], su alta sensibilidad de recepción (**-161 dBm**), y su compatibilidad con protocolos **UART**, lo que facilita la integración con sistemas embebidos de bajo consumo. Esta elección responde también a los requerimientos de confiabilidad, bajo consumo, bajo costo y fácil instalación establecidos en la etapa de levantamiento de requerimientos.

4.1.3 Red de comunicación

Para la transmisión de datos en este sistema, se consideraron diversas tecnologías de comunicación IoT. Entre las opciones más utilizadas se encuentran Sigfox, LoRaWAN y NB-IoT, cada una con características específicas en términos de consumo energético, cobertura y escalabilidad. A continuación, se presenta una comparación entre estas tecnologías.

Característica	Sigfox	LoRaWAN	NB-IoT
Frecuencias	Bandas ISM sin licencia: -868 MHz en Europa -915 MHz en Norte América -433 MHz en Asia	Bandas ISM sin licencia: -868 MHz en Europa -915 MHz en Norte América -433 MHz en Asia	Bandas de frecuencia LTE con licencia
Ancho de Banda	100 Hz	250 y 125 kHz	200 kHz
Tasa de Datos Máxima	100 bps	300 bps - 50 kbps	200 kbps (uplink) y 20 kbps (downlink)
¿Bidireccional?	Sí	Sí. Tres clases de dispositivos terminales.	Sí
Robustez Ante Interferencias	Muy Alta (Ultra Narrow Band)	Alta (Narrow Band)	Baja
Calidad del Servicio	Baja	Baja	Alta. Trabaja con Bandas de frecuencia licenciadas y protocolo síncrono basado en LTE.
Vida de la Batería	Muy Alta	Alta (LoRa A), en ocasiones se consume un extra de energía para reducir la latencia.	Baja. Los dispositivos terminales consumen energía adicional para proveer mejor calidad de servicio.
Latencia	Baja	Alta (LoRa C)	Muy alta

Característica	Sigfox	LoRaWAN	NB-IoT
Escalabilidad	Soporta hasta 50k dispositivos terminales.	Soporta hasta 50k dispositivos terminales.	Soporta hasta 100k dispositivos terminales.
Tamaño Máximo del Mensaje	12 bytes (uplink) y 8 bytes (downlink)	243 bytes	1600 bytes
Alcance	> 40km	< 20km	< 10km
Comercialización	Presencia en 75 países	Presencia en más de 25 países.	Sobre todo China, pero creciente en Europa
Costos	-Espectro: \$0 -Despliegue: >\$92,246/ Estación Base -Dispositivo terminal: <\$46.12	-Espectro: \$0 -Despliegue: >\$23061/ Estación Base, >\$23,061/ gateway -Dispositivo terminal: \$69.18-115.31	Espectro: >500M€/MHz -Despliegue: >15000€/Estación Base -Dispositivo terminal: 8-20€

Tabla 4.3: Comparación de redes de comunicación IoT.

Como puede observarse en la Tabla 4.3, existen múltiples tecnologías de red con características particulares en cuanto a alcance, consumo energético y complejidad de implementación. A partir de esta comparación, se identificó a la red **Sigfox** como la que mejor se alinea con los requerimientos del sistema, en función de los criterios establecidos, como consumo, cobertura y facilidad de despliegue. Esto se detalla en la sección de Definición del concepto.

4.1.4 Comparación de servicios de software IoT en la nube: Ubidots y Losant

Como parte del proceso de selección de herramientas para el desarrollo del sistema, se evaluaron las plataformas aplicativos de servicios en la nube orientadas al Internet de las Cosas (IoT) *Ubidots* y *Losant*, por su accesibilidad, documentación y funcionalidad. A continuación, se presenta la tabla comparativa 4.4 basada en criterios técnicos y de usabilidad, con el objetivo de identificar cuál de estas plataformas se adapta mejor a los requerimientos específicos del proyecto.

Característica	Ubidots	Losant
Interfaz de usuario	Intuitiva y centrada en el usuario final, con widgets prediseñados y dashboards fáciles de usar.	Más orientada a desarrolladores, con enfoque técnico y posibilidad de personalización avanzada.
Nivel de personalización	Limitada a los widgets disponibles en la plataforma. La personalización profunda requiere HTML Canvas.	Alta personalización con HTML, JavaScript, y bloques de desarrollo avanzados. Permite lógica compleja directamente en el dashboard.
Flujos de trabajo (workflows)	Visuales y fáciles de usar para acciones básicas. En la versión gratuita tiene restricciones en cantidad y tiempo.	Sistema de workflows muy robusto, con bloques condicionales, funciones personalizadas y manejo avanzado de eventos.
Acceso a datos crudos	Permite ver datos crudos de los dispositivos y exportarlos.	También permite visualizar y manipular datos directamente desde los workflows.
Documentación y comunidad	Buena documentación, orientación a usuarios nuevos. Comunidad activa.	Documentación detallada y con enfoque para desarrolladores. Comunidad técnica y empresarial.
Planes gratuitos	Versión gratuita con dashboards limitados, pero útil para pruebas rápidas.	Versión gratuita con más herramientas para desarrollo profesional, aunque con límites más estrictos en dispositivos.
Casos de uso recomendados	Ideal para prototipos rápidos, educación, pruebas de concepto orientadas al usuario final.	Adecuada para desarrollos personalizados, soluciones escalables y pruebas con alto grado de control lógico.

Tabla 4.4: Comparación entre plataformas IoT: Ubidots y Losant

Como plataforma de despliegue de información, Ubidots se caracteriza por su interfaz amigable, ideal para principiantes, así como por su variedad de herramientas visuales listas para usarse, lo que permite crear dashboards de manera rápida y sencilla.

En contraste, Losant ofrece una plataforma **más robusta**, orientada a desarrolladores, con un sistema de **workflows visuales altamente personalizables** y una gran capacidad de integración con diversos protocolos. Aunque Ubidots facilita el diseño visual, Losant fue la elegida para este proyecto por su **mayor flexibilidad, escalabilidad y control en la lógica del sistema**, permitiendo una implementación más detallada del flujo de datos y la automatización del comportamiento del dispositivo.

4.2 Definición del concepto

Las herramientas hasta ahora empleadas han servido para clarificar, delimitar y consolidar la idea de la solución buscada. Dado que la necesidad principal es optimizar el uso del transporte Pumabús, el concepto resultante se basa en la implementación de un sistema de monitoreo capaz de rastrear constantemente la ubicación y ocupación de las unidades mediante reportes de actividad.

El sistema propuesto integra sensores infrarrojos para la detección de pasajeros, un módulo GPS para la geolocalización de las unidades y una tecnología de transmisión de datos que garantiza información precisa y actualizada tanto a operadores como a pasajeros. Para una integración eficiente de los componentes electrónicos, resulta conveniente el desarrollo de una placa de circuito impreso (PCB, por sus siglas en inglés). Este tipo de diseño permite optimizar el espacio, mejora la estabilidad eléctrica, facilita el montaje, reduce el riesgo de conexiones incorrectas y permite una reproducción estandarizada del dispositivo, al tratarse de un prototipo que podría escalarse o replicarse en múltiples unidades.

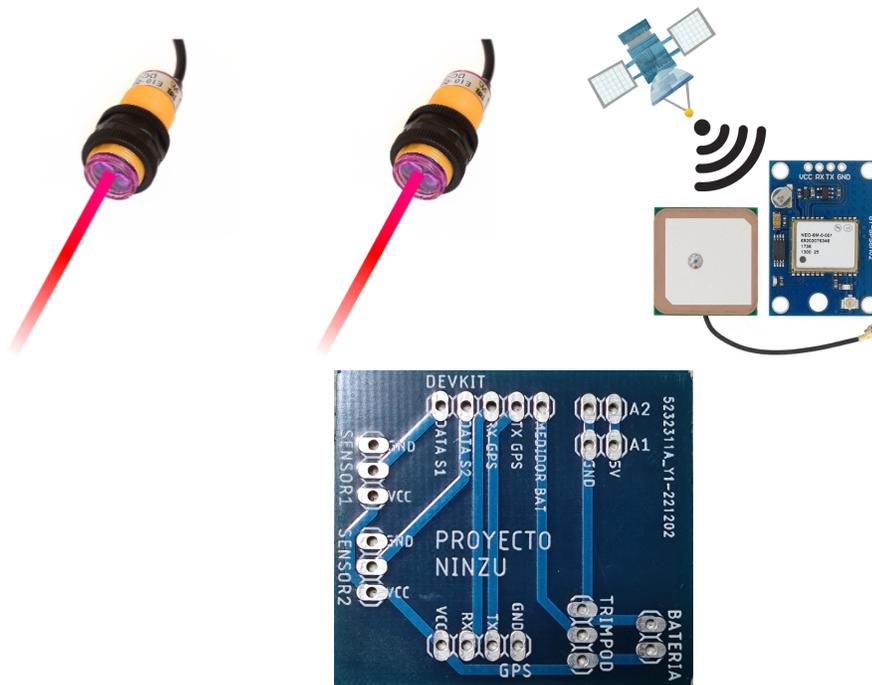


Figura 4.1: Componentes elegidos para conteo de pasajeros y geolocalización y PCB como herramienta de integración de diseño.

La elección de sensores infrarrojos sobre otras opciones, como los ultrasónicos, inductivos, capacitivos, etc., se debe a su precisión en la detección de pasajeros sin necesidad de contacto físico, su bajo consumo energético y su facilidad de integración con el sistema propuesto. A diferencia de los sensores ultrasónicos, que pueden verse afectados por interferencias acústicas en entornos ruidosos, o los sensores de proximidad inductivos y capacitivos, que requieren materiales específicos para una detección eficiente, los infrarrojos ofrecen un equilibrio entre confiabilidad, costo y adaptabilidad a distintas condiciones operativas.

Dado que el dispositivo desarrollado debe ser de bajo consumo energético y transmitir pequeños paquetes de datos de manera eficiente, Sigfox resulta ser la alternativa más adecuada. Esta tecnología ofrece una autonomía significativamente mayor en comparación con NB-IoT, gracias a su protocolo de ultra bajo consumo. Además, a diferencia de LoRaWAN, no requiere la instalación de gateways propios, lo que facilita su despliegue en diversas ubicaciones sin infraestructura adicional.

Si bien la tasa de datos y el tamaño máximo del mensaje en Sigfox son menores en comparación con LoRaWAN y NB-IoT, esta limitación no impacta el desempeño del sistema, ya que los datos transmitidos por los sensores requieren paquetes de información pequeños. Adicionalmente, el mayor alcance de Sigfox en comparación con LoRaWAN y NB-IoT permite una cobertura más amplia, reduciendo la necesidad de múltiples puntos de acceso.

Por estas razones, Sigfox fue seleccionado como el protocolo de comunicación para este desarrollo, optimizando la eficiencia energética, la facilidad de implementación y la cobertura del sistema.

Una vez elegida la red de comunicación Sigfox, se seleccionó el Devkit 2.0 como dispositivo de desarrollo para la etapa de pruebas. Este módulo, diseñado específicamente para operar con la red Sigfox, incluye un microcontrolador integrado, una antena y conectores accesibles para sensores externos. Esto lo convierte en una opción práctica para prototipado. Permite validar rápidamente la comunicación entre el dispositivo y la nube, además de facilitar la programación y pruebas de flujo sin necesitar ensamblajes complejos desde cero y, así, se reducen los tiempos de desarrollo y se minimizan errores en las primeras fases del proyecto.



Figura 4.2: Red Sigfox elegida y Kit de desarrollo Devkit 2.0.

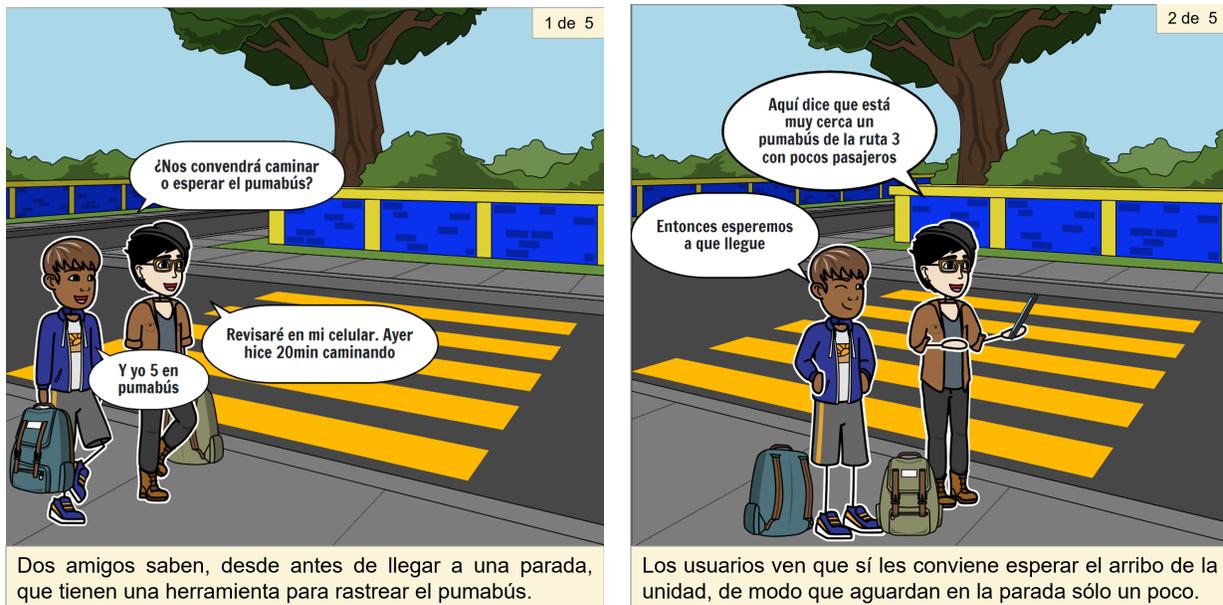
4.3 Cadena de valor IoT de la solución propuesta

Representa los principales módulos funcionales del sistema. Incluye sensores, unidades de procesamiento y módulos de comunicación y GPS. Cada uno de estos elementos cumple un papel específico dentro del flujo de adquisición, procesamiento y transmisión de datos y permite el monitoreo constante y toma de decisiones basadas en información contextual.



4.4 Representación gráfica de la solución propuesta

En la figura 4.4, a través de recursos gráficos, se presentan situaciones similares a las representadas en la sección de definición y descripción del problema, con la diferencia de que, en estas viñetas, los personajes ilustrados tienen acceso desde sus teléfonos celulares a la herramienta del presente proyecto, la cual les resulta útil para decidir si esperar o no la unidad Pumabús.



(a) Los usuarios conocen la existencia de la herramienta digital.

(b) Los usuarios consultan la ubicación actualizada del pumabús.

Figura 4.4: Ejemplo de uso práctico de la solución propuesta (a) y (b).

En la situación de la viñeta (a), se puede observar a un par de amigos consideran dos opciones para llegar a su destino: caminar o esperar el pumabús. Con base en su experiencia previa, conocen el tiempo aproximado que tomaría cada alternativa. Gracias a la herramienta desarrollada, pueden consultar rápidamente la ubicación y la ocupación de las unidades, por lo que acceden a la interfaz del sistema. En la viñeta (b), identifican la posición del pumabús más cercano y su nivel de ocupación; al estimar que el tiempo de espera más el trayecto será menor que caminando, deciden esperar.



(c) La unidad llega tal como se indicaba en la plataforma.



(d) Los usuarios utilizan la plataforma para planificar su traslado de forma eficiente.



(e) Decisiones para asignación de rutas y unidades basadas en datos de la plataforma.

Figura 4.5: Ejemplo de uso práctico de la solución propuesta (c), (d) y (e).

Poco tiempo después, en (c), llega la unidad esperada. Esta secuencia ilustra una toma de decisión consciente e informada por parte de los usuarios, apoyada en los datos proporcionados por el sistema.

La viñeta (d) muestra a una usuaria que revisó en su teléfono celular la ubicación del pumabús y optó por llegar caminando a su destino y avisar al resto de usuarios que esperaban la unidad. Esto evitó una experiencia incómoda y la innecesaria espera para los usuarios.

La situación representada en e) muestra a personal de la DGSGM en la oficina de monitoreo observando las cámaras de algunos paraderos y la información del tablero de coordinadores. Gracias a ello, pueden detectar a tiempo cualquier problema en la distribución de unidades y tomar acciones para resolverlo. Además, observan cómo los usuarios utilizan la información desplegada en el tablero de pasajeros —proporcionada por el sistema de conteo y monitoreo de ubicación— para tomar decisiones que les permitan planificar mejor sus trayectos, contribuyendo así a reducir las aglomeraciones en las estaciones del Pumabús.

5 DESARROLLO DE CONCEPTOS

Para su instalación, el sistema estará contenido en una carcasa de plástico, lo que permitirá la transmisión de datos sin inhibir la señal de comunicación. Su diseño tendrá una geometría simple y un peso ligero, facilitando su colocación cerca de la entrada del pumabús sin obstruir el paso.

5.1 Bocetos y Representaciones Visuales



Figura 5.1: Boceto a mano alzada

Con el fin de comenzar a plasmar las ideas y características que debía cumplir el dispositivo en elementos visuales, se exploraron distintas técnicas de representación. La primera consistió en la elaboración de bocetos a mano alzada con lápiz, lo que permitió aclarar conceptos sobre posibles métodos de colocación y sujeción, así como ubicar zonas viables para su instalación. Un ejemplo de estos primeros trazos se muestra en la figura 5.

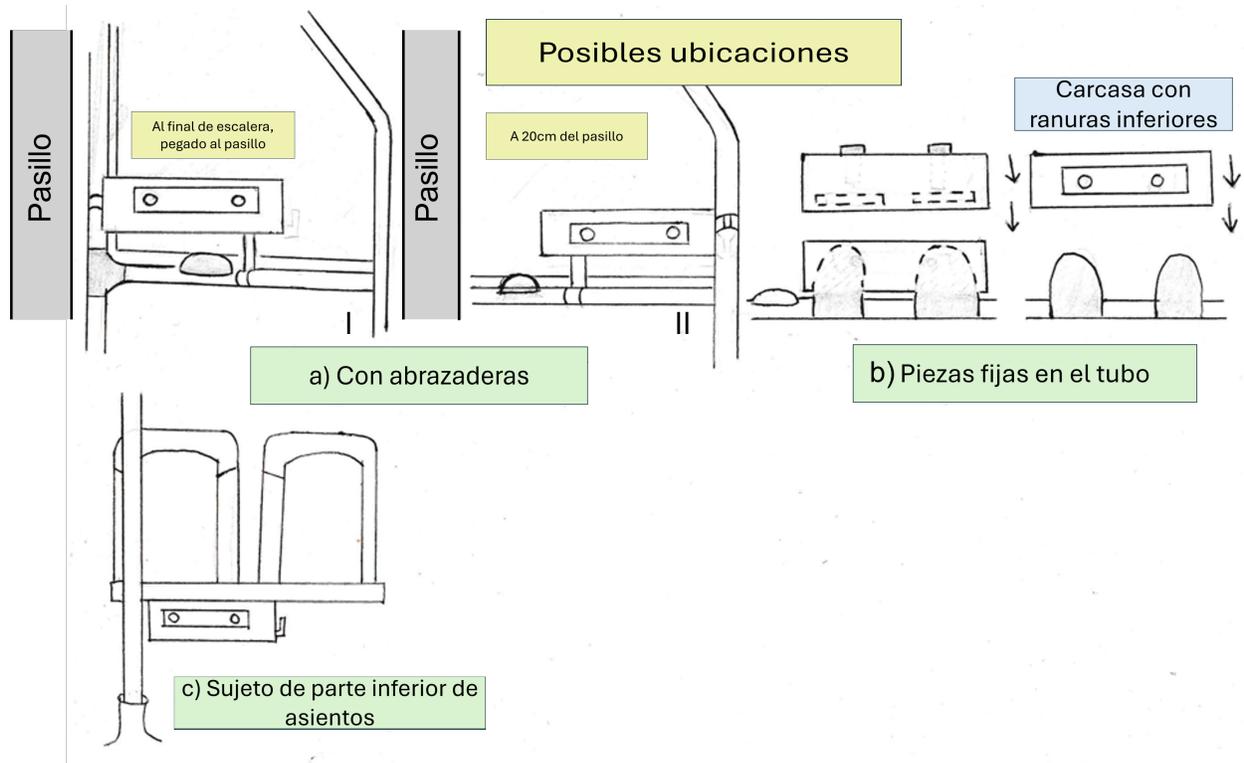


Figura 5.2: Boceto realizado con bolígrafo

Luego, se remarcaron los trazos más importantes con pluma, para que al ser escaneado resultaran más fáciles de distinguir los trazos y se incluyeron señalamientos auxiliares a computadora para complementar el entorno del dibujo.

En la figura 5.2 se muestran cuatro casos de propuesta para colocación del dispositivo pensado. Los dos dibujos de a) son a través de abrazaderas sujetas a los tubos de la entrada del pumabús. I está más cerca del pasillo que de la puerta de entrada y salida, y II está sujeto del tubo más cercano a la puerta y alejado del pasillo. En ambos casos se trata de una caja con forma de prisma rectangular con lado mayor horizontal, una abrazadera lateral y otra inferior cargada al lado opuesto de la otra.

El dibujo de b) es el mismo cuerpo pero con dos ranuras delgadas en la parte inferior de la caja para que esta sea insertada, colocándose por arriba, en dos placas fijadas en el tubo inferior horizontal. Las placas se suponen cada una compuesta por dos mitades, fijas entre sí a través de tornillos y atrapadas entre el tubo inferior y la placa mayor que comunica todos los tubos, con un pequeño espacio entre la placa que se inserta y la mayor.

El dibujo de c) es la misma caja, pero colocada debajo de los asientos de otro modelo de Pumabús, también con sensores apuntando al área de ascenso y descenso de pasajeros.

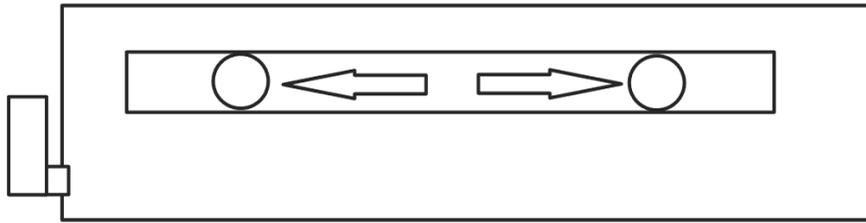


Figura 5.3: Vista frontal. Creado con Google Jamboard.

La figura 5.3 es un esquema realizado con la herramienta Google Jamboard que presenta sólo la cara frontal del cuerpo del dispositivo. De lado izquierdo deja ver la antena que sobresale para la transmisión de datos, así como la parte visible de dos sensores infrarrojos, que integran los emisores y receptores de luz. La separación entre ambos sensores es ajustable dentro de un rango determinado.

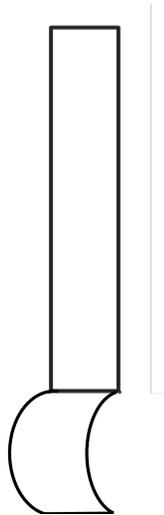


Figura 5.4: Sujetador removible realizado con Google Jamboard.

La figura 5.4, también hecha en Google Jamboard, es una abrazadera con un brazo que proporciona la extensión necesaria para insertarse en alguna ranura del cuerpo principal del dispositivo y así permitir su sujeción.

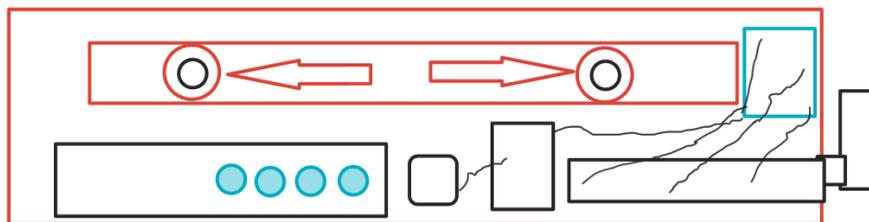


Figura 5.5: Vista trasera e interior con componentes. Creado con Google Jamboard.

El esquema de la Figura 5.5 también se realizó en Google Jamboard presenta sólo la cara trasera del cuerpo del dispositivo. Es la vista que se tendría si se retira la tapa de la parte posterior y se puede distinguir que, dentro, se encuentran distribuidos los componentes fundamentales que harían posible el funcionamiento del sistema. Entre los elementos visibles se hayan los sensores infrarrojos, un banco de batería con su nivel de carga visible, unas placas que representan el módulo GPS, la placa de conexiones, la placa de desarrollo con su antena de transmisión de datos y el cableado entre los elementos.

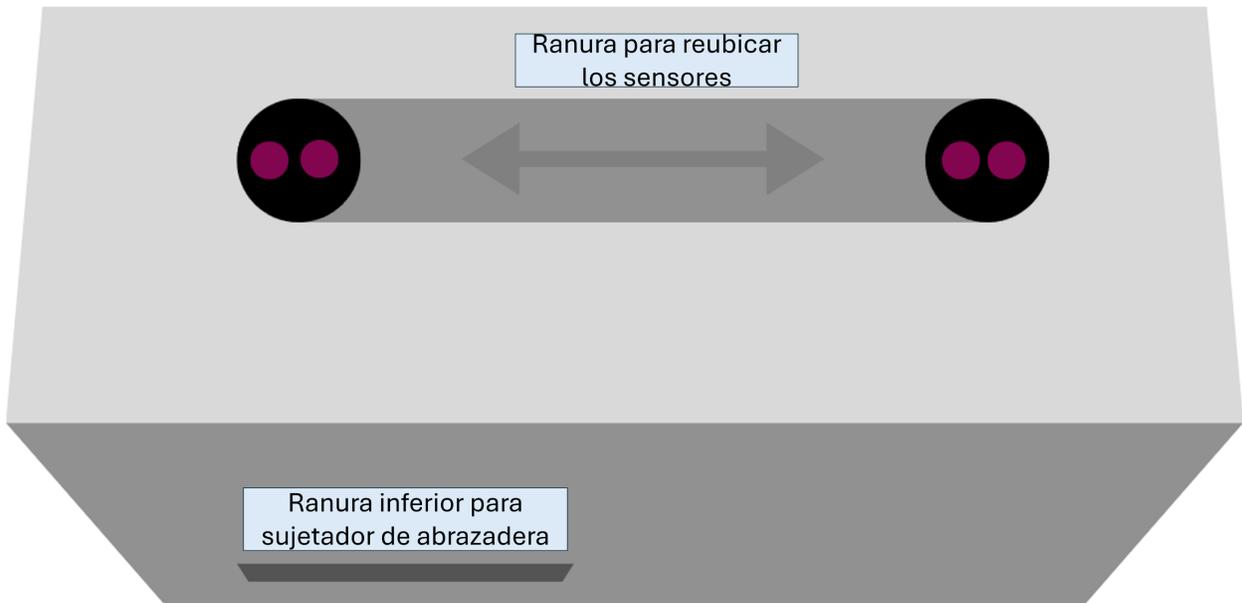


Figura 5.6: Vista frontal 3D. Realizado en Canva.

La figura 5.6, realizada con la herramienta Canva, es resultado de una representación de la caja, ya con profundidad y con una vista desde un punto situado debajo y frente a la cara principal de la caja, donde ya se observan mejor dos sensores infrarrojos y que la distancia entre ambos se puede ajustar.

En esta imagen se observa únicamente la caja con los elementos que quedarían visibles desde el exterior y sin abrazaderas ni placas colocadas, puesto que esta imagen del cuerpo aislado no pretende explicar las posibles ubicaciones ni su estado una vez colocado. Más allá de lo ya mencionado, sólo se aprecia una ranura inferior, en la que podría entrar una abrazadera para su fijación posterior o alguna pieza que se requiera.

5.1.1 Diagramas de concepto

Los diagramas de concepto utilizan representaciones simplificadas y abstractas para ayudar a identificar los elementos clave y establecer las bases para el desarrollo posterior, al permitir comprender mejor la estructura y funcionamiento general de la propuesta de solución antes de proceder a etapas más detalladas del diseño. Permiten visualizar la interacción entre los diferentes componentes del sistema y su relación con el entorno en el que operarán. Existen diferentes tipos de diagramas conceptuales, dependiendo del nivel de abstracción requerido para facilitar la comprensión del sistema.

5.1.2 Diagramas de caja negra

Los diagramas de caja negra son una representación abstracta y simplificada del sistema que no muestra detalles sobre su funcionamiento interno. Modelan el comportamiento del sistema únicamente en términos de las entradas y salidas y en las relaciones entre ellas [13], es decir, en cómo el sistema general interactúa con su entorno. Establecen una base sobre la cual definir las especificaciones detalladas y el diseño técnico del sistema. Las entradas son todos aquellos elementos que el sistema recibe del entorno para funcionar y las salidas son los resultados o productos generados por el sistema tras procesar esas entradas.

En la Figura 5.8 se muestra el diagrama de caja negra genérico que se va a estar empleando para la descripción del sistema y sus subfunciones.



Figura 5.7: Diagrama de caja negra genérico.

5.2 Arquitectura del sistema

Se eligió utilizar el Devkit Sigfox v2.0, que cuenta con un microcontrolador ATmega328, un módem Sigfox para la región RCZ2, interfaces SPI, UART e I2C, así como salidas de voltaje de 3.3V y 5V, lo que facilita la conexión con sensores y módulos externos [1].

Esta elección permitió una integración sencilla con los sensores y el módulo GPS, facilitando el desarrollo del firmware y la validación del sistema en condiciones reales. Además, el devkit incluye antenas y conectores optimizados para la comunicación con la red Sigfox, lo que permitió concentrarse en el diseño lógico del sistema sin complicaciones adicionales de hardware.

Para comprender el funcionamiento del sistema propuesto, se ha descompuesto en varias subfunciones o subsistemas interrelacionados que permiten una operación modular, eficiente y escalable. Esta organización facilita tanto el análisis técnico como la futura implementación y mantenimiento del sistema.

En cuanto al módulo de geolocalización, se eligió el NEO-6M-V2 por su bajo consumo energético, su compatibilidad con múltiples interfaces de comunicación (UART, I2C) y su buena sensibilidad de señal, incluso en entornos urbanos. Este módulo proporciona coordenadas con un nivel de precisión adecuado para el propósito del sistema, permitiendo asociar eventos de ascenso o descenso con una ubicación geográfica aproximada sin requerir dispositivos GPS de mayor costo o complejidad.

Para el subsistema de detección se seleccionaron sensores infrarrojos modelo E18-D80NK, ampliamente utilizados en aplicaciones industriales y de automatización por su alta confiabilidad y bajo costo. Además, opera a un voltaje de 5 V y una corriente de salida baja, compatible con microcontroladores de bajo consumo energético como el ATmega328, lo que favorece un bajo consumo energética del sistema.

En la Figura 5.8 se muestra el diagrama de caja negra elaborado para esta propuesta de solución, y posteriormente se incluyen diagramas de caja negra para cada subsistema junto con una descripción al respecto.



Figura 5.8: Diagrama de caja negra. Sistema general.

Entradas:

- **Alimentación eléctrica:** La energía eléctrica necesaria para el funcionamiento de todos los componentes del sistema. Este suministro se puede proporcionar mediante baterías, fuentes de poder externas o conexión a la red eléctrica de la unidad Pumabús.
- **Señales de sensores:** Señal digital binaria (0 o 1), que cambia de un estado alto (1) cuando no detecta objetos, a un estado bajo (0) cuando sí lo hace. Un par de sensores infrarrojos, colocados uno más al interior que el otro, permite la distinción entre ascensos y descensos.
- **Datos del GPS:** Los datos obtenidos directamente son cadenas de caracteres en formato NMEA (National Marine Electronics Association), que es un estándar utilizado por la mayoría de los dispositivos GPS, del cual se extraen valores como la latitud, longitud, número de satélites, velocidad y más.

Salidas:

- **Número de pasajeros a bordo:** Cantidad actual de pasajeros dentro del pumabús. Resultado de la suma y resta de personas que ascendieron y descendieron. Se lleva registro de cada estación.
- **Ubicación geográfica:** Posición actual de las unidades Pumabús y la visualización con pines y trazos del recorrido en un mapa de la zona.
- **Más reciente y próxima estación:** Información sobre la última parada realizada y la siguiente parada programada.
- **Registro histórico con gráficas:** Representación mediante gráficas del almacenamiento de datos pasados, como el número de pasajeros, ascensos y descensos por estación o por hora. Estos registros permiten análisis a largo plazo y pueden ser utilizados para mejorar la eficiencia del sistema.

5.2.1 Subsistemas

- **Sistema de alimentación:** Suministra a todo el sistema la energía eléctrica necesaria para el funcionamiento de todos los componentes.

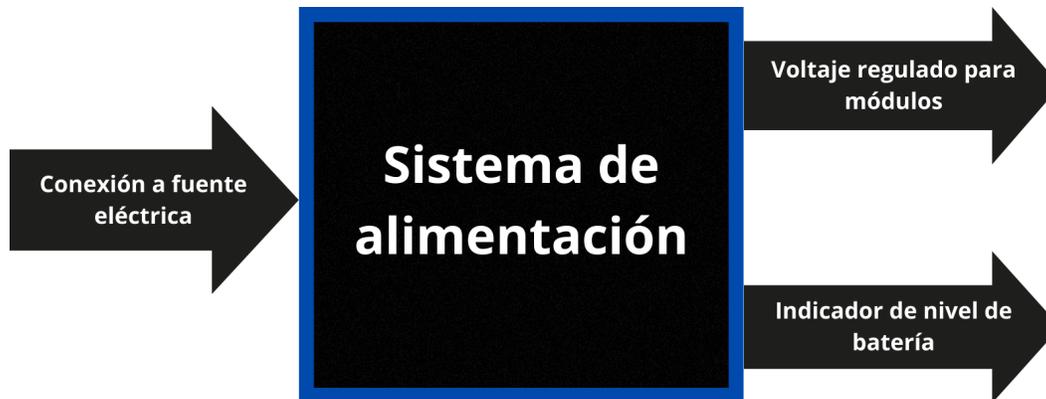


Figura 5.9: Diagrama de caja negra. Sistema de alimentación.

- **Sistema de adquisición de datos/sensado:** Incluye todos los sensores utilizados para la recolección de datos constante, es decir los sensores infrarrojos de conteo de personas y los módulos GPS. Provee datos crudos a la Unidad de Procesamiento.

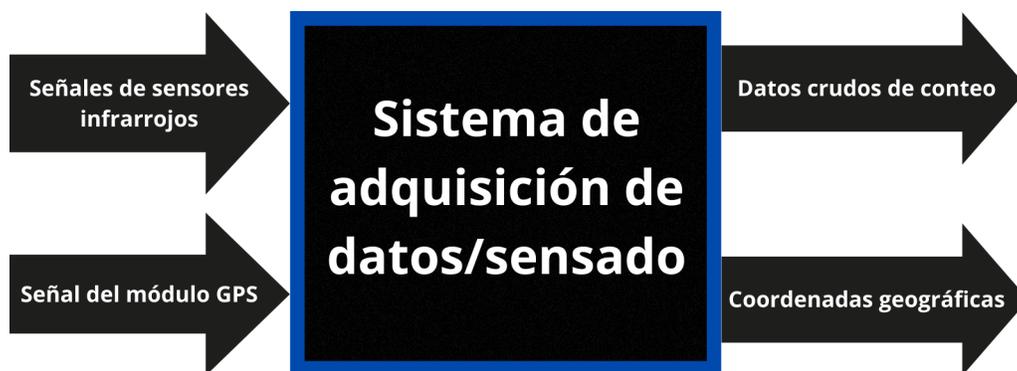


Figura 5.10: Diagrama de caja negra. Sistema de recopilación de datos/sensado.

- **Sistema de geolocalización:** Su función es determinar en todo momento la posición del pumabús mediante un módulo GPS. Un concepto clave para el desarrollo del proyecto es el de la **geocerca**, que consiste en un perímetro virtual definido alrededor de una ubicación geográfica específica. Esta herramienta permite monitorear y controlar la entrada o salida de vehículos en zonas determinadas, con la generación de alertas inmediatas cuando una unidad equipada con GPS cruza dichos límites virtuales.

El GPS opera continuamente y compara la posición de la unidad con las coordenadas predefinidas de cada estación. Al establecer geocercas alrededor de estaciones o paradas, es posible habilitar funciones específicas como el conteo de pasajeros únicamente cuando se detecte que el vehículo ha ingresado a una de estas áreas designadas, lo que asegura que el sistema registre con precisión las subidas y bajadas de pasajeros, evitando datos erróneos y mejorando la gestión del servicio. (Mirar última viñeta de las figuras 5.17 y 5.18 del sistema de conteo de pasajeros.)

Además, la implementación de geocercas contribuye a la seguridad y al cumplimiento de las rutas predefinidas. Esto permite a los coordinadores recibir notificaciones si un vehículo se desvía de su trayecto. Esta capacidad de monitoreo en tiempo real facilita la toma de decisiones informadas.

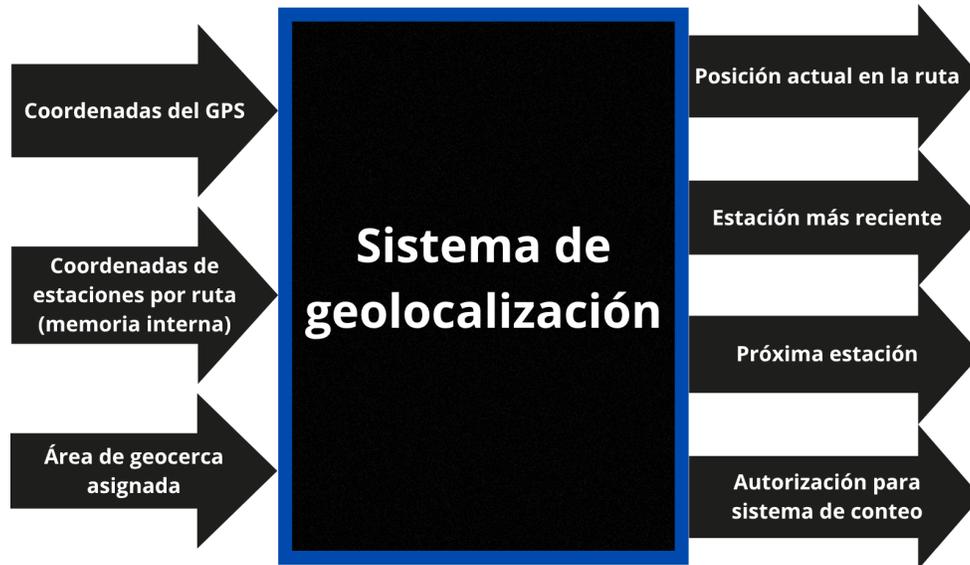


Figura 5.11: Diagrama de caja negra. Sistema de geolocalización.

- Sistema de conteo de pasajeros:** Es responsable de detectar y registrar los ascensos y descensos de los usuarios en el pumabús. Para lograrlo, se utilizan sensores infrarrojos dispuestos en la entrada de la unidad, los cuales funcionan en pares para determinar la dirección del movimiento del pasajero.

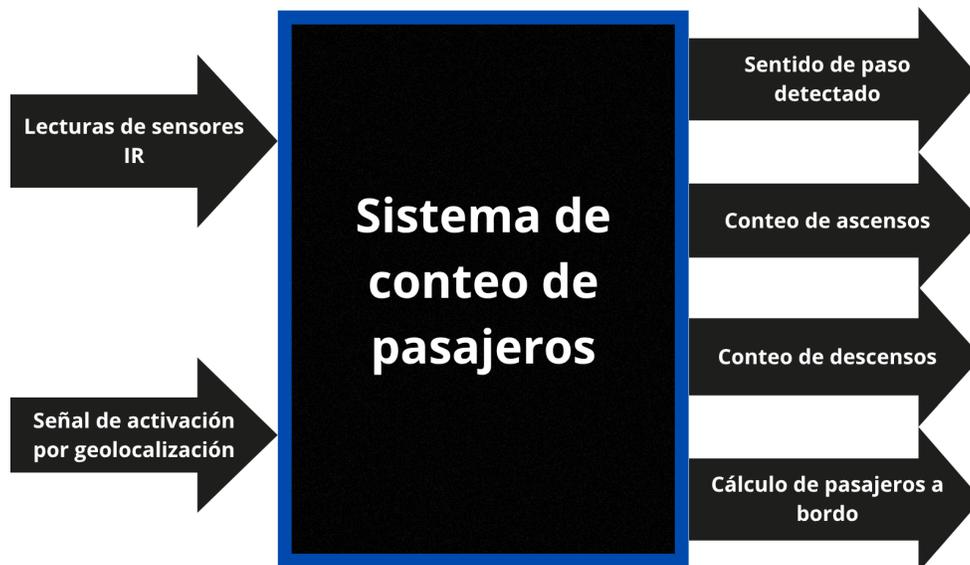


Figura 5.12: Diagrama de caja negra. Sistema de conteo de pasajeros.

- Unidad de procesamiento:** Realiza el procesamiento, análisis y filtrado de los datos recolectados antes de su transmisión. Se asegura de que solo se registren ascensos y descensos cuando el pumabús se encuentre dentro de la geocerca de una estación al restablecer sus respectivos contadores después de cada transmisión, pero mantiene el estado de ocupación total del vehículo para su consulta.



Figura 5.13: Diagrama de caja negra. Unidad de procesamiento.

- Sistema de comunicación:** Es responsable de la transmisión de datos entre la Unidad de monitoreo y procesamiento y los servidores remotos o aplicaciones a través de la red de comunicación Sigfox. Cuando el pumabús deja la estación, el sistema transmite los datos acumulados sobre ascensos, descensos, total de pasajeros al momento y estación que deja, en formato hexadecimal, a la red de comunicación Sigfox una sola vez para optimizar el uso de la red y minimizar el consumo de energía.

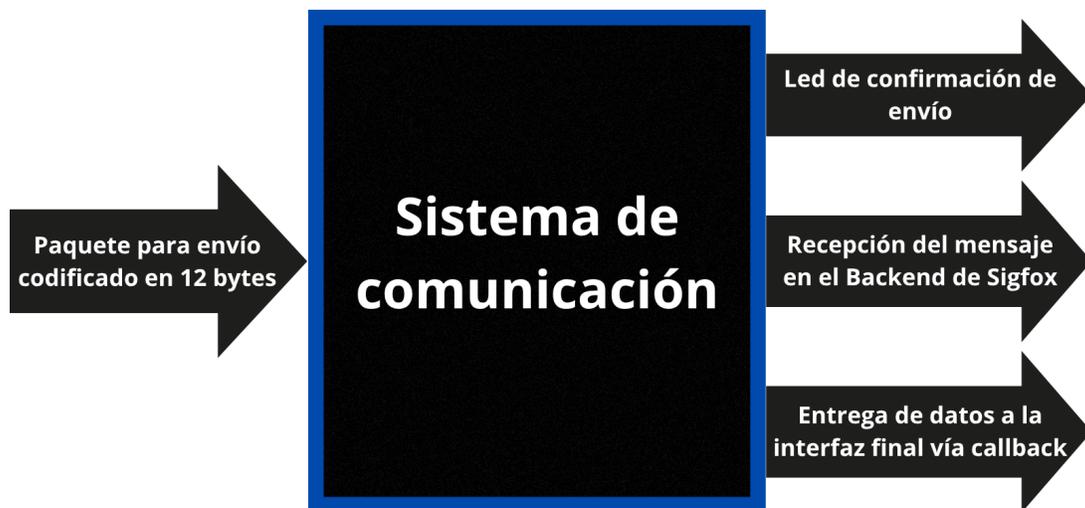


Figura 5.14: Diagrama de caja negra. Sistema de comunicación.

- **Interfaz de usuario:** Permite la consulta y visualización de los datos actualizados a diferentes tipos de usuarios. Losant permite a los desarrolladores configurar la recepción de datos desde Sigfox y gestionar los dispositivos de monitoreo.

A través de un tablero en Losant, los pasajeros pueden consultar información sobre la ocupación y la ubicación actualizadas de los vehículos. Los coordinadores pueden acceder registros históricos de datos con gráficas y paneles con información sobre la ocupación de las unidades y su ubicación actualizadas.

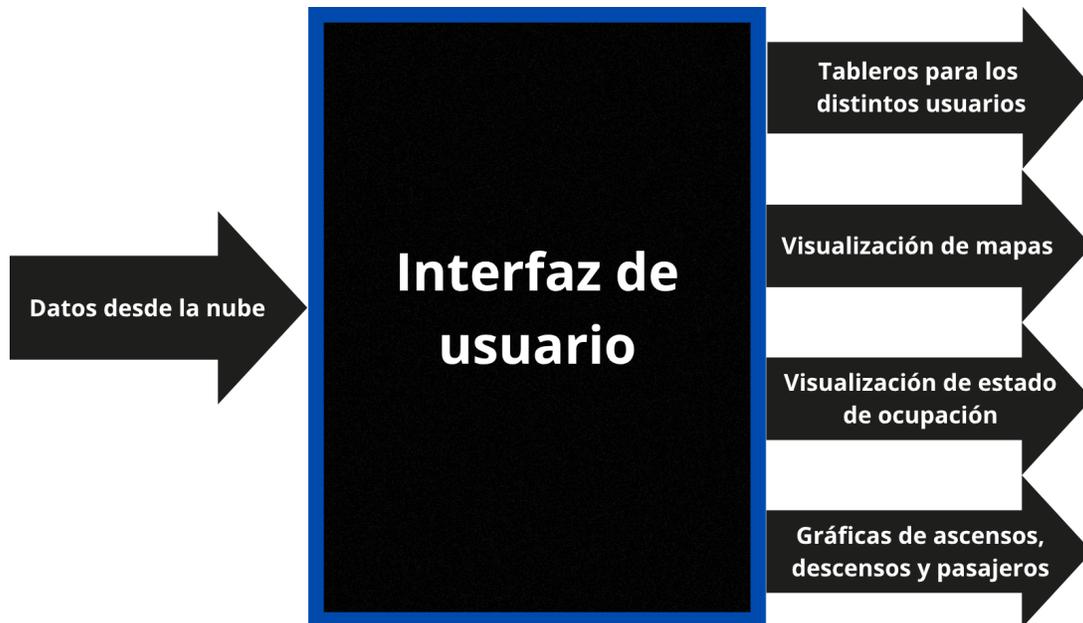


Figura 5.15: Diagrama de caja negra. Interfaz de usuario.

Los diagramas de caja negra permitieron representar esquemáticamente la organización funcional del sistema propuesto, con enfoque en las entradas y salidas de cada subsistema y omisión de detalles internos. Esta abstracción permitió identificar con claridad el rol de cada componente y cómo se integran entre sí. Además, esta representación modular apoyó la toma de decisiones durante el proceso de diseño, al permitir visualizar anticipadamente los flujos de datos y la dependencia entre subsistemas. En conjunto, los diagramas sirvieron como guía para un desarrollo técnico con coherencia entre distintas etapas del proyecto.

5.2.2 Diagrama de flujo de operación

El diagrama de flujo es una herramienta visual que permite representar de manera clara y secuencial los pasos o procesos involucrados en un sistema. Facilita la comprensión, análisis y comunicación del funcionamiento lógico del sistema, ayudando a identificar posibles errores, optimizar procesos y guiar el desarrollo o la implementación de soluciones técnicas.

El diagrama de la figura 5.16 describe la secuencia de eventos dentro del sistema del presente proyecto, desde la captura de datos hasta su transmisión, procesamiento y despliegue de la información.

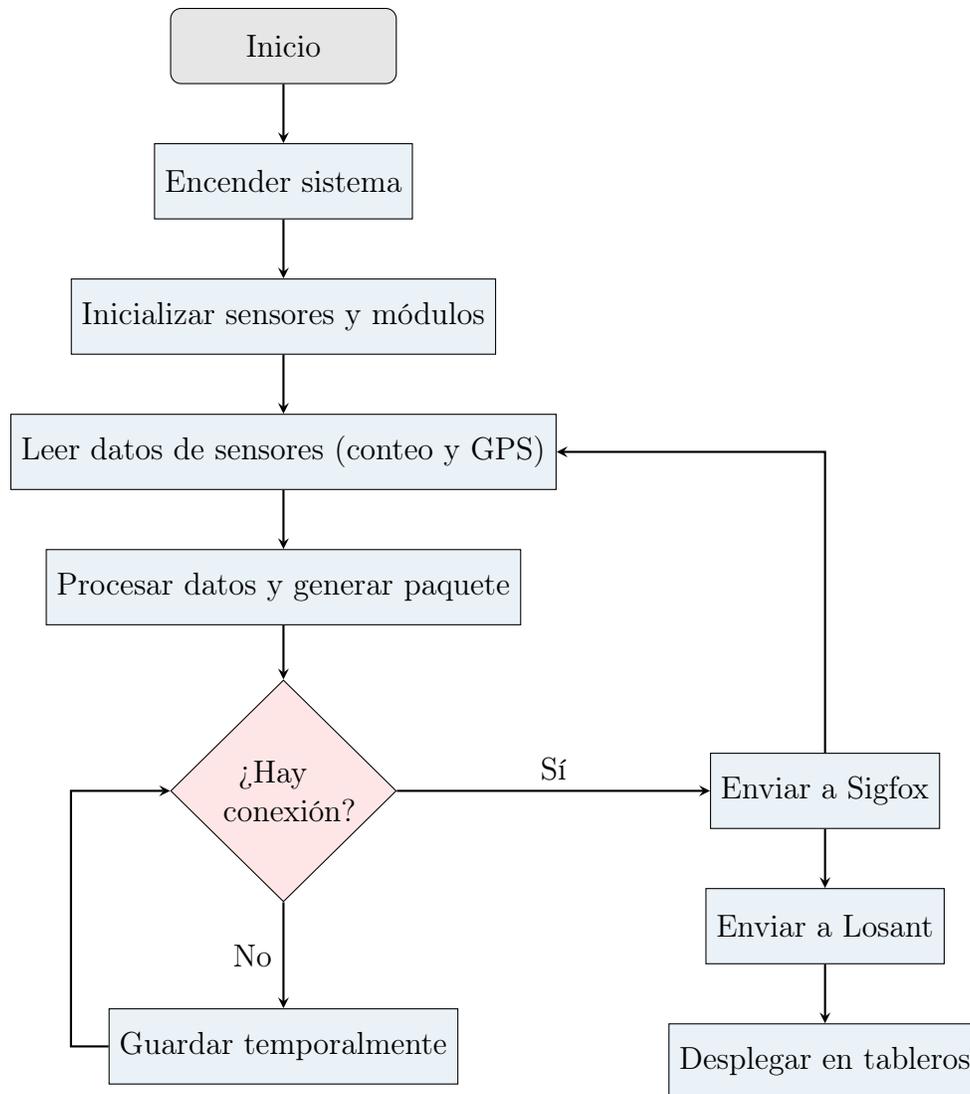


Figura 5.16: Diagrama de flujo de operación.

5.3 Representación gráfica de la detección de ascensos y descensos por el sistema de conteo de pasajeros.

Las viñetas de las secuencias gráficas ilustran detalladamente cómo los sensores infrarrojos detectan el paso de los usuarios al abordar o descender de las unidades, y se actualiza la cuenta y registro del número de pasajeros a bordo.

Este esquema visual busca dar claridad sobre las interacciones clave entre los usuarios, los sensores y la plataforma de despliegue de información, resaltando su impacto en la experiencia de viaje al hacer uso del sistema de transporte Pumabús. A continuación, se representa el proceso de **detección del ascenso de un pasajero por el sistema de conteo de pasajeros**:

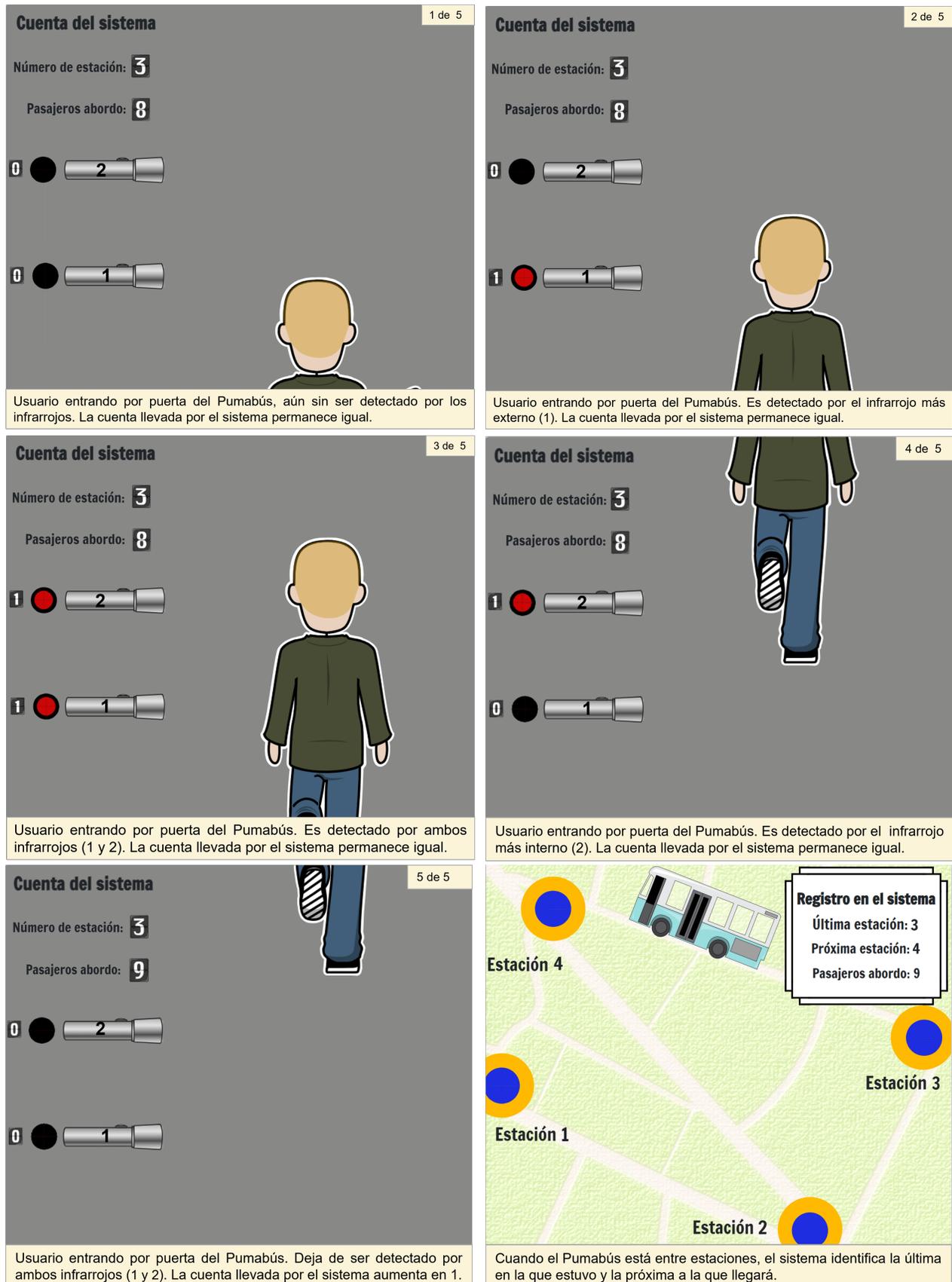


Figura 5.17: Secuencia de activación de infrarrojos a la entrada de un pasajero.

La secuencia de la Figura 5.17 es el caso de ingreso de pasajeros. Inicia con el pasajero fuera de la unidad Pumabús, aún sin activar ninguno de los sensores infrarrojos. **Cuando ingresa al autobús, activa primero el sensor 1, más cercano al exterior, y luego el 2**, ubicado más cerca del interior. Una vez que el pasajero ha cruzado completamente ambos sensores, deja de activar cualquier señal, y en ese momento el sistema de detección **registra su ingreso**. La lógica del sistema interpreta la **secuencia 1 → 2** como un ascenso e incrementa la cuenta de pasajeros a bordo y la de subidas en una unidad.

Asimismo, la última viñeta muestra cómo **el sistema de geolocalización monitorea la ubicación del vehículo y ayuda a que se reflejen los cambios detectados por el sistema de conteo** en la transición de una estación a otra, proporcionando información actualizada sobre la ruta en curso, las paradas anterior y próxima.

Si la activación ocurre en orden inverso, el sistema **interpreta que el pasajero está descendiendo** (Figura 5.18). Inicialmente no está activando ningún sensor. Al acercarse a la salida, **activa el infrarrojo 2**, más cercano al interior. Mientras **avanza hacia el exterior, activa el sensor 1**. Como en el ascenso, cuando el pasajero cruza completamente ambos sensores, deja de activar señales y el **sistema detecta la secuencia de activación 2 → 1** como **un descenso**. Automáticamente, se ajusta la cuenta de pasajeros a bordo: disminuye en una unidad, y se incrementa la cuenta de bajadas en el mismo valor.

Para evitar errores en la contabilización, debidos a usuarios que permanezcan en la zona de activación entre estaciones, y minimizar el consumo de energía y datos, el conteo de pasajeros solo se activa cuando la unidad se encuentra detenida en una estación. Esto se logra mediante la integración con el sistema de geolocalización, que permite identificar si el autobús está dentro de la geocerca de una parada.

Cuando el pumabús abandona la estación, el sistema de conteo de pasajeros envía una sola vez los datos de ascensos y descensos a través de Sigfox y posteriormente reinicia estos valores para la siguiente estación. Sin embargo, el número total de pasajeros a bordo se mantiene de principio a fin del recorrido para reflejar la ocupación de la unidad.

La distancia entre los sensores infrarrojos se determinó procurando un balance entre precisión y robustez en la detección de ascensos y descensos. Se buscó que la separación permitiera captar el movimiento de una persona cruzando la entrada sin interferencias de movimientos parciales o erróneas activaciones. Además, la distancia debía ser suficiente para que ambos sensores detectaran una diferencia clara en la secuencia de paso, minimizando la posibilidad de lecturas incorrectas por oscilaciones del usuario o presencia de objetos cercanos. Esta distancia fue probada empíricamente para asegurar un funcionamiento estable en condiciones reales de uso y el rango que mostró mejores resultados fue de 12 a 20 cm.

A continuación se representa el proceso de **detección del descenso de un pasajero por el sistema de conteo de pasajeros** y su resultado reflejado también gracias al monitoreo de la ubicación:

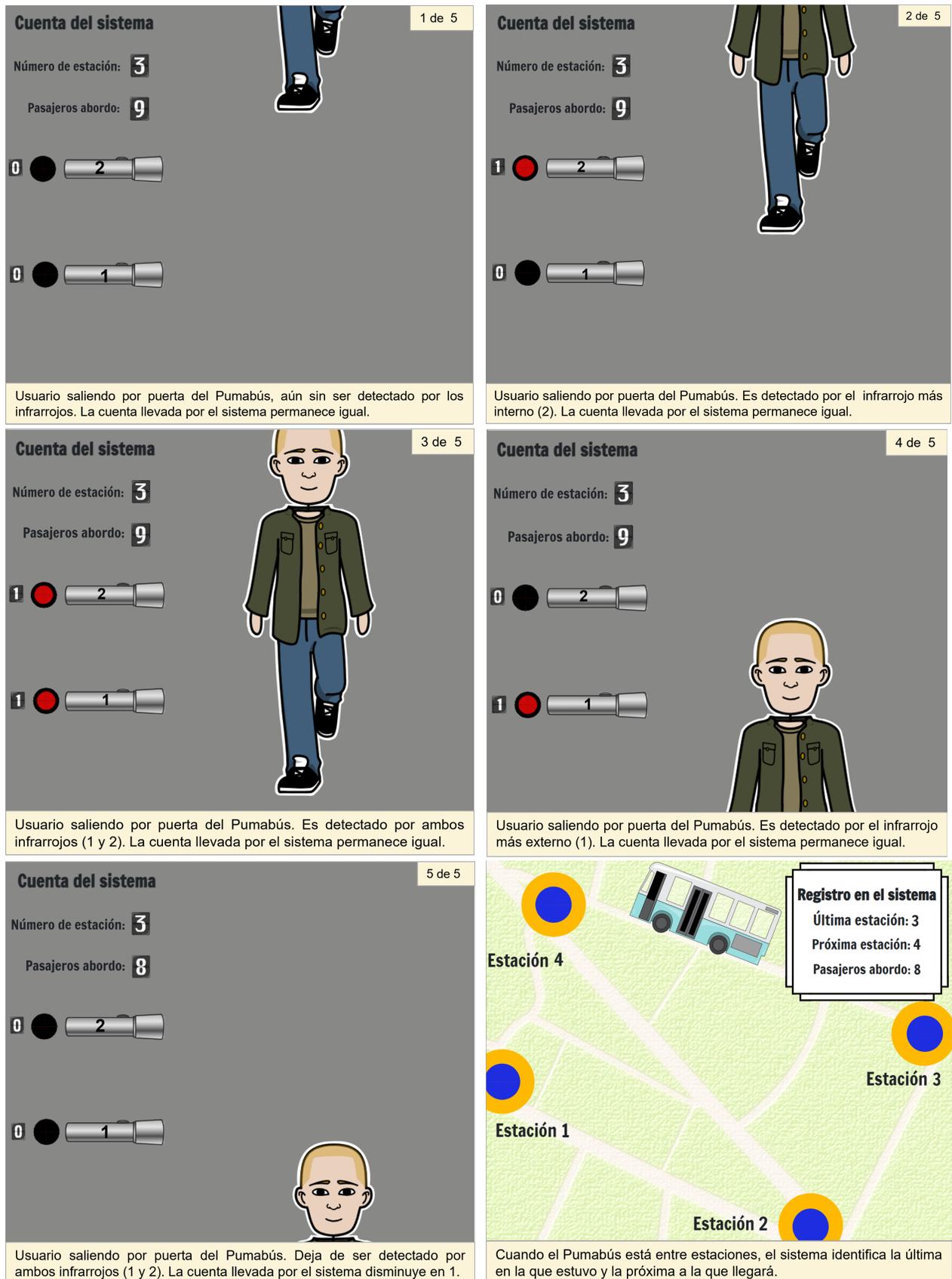


Figura 5.18: Secuencia de activación de infrarrojos a la bajada de un pasajero.

6 DESARROLLO DE CONCEPTO EN HARDWARE

Para construir la transición entre la etapa de desarrollo de concepto y el diseño a nivel hardware de la solución, se emplearon recursos como diagramas de caja negra o de entradas y salidas, que nos permiten comprender y visualizar la manera en que funciona el sistema general e incluso cómo interaccionan los diferentes subsistemas entre sí, sin necesidad de conocer los detalles internos de cada uno. Es decir, dejan ver qué hace un sistema, mas no cómo lo hace.

Estos diagramas ayudan a identificar y definir claramente las interfaces del sistema, especificando las entradas que recibe y las salidas que genera. Estas herramientas facilitan la elaboración de un diseño coherente y bien estructurado del proyecto de solución y dan pie a la presente etapa de diseño en el nivel de hardware. En esta sección sí interesa cómo es que los distintos sistemas funcionan para cumplir con las tareas deseadas y también qué elementos los conforman.

6.1 Componentes del hardware

El diseño del hardware contempla una selección de componentes que permiten cumplir con las funciones de sensado, geolocalización, procesamiento, comunicación y alimentación eléctrica. A continuación se describen los principales elementos que integran el prototipo del sistema:

- **Carcasa:** Contiene y protege en su interior los elementos electrónicos del sistema, resguardándolos de factores ambientales como humedad, polvo o impactos accidentales, así como manipulaciones indebidas. Su diseño considera la facilidad de montaje.
- **Sensores infrarrojos:** Permiten la detección del paso de usuarios para realizar el conteo de ascensos y descensos dentro del vehículo. Su colocación y orientación están optimizadas para minimizar lecturas erróneas y garantizar confiabilidad.
- **Módulo GPS:** Responsable de proporcionar la ubicación geográfica actualizada del vehículo, lo cual es esencial para el trazado de rutas y la activación de eventos mediante geocercas.
- **Batería portátil externa:** Fuente principal de alimentación del sistema, seleccionada por su capacidad, voltaje compatible y autonomía suficiente para mantener el sistema operativo durante trayectos enteros.
- **Cableado:** Conecta los distintos módulos y componentes. Se utilizan cables con conectores adecuados para facilitar el montaje y mantenimiento.
- **PCB (Placa de circuito impreso):** Substituye el uso protoboards en el armado final del sistema, lo que lo vuelve más ordenado y compacto. Fue diseñada considerando el

ruteo eficiente de pistas, la disposición lógica de componentes y la compatibilidad con el encapsulado de la carcasa.

- **Kit de desarrollo Devkit 2.0:** Es el núcleo de operación del sistema. Este microcontrolador ofrece capacidad de procesamiento, entrada/salida de datos y conectividad con la red IoT Sigfox.

La selección de cada elemento se basó en criterios de disponibilidad, costo, compatibilidad eléctrica y facilidad de integración. En la tabla 6.1 se muestran los diferentes componentes y materiales que estuvieron involucrados en el proceso de elaboración de los modelos funcionales en las distintas etapas y hasta el prototipo final del dispositivo, desde sensores hasta materiales estructurales para la fabricación de carcasas y soportes.

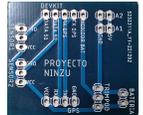
COMPONENTE	IMAGEN	CANTIDAD
SIGFOX DEVKIT 2.0		1
SENSOR E18-D80NK		2
Módulo GPS NEO-6MV2		1
PCB Diseñada		1
Trimpot de 10[KΩ]		1
Tira de header macho 40 pines		1
Cable USB a micro USB		1
Jumpers		24
Cartón reciclado/- Foamboard		-1 caja de teléfono/ ≈0.5 m ²
Abrazadera con tornillo y tuerca		2

Tabla 6.1: Componentes y materiales utilizados en el proyecto.

6.2 Esquemas eléctricos

El diagrama de la figura 6.1 representa la relación entre los principales componentes del sistema. La batería proporciona la alimentación eléctrica necesaria para el funcionamiento del Devkit, que es el núcleo del sistema. El Devkit se conecta a la PCB, la cual organiza y distribuye las conexiones de los sensores infrarrojos y otros módulos. Finalmente, los sensores capturan datos del entorno y los envían al sistema para su procesamiento y transmisión.

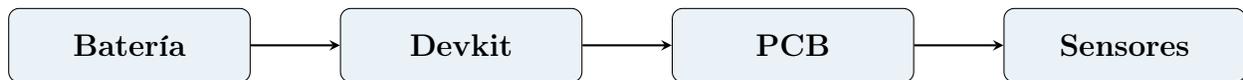


Figura 6.1: Esquema de flujo eléctrico del sistema.

6.3 Diseño de PCB

Para el diseño de la placa de circuito impreso (PCB), se consideró la integración del alambrado de los sensores, la alimentación y las conexiones necesarias hacia la placa Devkit 2.0. El diseño fue elaborado en el software Eagle. La placa se pensó para soldadura de headers macho en cada pin, por lo que en el esquemático se empleó la biblioteca correspondiente de headers. A cada grupo de conexiones se le asignó un nombre específico según su función, ya sea para sensores o alimentación.

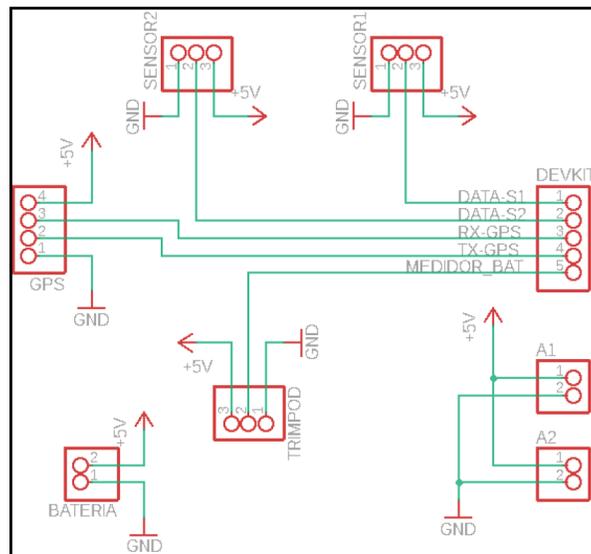


Figura 6.2: Diseño de PCB en Eagle.

Después, los grupos de pines se organizaron en función de sus conexiones con los sensores, la batería y la placa Devkit, con el objetivo de evitar cruces de cables y optimizar la disposición del circuito. Las pistas de conexión de la PCB se diseñaron con un grosor de 1 mm, y en la capa inferior se incorporó un plano de tierra para mejorar la estabilidad eléctrica. Además, las pistas en color rojo corresponden a la capa superior, mientras que las de color azul pertenecen a la capa inferior.

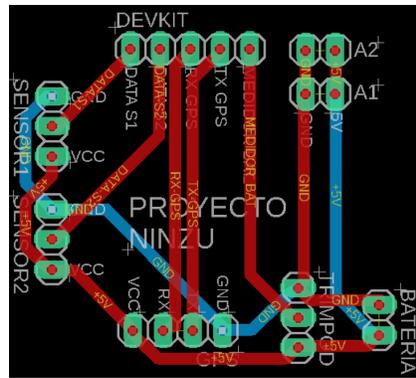
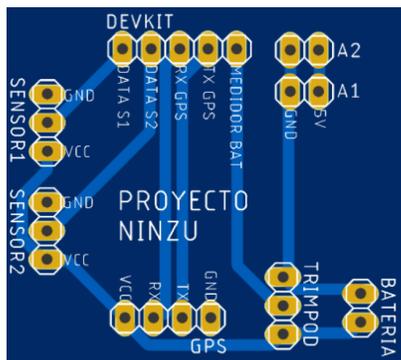
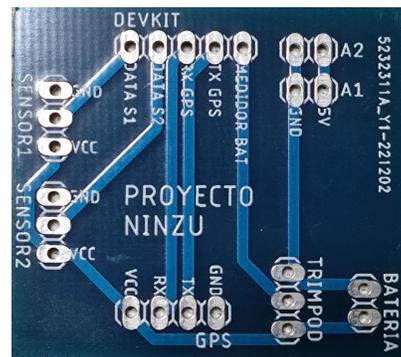


Figura 6.3: Vista de las pistas de conexión.



(a) Vista final de la PCB digital.



(b) Vista final de la PCB física.

Figura 6.4: Vista final de la PCB.

El diseño de la PCB fue desarrollado con el equipo original [18], como parte del proyecto realizado durante la asignatura *Temas selectos de ingeniería mecatrónica*, impartida en el semestre 2023-2.

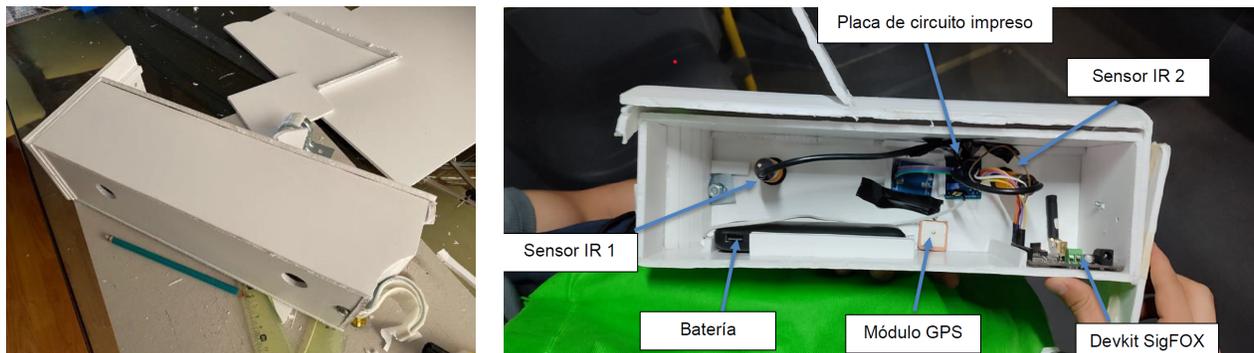
6.4 Evolución del diseño del dispositivo

El dispositivo ha pasado por una serie de cambios desde su concepción inicial para adaptarse a las necesidades específicas del sistema de transporte. Durante el proceso de desarrollo, se evaluaron distintas versiones del modelo funcional, lo que implicó ajustes en las dimensiones generales del dispositivo, en la distribución interna de sus componentes electrónicos y en la estructura física de la carcasa. Estos cambios respondieron tanto a criterios técnicos como de integración, buscando optimizar el aprovechamiento del espacio, facilitar el ensamblaje y mejorar la protección del sistema.

Durante el desarrollo del sistema, se consideraron también soluciones comerciales como modelos de caja estanca (IP65/IP67), las cuales ofrecen una protección adecuada frente a polvo y humedad. No obstante, su uso permitiría muy pocas modificaciones por cada pieza, de modo que volver a redimensionar y redistribuir componentes implicaría comprar más piezas cada vez y eso volvía restrictivo el proceso. Lo mismo se consideró sobre elegir acrílico o PVC cortado con láser, chapa metálica o cajas modulares.

Dado que uno de los objetivos del proyecto era conservar la flexibilidad en el diseño —especialmente para futuras iteraciones que pudieran variar en tamaño, geometría o disposición interna—, se optó por un enfoque que permitiera una mayor personalización. En este sentido, se priorizó el uso de carcasas diseñadas a medida con materiales reciclables o de fácil obtención, permitiendo optimizar el aprovechamiento del espacio, facilitar el montaje y mantenimiento, y mantener la funcionalidad del prototipo.

Inicialmente, las dimensiones del dispositivo fueron de 35 x 20 x 20 cm. Sin embargo, su manipulación durante la colocación y retiro resultaba incómoda y se sentía estorposa, por lo que se realizaron reducciones en las dimensiones y reacomodo de los componentes, gracias a que había espacio entre ellos.



(a) Elaboración con foamboard.

(b) Fijación interna de componentes.

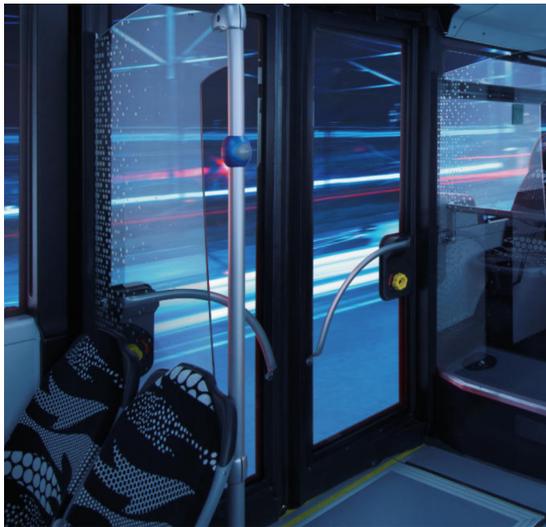
Figura 6.5: Fotografías del Dispositivo en elaboración y distribución interna de componentes.

La figura 6.6 se trata de dos fotos del dispositivo a bordo de una unidad Pumabús, una cuando está por colocarse y otra con el dispositivo ya en su lugar de funcionamiento; una foto del sistema de puertas de una unidad del mismo modelo que algunas unidades de Pumabús y sin gente; una foto de la parte trasera de los asientos adyacentes a la entrada o zona de ascenso y descenso, que significa una posibilidad para la ubicación del dispositivo en este modelo de autobús.



(a) Dispositivo retirado.

(b) Dispositivo colocado.



(c) Interior de las puertas.



(d) Espacio para posible colocación.

Figura 6.6: Interior de unidad Pumabús.

En otra de las últimas versiones, sus medidas eran ya de 26 x 9 x 7.5 cm (Figura 6.7) y se incluía una apertura para que sobresaliera la antena y fuera mínima la interferencia de la misma carcasa. Para todos los ajustes de dimensiones, se procuró y probó empíricamente que el dispositivo, con sus sujetadores y abrazaderas, realmente se mantuviera fijo en los pasamanos y soportes de las entradas del pumabús.



(a) Interior del dispositivo (1).



(b) Interior del dispositivo (2).



(c) Vista de la tapa.



(d) Vista de la abrazadera lateral.



(e) Sujetador removible.



(f) Carcasa con sujetador corto.



(g) Carcasa con sujetador largo.

Figura 6.7: Fotografías del dispositivo.

En la figura 6.7 se muestran fotos del dispositivo en diferentes posiciones, algunas con la abrazadera colocada en la posición inferior, ya sea con brazo largo o corto; la mayoría son del dispositivo con la tapa puesta, pero también hay dos en las que no la tiene colocada y deja ver los componentes internos y fijados a las paredes internas con cinta de aislar.

Además, se incorporó una mejora en el diseño que permite ajustar la distancia entre los sensores infrarrojos, de manera que puedan adaptarse a diferentes anchos de paso sin dejar espacios abiertos que expongan el interior del dispositivo. Esta característica ofrece mayor seguridad y versatilidad para su instalación.



Figura 6.8: Fotografías del Dispositivo. Distancia entre infrarrojos ajustable.



Figura 6.9: Fotografías del Dispositivo. Antena sobresaliente. Dimensiones reducidas.

Posteriormente, se hizo un ajuste final de dimensiones para reducir al mínimo el tamaño del dispositivo, a 22.4 x 5.5 x 7.5 cm, con los mismos componentes internos, distribuidos de la manera antes vista, manteniendo el uso de materiales reciclados.



Figura 6.10: Fotografías del Dispositivo. Antena sobresaliente. Dimensiones finales.

Por último, se tomaron las dimensiones finales para realizar un diseño asistido por computadora (CAD), en el software *Autodesk Inventor Professional 2026*, e imprimirlo en 3D. Esto permitió verificar tolerancias y asegurar el acomodo adecuado de los componentes internos. Se conservó la ranura para ajuste de distancia entre los sensores infrarrojos. Se incluyó un agujero para la colocación lateral de la abrazadera y rieles internos para deslizar la tapa.

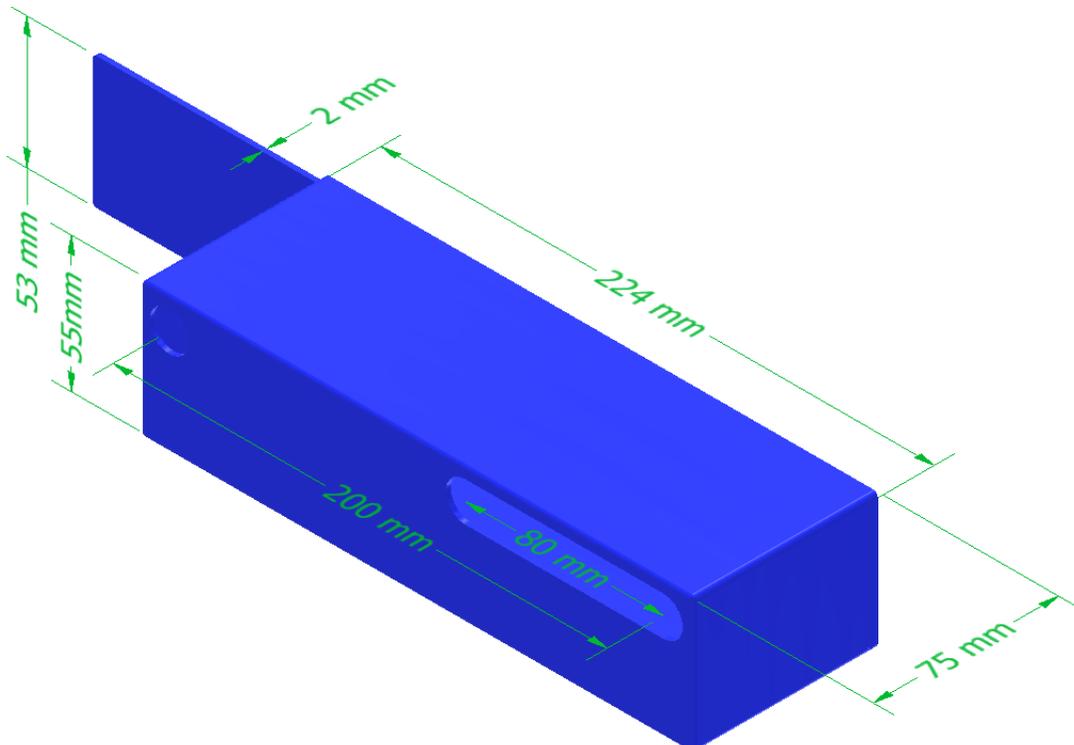


Figura 6.11: CAD de carcasa con dimensiones finales.

Para facilitar la comprensión del diseño de la carcasa, se presentan distintas vistas que permiten observar su forma y características desde diferentes ángulos. Se incluyen representaciones de las seis caras de la carcasa: vista superior, inferior, frontal, trasera y laterales (Figura 6.12).



(a) Vista frontal y lateral derecha sin tapa.



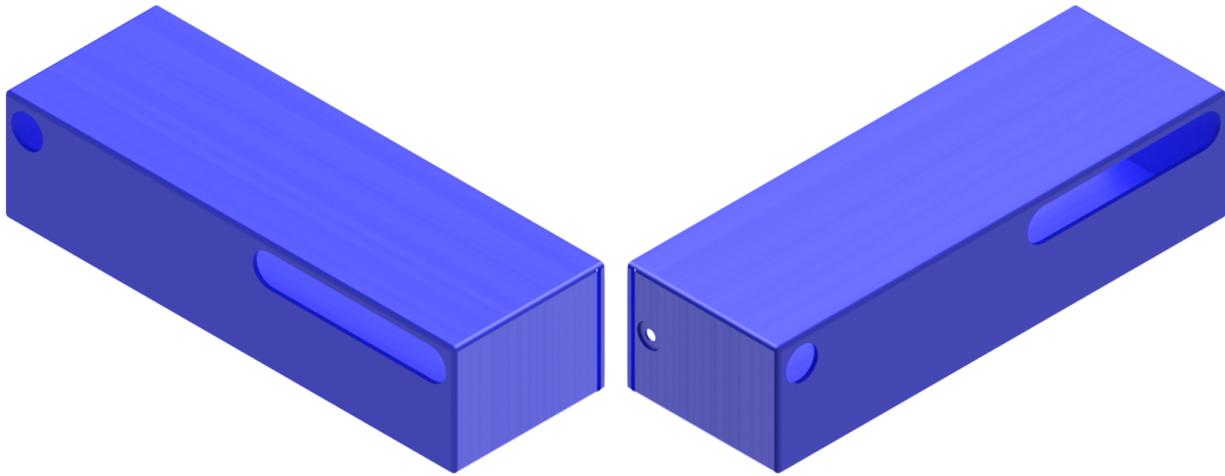
(b) Vista trasera con tapa y lateral izquierda con orificio para abrazadera.



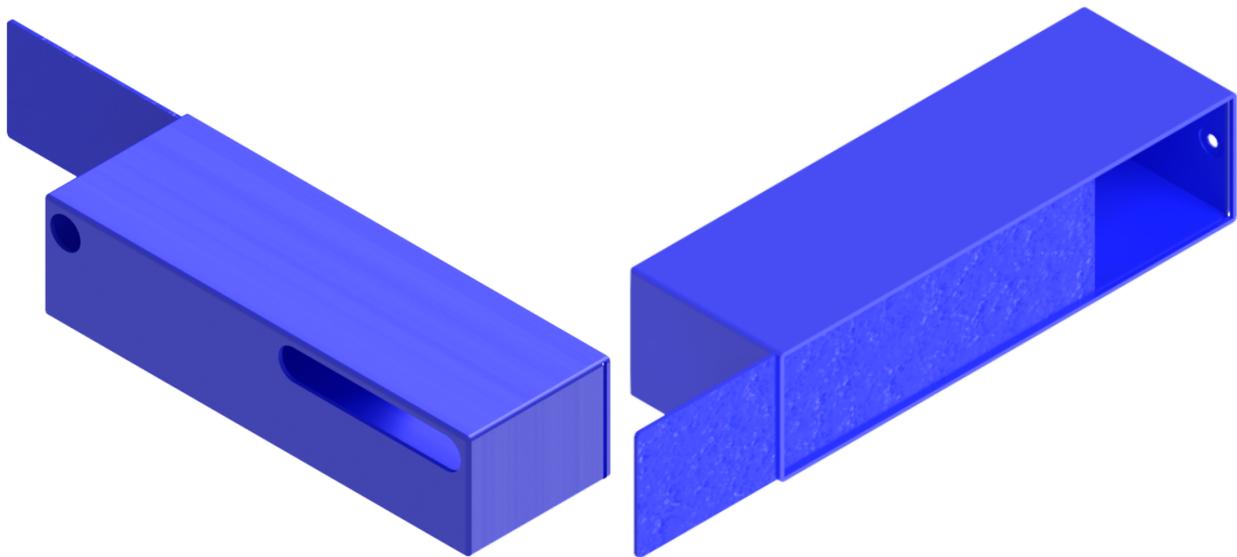
(c) Vista superior y vista inferior con ranura de montaje.

Figura 6.12: Vistas del prototipo desde las seis caras principales: frontal, trasera, superior, inferior, lateral derecha y lateral izquierda.

Además, se muestran vistas en perspectiva o isométricas, que permiten visualizar simultáneamente varias caras (Figura 6.14). Esto brinda una apreciación más completa de su volumen y geometría.



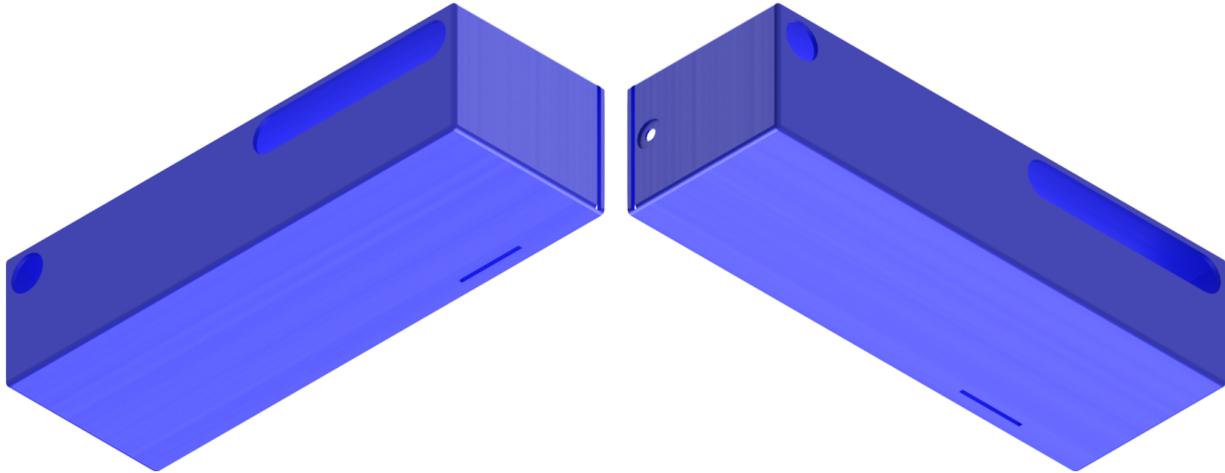
(a) Vistas isométricas desde las esquinas frontales superiores derecha e izquierda.



(b) Vista isométrica superior con tapa parcialmente deslizada sobre los rieles de la carcasa, mostrando frente (izquierda) y parte trasera (derecha).

Figura 6.13: Vistas isométricas superiores e inferiores del CAD de la carcasa.

En la Figura 6.14 (b) se observa que la carcasa completa se compone de un cuerpo principal y una tapa que se inserta en la parte trasera y se ajusta a un riel inferior y uno superior, sobre los que desliza para cubrir por completo el interior de la carcasa.



(c) Vistas isométricas inferiores, donde se observa el orificio para abrazadera y la ranura de montaje.

Figura 6.14: Vistas isométricas superiores e inferiores del CAD de la carcasa.

También se presenta, en la Figura 6.15, la vista de dibujo del plano técnico con las medidas del diseño final de la carcasa, así como una vista explosionada del ensamble, donde las piezas aparecen ligeramente separadas entre sí de manera intencionada para mostrar cómo se ensamblan entre ellas. Esta representación permite visualizar la disposición relativa de la tapa y la base, así como los puntos de acoplamiento considerados para facilitar el armado del prototipo.

Para facilitar su comprensión, el plano incluye distintas vistas ortogonales, que son representaciones desde ángulos específicos del objeto. En particular, se muestran:

- **Vista frontal**, que representa el objeto tal como se vería si se observa de frente.
- **Vista superior**, que muestra el objeto visto desde arriba.
- **Vista lateral**, que corresponde a uno de los lados del objeto (comúnmente el derecho).

Además, se incluye una **vista isométrica** explosionada a color, la cual ofrece una perspectiva tridimensional donde las piezas están separadas para ilustrar su posición relativa en el ensamblaje.

El prototipo fue fabricado mediante modelado por deposición fundida (impresión 3D), utilizando PLA (ácido poliláctico), un material termoplástico biodegradable. Este polímero es uno de los más comunes y económicos en procesos de fabricación aditiva. Pese a que el PLA presenta una resistencia mecánica y térmica menor en comparación con otros materiales como el PETG, ABS o Nylon, se usa frecuentemente para la fabricación de prototipos.

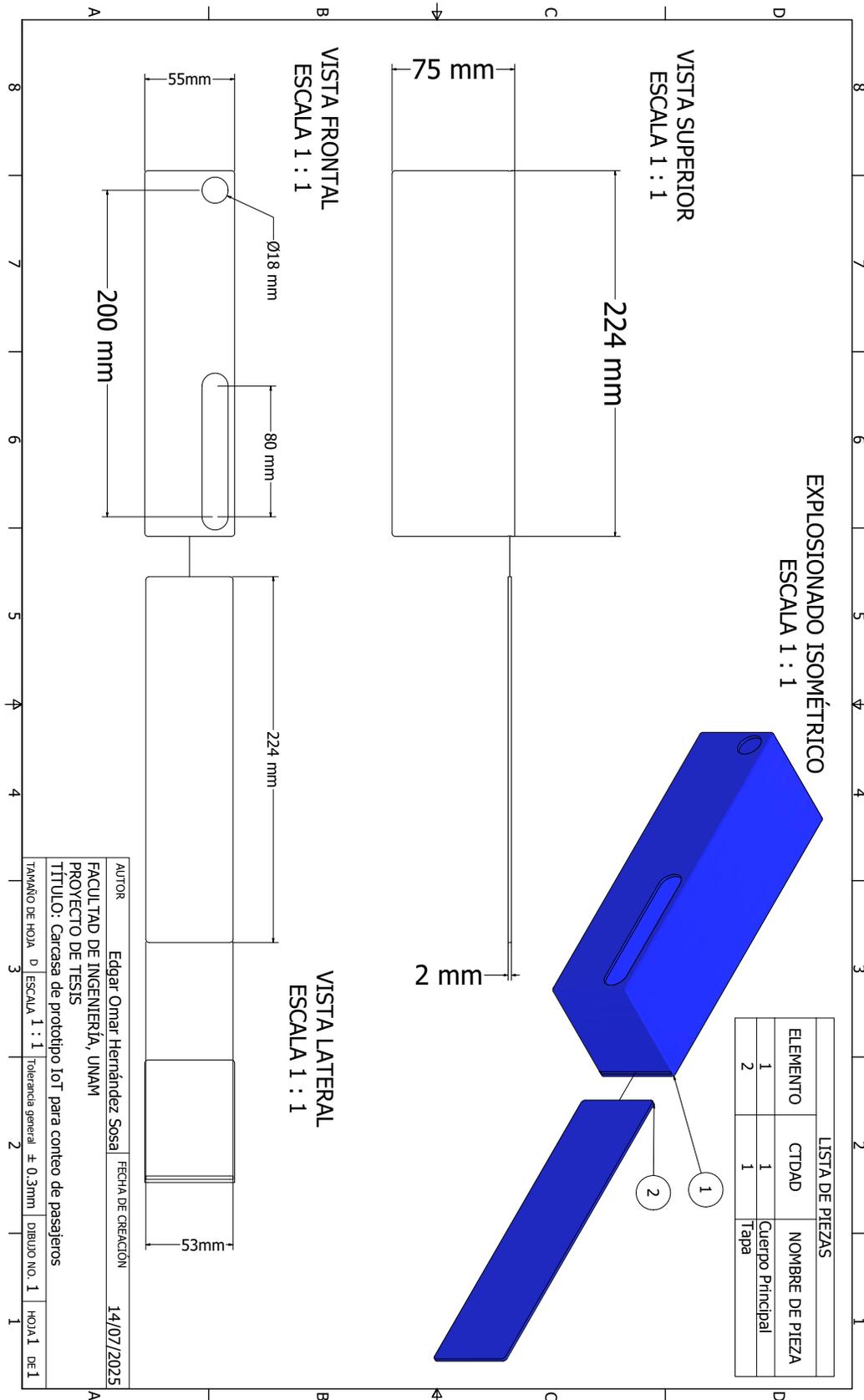


Figura 6.15: Planos y explosionado isométrico del CAD de la carcasa. Dimensiones finales.

La figura 6.16 muestra el dispositivo final ensamblado en la carcasa impresa en 3D, diseñada para alojar todos los componentes electrónicos del sistema de conteo y geolocalización. Esta versión incorpora los ajustes de tamaño y disposición interna adoptados tras las iteraciones previas para garantizar la protección de los componentes y facilitar su instalación a bordo del pumabús con el sistema de fijación mecánica por abrazadera.



Figura 6.16: Carcasa impresa en 3D con componentes ensamblados.

En la Figura 6.17 se muestra la vista trasera de la carcasa con la tapa colocada y en la Figura 6.18 se aprecia la vista trasera con los componentes ensamblados en su interior. En la Figura 6.19, se muestra el dispositivo montado en la estructura de la entrada de la unidad.



Figura 6.17: Carcasa impresa en 3D. Vista trasera con tapa puesta.



Figura 6.18: Carcasa impresa en 3D. Vista trasera de componentes internos.

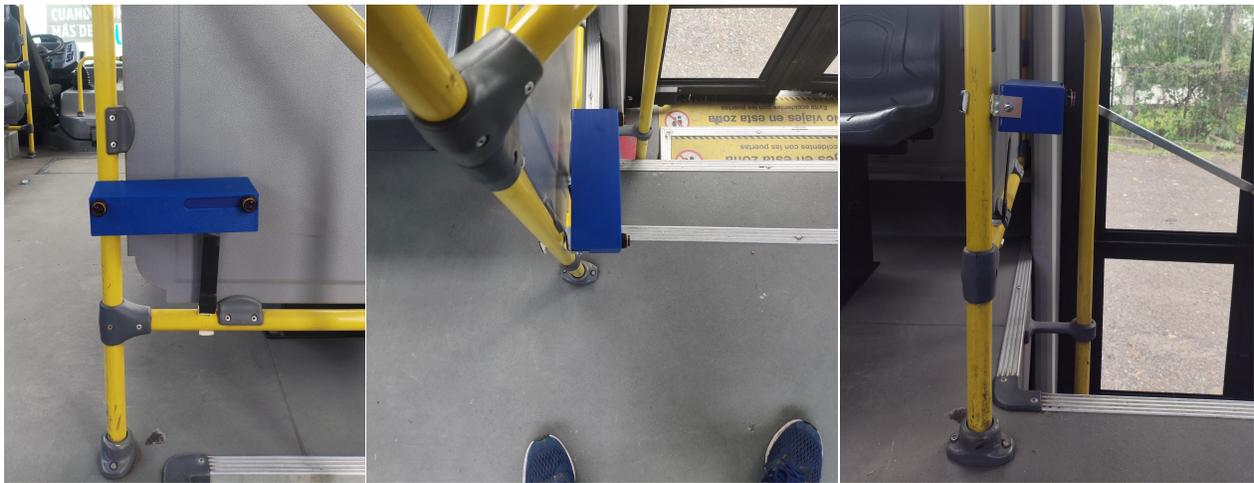


Figura 6.19: Carcasa impresa en 3D montada en unidad Pumabús.

Características físicas y electrónicas finales del dispositivo:

Alimentación	Material	Peso	Dimensiones
5V DC	PLA	500 g	22.4 × 5.5 × 7.5 cm

Sensores para conteo de pasajeros	Módulo de geolocalización
<p>Sensor de Proximidad Fotoeléctrico Infrarrojo E18-D80NK</p> <p>-Distancia entre sensores: 12 - 20 cm</p> <p>-Rango de detección: 0 – 90 cm</p>	<p>Módulo GPS NEO-6MV2</p> <p>Tiempo de conexión: Primera vez: Mínimo 5 min o Hasta casi 10 min.</p> <p>Segunda vez: Mínimo 40 segundos o Hasta 1 minuto</p>

Tabla 6.2: Características finales del dispositivo.

7 PROCESAMIENTO DE INFORMACIÓN (FIRMWARE) Y TRANSMISIÓN A TRAVÉS DE LA RED DE COMUNICACIÓN SIGFOX

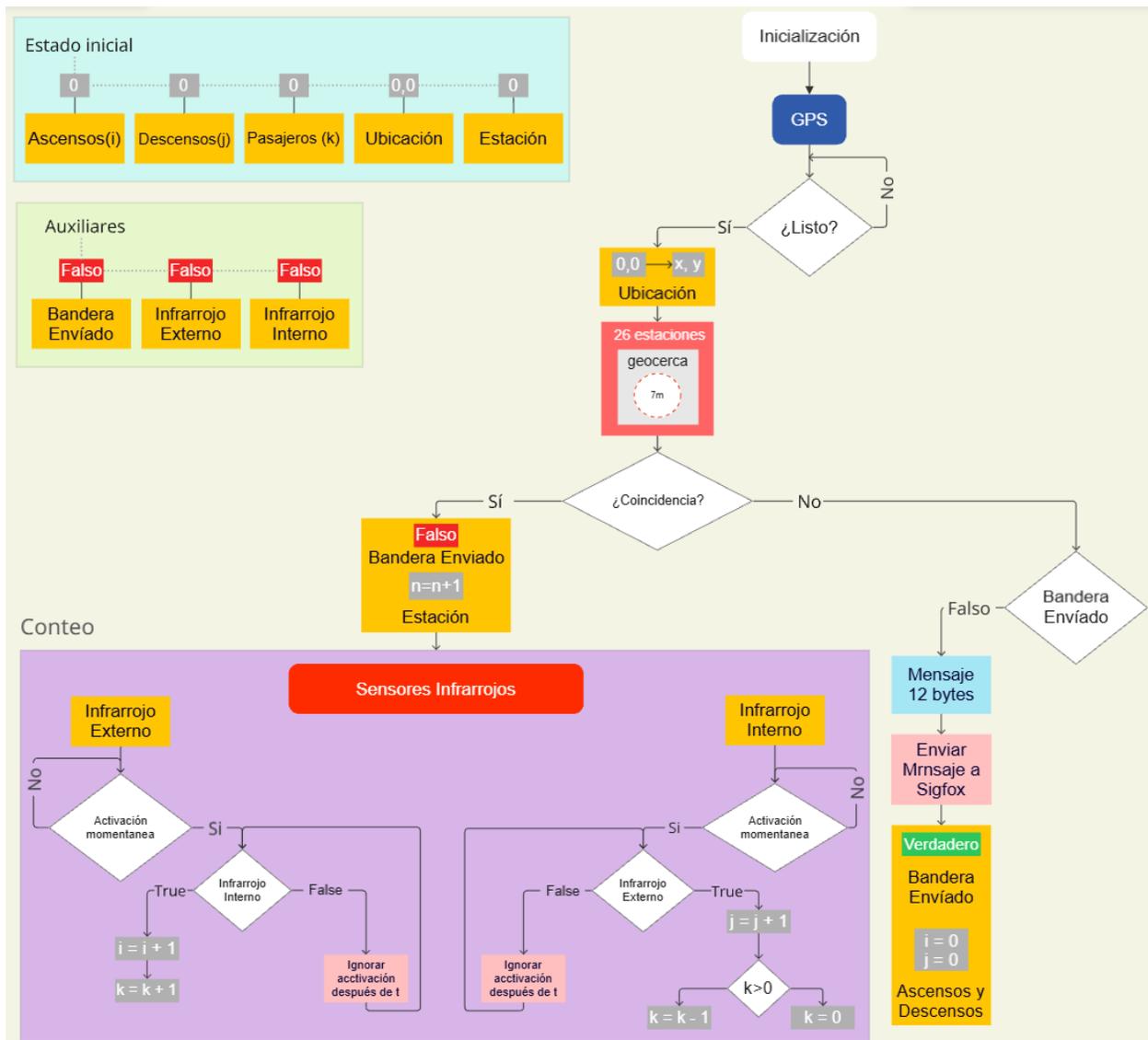


Figura 7.1: Diagrama de flujo del código del sistema.

El diagrama de flujo de la figura 7.1 describe la operación del sistema de conteo de pasajeros y geolocalización a implementar en el Pumabús, de acuerdo con la lógica del código elaborado. En la fase de inicialización, el sistema activa el módulo GPS para determinar la ubicación del vehículo y verificar si se encuentra dentro de la geocerca de alguna estación. Solo en ese caso se habilita el conteo de pasajeros mediante sensores infrarrojos, los cuales registran los

ascensos y descensos en función de la secuencia de activación de los sensores.

La información recolectada se almacena temporalmente y, al salir de la estación, se envía un paquete de datos a la plataforma de comunicación Sigfox. Finalmente, los valores de ascensos y descensos se reinician para la siguiente estación, mientras que el número total de pasajeros a bordo se mantiene actualizado.

A continuación, se incluye el código desarrollado en Arduino para el almacenamiento, filtrado, procesamiento, empaquetado y envío de datos a través de la red de comunicación IoT Sigfox.

```

1
2 //-----
3 // ***** BIBLIOTECAS *****
4 //-----
5
6 #include <TinyGPSPlus.h> // Biblioteca del mdulo GPS
7 #include <SoftwareSerial.h> // Biblioteca para la comunicacin serial
8 #include <AltSoftSerial.h> // Biblioteca para la lectura del mdulo GPS
9
10 //-----
11 // ***** VARIABLES GLOBALES *****
12 //-----
13
14 int conteo = 0; //Variable para el nmero de pasajeros a bordo
15 int sub = 0; //Variable para el nmero de personas que suben
16 int baj = 0; //Variable para el nmero de personas que bajan
17 int estacion = 0; //Variable para el nmero de estacin en la que se encuentra la
    Unidad
18 float latitude, longitude, distance; //Variables para almacenar la latitud, longitud
    y distancia
19
20 //-----
21 // ***** BANDERAS *****
22 //-----
23
24 bool banderaRecorrido = false; // Variable para indicar que el vehculo se
    encuentra haciendo su recorrido
25 bool banderaInicio = false; // Variable para enviar un mensaje cuando se encienda el
    dispositivo
26
27 //-----
28 // ***** CONSTANTES GLOBALES *****
29 //-----
30
31 const int numEstaciones = 26; // Variable para determinar cuntas estaciones que tiene
    la ruta 3
32 // Arreglo de coordenadas de latitud para las estaciones del pumabs
33 const float initialLatitude[26] = { 19.324535, 19.320570, 19.317781, 19.317236,
    19.315407, 19.314495, 19.310020, 19.309219, 19.311516, 19.311781, 19.310846,
    19.309822, 19.308081, 19.314003, 19.314102, 19.314131, 19.311541, 19.311996,
    19.314518, 19.315815, 19.317368, 19.316908, 19.317596, 19.318963, 19.320919,
    19.323685};
34 // Arreglo de coordenadas de lingitud para las estaciones del pumabs

```

```

35 const float initialLongitude[26] = { -99.174789, -99.176414, -99.176385, -99.180710,
    -99.186447, -99.186356, -99.186600, -99.185860, -99.185814, -99.184944,
    -99.182190, -99.180397, -99.180084, -99.179962, -99.181038, -99.181435,
    -99.182121, -99.184738, -99.185959, -99.186111, -99.183937, -99.180428,
    -99.176292, -99.175842, -99.176239, -99.174866};
36 const float maxDistance = 6; // Radio de la geocerca en metros
37 const bool ecoSerial = true; // Constante para mostrar informacin en el monitor
    serial
38 const int baudRate = 9600; // Constante para la comunicacin serial
39
40 //-----
41 // ***** PINES DE CONEXION *****
42 //-----
43
44 const int boton = 6; // Pin de entrada para el envio de datos (botn
    azul)
45 static const int LEFT = A1, RIGHT = A0; // Pines de lectura para los sensores
    infrarrojos
46 static const int RXPin = A3, TXPin = A2; //Definiendo pines para RX y TX
47
48 //-----
49 // ***** CADENAS DE BUFFER PARA PAYLOAD *****
50 //-----
51
52 String bufer; // Variable para almacenar el payload
53 String bufer2 = "\n"; // Salto de lnea para marcar el final del payload
54
55 //-----
56 // ***** OBJETOS PARA EL GPS *****
57 //-----
58
59 TinyGPSPlus gps; // Objeto GPS
60 SoftwareSerial ss(RXPin, TXPin); // Conexin Serial para el mdulo GPS
61
62 //-----
63 // ***** CONFIGURACION *****
64 //-----
65
66 void setup() {
67 //-----Configuracin de comunicacin serial-----//
68 Serial.begin(baudRate); // Conexin serial de Arduino
69 ss.begin(baudRate); // Conexin serial del mdulo GPS
70 //-----Declaracin del caracter de los pines-----//
71 pinMode(LEFT, INPUT); // Pin del sensor infrarrojo izquierdo
72 pinMode(RIGHT, INPUT); // Pin del sensor infrarrojo derecho
73 pinMode(TXPin, INPUT); // Pin TX del GPS
74 pinMode(RXPin, INPUT); // Pin RX del GPS
75 pinMode(boton, INPUT); // Pin del botn para envio de mensajes
76 pinMode(7, OUTPUT); // Habilitacin del mdulo Wisol del Devkit
77 }
78
79 //-----
80 // ***** FUNCION PRINCIPAL *****
81 //-----

```

```

82
83 void loop() {
84     creaBuffer();           // Llamada a la funcin para crear el buffer
85     while (ss.available() > 0) // Condicin para verificar la disponibilidad del mdulo
        GPS
86     {
87         if (gps.encode(ss.read())) // Condicin para verificar que el mdulo GPS est listo
            para enviar la posicin
88         {
89             coordenadasEstacion(); //Funcion que despliegar la ifnromacin del GPS
90         }
91     }
92     if (millis() > 5000 && gps.charsProcessed() < 10) // Si pasa cierto tiempo sin
        actividad del gps
93     {
94         Serial.println(F("No GPS detected: check wiring.")); // Mostrar mensaje de no
            deteccin
95         while (true);
96     }
97     conteoPersonas(); // Funcin para el conteo de personas que suben y bajan de la
        unidad
98 }
99
100 //-----
101 // ***** FUNCIONES PARA EL ENVO A SIGFOX *****
102 //-----
103
104 void creaBuffer() {
105     // Envio de mensaje cuando se enciende el dispositivo y cuando se activa el botn
        azul
106     if ((distance > maxDistance && banderaInicio == false) || digitalRead(boton) == LOW
        ) {
107         //-----
108         //AT$SF= comando para mandar la informacion por sigfox
109         bufer = "AT$SF="; // Mximo 12 bytes
110         //-----
111         add_int(sub); // Se agrega 1 byte al payload
112         //-----
113         add_int(baj); // Se agrega 1 byte al payload
114         //-----
115         add_int(conteo); // Se agrega 1 byte al payload
116         //-----
117         add_float(latitude); // Se agregan 4 bytes al payload
118         //-----
119         add_float(longitude); // Se agregan 4 bytes al payload
120         //-----
121         add_int(estacion); // Se agrega 1 byte al payload
122         //-----
123         send_message(bufer); // Funcin para el envo de informacin
124         banderaInicio = true; // Se activa la bandera de inicio para saber que ya se envi
            un mensaje
125         sub = 0;           // Reinicio de variable de subidas a 0
126         baj = 0;          // Reinicio de variable de bajadas a 0
127     }

```

```

128 }
129
130 // Funcion para agregar enteros menores a 256 al payload
131 void add_int(int var2) {
132     byte* a2 = (byte*)&var2; // Conversin del dato a bytes
133     String str2;             // Declaracin de cadena para almacenar el entero
134     str2 = String(a2[0], HEX); // Conversin del valor hex a string
135     if (str2.length() < 2)    // Comprobacin de la longitud del byte
136     {
137         bufer += 0 + str2; // Si no esta completo, se agrega un cero
138     } else {
139         bufer += str2; // Si esta completo, se copia tal cual
140     }
141 }
142
143 // Funcion para agregar flotantes al payload
144 void add_float(float var1) {
145     byte* a1 = (byte*)&var1; // Se convierte el dato a bytes
146     String str1;
147     // Se agrega al comando AT$SF= la informacin a enviar
148     for (int i = 0; i < 4; i++) // Se toman 4 bytes para el flotante
149     {
150         str1 = String(a1[i], HEX); // Se convierte el valor hex a string
151         if (str1.length() < 2)    // Verificacin de la longitud del flotante
152         {
153             bufer += 0 + str1; // Si no est completo, se agrega un cero
154         } else {
155             bufer += str1; // Si est completo, se copia tal cual
156         }
157     }
158 }
159
160 // Funcion para enviar el mensaje con el payload
161 void send_message(String payload) {
162     bufer += bufer2; // Se aade el salto de linea "\n"
163     digitalWrite(7, HIGH); // Se habilita el modulo Sigfox
164     delay(1000); // Tiempo de espera para completar la tarea
165     Serial.print("AT$RC\n"); // Reset del canal para asegurar que manda en la
        frecuencia correcta
166     Serial.print(bufer); // Se enva la informacion por sigfox
167     delay(3000); // Tiempo de espera para completar la tarea
168     digitalWrite(7, LOW); // Se deshabilita el modulo Sigfox
169 }
170
171 //-----
172 // ***** FUNCIONES DEL MDULO GPS *****
173 //-----
174
175 void coordenadasEstacion() // Funcin para despelegar la informacin del GPS y el nmero
        de estacin
176 {
177     banderaRecorrido = false; // Variable para indicar que el vehculo se encuentra
        dentro de la geocerca
178     Serial.println(F("Ubicacin: ")); // Impresin en el monitor serial

```

```

179  if (gps.location.isValid())      // Comprobacin de la validez de la posicin
180  {
181  latitude = (float)gps.location.lat(); //Obteniendo latitud actual
182  Serial.print(F("Latitud: "));      // Impresin en el monitor serial
183  Serial.print(gps.location.lat(), 6); //Imprimiendo latitud actual
184  Serial.print(F(", "));             // Impresin en el monitor serial
185  longitude = (float)gps.location.lng(); //Obteniendo longitud actual
186  Serial.print(F("Longitud: "));     // Impresin en el monitor serial
187  Serial.println(gps.location.lng(), 6); //Imprimiendo longitud actual
188  for (int i = 0; i < numEstaciones; i++) // Ciclo para verificar todas las
      estaciones
189  {
190  // Funcion para calcular la distancia de las coordenadas actuales a la estacin i
      de la ruta 3
191  distance = getDistance(latitude, longitude, initialLatitude[i], initialLongitude
      [i]);
192  if (distance < maxDistance) // Condicion para saber si se est dentro de una
      geocerca
193  {
194  estacion = i + 1;                  // Se almacena el nmero de
      la estacin
195  Serial.print("El Pumabus se encuentra en la parada "); // Impresin en el
      monitor serial
196  Serial.println(estacion);         // Se despliega el nmero de
      estacin
197  banderaRecorrido = true;          // Se establece la bandera
      de recorrido como activa
198  banderaInicio = false;            // Se establece la bandera
      de inicio como falsa
199  i = numEstaciones;                // Se incrementa i para
      detener el ciclo
200  break;                            // Ruptura de la iteracin
201  }
202  }
203
204  if (banderaRecorrido == false) // Condicin para verificar que el pumabs no este
      en su recorrido
205  {
206  if (estacion == 0) // Si la estacin es 0, an no llega a ninguna estacin
207  {
208  Serial.println(F("El pumabus se encuentra todava no inicia el recorrido"));
      // Impresin en el monitor serial
209  } else if (estacion == numEstaciones) // Si la estacin es la mxima posible, ya
      termin su recorrido
210  {
211  Serial.println(F("El pumabus termin su recorrido")); // Impresin en el
      monitor serial
212  }
213  else // Si la estacin no es 0 ni la ltima, se encuentra entre dos estaciones
214  {
215  Serial.print("El pumabus se encuentra entre las estaciones "); // Impresin
      en el monitor serial
216  Serial.print(estacion); // Se muestra el nmero de la ltima estacin que se
      detect

```

```

217     Serial.print(" y ");          // Impresin en el monitor serial
218     Serial.println(estacion + 1); // Se muestra el nmero de la siguien estacin
219 }
220 }
221 } else // Si la posicin del GPS no es vlida, se muestra
222 {
223     Serial.print(F("INVALID")); // Impresin en el monitor serial
224     latitude = 0;                // Se restablecen los valores de latitud a 0
225     longitude = 0;              // Se restablecen los valores de longitud a 0
226 }
227 Serial.println(); // Salto de lnea
228 }
229 // Funcin para calcular la distancia de las coordenadas actuales a la estacin
230 float getDistance(float flat1, float flon1, float flat2, float flon2) { //Formula de
    Haversine para calcular distancia entre dos coordenadas GPS
231 // Variables
232 float dist_calc = 0; // Se inicializa con valor nulo (0)
233 float dist_calc2 = 0; // Se inicializa con valor nulo (0)
234 float diflat = 0;    // Se inicializa con valor nulo (0)
235 float diflon = 0;    // Se inicializa con valor nulo (0)
236
237
238 // Calculos
239 diflat = radians(flat2 - flat1); // Se calcula el ngulo entre las dos coordenadas
    de latitud
240 flat1 = radians(flat1);          // Se recalcula el valor de la coordenada de
    latitud 1 en radianes
241 flat2 = radians(flat2);          // Se recalcula el valor de la coordenada de
    latitud 2 en radianes
242 diflon = radians((flon2) - (flon1)); // Se calcula la distancia plana entre las dos
    coordenadas de longitud
243
244 //Distancias
245 dist_calc = (sin(diflat / 2.0) * sin(diflat / 2.0)); // Funciones de seno para
    calcular la distancia geoide de latitud
246 dist_calc2 = cos(flat1);          // Funciones de coseno
    para calcular la distancia geoide de latitud 1
247 dist_calc2 *= cos(flat2);          // Funciones de coseno
    para calcular la distancia geoide de latitud 2
248 dist_calc2 *= sin(diflon / 2.0);  // Funciones de coseno
    para calcular la distancia geoide de longitud
249 dist_calc2 *= sin(diflon / 2.0);  // Funciones de coseno
    para calcular la distancia geoide de longitud
250 dist_calc += dist_calc2;          // Suma de las
    distancias 1 y 2
251 dist_calc = (2 * atan2(sqrt(dist_calc), sqrt(1.0 - dist_calc))); // Clculo final de
    la distancia
252 dist_calc *= 6371000.0;          // Conversin a metros
253 return dist_calc;                // Retorno de la
    variable de distancia
254 }
255
256 //-----
257 // ***** FUNCIN DE CONTEO DE PERSONAS *****

```

```

258 //-----
259
260 void conteoPersonas() {
261     if (banderaRecorrido = true) // Se asegura de realizar conteo solamente en las
        estaciones
262     {
263         int L = digitalRead(LEFT); // Asignacin a L de valores ledos en ADC0
264         int R = digitalRead(RIGHT); // Asignacin a R de valores ledos en ADC1
265         if (L == 0) // Verificacin de deteccin en el exterior
266         {
267             delay(80); //Tiempo de estabilizacin para que se cumpla la
                tarea
268             for (int i = 0; i < 200; i++) // Ciclo para que el sensor interior detecte
269             {
270                 delay(1); //Tiempo de estabilizacin para que la tarea se cumpla
271                 R = digitalRead(RIGHT); // Lectura del sensor interior
272                 if (R == 0) // Verificacin de deteccin en el interior
273                 {
274                     delay(80); // Tiempo de estabilizacin para que se cumpla la
                            tarea
275                     for (int j = 0; j < 200; j++) // Ciclo para la verificacin de la subida
276                     {
277                         delay(1); // Tiempo de estabilizacin para que la tarea se
                            cumpla
278                         R = digitalRead(RIGHT); // Lectura del sensor interior
279                         if (R == 1) // Verificacin de deteccin en el interior
280                         {
281                             conteo++; // Incremento del nmero de personas en el
                                    camin
282                             sub++; // Incremento del nmero de subidas en la
                                    estacin
283                             Serial.println("Pasajero sube"); // Impresin en el monitor serial
284                             delay(80); //Tiempo de estabilizacin para que la
                                    tarea se cumpla
285                             Serial.println(conteo); // Se muestra el nmero de personas en la
                                    unidad
286                             Serial.print("Sube: "); // Impresin en el monitor serial
287                             Serial.println(sub); // Se muestra el nmero de subidas en esa
                                    estacin
288                             Serial.println("Esperando..."); // Impresin en el monitor serial
289                             break; // Fuera del ciclo
290                         }
291                     }
292                 }
293             }
294         } else if (R == 0) // Verificacin de deteccin en el interior
295         {
296             delay(80); // Tiempo de estabilizacin para que se cumpla la
                tarea
297             for (int k = 0; k < 200; k++) // Ciclo para que el sensor exterior detecte
298             {
299                 delay(1); // Tiempo de estabilizacin para que se cumpla la tarea
300                 L = digitalRead(LEFT); // Lectura del sensor exterior
301                 if (L == 0) // Verificacin de deteccin en el exterior

```

```

302     {
303     delay(80);           // Tiempo de estabilizacin para que se cumpla la
                          tarea
304     for (int l = 0; l < 200; l++) // Ciclo para la verificacin de la bajada
305     {
306     delay(1);           // Tiempo de estabilizacin para que se cumpla la
                          tarea
307     L = digitalRead(LEFT); // Lectura del sensor exterior
308     if (L == 1)         // Verificacin de deteccin en el exterior
309     {
310     if (conteo > 0) // Comprobacin del nmero de pasajeros
311     {
312     conteo--; // Si es positivo, se disminuye
313     }
314     Serial.println("Pasajero baja"); // Impresin en el monitor serial
315     baj++;           // Incremento del nmero de bajadas en la
                          estacin
316     delay(80);       // Tiempo de estabilizacin para que se
                          cumpla la tarea
317     Serial.println(conteo); // Se muestra el nmero de personas en la
                          unidad
318     Serial.print("Baja: "); // Impresin en el monitor serial
319     Serial.println(baj); // Se muestra el nmero de bajadas en esa
                          estacin
320     Serial.println("Esperando..."); // Impresin en el monitor serial
321     break;           // Fuera del ciclo
322     }
323     }
324     }
325     }
326     }
327     delay(40); //Tiempo de estabilizacin para que se cumpla la tarea
328     }
329     }

```

Listing 7.1: Código en arduino del sistema de conteo y geolocalización.

7.1 Creación del paquete de datos para envío a través de la red de comunicación Sigfox

Los dispositivos compatibles con Sigfox utilizan el comando `AT$SF=` para enviar datos a la red. Este comando recibe una cadena hexadecimal que representa el paquete de datos a transmitir, con un límite de 12 bytes (24 caracteres hexadecimales) por mensaje.

La estructura del mensaje considera esta restricción del protocolo Sigfox, que permite únicamente 12 bytes por envío ascendente. Esta limitación obliga a empaquetar y codificar la información en formato binario mediante funciones que convierten enteros y flotantes a cadenas hexadecimales. El mensaje se ha optimizado para incluir toda la información esencial del sistema —conteo de pasajeros, coordenadas GPS y estación— en un solo envío, maximizando así la eficiencia sin rebasar el límite establecido por el protocolo.

Se construye en la función `creaBuffer()`. Este proceso consiste en convertir cada valor (int o float) a bytes individuales, y luego a una cadena en hexadecimal que se concatena al comando `AT$SF=`.

Esto se hace en funciones auxiliares:

- `add_int(int var)`: Convierte un entero de 1 byte (hasta 255) a su representación en hexadecimal y lo agrega al string del payload.
- `add_float(float var)`: Convierte un número flotante en sus 4 bytes equivalentes y los agrega uno a uno al string en hexadecimal. Esto asegura que, aunque el número tenga decimales, se codifique correctamente y de forma compacta.

Un ejemplo del resultado final del paquete es la cadena:

`AT$SF = 0102033F8A9FBD40B4C0001A`, donde cada par de caracteres representa un byte en hexadecimal. La distribución de bytes del paquete de datos para mensajes que se envían a través de la red Sigfox se muestra en la tabla 7.1.

Campo	Tipo de dato	Tamaño (Bytes)
Personas que suben (sub)	int	1
Personas que bajan (baj)	int	1
Conteo total (conteo)	int	1
Latitud (latitude)	float	4
Longitud (longitude)	float	4
Estación (estacion)	int	1
Total	-	12 bytes

Tabla 7.1: Estructura del paquete de datos (payload) enviado a través de la red Sigfox

ESTRUCTURA DE PAQUETE DE DATOS				
BYTE	EJEMPLO HEXADECIMAL	EJEMPLO DECIMAL	DESCRIPCIÓN	TIPO DE DATO
0	0A	10	SUBIDAS	ENTERO
1	6	6	BAJADAS	ENTERO
2	2A	42	CONTEO	ENTERO
3	41	19.320563	LATITUD	FLOTANTE
4	9A			
5	90			
6	83			
7	C2	-99.176384	LONGITUD	FLOTANTE
8	C6			
9	5A			
10	4F			
11	5D	15	ESTACIÓN	ENTERO

Tabla 7.2: Ej. de valores contenidos por byte en la estructura del paquete de datos construido.

La estructura del mensaje binario enviada mediante el comando AT\$SF= se alinea con la configuración personalizada de decodificación establecida en el Backend de Sigfox, cuyo campo es “Custom payload config”, de modo que la correspondencia de valores entre el buffer del código cargado en el Devkit y los datos manejados en Sigfox resultan como se aprecia en la tabla 7.3, donde se utilizan tipos de datos estándar como uint (entero sin signo) y float (punto flotante de 32 bits). El sufijo :little-endian indica que los bytes del número se ordenan del menos al más significativo, conforme al formato little-endian.

Campo	Tipo de dato	Tamaño (bits)	Posición de byte(s)	Descripción
sub	uint:8	8	0	Cantidad de ascensos
baj	uint:8	8	1	Cantidad de descensos
conteo	uint:8	8	2	Pasajeros a bordo
latitude	float:32:LE	32	3-6	Latitud (little-endian)
longitude	float:32:LE	32	7-10	Longitud (little-endian)
estacion	uint:8	8	11	Identificador de estación

Tabla 7.3: Datos manejados en el backend de Sigfox

Esta decodificación permite al backend interpretar correctamente cada campo del paquete y mapearlo a un formato estructurado en JSON, como se muestra en la estructura del

paquete de la figura 7.2, donde los valores decodificados se asocian con etiquetas clave (subidas, bajadas, pasajeros, position, estacion, hora) que luego pueden ser utilizados por otras plataformas o servicios conectados.

Device type DEVKITS - Callback edition

Callbacks

Type **DATA** **UPLINK**

Channel **URL**

Custom payload config **sub::uint:8 baj::uint:8 conteo::uint:8 latitude::float:32:little-endian longitude::float:32:lit**

URL syntax: `http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...`
Available variables: `device, time, station, data, rssi, seqNumber, deviceId`
Custom variables: `customData#sub, customData#baj, customData#conteo, customData#latitude, customData#longitude, customData#estacion`

Url pattern **https://triggers.losant.com/webhooks/pBon4lRtnKS0vwwjFv1_e4N2zTZQssGDas-Clv**

Use HTTP Method **POST**

Send SNI (Server Name Indication) for SSL/TLS connections

Headers header value

Content type **application/json**

Body

```
{
  "subidas": {"value": "{customData#sub}"},
  "bajadas": {"value": "{customData#baj}"},
  "pasajeros": {"value": "{customData#conteo}"},
  "position": {"value": 1, "context": {"lat": "{customData#latitude}", "lng": "{customData#longitude}"},
  "estacion": {"value": "{customData#estacion}"},
  "hora": {"value": "{time}"}
```

Figura 7.2: Callback en Sigfox para Losant.

8 DESARROLLO A NIVEL SOFTWARE

8.1 Aplicativo de servicios de software en la nube: Ubidots

Ubidots es una plataforma en la nube diseñada específicamente para proyectos de Internet de las Cosas (IoT). Facilita la recopilación, visualización y análisis de datos en tiempo real provenientes de dispositivos conectados, como sensores y sistemas de monitoreo. Permite a los usuarios integrar, almacenar y procesar grandes volúmenes de datos de diferentes dispositivos IoT, generando paneles, tableros o dashboards, alertas y reportes personalizados, lo que facilita la toma de decisiones basadas en los datos. Representa una solución para crear aplicaciones IoT de manera rápida y eficiente.

La configuración inicial de la plataforma Ubidots se realizó con el equipo original [18], como parte del proyecto realizado durante la asignatura *Temas selectos de ingeniería mecánica*, impartida en el semestre 2023-2.

Variables para el procesamiento de información

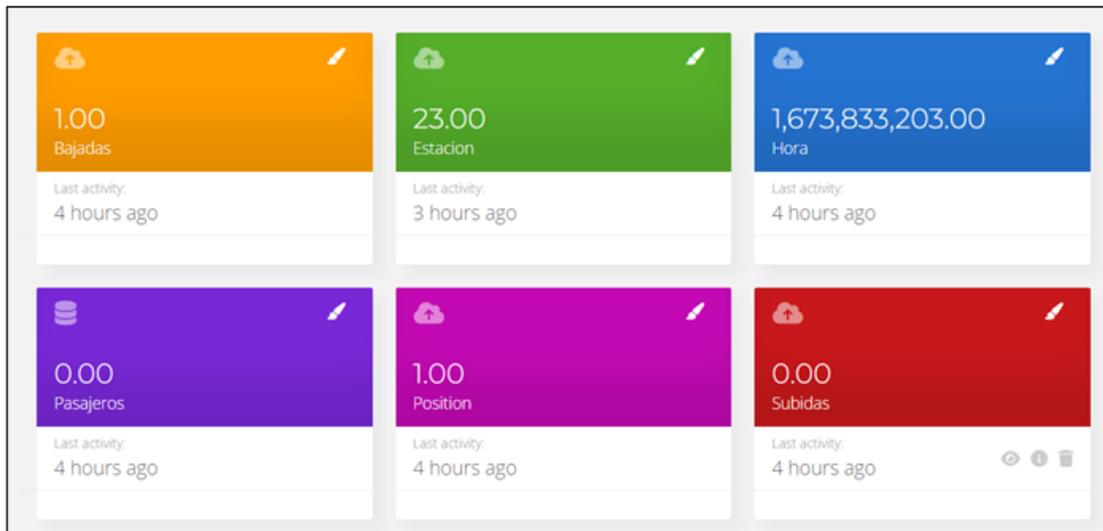


Figura 8.1: Variables en Ubidots.

Dentro del dispositivo se establecieron las siguientes variables:

- **Subidas:** Número de personas que ascendieron al vehículo en la última estación.
- **Bajadas:** Número de personas que descendieron del vehículo en la última estación.
- **Estación:** Número de la última estación por la que pasó el pumabús. Todas las paradas de la ruta están numeradas.

- **Hora:** Fecha y hora que permite comprobar la sincronización de las plataformas.
- **Pasajeros:** Número de personas que se encuentran dentro de la unidad.
- **Position:** Coordenadas de latitud y longitud para conocer la localización del pumabús.

Descripción de Experiencia de usuario

Para procurar una experiencia de usuario satisfactoria y que los usuarios decidan seguir utilizando el sistema desarrollado, de manera recurrente, el diseño de la interfaz debe cumplir con las siguientes características:

- **Útil:** La interfaz del sistema debe facilitar al usuario la información que necesita para satisfacer su necesidad. La gran variedad de elementos gráficos, íconos o widgets que ofrece Ubidots, incluso en su versión gratuita, permiten desplegar al usuario información en una gran variedad de formas, de manera que habrá alguna que se adapte a la aplicación específica.

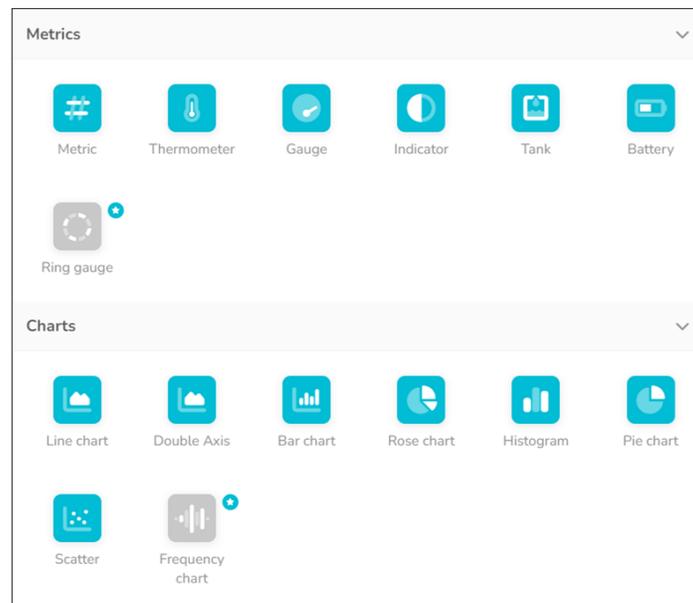


Figura 8.2: Íconos o widgets.

La herramienta más útil para adecuar los widgets a tus proyectos es HTML Canvas, que permite usar HTML para crear tu propio widget y modificarlo según las necesidades del proyecto.

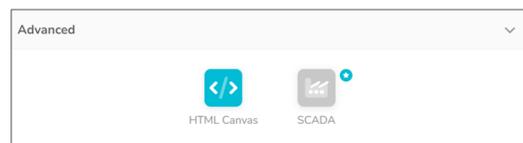


Figura 8.3: Opción para desarrollo con HTML en Ubidots.

- **Utilizable:** la interfaz debe ser fácil de usar e intuitiva para el usuario, por eso se recomienda colocar los widgets alineados de manera vertical para que el dashboard pueda recorrerse con facilidad en un solo sentido. La cabeceras de los widgets deben ser cortas y claras para que el usuario pueda identificar rápidamente cuál es la información que muestra y para qué sirve.
- **Deseable:** las sensaciones de la interfaz deben ser positivas para que el usuario tenga la misma percepción del proyecto. Esto está enfocado a la Interfaz de Usuario, pero también se debe tener en cuenta para la experiencia de usuario. Se recomienda usar imágenes y una paleta de colores que mantenga armonía en toda la interfaz, de manera que el usuario pueda sentir al proyecto como un todo, bien integrado y congruente.
- **Accesible:** el contenido de la plataforma debe ser visible para los usuarios y de fácil acceso para personas con discapacidades. Para esto se deben usar colores que no sean demasiado brillantes ni demasiado oscuros, tamaños de letra moderados y tipografías legibles.



Figura 8.4: Vista de tablero de desarrollador.

- **Creíble:** la información que se presenta debe ser confiable y transparente para que el usuario pueda confiar en ella. Para esto, se puede acceder al tablero o panel de control de desarrollador, donde estarán visibles los datos crudos recibidos por Ubidots y, en caso de ser necesario, la comprobación de estos al compararlos con los registrados en el backend Sigfox.
- **Valioso:** el contenido ofrecido debe ser relevante y único para el usuario, de manera que los datos tengan valor. Para esto, la información recabada por el dispositivo deberá ser precisa y suficiente para generar ventaja de los datos, obteniendo la información necesaria para generar ventaja de los datos. El despliegue adecuado de la información es crucial, pues de nada servirá tomar datos importantes si estos no pueden ser interpretados por los usuarios.

Valores registrados para las variables				
Fecha	# Pasajeros	Subidas	Bajadas	# Estación
2023/01/13 06:41:10	22	1	0	---

Figura 8.5: Registro de valores en Ubidots.

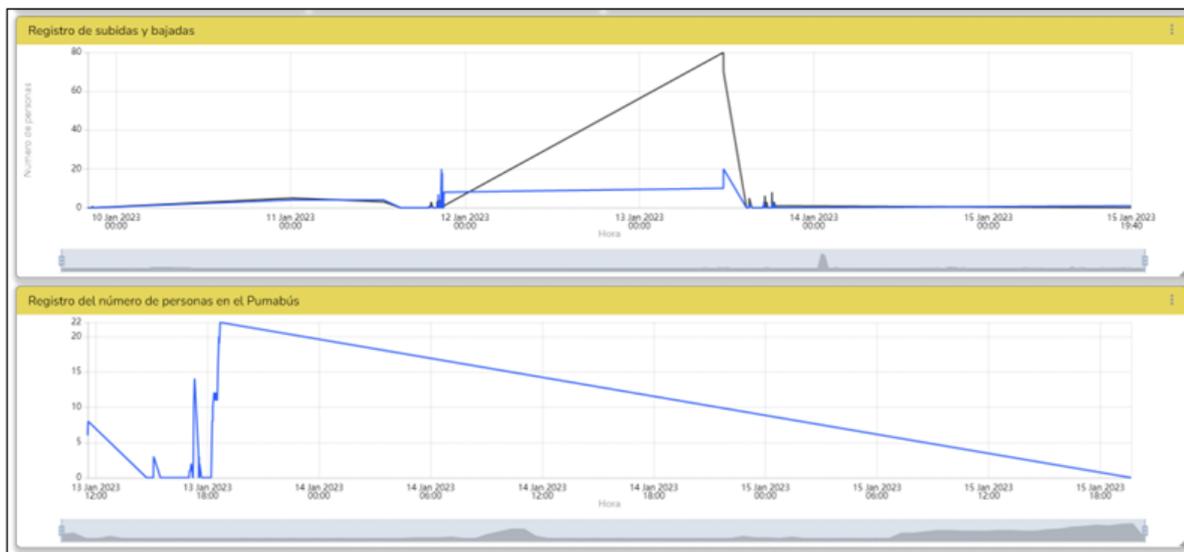


Figura 8.6: Gráficas de registro histórico de valores en Ubidots.

Tableros (Dashboard o frontend)

En la plataforma Ubidots se crearon tres opciones de tablero, panel de control o Dashboard, para ser usados por pasajeros, desarrolladores y coordinadores, cada uno con opciones diseñadas específicamente con la información que les puede ser de utilidad. A continuación, se describe a cada uno de los tipos de usuario:

- **Vista de pasajeros:** Se compone de dos recuadros de texto que muestran la ruta del pumabús que se está monitoreando y la placa de la unidad, un mapa donde se puede observar el recorrido del pumabús y tres widgets de HTML Canvas donde se muestra el estado del vehículo respecto al número de pasajeros y el nombre de la siguiente estación.

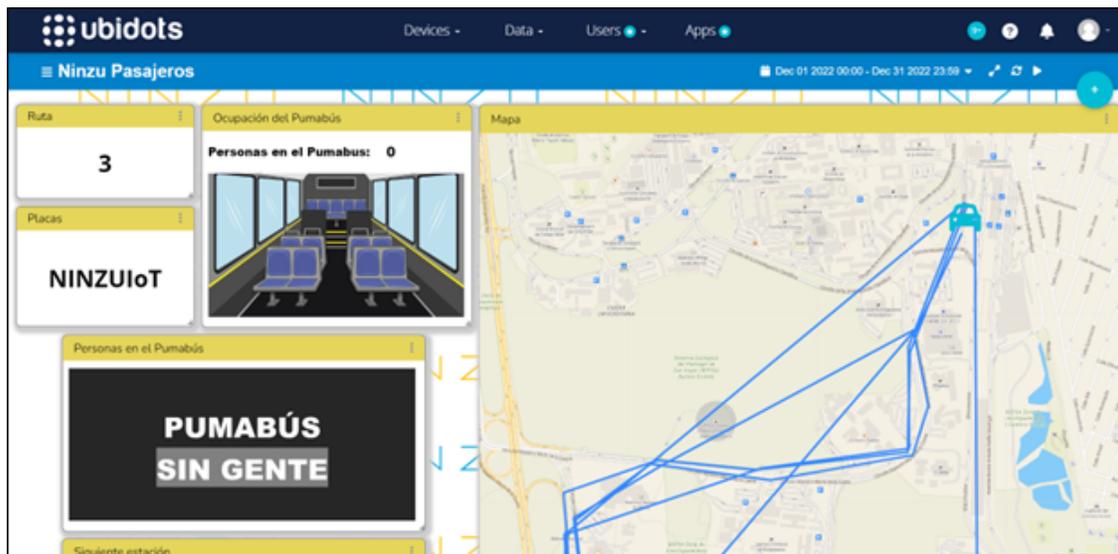


Figura 8.7: Vista de pasajeros.

- **Vista de desarrollador:** Se compone de dos recuadros de texto que muestran la ruta del pumabús que se está monitoreando, la placa de la unidad, un mapa donde se puede observar el recorrido del pumabús, un widget de HTML Canvas donde se muestra el estado del vehículo respecto al número de pasajeros, dos widgets métricos para el despliegue del número de bajadas y de subidas en la última estación y una tabla de datos con los datos históricos registrados.

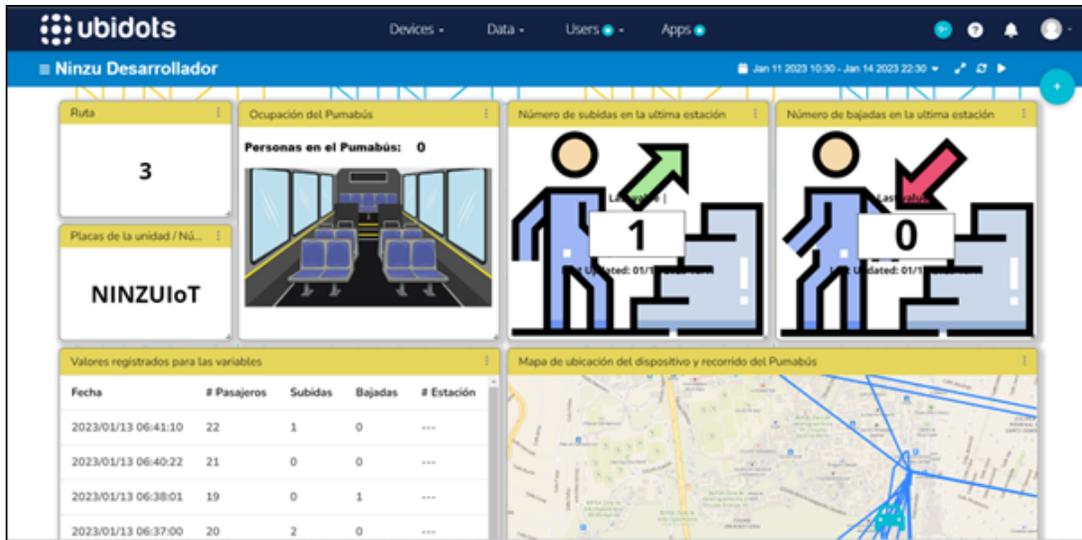


Figura 8.8: Vista de desarrollador.

- Vista de coordinador:** Se compone de dos recuadros de texto que muestran la ruta del pumabús que se está monitoreando y la placa de la unidad, un mapa donde se puede observar el recorrido del pumabús, un widget de HTML Canvas donde se muestra el estado del vehículo respecto al número de pasajeros, dos widgets métricos para el despliegue del número de bajadas y de subidas en la última estación y dos gráficas para los valores de subidas, bajadas y número de pasajeros.

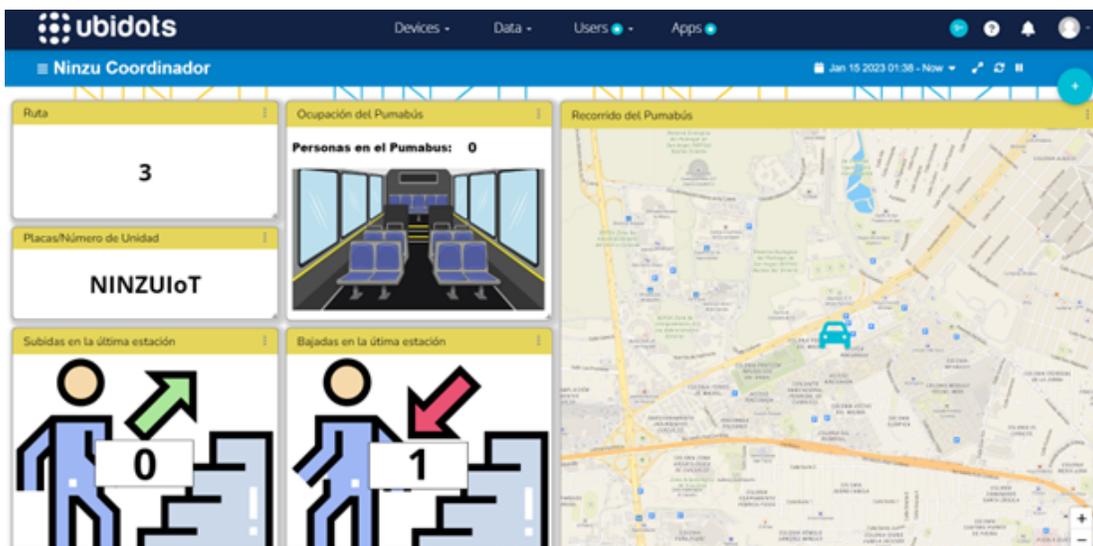


Figura 8.9: Vista de coordinador.

Elementos o widgets usados

- **Ruta:** Muestra el número de ruta del Puambús que se está monitoreando. Ya que el estudio se delimitó a la ruta 3, se seleccionó de tipo texto porque no era necesario que el valor cambiara.



Figura 8.10: Ruta.

- **Placas:** Muestra número de unidad o placas del vehículo monitoreado. Ya que la comunicación Sigfox está limitada a 12 bytes, de los cuales están todos ocupados por la información enviada, el número de placa o de unidad debe ser introducido de manera manual. Este número es estático, por lo que se seleccionó un widget de tipo texto.



Figura 8.11: Placas.

- **Ocupación:** Es un widget de HTML Canvas que muestra gráfica y numéricamente el número de pasajeros. Se seleccionó así por la libertad que se tiene con HTML Canvas para seleccionar lo que muestra el widget.



Figura 8.12: Ocupación (a) Poca gente (b) Algo de gente.

- **Personas en el pumabús:** Ligado al widget anterior, éste también es de HTML Canvas y representa con texto la cantidad de pasajeros en el pumabús. Ya que se debía cambiar entre varias imágenes para mostrar los distintos avisos de acuerdo con la cantidad de personas en el camión, se usó HTML.



Figura 8.13: Texto con la ocupación.

- **Siguiente estación:** Es un widget de HTML Canvas que muestra el nombre de la siguiente estación por la que pasará el autobús. Este widget debe mostrar una imagen distinta para cada estación, por lo que se seleccionó HTML.



Figura 8.14: Siguiente estación.

- **Subidas en la última estación:** Muestra cuántas personas subieron al pumabús en la última estación por la que pasó. Se usó un widget métrico porque permite mostrar el dato que se desea y colocar una imagen representativa de fondo.



Figura 8.15: Subidas en la última estación.

- **Bajadas en la última estación:** Muestra cuántas personas bajaron del pumabús en la última estación por la que pasó. Se usó un widget métrico porque permite mostrar el dato que se desea y colocar una imagen representativa de fondo.



Figura 8.16: Bajadas en la última estación.

- Mapa:** Muestra la ubicación del camión en un mapa utilizando las coordenadas de latitud y longitud. Se seleccionó porque es el único dentro de Ubidots que permite mostrar la posición del dispositivo.

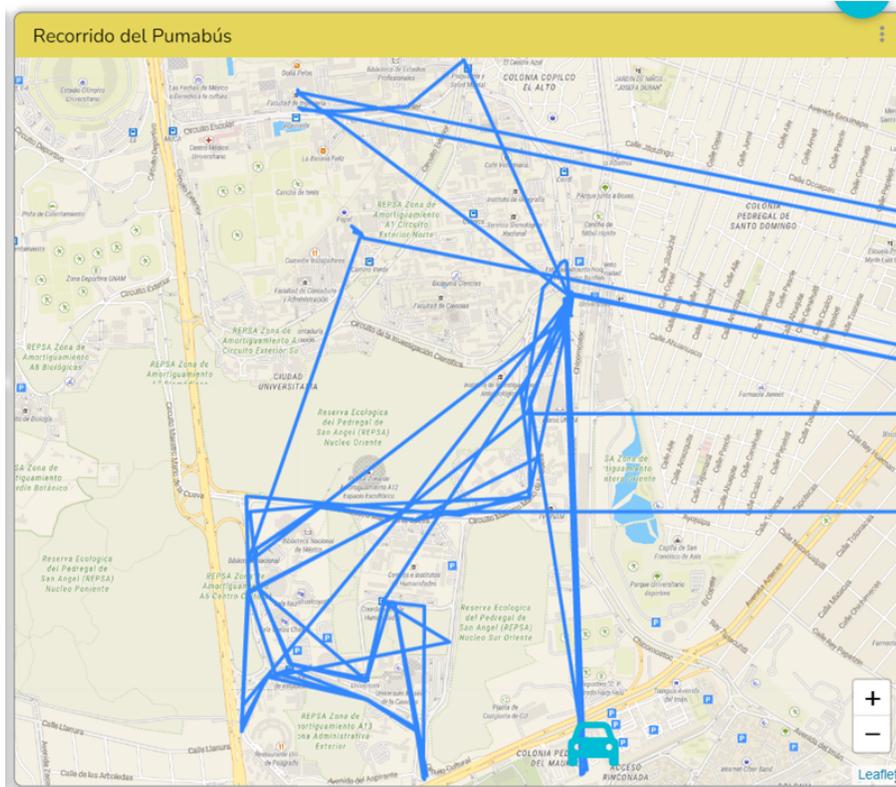


Figura 8.17: Recorrido de pumabús en el mapa.

- Registro de subidas y bajadas:** Gráfica que muestra el número de ascensos y descensos de la unidad a lo largo del tiempo.

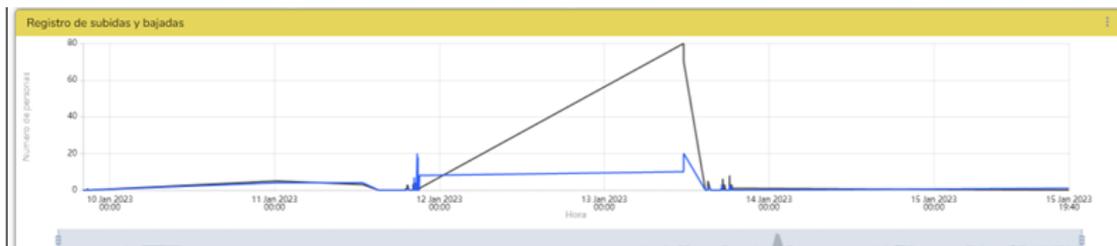


Figura 8.18: Gráfica subidas y bajadas.

- Registro de pasajeros:** Gráfica que muestra la cantidad de pasajeros en el pumabús a lo largo del tiempo, permite al coordinador observar la evolución de la ocupación y conocer el estado de la unidad.

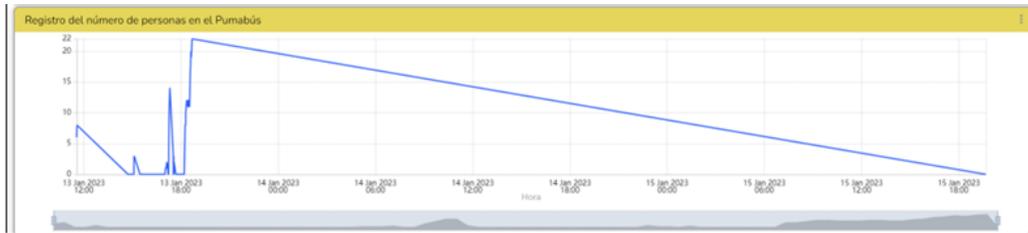


Figura 8.19: Gráfica de pasajeros.

- Tabla de valores registrados:** Es un widget de tipo Data Table, que muestra la información de los datos crudos recibidos por Ubidots.

Fecha	# Pasajeros	Subidas	Bajadas	# Estación
2023/01/13 06:41:10	22	1	0	---
2023/01/13 06:40:22	21	0	0	---
2023/01/13 06:38:01	19	0	1	---
2023/01/13 06:37:00	20	2	0	---
2023/01/13 06:36:01	18	1	0	---
2023/01/13 06:34:51	17	3	0	---
2023/01/13 06:33:16	14	1	0	---
2023/01/13 06:32:53	13	2	0	---
2023/01/13 06:31:48	11	0	0	---

Figura 8.20: Registro histórico de valores.

8.2 Aplicativo de servicios de software en la nube: Losant

Losant es una plataforma en la nube diseñada específicamente para proyectos de desarrollo y despliegue de aplicaciones de Internet de las Cosas (IoT). Facilita la recopilación, visualización y análisis de datos actualizados provenientes, ya sea de dispositivos conectados, como sensores y sistemas de monitoreo, o de diversas fuentes como API de terceros o bases de datos.

Permite a los usuarios integrar, almacenar y procesar grandes volúmenes de datos de diferentes dispositivos IoT, generando paneles, tableros o dashboards, alertas y reportes personalizados, lo que facilita la toma de decisiones basadas en los datos. Representa una solución para crear aplicaciones IoT de manera rápida y eficiente.

Variables para el procesamiento de información

Name ↕	Data Type ↕	Last Reported Value ↕	Last Reported Time ↕
bajadas	Number	4	20 de mar. de 2025 12:46:26.429
estacion	Number	26	20 de mar. de 2025 12:46:26.429
hora	String	martes, 11 de marzo de 2025, 3:00:00 p.m.	20 de mar. de 2025 12:46:26.429
pasajeros	Number	29	20 de mar. de 2025 12:46:26.429
position	GPS	19.323685,-99.174866	20 de mar. de 2025 12:46:26.429
subidas	Number	7	20 de mar. de 2025 12:46:26.429

Figura 8.21: Variables en Losant.

Dentro del dispositivo y en la plataforma, se establecieron las siguientes variables:

- **Subidas:** Número de personas que ascendieron al vehículo en la última estación.
- **Bajadas:** Número de personas que descendieron del vehículo en la última estación.
- **Estación:** Número de la última estación por la que pasó el pumabús. Todas las paradas de la ruta están numeradas.
- **Hora:** Fecha y hora que permite comprobar la sincronización de las plataformas.
- **Pasajeros:** Número de personas que se encuentran dentro de la unidad.
- **Position:** Coordenadas de latitud y longitud para conocer la localización del pumabús.

Descripción de Experiencia de usuario

Para procurar una experiencia de usuario satisfactoria y que los usuarios decidan seguir utilizando el sistema desarrollado, de manera recurrente, el diseño de la interfaz debe cumplir con las siguientes características:

- **Útil:** La interfaz debe proporcionar información clara y relevante para el usuario. Losant ofrece una amplia variedad de bloques o elementos visuales para visualizar datos en tiempo real, gráficos, indicadores y mapas, lo que permite presentar la información de manera adecuada para cada aplicación. Además, la capacidad de personalizar tableros facilita la adaptación del sistema a necesidades específicas. La herramienta más versátil para adecuar lo que se quiere mostrar al usuario es el bloque HTML.

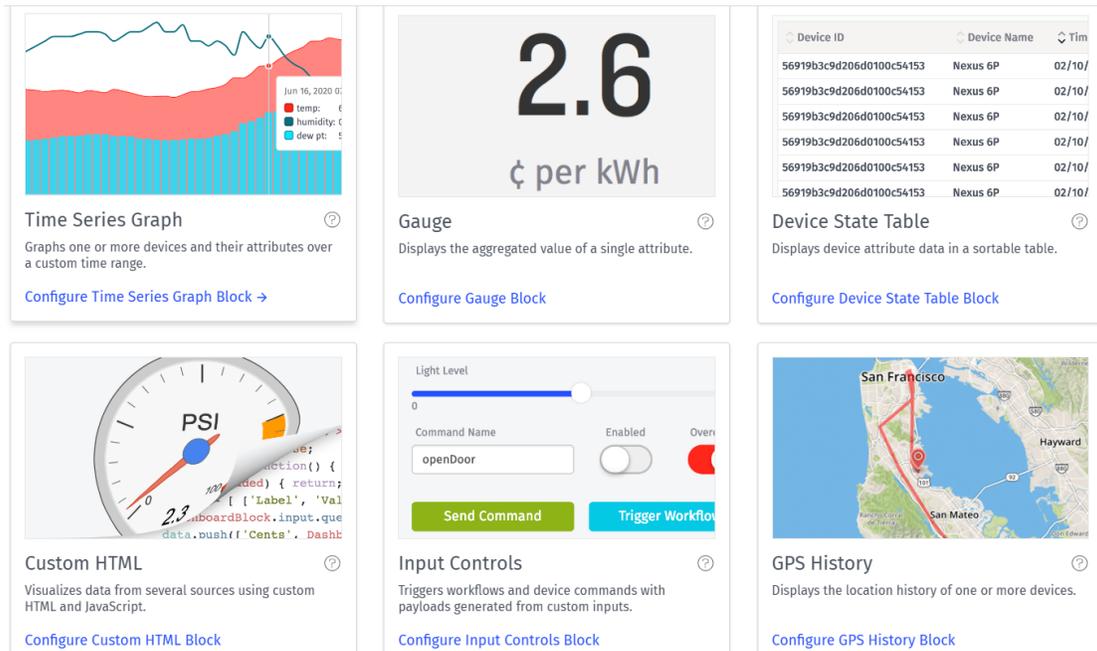


Figura 8.22: Bloques para tableros en losant.

- **Utilizable:** La plataforma debe ser intuitiva y fácil de navegar. Losant permite organizar los tableros con un diseño flexible, lo que permite distribuir los bloques de manera estratégica para mejorar la visualización de la información. Se recomienda usar nombres claros y concisos en los elementos visuales y distribuirlos en un orden lógico para que el usuario pueda identificar rápidamente la información más importante para él.
- **Deseable:** La interfaz debe generar una experiencia agradable y positiva para el usuario. Para ello, es importante seleccionar una paleta de colores armoniosa, fuentes legibles y una estructura visual coherente. Losant permite personalizar los tableros para mejorar la apariencia y alinearla con la identidad del proyecto, haciendo que el sistema sea más atractivo y congruente.
- **Accesible:** La información debe ser fácilmente visible y comprensible para todos los usuarios, incluidas personas con discapacidades. Para esto, es recomendable utilizar colores contrastantes pero no agresivos, fuentes de tamaño adecuado y evitar la sobrecarga de información en un solo espacio. Losant permite configurar accesos y permisos para diferentes tipos de usuarios, asegurando que la información clave llegue a quienes la necesitan.

- **Creíble:** La información desplegada en los tableros debe ser confiable y verificable. La aplicación creada en Losant permite acceder a los datos en crudo a través del tablero de desarrollador, lo que permite contrastar la información mostrada en los bloques con los valores originales enviados desde Sigfox. Además, la integración con herramientas como Sigfox, MQTT y Webhooks garantiza la trazabilidad de los datos en todo momento.

id	hora	subidas	bajadas	pasajeros	estacion
67da2c814a0c235fa489031c	2025-03-11T15:...	15	0	15	1
67da2f1b4a0c235fa48919de	2025-03-11T15:...	5	0	20	2
67da3063a6e653e634fe2317	2025-03-11T15:...	3	1	22	3
67da30994a0c235fa4892718	2025-03-11T15:...	2	0	24	4
67da30e9efa37f90b4a879cc	2025-03-11T15:...	3	3	24	5
67da32244a0c235fa48934b5	2025-03-11T15:...	4	5	23	6
67da329eefa37f90b4a888d2	2025-03-11T15:...	1	3	21	7
67da32f5d85be2b6c3fc66da	2025-03-11T15:...	1	1	21	8
67da3357ef18eb4350f5bbdc	2025-03-11T15:...	14	28	7	9

Figura 8.23: Registro de valores en Losant. Tablero de Desarrollador

- **Valioso:** El sistema debe proporcionar datos relevantes y útiles para los usuarios. Para ello, la información enviada por los dispositivos debe ser precisa y procesada adecuadamente antes de ser mostrada. Losant permite crear flujos de trabajo (Workflows) que procesan, formatean y enriquecen los datos antes de ser presentados, asegurando que la información desplegada en los tableros tenga un impacto real en la toma de decisiones.

Tableros (Dashboard o frontend)

En la plataforma Losant se crearon tres versiones de tablero de visualización, panel de control o Dashboard, para ser usados por cada uno de los tres tipos de usuarios: pasajeros, desarrolladores y coordinadores, cada uno con opciones diseñadas específicamente con la información que les puede ser de utilidad. A continuación, se describe a cada uno de los tipos de usuario:

■ Usuario desarrollador

El desarrollador es el encargado de la configuración, mantenimiento y optimización del sistema en Losant. Su función principal es gestionar la conectividad de los dispositivos IoT, diseñar flujos de trabajo para el procesamiento de datos y asegurar que la información fluya correctamente hacia la plataforma y los tableros. Además, se encarga de la depuración de errores y la implementación de mejoras en la visualización y funcionalidad del sistema. Sólo tendrá acceso al tablero (Figura 8.24) quien sea desarrollador.

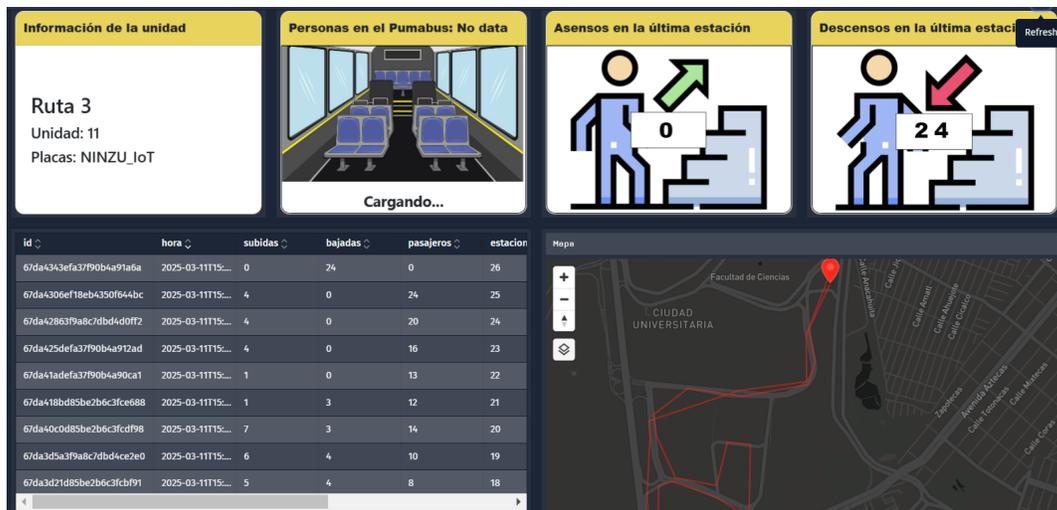


Figura 8.24: Vista de desarrollador.

■ **Usuario pasajero**

Los pasajeros son los principales usuarios finales del sistema, quienes acceden a la información del transporte para planificar sus desplazamientos. Necesitan interfaces intuitivas que les proporcionen datos en tiempo real sobre rutas, geolocalización y ocupación de las unidades en servicio. Su experiencia de usuario debe ser clara, rápida y confiable para fomentar el uso continuo del sistema. El tablero correspondiente debe ser público (Figura 8.25).

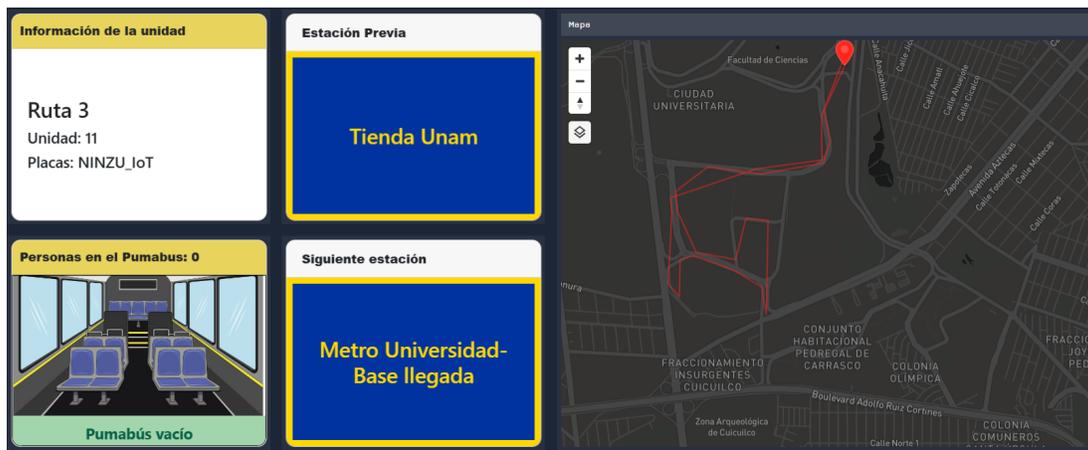


Figura 8.25: Vista de pasajeros.

■ **Usuario coordinador**

Los coordinadores son los responsables de la supervisión y posibles cambios en el sistema de movilidad. Utilizan los paneles de control para monitorear el estado del sistema, verificar el flujo de transporte y analizar métricas clave. Su principal objetivo es garantizar que el servicio funcione de manera eficiente y que los pasajeros tengan acceso a

información precisa y actualizada. Podrá tener acceso a este tablero (Figura 8.26) tanto el desarrollador como el equipo que coordina el sistema de transporte, con posibilidad de crear diferentes cuentas para distintas áreas de la coordinación.

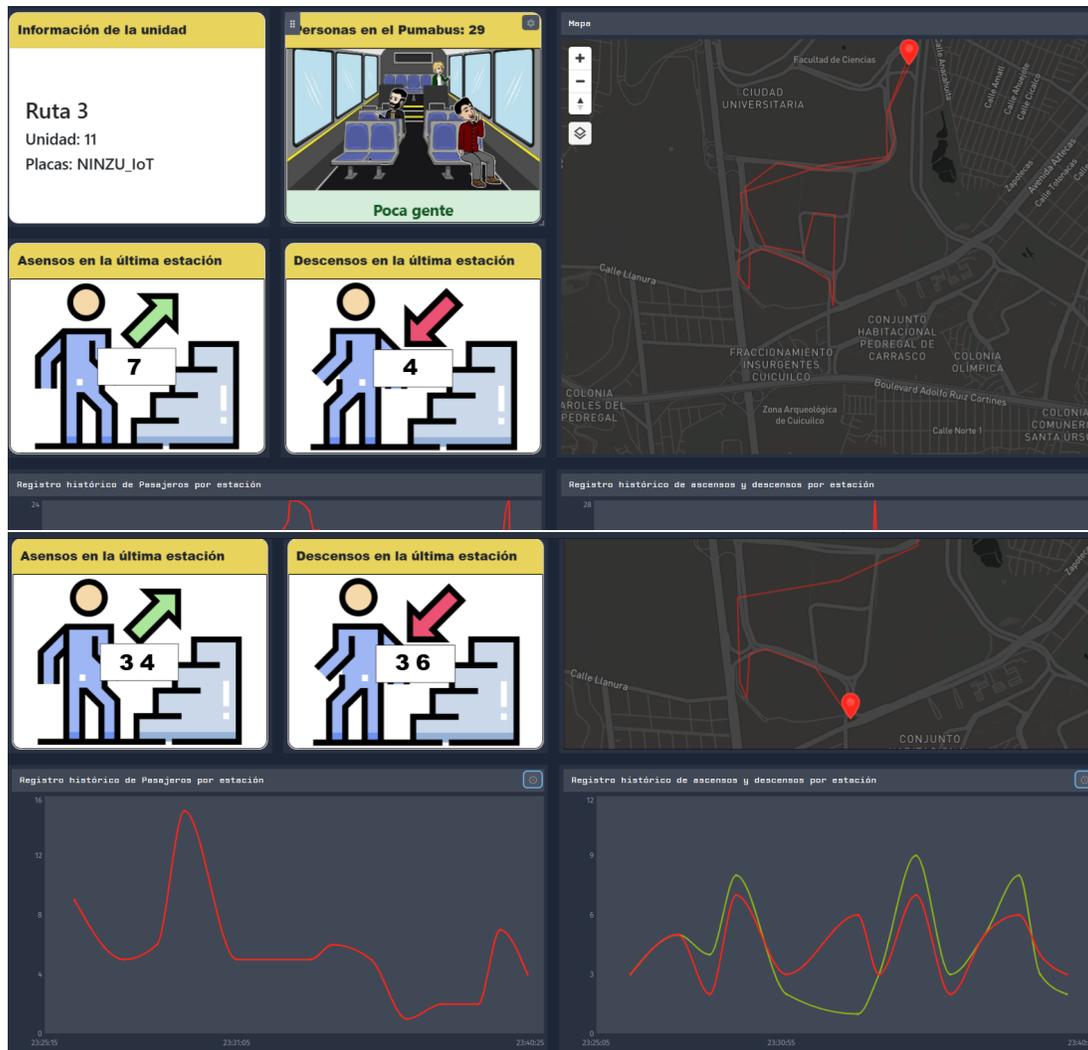


Figura 8.26: Vista de coordinador.

Bloques utilizados

- Información de la unidad:** Recuadro de texto que muestra la ruta del pumabús que se monitorea en ese momento y la placa de la unidad en cuestión. Por ahora, una función es estética y anticipatoria. Su contenido no se actualiza de forma dinámica con datos reales, ya que no forma parte del sistema actual. Sin embargo, su inclusión permite visualizar cómo podría integrarse esa funcionalidad en futuras versiones del sistema, cuando se cuente con los permisos y medios técnicos necesarios para acceder a dicha información directamente desde las unidades Pumabús.



Figura 8.27: Bloque de información de unidad.

- **Estación previa:** Letrero que muestra el nombre de la última estación en la que estuvo el pumabús.

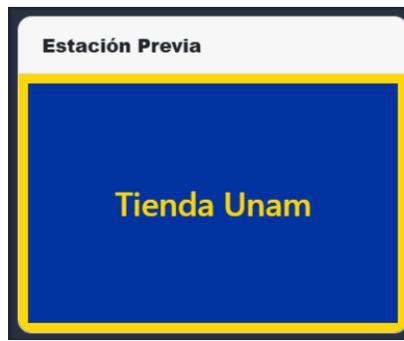


Figura 8.28: Bloque Estación previa.

- **Siguiente estación:** Letrero que muestra el nombre de la estación a la que se dirige el pumabús, anticipando el próximo punto de ascenso o descenso.



Figura 8.29: Bloque Siguiente estación.

- **Personas en el pumabús:** Un bloque que contiene una imagen dinámica, que cambia conforme aumenta o disminuye la cantidad de pasajeros en la unidad, junto con un letrero que resume el estado de ocupación y cambia de color según el estado.



Figura 8.30: Bloque Personas en el pumabús.

- **Mapa:** Un mapa, donde se puede observar el recorrido del pumabús y su última posición registrada sobre una vista geográfica.

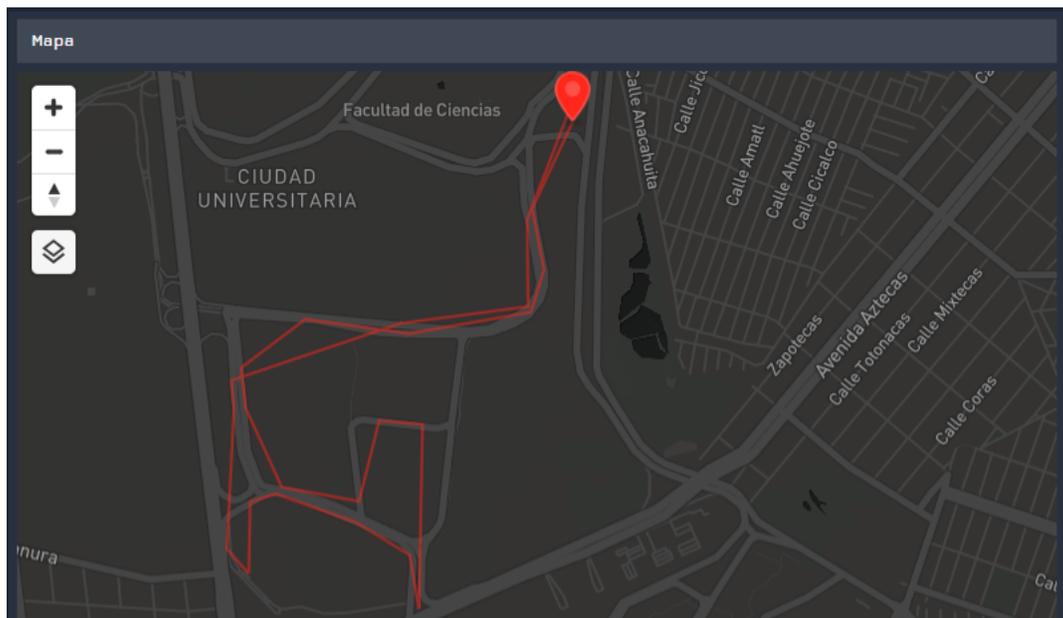


Figura 8.31: Bloque Mapa.

- **Ascensos en la última estación:** Imagen y número dinámicos que indican cuántas personas abordaron en la última estación registrada. El conteo se reinicia cada estación.



Figura 8.32: Bloque Ascensos en la última estación.

- **Descensos en la última estación:** Imagen y número dinámicos que indican cuántas personas descendieron en la última estación registrada. El conteo se reinicia cada estación.



Figura 8.33: Bloque Descensos en la última estación.

- **Tabla de registro histórico:** Presenta los datos históricos recibidos del backend de Sigfox en una tabla ajustable según el período de observación deseado.

id	hora	subidas	bajadas	pasajeros	estacion
67da4343efa37f90b4a91a6a	2025-03-11T15:...	0	24	0	26
67da4306ef18eb4350f644bc	2025-03-11T15:...	4	0	24	25
67da42863f9a8c7dbd4d0ff2	2025-03-11T15:...	4	0	20	24
67da425defa37f90b4a912ad	2025-03-11T15:...	4	0	16	23
67da41adefa37f90b4a90ca1	2025-03-11T15:...	1	0	13	22
67da418bd85be2b6c3fce688	2025-03-11T15:...	1	3	12	21
67da40c0d85be2b6c3cdf98	2025-03-11T15:...	7	3	14	20
67da3d5a3f9a8c7dbd4ce2e0	2025-03-11T15:...	6	4	10	19
67da3d21d85be2b6c3cbf91	2025-03-11T15:...	5	4	8	18

Figura 8.34: Bloque Tabla de registro histórico.

- **Gráficas:** Representaciones visuales que muestran, en función del tiempo, los valores de ascensos, descensos y número de pasajeros a bordo.



Figura 8.35: Bloque Gráficas de registro histórico.

En el capítulo anterior, en la sección 4.1.4, se realizó un análisis comparativo entre las plataformas de visualización de datos Ubidots y Losant. Como resultado, se eligió Losant por su mayor grado de personalización y control lógico sobre los flujos de trabajo. En la presente sección se detalló el desarrollo de la interfaz en dicha plataforma, así como la integración del sistema con la red de comunicación seleccionada (Sigfox). Inicialmente se realizaron pruebas también en Ubidots; sin embargo, el desarrollo principal se centró en Losant, de acuerdo con los resultados del análisis comparativo. A continuación se desarrollan cualidades exclusivas de Losant:

Diagrama de bloques

Un flujo de trabajo (*workflow*) en Losant es una herramienta visual que permite diseñar, automatizar y gestionar procesos dentro de una solución de Internet de las Cosas (IoT). Estos flujos están formados por una serie de bloques conectados entre sí, donde cada bloque representa una acción específica, como la recepción de datos desde un dispositivo, el procesamiento de información, la toma de decisiones o el envío de notificaciones.

Esta modalidad permite ahorrar tiempo al disminuir la necesidad de escribir código, lo que facilita la integración de dispositivos, servicios externos y bases de datos para optimizar la recolección, análisis y respuesta ante eventos de manera inmediata.

En Losant, en la modalidad de diagramas de flujo de trabajo hay dos tipos principales: **Application Workflows** o flujos de trabajo de aplicación y **Experience Workflows**, o flujos de trabajo de experiencia, cada uno con propósitos específicos.

Application Workflows: Se encargan de la automatización y procesamiento de datos. Se usan para manejar la lógica interna de la aplicación, como:

- Recibir y procesar datos de dispositivos (como en el caso de Sigfox).
- Realizar cálculos, transformar datos y enviarlos a otras plataformas.
- Controlar dispositivos en función de eventos o condiciones específicas.
- Guardar datos en bases de datos o tabular información.

Experience Workflows: Diseñados para gestionar la interacción con usuarios finales de la experiencia. Permiten crear aplicaciones personalizadas con interfaces de usuario, autenticación y APIs personalizadas. Se utilizan para:

- Manejar peticiones HTTP y responder con datos procesados.
- Autenticar usuarios y manejar sesiones.
- Renderizar páginas dinámicas en una interfaz web construida dentro de Losant.
- Enviar notificaciones o generar respuestas automatizadas para usuarios finales.

El Flujo de la figura 8.36 es un *application workflow* encargado de recibir los datos enviados por un dispositivo Sigfox, procesarlos y almacenarlos en Losant para su posterior uso. A continuación, se describen los nodos incluidos:

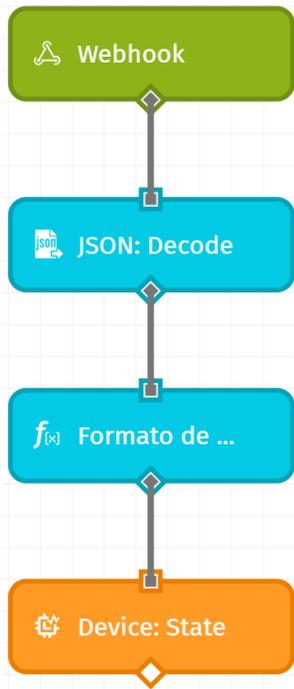


Figura 8.36: Diagrama de bloques del flujo de trabajo en Losant. Obtención de datos de sigfox.

- **Webhook:** Punto de entrada automática de datos al flujo de trabajo desde fuentes externas, como dispositivos y aplicaciones a través de solicitudes HTTP. En este caso Sigfox y Postman en formato JSON.
- **JSON: Decode:** Toma el contenido del webhook y lo convierte en un objeto JSON legible, lo que permite acceder a cada uno de los valores enviados por el dispositivo.
- **Función Formato de hora:** Ajusta el formato de la hora recibida en el payload de Sigfox para que sea más comprensible y utilizable dentro de Losant.
- **Device: State:** Actualiza y almacena la información más reciente de un bloque anterior en la base de datos de Losant y la asocia a un dispositivo específico del sistema.
- **Debug opcional en cada nodo:** Herramienta únicamente de visualización que muestra en una consola los datos que pasan a través del flujo. Facilita la identificación de errores o validación de la información procesada. Permite monitorear y depurar el funcionamiento del flujo sin afectar su comportamiento.

Para que pueda iniciar el flujo del diagrama de bloques con el Webhook, es necesario que en la sección de webhooks se agregue uno nuevo y el enlace generado (Figura 8.37) se introduzca en el backend de Sigfox, como se observa en la Figura 7.2.

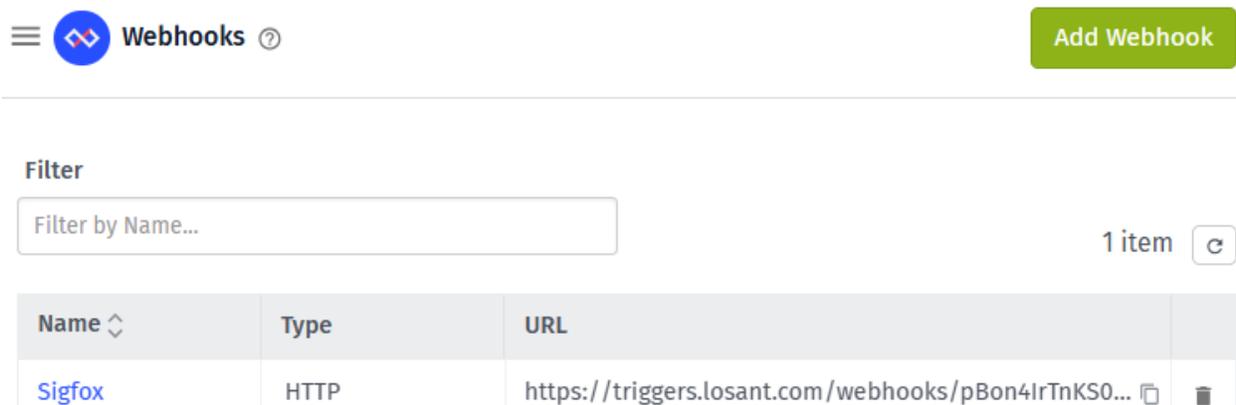


Figura 8.37: Webhook con enlace generado.

Hay bloques que no son absolutamente indispensables para el funcionamiento del sistema pero sí facilitan la visualización de los datos registrados. Tal es el caso del bloque de Función Formato de hora, cuyo código para el manejo del dato que contiene la hora aparece en la Figura 8.38.

Function (JavaScript)

```
1 // Verifica si la hora existe en data.body
2 if (payload.data.body.hora) {
3   let fecha = new Date(payload.data.body.hora);
4
5   let opciones = {
6     weekday: 'long', year: 'numeric', month: 'long', day: 'numeric',
7     hour: 'numeric', minute: 'numeric', second: 'numeric', hour12: true
8   };
9
10  // Guarda la hora formateada en data.body.hora
11  payload.data.body.hora = fecha.toLocaleString('es-MX', opciones);
12 }
```

Figura 8.38: Código del bloque de Función Formato de hora.

El bloque de salida del diagrama, Device:State, requiere que se especifique qué variables podrán ser tomadas del mismo para su uso en otros flujos de trabajo, especificando la ruta de cada una en el paquete de datos (Figura 8.39).

Attribute	Value
bajadas	{{data.body.bajadas}}
estacion	{{data.body.estacion}}
hora	{{data.body.hora}}
pasajeros	{{data.body.pasajeros}}
position	{{data.body.position.latitude}},{{data.body.position.

Figura 8.39: Bloque Device:State de salida del flujo de trabajo.

El flujo de la figura 8.40 se dispara con la actualización de datos en un dispositivo en Losant y se encarga de formatearlos adecuadamente y agregarlos a una tabla de datos para su almacenamiento y visualización posterior en uno de los tableros visuales para usuario final. A continuación, se describen los nodos incluidos:

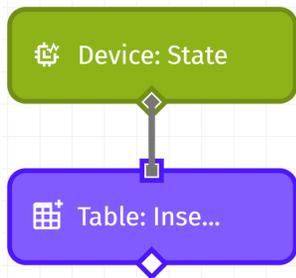


Figura 8.40: Diagrama de bloques del flujo de trabajo en Losant. Agregación a tabla de registro histórico.

- **Trigger: Device State:** Nodo disparador del del flujo. Se activa cuando el dispositivo envía un nuevo estado a Losant, lo que indica que se han recibido datos actualizados desde Sigfox.
- **Table: Insert Rows:** Inserta una nueva fila en una tabla de datos de Losant, previamente creada, con la información procesada para tener un registro histórico visual de datos.
- **Debug opcional en cada nodo:** Herramienta únicamente de visualización que muestra en una consola los datos que pasan a través del flujo. Facilita la identificación de errores o validación de la información procesada. Permite monitorear y depurar el funcionamiento del flujo sin afectar su comportamiento.

El bloque de entrada Device: State, que inicia el flujo de trabajo, requiere que se especifique el dispositivo del cual se obtendrán los atributos que utilizarán los bloques posteriores y qué atributos específicos serán obtenidos.(Figura 8.41).

DEVICE STATE

Allowed Attributes

Only the following ...

Select Attributes

bajadas x hora x estacion x pasajeros x position x subidas x

Figura 8.41: Bloque Device:State de inicio de flujo de trabajo.

El bloque de inserción de filas a la tabla, Device:State, requiere que se especifique la tabla a la que se desean agregar nuevas filas de datos y la columna a la que será asignado cada valor.(Figura 8.42).

TABLE: INSERT ROWS

Data Table ID Template

DataTableDevkit

ROW DATA

Choose a data entry method and map values to your new row or rows (JSON Template or Payload Path).

Data Entry Method

Individual Fields

"DataTableDevkit" has the following columns:

- **hora** (optional string)
- **subidas** (optional number)
- **bajadas** (optional number)
- **pasajeros** (optional number)
- **estacion** (optional number)
- **position** (required string)

Column Template

hora

Value Template **Null**

{{data.hora}}

Column Template

subidas

Figura 8.42: Bloque de inserción de datos a la tabla de registro y despliegue.

En etapas iniciales del proyecto, se realizaron pruebas en Ubidots que permitieron validar conceptos básicos de visualización y transmisión de datos. Sin embargo, durante el desarrollo

del sistema completo, se presentaron limitaciones en la personalización de lógica interna y control sobre los flujos, lo que llevó a migrar el despliegue principal a Losant.

Esta plataforma no sólo permitió replicar las funcionalidades logradas en Ubidots, sino que las superó al facilitar el diseño de flujos de trabajo más complejos, y la manipulación directa de datos desde el backend. Así, el uso de Losant otorgó mayor flexibilidad, escalabilidad y control sobre el sistema desarrollado, lo que es clave para alcanzar los objetivos definidos en esta tesis.

Con el fin de ofrecer una visión clara y resumida de las decisiones técnicas tomadas durante el desarrollo del sistema, se presenta la tabla 8.1, que muestra cada uno de los elementos clave seleccionados.

Elemento	Opción elegida	Justificación
Sensor de conteo	Infrarrojos E18-D80NK	Bajo costo, fácil integración, buena tasa de detección en condiciones controladas.
Módulo GPS	NEO-6M V2	Amplia documentación, costo accesible, precisión suficiente para el caso de uso.
Red de comunicación	Sigfox	Bajo consumo de energía, cobertura adecuada, mensajes cortos suficientes para el sistema.
Aplicativo de visualización	Losant	Interfaz personalizable, buena integración con Sigfox, adecuada para pruebas de despliegue.
Ubicación de dispositivo	Tubos de sujeción a un costado de los escalones de una puerta del pumabús	Mejor control de flujo, más representativo para el análisis de ocupación.
Material de carcasa	PLA impreso en 3D	Material accesible, buena rigidez para uso en interiores, fácil de fabricar en laboratorio y económico.

Tabla 8.1: Resumen de decisiones técnicas y su justificación.

9 VALIDACIÓN DE FUNCIONAMIENTO DE LA SOLUCIÓN

Con el sistema ya ensamblado, se siguió la metodología descrita en la Tabla 9.1 para llevar a cabo pruebas de funcionamiento que tenían el objetivo de verificar que cada componente —tanto de hardware como de software— operara conforme a lo esperado. Estas pruebas permitieron evaluar la correcta detección de pasajeros mediante el sensor seleccionado, así como la precisión del módulo GPS, que se traducía en la consistencia de estaciones detectadas de acuerdo con las geocercas establecidas, y la transmisión de datos a través de la red Sigfox.

Se evaluó el comportamiento del sistema en condiciones reales de operación, validando aspectos como el consumo energético, la estabilidad del firmware y la correcta visualización de los datos en la plataforma de monitoreo. Los resultados obtenidos sirvieron como base para realizar ajustes menores y confirmar la viabilidad del prototipo en un entorno práctico.

Prueba: Comprobación del funcionamiento código de conteo de personas e identificación de estaciones de Pumabús con envío de Mensajes a Sigfox y despliegue de información en Losant.

Objetivo: verificar el funcionamiento del código desarrollado para el conteo de personas e identificación de estaciones con el sistema montado sobre una unidad Pumabús, así como el envío de mensajes al backend de Sigfox y el callback para despliegue correcto de los datos en los tableros desarrollados en el aplicativo Losant como interfaz de usuario final.

No. Paso	Descripción Actividad	Resultados Esperados
1	Conectar el sistema a la batería externa para alimentar eléctricamente los componentes.	 Luz indicadora de nivel de batería.
2	Esperar a que el GPS detecte su primera ubicación y se envíe mensaje inicial.	<pre>0100048aad9a41825cc6c209 sub: 1 baj: 0 conteo: 4 2025-03-19 11:49:30 latitude: 19.334736 longitude: -99.18068 estacion: 9</pre> Visualización de mensaje en Backend de Sigfox.

No. Paso	Descripción Actividad	Resultados Esperados																																																												
3	Colocar el dispositivo en los soportes internos, cercanos a una puerta, de un pumabús de la ruta 3, para que permanezca ahí durante todo el recorrido.	 <p>Dispositivo fijado cerca de la puerta.</p>																																																												
4	Llevar un conteo y registro manual de las personas que ascienden y descienden a través de la puerta que tiene el dispositivo.	-																																																												
5	Revisar la recepción de mensajes en Sigfox Backend.	<table border="1" data-bbox="802 877 1398 1276"> <thead> <tr> <th>hora</th> <th>subidas</th> <th>bajadas</th> <th>pasajeros</th> <th>estacion</th> </tr> </thead> <tbody> <tr><td>martes, 11 de mar...</td><td>7</td><td>4</td><td>29</td><td>26</td></tr> <tr><td>2025-03-11T15:00:00</td><td>0</td><td>24</td><td>0</td><td>26</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>24</td><td>25</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>20</td><td>24</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>16</td><td>23</td></tr> <tr><td>2025-03-11T15:00:00</td><td>1</td><td>0</td><td>13</td><td>22</td></tr> </tbody> </table> <p>Registro de conteo y estaciones detectadas.</p>	hora	subidas	bajadas	pasajeros	estacion	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	2025-03-11T15:00:00	0	24	0	26	2025-03-11T15:00:00	4	0	24	25	2025-03-11T15:00:00	4	0	20	24	2025-03-11T15:00:00	4	0	16	23	2025-03-11T15:00:00	1	0	13	22
hora	subidas	bajadas	pasajeros	estacion																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
2025-03-11T15:00:00	0	24	0	26																																																										
2025-03-11T15:00:00	4	0	24	25																																																										
2025-03-11T15:00:00	4	0	20	24																																																										
2025-03-11T15:00:00	4	0	16	23																																																										
2025-03-11T15:00:00	1	0	13	22																																																										
6	Para cada estación en la que se registre un mensaje en el backend de Sigfox, verificar que se agreguen los datos correspondientes a la tabla del dispositivo en Losant.	<table border="1" data-bbox="802 1350 1398 1749"> <thead> <tr> <th>hora</th> <th>subidas</th> <th>bajadas</th> <th>pasajeros</th> <th>estacion</th> </tr> </thead> <tbody> <tr><td>martes, 11 de mar...</td><td>7</td><td>4</td><td>29</td><td>26</td></tr> <tr><td>2025-03-11T15:00:00</td><td>0</td><td>24</td><td>0</td><td>26</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>24</td><td>25</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>20</td><td>24</td></tr> <tr><td>2025-03-11T15:00:00</td><td>4</td><td>0</td><td>16</td><td>23</td></tr> <tr><td>2025-03-11T15:00:00</td><td>1</td><td>0</td><td>13</td><td>22</td></tr> </tbody> </table> <p>Registro de conteo y estaciones detectadas.</p>	hora	subidas	bajadas	pasajeros	estacion	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	martes, 11 de mar...	7	4	29	26	2025-03-11T15:00:00	0	24	0	26	2025-03-11T15:00:00	4	0	24	25	2025-03-11T15:00:00	4	0	20	24	2025-03-11T15:00:00	4	0	16	23	2025-03-11T15:00:00	1	0	13	22
hora	subidas	bajadas	pasajeros	estacion																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
martes, 11 de mar...	7	4	29	26																																																										
2025-03-11T15:00:00	0	24	0	26																																																										
2025-03-11T15:00:00	4	0	24	25																																																										
2025-03-11T15:00:00	4	0	20	24																																																										
2025-03-11T15:00:00	4	0	16	23																																																										
2025-03-11T15:00:00	1	0	13	22																																																										

No. Paso	Descripción Actividad	Resultados Esperados																																																												
7	Para cada estación en la que se registre un mensaje en el backend de Sigfox, corroborar que los bloques del tablero de usuarios pasajeros en Losant se actualicen correctamente.	 <p>Tablero de pasajeros actualizado.</p>																																																												
8	Para cada estación en la que se registre un mensaje en el backend de Sigfox, corroborar que los bloques del tablero de coordinadores del sistema Pumabús en Losant se actualicen correctamente.	 <p>Tablero de Coordinadores actualizado.</p>																																																												
9	Para cada estación en la que se registre un mensaje en el backend de Sigfox, corroborar que los bloques del tablero de desarrolladores en Losant se actualicen correctamente.	 <table border="1" data-bbox="802 1123 1128 1302"> <thead> <tr> <th>M.</th> <th>hora</th> <th>salidas</th> <th>llegadas</th> <th>pasajeros</th> <th>estación</th> </tr> </thead> <tbody> <tr><td>EN6A336F190404F04a</td><td>2025-03-11T19:...</td><td>0</td><td>26</td><td>0</td><td>26</td></tr> <tr><td>EN6A336F190404F04b</td><td>2025-03-11T19:...</td><td>4</td><td>0</td><td>26</td><td>25</td></tr> <tr><td>EN6A336F190404F04f</td><td>2025-03-11T19:...</td><td>4</td><td>0</td><td>20</td><td>24</td></tr> <tr><td>EN6A336F190404F04d</td><td>2025-03-11T19:...</td><td>4</td><td>0</td><td>16</td><td>23</td></tr> <tr><td>EN6A336F190404F04e1</td><td>2025-03-11T19:...</td><td>1</td><td>0</td><td>13</td><td>22</td></tr> <tr><td>EN6A336F190404F04e8</td><td>2025-03-11T19:...</td><td>1</td><td>3</td><td>12</td><td>21</td></tr> <tr><td>EN6A336F190404F04f6</td><td>2025-03-11T19:...</td><td>7</td><td>3</td><td>14</td><td>20</td></tr> <tr><td>EN6A336F190404F04d0</td><td>2025-03-11T19:...</td><td>6</td><td>4</td><td>10</td><td>19</td></tr> <tr><td>EN6A336F190404F04f91</td><td>2025-03-11T19:...</td><td>5</td><td>4</td><td>8</td><td>18</td></tr> </tbody> </table> <p>Tablero de desarrolladores actualizado.</p>	M.	hora	salidas	llegadas	pasajeros	estación	EN6A336F190404F04a	2025-03-11T19:...	0	26	0	26	EN6A336F190404F04b	2025-03-11T19:...	4	0	26	25	EN6A336F190404F04f	2025-03-11T19:...	4	0	20	24	EN6A336F190404F04d	2025-03-11T19:...	4	0	16	23	EN6A336F190404F04e1	2025-03-11T19:...	1	0	13	22	EN6A336F190404F04e8	2025-03-11T19:...	1	3	12	21	EN6A336F190404F04f6	2025-03-11T19:...	7	3	14	20	EN6A336F190404F04d0	2025-03-11T19:...	6	4	10	19	EN6A336F190404F04f91	2025-03-11T19:...	5	4	8	18
M.	hora	salidas	llegadas	pasajeros	estación																																																									
EN6A336F190404F04a	2025-03-11T19:...	0	26	0	26																																																									
EN6A336F190404F04b	2025-03-11T19:...	4	0	26	25																																																									
EN6A336F190404F04f	2025-03-11T19:...	4	0	20	24																																																									
EN6A336F190404F04d	2025-03-11T19:...	4	0	16	23																																																									
EN6A336F190404F04e1	2025-03-11T19:...	1	0	13	22																																																									
EN6A336F190404F04e8	2025-03-11T19:...	1	3	12	21																																																									
EN6A336F190404F04f6	2025-03-11T19:...	7	3	14	20																																																									
EN6A336F190404F04d0	2025-03-11T19:...	6	4	10	19																																																									
EN6A336F190404F04f91	2025-03-11T19:...	5	4	8	18																																																									

Tabla 9.1: Metodología para realización de pruebas de funcionamiento de la solución.

El encendido del módulo GPS fue conforme a lo indicado por el fabricante, es decir, en menos de un minuto. Una vez listo el dispositivo para las pruebas, se realizaron tres recorridos completos en la ruta 3 del Pumabús, distribuidos en dos días. Un recorrido el primer día y dos recorridos consecutivos el segundo día. La presencia del LED indicador de la batería portátil ayudó en las pruebas, al eliminar la necesidad de utilizar una computadora portátil con el monitor serie del IDE de Arduino. **Los reportes manual y del prototipo fueron simultáneos para poder evaluar el desempeño del sistema desarrollado.**

Los registros de las pruebas realizadas se incluyen por separado en las tablas 9.2, del funcionamiento de la geolocalización, y 9.3, 9.4 y 9.5, del funcionamiento del conteo de pasajeros. Que una celda tenga únicamente “-”, significa que no se registró el dato respectivo y eso explica algunas diferencias entre ambos tipos de registro.

Geolocalización

Número de estación	Nombre de la estación	¿Se identificó la estación cuando el pumabús llegó a la parada?		
		Vuelta 1	Vuelta 2	Vuelta 3
1	Metro Universidad-Base salida	SÍ	SÍ	SÍ
2	Química Conjunto D Y E	SÍ	SÍ	SÍ
3	Tienda Unam - 2	SÍ	SÍ	SÍ
4	Facultad de Ciencias Políticas	SÍ	SÍ	SÍ
5	Investigaciones Jurídicas 2	SÍ	SÍ	SÍ
6	Biblioteca Nacional 2	SÍ	SÍ	SÍ
7	Zona Cultural	SÍ	NO	NO
8	Unidad de Posgrado	NO	SÍ	NO
9	Posgrado de Economía	SÍ	SÍ	SÍ
10	D.G.I.R.E	SÍ	SÍ	SÍ
11	D.G.A.P.A	SÍ	SÍ	SÍ
12	Archivo General	SÍ	SÍ	SÍ
13	Avenida del Iman	SÍ	SÍ	SÍ
14	Investigación Filosóficas	SÍ	SÍ	SÍ
15	Investigaciones Filológicas	NO	NO	SÍ
16	Coordinación de Humanidades	SÍ	SÍ	SÍ
17	Universum	SÍ	SÍ	SÍ
18	Teatro Y Danza	SÍ	SÍ	SÍ
19	MUAC	SÍ	SÍ	SÍ
20	Biblioteca Nacional	SÍ	SÍ	SÍ
21	Espacio Escultórico	SÍ	SÍ	SÍ
22	Investigaciones Jurídicas	SÍ	SÍ	SÍ
23	Tv Unam	SÍ	SÍ	SÍ
24	C.U.E.C	SÍ	SÍ	SÍ
25	D.G.S.A	SÍ	SÍ	SÍ
26	Tienda Unam	SÍ	SÍ	SÍ
27	Metro Universidad-Base llegada	SÍ	NO	SÍ

Tabla 9.2: Pruebas de funcionamiento general. Geolocalización

En amarillo se resaltan las estaciones que no fueron detectadas en al menos una vuelta y en rojo en específico la celda de la vuelta en la que no se detectó.

Conteo de pasajeros

Número de estación	Registro manual			Prototipo elaborado			
	Puertas sin dispositivo ¹	Subidas	Bajadas	Pasajeros	Subidas	Bajadas	Pasajeros
1	-	8	0	8	15	0	15
2	-	0	0	8	0	0	15
3	-	0	0	8	0	0	15
4	-	2	0	10	1	0	16
5	-	0	1	9	0	1	15
6	-	0	0	9	0	0	15
7	-	0	2	7	0	1	14
8	-	4	3	8	-	-	-
9	-	2	0	10	2	0	16
10	-	1	0	11	1	0	17
11	-	0	0	11	0	0	17
12	-	1	0	12	1	1	17
13	-	1	1	12	1	1	17
14	-	0	0	12	0	0	17
15	-	1	0	13	-	-	-
16	-	0	1	12	0	0	17
17	-	4	0	16	4	0	21
18	-	1	0	17	1	0	22
19	-	3	0	20	3	0	25
20	-	3	0	23	3	0	28
21	-	0	0	23	0	0	28
22	-	2	0	25	2	0	30
23	-	0	0	25	0	0	30
24	-	0	0	25	0	0	30
25	-	0	0	25	0	0	30
26	-	0	0	25	0	0	30
27	-	0	-	0	0	15	15

Tabla 9.3: Pruebas de funcionamiento general. Conteo de pasajeros. Primera vuelta.

En rojo se resaltan las estaciones no detectadas, en amarillo aquellas en las que hubo diferencia entre el conteo manual y el del sistema y en gris las estaciones con desfase ya considerado.

¹Registro de cambios en el número de pasajeros por ascensos o descensos observados en puertas donde no había dispositivo colocado.

Conteo de pasajeros

Número de estación	Registro manual				Prototipo elaborado		
	Puertas sin dispositivo	Subidas	Bajadas	Pasajeros	Subidas	Bajadas	Pasajeros
1	-	8	0	8	16	0	16
2	-	1	0	9	1	7	10
3	-	-	-	-	0	0	10
4	-2	0	0	9	0	0	11
5	-	2	0	11	2	0	13
6	-	1	0	12	1	0	14
7	-3	0	1	-	-	-	-
8	-	-	-	-	0	0	14
9	-2	0	2	-	0	2	12
10	-	-	-	-	2	0	14
11	-	2	0	9	2	0	16
12	-	0	0	9	0	0	16
13	-	4	8	5	5	3	18
14	-	0	1	-	1	1	18
15	-	-	-	-	0	0	18
16	-	-	-	-	0	0	18
17	-	1	0	-	1	0	19
18	-	2	0	7	2	0	21
19	-	-	-	-	0	0	21
20	-	2	0	9	2	0	23
21	-	0	0	9	0	0	23
22	-	0	0	-	0	0	23
23	-	-	-	-	0	0	23
24	-	-	-	-	0	0	23
25	-	-	-	-	0	0	23
26	-	-	-	-	0	0	23
27	-	-	-	-	-	-	-

Tabla 9.4: Pruebas de funcionamiento general. Conteo de pasajeros. Segunda vuelta.

En rojo se resaltan las estaciones no detectadas, en amarillo aquellas en las que hubo diferencia entre el conteo manual y el del sistema y en gris las estaciones con desfase ya considerado.

Conteo de pasajeros

Número de estación	Registro manual				Prototipo elaborado		
	Puertas sin dispositivo	Subidas	Bajadas	Pasajeros	Subidas	Bajadas	Pasajeros
1	-	6	0	6	4	0	27
2	-1	0	0	5	0	0	27
3	-	0	0	5	0	0	27
4	-	2	0	7	1	0	28
5	-	0	1	6	0	1	27
6	0	0	1	5	0	1	26
7	-	-	-	-	-	-	-
8	-	-	-	-	-	-	-
9	-1+1	-	-	8	1	0	27
10	-	2	0	10	2	0	29
11	2	0	0	12	0	0	29
12	-3	-	-	-	0	0	29
13	-	-	-	9	1	1	29
14	-	0	0	9	0	0	29
15	-	0	0	9	0	0	29
16	-	0	0	9	0	0	29
17	-	1	0	10	1	0	30
18	1	8	0	19	8	0	38
19	1	1	0	21	0	0	38
20	-	0	0	21	0	0	38
21	-	1	0	22	1	0	39
22	-	1	0	23	1	0	40
23	-	-	-	-	0	0	40
24	-	0	0	25	0	0	40
25	-	0	2	23	0	2	38
26	-	0	2	21	0	2	36
27	-	0	-	0	0	25	11

Tabla 9.5: Pruebas de funcionamiento general. Conteo de pasajeros. Tercera vuelta, inmediatamente después de la segunda.

En rojo se resaltan las estaciones no detectadas y en amarillo aquellas en las que hubo diferencia entre el conteo manual y el del sistema. En la primera, aparte de desfase en pasajeros, hubo error.

10 ANÁLISIS DE RESULTADOS

10.1 Geolocalización

Durante los recorridos, se presentaron errores puntuales en la detección de 2 o 3 estaciones por vuelta; sin embargo, todas fueron detectadas al menos en un recorrido. Esto sugiere que las coordenadas establecidas son correctas, y que las omisiones se relacionan con la frecuencia de muestreo del módulo GPS. Es posible que el pumabús se desplazara demasiado rápido en ciertos tramos, impidiendo que el módulo captara la geocerca a tiempo o que los conductores no se detuvieron exactamente en donde corresponde a la parada.

Otro factor a tomar en cuenta es que existen estaciones en las que coinciden varias rutas y, en ocasiones, hay aglomeraciones de autobuses en el lugar y los conductores tienen que hacer alto total y permitir el flujo de pasajeros desde antes de llegar a la geocerca o después de haberla pasado. En la tabla 10.1 se observa la tasa de detección correcta para cada estación, según el número de veces que hayan sido identificadas al final de las tres vueltas de prueba registradas. Puesto que no hubo detección de estaciones donde no hubiera geocerca y no hubo ni puede haber plena igualdad de condiciones en todos los recorridos, no se hace hincapié en una precisión del sistema de detección de estaciones.

Número de estación	Nombre de estación	Tasa de identificación correcta
1	Metro Universidad-Base salida	3/3 (100 %)
2	Química Conjunto D Y E	3/3 (100 %)
3	Tienda Unam - 2	3/3 (100 %)
4	Facultad de Ciencias Políticas	3/3 (100 %)
5	Investigaciones Jurídicas 2	3/3 (100 %)
6	Biblioteca Nacional 2	3/3 (100 %)
7	Zona Cultural	1/3 (33.3 %)
8	Unidad de Posgrado	1/3 (33.3 %)
9	Posgrado de Economía	3/3 (100 %)
10	D.G.I.R.E	3/3 (100 %)
11	D.G.A.P.A	3/3 (100 %)
12	Archivo General	3/3 (100 %)
13	Avenida del Imán	3/3 (100 %)

Número de estación	Nombre de estación	Tasa de identificación correcta
14	Investigación Filosóficas	3/3 (100 %)
15	Investigaciones Filológicas	1/3 (33.3 %)
16	Coordinación de Humanidades	3/3 (100 %)
17	Universum	3/3 (100 %)
18	Teatro Y Danza	3/3 (100 %)
19	MUAC	3/3 (100 %)
20	Biblioteca Nacional	3/3 (100 %)
21	Espacio Escultórico	3/3 (100 %)
22	Investigaciones Jurídicas	3/3 (100 %)
23	Tv Unam	3/3 (100 %)
24	C.U.E.C	3/3 (100 %)
25	D.G.S.A	3/3 (100 %)
26	Tienda Unam	3/3 (100 %)
27	Metro Universidad-Base llegada	2/3 (66.6 %)

Tabla 10.1: Tasa de identificación correcta de estaciones en 3 vueltas

En amarillo se resaltan las estaciones que fueron detectadas en dos vueltas y en rojo las que fueron detectadas en sólo una vuelta.

$$\text{Tasa de identificación correcta (\%)} = \left(\frac{\text{Veces que se detectó la estación}}{\text{Número de vueltas}} \right) \times 100$$

$$\text{Tasa de identificación promedio} = \left(\frac{\sum_{i=1}^{27} \text{Tasa de identificación}_i}{27} \right),$$

donde Tasa de identificación_{*i*} es el porcentaje de aciertos para la estación *i* (ej: 100 %, 33.3 %).

$$\text{Tasa de identificación promedio} = \frac{(23 \times 100 \%) + (3 \times 33.3 \%) + 66.6 \%}{27} \approx 91.36 \%$$

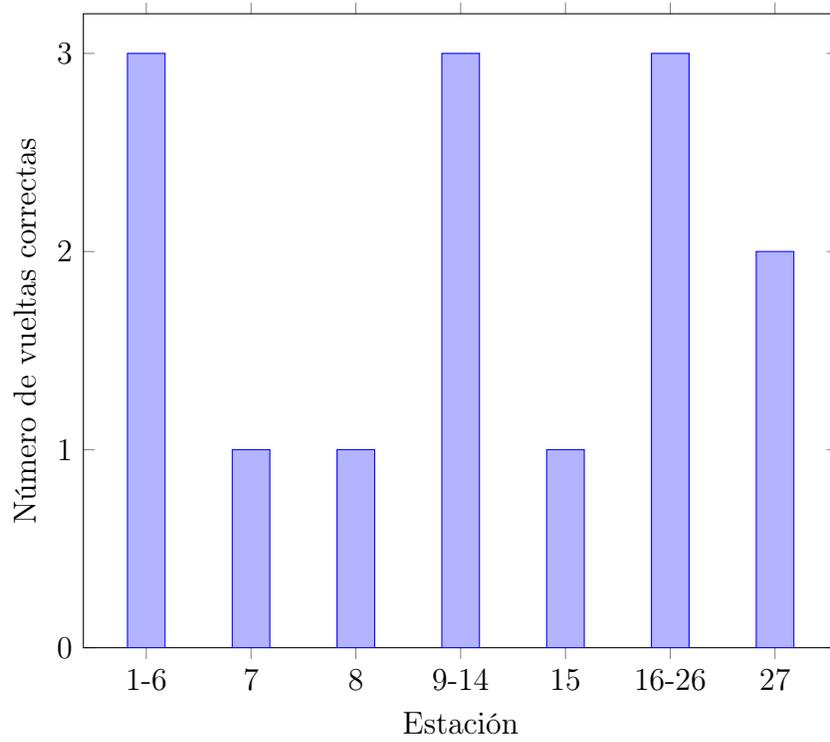


Figura 10.1: Detecciones correctas por estación.

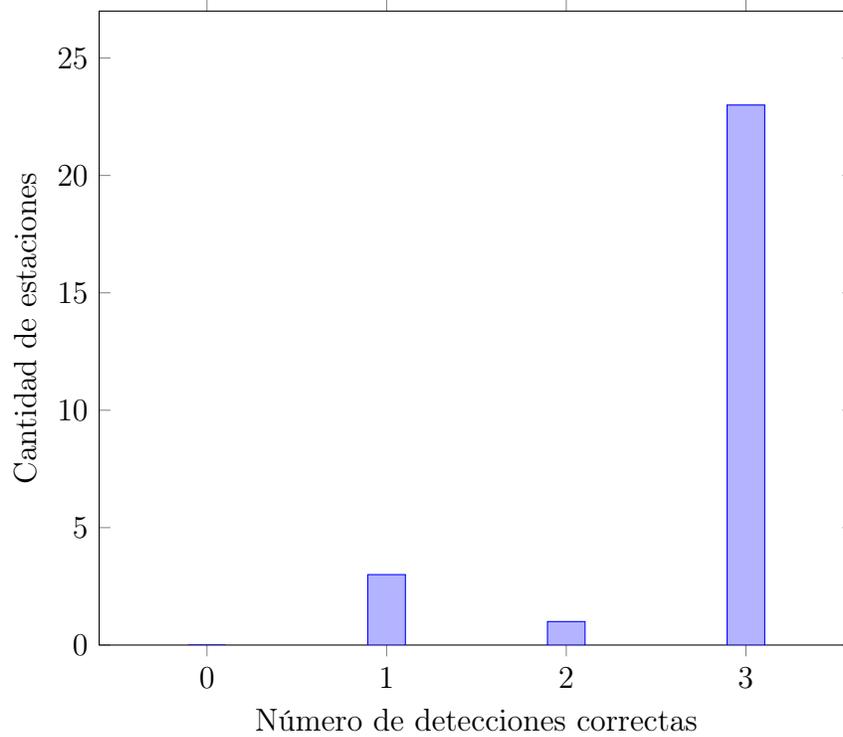


Figura 10.2: Consistencia en la identificación de estaciones.

10.2 Conteo de pasajeros

Se presentaron diferencias entre los datos de ambos métodos de registro, en la primera estación, en los tres recorridos (+7, +8 y +23, registrados por el sistema, respectivamente). Las diferencias fueron atribuidas a la activación de los sensores por la manipulación del dispositivo en el proceso de colocación, lo que provocó un desfase en los valores de todo el recorrido.

Dado que la última estación de la segunda vuelta no se detectó, el desfase de la tercera vuelta correspondía al último número de pasajeros de la vuelta anterior y que hay que corregir el sesgo antes de comparar los resultados de ambos métodos. Por ejemplo, para la primera estación, el error se calcula de la siguiente manera:

$$\text{Error absoluto} = |\text{Valor manual} - (\text{Valor del sistema} - \text{desfase})|$$

Para la primera estación de cada una de las 3 vueltas, los errores son los siguientes:

$$\begin{aligned} |8 - (15 - 7)| &= 0 \text{ pasajeros,} \\ |8 - (16 - 8)| &= 0 \text{ pasajeros,} \\ |6 - (27 - 23)| &= 2 \text{ pasajeros} \end{aligned}$$

El error absoluto promedio para las vueltas resulta de la siguiente manera:

$$\frac{\sum_{i=1}^n |\text{Valor manual}_i - (\text{Valor del sistema}_i - \text{desfase})|}{n}$$

Y el error relativo resulta de la siguiente manera:

$$\frac{\sum_{i=1}^n |\text{Valor manual}_i - (\text{Valor del sistema}_i - \text{desfase})|}{\sum_{i=1}^n \text{Valor manual}_i}$$

En ambas, n es el número de estaciones detectadas y se toman en cuenta subidas y bajadas. Y, como el porcentaje de exactitud se calcula así:

$$(1 - \text{Error relativo}) \times 100,$$

Entonces los valores para cada vuelta son los siguientes:

Vuelta 1 (25 estaciones detectadas)

$$\text{Error absoluto} = \frac{1 + 1 + 1 + 1}{25} = \frac{4}{25} = 0.16 \text{ pasajeros}$$

$$\text{Exactitud (\%)} = \left(1 - \frac{1 + 1 + 1 + 1}{33}\right) \times 100 = 87.9\%$$

Vuelta 2 (24 estaciones detectadas)

$$\text{Error absoluto} = \frac{7 + 1 + 5 + 1}{24} = \frac{14}{24} \approx 0.58 \text{ pasajeros}$$

$$\text{Exactitud (\%)} = \left(1 - \frac{7 + 1 + 5 + 1}{34}\right) \times 100 = 41.1\%$$

Vuelta 3 (25 estaciones detectadas)

$$\text{Error absoluto} = \frac{2 + 1 + 1}{25} = \frac{4}{25} = 0.16 \text{ pasajeros}$$

$$\text{Exactitud (\%)} = \left(1 - \frac{2 + 1 + 1}{28}\right) \times 100 = 85.7\%$$

Resultado Global

$$\text{Error absoluto promedio} = \frac{4 + 14 + 4}{25 + 24 + 25} = \frac{22}{74} \approx 0.3 \text{ pasajeros}$$

$$\text{Exactitud promedio} = \frac{87.9\% + 41.1\% + 85.7\%}{3} \approx 71.6\%$$

Resultados clave:

- **Geolocalización:** Exactitud promedio de $\approx 91.36\%$, con inconsistencias en estaciones 7, 8, 15 y 27.
- **Conteo de pasajeros:** Exactitud promedio de $\approx 71.6\%$.

Las únicas complicaciones ocurrieron durante pruebas realizadas fuera del pumabús, especialmente cuando la luz solar era particularmente intensa. En esos casos, el sensor permanecía activado de manera continua, lo cual se atribuye, probablemente, a la interferencia de los rayos infrarrojos presentes en la radiación solar.

Prueba de callback Sigfox-Losant

La vinculación entre Sigfox Backend y la plataforma Losant funcionó correctamente. En todos los recorridos realizados, los datos registrados en ambas plataformas coincidieron y resultaron congruentes.

subidas ↕	bajadas ↕	pasajeros ↕	estacion ↕	position ↕
0	15	15	27	19.323685,-99.174866
0	0	30	26	19.323685,-99.174866
0	0	30	25	19.323685,-99.174866
0	0	30	24	19.323685,-99.174866
0	0	30	23	19.323685,-99.174866
2	0	30	22	19.323685,-99.174866
0	0	28	21	19.323685,-99.174866
3	0	28	20	19.323685,-99.174866
3	0	25	19	19.323685,-99.174866
1	0	22	18	19.323685,-99.174866
4	0	21	17	19.323685,-99.174866
0	0	17	16	19.323685,-99.174866
0	0	17	14	19.323685,-99.174866
1	1	17	13	19.323685,-99.174866
1	1	17	12	19.323685,-99.174866
0	0	17	11	19.323685,-99.174866
1	0	17	10	19.323685,-99.174866
2	0	16	9	19.323685,-99.174866
0	1	14	7	19.323685,-99.174866
0	0	15	6	19.323685,-99.174866

Figura 10.3: Registro de mensajes recibidos en Losant.

La Tabla de la Figura 10.3 muestra el conjunto de datos recibidos, registrados y desplegados en Losant, correspondientes al número de pasajeros a bordo, así como al conteo de ascensos y descensos por estación.

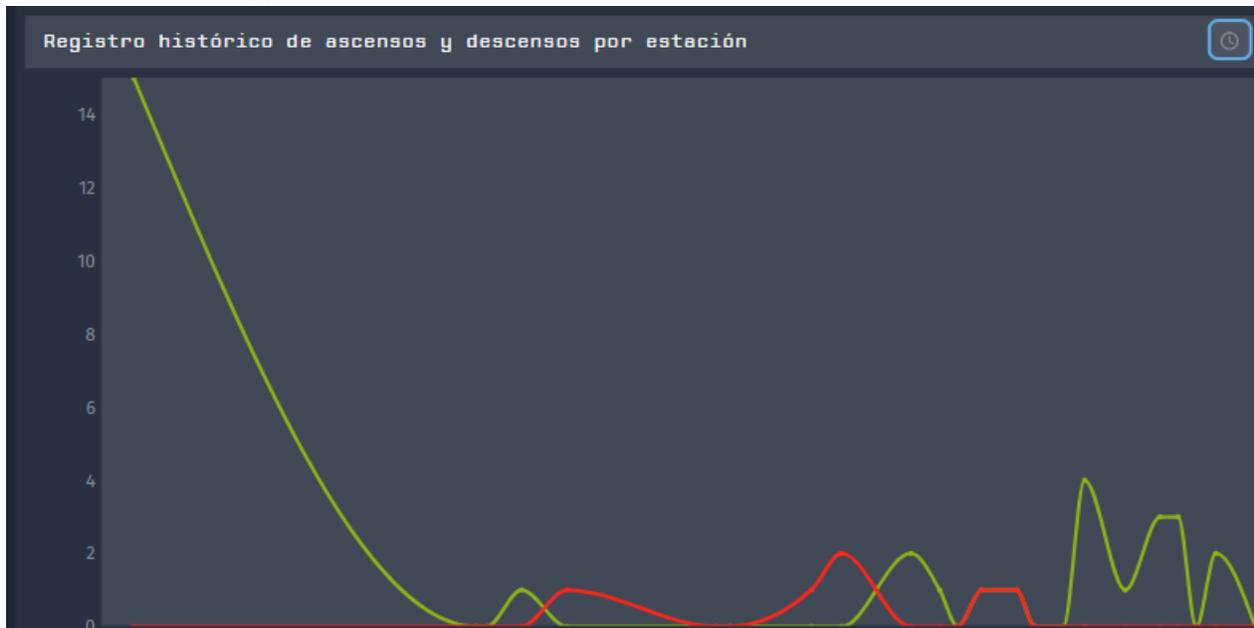
La gráfica de pasajeros a bordo (Figura 10.4 a) refleja un aumento progresivo, alcanzando un máximo de 30 pasajeros, seguido por una disminución sostenida hasta volver al estado de unidad sin pasajeros. Esta tendencia coincide con los datos de la tabla, donde se observa una

acumulación inicial de pasajeros y la descarga al acercarse a la base.

Por su parte, la gráfica de ascensos y descensos (Figura 10.4 b) evidencia mayor actividad de ascensos (verde) y descensos (rojo) en distintos puntos del recorrido. Estos son puntos clave de afluencia, lo cual resulta útil para la toma de decisiones operativas.



(a) Gráfica de pasajeros en el tiempo de recorridos de prueba.



(b) Gráfica de ascensos y descensos en el tiempo de recorridos de prueba.

Figura 10.4: Gráficas de los datos recibidos en Losant durante las pruebas.

11 CONCLUSIONES

El sistema desarrollado fue capaz de cumplir con los principales requerimientos definidos en la Tabla 3.4, especialmente aquellos relacionados con la necesidad de contar con información para mejorar la eficiencia operativa del servicio de transporte Pumabús (Requerimiento 1), y la posibilidad de conocer la ubicación de las unidades y el número de personas a bordo, de manera actualizada (Requerimientos 2, 23, 25).

Durante las pruebas de campo, el **módulo GPS NEO-6MV2** mostró un desempeño adecuado para la detección de las geocercas establecidas de 7 m de radio, logrando identificar correctamente el paso por las estaciones (Requerimientos 3, 4, 17, 24). Aunque en algunos casos la entrada a la geocerca se registró con ligeras variaciones respecto a la ubicación esperada, estas desviaciones fueron consideradas aceptables dado el nivel de precisión requerido para la aplicación, las características propias del entorno y los escenarios que se viven cotidianamente (Requerimiento 23).

El **sistema de conteo de pasajeros**, basado en **sensores infrarrojos**, demostró un funcionamiento confiable que permite prescindir de registros y anotaciones manuales (Requerimiento 16). La validación de los datos mostró un error relativo aceptable, y la información generada fue útil para estimar el nivel de ocupación a lo largo del recorrido (Requerimientos 2, 5, 6).

En cuanto al despliegue de información, el uso de la plataforma **Losant** como interfaz final permitió presentar los datos de forma gráfica, clara y accesible desde dispositivos móviles (Requerimientos 19, 22). Esto facilita el análisis posterior de patrones como las estaciones más concurridas y los horarios pico, si es que los hay, y abre la posibilidad de optimizar frecuencias de paso en función de la demanda (Requerimientos 1, 14, 8).

Además, el sistema fue montado en una carcasa compacta, con una batería portátil (Requerimientos 20, 21), y su diseño con geometría simple buscó minimizar la interferencia con el espacio físico del pumabús (Requerimientos 26, 27, 28, 29, 30).

En conjunto, los resultados obtenidos permiten concluir que la solución propuesta cumplió con los objetivos establecidos, al satisfacer los requerimientos más relevantes del usuario final y del personal operativo. Se generaron las bases y posee el potencial para evolucionar hacia un proyecto integral de monitoreo y desarrollo de mejoras en la aplicación como parte de un producto final. La aplicación desarrollada, si bien presenta algunas limitaciones, permite visualizar el estado más reciente de las unidades del sistema de transporte Pumabús, tanto en términos del número de personas a bordo como de la última estación registrada.

La solución desarrollada cumplió satisfactoriamente con los objetivos planteados inicialmente, como se detalla a continuación en una comparación entre los objetivos propuestos y los resultados obtenidos.

Objetivo General: Desarrollar un prototipo de sistema capaz de identificar las estaciones de una ruta del servicio Pumabús, durante sus trayectos, realizar el conteo del número de pasajeros a bordo y transmitir esta información actualizada mediante tecnología IoT utilizando la red 0G-Sigfox, con una interfaz accesible para su consulta y análisis.

Resultados Alcanzados:

- Se desarrolló un prototipo funcional que integra:
 - Sensores infrarrojos para conteo de ascensos y descensos
 - Módulo GPS NEO6MV2 para geolocalización
 - Transmisión de datos vía red 0G-Sigfox
 - Plataforma Losant para visualización de información
- El sistema provee información actualizada sobre:
 - Número de pasajeros a bordo
 - Ubicación geográfica de la unidad
 - Última estación en la que estuvo y próxima estación
- Validación satisfactoria en pruebas de campo en la Ruta 3 del Pumabús.

Objetivos Específicos vs. Resultados

1. **Obtener las coordenadas geográficas (latitud y longitud) de la ubicación de una unidad Pumabús a lo largo de su recorrido por la ruta 3 en Ciudad Universitaria.**
 - *Resultado:* Módulo GPS NEO-6MV2 integrando geocercas de 7 m de radio para detección precisa de estaciones; registro continuo de latitud y longitud.
2. **Contar el número de pasajeros que ascienden y descienden de la unidad Pumabús en las estaciones que conforman la ruta 3.**
 - *Resultado:* Sensores infrarrojos E18-D80NK instalados en puerta de acceso; registro exitoso de ascensos y descensos en pruebas de campo; cálculo del número de pasajeros a bordo actualizado.
3. **Procesar los datos recopilados para calcular el número total de pasajeros a bordo y asociar esta información con la estación correspondiente.**
 - *Resultado:* Algoritmo implementado en firmware para:
 - Calcular pasajeros a bordo en cada estación
 - Asociar registro a la estación detectada
4. **Transmitir los datos procesados mediante la red 0G-Sigfox cada vez que la unidad abandone una estación.**
 - *Resultado:* Configuración de transmisión tras salida de cada geocerca; mensajes enviados dentro de las limitaciones de 144 mensajes/día de Sigfox.

5. Desplegar la información en una plataforma IoT con visualizaciones diferenciadas según el tipo de usuario (pasajero, coordinador del servicio o desarrollador), actualizadas conforme se reciban nuevos datos.

- *Resultado:* Interfaz implementada en Losant; visualización de datos actualizados con opción de análisis histórico; diseño escalable para diferentes perfiles de usuario.

El proyecto cumplió satisfactoriamente con los objetivos planteados, al haber desarrollado un sistema funcional que:

1. Provee información actualizada sobre ocupación y ubicación
2. Integra tecnologías IoT de bajo costo y consumo
3. Fue validado en condiciones reales de operación
4. Sienta bases para futuras mejoras y escalamiento

Las limitaciones identificadas no comprometen los logros alcanzados, sino que representan oportunidades de mejora para etapas posteriores del proyecto. El sistema demostró ser técnicamente viable y potencialmente escalable para mejorar la eficiencia del servicio Pumabús.

Existen diversos factores que pudieron afectar el rendimiento y provocar algunos de los errores que se presentaron. En el caso de la geolocalización, el error inherente del módulo GPS y frecuencia de muestreo nominal pueden causar variaciones en la detección del borde de las geocercas. Aunado a esto, el factor del comportamiento de cada operador en los distintos viajes, puede provocar variaciones al momento de la entrada real a las geocercas.

En el caso del conteo, los principales factores que pueden afectar la detección correcta son: el flujo simultáneo de pasajeros por la entrada a la unidad, interferencia lumínica, interferencia por objetos que llevan los pasajeros, velocidad de paso y permanencia de gente frente a los sensores.

La realización del proyecto implicó un proceso completo que abarcó desde la investigación sobre el funcionamiento de sistemas de transporte hasta el análisis comparativo de tecnologías aplicables, pasando por etapas de diseño, planificación, implementación y validación de un sistema funcional. A lo largo del desarrollo, se integraron habilidades técnicas y se tomaron decisiones fundamentadas para resolver problemas reales en un entorno dinámico.

Entre los aprendizajes más significativos se encuentra la comprensión de la complejidad que implica desarrollar soluciones robustas para contextos como el transporte público, así como la importancia del trabajo interdisciplinario y la planificación detallada para superar limitaciones físicas, tecnológicas y operativas.

A nivel personal, este proyecto representó una experiencia formativa profundamente valiosa. Brindó la oportunidad de aportar a una problemática concreta dentro de la comunidad universitaria y fortaleció el compromiso de seguir desarrollando soluciones tecnológicas con impacto social. El proceso fue sumamente enriquecedor y ha dejado bases sólidas para futuros proyectos académicos y profesionales.

11.1 Etapas siguientes del proyecto

Entre las principales áreas de mejora se encuentra la necesidad de implementar un plan de conectividad que posibilite el envío más frecuente de mensajes. Esto resulta particularmente relevante debido a las restricciones propias de la red Sigfox, la cual permite un máximo de 144 mensajes por día. A pesar de estas limitaciones, la solución desarrollada cumplió con el alcance previsto en la etapa inicial del proyecto.

Aunque el plan base de Sigfox contempla un límite de 140 mensajes de subida por día (equivalente a un mensaje cada 10 minutos aproximadamente), la plataforma ofrece opciones de escalabilidad mediante la contratación de planes superiores o acuerdos con el operador regional. Esta flexibilidad permite adaptar el sistema en caso de que, en etapas posteriores del proyecto, se busque aumentar la frecuencia de transmisión o incorporar dispositivos adicionales sin comprometer la viabilidad técnica.

Cabe destacar que la problemática identificada no se resuelve únicamente mediante el monitoreo de una sola ruta. No obstante, contar con información actualizada sobre la capacidad de una unidad Pumabús de la ruta 3 representa un avance significativo. Además, los datos obtenidos muestran un nivel de precisión suficiente para proyectar que la solución podría escalarse exitosamente, siempre que se amplíe su cobertura a un mayor número de rutas, unidades y puertas monitoreadas.

Inicialmente, el dispositivo fue elaborado para detectar el paso de usuarios en una sola puerta del pumabús. Este diseño permitió un enfoque simple y funcional, pero se reconoce la necesidad de expandir la capacidad del sistema, a través de la instalación de más sensores y su consideración en el código, para cubrir múltiples puntos de ascenso y descenso en el pumabús y registrar de manera precisa las subidas, bajadas y total de pasajeros en la unidad.

Se recomienda ajustar el radio de las geocercas de forma específica para cada estación, ya que algunas requieren ampliarlo y otras reducirlo. Esto depende del tamaño de la estación, así como de la proximidad con otras estaciones, con el fin de evitar interferencias entre geocercas que puedan generar errores en la detección. Asimismo, se sugiere considerar solicitar al personal que se instruya a los conductores pasar y detenerse claramente en cada estación y hacer alto total que favorezca la correcta identificación de la ubicación.

La integración de módulos GPS en dos puertas, o uso de módulos de mejores características, podría mejorar la precisión en la detección de ingresos y salidas del pumabús en una geocerca.

Para validar completamente el sistema se recomienda:

- Pruebas en diferentes horarios (matutino, vespertino, nocturno)
- Evaluación en condiciones climáticas variadas (lluvia, sol intenso)
- Monitoreo continuo durante periodos completos de operación (1-2 semanas)
- Comparación con conteos manuales paralelos para validación
- Pruebas con diferentes configuraciones de sensores (ángulos, separación)

Considerando los resultados obtenidos, para el caso específico del servicio Pumabús, donde el principal requerimiento es proveer información útil para la toma de decisiones operativas y de

usuarios, un sistema de estimación por niveles (vacío, pocos pasajeros, medio lleno, lleno, etc) podría ser suficiente y potencialmente más robusto. Sin embargo, para fines de planeación a mediano y largo plazo, el conteo exacto sigue siendo valioso. Para ello se pueden considerar tecnologías que empleen cámaras con algoritmos de visión artificial.

Bibliografía

- [1] Oracle, “¿Qué es IoT?”, [En línea]. Disponible en: <https://www.oracle.com/mx/internet-of-things/what-is-iot/>. [Accedido: 19-mar-2025].
- [2] IBM, “Internet de las Cosas”, [En línea]. Disponible en: <https://www.ibm.com/mx-es/topics/internet-of-things>. [Accedido: 19-mar-2025].
- [3] Unión Internacional de Telecomunicaciones (ITU), “El IFT presenta encuesta usuarios de servicios de telecomunicaciones sobre su conocimiento del IoT”, [En línea]. Disponible en: <https://www.ift.org.mx/comunicacion-y-medios/comunicados-ift/es/el-ift-presenta-encuesta-usuarios-de-servicios-de-telecomunicaciones-sobre-su-conocimiento-del>. [Accedido: 19-mar-2025].
- [4] P. Gokhale, “Introduction to IoT”, IARJSET, [En línea]. Disponible en: https://www.researchgate.net/profile/Omkar-Bhat/publication/330114646_Introduction_to_IOT/links/5c2e31cf299bf12be3ab21eb/Introduction-to-IOT.pdf. [Accedido: 19-mar-2025].
- [5] Dirección General de Servicios Generales y Movilidad, “Movilidad UNAM”, [En línea]. Disponible en: <https://www.dgsgm.unam.mx/conocenos.html>. [Accedido: 12-abr-2025].
- [6] Universidad Autónoma del Estado de Hidalgo, “Definición de red de comunicaciones y su importancia”, [En línea]. Disponible en: http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro27/135_definicion_de_red_de_comunicaciones_y_su_importancia.html. [Accedido: 19-mar-2025].
- [7] K. Mekki, E. Bajic, F. Chaxel y F. Meyer, “A comparative study of LPWAN technologies for large-scale IoT deployment”, *ICT Express*, vol. 5, n.º 1, pp. 1–7, 2019. [En línea]. Disponible en: <https://doi.org/10.1016/j.icte.2017.12.005>. [Accedido: 19-mar-2025].
- [8] K. T. Ulrich y S. D. Eppinger, *Diseño y desarrollo de productos*, 5ta ed. Ciudad de México: McGraw-Hill, 2012.
- [9] Asana, “What are SMART goals? Examples and Templates [2025]”, [En línea]. Disponible en: <https://asana.com/resources/smart-goals>. [Accedido: 12-ago-2025].
- [10] Rock Content, “Metodología 5W2H”, [En línea]. Disponible en: <https://rockcontent.com/es/blog/metodologia-5w2h/>. [Accedido: 19-mar-2025].
- [11] CT Calidad, “¿En qué consiste la técnica 5W2H?”, [En línea]. Disponible en: <http://ctcalidad.blogspot.com/2017/11/en-que-consiste-la-tecnica-5w2h-de.html>. [Accedido: 19-mar-2025].

- [12] O. Bhat, "Introduction to IoT", ResearchGate, 2019. [En línea]. Disponible en: https://www.researchgate.net/profile/Omkar-Bhat/publication/330114646_Introduction_to_IOT/links/5c2e31cf299bf12be3ab21eb/Introduction-to-IOT.pdf. [Accedido: 19-mar-2025].
- [13] G. Pahl, W. Beitz, J. Feldhusen y K.-H. Grote, *Engineering Design: A Systematic Approach*, 3ª ed., Londres, Reino Unido: Springer, 2007.
- [14] Freepik, "Parada de autobús vacía diseño plano", Freepik, [En línea]. Disponible en: https://www.freepik.es/vector-gratis/parada-autobus-vacia-diseno-plano_2324727.htm. [Imagen modificada]. [Accedido: 19-mar-2025].
- [15] Sigfox, "Devkit v2.0 Data Sheet", [En línea]. Disponible en: https://storage.googleapis.com/public-assets-xd-sigfox-production-338901379285/jcl7peio_devkit-v2.0-data-sheet.pdf. [Accedido: 4-abr-2025].
- [16] ProColombia, "Plataforma del Internet de las Cosas", [En línea]. Disponible en: <https://b2bmarketplace.procolombia.co/es/software-servicios-ti/software-empresarial/plataforma-del-internet-de-las-cosas-23817>. [Accedido: 19-mar-2025].
- [17] Ubidots, "Acerca de Ubidots", [En línea]. Disponible en: <https://es.ubidots.com/about>. [Accedido: 19-mar-2025].
- [18] C. Franco, E. Flores, E. Hernández, J. Cazas y J. Ramírez, "Proyecto Ninzu", Trabajo académico presentado en la asignatura Temas selectos de Ingeniería Mecatrónica, Facultad de Ingeniería, UNAM, 2023.