



UNIVERSIDAD ESTATAL DE BOLÍVAR

CARRERA DE SOFTWARE

TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIEROS EN SOFTWARE

FORMA: PROYECTO TECNOLÓGICO

TEMA:

**PROTOTIPO FUNCIONAL DE GEOLOCALIZACIÓN PARA LA VISUALIZACIÓN EN
TIEMPO REAL DE RUTAS Y PARADAS DEL TRANSPORTE URBANO EN LA CIUDAD
DE GUARANDA.**

AUTORES:

Erick Alexander Alarcón Zumba

Juan Raúl Pasto Patín

DIRECTOR(A):

Ing. Mónica Bonilla Manobanda, MsC.

GUARANDA – ECUADOR

2026

TEMA PROYECTO TECNOLÓGICO

PROTOTIPO FUNCIONAL DE GEOLOCALIZACIÓN PARA LA VISUALIZACIÓN EN TIEMPO REAL DE RUTAS Y PARADAS DEL TRANSPORTE URBANO EN LA CIUDAD DE GUARANDA.

DEDICATORIA

Esta tesis está dedicada a mis padres, quienes siempre creyeron en mí, incluso cuando yo mismo dudé, incluso cuando me equivoqué y las cosas no salieron como esperaba. Nunca necesitaron que fuera perfecto para estar ahí, y eso es algo que llevaré conmigo toda la vida.

Su apoyo constante, su paciencia, sus consejos y su amor incondicional fueron el motor que me impulsó a seguir adelante en los momentos más difíciles de este camino. Cada vez que sentí que las fuerzas me faltaban, recordé que tenía a alguien confiando en mí, y esa confianza fue suficiente para levantarme y continuar.

Gracias por cada sacrificio que hicieron para que pudiera llegar hasta aquí, por cada palabra de aliento cuando más la necesitaba, por comprender mis ausencias, mis preocupaciones y los desafíos que implicó este proceso. Todo lo que han hecho por mí ha dejado una huella que nunca podré olvidar.

También quiero expresar mi más sincero agradecimiento a todas las personas que, de una u otra manera, formaron parte de este recorrido. A mi familia, por su cariño y comprensión; a mis docentes, por compartir sus conocimientos y orientarme con paciencia; y a mis amigos, por brindarme su apoyo y motivación durante este camino. Cada consejo, cada palabra de ánimo y cada gesto de confianza contribuyeron a que este sueño se hiciera realidad.

Todo lo que soy y todo lo que he logrado lleva el reflejo del amor, el esfuerzo y los valores que ustedes me han enseñado. Este logro no es solamente mío; también les pertenece a quienes caminaron a mi lado y nunca dejaron de creer en mí.

Con todo mi amor, respeto y profunda gratitud, les dedico este logro. Gracias por ser mi mayor inspiración y por demostrarme que, con esfuerzo, perseverancia y el apoyo de quienes nos aman, los sueños pueden hacerse realidad.

Erick Alexander Alarcón Zumba

Este trabajo de tesis está dedicado, en primer lugar, a mi familia, quienes han sido el pilar fundamental en cada etapa de mi vida. Gracias a su apoyo constante, sus consejos y su confianza, he podido seguir adelante incluso en los momentos más difíciles.

De manera muy especial, dedico este logro a mi madre, quien ha sido mi mayor inspiración. Su esfuerzo, sacrificio y amor incondicional han sido la fuerza que me ha impulsado a no rendirme. Cada paso que doy es también gracias a ella, por enseñarme a luchar por mis sueños y a creer en mí mismo.

Juan Raúl Pasto Patín

AGRADECIMIENTO

Al culminar esta importante etapa de nuestra formación profesional, queremos expresar nuestro más sincero agradecimiento a todas las personas que hicieron posible la realización de este trabajo de titulación.

En primer lugar, agradecemos a Dios por brindarnos la fortaleza, la sabiduría y la perseverancia necesarias para alcanzar esta meta. Asimismo, expresamos nuestra gratitud a nuestras familias por su amor, apoyo incondicional y motivación constante durante todo este proceso.

De manera especial, agradecemos a la ingeniera Mónica Bonilla, directora de nuestro trabajo de titulación, por su guía, paciencia y valiosos conocimientos durante el desarrollo de la investigación titulada "Prototipo funcional de geolocalización para la visualización en tiempo real de rutas y paradas del transporte urbano en la ciudad de Guaranda".

También extendemos nuestro agradecimiento a los ingenieros Edgar Rivadeneira y Rodrigo del Pozo por sus valiosos aportes y enseñanzas, que contribuyeron significativamente al desarrollo de este proyecto.

Finalmente, agradecemos a nuestros profesores y compañeros por compartir con nosotros este camino de aprendizaje y crecimiento profesional.

Erick Alexander Alarcón Zumba, Juan Raul Pasto Patín

CERTIFICADO DE VALIDACIÓN



FACULTAD DE CIENCIAS
ADMINISTRATIVAS,
GESTIÓN EMPRESARIAL
E INFORMÁTICA

CERTIFICADO DE VALIDACIÓN

Ing. Mónica Bonilla, Ing. Edgar Rivadeneira e Ing. Rodrigo Del Pozo en su orden de directora y Pares Académicos del Trabajo de Integración Curricular "PROTOTIPO FUNCIONAL DE GEOLOCALIZACIÓN PARA LA VISUALIZACIÓN EN TIEMPO REAL DE RUTAS Y PARADAS DEL TRANSPORTE URBANO EN LA CIUDAD DE GUARANDA." desarrollado por los estudiantes Erick Alexander Alarcon Zumba y Juan Raul Pasto Patin

CERTIFICAN

Que, luego de revisado el Trabajo de Integración Curricular en su totalidad, cumple con las exigencias académicas de la carrera SOFTWARE, por lo tanto, autorizamos su presentación y defensa.

Guaranda, 04 de septiembre del 2026



Ing. Mónica Bonilla

Ing. Mónica Bonilla

Directora



Ing. Rodrigo Del Pozo

Ing. Rodrigo Del Pozo

Par Académico



Ing. Edgar Rivadeneira

Ing. Edgar Rivadeneira

Par Académico

Dirección: Av. Ernesto Che Guevara y Gabriel Secaira
Guaranda-Ecuador
Teléfono: (593) 3220 6059
www.ueb.edu.ec

DERECHOS DE AUTOR

Nosotros, **ERICK ALEXANDER ALARCON ZUMBA Y JUAN RAUL PASTO PATÍN**, portadores de las cédulas de identidad No. **1805082888 y 0250171923**, respectivamente, en calidad de autores y titulares de los derechos morales y patrimoniales del Trabajo de Titulación: **PROTOTIPO FUNCIONAL DE GEOLOCALIZACIÓN PARA LA VISUALIZACIÓN EN TIEMPO REAL DE RUTAS Y PARADAS DEL TRANSPORTE URBANO EN LA CIUDAD DE GUARANDA**, modalidad: **PROYECTO TECNOLÓGICO**, de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, concedemos a favor de la Universidad Estatal de Bolívar una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Asimismo, autorizamos a la Universidad Estatal de Bolívar para que realice la digitalización y publicación de este trabajo de titulación en el Repositorio Digital, de conformidad con lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Declaramos que la obra objeto de la presente autorización es original en su forma de expresión y no infringe derechos de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.



Nombres y Apellidos
Erick Alexander Alarcon Zumba



Nombres y Apellidos
Juan Raúl Pasto Patín

ÍNDICE DE CONTENIDOS

DEDICATORIA.....	III
AGRADECIMIENTO	V
CERTIFICADO DE VALIDACIÓN	VI
ÍNDICE DE TABLAS	X
ÍNDICE DE ILUSTRACIONES	XIII
INTRODUCCIÓN	1
RESUMEN	3
ABSTRACT.....	4
PISIYACHISQA	5
CAPÍTULO I	6
FORMULACIÓN GENERAL DEL PROYECTO.....	6
1.1 Tema	6
1.2 Descripción del Problema	6
1.3 Justificación.....	7
1.4 Objetivos: General y Específicos	8
1.4.1 Objetivo General	8
1.4.2 Objetivos Específicos	8
CAPÍTULO II	9
MARCO TEÓRICO.....	9
2.1 Antecedentes.....	9
2.2 Científico	12
2.3 Conceptual.....	1
2.4 Legal.....	5

2.5 Georreferencial.....	12
CAPÍTULO III.....	13
METODOLOGÍA	13
3.1 Metodología de Desarrollo de Software.....	13
3.2 Técnicas e Instrumentos de Recopilación de Datos	16
CAPÍTULO IV.....	17
INGENIERÍA DEL PROYECTO	17
4.1 Planificación.....	17
4.1.1 Alcance del Producto	17
4.1.2 Restricciones del sistema.....	20
4.2 SPRINT 1: ANÁLISIS Y LEVANTAMIENTO DE REQUERIMIENTOS	22
4.2.1 Objetivo del Sprint 1	22
4.2.2 Actividades Principales	22
4.2.3 Sprint Backlog	22
4.2.4 Resultados del Sprint 1	23
4.3 SPRINT 2: DISEÑO DEL SISTEMA.....	36
4.3.1 Objetivo del Sprint 2	36
4.3.2 Actividades Principales	36
4.3.3 Sprint Backlog	36
4.3.4 Resultados del Sprint 2	37
4.4 SPRINT 3: DESARROLLO DEL SISTEMA.....	79
4.4.1 Objetivo del Sprint 3	79
4.4.2 Actividades Principales	79
4.4.3 Sprint Backlog	81
4.4.4 Resultados del Sprint 3	84

4.5 SPRINT 4: VALIDACIÓN, PRUEBAS E IMPLEMENTACIÓN	98
4.5.1 Objetivo del Sprint 4	98
4.5.2 Actividades Principales	98
4.5.3 Sprint Backlog	99
4.5.4 Ejecución y resultados	100
CONCLUSIONES	115
RECOMENDACIONES	116
BIBLIOGRAFÍA	117
ANEXOS	121
Anexo 1.	121
Cronograma del Proyecto	121
Anexo 2.	122
Detalle del Presupuesto Estimado	122
Anexo 3.	123
Instrumentos de recopilación de datos (Entrevista).....	123
Anexo 4.	127
Repositorios del Código Fuente en GitHub.....	127
Anexo 5.	129
Manual de Usuario - Panel Web Administrativo	129
Anexo 6.	181
MANUAL DE DESCARGA E INSTALACIÓN EN TELÉFONO	181
Anexo 7	192
Fotografías y evidencias	192
Anexo 8	193
Certificado Antiplagio	193

ÍNDICE DE TABLAS

Tabla 1. Características principales de las metodologías de desarrollo de software.....	15
Tabla 2. Sprint Backlog del Sprint 1	22
Tabla 3. Historia de usuario 1	23
Tabla 4 Historia de usuario 2	24
Tabla 5. Historia de usuario 3	24
Tabla 6. Historia de usuario 4	25
Tabla 7. Historia de usuario 5	25
Tabla 8. Historia de usuario 6	26
Tabla 9. Historia de usuario 7	26
Tabla 10. Historia de usuario 8	27
Tabla 11. Historia de usuario 9	27
Tabla 12. Historia de usuario 10	28
Tabla 13. Historia de usuario 11	28
Tabla 14. Historia de usuario 12	29
Tabla 15. Historia de usuario 13	29
Tabla 16. Historia de usuario 14	30
Tabla 17. Historia de usuario 15	31
Tabla 18. Historia de usuario 16	31
Tabla 19. Historia de usuario 17	32
Tabla 20. Historia de usuario 18	32
Tabla 21 Historia de usuario 19	33
Tabla 22 Historia de usuario 20	33
Tabla 23 Historia de usuario 21	34
Tabla 24 Historia de usuario 22	34
Tabla 25. Stack Tecnológico Definido — Sprint 1	35
Tabla 26. Sprint Backlog del Sprint 2.....	36
Tabla 27 Aplicación de los lineamientos de diseño en entornos web y móvil.....	50
Tabla 28. Sprint Backlog del Sprint 3.....	81
Tabla 29. Sprint Backlog del Sprint 4.....	99
Tabla 30 Pruebas funcionales del sistema UniBolivar Transit.....	101

Tabla 31 Prueba de rendimiento del endpoint GPS (P-20).	106
Tabla 32 Prueba de rendimiento del endpoint de ubicación GPS (P-21).....	107
Tabla 33 Prueba de GPS en segundo plano (P-22).	107
Tabla 34 Análisis de seguridad con OWASP ZAP (P-23).....	109
Tabla 35 Pruebas de compatibilidad en Android.	110
Tabla 36 Tiempos de respuesta de endpoints API REST.	113
Tabla 37. Tabla del presupuesto estimado para el desarrollo del proyecto.	122

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Georreferencial	12
Ilustración 2 Diagrama Entidad-Relación de UniBolívar Transit.....	38
Ilustración 3 Modelo relacional de UniBolívar Transit	39
Ilustración 4 Diagrama de Clases de UniBolívar Transit.....	40
Ilustración 5 Diagramas de Secuencia Gestión del Administrador.....	41
Ilustración 6 Diagrama de secuencia del proceso de inicio de sesión (chofer y administrador). .	42
Ilustración 7 Diagrama de secuencia del proceso de geolocalización y notificación al pasajero.	43
Ilustración 8 Diagrama de secuencia del proceso de suscripción a notificaciones de parada.	44
Ilustración 9 Diagrama de secuencia del proceso de registro de viaje y marcaciones	45
Ilustración 10 Diagrama de Actividades del Seguimiento y Notificación de Buses en Tiempo Real	46
Ilustración 11 Diagrama de Componentes de tu sistema UniBolívar Transit.	47
Ilustración 12 Diagrama de Despliegue de UniBolívar Transit	48
Ilustración 13 Sistema web: Página de inicio - vista principal	51
Ilustración 14 Página de inicio - funcionalidades e información institucional.....	52
Ilustración 15 Inicio de sesión del Panel Administrativo.....	53
Ilustración 16 Dashboard del Panel Administrativo	53
Ilustración 17 Gestión de usuarios.....	54
Ilustración 18 Gestión de líneas	55
Ilustración 19 Gestión de rutas	57
Ilustración 20 Gestión de paradas	57
Ilustración 21 Gestión de horarios	57
Ilustración 22 Monitoreo en tiempo real.....	57
Ilustración 23 Despacho.....	58
Ilustración 24 Flota de buses.....	59
Ilustración 25 Reportes por unidad	59
Ilustración 26 Logo de la aplicación	61
Ilustración 27 Pantalla de bienvenida y carga (Splash)	63
Ilustración 28 Mapa principal (pestaña Rutas)	64

Ilustración 29 Lista de líneas / Rutas	65
Ilustración 30 Paradas y tiempos de llegada	66
Ilustración 31 Horarios por línea sentido de Ida y Vuelta	67
Ilustración 32 Alertas activas	68
Ilustración 33 Alertas activas con contexto en el mapa	69
Ilustración 34 Mapa completo del recorrido	70
Ilustración 35 Logo de la aplicación	71
Ilustración 36 Pantalla de bienvenida y carga (splash)	73
Ilustración 37 Inicio de sesión	74
Ilustración 38 Selección de ruta y sentido	75
Ilustración 39 Pantalla «En servicio»	76
Ilustración 40 Estado de la unidad	77
Ilustración 41 Mi historial	78
Ilustración 42 T-13: API REST — Autenticación de choferes y administrador	84
Ilustración 43 T-14: API REST — Geolocalización y WebSocket en tiempo real	85
Ilustración 44 T-15: API REST — Gestión de líneas, rutas y paradas	86
Ilustración 45 T-16: API REST — Notificaciones push con FCM	87
Ilustración 46 T-17: API REST — Registro de viajes y marcaciones	88
Ilustración 47 T-18: API REST — Reportes de puntualidad	89
Ilustración 48 T-19: API REST — Despacho y permisos de choferes	90
Ilustración 49 T-20: Panel administrativo web — Dashboard y mapa	91
Ilustración 50 T-21: Panel administrativo web — Gestión de usuarios, líneas, rutas, paradas y horarios	92
Ilustración 51 T-22: Panel administrativo web — Despacho y reportes PDF	93
Ilustración 52 T-23: Aplicación móvil para pasajeros — Mapa, rutas, ETA y alertas	94
Ilustración 53 T-24: Aplicación móvil para choferes — GPS en segundo plano e historial de viajes	95
Ilustración 54 T-25 y T-26: Integración WebSocket Soketi y notificaciones FCM	96
Ilustración 55 T-27, T-28 y T-29: Configuración del servidor, base de datos y SSL	97
Ilustración 56 P-20 — Rendimiento endpoint GPS con 48 choferes	105

Ilustración 57 Prueba de rendimiento del endpoint de ubicación bajo carga concurrente — P-21	106
Ilustración 58 Prueba de seguridad	108
Ilustración 59 P-28 — /api/auth/login-chofer	111
Ilustración 60 P-29 — /api/marcaciones/reporte	112
Ilustración 61 P-30 — /api/chofer/ubicación	113
Ilustración 62 Sistema en el servidor de producción DigitalOcean Ubuntu 22.04	114
Ilustración 63 Cronograma (Gantt)	121
Ilustración 64 Página de inicio	131
Ilustración 65 Inicio de sesión	132
Ilustración 66 Panel Web Administrativo - Inicio	133
Ilustración 67 Gestión de Usuarios	134
Ilustración 68 Modal “Nuevo Usuario”	136
Ilustración 69 Gestión de Líneas	137
Ilustración 70 Modal “Nueva Línea	138
Ilustración 71 Gestión de Rutas - Selección de línea	140
Ilustración 72 Gestión de Rutas - Detalle IDA/VUELTA	141
Ilustración 73 Gestión de Paradas - Selección de sentido	142
Ilustración 74 Gestión de Paradas - Listado y acciones	143
Ilustración 75 Gestión de Horarios - Selección	144
Ilustración 76 Monitoreo en Tiempo Real	145
Ilustración 77 Despacho - Programación diaria	146
Ilustración 78 Flota de Buses	147
Ilustración 79 Reporte de Viajes	148
Ilustración 80 Reportes por Unidad	149
Ilustración 81 Bienvenida y carga (Splash)	152
Ilustración 82 Mapa principal (pestaña Rutas)	154
Ilustración 83 Lista de líneas / Rutas	156
Ilustración 84 Paradas y tiempos de llegada	158
Ilustración 85 Horarios por línea, sentido y día	160
Ilustración 86 Mis alertas activas	162

Ilustración 87 Alertas activas con contexto en el mapa	164
Ilustración 88 Mapa completo del recorrido	166
Ilustración 89 Bienvenida y carga (Splash)	169
Ilustración 90 Inicio de sesión	171
Ilustración 91 Selección de ruta y sentido	173
Ilustración 92 Pantalla En servicio	175
Ilustración 93 Estado de la unidad	177
Ilustración 94 Mi historial.....	179
Ilustración 95 Pantalla principal del portal web de UniBolívar Transit para seleccionar la App Pasajero o la App Chofer.....	181
Ilustración 96 Resultado de búsqueda del sitio oficial de UniBolívar Transit.....	182
Ilustración 97 Portal web oficial	183
Ilustración 98 Confirmación de instalación de la App Pasajero.	184
Ilustración 99 Mensaje de instalación completada de la App Pasajero.	185
Ilustración 100 Solicitud de permiso de notificaciones	186
Ilustración 101 Solicitud de permiso de ubicación.	187
Ilustración 102 Confirmación de instalación de la App Chofer.....	188
Ilustración 103 Validación de seguridad de la App Chofer mediante Google Play Protect.....	189
Ilustración 104 Mensaje de instalación completada de la App Chofer.	190
Ilustración 105 fotografías de reuniones mantenidas con el gerente de la Cooperativa Universidad de Bolívar	192

INTRODUCCIÓN

El transporte urbano constituye uno de los servicios públicos más demandados en las ciudades intermedias del Ecuador. En este contexto, la ciudad de Guaranda, capital de la provincia de Bolívar, cuenta con la Cooperativa de Transporte Urbano Universidad de Bolívar como uno de los principales prestadores del servicio de transporte público colectivo. Esta cooperativa opera múltiples líneas de recorrido que conectan los principales sectores de la ciudad, atendiendo diariamente a un número considerable de usuarios que dependen de este servicio para desarrollar sus actividades cotidianas.

A pesar de la importancia del servicio que ofrece, la cooperativa continúa operando bajo un modelo de gestión predominantemente manual y analógico. Los choferes no disponen de herramientas tecnológicas que permitan registrar automáticamente sus recorridos ni transmitir su ubicación en tiempo real. De igual manera, los pasajeros carecen de información actualizada sobre la ubicación de las unidades, los tiempos estimados de llegada a las paradas y el estado general del servicio. Esta situación ocasiona una experiencia de usuario poco satisfactoria, tiempos de espera inciertos especialmente durante los fines de semana, cuando disminuye el número de unidades en circulación y una progresiva pérdida de confianza por parte de la ciudadanía en el sistema de transporte.

En el ámbito administrativo, la cooperativa tampoco dispone de herramientas digitales que faciliten el monitoreo del cumplimiento de horarios, la identificación de retrasos recurrentes o la generación automática de reportes sobre el desempeño de las unidades y sus conductores. La gestión se realiza mediante registros físicos y canales de comunicación informales, lo que dificulta la toma de decisiones basada en información objetiva y limita la capacidad de la organización para aplicar de manera eficiente y transparente su reglamento interno relacionado con el control de horarios y la operación del servicio.

Frente a esta problemática, el presente proyecto propone el desarrollo de **UniBolívar Transit**, un prototipo funcional de geolocalización para el transporte urbano que integra tres componentes principales: una aplicación móvil para pasajeros, una aplicación móvil para choferes y un panel administrativo web. Estos componentes se comunican en tiempo real mediante la tecnología **WebSocket**, permitiendo la actualización continua de la información sin necesidad de adquirir dispositivos GPS especializados.

La solución aprovecha el dispositivo móvil del conductor como fuente de geolocalización, eliminando la dependencia de hardware adicional y reduciendo significativamente los costos de implementación en comparación con las soluciones tradicionales basadas en equipos GPS dedicados. Esta característica convierte a **UniBolívar Transit** en una alternativa tecnológica viable, accesible y escalable para cooperativas de transporte urbano de mediana escala, como la Cooperativa de Transporte Urbano Universidad de Bolívar, contribuyendo a mejorar la calidad del servicio, optimizar la gestión operativa y brindar información oportuna tanto a los usuarios como a la administración.

RESUMEN

El presente proyecto de titulación describe el diseño, desarrollo e implementación de UniBolívar Transit, un sistema integral de geolocalización en tiempo real para la visualización de rutas, paradas y unidades de transporte urbano de la Cooperativa de Transporte Urbano Universidad de Bolívar, en la ciudad de Guaranda, provincia de Bolívar, Ecuador.

El sistema está compuesto por tres componentes: una aplicación móvil para pasajeros, una aplicación móvil para choferes que transmite la ubicación GPS en tiempo real incluso en segundo plano, y un panel administrativo web. El backend fue desarrollado con Laravel 11, PHP 8.4, PostgreSQL 16 y Soketi como servidor WebSocket.

Mediante la metodología ágil Scrum, el proyecto se desarrolló en cuatro sprints: análisis y levantamiento de requerimientos, diseño del sistema, desarrollo e integración de módulos, y validación, pruebas e implementación. El sistema fue desplegado en un servidor DigitalOcean con dominio unibolivartransit.com y certificado SSL/HTTPS mediante Let's Encrypt.

Los resultados demuestran que el sistema cumple con los requerimientos funcionales establecidos. Las pruebas con k6 demostraron un tiempo de respuesta de 2.71 segundos con 48 choferes simultáneos y 0% de fallos. Las pruebas de seguridad con OWASP ZAP no detectaron vulnerabilidades de nivel alto, y las pruebas de compatibilidad confirmaron el funcionamiento correcto en Android 10, 11, 12 y 13.

ABSTRACT

This thesis project describes the design, development, and implementation of UniBolívar Transit, a comprehensive real-time geolocation system for displaying routes, stops, and urban transit vehicles operated by the Universidad de Bolívar Urban Transit Cooperative in the city of Guaranda, Bolívar Province, Ecuador.

The system consists of three components: a mobile app for passengers, a mobile app for drivers that transmits real-time GPS location even when running in the background, and a web-based administrative dashboard. The backend was developed using Laravel 11, PHP 8.4, PostgreSQL 16, and Soketi as the WebSocket server.

Using the Scrum agile methodology, the project was developed in four sprints: analysis and requirements gathering, system design, module development and integration, and validation, testing, and implementation. The system was deployed on a DigitalOcean server with the domain unibolivartransit.com and an SSL/HTTPS certificate provided by Let's Encrypt.

The results show that the system meets the established functional requirements. Tests using k6 showed a response time of 2.71 seconds with 48 simultaneous drivers and 0% failures. Security tests with OWASP ZAP did not detect any high-level vulnerabilities, and compatibility tests confirmed proper operation on Android 10, 11, 12, and 13.

PISIYACHISQA

Kay tesis llamkayqa UniBolívar Transit nisqapa ruwayninmanta, wiñachisqanmanta, ruwayninmantam willakun, chaymi tukuy ima sistema geolocalización nisqa chiqap pachapi qawarinapaq, Cooperativa de Transporte Urbano Universidad de Bolívar nisqapa, Guaranda llaqtapi, Bolívar provinciapi, Ecuador suyupi, ñankunata, paradakunata, unidad de transporte urbano nisqakunatapas qawarinapaq.

Kay sistema kimsa componentes kaqmanta ruwasqa: huk aplicación móvil kaqmanta pasajerkunapaq, huk aplicación móvil kaqmanta choferkunapaq mayqinchus GPS tarikuyninta chiqap pachapi apachin, qhipamanpas, chaymanta huk panel administrativo web kaqpi ruwasqa. Qhipaman Laravel 11, PHP 8.4, PostgreSQL 16, Soketi ima WebSocket sirwinq hina ruwasqa karqan.

Scrum ágil metodología kaqwan, proyecto tawa sprint kaqpi ruwasqa karqa: requisitos t'aqwiychaymanta huñuy, sistema diseño, módulo ruway chaymanta tinkiynin, chaymanta validación, pruebakuna chaymanta implementación. Llamkana huk DigitalOcean sirwiqqi churasqa karqa kay dominio unibolivartransit.com kaqwan chaymanta huk SSL/HTTPS certificado kaqwan Let's Encrypt kaqwan.

Chay ruwasqakunam qawarichin chay sistemaqa takyasqa ruway kamachikuykunata huntasqanmanta. k6 kaqwan prueba ruwayqa 2,71 segundos kutichiy pachata rikuchirqa 48 simultáneos conductores kaqwan chaymanta 0% tasa de fracaso kaqwan. OWASP ZAP kaqwan harkasqa pruebakuna mana ima hatun pata mana allinkuna tarisqachu, chaymanta tupachiy prueba allin llamk'ayta Android 10, 11, 12 chaymanta 13 kaqpi takyachirqa.

CAPÍTULO I

FORMULACIÓN GENERAL DEL PROYECTO

1.1 Tema

Prototipo funcional de geolocalización para la visualización en tiempo real de rutas y paradas del transporte urbano en la ciudad de Guaranda.

1.2 Descripción del Problema

En la actualidad, la Cooperativa de Transporte Urbano Universidad de Bolívar, que brinda el servicio de movilidad en la ciudad de Guaranda, atraviesa una serie de dificultades que afectan directamente la calidad del servicio que reciben sus usuarios.

Ante estas carencias, los ciudadanos se ven obligados a depender de información verbal transmitida, de una señalización física muchas veces incompleta o deteriorada, o simplemente de la experiencia acumulada de otros pasajeros. Esta situación genera incertidumbre, desorganización y una pérdida considerable de tiempo durante los desplazamientos diarios de la población.

El desconocimiento de los horarios y rutas de operación provoca que los ciudadanos se vean forzados a esperar largos periodos de tiempo en las paradas, especialmente los fines de semana en los que no operan todas las unidades de transporte. Esta espera innecesaria genera malestar e incomodidad a los usuarios, así como aglomeraciones en espacios públicos que podrían evitarse con una mejor comunicación de los horarios de atención.

A esto se suma la ausencia total de un sistema tecnológico que permita visualizar la ubicación de los buses en tiempo real. Sin esta herramienta, los usuarios no tienen forma de estimar con precisión el tiempo de llegada de una unidad a una parada específica, lo que incrementa la sensación de abandono e ineficiencia del servicio.

Desde la perspectiva administrativa, la problemática no es menor. La falta de una plataforma digital centralizada limita significativamente la capacidad de los directivos de la Cooperativa de Transporte Urbano Universidad de Bolívar para realizar el seguimiento y control de sus unidades, gestionar de manera eficiente las rutas, paradas y horarios, y monitorear adecuadamente las unidades en operación. Esta carencia tecnológica dificulta la toma de decisiones oportunas, impide identificar fallas en el servicio con rapidez y obstaculiza cualquier proceso de mejora continua que se quiera implementar.

Frente a esta realidad, se identifica la necesidad urgente de desarrollar una aplicación móvil que brinde a los ciudadanos información en tiempo real sobre la ubicación de los buses, rutas disponibles, paradas y los horarios de cada línea. Una solución de este tipo contribuiría a brindar información oportuna a los usuarios y dotaría a los administradores del servicio de una herramienta eficaz para optimizar la gestión del transporte público en la ciudad de Guaranda.

1.3 Justificación

El desarrollo de una aplicación móvil que permita la visualización en tiempo real de rutas y paradas del transporte público en la ciudad de Guaranda responde a una necesidad concreta y urgente de la población: contar con información oportuna, confiable y accesible sobre el servicio de transporte urbano que utilizan a diario. La ausencia de herramientas tecnológicas orientadas a este fin ha generado una incertidumbre significativa entre las expectativas de los usuarios y la realidad del servicio, afectando la movilidad, la puntualidad y la calidad de vida de miles de ciudadanos.

La implementación de esta solución tecnológica permite centralizar y digitalizar la información relacionada con rutas, paradas, horarios y ubicación en tiempo real de las unidades de transporte, poniendo estos datos al alcance de cualquier usuario a través de su dispositivo móvil. De esta manera, se eliminan las barreras de acceso a la información, se reducen los tiempos de espera innecesarios y se brinda al pasajero la posibilidad de planificar sus desplazamientos con mayor precisión y seguridad. Adicionalmente, el sistema de notificaciones automáticas, que alerta al usuario cuando el bus se aproxima a una parada seleccionada, representa un avance significativo en la personalización del servicio y en la comodidad del pasajero.

Desde la perspectiva administrativa, la plataforma de gestión desarrollada en Next.js ofrece a los operadores y entidades responsables de la Cooperativa Universidad de Bolívar una herramienta centralizada para administrar eficientemente las rutas y horarios, permitiendo monitorear en tiempo real el comportamiento de las unidades en circulación. Esto fortalece la capacidad de toma de decisiones, facilita la identificación de fallas en el servicio y la operación del transporte urbano en la ciudad.

En cuanto al alcance del producto, la aplicación está diseñada para atender a tres tipos de usuarios: los pasajeros, quienes podrán visualizar rutas, paradas, horarios y recibir notificaciones en tiempo real a su dispositivo móvil; los choferes, quienes activarán el servicio de ubicación (GPS) de su unidad de transporte directamente desde la aplicación, además, el gerente de la Cooperativa de

Transporte Urbano Universidad de Bolívar, quien gestionará toda la información del servicio desde el panel web administrativo.

El proyecto se vincula con la línea de investigación de la carrera de Software, específicamente con la sublínea "Diseño e implementación de sistemas de información y aplicaciones móviles", lo cual se refleja en la construcción de una solución tecnológica que responde a un problema real de la sociedad y que se sustenta en principios de arquitectura de software, desarrollo móvil, comunicación en tiempo real y usabilidad orientada al usuario.

En definitiva, este proyecto representa una contribución concreta a la modernización del servicio de transporte urbano de la Cooperativa Universidad de Bolívar, demostrando que la tecnología, aplicada con un enfoque centrado en las necesidades de los usuarios, tiene el potencial de mejorar significativamente la movilidad y la experiencia diaria de la comunidad.

1.4 Objetivos: General y Específicos

1.4.1 Objetivo General

Desarrollar un Prototipo funcional de geolocalización para la visualización en tiempo real de rutas y paradas del transporte urbano en la ciudad de Guaranda.

1.4.2 Objetivos Específicos

- Analizar la situación actual del sistema de transporte de la Cooperativa Universidad de Bolívar en la ciudad de Guaranda, identificando las necesidades operativas y de gestión de los procesos de control de rutas, paradas y horarios existentes.
- Diseñar e implementar un prototipo funcional de geolocalización para el monitoreo en tiempo real de las unidades de transporte urbano, mediante la ubicación geográfica del teléfono móvil de los choferes y el panel administrativo web para la gestión de la Cooperativa de Transporte Urbano Universidad de Bolívar.
- Validar el funcionamiento del sistema mediante pruebas funcionales, de rendimiento, de seguridad y de compatibilidad, verificando el cumplimiento de los requerimientos definidos en las historias de usuario.

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes

Hidalgo Astudillo (2018) desarrolló un sistema de monitoreo y ubicación de buses basado en tecnología GPS para la Cooperativa de Transporte Urbano 7 de octubre, en la ciudad de Quevedo. El objetivo fue optimizar el control operativo de las 37 unidades de transporte mediante la captura y visualización de información de geolocalización en tiempo real, reduciendo el incumplimiento de rutas y horarios. La investigación se rigió bajo una metodología aplicada con enfoque experimental, estructurada en fases de análisis de requerimientos, implementación y pruebas de validación. Técnicamente, la solución empleó una arquitectura cliente-servidor bajo el modelo LAMP, con equipos GPS instalados en los vehículos, transmisión de datos vía protocolo TCP y almacenamiento en una base de datos relacional MySQL. Los resultados obtenidos permitieron el seguimiento en tiempo real de las unidades y el control automatizado de rutas, horarios e infracciones operativas. Este antecedente aporta al presente proyecto un referente ecuatoriano de monitoreo de transporte urbano; a diferencia de dicha propuesta, UniBolivar Transit utiliza el teléfono móvil del chofer como fuente de ubicación, eliminando la dependencia de hardware GPS dedicado y reduciendo los costos de implementación (Hidalgo Astudillo, 2018).

Almeida Muñoz y Solís Cuñez (2019) desarrollaron una aplicación móvil Android orientada a la consulta de rutas de buses urbanos en la ciudad de Quito, tomando como referencia los puntos de partida y destino del usuario. El objetivo fue facilitar la identificación de recorridos de transporte cercanos a la ubicación del pasajero, mejorando su experiencia de movilidad mediante una herramienta clara e intuitiva. El proyecto se desarrolló bajo el marco de trabajo Scrum, estructurado en cinco sprints que permitieron una gestión flexible y progresiva de los requerimientos. La arquitectura implementada siguió el modelo cliente-servidor en tres capas, con una aplicación móvil Android, una API REST en Node.js y una base de datos espacial PostgreSQL con PostGIS para el manejo de datos geoespaciales. Como resultado, se obtuvo una herramienta funcional que redujo los tiempos de búsqueda de rutas y mejoró la planificación de los desplazamientos urbanos de los ciudadanos. Su aporte al presente proyecto radica en demostrar la utilidad de las aplicaciones móviles para acercar información de rutas al pasajero, sirviendo como

referente para la organización del componente móvil de UniBolívar Transit (Almeida Muñoz & Solís Cuñez, 2019).

González Tigrero (2019) desarrolló el módulo de servicios web y geográficos de un sistema de mapeo y visualización de rutas de buses urbanos para la Agencia Nacional de Tránsito de la provincia de Santa Elena. El objetivo fue gestionar y disponibilizar información del transporte público de 240 unidades pertenecientes a ocho cooperativas mediante una capa de servicios web, permitiendo a los entes de control monitorear en tiempo real la velocidad, ubicación y estado operativo de las unidades. La investigación aplicó el marco de trabajo Scrum con un enfoque exploratorio, empleando entrevistas y observación directa como técnicas de recopilación de información. Técnicamente, los servicios fueron implementados en Java bajo una arquitectura Modelo-Vista-Controlador con Spring Boot como framework y CouchBase como base de datos no relacional. Los resultados demostraron que la arquitectura MVC con API REST es adecuada para sistemas de monitoreo de transporte a gran escala. Este trabajo aporta al presente proyecto un referente para la separación entre los clientes de usuario y la capa de servicios, principio que se refleja directamente en la API REST y los mecanismos de comunicación implementados en UniBolívar Transit (González Tigrero, 2019).

Díaz Concha (2022) desarrolló el trabajo titulado Geolocalización de rutas de buses en tiempo real: caso de estudio ciudad de Otavalo, orientado a construir y evaluar un prototipo de localización de unidades de transporte público en tiempo real. La propuesta se desarrolló mediante una metodología de prototipado con pruebas de rendimiento y desempeño, implementando el patrón arquitectónico Modelo-Vista-Vista Modelo (MVVM). Técnicamente, el sistema utilizó el GPS del dispositivo móvil del conductor como fuente de datos de ubicación, con una arquitectura cliente-servidor apoyada en Firebase para la transmisión de datos en tiempo real mediante protocolo HTTP (Díaz Concha, 2022).

Bravo Cobeña y Cedeño Intriago (2025) desarrollaron la investigación titulada "Aplicación móvil para la visualización y seguimiento de rutas de la cooperativa de buses urbanos del cantón Bolívar", en la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López. El objetivo fue mejorar el acceso a información de recorridos y reducir los tiempos de espera de los usuarios mediante una aplicación móvil y un panel administrativo web para el monitoreo de unidades en tiempo real. El proyecto se organizó bajo la metodología ágil Scrum en cinco fases

iterativas: inicio, planificación, implementación, revisión y lanzamiento. Técnicamente, la solución empleó una arquitectura cliente-servidor potenciada por microservicios, con Firebase y Google Cloud para la comunicación en tiempo real mediante APIs REST bajo protocolo HTTPS, e implementó el patrón Modelo-Vista-Controlador (MVC) para separar la lógica de negocio de la interfaz de usuario. Los resultados muestran que la aplicación funcionó satisfactoriamente, mejorando la experiencia de los usuarios y brindando a los administradores una herramienta efectiva de monitoreo en tiempo real. Su aporte al presente proyecto radica en confirmar la pertinencia de aplicar soluciones móviles ágiles para mejorar el acceso a información de transporte en operadores locales ecuatorianos, sustentando la elección de Scrum y el enfoque móvil adoptado en UniBolivar Transit (Bravo Cobeña & Cedeño Intriago, 2025).

Síntesis crítica de antecedentes

Los trabajos revisados permiten identificar una evolución progresiva en las soluciones de geolocalización para el transporte público. Los antecedentes más tempranos, como el de Hidalgo Astudillo (2018), dependían de hardware GPS dedicado instalado en los vehículos, lo que implicaba costos elevados de implementación y mantenimiento. Los trabajos posteriores, como los de Almeida Muñoz y Solís Cuñez (2019), González Tigrero (2019) y Díaz Concha (2022), migraron hacia el uso del GPS del dispositivo móvil del conductor como fuente de ubicación, reduciendo significativamente los costos y facilitando la escalabilidad de las soluciones. En cuanto a la transmisión de datos, la mayoría de los trabajos revisados emplearon protocolos HTTP con consultas periódicas al servidor, mientras que UniBolivar Transit adopta WebSocket mediante Socketi para la transmisión en tiempo real, eliminando la latencia asociada al modelo de consulta-respuesta. A diferencia de los antecedentes revisados, el presente proyecto integra en una sola solución tres componentes complementarios: una aplicación móvil para pasajeros con notificaciones push de proximidad, una aplicación móvil para choferes con transmisión GPS en segundo plano, y un panel administrativo web con módulos de despacho, reportes de puntualidad y monitoreo de flota en tiempo real, lo que representa un nivel de integración funcional superior al de los trabajos referenciados.

2.2 Científico

Metodología de Desarrollo de Software

Las necesidades de desarrollar aplicaciones de alta calidad dando cumplimiento a cronogramas y costos se convirtieron en el objetivo principal de las metodologías de desarrollo de software, las cuales son marcos de trabajo para la toma de decisiones en proyectos de ingeniería de software que toman como base el desarrollo iterativo e incremental, proponiendo guías de trabajo que evitan riesgos y logran un mayor control de los proyectos (Hernández Salazar & Beltrán, 2020).

Metodologías Ágiles

En la década de los noventa surgieron metodologías de desarrollo de software ligeras, más adelante nombradas como metodologías ágiles, que buscaban reducir la probabilidad de fracaso por subestimación de costos, tiempos y funcionalidades en los proyectos de desarrollo de software. Estas metodologías nacieron como reacción a las metodologías tradicionales con el propósito de disminuir la burocracia que implica su aplicación en proyectos de pequeña y mediana escala. Las metodologías ágiles tienen dos diferencias fundamentales con las metodologías tradicionales: la primera es que los métodos ágiles son adaptativos y no predictivos; la segunda es que están orientadas a las personas y no orientadas a los procesos (Navarro Cadavid et al., 2013).

Los proyectos ágiles se subdividen en proyectos más pequeños mediante una lista ordenada de características. Cada proyecto es tratado de manera independiente y desarrolla un subconjunto de características durante un período de tiempo corto, de entre dos y seis semanas. La comunicación con el cliente es constante, los proyectos son altamente colaborativos y se adaptan mejor a los cambios; el cambio en los requerimientos es una característica esperada y deseada, al igual que las entregas constantes y la retroalimentación por parte del cliente (Navarro Cadavid et al., 2013).

Extreme Programming (XP)

Extreme Programming (XP) es una metodología ágil de desarrollo de software que pone más énfasis en la adaptabilidad que en la previsibilidad. Al igual que otros métodos ágiles, XP se diferencia de las metodologías tradicionales principalmente en que considera los cambios de requisitos sobre la marcha como un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. XP está basada en la retroalimentación continua entre el cliente y el equipo de

desarrollo, la comunicación fluida entre todos los participantes, la simplicidad en las soluciones implementadas y la disposición permanente para realizar cambios (Navarro Cadavid et al., 2013).

El ciclo de vida de XP se organiza en seis fases: Exploración, en la que los clientes plantean las historias de usuario; Planificación de la Entrega, donde se priorizan las historias y se define el plan de entrega; Iteraciones, en las que se desarrollan las funcionalidades planificadas; Producción, donde se realizan pruebas adicionales antes de la entrega; Mantenimiento, para atender las necesidades del cliente durante el uso del sistema; y Muerte del Proyecto, cuando el cliente no tiene más historias de usuario que agregar (Navarro Cadavid et al., 2013).

Kanban

Kanban es una metodología ágil fundamentada en la visualización del flujo de trabajo y en la limitación de la cantidad de trabajo en curso. En la actualidad ha cobrado relevancia en el desarrollo de software gracias a su enfoque ágil y su capacidad para optimizar la gestión del flujo de trabajo. Se materializa mediante tableros visuales que muestran el progreso del trabajo en etapas claramente definidas, tales como Backlog, En progreso y Listo. Los beneficios identificados incluyen una mayor eficiencia y transparencia en el flujo de trabajo, el fomento del trabajo en equipo y una mejor planificación y organización del proceso de desarrollo. Kanban es una metodología muy útil que permite gestionar productos abiertos a cambios, también en casos de difícil planificación de trabajo, y suele usarse además cuando no se puede establecer un tiempo fijo de entrega (León Ardila, 2020).

Lean Development (LD)

Lean Development es una metodología ágil que se originó a partir del sistema de producción Toyota y que fue adaptada al desarrollo de software por Mary Poppendieck y Tom Poppendieck en 2003. Su filosofía se fundamenta en siete principios: eliminar desperdicios, construir con calidad, crear conocimiento, aplazar el compromiso, entregar rápidamente, respetar a las personas y mejorar el sistema. Entiende como desperdicio todo aquello que no agrega valor al producto final desde el punto de vista del cliente (Sanz Mateo, 2015).

Lean prioriza la eficiencia del equipo, la entrega rápida de valor al cliente y la mejora continua del proceso. Promueve la toma de decisiones lo más tarde posible, con el propósito de contar con la mayor cantidad de información disponible antes de comprometerse con una solución técnica. Al

ser una metodología centrada únicamente en lo que aporta valor real desde la perspectiva del cliente, permite obtener un producto lo más cercano posible a lo requerido de una manera más eficiente (Sanz Mateo, 2015).

Scrum

Scrum es uno de los marcos de trabajo ágiles más utilizados en la actualidad para el desarrollo de software. Sus inicios se remontan a 1986, cuando Hirotaka Takeuchi e Ikujiro Nonaka observaron que las organizaciones más efectivas trabajaban de manera colaborativa y multidisciplinaria. A principios de los años noventa, Jeff Sutherland y Ken Schwaber implementaron este modelo en proyectos de software, convirtiéndose en los padres de la metodología (Hernández Salazar & Beltrán, 2020).

Scrum describe un marco de trabajo de distintas actividades orientadas a crear un ambiente de trabajo colaborativo en proyectos de software, enfocando su criterio en la conformación de equipos altamente eficientes y productivos. Se centra en identificar las funcionalidades de mayor prioridad que pueden desarrollarse en cortos períodos de tiempo denominados sprints, produciendo incrementos funcionales del sistema al finalizar cada ciclo de trabajo (Hernández Salazar & Beltrán, 2020).

La implementación de Scrum se sustenta en tres pilares fundamentales (Hernández Salazar & Beltrán, 2020).

Transparencia: todos los aspectos dentro del proceso son visibles para todos los integrantes del equipo, garantizando claridad sobre el estado del proyecto en cada momento.

Inspección: se revisan frecuentemente todos los pasos del seguimiento para garantizar la correcta planificación y ejecución de las tareas.

Adaptación: se realizan los ajustes necesarios cuando se detectan desviaciones, asegurando que el producto final cumpla con los requisitos establecidos.

Tabla 1. Características principales de las metodologías de desarrollo de software.

Tabla 1.				
<i>Características principales de las metodologías de desarrollo de software.</i>				
Características	Extreme Programming XP	Kanban	LEAN	Scrum
Objetivo principal	Permite trabajar en equipo por metas cortas, revisando siempre lo que hacemos para mejorar sobre la marcha y adaptarnos rápido al cambio en lugar de seguir un plan rígido.	Tablero visual que nos permite separar nuestras actividades por etapas, permitiéndonos una estructuración del proyecto.	Su enfoque de desarrolla en eliminar los desperdicios, para facilitar la entrega rápida al cliente.	Es un marco de trabajo basado en sprints y roles definidos. Permite la estructuración y gestión de proyectos de gran envergadura.
Lapsos de entrega	Lapsos cortos de entregas, con un tiempo aproximado de 1 a 3 semanas de duración.	No existe los lapsos de entregas, las actividades desplegadas en el tablero, fluye en base a las iteraciones definidas.	Las entregas no son fijas, depende de la finalización independiente de las actividades, y se entrega cuando esté listo.	Utiliza sprint de una duración de entre 1 a 4 semanas. Al final de cada sprint se realiza la entrega de un producto funcional.
Roles	No tiene roles fijos como Scrum, pero promueve una un feedback constante entre el cliente y el equipo.	No define roles específicos, pero sí promueve la colaboración y el liderazgo a todos los niveles.	No contiene roles prescritos, pero promueve a la mejora continua y el empoderamiento de todo el equipo de trabajo.	Product Owner: Prioriza el trabajo. Scrum Master: Facilita el proceso y elimina obstáculos.

				Equipo de Desarrollo: Autogestionado y multidisciplinar.
Gestión del trabajo	Se definen las historias de usuarios y se divide el proyecto en interacciones cortas.	Utiliza un tablero donde se dividen las tareas por etapas, las cuales son: por hacer, en proceso y realizado.	Se trata de analizar cada paso de tu proceso para detectar qué no aporta nada y eliminarlo.	Se aplica un "Product Backlog" definiendo un "Sprint Backlog" para las tareas del sprint actual.
Ventajas	Permite un feedback constante con el cliente. Fomenta la retroalimentación. Busca un código impecable usando trucos como programar de a dos o escribir pruebas antes de empezar. La clave aquí es que el equipo no deje de hablarse ni de corregirse para que el resultado final sea de primer nivel.	Visualización del flujo de trabajo. Organización de tareas en etapas. Control progresivo del proyecto. Mejora continua del proceso.	Reduce los costes de desarrollo y producción del sistema. Entregas de valor al cliente. Metodología aplicable para cualquier proyecto, no solo enfocado a software.	Permite organización y gestión. Fomenta la colaboración del equipo de desarrollo. Permite la revisión y mejora constante. Apto para cualquier envergadura de proyecto.
¿Cuándo utilizarla?	Proyectos en donde la calidad del código es prioridad.	Para trabajos que requieran una organización de sus tareas por etapas. Facilitando	Para proyecto que requieran un rápido avance, evitando	Para proyectos de gran envergadura.

	Cuando los requerimientos con cambiantes. Apto para equipos pequeños, y altamente colaborativos. Cuando el cliente solicita interactuar constantemente con el equipo de desarrollo.	el cumplimiento absoluto de todas las tareas.	los desperdicios o eliminándolos. Proyectos que tengan definidos su flujo de trabajo y permitiendo su futura mejora.	Cuando se requiera para la gestión de los trabajos. Proyectos donde la retroalimentación es prioridad. Equipo de desarrollo que requieran forzar su colaboración continua.
--	--	--	--	---

Para el desarrollo del prototipo funcional de geolocalización para la visualización en tiempo real de rutas y paradas del transporte urbano en la ciudad de Guaranda, la elección de Scrum como marco de trabajo ágil responde a la necesidad de gestionar un proyecto con requerimientos que evolucionan conforme se desarrollan las reuniones con los usuarios, además el sistema depende de la precisión de datos geográficos y la interacción constante con el usuario, Scrum permite abordar estos problemas complejos mediante entregas parciales y funcionales al final de cada Sprint. esto asegura que el producto entregue el máximo valor posible de manera productiva y creativa desde las primeras etapas.

2.3 Conceptual

Alertas Tempranas

Son notificaciones automáticas generadas por el sistema cuando se detecta que un bus se aproxima a una parada previamente seleccionada por el usuario. Este mecanismo, basado en la geolocalización en tiempo real, permite informar al pasajero con antelación suficiente para que pueda dirigirse a la parada sin necesidad de permanecer esperando, optimizando su tiempo y mejorando su experiencia con el transporte público.

Base de Datos

Una base de datos es un sistema organizado que permite almacenar y gestionar información de manera estructurada, garantizando la integridad y disponibilidad de los datos para su consulta ágil y segura. Se define como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una organización. Entre sus características principales se encuentran la independencia lógica y física de los datos, la redundancia mínima, el acceso concurrente por múltiples usuarios y la seguridad de acceso (Universidad Estatal de Milagro, 2020).

Docker

Docker es una plataforma de código abierto para desarrolladores y administradores de sistemas que permite desarrollar, implementar y ejecutar aplicaciones con contenedores. Un contenedor es un paquete ligero, autónomo y ejecutable que incluye todo lo necesario para ejecutar una aplicación: código, entorno de ejecución, herramientas del sistema, librerías y configuración. Docker garantiza que las aplicaciones funcionen de manera consistente independientemente del

entorno donde se ejecuten, facilitando el trabajo en equipos de desarrollo y la implementación en producción (Ponsico Martín, 2017).

Fichas de Control

Son los registros administrativos utilizados por la Cooperativa de Transporte Urbano Universidad de Bolívar para documentar el comportamiento operativo de cada unidad. Incluyen información sobre el cumplimiento de horarios, el recorrido realizado, las paradas atendidas y posibles incidencias durante la jornada. Constituyen una herramienta clave para el monitoreo y la evaluación del desempeño del servicio.

Georreferencia

La georreferencia consiste en ubicar una entidad dentro de un espacio determinado, asignándole una posición geográfica única y claramente definida mediante un sistema de coordenadas y datos específicos. Es el proceso fundamental de la cartografía digital que permite asociar información a una ubicación precisa en la superficie terrestre, siendo esencial para la creación de mapas, la planificación de proyectos y la gestión de recursos en campos como la ingeniería, la geografía y las ciencias geoespaciales (Fallas, 2002).

Horarios de Servicio

Son los tiempos establecidos por la cooperativa de transporte para la salida y llegada de cada unidad en las rutas asignadas. Definen los intervalos de frecuencia con los que los buses deben circular a lo largo del día, garantizando la regularidad del servicio para los usuarios. Su correcta gestión es fundamental para reducir los tiempos de espera y mejorar la planificación del transporte urbano.

Laravel

Laravel es un framework de desarrollo web de código abierto que utiliza el lenguaje de programación PHP. Se basa en el patrón arquitectónico MVC (Modelo-Vista-Controlador), que permite separar la lógica de negocio, la capa de datos y la interfaz de usuario. Cuenta con un amplio ecosistema de librerías y módulos que facilitan tareas como autenticación, enrutamiento, acceso a bases de datos y gestión de APIs RESTful, siendo uno de los frameworks PHP más utilizados en el desarrollo web moderno (Jaramillo Zambrano, 2018).

Next.js

Next.js es un framework full-stack basado en React que permite gestionar tanto el frontend como el backend en una misma plataforma. Su arquitectura y la estructura de App Router optimizan la construcción de aplicaciones web modernas, ofreciendo funcionalidades como la renderización del lado del servidor (SSR), la generación de páginas estáticas y el enrutamiento basado en el sistema de archivos, permitiendo desarrollar aplicaciones web completas con una sola base de código JavaScript (Iborra, 2023).

Node.js

Node.js es un entorno de ejecución gratuito y de código abierto que permite utilizar JavaScript en el lado del servidor. Utiliza el motor V8 de Google Chrome, el cual compila y ejecuta el código JavaScript transformándolo en código máquina optimizado. Su arquitectura basada en eventos y su modelo de entrada/salida no bloqueante lo hacen especialmente adecuado para aplicaciones en tiempo real y servicios web de alto rendimiento que requieren atender múltiples solicitudes simultáneas (Iborra, 2023).

PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional de código abierto bajo licencia BSD. Su arquitectura implementa un modelo de cliente-servidor basado en multiprocesos en lugar de hilos, garantizando que cualquier fallo en un proceso individual no comprometa la integridad global del sistema, permitiendo que continúe funcionando sin interrupciones. Es reconocido por su robustez, extensibilidad y cumplimiento de estándares SQL, siendo una de las opciones más utilizadas en entornos de producción empresarial y académico (Universidad Estatal de Milagro, 2020).

React

React es una biblioteca de código abierto de JavaScript, desarrollada por Facebook, diseñada para crear interfaces de usuario intuitivas y dinámicas. Permite crear aplicaciones web de gran envergadura con la capacidad de actualizar datos sin necesidad de refrescar la página completa, gracias a su sistema de componentes reutilizables y su enfoque en la programación declarativa con

flujo de datos unidireccional. Muchos de sus conceptos son la base del framework React Native (Lazcano Calixto et al., 2019).

React Native

React Native es un framework de desarrollo de aplicaciones móviles de código abierto creado por Meta (Facebook), que permite construir aplicaciones para iOS y Android utilizando una base de código compartida en JavaScript. Con React Native no se crea una aplicación web móvil ni híbrida, sino una aplicación móvil nativa que no se distingue de una desarrollada con Java para Android o con Objective-C para iOS. Su principal ventaja es la reutilización de código entre plataformas, reduciendo tiempos de desarrollo y costos de mantenimiento (Lazcano Calixto et al., 2019).

Rutas de Transporte

Corresponden al trayecto geográfico definido que sigue cada unidad de transporte, desde el punto de partida hasta el punto de llegada, pasando por un conjunto de paradas previamente establecidas. Las rutas determinan el área de cobertura del servicio y son un componente esencial para la planificación operativa y la orientación de los usuarios.

Sistema de Posicionamiento Global (GPS)

El Sistema de Posicionamiento Global (GPS, por sus siglas en inglés) es una herramienta de navegación formada por 24 satélites a 20.000 kilómetros por encima de la Tierra con órbitas móviles, enviando datos precisos de la hora y posición del satélite, lo que permite calcular con exactitud la ubicación geográfica de cualquier objeto o persona de manera constante y sin interrupciones temporales, utilizando siempre un marco de referencia global coordinado (Leguizamón, 2006).

Sistema Global de Navegación por Satélite (GNSS)

El Global Navegación Satélite Sistema (GNSS) es el término que agrupa a todos los sistemas de navegación por satélite existentes, incluyendo el GPS estadounidense, el GLONASS ruso, el Galileo europeo y el BeiDou chino. Permite determinar con precisión la ubicación de un objeto o persona en la superficie terrestre mediante señales satelitales que proporcionan coordenadas geográficas exactas dentro de un sistema de referencia global. Sus prestaciones se evalúan según criterios de precisión, integridad, continuidad y disponibilidad (Leguizamón, 2006).

Ubicación en Tiempo Real

Corresponde a la visualización instantánea de la posición actual de las unidades de transporte dentro del mapa digital del sistema. Esta funcionalidad permite a los usuarios conocer el lugar exacto donde se encuentra cada bus durante su recorrido, facilitando una mejor planificación de sus desplazamientos y reduciendo los tiempos de espera. Asimismo, contribuye al control y supervisión de las operaciones por parte de la cooperativa, mejorando la eficiencia y calidad del servicio prestado.

VLBI (Very Long Baseline Interferometry)

La Interferometría de Línea de Base Muy Larga (VLBI) consiste en medir con gran precisión la diferencia de tiempo en que una señal de radio proveniente de un cuásar distante llega a distintas antenas en la Tierra. Mediante múltiples observaciones y una red global de estaciones, permite establecer un sistema de referencia basado en los cuásares y calcular con alta exactitud tanto la posición de las antenas como la del propio cuásar, siendo un pilar fundamental de la geodesia espacial (Fallas, 2002).

2.4 Legal

El presente proyecto tecnológico, orientado al desarrollo de un prototipo funcional de geolocalización para la visualización en tiempo real de rutas y paradas del transporte urbano en la ciudad de Guaranda, capital de la provincia Bolívar, se sustenta en el marco normativo vigente en el Ecuador, que regula el transporte terrestre como servicio público, la organización de las cooperativas de transporte, las competencias del Gobierno Autónomo Descentralizado del cantón Guaranda en materia de tránsito, la protección de datos personales y la propiedad intelectual del software desarrollado. Este marco resulta especialmente relevante en Guaranda, donde el transporte urbano es operado por cooperativas locales, como la Cooperativa de Transporte Urbano Universidad de Bolívar, cuyas rutas, paradas y frecuencias son reguladas por la autoridad municipal de tránsito del cantón.

Constitución de la República del Ecuador

La Constitución de la República del Ecuador (2008) constituye la norma suprema que fundamenta el desarrollo del presente proyecto. En su artículo 394 dispone que “el Estado garantizará la libertad de transporte terrestre, aéreo, marítimo y fluvial dentro del territorio nacional, sin

privilegios de ninguna naturaleza. La promoción del transporte público masivo y la adopción de una política de tarifas diferenciadas de transporte serán prioritarias” (Asamblea Nacional Constituyente, 2008). Esta disposición respalda el desarrollo de herramientas tecnológicas que fortalezcan la calidad del transporte público urbano, como el prototipo de geolocalización propuesto para la ciudad de Guaranda.

Asimismo, el artículo 264, numeral 6, establece como competencia exclusiva de los gobiernos autónomos descentralizados municipales “planificar, regular y controlar el tránsito y el transporte público dentro de su territorio cantonal” (Asamblea Nacional Constituyente, 2008), lo que sitúa al Gobierno Autónomo Descentralizado del cantón Guaranda como la autoridad competente para regular el servicio de transporte urbano en el que opera la Cooperativa de Transporte Urbano Universidad de Bolívar.

De igual manera, el artículo 66, numeral 19, reconoce y garantiza a las personas “el derecho a la protección de datos de carácter personal, que incluye el acceso y la decisión sobre información y datos de este carácter, así como su correspondiente protección. La recolección, archivo, procesamiento, distribución o difusión de estos datos o información requerirán la autorización del titular o el mandato de la ley” (Asamblea Nacional Constituyente, 2008). Este derecho constitucional fundamenta las medidas de protección de datos implementadas en las aplicaciones web y móvil del proyecto.

Ley Orgánica de Transporte Terrestre, Tránsito y Seguridad Vial (LOTTTSV)

La Ley Orgánica de Transporte Terrestre, Tránsito y Seguridad Vial regula la organización, planificación, fomento, regulación, modernización y control del transporte terrestre en el país. Su artículo 46 define la naturaleza del servicio en el cual se enmarca este proyecto:

El transporte terrestre automotor es un servicio público esencial y una actividad económica estratégica del Estado, que consiste en la movilización libre y segura de personas o de bienes de un lugar a otro, haciendo uso del sistema vial nacional, terminales terrestres y centros de transferencia de pasajeros y carga en el territorio ecuatoriano (Asamblea Nacional Constituyente, 2008).

El artículo 47 de la misma ley señala que el transporte terrestre de personas o bienes responderá a las condiciones de responsabilidad, universalidad, accesibilidad, comodidad, continuidad,

seguridad, calidad y tarifas equitativas (Asamblea Nacional Constituyente, 2008). El prototipo de geolocalización desarrollado contribuye directamente al cumplimiento de las condiciones de continuidad, calidad y accesibilidad del servicio, al permitir que los usuarios conozcan en tiempo real la ubicación de las unidades, sus rutas y paradas.

En cuanto al ámbito de operación, los artículos 65 y 66 establecen que el servicio de transporte público comprende, entre otros, el ámbito intracantonal, entendido como aquel que opera dentro de los límites cantonales, siendo la celebración de los contratos y/o permisos de operación de estos servicios atribución de los Gobiernos Autónomos Descentralizados Municipales o de la Agencia Nacional de Tránsito en los cantones que no hayan asumido la competencia (Asamblea Nacional del Ecuador, 2008). La Cooperativa de Transporte Urbano Universidad de Bolívar presta el servicio de transporte público intracantonal en la ciudad de Guaranda bajo esta modalidad.

Respecto al control del cumplimiento de horarios, rutas y frecuencias, aspecto central del módulo de multas y fichas de control del sistema desarrollado, la ley contempla un régimen sancionatorio aplicable a las operadoras de transporte. Entre las infracciones de transporte se incluye el incumplimiento de las frecuencias autorizadas y la interrupción o suspensión injustificada de la prestación de los servicios; el artículo 83 dispone que las sanciones por infracciones en contra de las operadoras serán impuestas por la autoridad competente, pudiendo llegar incluso a la revocatoria del contrato, permiso o autorización de operación, de acuerdo con la gravedad de la falta y el interés público comprometido, garantizando el debido proceso (Asamblea Nacional Constituyente, 2008). Este régimen legal justifica que las operadoras, como la cooperativa objeto de estudio, mantengan mecanismos internos de control de horarios y rutas, los cuales son fortalecidos mediante el monitoreo en tiempo real que ofrece el prototipo desarrollado.

Código Orgánico de Organización Territorial, Autonomía y Descentralización (COOTAD)

El COOTAD, en su artículo 55, literal f), establece como competencia exclusiva del gobierno autónomo descentralizado municipal “planificar, regular y controlar el tránsito y el transporte terrestre dentro de su circunscripción cantonal (Asamblea Nacional del Ecuador, 2010). En concordancia, el artículo 130, referente al ejercicio de la competencia de tránsito y transporte, señala que a los gobiernos autónomos descentralizados municipales les corresponde de forma exclusiva planificar, regular y controlar el tránsito, el transporte y la seguridad vial dentro de su

territorio cantonal, definiendo el modelo de gestión correspondiente de conformidad con la ley (Asamblea Nacional del Ecuador, 2010).

Regulación municipal del transporte urbano en el cantón Guaranda

En el caso específico del cantón Guaranda, la competencia de tránsito y transporte terrestre es ejercida a través de la Unidad Municipal de Transporte Terrestre, Tránsito y Seguridad Vial del Gobierno Autónomo Descentralizado del cantón Guaranda, dependencia encargada de planificar, organizar, regular y controlar el transporte terrestre, el tránsito y la seguridad vial Inter parroquial, Inter cantonal y urbana en todo el territorio de la jurisdicción cantonal, en coordinación con los órganos nacionales competentes y en aplicación de su ordenanza de creación (Gobierno Autónomo Descentralizado del Cantón Guaranda, s. f.). Entre sus atribuciones de planificación constan la planificación de la circulación de los vehículos y servicios de transporte público de pasajeros, la generación de sistemas integrales para la administración del tránsito urbano y la determinación de la infraestructura adecuada para la prestación del servicio, atribuciones directamente vinculadas con el objeto del presente proyecto, pues el prototipo de geolocalización constituye una herramienta tecnológica que apoya la visualización y el seguimiento de las rutas y paradas del transporte urbano guarandeño.

La actuación de esta autoridad municipal sobre el servicio que presta la Cooperativa de Transporte Urbano Universidad de Bolívar se evidencia, por ejemplo, en los acuerdos alcanzados entre la Federación de Estudiantes Universitarios del Ecuador (FEUE) filial Bolívar y los transportistas de dicha cooperativa, proceso en el que el Gobierno Autónomo Descentralizado del cantón Guaranda intervino en la regulación del servicio urbano que conecta la ciudad con la Universidad Estatal de Bolívar (Gobierno Autónomo Descentralizado del Cantón Guaranda, s. f.). El prototipo desarrollado se alinea con esta regulación municipal al visualizar en tiempo real las rutas y paradas oficialmente establecidas para el servicio en la ciudad de Guaranda, aportando transparencia tanto a los usuarios como a los organismos de control local.

Ley Orgánica de Economía Popular y Solidaria (LOEPS)

La Cooperativa de Transporte Urbano Universidad de Bolívar, como organización del sector cooperativo, se rige por la Ley Orgánica de Economía Popular y Solidaria. El artículo 21 define al sector cooperativo como “el conjunto de cooperativas entendidas como sociedades de personas

que se han unido en forma voluntaria para satisfacer sus necesidades económicas, sociales y culturales en común, mediante una empresa de propiedad conjunta y de gestión democrática, con personalidad jurídica de derecho privado e interés social” (Asamblea Nacional del Ecuador, 2011).

El artículo 28 de la misma ley clasifica a las cooperativas de transporte dentro del grupo de cooperativas de servicios, definidas como aquellas que “se organizan con el fin de satisfacer diversas necesidades comunes de los socios o de la colectividad” (Asamblea Nacional del Ecuador, 2011). Complementariamente, el Reglamento General a la LOEPS establece que las cooperativas de transporte son las constituidas para prestar el servicio de transporte de personas o bienes y podrán ser de propietarios o de trabajadores, pudiendo las cooperativas de propietarios ser de caja común o de caja individual; asimismo, dispone que las autorizaciones para la prestación del servicio de transporte se concederán por la autoridad competente a favor de las cooperativas y no individualmente a sus socios (Presidencia de la República del Ecuador, 2012). Este marco explica la estructura organizativa de la cooperativa y la necesidad de mecanismos internos de control operativo sobre sus socios y conductores, como los que digitaliza el sistema desarrollado.

Ley Orgánica de Protección de Datos Personales (LOPD)

Dado que las aplicaciones web y móvil del proyecto recopilan y tratan datos personales de los conductores (nombres, correo electrónico, número de celular) y datos de ubicación de las unidades, resulta aplicable la Ley Orgánica de Protección de Datos Personales. El artículo 10 de dicha ley establece los principios que rigen el tratamiento de datos personales, entre ellos los de juridicidad, lealtad, transparencia, finalidad, pertinencia y minimización, confidencialidad y seguridad de datos personales, los cuales orientan el diseño de los formularios de registro, la gestión de credenciales de acceso y el almacenamiento de la información en el sistema (Asamblea Nacional del Ecuador, 2021).

Adicionalmente, el artículo 8 dispone que el consentimiento del titular constituye una de las bases legitimadoras del tratamiento, el cual deberá ser libre, específico, informado e inequívoco (Asamblea Nacional del Ecuador, 2021). Por ello, el sistema desarrollado incorpora la aceptación expresa de términos y condiciones durante el registro de los usuarios, así como mecanismos de autenticación que restringen el acceso a la información únicamente a personal autorizado de la cooperativa. Las obligaciones relativas a la evaluación de impacto y a la notificación de

vulneraciones de seguridad, previstas en los artículos 42 y 43 de la misma ley, se detallaron previamente en esta sección.

El Reglamento General a la Ley Orgánica de Protección de Datos Personales, emitido mediante Decreto Ejecutivo 904 y publicado en el Registro Oficial Suplemento 435, establece las disposiciones complementarias para la aplicación de la ley. En particular, regula las medidas técnicas y organizativas que los responsables del tratamiento de datos deben implementar para garantizar la seguridad de la información personal. En el contexto del sistema UniBolívar Transit, estas disposiciones orientan la implementación de mecanismos de autenticación, cifrado de comunicaciones mediante HTTPS y restricción de acceso a los datos de geolocalización de los choferes (Presidencia de la República del Ecuador, 2023).

Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos

Considerando que el sistema opera mediante plataformas web y móviles que intercambian mensajes de datos, es pertinente la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos. Su artículo 9 dispone que, para la elaboración, transferencia o utilización de bases de datos obtenidas directa o indirectamente del uso o transmisión de mensajes de datos, se requerirá el consentimiento expreso del titular de éstos, quien podrá seleccionar la información a compartirse con terceros (Congreso Nacional del Ecuador, 2002). Esta disposición refuerza la obligación de recabar el consentimiento de los conductores registrados en el sistema y de limitar el uso de sus datos a los fines operativos del servicio de transporte.

Reglamento Interno de la Cooperativa de Transporte Urbano Universidad de Bolívar

La Cooperativa de Transporte Urbano Universidad de Bolívar cuenta con un reglamento interno que regula el funcionamiento operativo de sus unidades y establece las normas de conducta que deben cumplir los conductores, en concordancia con las atribuciones de autorregulación que la LOEPS y su reglamento reconocen a las organizaciones cooperativas. Este reglamento define los procedimientos para el control de rutas y horarios de servicio, establece los mecanismos de sanción ante incumplimientos operativos y regula el uso de las fichas de control como herramienta de seguimiento diario.

En materia de multas, el reglamento interno estipula las infracciones sancionables, entre las que se encuentran el incumplimiento de horarios de salida, el abandono de la ruta asignada y la falta de

atención a los usuarios. Las multas son registradas administrativamente y descontadas del pago correspondiente al conductor, constituyendo un mecanismo de control disciplinario que busca garantizar la calidad y puntualidad del servicio de transporte urbano en la ciudad de Guaranda, en armonía con las condiciones de continuidad y calidad exigidas por el artículo 47 de la LOTTTSV y con el régimen de responsabilidad de las operadoras previsto en dicha ley. El sistema desarrollado en este proyecto se alinea con estos procesos internos al permitir el registro digital de los recorridos, el monitoreo de horarios en tiempo real y la generación de reportes que facilitan la aplicación del reglamento interno por parte de los directivos de la cooperativa.

2.5 Georreferencial

Cooperativa Universidad de Bolívar

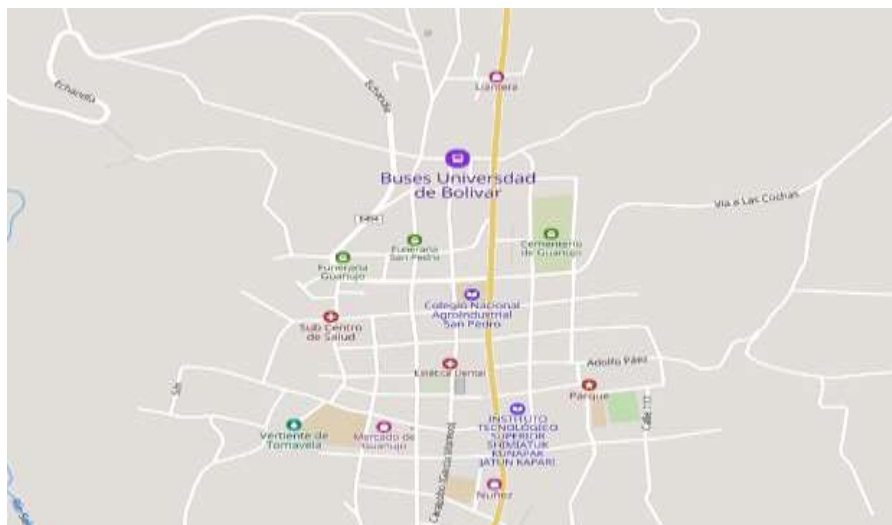
Vía Ambato Guaranda

Guanujo, Bolívar

Latitud: 1°33'18.1"S → -1.555028

Longitud: 79°00'28.4"W → -79.007889

Ilustración 1 Georreferencial



Nota: Google Maps (2026)

CAPÍTULO III

METODOLOGÍA

3.1 Metodología de Desarrollo de Software

En el desarrollo del sistema de geolocalización y visualización en tiempo real de rutas, horarios y paradas de la Cooperativa de Transporte Urbano Universidad de Bolívar, resultó fundamental seleccionar una metodología de trabajo que se adapte a las características reales del equipo y a las condiciones operativas de la institución. Considerando que el proyecto requería entregas progresivas, validación constante con los usuarios y capacidad de respuesta ante cambios en los requerimientos, se optó por implementar la metodología ágil Scrum como marco de trabajo principal para la gestión y desarrollo del sistema.

Scrum es un marco de trabajo ágil para el desarrollo de software, caracterizado por ser adaptable, iterativo, rápido y flexible, diseñado para ofrecer valor al cliente de forma progresiva a lo largo del proyecto. Garantiza transparencia en la comunicación, fomenta la responsabilidad compartida dentro del equipo y permite realizar ajustes continuos en función de los resultados obtenidos en cada ciclo de desarrollo (Chávez, 2018).

La elección de Scrum se fundamentó en su capacidad para organizar el trabajo de desarrollo en ciclos cortos denominados sprints, donde al finalizar cada uno se obtiene un incremento funcional del sistema listo para ser validado por los usuarios. Esta característica resultó especialmente valiosa en el presente proyecto, dado que el sistema involucra múltiples módulos panel administrativo web, aplicación móvil del chofer y aplicación móvil del pasajero que debían desarrollarse, integrarse y verificarse de forma progresiva para garantizar su correcto funcionamiento en conjunto. Scrum permitió mantener al gerente de la cooperativa informado del avance en cada etapa, facilitando la incorporación de observaciones y ajustes antes de continuar con el siguiente módulo del sistema.

Funcionamiento de la Metodología Scrum

Scrum organiza el desarrollo de software en torno a tres elementos fundamentales que estructuran el trabajo del equipo a lo largo del proyecto (Hernández Salazar & Beltrán, 2020):

Artefactos: Proporcionan transparencia e información clave sobre el estado del proyecto. El Product Backlog contiene la lista completa de funcionalidades requeridas por el sistema, el Sprint

Backlog agrupa las tareas seleccionadas para desarrollarse durante el sprint en curso, y el Incremento representa el resultado funcional y verificable entregado al finalizar cada sprint (Hernández Salazar & Beltrán, 2020).

Eventos: Corresponden a las instancias formales de planificación, seguimiento y evaluación del trabajo. La Planificación del Sprint define las tareas a ejecutar; las Reuniones Diarias permiten revisar el avance y resolver impedimentos; la Revisión del Sprint presenta el resultado al usuario para su validación; y la Retrospectiva evalúa el proceso para identificar mejoras en el siguiente ciclo (Hernández Salazar & Beltrán, 2020).

Roles: Scrum establece tres roles principales: el Product Owner, responsable de priorizar las funcionalidades del sistema según las necesidades del usuario; el Scrum Master, encargado de facilitar el proceso y asegurar el cumplimiento del marco de trabajo; y el Development Team, quienes ejecutan las tareas de análisis, diseño, codificación e implementación (Hernández Salazar & Beltrán, 2020).

El ciclo de trabajo de cada sprint inicia con la selección de tareas del Product Backlog, continúa con el desarrollo de las funcionalidades planificadas y concluye con la entrega de un incremento funcional al usuario para su revisión. Este proceso iterativo se repite hasta completar todas las funcionalidades requeridas por el sistema (Hernández Salazar & Beltrán, 2020).

Planificación y Fases de Implementación

La ejecución del proyecto se estructuró en cuatro sprints iterativos donde intervinieron directamente los desarrolladores y los usuarios de la cooperativa. El ciclo de vida del desarrollo se organizó de la siguiente manera:

Sprint 1: Análisis y Levantamiento de Requerimientos:

Se realizó el levantamiento de información mediante entrevistas con el gerente y personal operativo de la Cooperativa de Transporte Urbano Universidad de Bolívar, con el propósito de comprender el funcionamiento actual del servicio de transporte e identificar los problemas existentes. En esta etapa se definieron los actores del sistema, se establecieron los requerimientos funcionales y no funcionales, se elaboraron las historias de usuario y se construyeron los diagramas de casos de uso iniciales que sirvieron como base para las siguientes etapas del desarrollo.

Sprint 2: Diseño del Sistema:

Se diseñó la arquitectura del sistema bajo el modelo Cliente-Servidor, se realizó el modelado de la base de datos y se elaboraron los diagramas UML bajo el enfoque de vistas 4+1. Adicionalmente, se diseñaron las interfaces gráficas de las aplicaciones móvil y web, las cuales fueron presentadas al gerente de la cooperativa para su revisión y validación antes de iniciar el proceso de desarrollo.

Sprint 3: Desarrollo del Sistema:

Se implementaron todos los módulos del sistema de forma integrada. El equipo desarrolló el backend con sus servicios API REST, las interfaces gráficas del panel administrativo web, las aplicaciones móviles para choferes y pasajeros, la integración de geolocalización GPS y la visualización en tiempo real de las unidades de transporte. Al finalizar cada funcionalidad se realizó una entrega parcial al usuario para su revisión y aprobación inmediata, incorporando las observaciones recibidas antes de continuar con el siguiente módulo.

Sprint 4: Validación, Pruebas e Implementación

Se verificó el funcionamiento del sistema mediante la ejecución de pruebas funcionales, pruebas de rendimiento, pruebas de latencia del WebSocket, pruebas de GPS en segundo plano, pruebas de compatibilidad en diferentes versiones de Android, pruebas de tiempo de respuesta de los endpoints y pruebas de seguridad. A partir de los resultados obtenidos, se corrigieron los errores identificados y se realizaron los ajustes necesarios para garantizar el correcto funcionamiento del sistema.

Posteriormente, se realizó la capacitación al administrador de la cooperativa sobre el uso del panel administrativo, incluyendo la gestión de usuarios, líneas, rutas, paradas, horarios, monitoreo de flota y generación de reportes. Finalmente, se efectuó el despliegue del sistema en el servidor de producción y se entregó el prototipo funcional a la Cooperativa de Transporte Urbano Universidad de Bolívar.

Roles Scrum Aplicados al Proyecto

Para la ejecución del proyecto se definieron los siguientes roles dentro del marco de trabajo Scrum:

Scrum Master: Erick Alarcón, responsable de facilitar la aplicación de la metodología Scrum, coordinar las reuniones de planificación y retrospectiva, y eliminar los impedimentos que afecten el avance del equipo de desarrollo.

Product Owner: Gerente de la Cooperativa de Transporte Urbano Universidad de Bolívar, responsable de representar los intereses de la organización, validar las funcionalidades implementadas al finalizar cada sprint y aprobar el Product Backlog.

Development Team: Erick Alarcón y Juan Pasto, encargados del análisis, diseño, codificación, pruebas e implementación del sistema.

3.2 Técnicas e Instrumentos de Recopilación de Datos

En el desarrollo del sistema de geolocalización de la Cooperativa de Transporte Urbano Universidad de Bolívar, se requirió obtener información directamente de los actores involucrados en la operación del servicio de transporte. Con este propósito, se seleccionó la entrevista como técnica de recopilación de datos, la cual permitió identificar los problemas existentes, los flujos de trabajo actuales y los requerimientos del sistema a implementar.

Entrevista: Se aplicó al gerente y a los choferes de la Cooperativa de Transporte Urbano Universidad de Bolívar, con el objetivo de identificar los problemas relacionados con la gestión de rutas, horarios y control operativo de las unidades de transporte. El instrumento empleado fue una guía de entrevista estructurada, que permitió obtener información detallada sobre las dificultades existentes y las funcionalidades requeridas en el sistema.

La aplicación de esta técnica permitió contar con información precisa y confiable sobre el funcionamiento de la cooperativa, lo que sirvió de base para diseñar un sistema que responda de manera efectiva a las necesidades reales de sus usuarios.

CAPÍTULO IV

INGENIERÍA DEL PROYECTO

4.1 Planificación

En esta sección se presenta la planificación de los sprints utilizados para el desarrollo del sistema de geolocalización para la ubicación en tiempo real de las rutas y paradas del transporte urbano de la Cooperativa Universidad de Bolívar.

4.1.1 Alcance del Producto

El sistema UniBolívar Transit está conformado por tres componentes principales que interactúan entre sí para brindar un servicio de información de transporte urbano en tiempo real: la Aplicación móvil del Pasajero, la Aplicación móvil del Chofer y el Panel de Administración Web. A continuación, se describen las funciones principales de cada componente.

Funciones Principales del Sistema

Aplicación Móvil del Pasajero

- Permitir el uso de la aplicación desde el primer inicio sin necesidad de crear una cuenta ni ingresar datos personales, garantizando el anonimato y la facilidad de uso.
- Visualizar en un mapa interactivo todas las rutas de transporte urbano disponibles, con sus paradas georreferenciadas, permitiendo al pasajero explorar el recorrido completo de cada línea antes de abordar.
- Mostrar en tiempo real la posición geográfica de los buses activos en la ruta seleccionada, con actualización cada tres segundos y animación suave del marcador sobre el mapa, proporcionando información precisa sobre el avance de cada unidad.
- Calcular y presentar el tiempo estimado de llegada de los tres próximos buses a cada parada del recorrido, utilizando el cálculo por secuencia de paradas para una mayor precisión, e identificando el número de unidad de cada bus.
- Identificar y resaltar automáticamente la parada más cercana a la ubicación actual del pasajero, indicando la distancia en metros o kilómetros y señalando visualmente el punto óptimo de abordaje.

- Habilitar la suscripción a notificaciones push de parada con un único toque, permitiendo al pasajero recibir alertas cuando el bus esté próximo a llegar, y cancelar la suscripción de igual manera una vez que el bus haya pasado.
- Las notificaciones se entregan mediante notificaciones push nativas del dispositivo Android, sin costo adicional para el usuario, sin necesidad de SMS ni aplicaciones de mensajería como WhatsApp.
- Consultar los horarios de servicio de cada línea de transporte por día de la semana, visualizando las franjas horarias con su frecuencia de despacho y resaltando la franja vigente al momento de la consulta.
- Gestionar las alertas activas desde una pestana dedicada, donde el pasajero puede revisar todas sus suscripciones y desactivarlas individualmente.
- Activar automáticamente el modo nocturno del mapa a partir de las 19h00, reduciendo el brillo y ajustando la paleta de colores para una experiencia visual confortable en condiciones de poca luz.
- Notificar al usuario mediante un aviso visible cuando el dispositivo no tenga conexión a internet, e intentar restablecer la información automáticamente al recuperar la conectividad.

Aplicación Móvil del Chofer

- Permitir al chofer iniciar sesión de manera segura utilizando sus credenciales institucionales asignadas por el administrador, garantizando que únicamente el personal autorizado pueda registrar recorridos y transmitir ubicaciones.
- Habilitar el inicio de servicio mediante la selección de la ruta a recorrer, activando automáticamente el rastreo GPS y el registro del viaje en el sistema backend al momento de confirmar el inicio.
- Transmitir la ubicación geográfica del bus al servidor cada tres segundos de forma automática y continua, incluso cuando la pantalla del dispositivo este apagada o la

aplicación se encuentre en segundo plano, mediante el uso de un servicio de ubicación en segundo plano.

- Mostrar en pantalla durante el servicio activo el tiempo transcurrido del recorrido, la velocidad instantánea, la velocidad máxima alcanzada y la distancia recorrida, proporcionando al chofer un panel de información operativa en tiempo real.
- Habilitar la finalización del servicio con confirmación explícita del chofer, registrando automáticamente la hora de llegada, la distancia total y la velocidad máxima del recorrido en el sistema.
- Mantener la transmisión de ubicación GPS durante el servicio activo, incluso cuando la aplicación se encuentre en segundo plano o la pantalla del dispositivo esté apagada, evitando interrupciones en el seguimiento de la unidad.
- Cerrar automáticamente el servicio a las 21h00 con notificación previa al chofer, garantizando el registro correcto del horario de finalización del recorrido nocturno.
- Visualizar el historial de viajes realizados durante el año en curso mediante un calendario interactivo, donde cada día trabajado queda marcado visualmente. Al seleccionar un día específico, se muestra el detalle de cada recorrido con hora de salida, hora de llegada, duración, distancia recorrida y velocidad máxima. Los días con atrasos se resaltan en rojo y los días trabajados sin inconvenientes en verde, permitiendo al chofer llevar un control completo de su desempeño a lo largo del año.

Panel de Administración Web

- Implementar el control de acceso mediante autenticación con credenciales de administrador, garantizando que únicamente el personal autorizado acceda a las funcionalidades de gestión del sistema.
- Gestionar de manera centralizada el registro, edición, activación y desactivación de las cuentas de choferes, incluyendo la asignación del número de unidad correspondiente a cada operador.

- Administrar la información de las líneas de transporte, rutas de ida y vuelta, y paradas georreferenciadas, permitiendo la creación, edición y reordenamiento de paradas mediante una interfaz de arrastre intuitiva.
- Gestionar los horarios de servicio de cada línea por día de la semana, permitiendo configurar las franjas horarias y frecuencias de despacho que se visualizan en la aplicación móvil del pasajero.
- Monitorear en tiempo real la posición de todos los buses activos de la flota en un mapa centralizado, visualizando el número de unidad, la velocidad y la ruta de cada bus con actualización automática cada tres segundos.
- Visualizar el estado operativo de la flota de 48 unidades con información detallada de cada bus, incluyendo el chofer asignado, los viajes completados, la hora del primer y último recorrido del día.
- Generar reportes detallados de los recorridos realizados por unidad y fecha, con indicación del estado de puntualidad basado en la regla de atraso cuando el recorrido supera los treinta minutos, y exportación del reporte en formato PDF.
- Presentar métricas estadísticas mediante indicadores clave de desempeño sobre la operación de la flota, incluyendo el promedio de duración de recorridos, los buses activos en tiempo real y el total de viajes del día.
- Visualizar métricas estadísticas mediante indicadores clave de desempeño sobre la operación de la flota, incluyendo el total de buses activos, viajes del día, pasajeros suscritos y notificaciones enviadas.

4.1.2 Restricciones del sistema

El alcance del sistema UniBolivar Transit está delimitado por los siguientes aspectos técnicos, funcionales y operativos que definen el perímetro de la solución desarrollada:

- La solución tecnológica propuesta ha sido desarrollada para ejecutarse en dispositivos móviles con sistema operativo Android, debido a que esta plataforma representa la mayor parte de los equipos utilizados por la cooperativa y sus usuarios.

Por esta razón, la compatibilidad con dispositivos iOS no se aborda dentro de esta investigación.

- La distribución de la aplicación se realiza mediante descarga directa desde la página web oficial del sistema <https://unibolivartransit.com>, donde los usuarios pueden acceder desde su dispositivo móvil y descargar el archivo de instalación (APK) correspondiente a su perfil: App Pasajero o App Chofer. Este mecanismo de distribución elimina la dependencia de tiendas de aplicaciones externas, permitiendo una instalación inmediata y gratuita en cualquier dispositivo Android 8.0 o superior.
- La determinación del tiempo estimado de llegada de las unidades se fundamenta en la información generada por el propio sistema, considerando la secuencia de paradas establecidas en cada recorrido. En consecuencia, no se emplean servicios externos especializados en navegación o cálculo de rutas vehiculares.
- El alcance funcional de la propuesta se centra en la gestión y monitoreo de las unidades de transporte, así como en la notificación de eventos relevantes a los usuarios. Por ello, mecanismos de interacción directa entre pasajeros y personal administrativo no forman parte de los objetivos definidos para esta investigación.
- La obtención de la ubicación geográfica de las unidades se realiza mediante los dispositivos móviles utilizados por los conductores durante la operación del servicio, constituyendo la fuente principal de información para el seguimiento en tiempo real.
- La estimación de los tiempos de arribo se efectúa mediante parámetros operativos asociados a la posición geográfica, velocidad de desplazamiento y recorrido definido. El estudio no considera la incorporación de modelos predictivos avanzados basados en inteligencia artificial o aprendizaje automático.
- El funcionamiento adecuado del servicio de notificaciones depende de la disponibilidad de conexión a Internet en los dispositivos de los usuarios al momento de la transmisión de la información.

- Las funcionalidades administrativas han sido concebidas para su utilización en entornos de escritorio mediante navegadores web, priorizando las actividades de supervisión y gestión operativa de la cooperativa.
- La implementación del sistema se encuentra orientada al entorno operativo de la ciudad de Guaranda, provincia de Bolívar.

4.2 SPRINT 1: ANÁLISIS Y LEVANTAMIENTO DE REQUERIMIENTOS

4.2.1 Objetivo del Sprint 1

Levantar los requerimientos del sistema mediante entrevistas con choferes y administradores de la cooperativa de transporte Universidad de Bolívar, definir las historias de usuario y configurar el entorno de desarrollo.

4.2.2 Actividades Principales

- Entrevista con choferes y administrador de la Cooperativa de Transporte Universidad de Bolívar.
- Definición de historias de usuario HU-01 a HU-22.
- Definición del stack tecnológico
- Configuración del entorno de desarrollo.

4.2.3 Sprint Backlog

Tabla 2. Sprint Backlog del Sprint 1

Id	Tarea	Estimación	Responsables	Estado
T-01	Entrevista con choferes y administrador de la cooperativa	8h	Erick Alarcón, Juan Pasto	Completado

T-02	Definición de historias de usuario	6h	Erick Alarcón, Juan Pasto	Completado
T-03	Definición del stack tecnológico	4h	Erick Alarcón, Juan Pasto	Completado
T-04	Configuración del entorno de desarrollo local	8h	Erick Alarcón, Juan Pasto	Completado

4.2.4 Resultados del Sprint 1

Para el cumplimiento del objetivo del Sprint 1, se aplicó la técnica de la entrevista dirigida a los choferes y al administrador de la Cooperativa de Transporte Urbano Universidad de Bolívar, cuyos instrumentos se encuentran detallados en el Anexo 3. A partir de la información recopilada, se identificaron las necesidades operativas del sistema y se definieron las 22 historias de usuario que representan los requerimientos funcionales del proyecto, distribuidas entre los tres actores: pasajero, chofer y administrador.

A continuación, se detallan las historias de usuario que servirán para recopilar la información necesaria durante el proceso de desarrollo del sistema.

Tabla 3. Historia de usuario 1

Historia de usuario	
Número: HU-01	Usuario: pasajero
Nombre de historia: Registro automático de pasajero	Riesgo en desarrollo: Bajo
Requisitos previos: Tener la aplicación instalada y contar con conexión a internet.	
Descripción:	La aplicación debe registrar automáticamente al pasajero sin necesidad de crear una cuenta, para poder usar el sistema de notificaciones sin fricciones.

Criterio de validación:	El sistema genera un device_id único por dispositivo. Se crea un usuario anónimo en la base de datos. El pasajero puede suscribirse a paradas sin iniciar sesión. El token FCM se registra correctamente en el servidor.
-------------------------	--

Tabla 4 Historia de usuario 2

Historia de usuario	
Número: HU-02	Usuario: pasajero
Nombre de historia: Descarga de la aplicación desde la página web	Riesgo en desarrollo: Bajo
Requisitos previos: Tener un dispositivo Android 8.0 o superior y conexión a internet.	
Descripción:	El pasajero debe descargar la aplicación móvil directamente desde la página web oficial del sistema, para instalarla en su dispositivo sin necesidad de acceder a una tienda de aplicaciones.
Criterio de validación:	La página web muestra dos opciones de descarga: App Pasajero y App Chofer. Al tocar el botón de descarga desde el celular, el archivo APK se descarga automáticamente. Se muestran las instrucciones de instalación paso a paso. Se indica que la app es compatible con Android 8.0 o superior. La descarga es gratuita y no requiere registro previo.

Tabla 5. Historia de usuario 3

Historia de usuario	
Número: HU-03	Usuario: pasajero
Nombre de historia: Visualización de rutas en mapa	Riesgo en desarrollo: Medio
Requisitos previos: Tener la app instalada, conexión a internet y permisos de ubicación	
Descripción:	El pasajero debe visualizar las rutas de buses disponibles en un mapa interactivo, para conocer las paradas y el recorrido de cada línea.

Criterio de validación:	El mapa muestra todas las rutas disponibles. Al seleccionar una ruta se muestran todas sus paradas. El mapa hace zoom automático para mostrar la ruta completa. El mapa tiene modo nocturno automático después de las 19h00.
-------------------------	--

Tabla 6. Historia de usuario 4

Historia de usuario	
Número: HU-04	Usuario: pasajero
Nombre de historia: Visualización de buses en tiempo real	Riesgo en desarrollo: Alto
Requisitos previos: Al menos un chofer debe tener servicio activo y el dispositivo debe tener conexión a internet.	
Descripción:	El pasajero debe visualizar la posición de los buses en tiempo real en el mapa, para saber dónde se encuentra el bus y cuánto tiempo falta para que llegue a su parada.
Criterio de validación:	La posición del bus se actualiza cada 3 segundos. El icono del bus se mueve con animación suave. Si hay varios buses se muestran todos. Si no hay buses activos se muestra el banner Sin servicio.

Tabla 7. Historia de usuario 5

Historia de usuario	
Número: HU-05	Usuario: pasajero
Nombre de historia: Calculo de tiempo estimado de llegada por parada	Riesgo en desarrollo: Alto
Requisitos previos: Al menos un bus debe estar activo en la ruta seleccionada con paradas y coordenadas GPS registradas.	
Descripción:	El pasajero debe visualizar el tiempo estimado de llegada del bus a cada parada, para planificar su tiempo y saber con anticipación en qué parada esperar el servicio.

Criterio de validación:	Cada parada muestra el Tiempo Estimado de Llegada de los 3 próximos buses. se calcula por secuencia de paradas, no en línea recta. Se muestra el número de unidad de cada bus. Las paradas que el bus ya pasó se muestran opacas con badge "Ya pasó".
-------------------------	---

Tabla 8. Historia de usuario 6

Historia de usuario	
Número: HU-06	Usuario: pasajero
Nombre de historia: Suscripción a notificaciones de parada	Riesgo en desarrollo: Medio
Requisitos previos: El pasajero debe tener la app instalada y haber aceptado los permisos de notificaciones y al menos una ruta con paradas registradas.	
Descripción:	El pasajero debe activar una alerta en una parada específica, para recibir una notificación cuando el bus esté llegando a esa parada.
Criterio de validación:	La aplicación debe permitir activar o desactivar la alerta con un solo toque. Las alertas activas se listan en la pestaña Alertas. La suscripción se cancela automáticamente cuando el bus pasa.

Tabla 9. Historia de usuario 7

Historia de usuario	
Número: HU-07	Usuario: pasajero
Nombre de historia: Notificación push cuando el bus está llegando	Riesgo en desarrollo: Alto
Requisitos previos: El pasajero debe tener suscripción activa y el chofer debe tener servicio activo en esa ruta.	

Descripción:	El pasajero debe recibir notificaciones push en distintos niveles de proximidad del bus hacia su parada, para salir a tiempo y no perder el servicio.
Criterio de validación:	El sistema envía una notificación cuando el bus se dirige hacia la parada del pasajero, indicando que el bus está en camino. Se envía una segunda notificación cuando el bus se encuentra a 300 metros de la parada, indicando que el bus está llegando. Se envía una tercera notificación cuando el bus se encuentra en la parada seleccionada, indicando que el bus está ahí y debe abordar. Si el bus supera la parada sin que el pasajero haya abordado, el sistema envía una notificación indicando que el bus ya pasó por esa parada. Cada notificación indica la línea, el sentido, la parada y el número de unidad del bus.

Tabla 10. Historia de usuario 8

Historia de usuario	
Número: HU-08	Usuario: pasajero
Nombre de historia: Consulta de horarios por línea y día.	Riesgo en desarrollo: Bajo
Requisitos previos: Deben existir horarios registrados en el sistema y el pasajero debe tener conexión a internet.	
Descripción:	El pasajero debe consultar los horarios de cada línea por día de la semana, para planificar sus viajes con anticipación.
Criterio de validación:	Se pueden consultar horarios de lunes a viernes, sábados y domingos. Los horarios se agrupan por franja horaria con frecuencia. La franja horaria actual se resalta en azul.

Tabla 11. Historia de usuario 9

Historia de usuario	
Número: HU-09	Usuario: chofer

Nombre de historia: Inicio de sesión del chofer	Riesgo en desarrollo: Bajo
Requisitos previos: El chofer debe contar con credenciales registradas por el administrador, dispositivo con conexión a internet y la aplicación instalada.	
Descripción:	El chofer debe tener credenciales registradas por el administrador y conexión a internet.
Criterio de validación:	La aplicación debe permitir el inicio de sesión mediante email y contraseña. La aplicación debe mostrar un mensaje de error si las credenciales son incorrectas. El token debe guardarse localmente. La aplicación debe mostrar el nombre del chofer y número de unidad.

Tabla 12. Historia de usuario 10

Historia de usuario	
Número: HU-10	Usuario: chofer
Nombre de historia: Inicio y fin de servicio del chofer	Riesgo en desarrollo: Alto
Requisitos previos: El chofer debe iniciar y finalizar su servicio desde la aplicación, para que el sistema registre el viaje y envíe su ubicación GPS en tiempo real.	
Descripción:	Como chofer, quiero iniciar y finalizar mi servicio desde la app, para que el sistema registre mi viaje y envíe mi ubicación GPS en tiempo real.
Criterio de validación:	La aplicación debe permitir seleccionar la ruta antes de iniciar el servicio. Al iniciar, la aplicación debe activar el GPS y crear el viaje en el backend. Al finalizar, debe registrar la hora de llegada. El servicio debe cerrarse automáticamente a las 21h00.

Tabla 13. Historia de usuario 11

Historia de usuario	
Número: HU-11	Usuario: chofer
Nombre de historia: GPS en segundo plano	Riesgo en desarrollo: Alto

Requisitos previos: Tener servicio activo con permisos de ubicación "Permitir todo el tiempo".	
Descripción:	La aplicación debe continuar enviando la ubicación GPS del chofer, aunque la pantalla esté apagada, para que los pasajeros puedan visualizar la posición del bus en todo momento.
Criterio de validación:	La aplicación debe continuar enviando la ubicación con la pantalla apagada. La aplicación debe mostrar una notificación persistente de GPS activo. La ubicación debe actualizarse cada 3 segundos. El servicio no debe detenerse al recibir llamadas.

Tabla 14. Historia de usuario 12

Historia de usuario	
Número: HU-12	Usuario: chofer
Nombre de historia: Historial de viajes del chofer	Riesgo en desarrollo: Bajo
Requisitos previos: El chofer debe haber completado al menos un viaje e iniciado sesión.	
Descripción:	El chofer debe consultar el historial de viajes organizados por mes y día, para llevar un control de los recorridos realizados y verificar su puntualidad.
Criterio de validación:	La aplicación debe mostrar todos los viajes del día seleccionado con hora de salida y hora de llegada. Cada viaje debe mostrar la duración del recorrido, la ruta realizada y, si el viaje tuvo atraso, debe indicarse en rojo con los minutos de retraso. El historial debe organizarse por mes y permitir navegar entre días para consultar viajes anteriores. La aplicación debe actualizar automáticamente el historial al finalizar cada viaje.

Tabla 15. Historia de usuario 13

Historia de usuario

Número: HU-13	Usuario: administrador
Nombre de historia: Gestión de usuarios en el panel del administrador.	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión con rol de administrador.	
Descripción:	El administrador debe gestionar los usuarios choferes del sistema, para registrar, editar y desactivar cuentas según las necesidades de la cooperativa.
Criterio de validación:	El sistema debe permitir crear, editar y cambiar el estado de los choferes. El sistema debe registrar el número de unidad de cada chofer. El sistema debe permitir activar o desactivar un chofer. El sistema debe validar que el email no esté duplicado.

Tabla 16. Historia de usuario 14

Historia de usuario	
Número: HU-14	Usuario: administrador
Nombre de historia: Gestión de rutas y paradas	Riesgo en desarrollo: Medio
Requisitos previos: Haber iniciado sesión y debe existir al menos una línea registrada.	
Descripción:	El administrador debe gestionar las rutas y paradas del sistema, para mantener actualizada la información de recorridos de la cooperativa.
Criterio de validación:	El sistema debe permitir que cada línea registrada cuente con dos rutas: IDA y VUELTA. Las paradas deben registrarse con nombre y coordenadas GPS de latitud y longitud obtenidas de ubicaciones reales de la ciudad de Guaranda. El sistema debe permitir reorganizar el orden de las paradas mediante arrastre dentro del panel administrativo. Los cambios realizados en las paradas y rutas deben reflejarse automáticamente en la aplicación del pasajero y en el mapa de monitoreo en tiempo real.

Tabla 17. Historia de usuario 15

Historia de usuario	
Número: HU-15	Usuario: administrador
Nombre de historia: Monitoreo de flota en tiempo real	Riesgo en desarrollo: Alto
Requisitos previos: Haber iniciado sesión y todos los choferes registrados deben tener el servicio activo.	
Descripción:	El administrador debe visualizar en tiempo real la posición de todos los buses de la flota en un mapa interactivo, para monitorear el estado del servicio de la cooperativa.
Criterio de validación:	El sistema debe mostrar todos los buses activos con su posición actualizada en tiempo real mediante WebSocket. Cada bus debe mostrar el número de unidad, el nombre del chofer, la velocidad actual y la parada más cercana a su ubicación. El panel debe indicar cuántos buses están activos, cuántos están en movimiento y cuántos están detenidos en un momento determinado. Al seleccionar un bus en el mapa, el sistema debe desplegar la información detallada de la unidad y su proximidad a la parada más cercana. Unidad y su proximidad a la parada más cercana.

Tabla 18. Historia de usuario 16

Historia de usuario	
Número: HU-16	Usuario: administrador
Nombre de historia: Reportes de viajes por unidad	Riesgo en desarrollo: Alto
Requisitos previos: Haber iniciado sesión y deben existir viajes registrados en el sistema.	
Descripción:	El administrador debe generar reportes filtrando por viaje, unidad y fecha, para evaluar el desempeño de cada chofer de la cooperativa.

Criterio de validación:	El sistema debe permitir filtrar la información por número de unidad y fecha. El sistema debe marcar un viaje con atraso cuando el tiempo de recorrido supere el límite configurado por el administrador para cada línea. Además, el sistema debe permitir descargar el reporte en formato PDF.
-------------------------	---

Tabla 19. Historia de usuario 17

Historia de usuario	
Número: HU-17	Usuario: administrador
Nombre de historia: Gestión de horarios por línea	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión y debe existir al menos una línea registrada.	
Descripción:	El administrador debe gestionar los horarios de cada línea por día de la semana, para mantener informados a los pasajeros sobre los tiempos de servicio.
Criterio de validación:	El sistema debe permitir registrar horarios para lunes a viernes, sábados y domingos. Los horarios deben mostrarse en la aplicación del pasajero. Los cambios deben reflejarse inmediatamente.

Tabla 20. Historia de usuario 18

Historia de usuario	
Número: HU-18	Usuario: administrador
Nombre de historia: Dashboard de estadísticas	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión y deben existir datos registrados en el sistema.	
Descripción:	El administrador debe visualizar un dashboard con estadísticas generales del sistema, para tener una visión rápida del estado de la flota y el servicio.
Criterio de validación:	Se muestra total de buses activos, viajes del día, pasajeros suscritos y notificaciones enviadas. Los datos se actualizan automáticamente

Tabla 21 Historia de usuario 19

Historia de usuario	
Número: HU-19	Usuario: administrador
Nombre de historia: Notificaciones en tiempo real de reportes de flota	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión con rol de administrador y tener al menos un chofer con servicio activo en el sistema.	
Descripción:	El administrador debe visualizar un dashboard con estadísticas generales del sistema, para tener una visión rápida del estado de la flota y el servicio.
Criterio de validación:	El sistema debe mostrar el total de buses activos, viajes del día, pasajeros suscritos y notificaciones enviadas. Los datos deben actualizarse automáticamente.

Tabla 22 Historia de usuario 20

Historia de usuario	
Número: HU-20	Usuario: administrador
Nombre de historia: Reporte de estado de unidad	Riesgo en desarrollo: Bajo
Requisitos previos: El chofer debe haber iniciado sesión y no tener un permiso o descanso activo registrado por el administrador para la fecha actual.	
Descripción:	El sistema debe permitir reportar el estado operativo de la unidad directamente desde la aplicación móvil, para informar al administrador si el bus se encuentra en línea, en mantenimiento o con una avería en tiempo real.
Criterio de validación:	La aplicación debe mostrar un selector con tres opciones de estado: en línea, en mantenimiento y avería. Al confirmar el cambio de estado, el sistema debe reflejar la actualización de forma inmediata en el panel administrativo. El botón de reporte de estado no debe

	mostrarse cuando el chofer tenga un permiso o descanso activo asignado por el administrador. El sistema debe enviar una notificación al administrador en la campana de reportes cada vez que un chofer registre un cambio de estado.
--	--

Tabla 23 Historia de usuario 21

Historia de usuario	
Número: HU-21	Usuario: administrador
Nombre de historia: Gestión de despacho semanal	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión con rol de administrador y tener choferes registrados en el sistema.	
Descripción:	El administrador debe asignar las unidades de transporte a cada línea mediante un calendario semanal, para organizar el despacho diario de la flota de la cooperativa.
Criterio de validación:	El sistema debe mostrar un calendario con los días lunes a domingo y las líneas L1, L2, L3, L4, L5 y L6. El sistema debe permitir asignar una o varias unidades a cada línea por día seleccionado. Los días anteriores a la fecha actual deben quedar bloqueados y no permitir modificaciones. El sistema debe permitir copiar las asignaciones realizadas en un día al siguiente día con un solo clic.

Tabla 24 Historia de usuario 22

Historia de usuario	
Número: HU-22	Usuario: administrador
Nombre de historia: Gestión de permisos y descansos de choferes	Riesgo en desarrollo: Bajo
Requisitos previos: Haber iniciado sesión con rol de administrador y tener choferes asignados en el despacho del día.	

Descripción:	El administrador debe registrar permisos y descansos de los choferes por día específico o rango de fechas, para bloquear automáticamente el inicio de servicio en la aplicación del chofer durante esos períodos.
Criterio de validación:	El sistema debe permitir registrar una ausencia de tipo descanso o permiso seleccionando el día o un rango de fechas desde y hasta. El sistema debe impedir que el chofer con ausencia activa inicie servicio y visualice el botón de estado de unidad en su aplicación. El panel de flota debe mostrar el ícono correspondiente al tipo de ausencia registrada. La restricción debe eliminarse automáticamente cuando la fecha de fin del período haya sido superada.

Adicionalmente, se seleccionó el stack tecnológico del proyecto considerando los requerimientos de rendimiento, escalabilidad y compatibilidad con los dispositivos móviles disponibles en el contexto local. Se configuró el entorno de desarrollo tanto de manera local como en el servidor de producción, dejando lista la infraestructura necesaria para iniciar el diseño del sistema en el sprint siguiente.

Tabla 25. Stack Tecnológico Definido — Sprint 1

Componente	Tecnología	Versión
Backend	Laravel	11
Lenguaje servidor	PHP	8.4
Base de Datos	PostgreSQL	16
WebSocket	Soketi	Latest
App Pasajero	React Native / Expo	SDK 52
App Chofer	React Native / Expo	SDK 52
Panel Administrativo	Next.js / TypeScript	14
Notificaciones push	Firebase Cloud Messaging	v1 API
Notificaciones locales	Notifee	9.x
GPS Background	Expo Taso Manager	12

Servidor	Digitalicen Ubuntu	22.04
Dominio	unibolivartransit.com	-

4.3 SPRINT 2: DISEÑO DEL SISTEMA

4.3.1 Objetivo del Sprint 2

Diseñar la arquitectura del sistema, los modelos de datos y los prototipos de interfaz de las tres aplicaciones que componen UniBolívar Transit.

4.3.2 Actividades Principales

- Diseño del modelo entidad-relación
- Modelo relacional
- Diagrama de clases
- Diagrama de secuencia
- Diagrama de actividades
- Diagrama de componentes y despliegue
- Guía de estilos
- Diseño de prototipos de interfaz

4.3.3 Sprint Backlog

Tabla 26. Sprint Backlog del Sprint 2.

Id	Tarea	Estimación	Encargados	Etapas
T-05	Diseño del modelo entidad-relación	4h	Erick Alarcón, Juan Pasto	Completado

T-06	Diseño del modelo relacional	8h	Erick Alarcón, Juan Pasto	Completado
T-07	Diagrama de clases	6h	Erick Alarcón, Juan Pasto	Completado
T-08	Diagramas de secuencia	10h	Erick Alarcón, Juan Pasto	Completado
T-09	Diagrama de actividades	4h	Erick Alarcón, Juan Pasto	Completado
T-10	Diagramas de componentes y despliegue	6h	Erick Alarcón, Juan Pasto	Completado
T-11	Elaboración del diagrama de componentes y despliegue	12h	Erick Alarcón, Juan Pasto	Completado
T-12	Definición de guía de estilos y diseño de prototipos de interfaz de las 3 aplicaciones	3h	Erick Alarcón, Juan Pasto	Completado

4.3.4 Resultados del Sprint 2

Para el cumplimiento del objetivo del Sprint 2, se elaboraron los artefactos de diseño del sistema que sirvieron como base para el desarrollo en el sprint siguiente. A continuación, se presentan los resultados obtenidos.

Ilustración 2 Diagrama Entidad-Relación de UniBolívar Transit

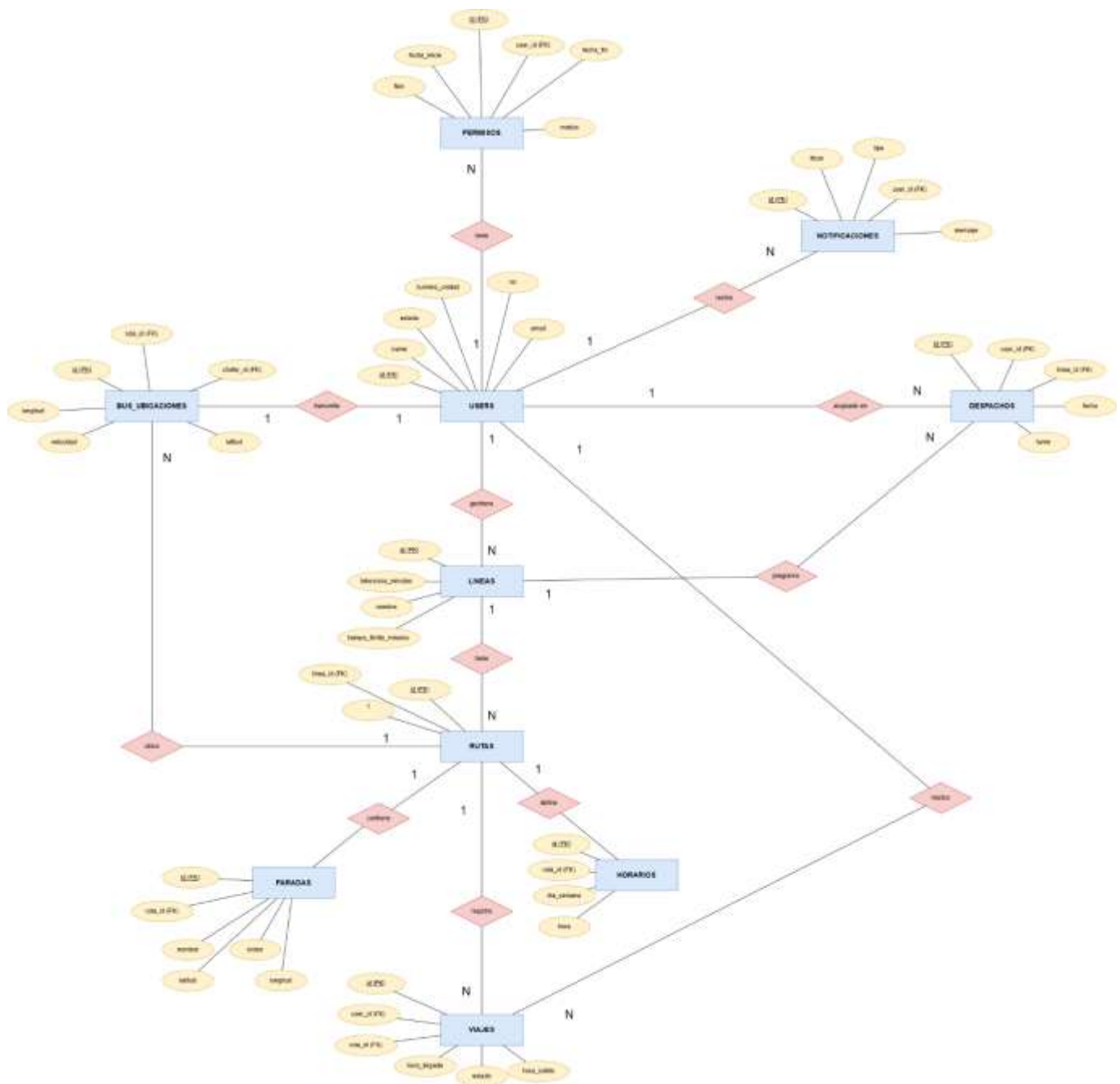


Ilustración 3 Modelo relacional de UniBolivar Transit

El modelo relacional del sistema UniBolivar Transit está compuesto por 10 tablas. Los campos subrayados corresponden a claves primarias (PK) y los campos en cursiva corresponden a claves foráneas (FK).

USERS (id, name, email, password, rol, numero_unidad, estado, created_at, updated_at)

LINEAS (id, nombre, tiempo_limite_minutos, tolerancia_minutos, created_at, updated_at)

RUTAS (id, linea_id, sentido, created_at, updated_at)

- linea_id → LINEAS.id

PARADAS (id, ruta_id, nombre, orden, latitud, longitud, created_at, updated_at)

- ruta_id → RUTAS.id

HORARIOS (id, ruta_id, hora, dia_semana, created_at, updated_at)

- ruta_id → RUTAS.id

VIAJES (id, user_id, ruta_id, hora_salida, hora_llegada, estado, velocidad_max, created_at, updated_at)

- user_id → USERS.id

- ruta_id → RUTAS.id

BUS_UBICACIONES (id, chofer_id, ruta_id, latitud, longitud, velocidad, updated_at)

- chofer_id → USERS.id

- ruta_id → RUTAS.id

DESPACHOS (id, user_id, linea_id, fecha, turno, created_at, updated_at)

- user_id → USERS.id

- linea_id → LINEAS.id

PERMISOS (id, user_id, fecha_inicio, fecha_fin, tipo, motivo, created_at, updated_at)

- user_id → USERS.id

NOTIFICACIONES (id, user_id, titulo, mensaje, tipo, created_at)

- user_id → USERS.id

Ilustración 4 Diagrama de Clases de UniBolivar Transit

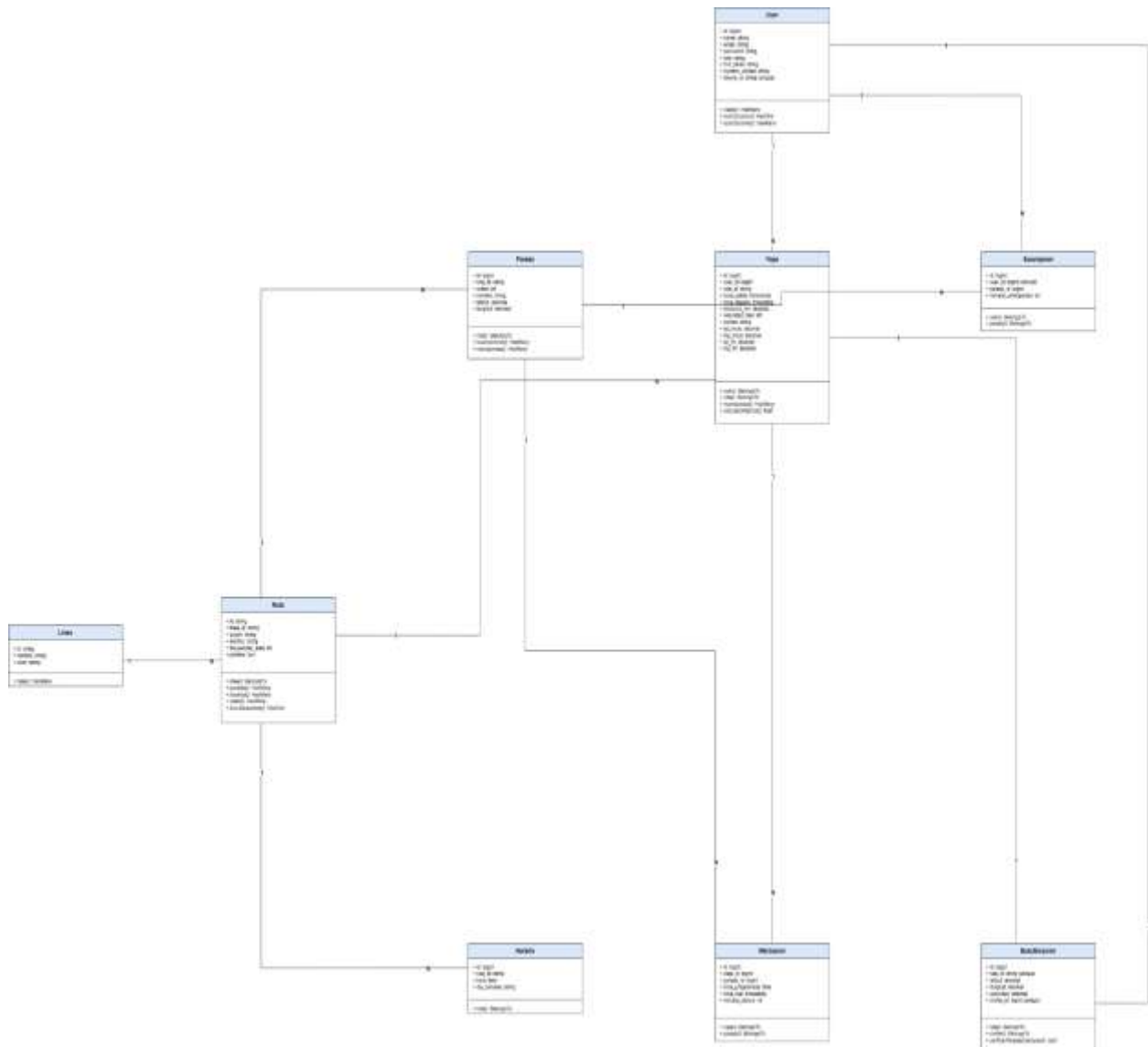


Ilustración 5 Diagramas de Secuencia Gestión del Administrador



Ilustración 6 Diagrama de secuencia del proceso de inicio de sesión (chofer y administrador).

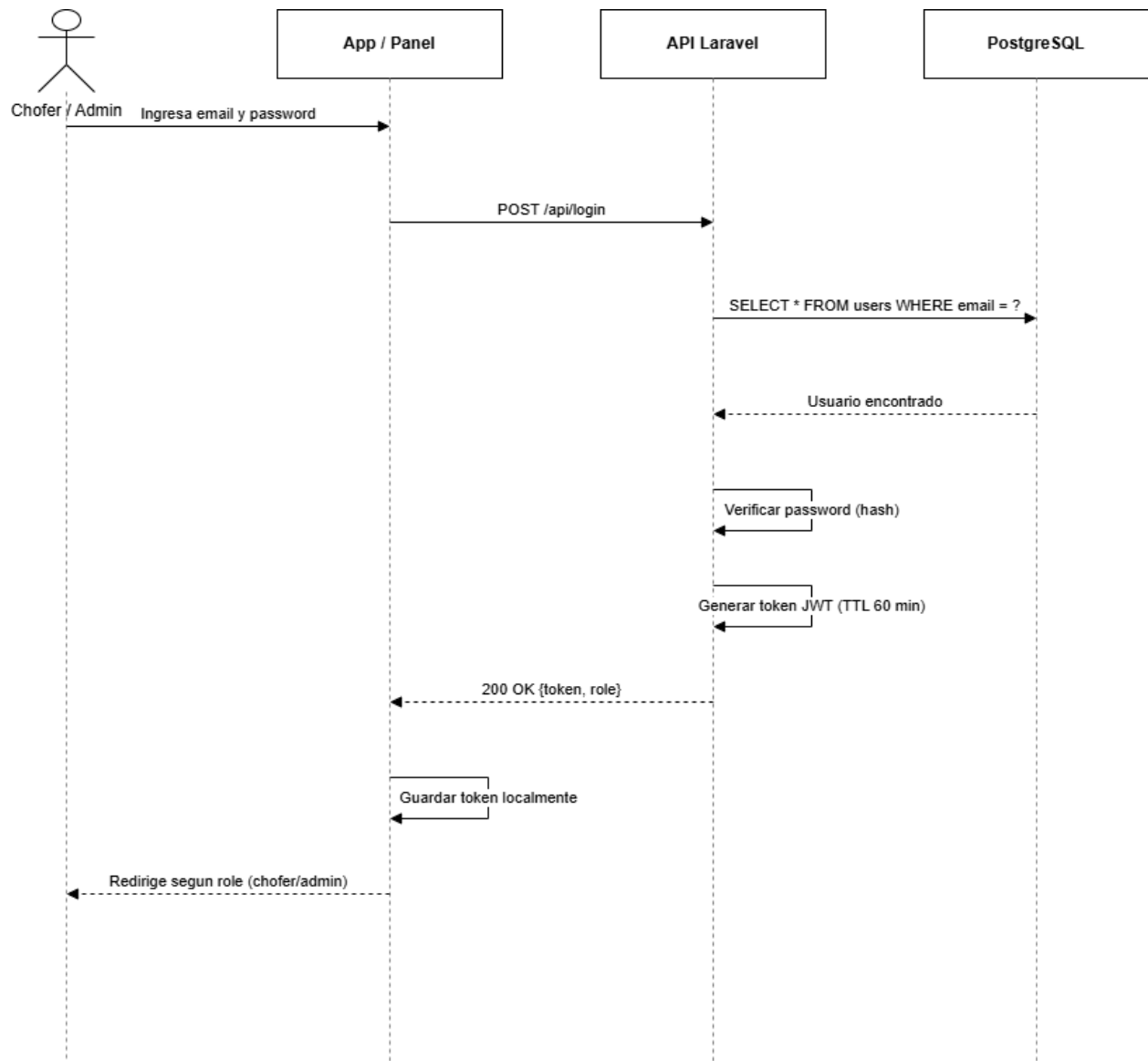


Ilustración 7 Diagrama de secuencia del proceso de geolocalización y notificación al pasajero.

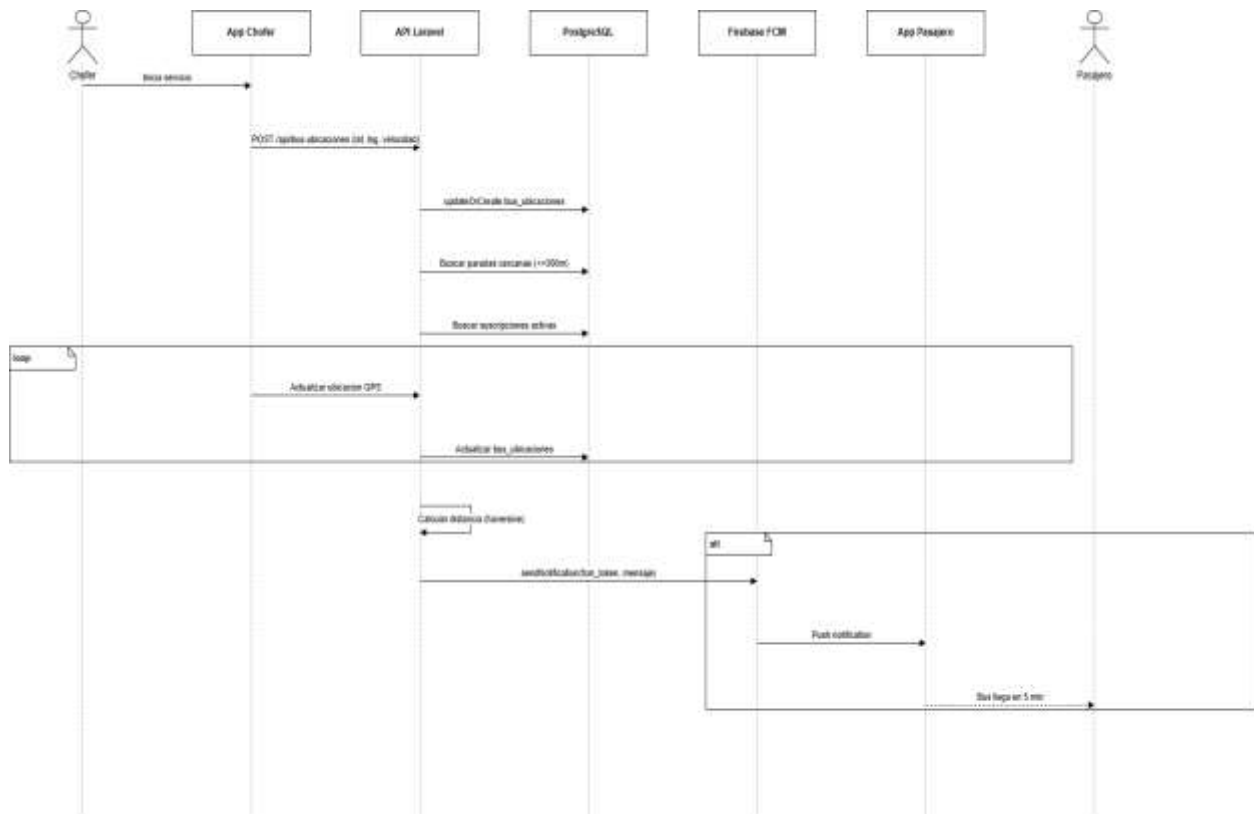


Ilustración 8 Diagrama de secuencia del proceso de suscripción a notificaciones de parada.

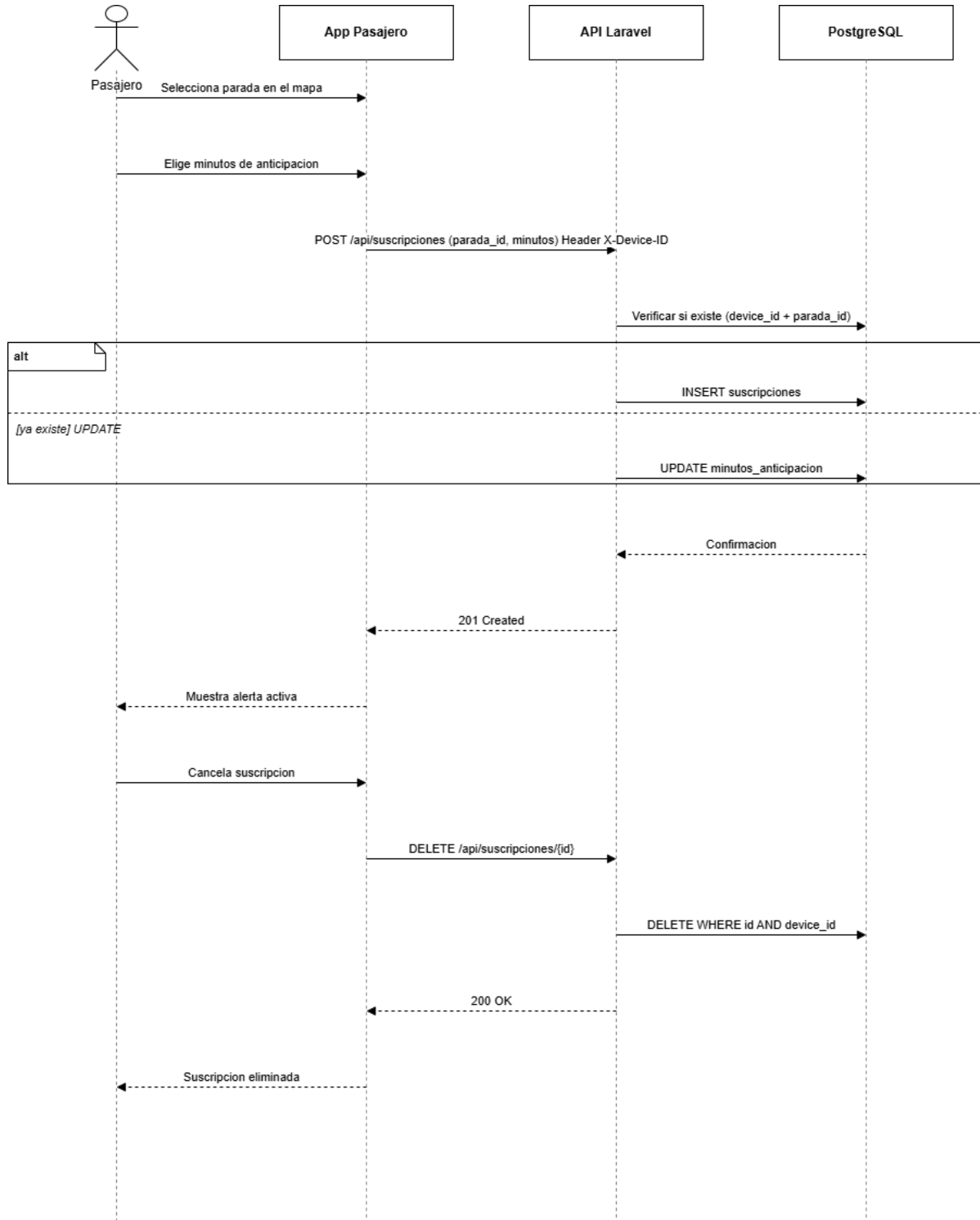


Ilustración 9 Diagrama de secuencia del proceso de registro de viaje y marcaciones

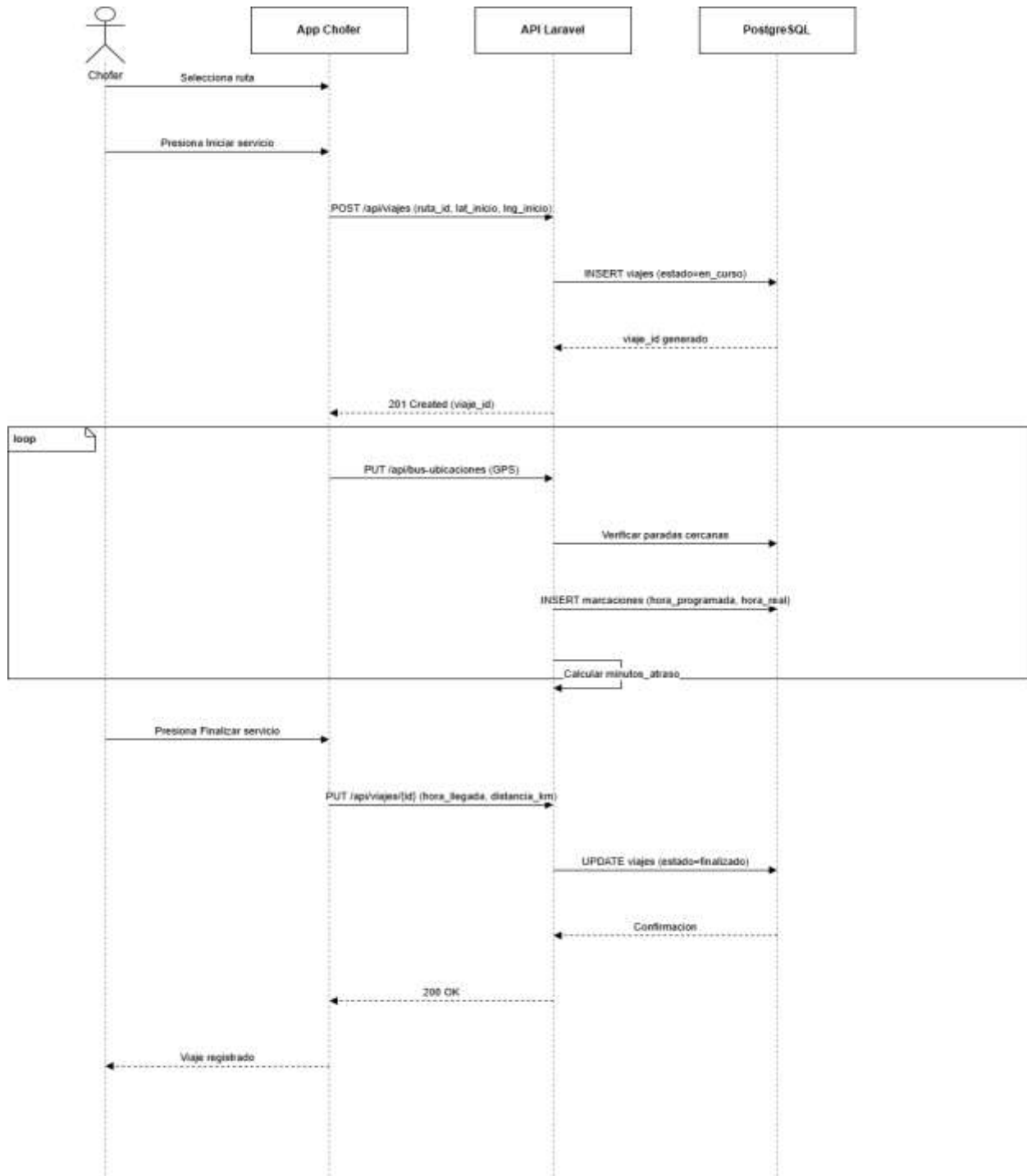


Ilustración 10 Diagrama de Actividades del Seguimiento y Notificación de Buses en Tiempo Real

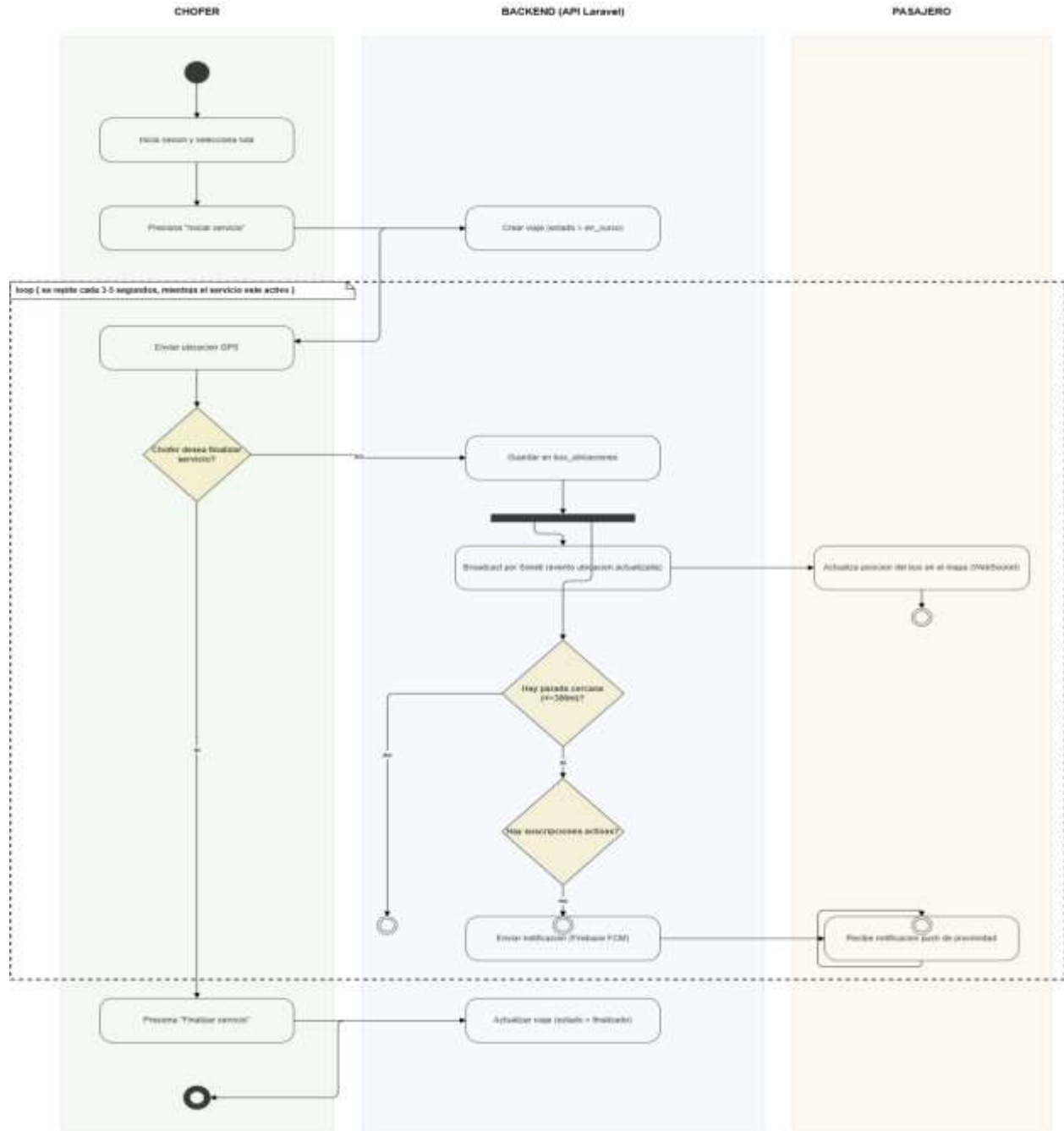


Ilustración 11 Diagrama de Componentes de tu sistema UniBolivar Transit.

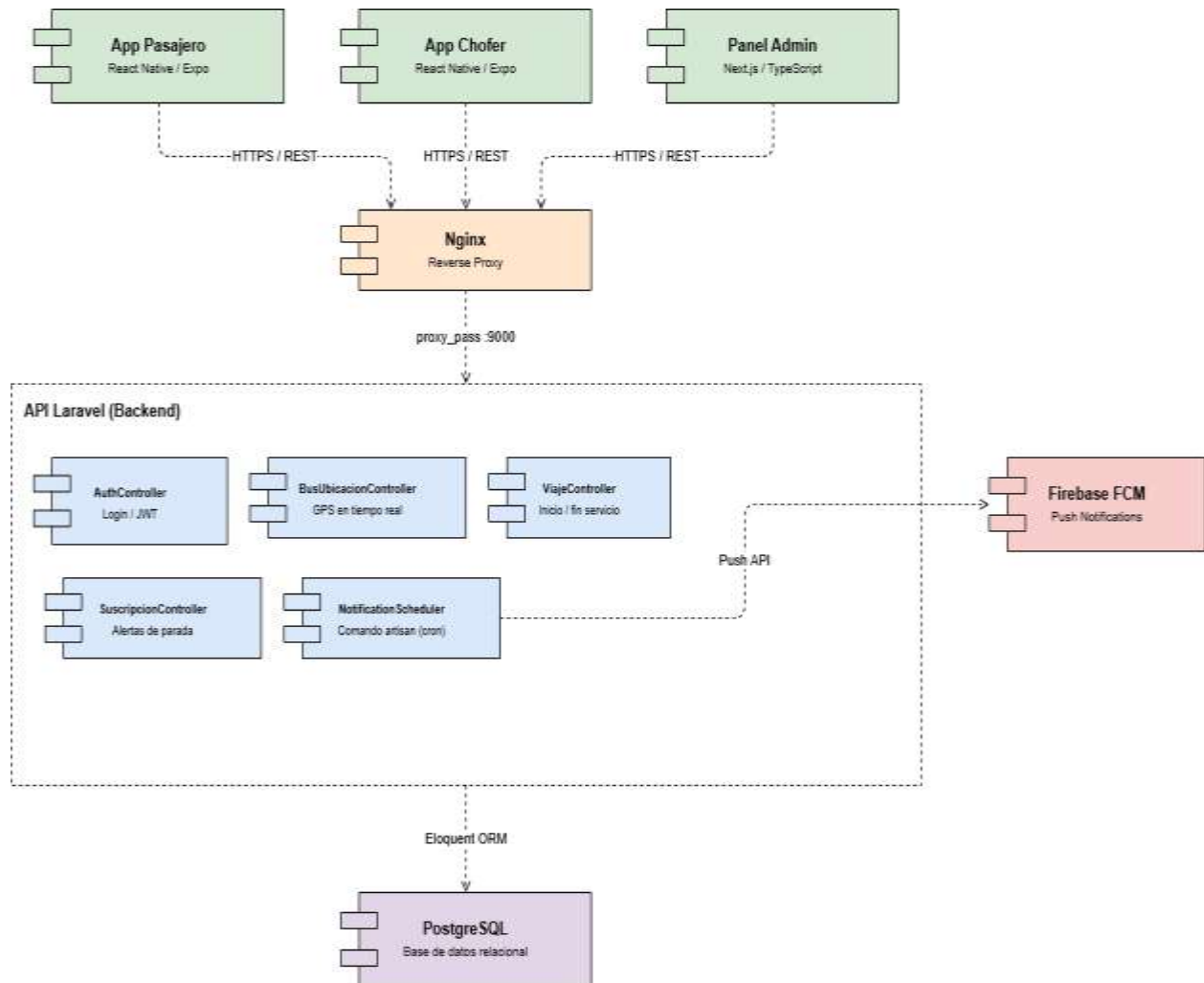
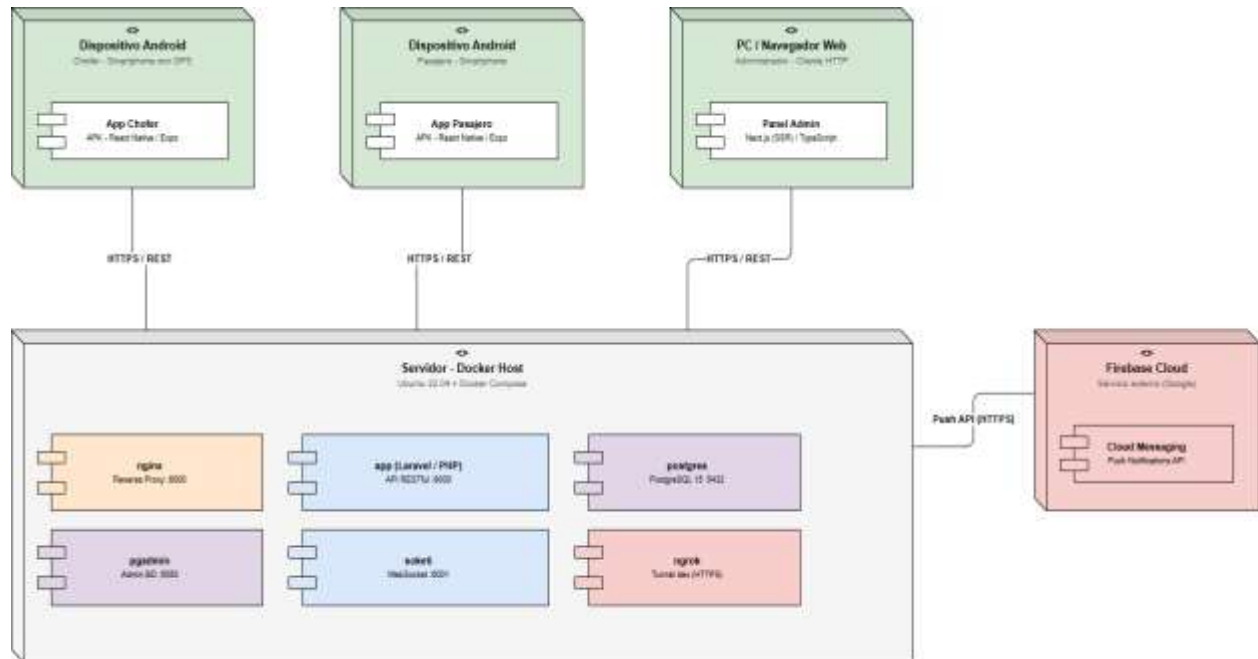


Ilustración 12 Diagrama de Despliegue de UniBolívar Transit



Diseño de Interfaz del Sistema UniBolívar Transit: plataforma web y aplicación móvil

Este documento desarrolla los prototipos de interfaz y explica la finalidad de cada estructura diferente. Los lineamientos visuales se definen una sola vez para toda la solución; después se desglosan las pantallas web y móviles sin repetir capturas o explicaciones equivalentes.

1. Lineamientos globales de diseño

1.1 Alcance

Los lineamientos se aplican al Panel Administrativo web, la aplicación móvil del pasajero y la aplicación móvil del chofer. Su propósito es mantener coherencia en colores, tipografía, botones, textos, imágenes, iconografía, formularios, tablas, tarjetas, mapas y estados.

1.2 Colores e identificación de rutas

El naranja es el color institucional y se reserva para acciones principales, elementos activos y llamados de atención. El blanco y los grises claros forman superficies y fondos; el azul oscuro o gris carbón se utiliza en textos, navegación y mapas. Cada línea conserva un color distintivo en su tarjeta, etiqueta, recorrido, marcadores e indicadores. Verde representa un estado normal o disponible; amarillo o ámbar, advertencia o pendiente; y rojo, avería, error o acción crítica.

1.3 Identificación de paradas y buses

Las paradas se muestran con marcadores y elementos de lista relacionados con la ruta seleccionada. Según la pantalla, incluyen nombre, orden, distancia, coordenadas, tiempo estimado y estado GPS. La parada seleccionada o más cercana utiliza mayor contraste. El bus emplea un símbolo diferente al marcador de parada, evitando confusiones en el mapa.

1.4 Tipografía, textos e imágenes

Se utiliza una tipografía sans-serif legible en tres niveles jerárquicos. Los títulos principales emplean un tamaño de 1.5rem (24px) con peso 800; los subtítulos, etiquetas y botones utilizan tamaños entre 0.82rem y 0.875rem (13-14px) con peso 600 o 700; y los textos secundarios, ayudas, fechas y metadatos se presentan en 0.68rem a 0.75rem (11-12px) con peso regular 400 o 500. Los mensajes son breves y consistentes en todo el sistema. Los logotipos, iconos, fotografías e

ilustraciones mantienen proporciones uniformes, resolución suficiente y cumplen una función informativa dentro de la interfaz.

1.5 Botones y componentes

Los botones primarios en naranja (#F97316) se utilizan para acciones de creación, guardado, ingreso, inicio o confirmación. Los botones secundarios, con fondo blanco o contorno (#e2e8f0), permiten volver, cancelar, filtrar o consultar información. El rojo (#dc2626) se reserva exclusivamente para eliminar registros, finalizar servicios o reportar una avería. Adicionalmente, se normalizan los siguientes componentes en toda la interfaz: campos de texto, tarjetas, tablas, filtros, pestañas, selectores, chips, badges, alertas, calendarios, buscadores, mapas y estados vacíos.

1.6 Tipo de mapa utilizado

El sistema utiliza OpenStreetMap como proveedor de mapas, implementado mediante la biblioteca Leaflet.js en el panel administrativo web y React Native Maps en las aplicaciones móviles. Se emplea el estilo de mapa base estándar de OpenStreetMap sin subdominios, con actualización de marcadores cada 3 segundos mediante tecnología WebSocket. Esta combinación garantiza disponibilidad del servicio sin costo de licencia y compatibilidad con los dispositivos Android utilizados en el contexto local de Guaranda.

Tabla 27 Aplicación de los lineamientos de diseño en entornos web y móvil.

Entorno	Estructura	Tipos de pantallas
Web	Menú lateral fijo, área de trabajo amplia, tablas, filtros e indicadores en columnas.	Acceso; dashboard y estadísticas; registros; configuración; monitoreo; despacho; reportes.
Móvil	Estructura vertical, controles táctiles, tarjetas, listas, navegación inferior y paneles sobre el mapa.	Bienvenida y acceso; consulta o selección; operación en tiempo real; alertas; estados; historial.

Ilustración 13 Sistema web: Página de inicio - vista principal



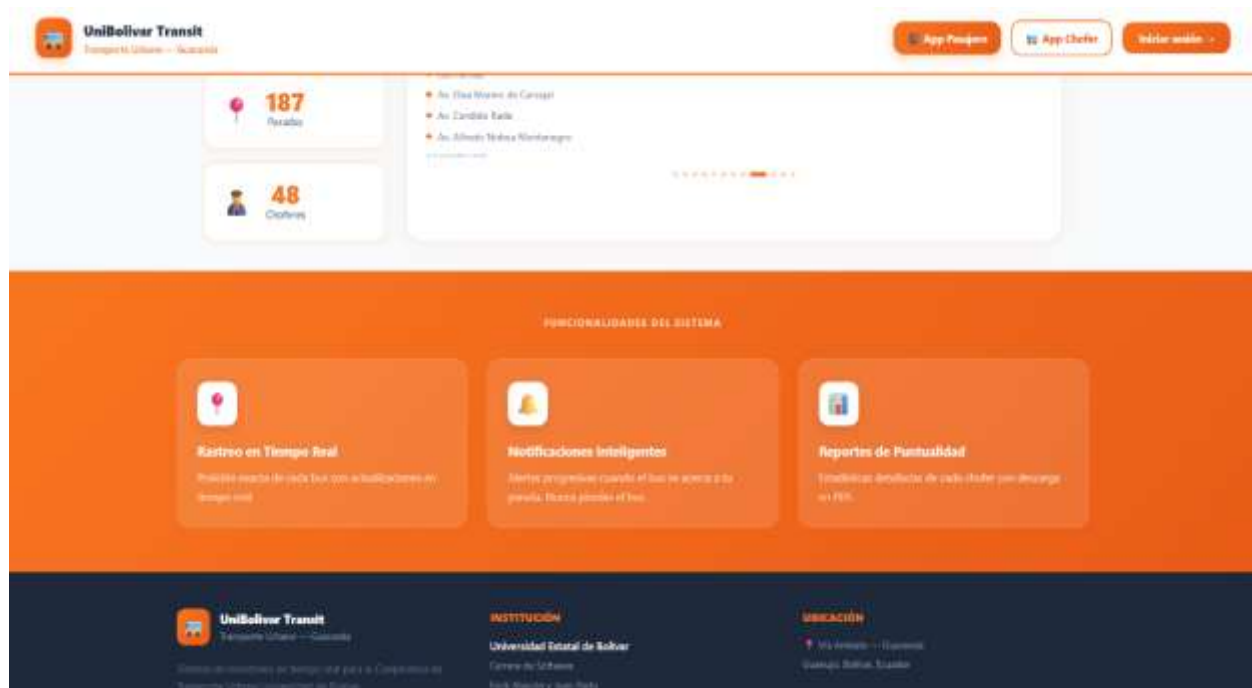
Descripción: La página de inicio presenta la identidad visual de UniBolívar Transit y el propósito principal del sistema, orientado al monitoreo en tiempo real del transporte urbano de la ciudad de Guaranda. En la parte superior se muestran el estado del servicio y los accesos directos a la App Pasajero, App Chofer y al módulo de inicio de sesión, permitiendo acceder a las diferentes funcionalidades de acuerdo con el tipo de usuario.

Además, la página incorpora la opción “Descargar App”, mediante la cual los usuarios pueden acceder a las aplicaciones móviles desarrolladas para el sistema. Esta opción facilita la distribución de las aplicaciones destinadas a pasajeros y choferes, permitiendo que cada usuario pueda obtener la herramienta correspondiente para utilizar las funcionalidades de consulta, monitoreo y gestión del servicio de transporte.

En la sección principal se presenta el nombre UniBolívar Transit, acompañado de una imagen representativa de las unidades de transporte urbano y una descripción del servicio de monitoreo. También se incluyen indicadores generales de la operación, correspondientes al número de líneas, buses, paradas y choferes registrados. Finalmente, se presenta información resumida de los recorridos, mostrando el sentido de circulación, el origen, el destino y las principales paradas de cada línea, proporcionando una visión general de la oferta de transporte urbano.

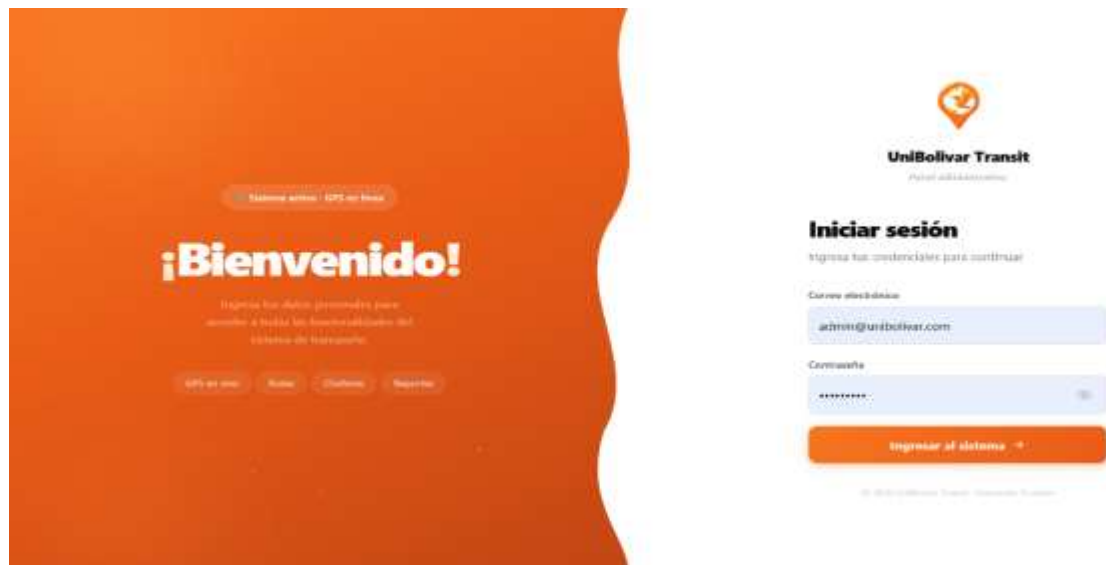
La plataforma web se encuentra disponible en <https://unibolivartransit.com>

Ilustración 14 Página de inicio - funcionalidades e información institucional



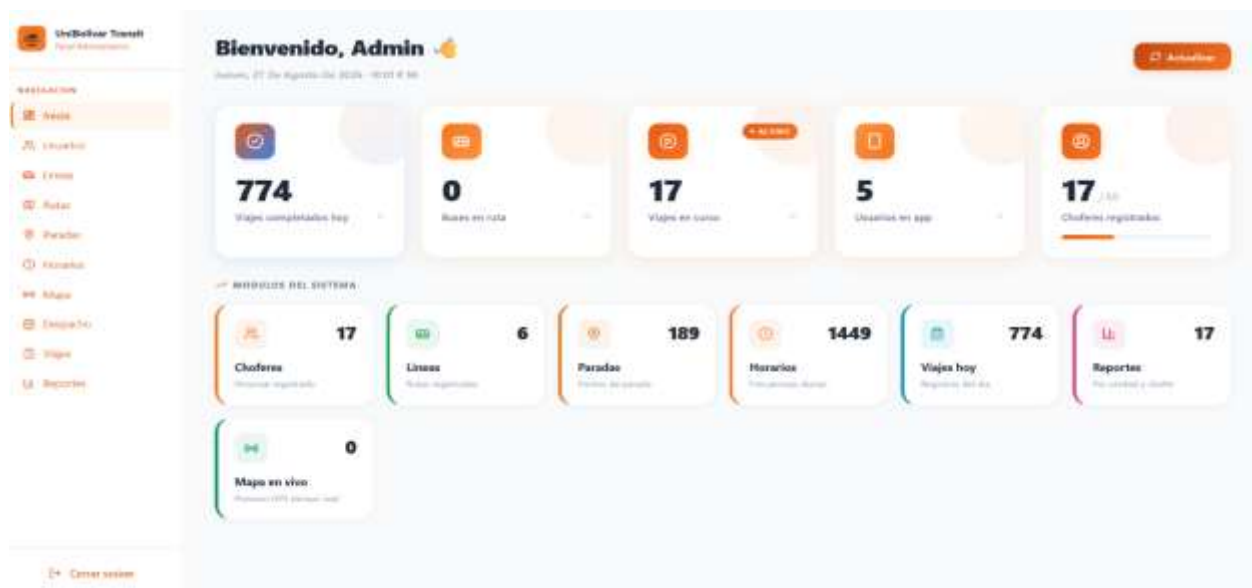
Descripción: Complementa la página principal mediante tarjetas que resumen las funcionalidades más importantes del sistema: rastreo en tiempo real, notificaciones inteligentes y reportes de puntualidad. Además, incorpora una opción para la descarga de las aplicaciones móviles del sistema, permitiendo a los usuarios acceder e instalar la App Pasajero y la App Chofer. En la parte inferior se incluye el pie de página con información institucional, ubicación y referencia del proyecto. La página principal se encuentra disponible en

Ilustración 15 Inicio de sesión del Panel Administrativo



Descripción: Permite autenticar al administrador mediante correo electrónico y contraseña. La pantalla separa la presentación institucional del formulario de acceso y dirige al usuario autorizado hacia el panel de gestión una vez que las credenciales son validadas.

Ilustración 16 Dashboard del Panel Administrativo



Descripción: Ofrece una vista consolidada del estado general del sistema. Presenta indicadores como viajes completados, buses en ruta, viajes en curso, usuarios y choferes registrados, además de accesos resumidos a los principales módulos administrativos. Su finalidad es facilitar una lectura rápida de la operación diaria.

Ilustración 17 Gestión de usuarios



USUARIO	ROL	UNIDAD	FECHA DE REGISTRO	ACCIONES
Andrés Santillán chofer018@unibolivar.com	Chofer	01	21 de agosto de 2026	Editar Eliminar
Marco Villaci chofer007@unibolivar.com	Chofer	03	21 de agosto de 2026	Editar Eliminar
Diego Paredes chofer003@unibolivar.com	Chofer	07	21 de agosto de 2026	Editar Eliminar
Roberto Gavilanes chofer004@unibolivar.com	Chofer	18	21 de agosto de 2026	Editar Eliminar
Pablo Endara chofer007@unibolivar.com	Chofer	22	21 de agosto de 2026	Editar Eliminar
Javier Salas chofer010@unibolivar.com	Chofer	25	21 de agosto de 2026	Editar Eliminar
Miguel Chango				

Descripción: Permite administrar las cuentas registradas en el sistema, muestra el total de usuarios, administradores y choferes, y organiza la información en una tabla con nombre, rol, unidad asignada, fecha de registro y acciones, desde esta interfaz se pueden crear nuevos usuarios, editar datos o eliminar registros según corresponda.

Ilustración 18 Gestión de líneas

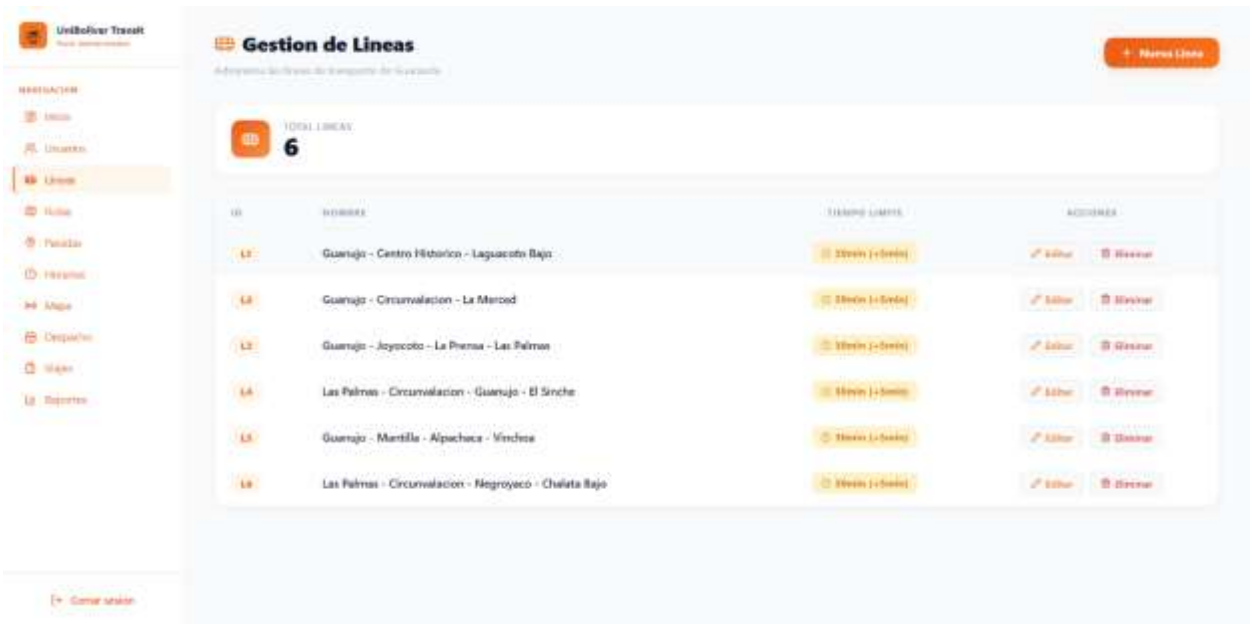
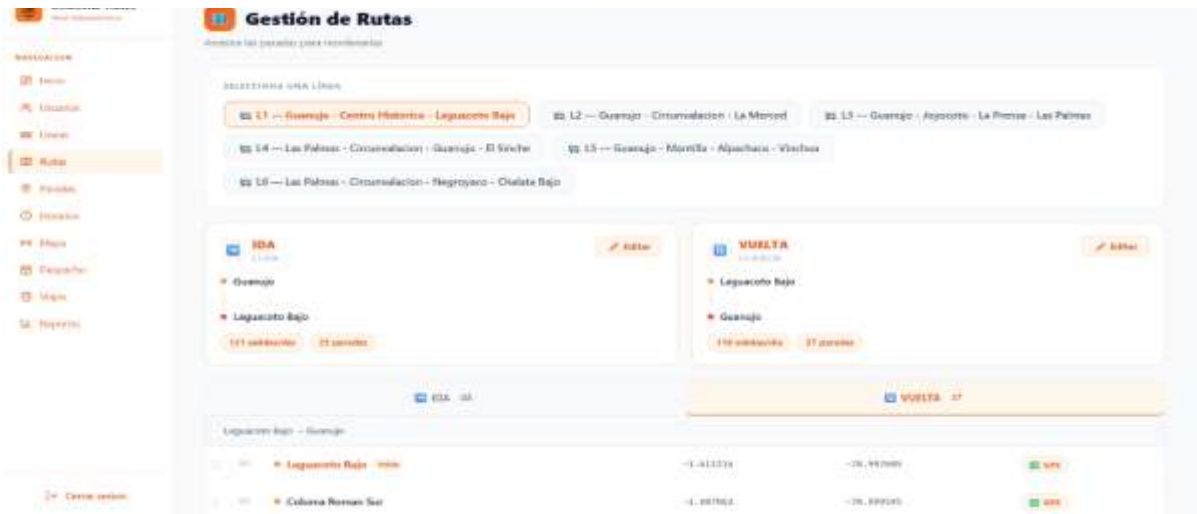


Ilustración Gestión de rutas

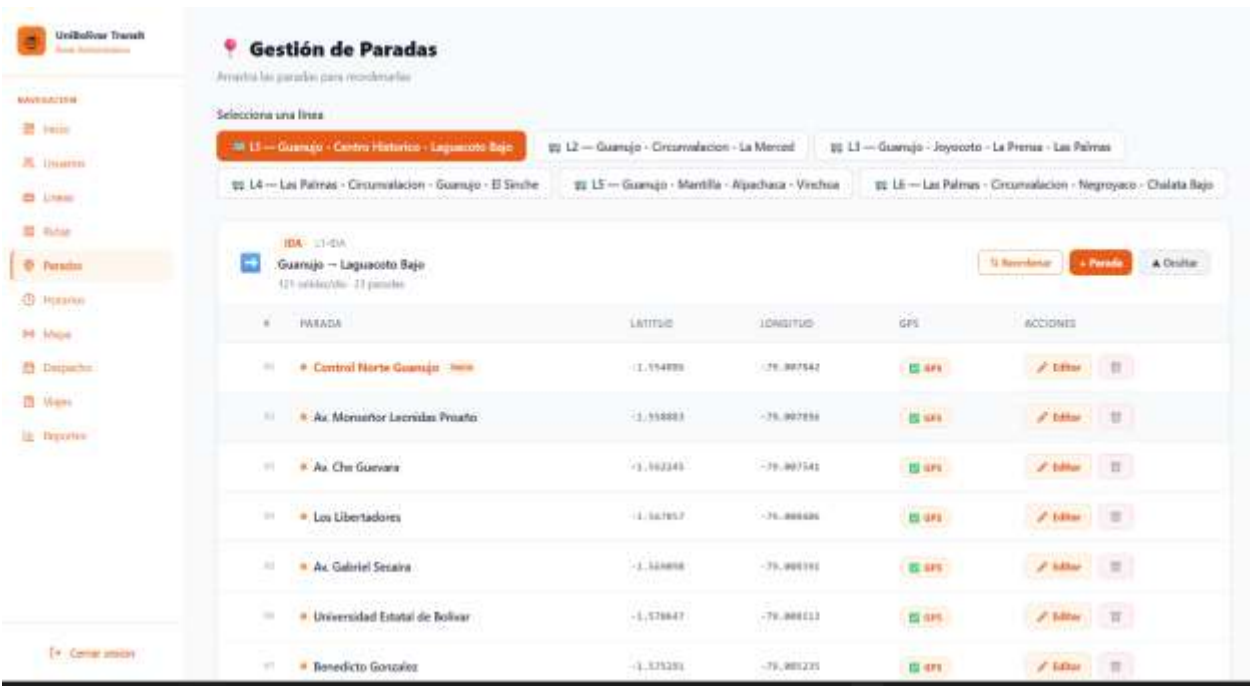
Descripción: Permite consultar y administrar las líneas de transporte urbano, presenta el identificador de cada línea, el nombre del recorrido y el tiempo límite configurado para el viaje. Incluye acciones para crear, editar o eliminar líneas, manteniendo actualizada la estructura general de recorridos del sistema.

Ilustración 18 Gestión de rutas



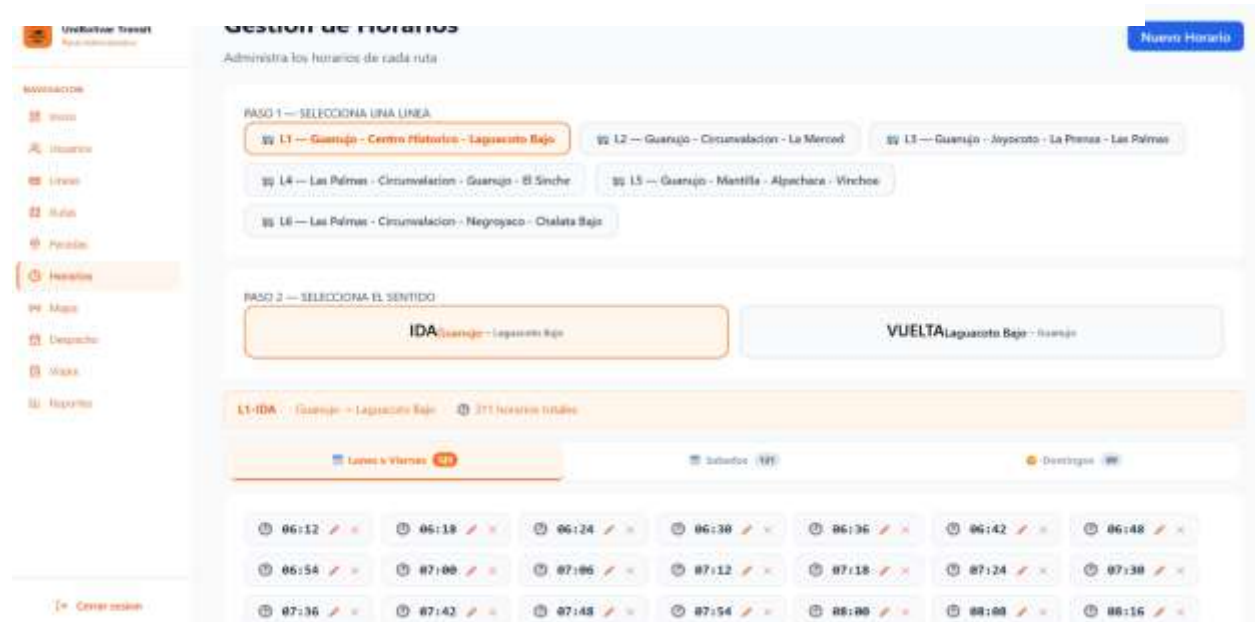
Descripción: Permite seleccionar una línea y revisar sus recorridos de ida y vuelta, la interfaz muestra origen, destino, cantidad de salidas y número de paradas, además del listado ordenado de paradas asociado a cada sentido. Facilita la edición de la configuración de cada ruta y la organización de su recorrido.

Ilustración 19 Gestión de paradas



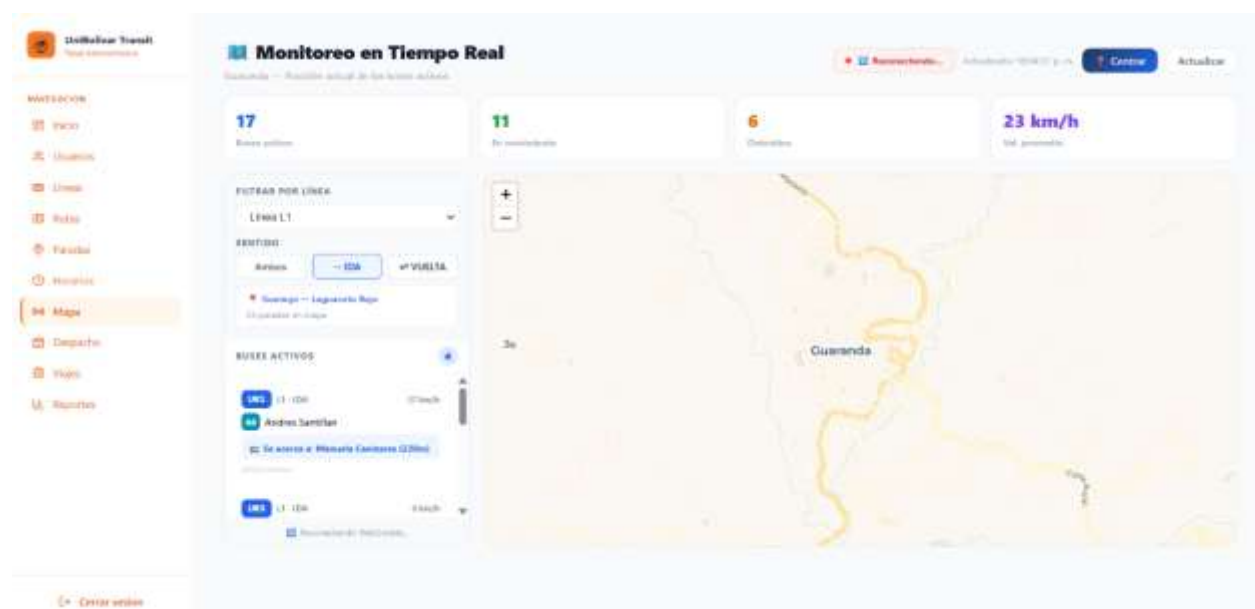
Descripción: Permite administrar las paradas georreferenciadas de cada ruta. Muestra el nombre de la parada, su posición en la secuencia, latitud, longitud y estado GPS. También ofrece funciones para agregar, editar, eliminar y reordenar paradas, garantizando que el recorrido mostrado en las aplicaciones corresponda con la configuración registrada.

Ilustración 20 Gestión de horarios



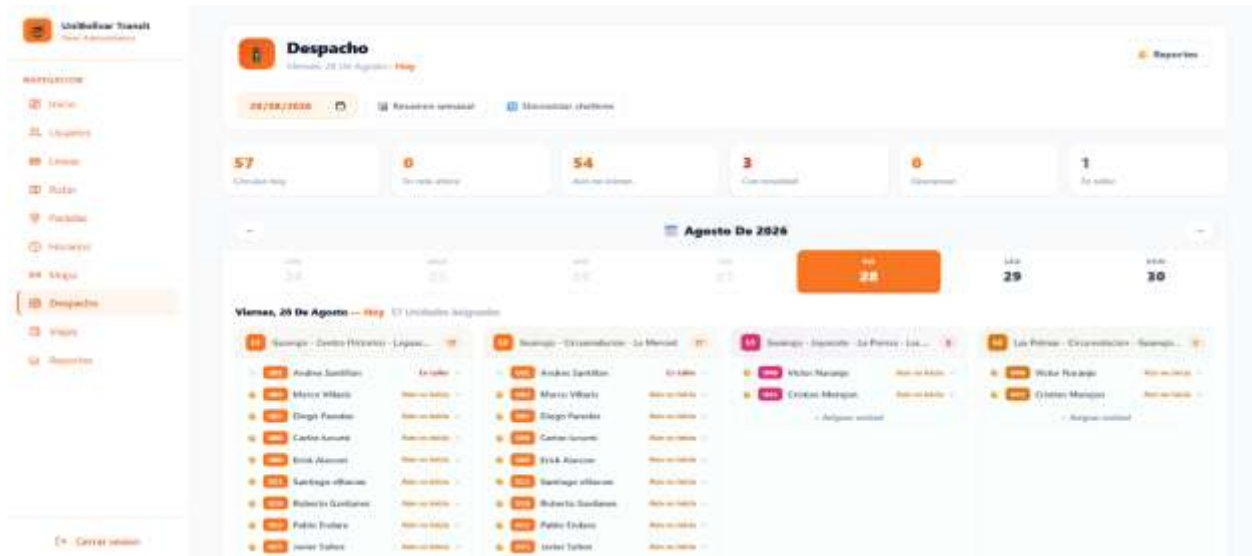
Descripción: Permite configurar y consultar los horarios de servicio por línea, sentido y tipo de día. El administrador puede seleccionar una línea, escoger ida o vuelta y revisar los horarios correspondientes a días laborables, sábados y domingos. Incluye acciones para crear, editar o eliminar horarios según la programación de la cooperativa.

Ilustración 21 Monitoreo en tiempo real



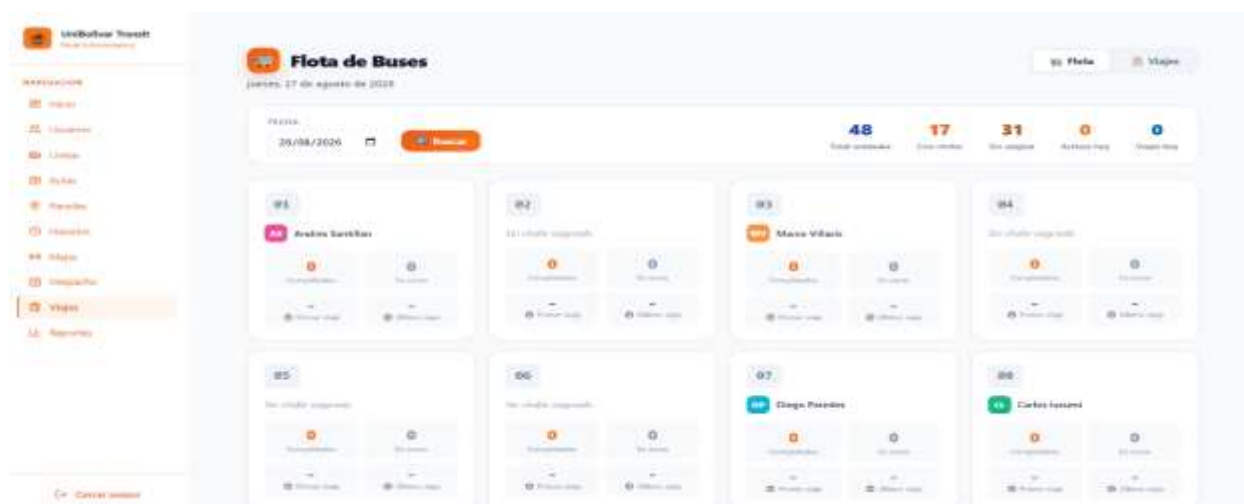
Descripción: Permite supervisar la ubicación de las unidades activas sobre el mapa de Guaranda. La interfaz muestra indicadores de buses activos, en movimiento, detenidos y velocidad promedio; además permite filtrar por línea y sentido. El listado lateral presenta datos de las unidades y su proximidad a las paradas, mientras que el mapa refleja la posición GPS recibida en tiempo real.

Ilustración 22 Despacho



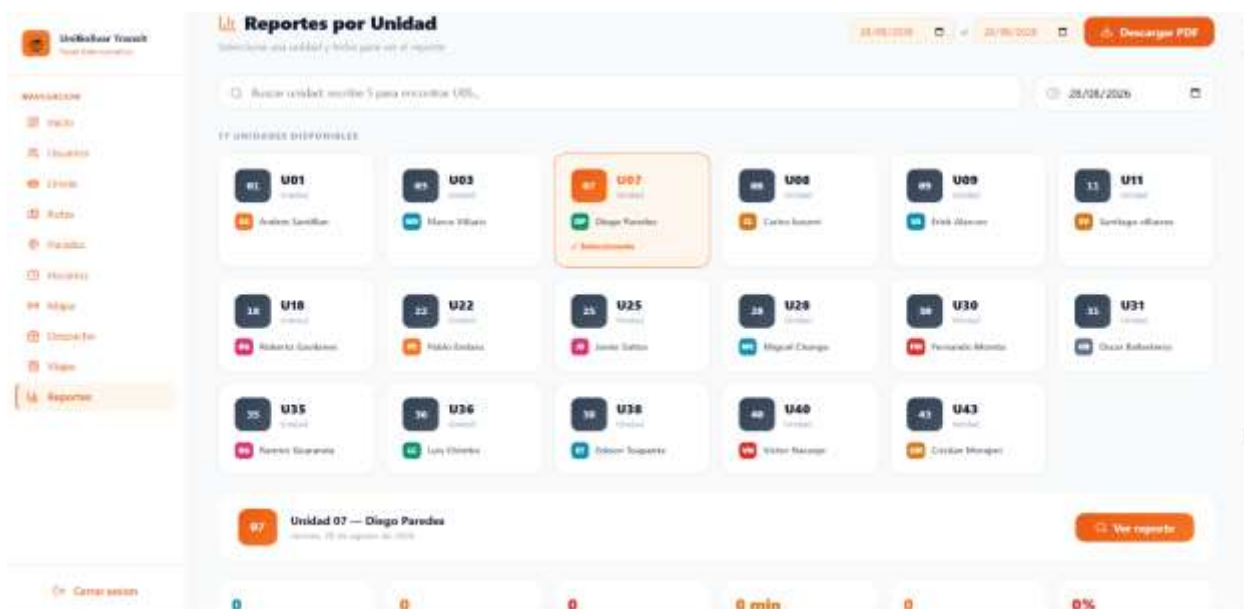
Descripción: Permite organizar la asignación diaria y semanal de unidades a las diferentes líneas. Integra un calendario, indicadores de estado y columnas de distribución por recorrido. El administrador puede verificar qué unidades circulan, cuáles aún no inician, cuáles presentan novedades, descansos o mantenimiento, y realizar ajustes en la programación operativa.

Ilustración 23 Flota de buses



Descripción: Permite consultar el estado general de las unidades y su relación con los choferes. La interfaz muestra el total de unidades, cuántas tienen conductor asignado, cuántas permanecen sin asignación, los buses activos y los viajes del día, cada tarjeta resume los recorridos completados, viajes en curso y referencias del primer y último viaje de la unidad.

Ilustración 24 Reportes por unidad



Descripción: Permite seleccionar una unidad y un rango de fechas para analizar su desempeño. Presenta las unidades disponibles, identifica al chofer correspondiente y permite acceder al reporte detallado. También incorpora indicadores de viajes y puntualidad, junto con la opción de generar o descargar el reporte en formato PDF.

Diseño de interfaces de la aplicación móvil del pasajero

Ilustración 25 Logo de la aplicación



Diseño general de las interfaces

Colores

El color principal de la interfaz es el naranja (#F97316), utilizado en botones, íconos, indicadores, elementos seleccionados, marcadores y componentes de navegación. El blanco se emplea como fondo principal para mantener claridad y legibilidad. Como colores complementarios se conservan el verde (#15803d) para estados activos o disponibles, el amarillo o ámbar (#d97706) para advertencias o pendientes, el rojo (#dc2626) para errores, averías o acciones críticas, el morado para diferenciación de líneas específicas y el gris (#94a3b8) para información secundaria y metadatos. En las vistas cartográficas se mantiene el fondo estándar de OpenStreetMap para favorecer el contraste con los marcadores y recorridos.

Tipografía

Se utiliza una tipografía sans-serif de alta legibilidad en las interfaces móviles y web. Los títulos y nombres de secciones emplean un tamaño de 1.5rem con peso 800; los textos informativos y etiquetas principales utilizan tamaños entre 0.82rem y 0.875rem con peso 600 o 700; y las etiquetas auxiliares, fechas y metadatos se presentan en 0.68rem a 0.75rem con peso regular 400 o 500. La jerarquía tipográfica permite identificar con rapidez la línea seleccionada, la parada, el horario, el estado del servicio y las alertas activas.

Componentes

Las interfaces integran barra de navegación inferior, tarjetas de información, botones, pestañas, chips de estado, indicadores de buses en ruta, listas de líneas y paradas, controles de selección, alertas, marcadores cartográficos y paneles inferiores asociados al mapa. Los controles principales utilizan naranja para señalar acciones activas o elementos seleccionados, mientras que los componentes secundarios emplean blanco, gris claro y colores de estado.

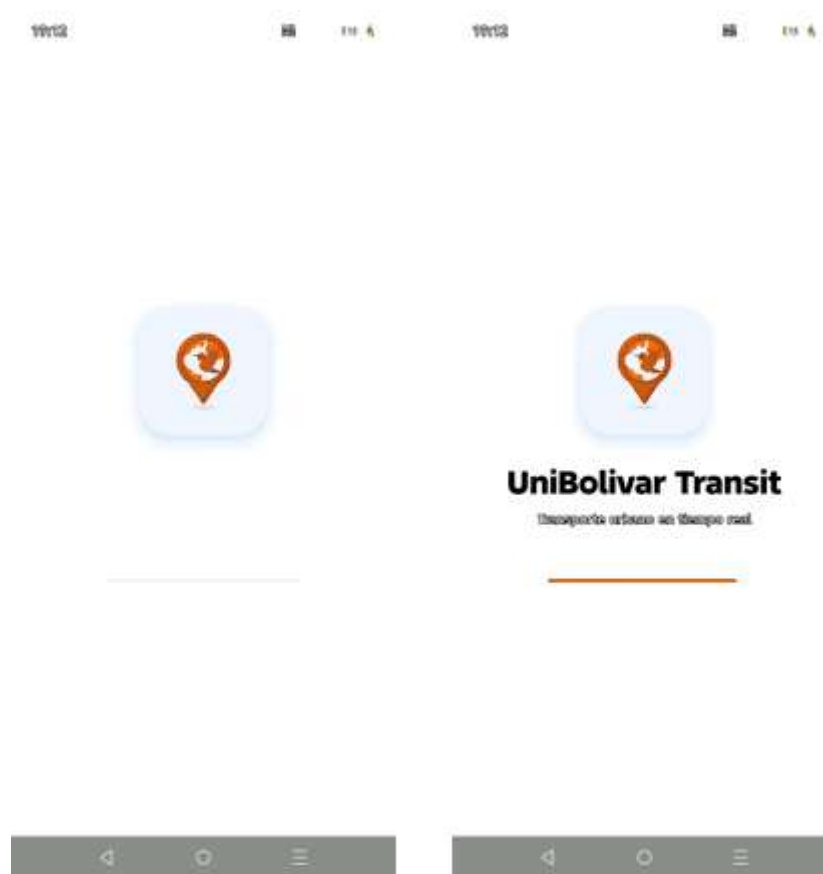
Distribución

La estructura general se organiza de forma vertical y adaptada a pantallas móviles con resolución mínima de Android 10. Las funciones de navegación se mantienen en la zona inferior mediante una barra de navegación fija, mientras que la información principal se presenta en tarjetas o listas dentro del área central. En las pantallas con mapa, este ocupa la mayor parte del espacio disponible y los controles se sitúan en zonas accesibles sin ocultar el recorrido. La información complementaria aparece en paneles inferiores o tarjetas superpuestas.

Características de la localización del mapa

El mapa utiliza OpenStreetMap implementado con Leaflet.js en el panel web y React Native Maps en las aplicaciones móviles. Permite visualizar el contexto geográfico del recorrido, las paradas distribuidas a lo largo de la ruta y la posición de las unidades en servicio actualizada cada 3 segundos mediante WebSocket. Los marcadores resaltados facilitan identificar puntos relevantes del trayecto. La interfaz incluye controles para centrar el mapa en la ciudad de Guaranda y conserva un panel lateral con filtros por línea y sentido. Los marcadores de buses utilizan un símbolo diferente al de las paradas para evitar confusiones en la visualización.

Ilustración 26 Pantalla de bienvenida y carga (Splash)



Descripción: Esta pantalla corresponde al inicio de la aplicación. Presenta el logotipo de UniBolivar Transit y, posteriormente, el nombre del sistema junto con una barra de carga. Su función es identificar visualmente la aplicación mientras se preparan los servicios necesarios antes de mostrar la pantalla principal.

Ilustración 27 Mapa principal (pestaña Rutas)



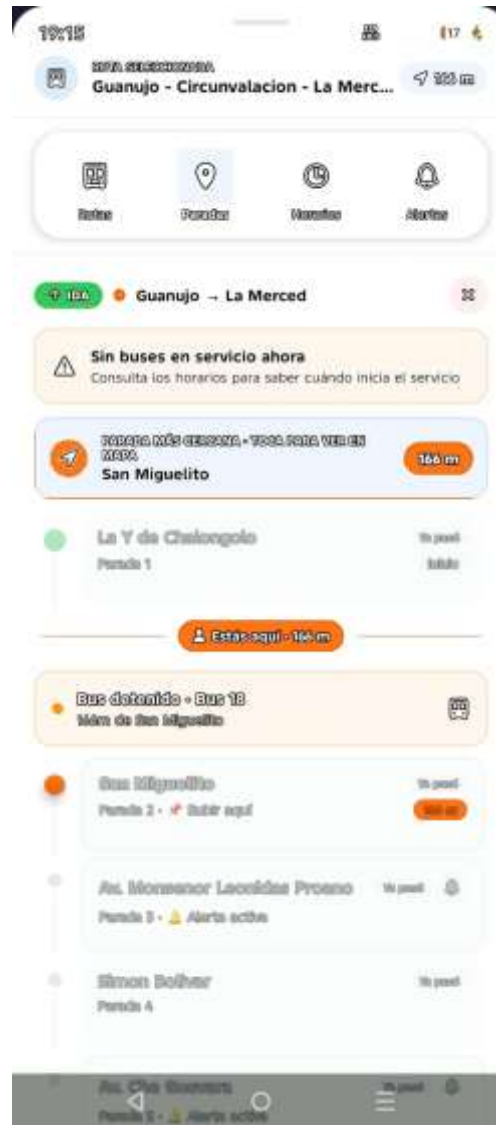
Descripción: La pantalla principal muestra el mapa como elemento central de la experiencia. Permite al pasajero observar el área de operación, reconocer la ubicación de referencia y consultar la existencia de buses en ruta. En la parte inferior se mantiene la navegación hacia Rutas, Paradas, Horarios y Alertas.

Ilustración 28 Lista de líneas / Rutas



Descripción: Esta interfaz presenta las líneas de transporte disponibles mediante tarjetas diferenciadas. Cada tarjeta resume el recorrido y su estado de servicio, permitiendo que el pasajero identifique y seleccione la línea que desea consultar antes de revisar sus paradas, horarios o recorrido.

Ilustración 29 Paradas y tiempos de llegada



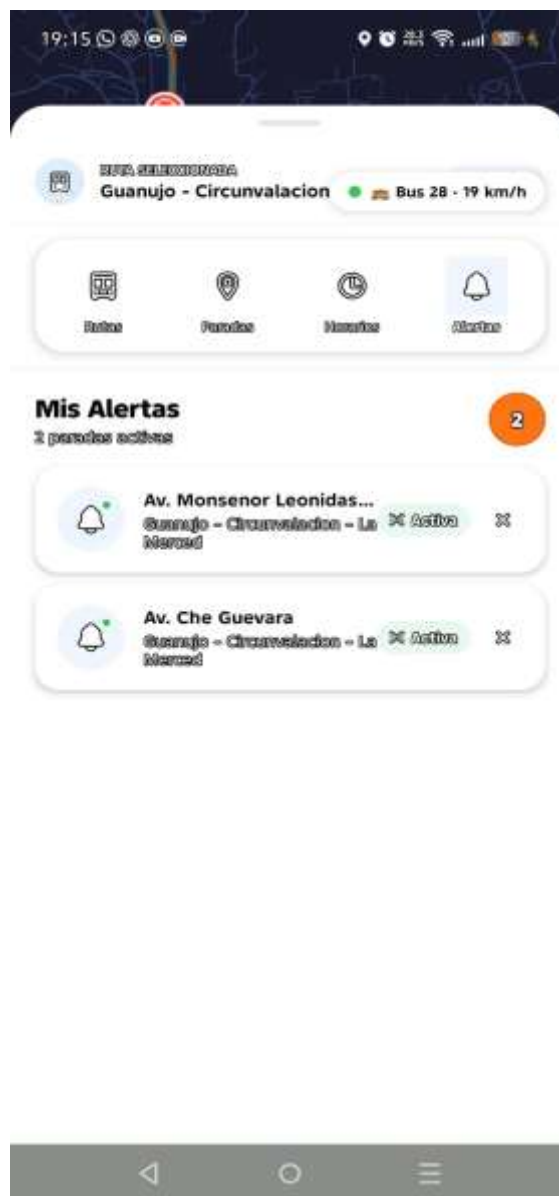
Descripción: La interfaz organiza las paradas de la ruta seleccionada en una secuencia vertical. Muestra la parada más cercana, la distancia de referencia, el estado del servicio y los tiempos disponibles para los próximos buses. También permite reconocer visualmente la posición del pasajero dentro de la secuencia de paradas.

Ilustración 30 Horarios por línea sentido de Ida y Vuelta



Descripción: Esta pantalla permite consultar los horarios del servicio mediante la selección de línea, sentido del recorrido y tipo de día. La información se presenta de forma ordenada para que el pasajero pueda revisar las franjas horarias y planificar su viaje antes de dirigirse a la parada.

Ilustración 31 Alertas activas



Descripción: La sección de alertas reúne las paradas para las cuales el pasajero mantiene una notificación activa. Cada tarjeta identifica la parada y el recorrido asociado, permitiendo revisar las suscripciones vigentes y gestionar las alertas desde una sola pantalla.

Ilustración 32 Alertas activas con contexto en el mapa



Descripción: Esta interfaz combina la información de alertas con el mapa del recorrido. Permite relacionar visualmente la parada seleccionada con la posición de los buses y las demás paradas de la ruta, manteniendo las alertas activas visibles en el panel inferior.

Ilustración 33 Mapa completo del recorrido



Descripción: La pantalla muestra una vista ampliada del trayecto completo de la ruta seleccionada. Los marcadores permiten reconocer la distribución de las paradas a lo largo del recorrido, mientras que el panel inferior mantiene visible la identificación de la ruta y facilita continuar con las demás funciones de la aplicación.

DISEÑO DE INTERFACES GRÁFICAS

Aplicación móvil del chofer - UniBolívar Transit

Ilustración 34 Logo de la aplicación



Colores

La identidad visual utiliza el naranja (#F97316) como color principal para botones, elementos seleccionados, indicadores de navegación y acentos de marca. El blanco se emplea como fondo predominante para mejorar la claridad y la lectura. Se conserva el azul oscuro o gris carbón (#1e293b) para textos y datos operativos, el verde (#15803d) para estados positivos como "En línea" o "En servicio", el rojo (#dc2626) para averías y acciones críticas, y el gris claro (#f1f5f9) para fondos secundarios, separadores y campos inactivos.

Tipografía

Para mantener una apariencia moderna y de fácil lectura se utiliza una tipografía sans-serif de alta legibilidad. Los títulos y nombres principales se presentan en tamaño 1.5rem con peso 800; los subtítulos y etiquetas operativas utilizan tamaños entre 0.82rem y 0.875rem con peso 600 o 700; y los textos explicativos, horarios, rutas y datos secundarios emplean tamaños entre 0.68rem y 0.75rem con peso regular 400 o 500. Los valores numéricos relevantes como tiempo de servicio, distancia o velocidad se destacan con mayor tamaño y contraste para facilitar su lectura durante la conducción.

Componentes

La interfaz integra logotipo institucional, encabezado con datos del chofer y número de unidad, indicadores de estado de conexión, tarjetas para líneas y sentidos de recorrido, botones de inicio y finalización del servicio, cronómetro de servicio activo, tarjetas de métricas, campos de autenticación, selectores de estado de la unidad, campo de observaciones, botones de confirmación, filtros temporales e historial de recorridos. Los componentes se diseñan con bordes redondeados de 8px, espacios amplios y jerarquía visual clara para reducir distracciones durante la operación.

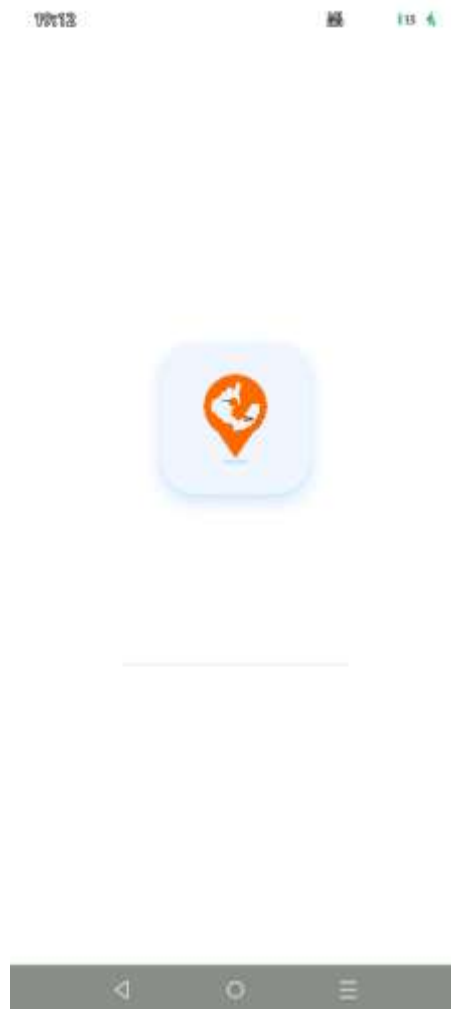
Distribución

La aplicación sigue una estructura móvil vertical compatible con Android 10 o superior. La parte superior concentra la identidad del sistema, el estado del chofer y los accesos principales. El cuerpo central contiene la información operativa, listas de rutas, métricas o historial. Las acciones críticas se ubican en zonas de fácil acceso, especialmente en la parte inferior de la pantalla. Las tarjetas se organizan de manera apilada y con suficiente separación para facilitar la lectura rápida durante la jornada de trabajo.

Características de la localización y del mapa

La aplicación móvil del chofer no presenta una pantalla cartográfica propia. La localización se implementa como una función operativa GPS en segundo plano mediante Expo Task Manager, que obtiene la posición del dispositivo cada 3 segundos y la transmite al servidor mediante el endpoint `/api/chofer/ubicacion` mientras el viaje se encuentra activo. La ruta y el sentido seleccionados por el chofer al iniciar el servicio determinan el contexto del recorrido transmitido. La interfaz informa claramente el estado de conexión mediante un indicador visible, gestiona pérdidas temporales de Internet y reanuda el envío de ubicación automáticamente sin requerir interacción del conductor. La representación cartográfica de esta información se visualiza en la aplicación del pasajero y en el panel administrativo web, mientras que la aplicación del chofer prioriza la conducción y el registro del servicio.

Ilustración 35 Pantalla de bienvenida y carga (splash)



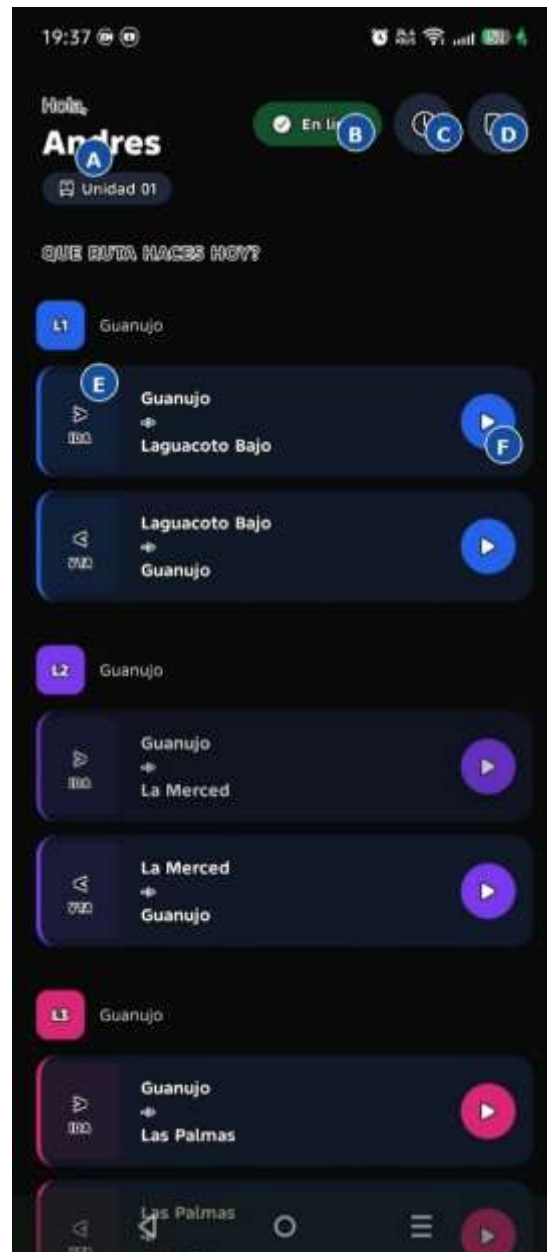
Descripción: La pantalla de bienvenida se muestra al iniciar la aplicación, presenta el identificador visual de UniBolívar Transit mientras el sistema realiza la carga inicial de los recursos necesarios. La composición es minimalista y utiliza fondo blanco con el logotipo en naranja, evitando elementos que puedan distraer al usuario durante el proceso de arranque.

Ilustración 36 Inicio de sesión



Descripción: Esta interfaz permite al chofer autenticarse mediante el correo electrónico y la contraseña asignados por la cooperativa. Los campos de acceso se presentan en una tarjeta central de lectura clara y el botón principal “Ingresar” utiliza el color naranja para indicar la acción primaria. La pantalla restringe el acceso al personal autorizado y dirige al usuario a la selección de ruta una vez validadas las credenciales.

Ilustración 37 Selección de ruta y sentido



Descripción: La pantalla permite al conductor revisar sus datos, la unidad asignada y el estado de conexión antes de iniciar el recorrido. Las líneas disponibles se organizan en tarjetas y cada recorrido ofrece las opciones de ida y vuelta. El chofer selecciona la ruta correspondiente mediante el botón de acción ubicado a la derecha de cada tarjeta; de esta manera, el sistema registra el contexto del viaje antes de activar el servicio y la localización GPS.

Ilustración 38 Pantalla «En servicio»



Descripción: Esta pantalla se utiliza durante un recorrido activo. En la parte superior se muestra el estado “En servicio”, la conexión y la ruta seleccionada. El cronómetro central permite visualizar el tiempo transcurrido, mientras que las tarjetas inferiores presentan métricas operativas como distancia recorrida y velocidad máxima. El botón “Finalizar turno” se conserva en rojo por tratarse de una acción crítica que cierra el servicio y detiene el registro del recorrido.

Ilustración 39 Estado de la unidad



Descripción: La interfaz permite al chofer reportar el estado operativo del bus. Se presentan las opciones “En línea”, “En mantenimiento” y “Avería”, cada una asociada con una condición de servicio. El color verde identifica una condición normal, mientras que el rojo se reserva para la avería. También se incorpora un campo de observación para describir la novedad y un botón naranja de confirmación que registra el cambio para que pueda ser consultado por la administración.

Ilustración 40 Mi historial



Descripción: La pantalla de historial permite consultar los recorridos realizados durante un período determinado. En la parte superior se muestra un resumen mensual con el total de viajes, recorridos completados y viajes en curso. Debajo se incluyen filtros por fecha y una lista de tarjetas que indican línea, sentido, horario, duración y estado de cada recorrido. El uso del naranja unifica la selección y los elementos activos, mientras que el verde identifica los viajes completados.

4.4 SPRINT 3: DESARROLLO DEL SISTEMA

4.4.1 Objetivo del Sprint 3

Desarrollar e integrar los módulos del sistema UniBolívar Transit, incluyendo la API REST, las aplicaciones móviles para pasajeros y choferes, el panel administrativo web y la configuración del servidor de producción.

4.4.2 Actividades Principales

Desarrollo de la API REST

- Desarrollo de la API REST — autenticación de choferes y administrador (herramienta: Visual Studio Code)
- Desarrollo de la API REST — geolocalización y WebSocket en tiempo real (herramienta: Visual Studio Code, Soketi)
- Desarrollo de la API REST — gestión de líneas, rutas y paradas (herramienta: Visual Studio Code)
- Desarrollo de la API REST — notificaciones push con FCM (herramienta: Firebase Console)
- Desarrollo de la API REST — registro de viajes y marcaciones (herramienta: Visual Studio Code)
- Desarrollo de la API REST — reportes de puntualidad (herramienta: Visual Studio Code)
- Desarrollo de la API REST — despacho y permisos de choferes (herramienta: Visual Studio Code)

Desarrollo del panel administrativo web

- Desarrollo del panel administrativo web — dashboard y mapa (herramienta: Visual Studio Code, Next.js)
- Desarrollo del panel administrativo web — gestión de usuarios, líneas, rutas, paradas y horarios (herramienta: Visual Studio Code)
- Desarrollo del panel administrativo web — despacho y reportes PDF (herramienta: Visual Studio Code, jsPDF)

Desarrollo de las aplicaciones móviles

- Desarrollo de la aplicación móvil para pasajeros — mapa, rutas, tiempo estimado de llegada y alertas (herramienta: Visual Studio Code, Expo Go)
- Desarrollo de la aplicación móvil para choferes — GPS en segundo plano e historial de viajes (herramienta: Visual Studio Code, Expo Go)

Integración de servicios

- Integración WebSocket con Soketi (herramienta: Docker)
- Integración de notificaciones push con Firebase Cloud Messaging (herramienta: Firebase Console)

Configuración del entorno de producción

- Configuración del servidor de producción en DigitalOcean con Nginx y Docker (herramienta: SSH, Terminal)
- Configuración de la base de datos PostgreSQL y migraciones (herramienta: DBeaver, Terminal)
- Registro de dominio y configuración SSL con Let's Encrypt (herramienta: Certbot)

4.4.3 Sprint Backlog

Tabla 28. Sprint Backlog del Sprint 3.

Id	Tarea	Estimación	Responsables	Estado
T-13	Desarrollo de la API REST — autenticación de choferes y administrador	6h	Erick Alarcón, Juan Pasto	Completado
T-14	Desarrollo de la API REST — geolocalización y WebSocket en tiempo real	10h	Erick Alarcón, Juan Pasto	Completado
T-15	Desarrollo de la API REST — gestión de líneas, rutas y paradas	8h	Erick Alarcón, Juan Pasto	Completado
T-16	Desarrollo de la API REST — notificaciones push con FCM	10h	Erick Alarcón, Juan Pasto	Completado
T-17	Desarrollo de la API REST — registro de viajes y marcaciones	10h	Erick Alarcón, Juan Pasto	Completado

T-18	Desarrollo de la API REST — reportes de puntualidad	12h	Erick Alarcón, Juan Pasto	Completado
T-19	Desarrollo de la API REST — despacho y permisos de choferes	10h	Erick Alarcón, Juan Pasto	Completado
T-20	Desarrollo del panel administrativo web — dashboard y mapa	20h	Erick Alarcón, Juan Pasto	Completado
T-21	Desarrollo del panel administrativo web — gestión usuarios, líneas, rutas, paradas, horarios	18h	Erick Alarcón, Juan Pasto	Completado
T-22	Desarrollo del panel administrativo web — despacho, reportes PDF	20h	Erick Alarcón, Juan Pasto	Completado
T-23	Desarrollo app móvil pasajero	6h	Erick Alarcón, Juan Pasto	Completado

	— mapa, rutas, ETA, alertas			
T-24	Desarrollo app móvil chofer — GPS background, historial viajes	4h	Erick Alarcón, Juan Pasto	Completado
T-25	Integración WebSocket Soketi	8h	Erick Alarcón, Juan Pasto	Completado
T-26	Integración notificaciones push FCM	3h	Erick Alarcón, Juan Pasto	Completado
T-27	Configuración servidor DigitalOcean — Nginx, Docker, SSL	6h	Erick Alarcón, Juan Pasto	Completado
T-28	Configuración base de datos PostgreSQL y migraciones	2h	Erick Alarcón, Juan Pasto	Completado
T-29	Registro de dominio y configuración SSL con Let's Encrypt	1h	Erick Alarcón, Juan Pasto	Completado

4.4.4 Resultados del Sprint 3

Para el cumplimiento del objetivo del Sprint 3, se desarrollaron e integraron todos los módulos del sistema UniBolívar Transit. A continuación, se presenta cada módulo implementado con su respectivo fragmento de código y descripción funcional, siguiendo el orden establecido en las actividades del sprint.

Ilustración 41 T-13: API REST — Autenticación de choferes y administrador

```
// Autenticación del administrador
public function loginAdmin(Request $request)
{
    // Validar que el correo y contraseña estén presentes
    $credentials = $request->validate([
        'email' => 'required|email',
        'password' => 'required|string',
    ]);

    // Intentar generar el token del usuario administrador
    if (($token = auth('api')->attempt($credentials))) {
        return response()->json(['message' => 'Credenciales incorrectas'], 400);
    }

    $user = auth('api')->user();

    // Verificar que el usuario tenga rol de administrador
    if ($user->role != 'admin') {
        auth('api')->logout();
        return response()->json(['message' => 'No tienes permisos de administrador'], 400);
    }

    // Retornar el token de acceso
    return response()->json([
        'token' => $token,
        'token_type' => 'bearer',
        'expires_in' => auth('api')->factory()->getTTL() * 60,
        'user' => $user,
    ]);
}

// Autenticación del chofer
public function loginChofer(Request $request)
{
    // Validar que el correo y contraseña estén presentes
    $credentials = $request->validate([
        'email' => 'required|email',
        'password' => 'required|string',
    ]);

    // Intentar generar el token del usuario chofer
    if (($token = auth('api')->attempt($credentials))) {
        return response()->json(['message' => 'Credenciales incorrectas'], 400);
    }

    $user = auth('api')->user();

    // Verificar que el usuario tenga rol de chofer
    if ($user->role != 'chofer') {
        auth('api')->logout();
        return response()->json(['message' => 'No tienes permisos de chofer'], 400);
    }

    // Retornar el token de acceso
    return response()->json([
        'token' => $token,
        'token_type' => 'bearer',
        'expires_in' => auth('api')->factory()->getTTL() * 60,
        'user' => $user,
    ]);
}
```

El módulo de autenticación implementa un sistema de inicio de sesión basado en tokens mediante Laravel Sanctum. El sistema valida las credenciales del chofer o administrador y genera un token de acceso que se almacena localmente en el dispositivo para mantener la sesión activa de forma segura.

Ilustración 42 T-14: API REST — Geolocalización y WebSocket en tiempo real

```
// Recibe la ubicación GPS del chofer y transmite en tiempo real
public function updateFromChofer(Request $request)
{
    // Validar los datos recibidos del dispositivo móvil
    $data = $request->validate([
        'ruta_id' => 'required|string|exists:rutas,id',
        'latitud' => 'required|numeric',
        'longitud' => 'required|numeric',
        'velocidad' => 'nullable|numeric',
    ]);

    $user = auth('api')->user();

    // Verificar que el usuario sea un chofer autorizado
    if ($user->role != 'chofer') {
        return response()->json(['message' => 'No autorizado'], 400);
    }

    // Actualizar o crear el registro de ubicación en la base de datos
    $ubicacion = updateOrCreate(
        ['chofer_id' => $user->id],
        [
            'ruta_id' => $data['ruta_id'],
            'latitud' => $data['latitud'],
            'longitud' => $data['longitud'],
            'velocidad' => $data['velocidad'] ?? 0,
        ]
    );

    // Transmitir la ubicación en tiempo real mediante websocket
    broadcast(new $ubicacionActualizada(
        rutaId: $data['ruta_id'],
        latitud: (float) $data['latitud'],
        longitud: (float) $data['longitud'],
        velocidad: (float) ($data['velocidad'] ?? 0),
        choferId: $user->id,
        chofer: $user->name,
        velocidad: $user->numero_licencia ?? null,
    ));

    // Verificar si hay pasajeros a bordo en paradas cercanas
    $this->verificarParadasCercanas(
        $data['ruta_id'],
        $data['latitud'],
        $data['longitud']
    );

    return response()->json(['success' => true, 'message' => 'Ubicación actualizada']);
}
```

El endpoint recibe las coordenadas GPS del chofer cada tres segundos y actualiza la posición en la base de datos. Simultáneamente transmite el evento mediante WebSocket a través de Sockets, actualizando en tiempo real la posición del bus en el mapa del pasajero y del panel administrativo sin necesidad de recargar la página.

Ilustración 43 T-15: API REST — Gestión de líneas, rutas y paradas

```
// Retorna todas las líneas con sus rutas y paradas para la app del pasajero
public function appRoutes()
{
    // Colores asignados a cada línea de transporte
    $colors = [
        'L1' => '■ #FF3B30',
        'L2' => '■ #007AFF',
        'L3' => '■ #34C759',
        'L4' => '■ #FF9500',
        'L5' => '■ #AF52DE',
        'L6' => '■ #FF2D55',
    ];

    // Obtener todas las líneas con sus rutas, paradas y horarios
    $lineas = Linea::with(['rutas.paradas', 'rutas.horarios'])->get();

    $busRoutes = [];
    foreach ($lineas as $linea) {
        $color = $colors[$linea->id] ?? '■ #8E8E93';

        foreach ($linea->rutas as $ruta) {
            // Ordenar las paradas según su orden en la ruta
            $paradas = $ruta->paradas->sortBy('orden')->values();

            // Mapear cada parada con sus coordenadas GPS reales
            $stops = $paradas->map(function ($parada, $index) use ($ruta) {
                $lat = $parada->latitud ? (float) $parada->latitud : null;
                $lng = $parada->longitud ? (float) $parada->longitud : null;
                return [
                    'id' => (string) $parada->id,
                    'name' => $parada->nombre,
                    'coordinate' => $lat && $lng ? [
                        'latitude' => $lat,
                        'longitude' => $lng,
                    ] : null,
                    'order' => $parada->orden,
                    'hasGps' => $lat && $lng,
                ];
            });
            $busRoutes[] = [
                'linea' => $linea,
                'ruta' => $ruta,
                'paradas' => $stops,
            ];
        }
    }
}
```

El módulo permite al administrador registrar y mantener actualizadas las 6 líneas de transporte con sus rutas de ida y vuelta. Cada ruta contiene las paradas ordenadas con sus coordenadas GPS reales de la ciudad de Guaranda, las cuales se visualizan en el mapa de la aplicación del pasajero.

Ilustración 44 T-16: API REST — Notificaciones push con FCM

```
// Envía una notificación push al dispositivo del pasajero mediante FCM
public function sendNotificationWithCollapse(
    string $fcmToken,
    string $title,
    string $body,
    ?string $collapseKey = null,
    array $data = []
): bool {
    try {
        // Obtener el token de acceso de Firebase
        $accessToken = $this->getAccessToken();

        // Construir el mensaje de notificación
        $message = [
            'token' => $fcmToken,
            'notification' => [
                'title' => $title,
                'body' => $body,
            ],
            // Configuración para dispositivos Android
            'android' => [
                'priority' => 'high',
                'collapse_key' => $collapseKey ?? 'gobolivar_default',
                'notification' => [
                    'sound' => 'default',
                    'channel_id' => 'gobolivar_notifications',
                    'tag' => $collapseKey ?? 'gobolivar_default',
                ],
            ],
        ];

        // Agregar datos adicionales si existen
        if (!empty($data)) {
            $message['data'] = array_map('strval', $data);
        }

        // Enviar la notificación a la API de Firebase
        $response = Http::withHeaders([
            'Authorization' => 'Bearer ' . $accessToken,
            'Content-Type' => 'application/json',
        ])->post("https://fcm.googleapis.com/v1/projects/{$this->projectId}/messages:send", [
            'message' => $message,
        ]);

        return $response->successful();
    } catch (\Exception $e) {
        \Log::error('Firebase error: ' . $e->getMessage());
        return false;
    }
}
```

El módulo implementa cuatro niveles de alerta progresiva para el pasajero: bus en camino, bus llegando a menos de 300 metros, bus en la parada y bus que ya pasó la parada. Las notificaciones se envían mediante Firebase Cloud Messaging y se gestionan localmente con Notiffee en el dispositivo del pasajero.

Ilustración 45 T-17: API REST — Registro de viajes y marcaciones

```
C:\Users\> Erick\Downloads\> Untitled-1.php
1
2 {
3     // Finalizar cualquier viaje en curso antes de iniciar uno nuevo
4     Viaje::where('user_id', $user->id)
5     ->where('estado', 'en_curso')
6     ->update([
7         'estado' => 'finalizado',
8         'hora_llegada' => Carbon::now('America/Guayaquil'),
9     ]);
10
11 // Registrar el nuevo viaje con hora de salida actual
12 $viaje = Viaje::create([
13     'user_id' => $user->id,
14     'ruta_id' => $data['ruta_id'],
15     'hora_salida' => Carbon::now('America/Guayaquil'),
16     'estado' => 'en_curso',
17     'lat_inicio' => $data['lat_inicio'] ?? null,
18     'lng_inicio' => $data['lng_inicio'] ?? null,
19 ]);
20
21 return response()->json([
22     'success' => true,
23     'viaje' => $viaje,
24     'message' => 'Servicio iniciado',
25 ]);
26 }
27
28 // Finaliza el servicio del chofer y registra la hora de llegada
29 public function finalizar(Request $request)
30 {
31     $data = $request->validate([
32         'distancia_km' => 'nullable|numeric',
33         'velocidad_max' => 'nullable|integer',
34         'lat_fin' => 'nullable|numeric',
35         'lng_fin' => 'nullable|numeric',
36     ]);
37
38     $user = auth('api')->user();
39
40 // Buscar el viaje en curso del chofer
41 $viaje = Viaje::where('user_id', $user->id)
42     ->where('estado', 'en_curso')
43     ->latest()->first();
44
45 if (!$viaje) {
46     return response()->json([
47         'success' => false,
48         'message' => 'No hay servicio activo',
49     ], 404);
50 }
51
52 // Registrar hora de llegada y datos finales del viaje
53 $viaje->update([
54     'hora_llegada' => Carbon::now('America/Guayaquil'),
55     'distancia_km' => $data['distancia_km'] ?? 0,
56     'velocidad_max' => $data['velocidad_max'] ?? 0,
57     'estado' => 'finalizado',
58     'lat_fin' => $data['lat_fin'] ?? null,
59     'lng_fin' => $data['lng_fin'] ?? null,
60 ]);
61
62 return response()->json([
63     'success' => true,
64     'viaje' => $viaje,
65     'message' => 'Servicio finalizado',
66 ]);
67 }
```

El módulo controla el inicio y fin de cada recorrido del chofer. Al iniciar el servicio se registra la hora de salida y la ruta seleccionada, y al finalizar se registra la hora de llegada. El sistema calcula automáticamente la duración del recorrido y determina si el viaje fue completado a tiempo o con atraso según la tolerancia configurada por línea.

Ilustración 46 T-18: API REST — Reportes de puntualidad

```
// Inicia el servicio del chofer y registra el viaje
public function iniciar(Request $request)
{
    // Validar la ruta seleccionada y coordenadas de inicio
    $data = $request->validate([
        'ruta_id' => 'required|string|exists:rutas,id',
        'lat_inicio' => 'nullable|numeric',
        'lng_inicio' => 'nullable|numeric',
    ]);

    $user = auth('api')->user();

    // Finalizar cualquier viaje en curso antes de iniciar uno nuevo
    $viaje = $viaje::where('user_id', $user->id)
        ->where('estado', 'en_curso')
        ->update([
            'estado' => 'finalizado',
            'hora_llegada' => Carbon::now('America/Guayaquil'),
        ]);

    // Registrar el nuevo viaje con hora de salida actual
    $viaje = $viaje::create([
        'user_id' => $user->id,
        'ruta_id' => $data['ruta_id'],
        'hora_salida' => Carbon::now('America/Guayaquil'),
        'estado' => 'en_curso',
        'lat_inicio' => $data['lat_inicio'] ?? null,
        'lng_inicio' => $data['lng_inicio'] ?? null,
    ]);

    return response()->json([
        'success' => true,
        'viaje' => $viaje,
        'message' => 'Servicio iniciado',
    ]);
}

// Finaliza el servicio del chofer y registra la hora de llegada
public function finalizar(Request $request)
{
    $data = $request->validate([
        'distancia_km' => 'nullable|numeric',
        'velocidad_max' => 'nullable|integer',
        'lat_fin' => 'nullable|numeric',
        'lng_fin' => 'nullable|numeric',
    ]);

    $user = auth('api')->user();

    // Buscar el viaje en curso del chofer
    $viaje = $viaje::where('user_id', $user->id)
        ->where('estado', 'en_curso')
        ->latest()->first();

    if (!$viaje) {
        return response()->json([
            'success' => false,
            'message' => 'No hay servicio activo',
        ], 404);
    }

    // Registrar hora de llegada y datos finales del viaje
    $viaje->update([
        'hora_llegada' => Carbon::now('America/Guayaquil'),
        'distancia_km' => $data['distancia_km'] ?? 0,
        'velocidad_max' => $data['velocidad_max'] ?? 0,
        'estado' => 'finalizado',
        'lat_fin' => $data['lat_fin'] ?? null,
        'lng_fin' => $data['lng_fin'] ?? null,
    ]);

    return response()->json([
        'success' => true,
        'viaje' => $viaje,
        'message' => 'Servicio finalizado',
    ]);
}
```

El módulo calcula el desempeño de cada chofer por fecha, determinando el número de viajes completados a tiempo y con atraso según el tiempo límite y la tolerancia configurada para cada línea. Los resultados se presentan en el panel administrativo y pueden descargarse en formato PDF.

Ilustración 47 T-19: API REST — Despacho y permisos de choferes

```
// Guarda las asignaciones y ausencias del despacho diario
public function guardar(Request $request)
{
    $fecha = $request->input('fecha');

    // Eliminar asignaciones anteriores de esa fecha
    DespachoAsignacion::where('fecha', $fecha)->delete();

    // Registrar las nuevas asignaciones de unidades por línea
    foreach ($request->input('asignaciones', []) as $a) {
        DespachoAsignacion::create([
            'fecha' => $fecha,
            'unidad' => $a['unidad'],
            'linea_id' => $a['linea_id'],
        ]);
    }

    // Eliminar ausencias anteriores y registrar las nuevas
    DespachoAusencia::where('fecha', $fecha)->delete();
    foreach ($request->input('ausencias', []) as $a) {
        DespachoAusencia::create([
            'fecha' => $fecha,
            'unidad' => $a['unidad'],
            'motivo' => $a['motivo'] ?? 'descanso',
        ]);
    }

    return response()->json(['success' => true]);
}

// Registra un permiso o descanso para una unidad por rango de fechas
public function crearPermiso(Request $request)
{
    // Validar unidad y rango de fechas
    $request->validate([
        'unidad' => 'required',
        'desde' => 'required|date',
        'hasta' => 'required|date|after_or_equal:desde',
    ]);

    // Crear el permiso en la base de datos
    DespachoPermiso::create(
        $request->only('unidad', 'desde', 'hasta', 'motivo')
    );

    // Retornar la lista actualizada de permisos
    $permisos = DespachoPermiso::orderBy('desde')->get()
        ->map(fn($p) => [
            'id' => $p->id,
            'unidad' => $p->unidad,
            'desde' => $p->desde->format('Y-m-d'),
            'hasta' => $p->hasta->format('Y-m-d'),
            'motivo' => $p->motivo,
        ]);

    return response()->json(['success' => true, 'permisos' => $permisos]);
}
```

El módulo permite al administrador asignar las unidades de transporte a cada línea mediante un calendario semanal y registrar permisos o descansos de los choferes. Cuando un chofer tiene un permiso activo, el sistema bloquea automáticamente el inicio de servicio en la aplicación móvil durante el período registrado.

Ilustración 48 T-20: Panel administrativo web — Dashboard y mapa

```
// Crea el icono visual del bus en el mapa según su estado
const crearIconoBus = (l: any, label: string,
  velocidad: number, mensaje: string) => {

  // Determinar color según si el bus está en movimiento
  const activo = velocidad >= 2;
  const color = activo ? '#2563eb' : '#94a3b8';
  const dotColor = activo ? '#22c55e' : '#f99e8b';

  // Crear icono SVG personalizado con forma de bus
  return l.divicon({
    className: '',
    html: `
    <div style="display:flex;flex-direction:column;align-items:center;">
    // Mensaje de proximidad a parada
    ${mensaje ? `<div style="background:#0f172a;color:white;
    border-radius:6px;padding:3px 7px;font-size:18px;
    font-weight:700;">${mensaje}</div>` : ''}

    // Circulo con icono de bus
    <div style="width:44px;height:44px;border-radius:50%;
    background:${color};border:3px solid white;">
    <svg width="22" height="22" viewBox="0 0 24 24">
    <rect x="2" y="6" width="20" height="12" rx="3" fill="white"/>
    <rect x="3" y="7" width="18" height="10" rx="2" fill="${color}"/>
    </svg>
    // Indicador de estado activo/inactivo
    <div style="background:${dotColor};border-radius:50%;"></div>
    </div>
    // Etiqueta con número de unidad
    <div style="background:${color};color:white;">${label}</div>
    </div>`,
    iconSize: [44, mensaje ? 80 : 62],
  });
};

// Dibuja las paradas de todas las líneas en el mapa
const dibujarTodasParadas = (routes: any[], filtroLinea: string) => {

  // Limpiar paradas anteriores del mapa
  paradaMarkersRef.current.forEach(m =>
    mapInstanceRef.current.removeLayer(m));
  paradaMarkersRef.current = [];

  // Colores por línea de transporte
  const coloresLinea: Record<string, string> = {
    L1: '#2563eb', L2: '#7c3aed', L3: '#db2777',
    L4: '#d97706', L5: '#859e69', L6: '#8891b2',
  };

  // Dibujar cada parada como punto en el mapa
  routes.forEach((ruta: any) => {
    const lineaId = ruta.id?.split('-')[0] || 'L1';
    if (!filtroLinea || filtroLinea === 'todas' || lineaId === filtroLinea) return;
    const color = coloresLinea[lineaId] || '#2563eb';
    ruta.stops?.forEach((p: any) => {
      if (!p.coordinate) return;
      // Crear marcador de parada con color de línea
      const marker = L.marker(
        [p.coordinate.latitude, p.coordinate.longitud],
        { icon: icono }
      ).addTo(mapInstanceRef.current);
      paradaMarkersRef.current.push(marker);
    });
  });
};
```

El panel administrativo web presenta un dashboard con estadísticas en tiempo real y un mapa de monitoreo de flota. El mapa recibe actualizaciones de posición mediante WebSocket y polling cada dos segundos, mostrando la ubicación, velocidad y parada actual de cada unidad activa en la ciudad de Guaranda.

Ilustración 49 T-21: Panel administrativo web — Gestión de usuarios, líneas, rutas, paradas y horarios

```
// Carga la lista de usuarios desde la API
const loadUsers = async () => {
  const token = auth.getToken();
  if (!token) return;
  setLoading(true);

  const response = await apiClient.getUsers(token);

  // Filtrar solo choferes y administradores
  if (response.success && response.users) {
    setUsers(response.users.filter(
      (u: User) => u.role === 'admin' || u.role === 'chofer'
    ));
  } else {
    setError(response.message || 'Error al cargar usuarios');
  }
  setLoading(false);
};

// Abre el modal para crear un nuevo usuario
const handleCreate = () => {
  setSelectedUser(null);
  setIsModalOpen(true);
};

// Abre el modal para editar un usuario existente
const handleEdit = (user: User) => {
  setSelectedUser(user);
  setIsModalOpen(true);
};

// Guarda el usuario nuevo o actualiza el existente
const handleSave = async (userData: {
  name: string; email: string;
  password?: string; role: UserRole
}) => {
  const token = auth.getToken();
  if (!token) return;
  setIsSaving(true);

  let response;
  if (selectedUser) {
    // Actualizar usuario existente
    response = await apiClient.updateUser(
      token, selectedUser.id, userData
    );
  } else {
    // Validar contraseña obligatoria para nuevo usuario
    if (!userData.password) {
      showToast({
        message: 'La contraseña es obligatoria',
        type: 'error'
      });
      setIsSaving(false);
      return;
    }
    // Crear nuevo usuario
    response = await apiClient.createUser(token, userData as {
      name: string; email: string;
      password: string; role: UserRole
    });
  }

  setIsSaving(false);

  if (response.success) {
    setIsModalOpen(false);
    loadUsers();
    showToast({
      message: selectedUser
        ? `Usuario "${userData.name}" actualizado correctamente`
        : `Usuario "${userData.name}" creado correctamente`,
      type: 'success',
    });
  } else {
    showToast({
      message: response.message || 'Error al guardar usuario',
      type: 'error'
    });
  }
};
```

El panel incluye módulos completos de gestión para cada componente del sistema. El administrador puede crear, editar y desactivar choferes, registrar líneas con sus rutas de ida y vuelta, agregar paradas con coordenadas GPS y configurar los horarios de frecuencia por día de la semana para cada línea.

Ilustración 50 T-22: Panel administrativo web — Despacho y reportes PDF

```

// Genera la tabla de viajes en el documento PDF
autoTable(doc, {
  startY: 120,
  // Encabezados de la tabla
  head: [['#', 'Ruta', 'Salida', 'Llegada',
    'Duración', 'Estado', 'Mini. atraso']],

  // Datos de cada viaje
  body: viajesRango.map((v: any, i: number) => [
    `${i+1}`,
    n(v.ruta_id?.replace('-', 'IDA', ' IDA')
      .replace('-', 'VUELTA', ' VTA') || ''),
    v.hora_salida?.slice(11,16) || '--',
    v.hora_llegada?.slice(11,16) || 'En ruta',
    v.duracion_minutos
      ? `${Math.round(v.duracion_minutos)} min` : '--',
    v.tiene_atraso ? 'ATRASO'
      : v.estado === 'finalizado' ? 'A tiempo' : 'En curso',
    v.tiene_atraso
      ? `${Math.round(v.minutos_atraso || 0)} min` : '--',
  ]),

  // Estilo del encabezado en color naranja institucional
  headStyles: {
    fillColor: [249,115,22],
    fontSize: 8,
    textColor: [255,255,255],
    fontStyle: 'bold'
  },
  bodyStyles: { fontSize: 8 },
  alternateRowStyles: { fillColor: [255,250,245] },

  // Colorear en rojo los viajes con atraso
  didParseCell: (data: any) => {
    if (data.section === 'body') {
      const v = viajesRango[data.row.index];
      if (v?.tiene_atraso) {
        // Filas con atraso en rojo
        data.cell.styles.textColor = [220, 38, 38];
        data.cell.styles.fontStyle = 'bold';
        data.cell.styles.fillColor = [255, 245, 245];
      } else if (data.column.index === 5
        && v?.estado === 'finalizado') {
        // Filas a tiempo en verde
        data.cell.styles.textColor = [22, 163, 74];
        data.cell.styles.fontStyle = 'bold';
      }
    }
  },
});

// Agregar pie de página en todas las páginas del PDF
const pageCount = (doc as any).internal.getNumberOfPages();
for (let i = 1; i <= pageCount; i++) {
  doc.setPage(i);
  // Barra naranja al pie de cada página
  doc.setFill(249, 115, 22);
  doc.rect(0, 285, 210, 12, 'f');
  doc.setTextColor(255, 255, 255);
  doc.setFontSize(7);
  doc.text(
    'UniBolívar Transit - Universidad Estatal de Bolívar - Guaranda',
    105, 291, { align: 'center' }
  );
  doc.text(`Pag. ${i} / ${pageCount}`, 198, 291, { align: 'right' });
}

// Guardar el PDF con nombre descriptivo
doc.save(`Reporte-US${String(reporte.unidad).padStart(2,'0')}-${pdfDesde}-${pdfHasta}.pdf`);

```

El módulo de reportes genera documentos PDF con el detalle de puntualidad de cada chofer por rango de fechas. Los viajes con atraso se resaltan en color rojo e incluyen los minutos de retraso. El documento incluye el logotipo institucional, estadísticas generales y el pie de página con la información de la Universidad Estatal de Bolívar.

Ilustración 51 T-23: Aplicación móvil para pasajeros — Mapa, rutas, ETA y alertas

```
// Inicializa la conexión WebSocket para recibir ubicaciones en tiempo real
const initWebSocket = useCallback(async () => {
  try {
    // Obtener instancia única de Pusher
    const pusher = Pusher.getInstance();
    pusherRef.current = pusher;

    // Configurar conexión con el servidor WebSocket Soketi
    await pusher.init({
      apiKey: 'gobolivar-key',
      cluster: 'mt1',
      wsHost: 'unibolivartransit.com',
      wsPort: 443,
      wssPort: 443,
      forceTLS: true,

      // Manejar cambios de estado de la conexión
      onConnectionStateChange: async (current: string) => {
        if (current === 'CONNECTED') {
          // Conexión exitosa = detener polling
          setWsConectado(true);
          detenerPolling();
        } else if (current === 'DISCONNECTED') {
          // Sin conexión = activar polling como respaldo
          setWsConectado(false);
          iniciarPolling();
          reconectar();
        }
      },
      onError: (message: string) => {
        console.error('WS Error:', message);
        setWsConectado(false);
        reconectar();
      },
    });

    // Suscribirse al canal de ubicaciones de buses
    await pusher.subscribe({
      channelName: 'bus-ubicaciones',
      onEvent: (event: PusherEvent) => {
        if (event.eventName === 'ubicacion.actualizada') {
          // Actualizar posición del bus en el mapa
          const data = typeof event.data === 'string'
            ? JSON.parse(event.data)
            : event.data;
          actualizarUbicacion(data);
        }
      },
    });

    // Establecer la conexión
    await pusher.connect();

  } catch (err) {
    console.error('Error init WS:', err);
    setWsConectado(false);
    // Usar polling como alternativa si WebSocket falla
    iniciarPolling();
  }
}, [actualizarUbicacion, detenerPolling, iniciarPolling, reconectar]);
```

La aplicación móvil del pasajero desarrollada en React Native con Expo permite visualizar en tiempo real la posición de los buses en el mapa, consultar las rutas y horarios de cada línea, ver el tiempo estimado de llegada por parada y activar alertas de proximidad. Las actualizaciones de posición se reciben mediante WebSocket en tiempo real.

Ilustración 52 T-24: Aplicación móvil para choferes — GPS en segundo plano e historial de viajes

```
taskManager.defineTask(LOCATION_TASK, async ({ data, error }: any) => {
  if (error) { console.error('GPS background error:', error); return; }
  if (!data) return;

  const { locations } = data;
  if (!locations.length) return;

  const loc = locations[0];

  // Obtener la ruta activa del chofer desde el almacenamiento local
  const rutaId = await AsyncStorage.getItem('chofer_ruta');
  if (!rutaId) return;

  // Convertir velocidad de m/s a km/h
  const velocidad = loc.coords.speed != null && loc.coords.speed >= 0
    ? Math.round(loc.coords.speed * 3.6)
    : 0;

  // Enviar ubicación al servidor en tiempo real
  await enviarUbicacion(
    loc.coords.latitude,
    loc.coords.longitude,
    velocidad,
    rutaId
  );

  // Registrar velocidad máxima alcanzada
  const velMaxStr = await AsyncStorage.getItem('chofer_velocidad_max');
  const velMax = velMaxStr ? parseFloat(velMaxStr) : 0;
  if (velocidad > velMax) {
    await AsyncStorage.setItem('chofer_velocidad_max', velocidad.toString());
  }
});

// Inicia el servicio GPS en segundo plano
export async function iniciarGPS(rutaId: string): Promise<boolean> {
  try {
    // Solicitar permisos de ubicación en primer y segundo plano
    const permisosOk = await solicitarPermisos();
    if (!permisosOk) return false;

    // Guardar datos de inicio del servicio
    await AsyncStorage.setItem('chofer_ruta', rutaId);
    await AsyncStorage.setItem('chofer_inicio_servicio', Date.now().toString());
    await AsyncStorage.setItem('chofer_velocidad_max', '0');

    // Iniciar actualizaciones de ubicación con notificación persistente
    await Location.startLocationUpdatesAsync(LOCATION_TASK, {
      accuracy: Location.Accuracy.High,
      timeInterval: GPS_INTERVAL_MS, // cada 3 segundos
      distanceInterval: 0,
      // Notificación visible mientras el GPS está activo
      foregroundService: {
        notificationTitle: 'UniBolívar Transit',
        notificationBody: '📍 GPS activo - En servicio',
        notificationColor: '#F97316',
      },
      pausesUpdatesAutomatically: false,
      showsBackgroundLocationIndicator: true,
    });

    return true;
  } catch (error) {
    console.error('Error iniciando GPS:', error);
    return false;
  }
}
```

La aplicación móvil del chofer transmite la ubicación GPS cada tres segundos incluso con la pantalla apagada, mediante un servicio en segundo plano con notificación persistente. El chofer puede iniciar y finalizar su servicio, reportar el estado de la unidad y consultar el historial de viajes con detalle de puntualidad.

Ilustración 53 T-25 y T-26: Integración WebSocket Soketi y notificaciones FCM

```
<?php
namespace App\Events;

use Illuminate\Broadcasting\Channel;
use Illuminate\Contracts\Broadcasting\ShouldBroadcastNow;

// Evento que se transmite en tiempo real cuando el chofer envía su ubicación
class BusUbicacionActualizada implements ShouldBroadcastNow
{
    public function __construct(
        // Datos de la ubicación del bus
        public string $rutaId, // Identificador de la ruta
        public float $latitud, // Coordenada latitud GPS
        public float $longitud, // Coordenada longitud GPS
        public float $velocidad, // Velocidad en km/h
        public ?int $choferId, // ID del chofer
        public ?string $chofer, // Nombre del chofer
        public ?string $numeroUnidad, // Numero de unidad del bus
    ) {}

    // Define el canal websocket donde se transmite el evento
    public function broadcastOn(): array
    {
        return [
            new Channel('bus-ubicaciones'),
        ];
    }

    // Nombre del evento que escuchan los clientes conectados
    public function broadcastAs(): string
    {
        return 'ubicacion.actualizada';
    }
}
```

La integración de Soketi como servidor WebSocket permite la transmisión de ubicaciones en tiempo real sin necesidad de consultas periódicas al servidor. Cada vez que el chofer envía su posición, el evento se transmite instantáneamente a todos los clientes conectados al canal bus-ubicaciones, logrando una latencia de transmisión inferior a un segundo.

Ilustración 54 T-27, T-28 y T-29: Configuración del servidor, base de datos y SSL

```
# Configuración del servidor Nginx para Unibolivar Transit
server {
    server_name unibolivartransit.com www.unibolivartransit.com;

    # Panel Administrativo Next.js - puerto 3000
    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }

    # API REST Laravel - puerto 8000
    location /api {
        proxy_pass http://localhost:8000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Proto $scheme;
    }

    # WebSocket Soketi - puerto 6001
    location /app {
        proxy_pass http://localhost:6001;
        proxy_http_version 1.1;
        # Habilitar protocolo WebSocket
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "Upgrade";
        proxy_set_header Host $host;
        proxy_read_timeout 60s;
    }

    # Certificado SSL gestionado por Let's Encrypt - Certbot
    listen 443 ssl;
    ssl_certificate /etc/letsencrypt/live/unibolivartransit.com/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/unibolivartransit.com/privkey.pem;
    include /etc/letsencrypt/options-ssl-nginx.conf;
}

# Redirigir HTTP a HTTPS automáticamente
server {
    listen 80;
    server_name unibolivartransit.com www.unibolivartransit.com;
    return 301 https://$host$request_uri;
}
```

El sistema fue desplegado en un servidor Ubuntu 22.04 de DigitalOcean. El backend Laravel opera dentro de un contenedor Docker con Nginx como proxy inverso. La base de datos PostgreSQL 16 se configuró en un contenedor independiente. El dominio unibolivartransit.com fue configurado con certificado SSL mediante Let's Encrypt, garantizando comunicaciones seguras mediante HTTPS.

4.5 SPRINT 4: VALIDACIÓN, PRUEBAS E IMPLEMENTACIÓN

4.5.1 Objetivo del Sprint 4

Validar el funcionamiento del sistema UniBolívar Transit mediante pruebas funcionales, de rendimiento, de latencia y de compatibilidad, corregir los errores encontrados e implementar la solución en el entorno de producción.

4.5.2 Actividades Principales

- Ejecución de pruebas funcionales por historia de usuario
- Pruebas de rendimiento y latencia de la API REST
- Pruebas de latencia del WebSocket
- Pruebas de GPS en segundo plano
- Pruebas de seguridad de la aplicación web
- Pruebas de compatibilidad en diferentes versiones de Android (herramienta: dispositivos Android)
- Pruebas de tiempo de respuesta de endpoints (herramienta: Postman)
- Registro de resultados y porcentaje de cumplimiento
- Corrección de errores detectados
- Capacitación al administrador de la cooperativa
- Despliegue final en servidor de producción

4.5.3 Sprint Backlog

Tabla 29. Sprint Backlog del Sprint 4.

Id	Tarea	Estimación	Encargados	Estado
T-30	Planificación de pruebas funcionales, de rendimiento y de seguridad	12h	Erick Alarcón, Juan Pasto	Completado
T-31	Ejecución de pruebas funcionales	8h	Erick Alarcón, Juan Pasto	Completado
T-32	Pruebas de rendimiento con k6	6h	Erick Alarcón, Juan Pasto	Completado
T-33	Pruebas de latencia WebSocket con k6	10h	Erick Alarcón, Juan Pasto	Completado
T-34	Pruebas de GPS en segundo plano	6h	Erick Alarcón, Juan Pasto	Completado
T-35	Pruebas de compatibilidad Android	5h	Erick Alarcón, Juan Pasto	Completado
T-36	Pruebas de seguridad con OWASP ZAP	6h	Erick Alarcón, Juan Pasto	Completado

T-37	Pruebas de tiempo de respuesta con Postman	4h	Erick Alarcón, Juan Pasto	Completado
T-38	Corrección de errores detectados	6h	Erick Alarcón, Juan Pasto	Completado
T-39	Capacitación al administrador de la cooperativa	7h	Erick Alarcón, Juan Pasto	Completado
T-40	Despliegue final en servidor de producción	3h	Erick Alarcón, Juan Pasto	Completado

4.5.4 Ejecución y resultados

Pruebas funcionales

Para la validación del sistema UniBolívar Transit se definieron criterios de aceptación por cada prueba funcional, estableciendo la condición mínima que debe cumplir el sistema para considerar que la funcionalidad evaluada satisface la historia de usuario correspondiente. Cada prueba se trazó directamente desde su historia de usuario hasta su resultado, garantizando la cobertura completa de los requerimientos definidos en el Sprint 1. Las evidencias fotográficas de cada prueba se encuentran detalladas en el Anexo 7.

Tabla 30 Pruebas funcionales del sistema UniBolívar Transit.

#	Prueba	HU	Herramienta	Criterio de aceptación	Resultado	Estado
P-01	Registro automático del pasajero	HU-01	Android	El pasajero accede a la app sin registrarse manualmente	Pasajero registrado automáticamente	Cumple
P-02	Visualización de rutas en mapa	HU-02	Android	Las rutas se muestran en el mapa con paradas ordenadas	Rutas visibles en mapa interactivo	Cumple
P-03	Visualización de buses en tiempo real	HU-03	Android	Los buses se actualizan en el mapa cada 3 segundos	Buses actualizados cada 3 segundos	Cumple
P-04	Cálculo de tiempo estimado de llegada	HU-04	Android	El sistema muestra el ETA en minutos por parada	ETA calculado por secuencia de paradas	Cumple
P-05	Suscripción a notificaciones de parada	HU-05	Android	El pasajero activa alerta en una parada seleccionada	Suscripción activada correctamente	Cumple

P-06	Notificación push cuando el bus llega	HU-06	Android FCM	El pasajero recibe notificación en 4 niveles de proximidad	Notificación recibida en 4 niveles	Cumple
P-07	Consulta de horarios por línea y día	HU-07	Android	Los horarios se muestran agrupados por franja horaria	Horarios agrupados correctamente	Cumple
P-08	Inicio de sesión del chofer	HU-08	Android	El chofer ingresa con credenciales y recibe token JWT	Token JWT generado correctamente	Cumple
P-09	Inicio y fin de servicio del chofer	HU-09	Android	El viaje se registra con hora de salida y llegada	Viaje registrado correctamente	Cumple
P-10	GPS en segundo plano	HU-10	Android	El GPS transmite ubicación con pantalla apagada	Ubicación enviada con pantalla apagada	Cumple
P-11	Historial de viajes del chofer	HU-11	Android	El chofer consulta viajes con detalle de puntualidad	Viajes listados con puntualidad	Cumple

P-12	Gestión de usuarios	HU-12	Chrome	El admin crea, edita y desactiva choferes	Usuarios gestionados correctamente	Cumple
P-13	Gestión de rutas y paradas	HU-13	Chrome	Las paradas se registran con coordenadas GPS	Paradas registradas con GPS	Cumple
P-14	Monitoreo de flota en tiempo real	HU-14	Chrome	Los buses aparecen en el mapa con posición actualizada	Buses visibles en tiempo real	Cumple
P-15	Gestión de despacho semanal	HU-18	Chrome	Las asignaciones se guardan por día y línea	Asignaciones guardadas correctamente	Cumple
P-16	Gestión de permisos y descansos	HU-19	Chrome	El chofer queda bloqueado durante el período registrado	Chofer bloqueado correctamente	Cumple
P-17	Reporte de estado de unidad	HU-20	Android	El estado se reporta en tiempo real al administrador	Estado reportado correctamente	Cumple

P-18	Notificaciónes en tiempo real de flota	HU-21	Chrome	La campana se actualiza sin recargar la página	Campana actualizada sin recargar	Cumple
P-19	Configuración de tolerancia por línea	HU-22	Chrome	La tolerancia se aplica correctamente al calcular atrasos	Tolerancia aplicada correctamente	Cumple

Pruebas de rendimiento

Para la ejecución de las pruebas de rendimiento se utilizó la herramienta k6, simulando diferentes cantidades de choferes enviando ubicación GPS simultáneamente al servidor de producción.

Ilustración 55 P-20 — Rendimiento endpoint GPS con 48 choferes

```

root@ubuntu-3-1vcpu-2gb-nyc1:~# k6 run -e TOKEN=$(cat /tmp/tokens.json | python3 -c "import json,sys; t=json.load(sys.stdin);
WARN[0000] "cli" level configuration override scenarios configuration entirely

Grafana
MKG

execution: local
script: /tmp/test_gps.js
output: -

scenarios: (100.00%) 1 scenario, 48 max VUs, 1m0s max duration (incl. graceful stop):
 * default: 48 looping VUs for 30s (gracefulStop: 30s)

TOTAL RESULTS
checks_total..... 000..... 30.000000/s
checks_succeeded.... 100.00% 000 out of 000
checks_failed..... 0.00% 0 out of 000

/ status 200
/ respuesta ok

HTTP
http_req_duration..... avg=2.71s min=701.21ms med=2.77s max=3.84s p(90)=2.92s p(95)=3.96s
 [ expected_response:true ]... avg=2.71s min=701.21ms med=2.77s max=3.84s p(90)=2.92s p(95)=3.96s
http_req_failed..... 0.00% 0 out of 483
http_reqs..... 483..... 13.023000/s

EXECUTION
iteration_duration..... avg=3.78s min=1.80s med=3.77s max=6.7s p(90)=3.93s p(95)=4.11s
iterations..... 483..... 13.023000/s
vus..... 10..... min=30 max=48
vus_max..... 48..... min=48 max=48

NETWORK
data_received..... 346 kB 10 kB/s
data_sent..... 331 kB 9.9 kB/s

running (0m33.5s), 90/48 VUs, 483 complete and 0 interrupted iterations
default / [=====] 48 VUs 30s
root@ubuntu-3-1vcpu-2gb-nyc1:~# client_loop: send disconnect: Connection reset
PS C:\Users\fricks\

```

La prueba se ejecutó bajo el criterio de aceptación de que el tiempo de respuesta promedio no supere los 5 segundos y el porcentaje de solicitudes fallidas sea inferior al 1%. Los resultados obtenidos 2.71 segundos promedio y 0% de fallos se encuentran dentro de los umbrales definidos, lo que indica que el sistema opera de manera estable bajo la carga máxima real de la cooperativa, equivalente a 48 choferes transmitiendo ubicación simultáneamente.

Tabla 31 Prueba de rendimiento del endpoint GPS (P-20).

Escenario	Herramienta	VUs	Duración	Tiempo promedio	Tiempo Mínimo	Tiempo Máximo	Criterio de aceptación	% Éxito
Endpoint GPS 48 choferes	k6	48	30 s	2.71 s	783 ms	3.84 s	≤ 5 s promedio, 0% fallos	100%

Ilustración 56 Prueba de rendimiento del endpoint de ubicación bajo carga concurrente — P-21

```

k6 run --vus 48 --duration 30s /tmp/test_latencia.js

Grafana
K6

execution: local
  script: /tmp/test_latencia.js
  output: -

scenarios: (100.00%) 1 scenario, 48 max VUs, 1m0s max duration (incl. graceful stop):
  * default: 48 looping VUs for 30s (gracefulStop: 30s)

TOTAL RESULTS
checks_total..... 770      22.429582/s
checks_succeeded... 100.00% 770 out of 770
checks_failed..... 0.00%   0 out of 770

✓ status 200
✓ respuesta ok

HTTP
http_req_duration..... avg=2.95s min=183.97ms med=2.97s max=4.29s p(90)=3.84s p(95)=4.05s
[ expected_response:true ]... avg=2.95s min=183.97ms med=2.97s max=4.29s p(90)=3.84s p(95)=4.05s
http_req_failed..... 0.00% 0 out of 385
http_reqs..... 385      11.224791/s

EXECUTION
iteration_duration..... avg=0.82s min=1.53s med=3.99s max=5.75s p(90)=3.94s p(95)=5.17s
iterations..... 385      11.224791/s
vus..... 8      0.248275/s max=25
vus_max..... 48      0.148275/s max=25

NETWORK
data_received..... 339 kB 9.9 kB/s
data_sent..... 329 kB 9.9 kB/s

running (0s34.3s), 00/48 VUs, 385 complete and 0 interrupted iterations
default / [=====] 48 VUs 30s
root@ubuntu-s-1vcpu-2gb-nyc1:~#

```

La prueba P-21 midió la latencia de respuesta del endpoint HTTP /api/chofer/ubicacion bajo carga concurrente, complementando la prueba P-20 con una segunda ejecución bajo las mismas condiciones. La prueba se ejecutó con k6 simulando 48 choferes enviando ubicación simultáneamente durante 30 segundos, obteniendo una latencia promedio de 2.95 segundos, un

tiempo mínimo de 463 ms y un tiempo máximo de 4.29 segundos, con un porcentaje de éxito del 100% y sin ninguna solicitud fallida. Es importante precisar que este tiempo corresponde únicamente a la duración de la solicitud HTTP (http_req_duration) hasta la confirmación de recepción por parte del servidor; la prueba no instrumenta un cliente WebSocket para medir el tiempo adicional hasta la difusión del evento a los pasajeros y al panel administrativo, por lo que ese tramo posterior queda fuera del alcance de esta métrica.

Tabla 32 Prueba de rendimiento del endpoint de ubicación GPS (P-21)

Escenario	Herramienta	VUs	Duración	Tiempo promedio	Tiempo Mínimo	Tiempo Máximo	% Éxito
Endpoint GPS 48 choferes	k6	48	30 s	2.95 s	463 ms	4.29 s	100%

Prueba de GPS en segundo plano

La prueba P-22 verificó el funcionamiento del GPS en segundo plano de la aplicación móvil del chofer. El criterio de aceptación definido fue que el sistema transmita un mínimo de 15 actualizaciones por minuto con la pantalla del dispositivo apagada, sin intervención del conductor. Con el servicio activo y la pantalla apagada durante 5 minutos continuos, el sistema continuó transmitiendo la ubicación GPS al servidor cada 3 segundos mediante el servicio en segundo plano de Expo Task Manager, obteniendo 20 actualizaciones por minuto de forma continua. Este resultado supera el umbral mínimo definido, confirmando que el GPS en segundo plano opera correctamente en dispositivos Android 10, 11, 12 y 13, garantizando el monitoreo ininterrumpido de la flota sin requerir que el chofer mantenga la aplicación activa en pantalla.

Tabla 33 Prueba de GPS en segundo plano (P-22).

#	Prueba	Herramienta	Duración	Resultado	Criterio de aceptación	% Éxito
---	--------	-------------	----------	-----------	------------------------	---------

P-22	GPS segundo plano	Android	5 min	20 actualizaci ones/min	$\geq$ 15 actualizaciones/min	100%
------	-------------------------	---------	-------	-------------------------------	----------------------------------	------

Ilustración 57 Prueba de seguridad

Resúmenes

Recuento de alertas por riesgo y nivel de confianza

Esta tabla muestra el número de alertas para cada nivel de riesgo y confianza incluidas en el informe.

(Los porcentajes entre paréntesis representan el recuento como porcentaje del número total de alertas incluidas en el informe, redondeado a una cifra decimal).

		Confianza				
		Confirmado por Usuario	Medios de comunicación		Baja	Total
Riesgo	Alto	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)
	Medio	0 (0,0 %)	1 (10,0 %)	2 (20,0 %)	0 (0,0 %)	3 (30,0 %)
	Bajo	0 (0,0 %)	2 (20,0 %)	2 (20,0 %)	0 (0,0 %)	4 (40,0 %)
	Informativo	0 (0,0 %)	0 (0,0 %)	1 (10,0 %)	2 (20,0 %)	3 (30,0 %)
	Total	0 (0,0 %)	3 (30,0 %)	5 (50,0 %)	2 (20,0 %)	10 (100%)

Recuento de alertas por sitio y riesgo

Esta tabla muestra, para cada sitio en el que se generó una o más alertas, el número de alertas generadas en cada nivel de riesgo.

Las alertas con un nivel de confianza de "Falso positivo" se han excluido de estos recuentos.

(Los números entre paréntesis indican la cantidad de alertas generadas para el sitio con un nivel de riesgo igual o superior a ese valor).

		Riesgo			
		Alto (= Alto)	Medio (>= Medio)	Bajo (>= Bajo)	Informativo (>= Informativo)
Sitio	https://unibolivartransit.com	0 (0)	3 (3)	4 (7)	3 (10)

Tabla 34 Análisis de seguridad con OWASP ZAP (P-23)

Nivel de riesgo	Cantidad	Descripción
Alto	0	Sin vulnerabilidades críticas
Medio	3	Configuración de cabeceras HTTP
Bajo	4	Divulgación de información de versión
Informativo	3	Referencias de configuración
Total	10	

La prueba P-23 se ejecutó mediante un análisis automatizado de seguridad con OWASP ZAP sobre el dominio unibolivartransit.com. El análisis detectó un total de 10 alertas distribuidas en cuatro niveles de riesgo. No se detectaron vulnerabilidades de nivel alto. Las alertas de nivel medio corresponden a configuraciones de cabeceras HTTP como Content Security Policy y Anti-Clickjacking. Las alertas de nivel bajo están relacionadas con la divulgación de información del servidor a través de cabeceras de respuesta HTTP. El resultado del análisis automatizado no detectó vulnerabilidades de nivel alto, lo que indica que el sistema cumple con los estándares básicos de seguridad evaluados por la herramienta. Las alertas de nivel medio y bajo identificadas no comprometen el funcionamiento del sistema y serán consideradas como mejoras en versiones futuras.

Pruebas de compatibilidad

Las pruebas de compatibilidad verificaron el funcionamiento de las aplicaciones móviles en diferentes versiones de Android. Las aplicaciones desarrolladas con Expo SDK 52 se instalaron y ejecutaron correctamente en dispositivos con Android 10, 11, 12 y 13, confirmando la compatibilidad del sistema con los dispositivos disponibles en el contexto local de Guaranda.

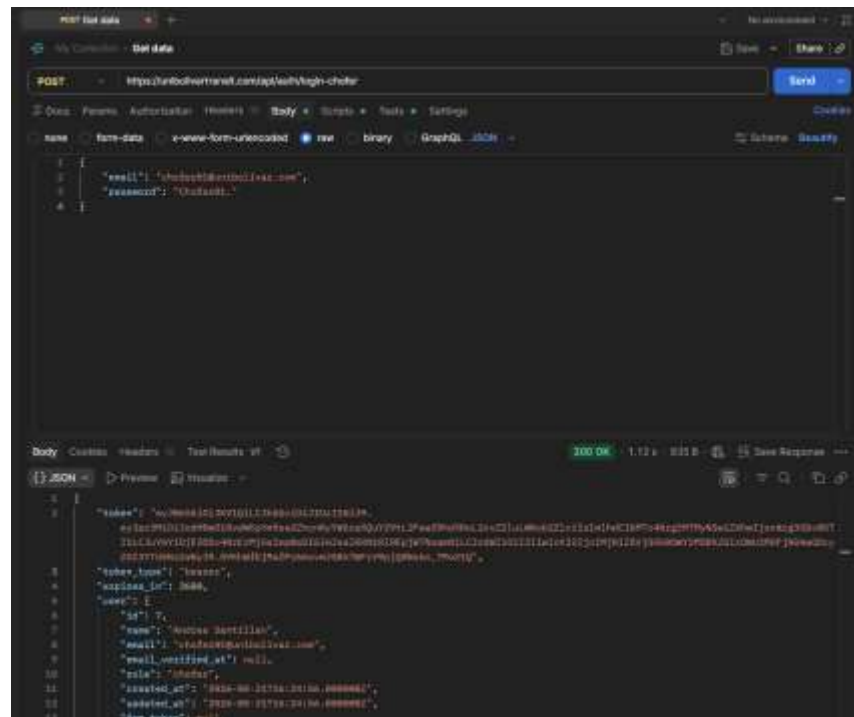
Tabla 35 Pruebas de compatibilidad en Android.

#	Versión Android	App Pasajero	App Chofer	Estado
P-24	Android 10	Compatible	Compatible	Cumple
P-25	Android 11	Compatible	Compatible	Cumple
P-26	Android 12	Compatible	Compatible	Cumple
P-27	Android 13	Compatible	Compatible	Cumple

Pruebas de tiempo de respuesta de endpoints

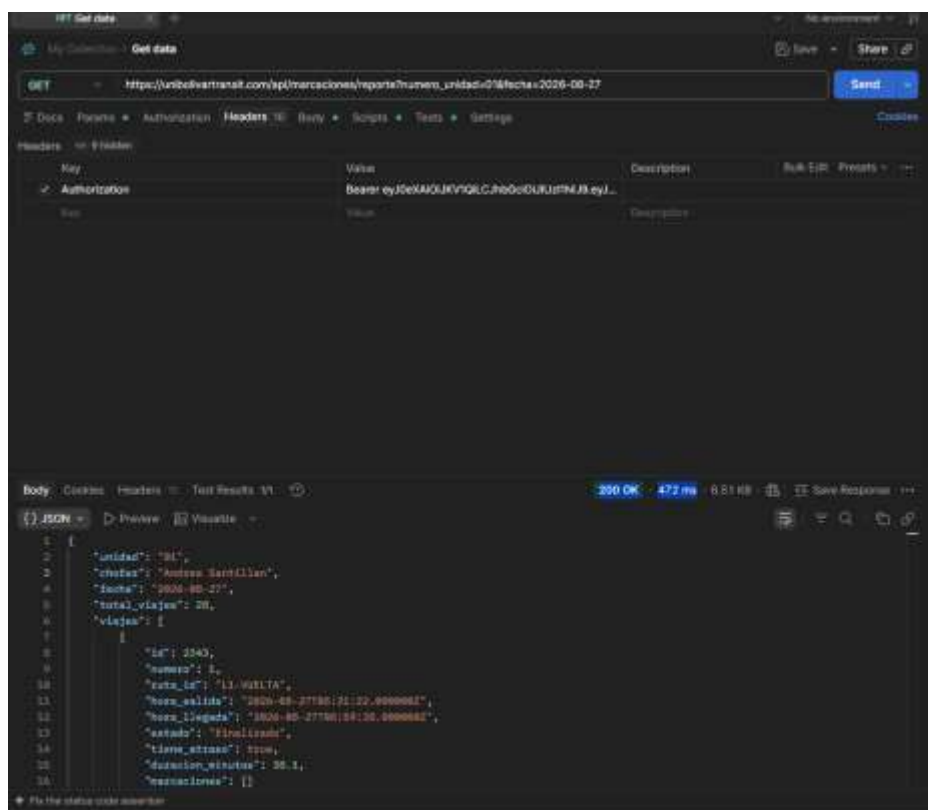
Para verificar el tiempo de respuesta de cada endpoint de la API REST se utilizó la herramienta Postman, ejecutando peticiones reales al servidor de producción unibolivartransit.com.

Ilustración 58 P-28 — /api/auth/login-chofer



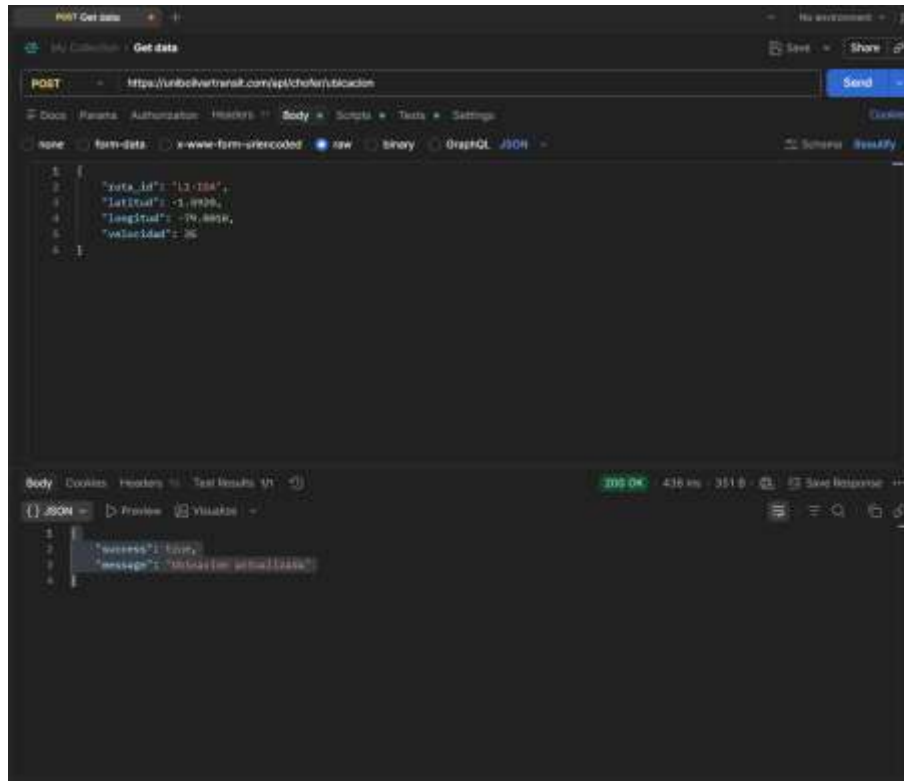
La prueba P-28 verificó el tiempo de respuesta del endpoint `/api/auth/login-chofer`, el cual autentica al chofer y genera el token JWT. La petición respondió con código 200 OK en 1.12 segundos.

Ilustración 59 P-29 — /api/marcaciones/reporte



La prueba P-29 verificó el tiempo de respuesta del endpoint `/api/marcaciones/reporte`, el cual genera el reporte de puntualidad por unidad y fecha. La petición respondió con código 200 OK en 472 ms.

Ilustración 60 P-30 — /api/chofer/ubicación



La prueba P-30 verificó el tiempo de respuesta del endpoint `/api/chofer/ubicacion`, el cual recibe la ubicación GPS del chofer y la transmite por WebSocket. La petición respondió con código **200 OK** en **438 ms**.

Tabla 36 Tiempos de respuesta de endpoints API REST.

#	Endpoint	Método	Tiempo	Estado
P-28	<code>/api/auth/login-chofer</code>	Post	1120 ms	200 OK
P-29	<code>/api/marcaciones/reporte</code>	Get	472 ms	200 OK
P-30	<code>/api/chofer/ubicacion</code>	Post	438 ms	200 OK

Corrección de errores detectados

Durante la ejecución de las pruebas se identificaron y corrigieron errores relacionados con la sincronización de la hora UTC entre el servidor y las aplicaciones móviles, la actualización en tiempo real de los marcadores en el mapa del panel administrativo y el bloqueo automático del chofer cuando tiene un permiso o descanso activo. Todos los errores fueron corregidos antes del despliegue final del sistema.

Capacitación al administrador de la cooperativa

Se realizó la capacitación al administrador de la Cooperativa de Transporte Urbano Universidad de Bolívar sobre el uso del panel administrativo web, cubriendo los módulos de gestión de usuarios, líneas, rutas, paradas, horarios, despacho semanal, monitoreo de flota en tiempo real y generación de reportes de puntualidad en formato PDF. La capacitación permitió orientar al administrador en el manejo de las principales funcionalidades del sistema y en la ejecución de las actividades necesarias para la gestión y supervisión del servicio de transporte. Como evidencia de esta actividad, en el **Anexo 7** se presentan los registros correspondientes a la capacitación realizada.

Despliegue final en servidor de producción

El sistema fue desplegado en el servidor de producción DigitalOcean Ubuntu 22.04 con dominio unibolivartransit.com y certificado SSL mediante Let's Encrypt, garantizando la seguridad y disponibilidad del servicio para los usuarios finales.

Ilustración 61 Sistema en el servidor de producción DigitalOcean Ubuntu 22.04



CONCLUSIONES

- Mediante la aplicación de la técnica de la entrevista dirigida al administrador y a los choferes de la Cooperativa de Transporte Urbano Universidad de Bolívar, se levantaron los requerimientos funcionales del sistema, identificando las necesidades operativas relacionadas con el control de rutas, paradas y horarios. Como resultado se definieron 22 historias de usuario que sirvieron de base para el desarrollo de las tres aplicaciones: panel administrativo web, aplicación móvil para pasajeros y aplicación móvil para choferes.
- Se diseñó e implementó un prototipo funcional de geolocalización que permite monitorear en tiempo real la ubicación de las unidades de transporte urbano mediante el GPS del teléfono móvil del chofer. El sistema transmite la posición de cada bus cada tres segundos mediante tecnología WebSocket, actualizando automáticamente el mapa del pasajero y el panel administrativo web. Adicionalmente, el panel administrativo permite al administrador gestionar choferes, líneas, rutas, paradas, horarios, despacho semanal y reportes de puntualidad en formato PDF.
- Se ejecutaron 30 pruebas distribuidas en cuatro categorías: funcionales, de rendimiento, de seguridad y de compatibilidad. Las 19 pruebas funcionales obtuvieron un porcentaje de cumplimiento del 100% en todas las funcionalidades evaluadas. Las pruebas de rendimiento ejecutadas con k6 demostraron que el endpoint GPS respondió en 2.71 segundos con 48 choferes simultáneos — equivalente a la flota completa de la cooperativa — con un porcentaje de éxito del 100% y sin ninguna solicitud fallida (P-20). La latencia del WebSocket fue de 2.95 segundos bajo la misma carga con 0% de fallos (P-21). Las pruebas de seguridad con OWASP ZAP no detectaron vulnerabilidades de nivel alto, aunque se identificaron 3 alertas de nivel medio relacionadas con configuraciones de cabeceras HTTP que serán consideradas como mejoras en versiones futuras. Las pruebas de compatibilidad confirmaron el funcionamiento correcto de las aplicaciones en Android 10, 11, 12 y 13.

RECOMENDACIONES

- Se recomienda a los directivos de la Cooperativa de Transporte Universidad de Bolívar realizar de manera anual el pago de los servicios necesarios para mantener el sistema UniBolívar Transit disponible en línea. Esto incluye el servicio de alojamiento del servidor en DigitalOcean, cuyo costo aproximado es de USD 12,00 mensuales, y la renovación del dominio administrado mediante Cloudflare, con un costo aproximado de USD 11,00 anuales. La renovación oportuna de estos servicios permitirá garantizar la continuidad de funcionamiento y acceso al sistema web y a las aplicaciones móviles.
- Se recomienda implementar las configuraciones de cabeceras HTTP de seguridad identificadas durante las pruebas con OWASP ZAP, tales como Content Security Policy, Anti-Clickjacking y Strict-Transport-Security, con el fin de fortalecer la seguridad del sistema en el entorno de producción.
- Se recomienda ejecutar pruebas de rendimiento periódicas con k6 a medida que la cantidad de usuarios del sistema aumente, con el propósito de identificar cuellos de botella en el servidor y garantizar tiempos de respuesta aceptables bajo condiciones de alta demanda.

BIBLIOGRAFÍA

- Almeida Muñoz, J. F., & Solís Cuñez, S. M. (2019). *Desarrollo de una aplicación móvil Android para la consulta de rutas de una línea de buses urbanos que circulan por la ciudad de Quito referenciando los puntos de partida y destino del usuario* [Trabajo de titulación, Universidad Politécnica Salesiana]. <https://dspace.ups.edu.ec/handle/123456789/16912>
- Asamblea Nacional Constituyente. (2008). *Constitución de la República del Ecuador*. Registro Oficial 449. <https://www.asambleanacional.gob.ec/documentos/Constitucion-2008.pdf>
- Asamblea Nacional del Ecuador. (2008). *Ley Orgánica de Transporte Terrestre, Tránsito y Seguridad Vial*. Registro Oficial Suplemento 398. https://www.mit.gob.ec/wp-content/uploads/downloads/2021/08/LOTAIP_6_Ley-Organica-de-Transporte-Terrestre-Transito-y-Seguridad-Vial-2021.pdf
- Asamblea Nacional del Ecuador. (2010). *Código Orgánico de Organización Territorial, Autonomía y Descentralización (COOTAD)*. Registro Oficial Suplemento 303. <https://www.cpccs.gob.ec/wp-content/uploads/2020/01/cootad.pdf>
- Asamblea Nacional del Ecuador. (2011). *Ley Orgánica de Economía Popular y Solidaria*. Registro Oficial 444. <https://www.vicepresidencia.gob.ec/wp-content/uploads/downloads/2018/09/Ley-Organica-de-Economia-Popular-y-Solidaria.pdf>
- Asamblea Nacional del Ecuador. (2021). *Ley Orgánica de Protección de Datos Personales*. Registro Oficial Quinto Suplemento 459. <https://www.telecomunicaciones.gob.ec/wp-content/uploads/2021/06/Ley-Organica-de-Datos-Personales.pdf>
- Bravo Cobeña, W. D., & Cedeño Intriago, B. A. (2025). *Aplicación móvil para la visualización y seguimiento de rutas de la cooperativa de buses urbanos urbanbass del cantón Bolívar* [Trabajo de titulación]. Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

- Chávez, J. (2018). *Implementación de una aplicación web para optimizar la gestión de la óptica Chávez*. [Tesis de licenciatura, Universidad Nacional Daniel Alcides Carrión]. http://repositorio.undac.edu.pe/bitstream/undac/1005/4/T026_72367456_T.pdf
- Congreso Nacional del Ecuador. (2002). *Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos*. Ley No. 2002-67, Registro Oficial Suplemento 557. <https://www.telecomunicaciones.gob.ec/wp-content/uploads/downloads/2012/11/Ley-de-Comercio-Electronico-Firmas-y-Mensajes-de-Datos.pdf>
- Díaz Concha, A. S. (2022). *Geolocalización de rutas de buses en tiempo real: Caso de estudio ciudad de Otavalo* [Trabajo de titulación, Universidad Politécnica Salesiana]. <http://dspace.ups.edu.ec/handle/123456789/22102>
- Fallas, J. (2002). *Sistema de posicionamiento global*. https://www.researchgate.net/publication/228389461_SISTEMA_DE_POSICIONAMIENTO_GLOBAL
- Gobierno Autónomo Descentralizado del Cantón Guaranda. (s. f.). *Unidad Municipal de Tránsito del cantón Guaranda*. GAD Municipal de Guaranda. Recuperado <https://www.guaranda.gob.ec/newsiteCMT/unidad-municipal-de-transito-del-canton-guaranda/>
- González Tigrero, D. A. (2019). *Desarrollo de sistema de mapeo y visualización de rutas de buses urbanos de la provincia de Santa Elena para la agencia nacional de tránsito. Módulo: Capa de servicios web y geográfico* [Trabajo de titulación previo a la obtención del Título de Ingeniero de Sistemas, Universidad Estatal Península de Santa Elena]. <https://repositorio.upse.edu.ec/server/api/core/bitstreams/25cdcb1b-81dd-4701-85c1-7547ba7086eb/content>
- Hernández Salazar, E. Y., & Beltrán, C. A. (2020). SCRUM, un enfoque práctico de metodología ágil para la ingeniería de software. *Revista Ibérica de Sistemas e Tecnologías de Informação*, 8(2). <https://www.redalyc.org/journal/5604/560467941011/560467941011.pdf>

- Hidalgo Astudillo, A. D. (2018). *Sistema de monitoreo y ubicación de buses basado en la tecnología GPS en la cooperativa de transporte urbano 7 de octubre en la ciudad de Quevedo*. [Trabajo de titulación, Universidad Técnica de Ambato]. <http://repositorio.uta.edu.ec/handle/123456789/27421>
- Iborra, N. (2023). *Introducción a Node.js*. <https://nachoiborraies.github.io/nodejs/md/es/01b.pdf>
- Jaramillo Zambrano, E. D. (2018). *Desarrollo de aplicación móvil con geolocalización de líneas de autobuses para el GAD Municipalidad de Ambato*. [Trabajo de titulación, Pontificia Universidad Católica del Ecuador]. <https://repositorio.puce.edu.ec/server/api/core/bitstreams/b4c177c8-5ced-45af-98f8-159f1d1b86c6/content>
- Lazcano Calixto, R. N. L. C., Valencia González, Luis Ángel, & Baena Díaz, Daniel Esteban. (2019). React Native: Acortando las distancias entre desarrollo y diseño móvil multiplataforma. *Revista Digital Universitaria*, 20(5). <https://doi.org/10.22201/codeic.16076079e.2019.v20n5.a5>
- Leguizamón, S. (2006). Acerca del Sistema de Posicionamiento Global (GPS). *Revista de la Facultad de Ingeniería*, 15. <https://dialnet.unirioja.es/descarga/articulo/6581740.pdf>
- León Ardila, K. Y. (2020). *Aplicabilidad de Metodologías Ágiles y su éxito en una empresa de desarrollo de software*. [Trabajo de grado, Universidad Surcolombiana]. <https://repositorio.usc.edu.co/server/api/core/bitstreams/fb7c125a-ce8f-482a-932c-dc7c6a5909f3/content>
- Navarro Cadavid, A., Fernández Martínez, J. D., & Morales Vélez, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 11(2), 30-39. <https://doi.org/10.15665/rp.v11i2.36>
- Ponsico Martín, P. (2017). *Docker: Gestión de contenedores* [Trabajo de fin de grado, Universitat Politècnica de Catalunya]. https://upcommons.upc.edu/bitstream/handle/2117/113040/Degree_thesis.pdf?sequence=1

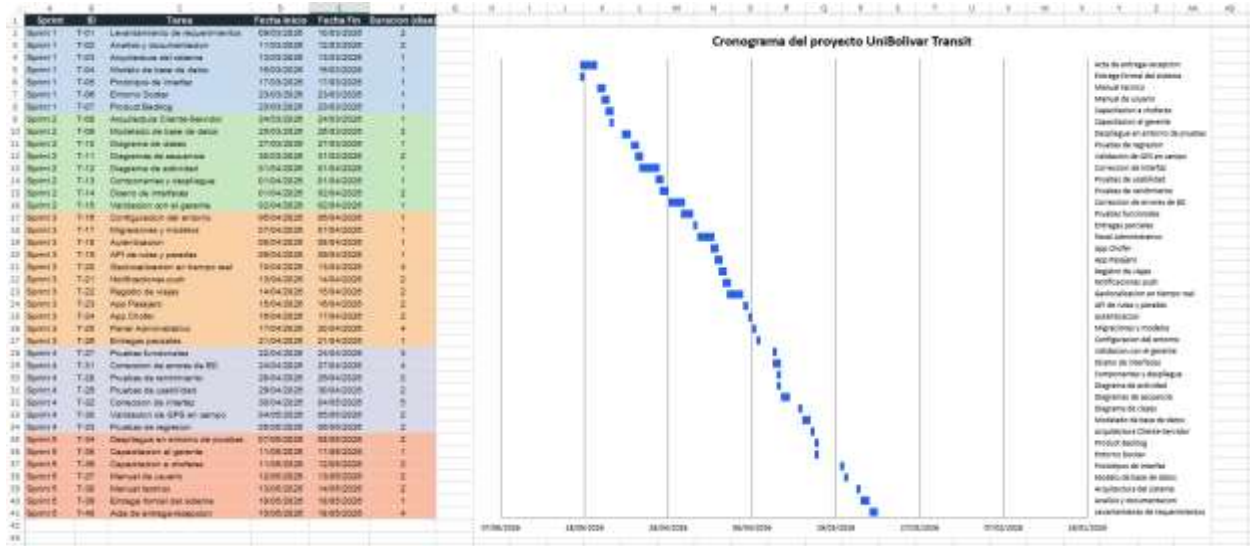
- Presidencia de la República del Ecuador. (2012). *Reglamento General de la Ley Orgánica de Economía Popular y Solidaria*. Decreto Ejecutivo 1061, Registro Oficial Suplemento 648.
https://www.oas.org/juridico/pdfs/mesicic4_ecu_regla2.pdf
- Presidencia de la República del Ecuador. (2023). *Reglamento General a la Ley Orgánica de Protección de Datos Personales* (Registro Oficial Suplemento 435).
<https://www.gob.ec/regulaciones/reglamento-general-ley-organica-proteccion-datos-personales-0>
- Sanz Mateo, M. I. (2015). *Metodología LEAN para el desarrollo de software. Ejemplo práctico de aplicación en empresa de desarrollo de software* [Trabajo de fin de grado, Universidad de Valladolid].
<https://uvadoc.uva.es/bitstream/handle/10324/13538/TFG-B.796.pdf;jsessionid=4884EFEF32ABC7F44A91EE2F2F078A66?sequence=1>
- Universidad Estatal de Milagro. (2020). *Conceptos básicos de base de datos*.
https://sga.unemi.edu.ec/media/recursotema/Documento_20201120171118.pdf

ANEXOS

Anexo 1.

Cronograma del Proyecto

Ilustración 62 Cronograma (Gantt)



Elaborado por: Erick Alarcón, Juan Pasto

Anexo 2.

Detalle del Presupuesto Estimado

Tabla del presupuesto estimado para el desarrollo del proyecto.

Tabla 37. Tabla del presupuesto estimado para el desarrollo del proyecto.

Recursos requeridos	Costo (USD)
Dominio web	15
Servidor	32
Transporte para levantamiento de información	50
Servicio de Internet	60
Teléfono	150
Computadora	700
Impresiones	5
Total, parcial	\$1,012.00
Imprevistos (10%)	\$101.20
Total, estimado	\$1,113.20

Nota: Presupuesto estimado de los recursos necesarios para el desarrollo del proyecto

Anexo 3.

Instrumentos de recopilación de datos (Entrevista).

Universidad Estatal de Bolívar

Facultad de Ciencias Administrativas, Gestión Empresarial e Informática

Carrera de Ingeniería de Software

Proyecto de Titulación

Tema: Sistema móvil y web para el seguimiento en tiempo real de buses urbanos de la Cooperativa de Transporte Universidad de Bolívar — UniBolivar Transit

ENTREVISTA N. 1 — GERENTE DE LA COOPERATIVA

Objetivo: Conocer la situación actual del sistema de control de buses, los principales problemas operativos y las expectativas de la directiva frente a la implementación del nuevo sistema digital.

Entrevistado Sr. Riquelme Danilo Vásconez Vásconez

Cargo Gerente General — Cooperativa Universidad de Bolívar

Fecha: 09-03-2026

Lugar Oficinas de la Cooperativa Universidad de Bolívar

1. ¿Cómo se controla actualmente la salida y llegada de los buses en cada recorrido?

Respuesta: Actualmente el control se realiza de forma manual mediante registros en hojas físicas o planillas en papel. Los despachadores anotan la hora de salida de cada unidad y al final del día se compilan los datos, lo que genera retrasos en la obtención de información y posibles errores de registro.

2. ¿Qué información considera indispensable registrar en el nuevo sistema?

Respuesta: Es fundamental registrar la ubicación en tiempo real de cada bus, el nombre del chofer asignado, el número de unidad, la línea de recorrido, los horarios

programados de salida y llegada, y los atrasos que se generen durante la operación diaria.

3. ¿Qué problemas principales enfrenta la cooperativa con el sistema manual actual?

Respuesta: Los principales problemas son la falta de información en tiempo real para los pasajeros, la dificultad para controlar los horarios de las unidades, la ausencia de reportes automáticos sobre el rendimiento de cada chofer y la imposibilidad de detectar atrasos a tiempo para tomar decisiones correctivas.

4. ¿Existen limitaciones tecnológicas — internet, equipos, personal capacitado — que se deban considerar?

Respuesta: Los choferes cuentan con teléfonos celulares básicos con acceso a internet por datos móviles. En algunas zonas de la ruta la señal puede ser intermitente. El personal administrativo dispone de computadoras con acceso a internet de banda ancha en la oficina central.

5. ¿Qué esperan lograr con la implementación del nuevo sistema UniBolívar Transit?

Respuesta: Se espera mejorar significativamente la experiencia del pasajero, quien podrá ver en tiempo real dónde se encuentra el bus más cercano y cuánto tiempo tarda en llegar a su parada. Internamente, se busca un mejor control de los horarios, reducción de atrasos y generación automática de reportes para la toma de decisiones.

6. ¿Cuál sería el impacto positivo esperado para la cooperativa con este sistema?

Respuesta: El principal impacto positivo sería la mejora en la imagen institucional de la cooperativa frente a los usuarios, quienes actualmente expresan insatisfacción por la falta de información. Además, se espera mayor eficiencia operativa, reducción de conflictos entre personal y pasajeros, y un mejor aprovechamiento de las unidades disponibles.

ENTREVISTA N. 2 — CHOFER DE LA COOPERATIVA

Objetivo: Conocer la experiencia operativa diaria del chofer, las dificultades que enfrenta en su turno y sus expectativas respecto al uso de la aplicación móvil del nuevo sistema.

Entrevistado: Andrés Coloma

Cargo: CHOFER — Cooperativa Universidad de Bolívar

Fecha:09-03-2026

Lugar: Oficinas de la Cooperativa Universidad de Bolívar

1. ¿Cómo inicia y registra su turno de trabajo actualmente?

Respuesta: Actualmente el registro del turno se hace de forma manual. Al llegar al punto de despacho, el despachador anota la hora de salida en una planilla física. No existe ningún sistema digital que confirme el inicio del recorrido de forma automática.

2. ¿Cómo se comunica con la oficina central durante su recorrido en caso de atrasos o emergencias?

Respuesta: La comunicación se hace exclusivamente por llamada telefónica. Si hay un imprevisto — tráfico, desperfecto mecánico, accidente — debo llamar al despachador para informar la situación. A veces no es posible hacerlo mientras se conduce, lo que genera demoras en la comunicación.

3. ¿Ha tenido problemas por falta de información de los pasajeros respecto a los horarios de llegada?

Respuesta: Sí, constantemente. Los pasajeros se quejan porque no saben cuánto tiempo falta para que llegue el bus, especialmente en las paradas más alejadas del centro. Muchos esperan mucho tiempo sin saber si el bus ya pasó o aún está por llegar, lo que genera molestia y conflictos.

4. ¿Estaría dispuesto a usar una aplicación móvil durante su turno para enviar su ubicación en tiempo real?

Respuesta: Sí, estaría dispuesto, siempre que la aplicación sea fácil de usar y no distraiga la conducción. Lo ideal sería que simplemente se active al iniciar el turno y funcione en segundo plano sin necesidad de estar revisando el teléfono constantemente durante el recorrido.

5. ¿Qué funciones le gustaría que tuviera la aplicación móvil para el chofer?

Respuesta: Las funciones más importantes serían: poder iniciar y finalizar el turno con un solo botón, que la aplicación envíe automáticamente la ubicación sin intervención manual, ver el horario programado de mi recorrido y recibir alertas cuando voy con atraso respecto al horario establecido.

6. ¿Considera que el uso de tecnología GPS mejoraría su desempeño y el servicio a los pasajeros?

Respuesta: Sí, definitivamente. Si los pasajeros pueden ver en su teléfono dónde está el bus, habrá menos conflictos en las paradas y se optimizará el tiempo de espera. También sería útil para la cooperativa poder verificar que los recorridos se están cumpliendo correctamente.

7. ¿Tiene acceso a datos móviles en su teléfono durante todo el recorrido?

Respuesta: Sí, tengo acceso a internet por datos móviles en la mayor parte del recorrido. En algunas zonas alejadas la señal puede debilitarse, pero en el recorrido urbano principal la cobertura es generalmente buena.

Anexo 4.

Repositorios del Código Fuente en GitHub

El presente anexo contiene los enlaces a los repositorios de código fuente del sistema UniBolívar Transit, almacenados en la plataforma GitHub bajo el usuario alarconerick612. Cada repositorio corresponde a un componente independiente del sistema.

Repositorios del Proyecto

Backend — API REST Laravel

- Repositorio: unibolivar-backend
- URL: <https://github.com/alarconerick612/unibolivar-backend>
- Rama principal: main
- Lenguaje: PHP 8.4 / Laravel 11

Contiene el código fuente completo del backend del sistema: controladores de API REST, modelos Eloquent, migraciones de base de datos, eventos de WebSocket (BusUbicacionActualizada), middlewares de autenticación JWT, rutas de API, y configuración de Docker Compose para despliegue en producción.

Panel Administrativo — Next.js

- Repositorio: unibolivar-admin
- URL: <https://github.com/alarconerick612/unibolivar-admin>
- Rama principal: main
- Lenguaje: TypeScript / Next.js 15

Contiene el código fuente completo del panel administrativo web: páginas del dashboard, módulos de gestión de líneas, rutas, paradas, horarios, usuarios, viajes y reportes. Incluye el cliente de API (apiClient.ts), sistema de autenticación, y componentes reutilizables de interfaz.

Aplicación Móvil del Pasajero — React Native

- Repositorio: GoBolivar (App Pasajero)
- URL: <https://github.com/alarconerick612/GoBolivar>
- Rama principal: main
- Lenguaje: TypeScript / React Native / Expo SDK 52

Contiene el código fuente de la aplicación móvil para pasajeros: pantallas de inicio, mapa en tiempo real con marcadores GPS de buses, lista de rutas y paradas, consulta de horarios, sistema de suscripciones a paradas para notificaciones de proximidad, y hook de WebSocket (useBusGPS.ts) para recepción de datos en tiempo real mediante Pusher/Soketi.

Aplicación Móvil del Chofer — React Native

- Repositorio: GoBolivarChofer (App Chofer)
- URL: <https://github.com/alarconerick612/GoBolivarChofer>
- Rama principal: main
- Lenguaje: TypeScript / React Native / Expo SDK 52

Contiene el código fuente de la aplicación móvil para choferes: pantalla de login, selección de ruta activa, inicio y fin de turno con un toque, transmisión automática de ubicación GPS en segundo plano cada 2 segundos mediante expo-location y TaskManager, e integración con el backend para registro automático de viajes.

Estructura de Ramas

Todos los repositorios utilizan una estructura de ramas simple basada en GitFlow simplificado:

- main: rama de producción, contiene el código estable desplegado en el servidor.
- develop (cuando aplica): rama de desarrollo activo, integrada periódicamente a main mediante pull requests.

Anexo 5.

Manual de Usuario - Panel Web Administrativo

Objetivo del manual

Este manual describe el uso de las principales interfaces del panel web administrativo de UniBolívar Transit. Las pantallas se presentan en el mismo orden del documento de prototipos proporcionado y utilizan las imágenes originales incluidas en dicho archivo.

Convención de lectura

En cada pantalla se emplean letras (A, B, C, etc.) para identificar grupos funcionales en la explicación. Estas letras son referencias del manual y no modifican la interfaz del sistema.

Orden de interfaces

Página de inicio (UniBolívar Transit)

Inicio de sesión (Panel Administrativo)

Panel Web Administrativo - Inicio (Dashboard)

Gestión de Usuarios

Modal “Nuevo Usuario”

Gestión de Líneas

Modal “Nueva Línea”

Gestión de Rutas - Selección de línea

Gestión de Rutas - Detalle IDA/VUELTA

Gestión de Paradas - Selección de sentido

Gestión de Paradas - Listado y acciones

Gestión de Horarios - Selección

Gestión de Horarios - Lista de frecuencias

Monitoreo en Tiempo Real

Despacho - Programación diaria

Despacho - Mi flota y disponibilidad

Flota de Buses

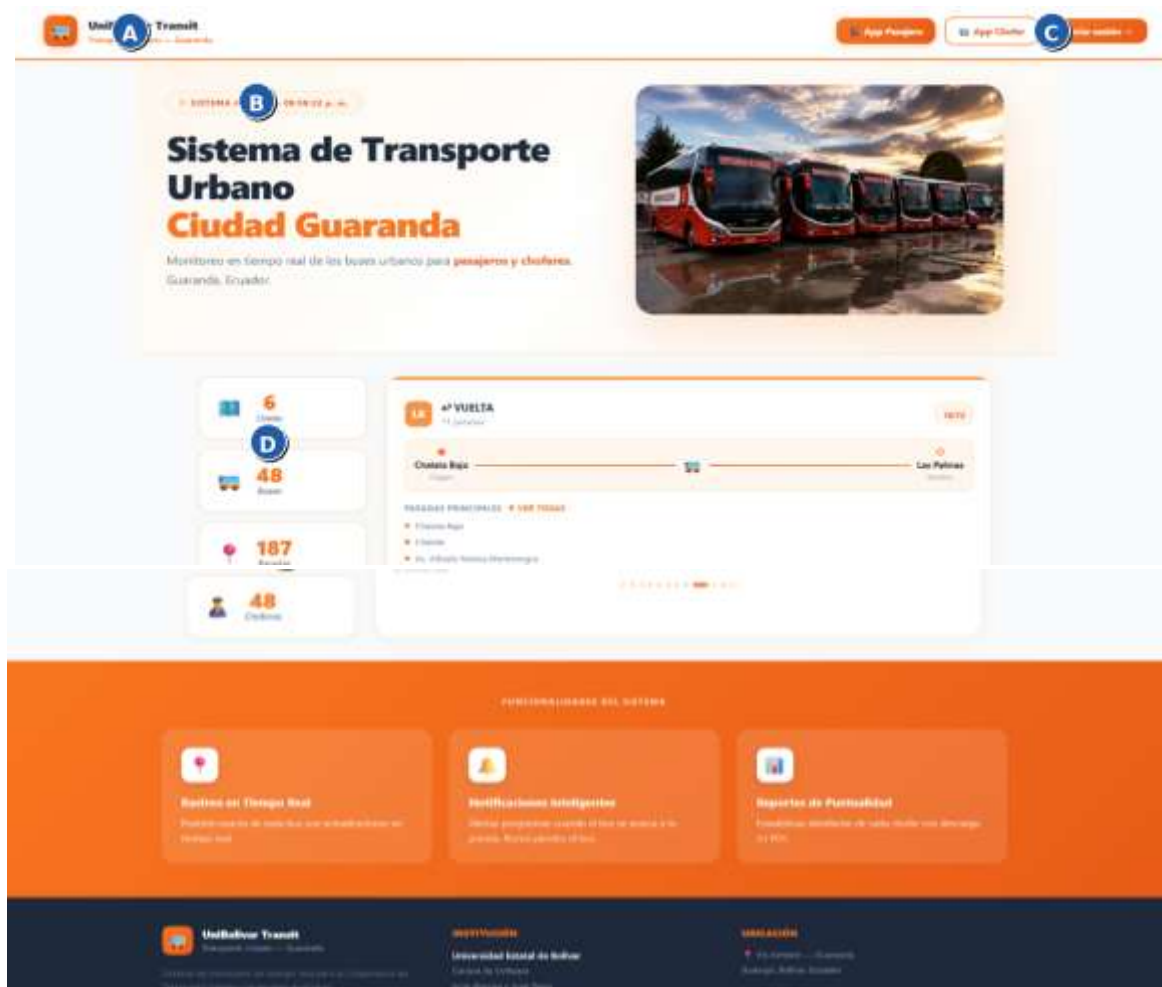
Reporte de Viajes

Reportes por Unidad

1. Página de inicio (UniBolívar Transit)

Objetivo: Presentar el sistema y ofrecer accesos directos al panel administrativo y a las aplicaciones móviles.

Ilustración 63 Página de inicio



Elementos principales

- A. Barra superior: muestra el logotipo y los accesos App Pasajero, App Chofer e Iniciar sesión.
- B. Estado del sistema: informa que el servicio se encuentra activo.

C. Acciones principales: Acceder al Panel y Descargar App.

D. Indicadores generales: resumen de líneas, buses, paradas, choferes y recorrido en tiempo real.

Procedimiento de uso

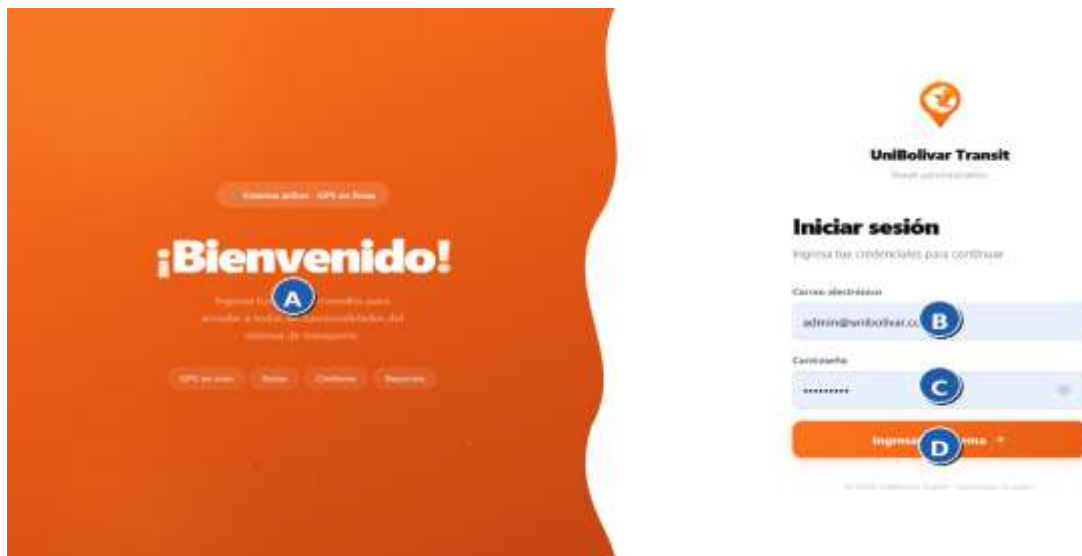
1. Revise el estado general del sistema en la parte central.
2. Para ingresar al área administrativa, seleccione “Acceder al Panel” o “Iniciar sesión”.
3. Para obtener una aplicación móvil, utilice “Descargar App” o los accesos App Pasajero/App Chofer.

2. Inicio de sesión (Panel Administrativo)

Objetivo: Permitir el acceso seguro al panel web mediante las credenciales del administrador.

Elementos principales

Ilustración 64 Inicio de sesión



A. Panel informativo: presenta el estado del sistema y referencias a GPS, rutas, choferes y reportes.

B. Campo Correo electrónico: permite ingresar el usuario asignado.

C. Campo Contraseña: recibe la clave de acceso y permite controlar su visibilidad.

D. Botón Ingresar al sistema: valida las credenciales y abre el panel administrativo.

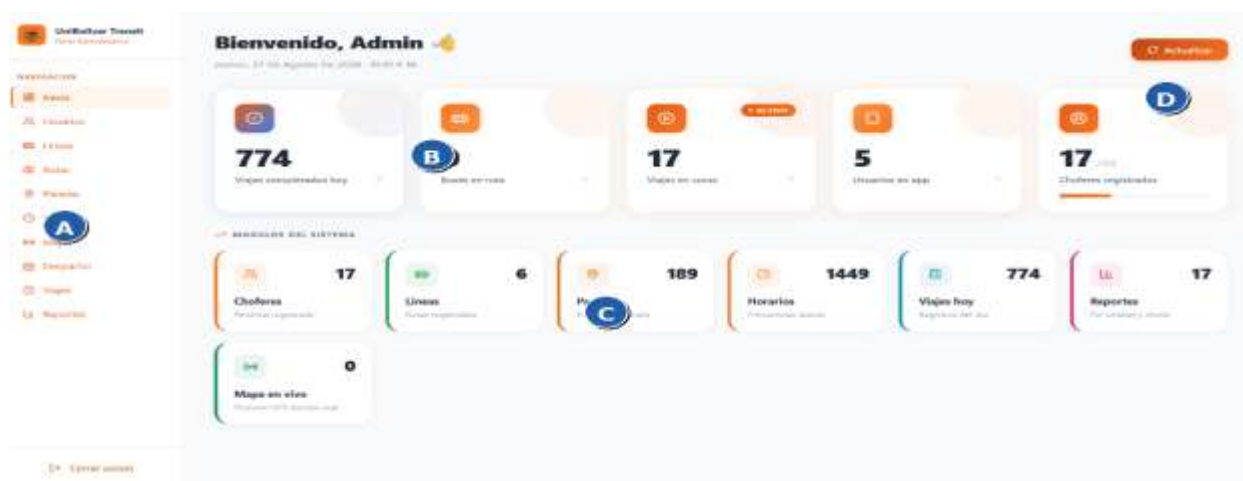
Procedimiento de uso

1. Ingrese el correo electrónico asignado al administrador.
2. Escriba la contraseña correspondiente.
3. Pulse “Ingresar al sistema”.
4. Si los datos son correctos, se abrirá el Dashboard; si no, revise las credenciales e inténtelo nuevamente.

3. Panel Web Administrativo - Inicio (Dashboard)

Objetivo: Mostrar una visión rápida del estado operativo del sistema y permitir el acceso a sus módulos principales.

Ilustración 65 Panel Web Administrativo - Inicio



Elementos principales

- A. Menú lateral: acceso a Usuarios, Líneas, Rutas, Paradas, Horarios, Mapa, Despacho, Viajes y Reportes.
- B. Indicadores superiores: resumen de la operación del día.
- C. Módulos del sistema: tarjetas con conteos y accesos rápidos.

D. Botón Actualizar: refresca la información mostrada en el panel.

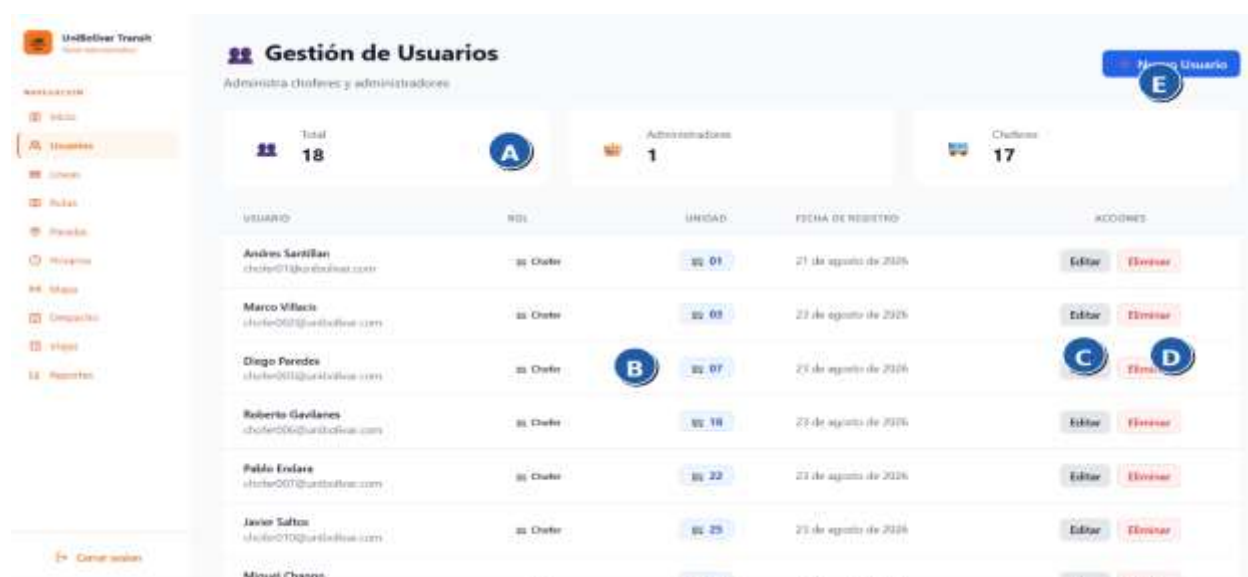
Procedimiento de uso

1. Revise los indicadores superiores para conocer el estado general de la operación.
2. Utilice las tarjetas de módulos o el menú lateral para abrir la función que necesite.
3. Pulse “Actualizar” cuando requiera refrescar los datos visibles.

4. Gestión de Usuarios

Objetivo: Administrar las cuentas registradas en el sistema, especialmente los usuarios con rol de chofer.

Ilustración 66 Gestión de Usuarios



USUARIO	ROL	UNIDAD	FECHA DE REGISTRO	ACCIONES
Andres Santillan chofer01@unibolivar.com	Chofer	U01	21 de agosto de 2026	Editar Eliminar
Marco Villacis chofer02@unibolivar.com	Chofer	U03	22 de agosto de 2026	Editar Eliminar
Diego Paredes chofer011@unibolivar.com	Chofer	U07	23 de agosto de 2026	Editar Eliminar
Roberto Gavilanes chofer006@unibolivar.com	Chofer	U018	23 de agosto de 2026	Editar Eliminar
Pedro Endara chofer007@unibolivar.com	Chofer	U022	23 de agosto de 2026	Editar Eliminar
Javier Salas chofer070@unibolivar.com	Chofer	U025	23 de agosto de 2026	Editar Eliminar
Miguel Chang				

Elementos principales

- A. Tarjetas de resumen: muestran el total de usuarios, administradores y choferes.
- B. Tabla de usuarios: presenta nombre, rol, unidad, fecha de registro y acciones.
- C. Editar: permite modificar los datos de un usuario.

D. Eliminar: permite retirar un registro cuando corresponda.

E. Nuevo Usuario: abre el formulario para registrar una cuenta.

Procedimiento de uso

1. Para registrar una cuenta, pulse “Nuevo Usuario”.
2. Para modificar un registro existente, pulse “Editar” en la fila correspondiente.
3. Para retirar un usuario, utilice “Eliminar” y confirme la acción si el sistema lo solicita.

5. Modal “Nuevo Usuario”

Objetivo: Registrar un nuevo usuario y asociarlo con el rol y, cuando corresponda, con una unidad.

Ilustración 67 Modal “Nuevo Usuario”

The image shows a web application interface for user management. A modal titled '+ Nuevo Usuario' is open in the center. The modal contains the following fields and controls:

- Nombre ***: A text input field with the value 'Ej. Juan Pérez'. A blue circular callout 'A' points to this field.
- Email ***: A text input field with the value 'admin@unibolivar.com'. A blue circular callout 'B' points to this field.
- Contraseña ***: A password input field with masked characters '*****'. A blue circular callout 'C' points to this field.
- Nombre**: A green text label indicating a required field.
- Rol ***: A dropdown menu with the selected value 'Choke'. A blue circular callout 'F' points to this field.
- Número de Unidad ****: A dropdown menu with the selected value '-- Seleccionar unidad --'. A blue circular callout 'E' points to this field.
- Buttons**: 'Cancelar' and 'Crear' buttons at the bottom right of the modal.

The background interface shows a sidebar with navigation options, a main header 'Gestión de Usuarios', and a table of existing users. A blue button 'Nuevo Usuario' is visible in the top right corner of the main interface.

Elementos principales

- A. Nombre: identifica al usuario.
- B. Email: registra el correo de acceso.
- C. Contraseña: define la clave inicial.
- D. Rol: permite seleccionar el tipo de usuario.
- E. Número de Unidad: permite asociar una unidad cuando aplica.
- F. Resumen de errores: muestra campos faltantes o datos que deben corregirse.

Procedimiento de uso

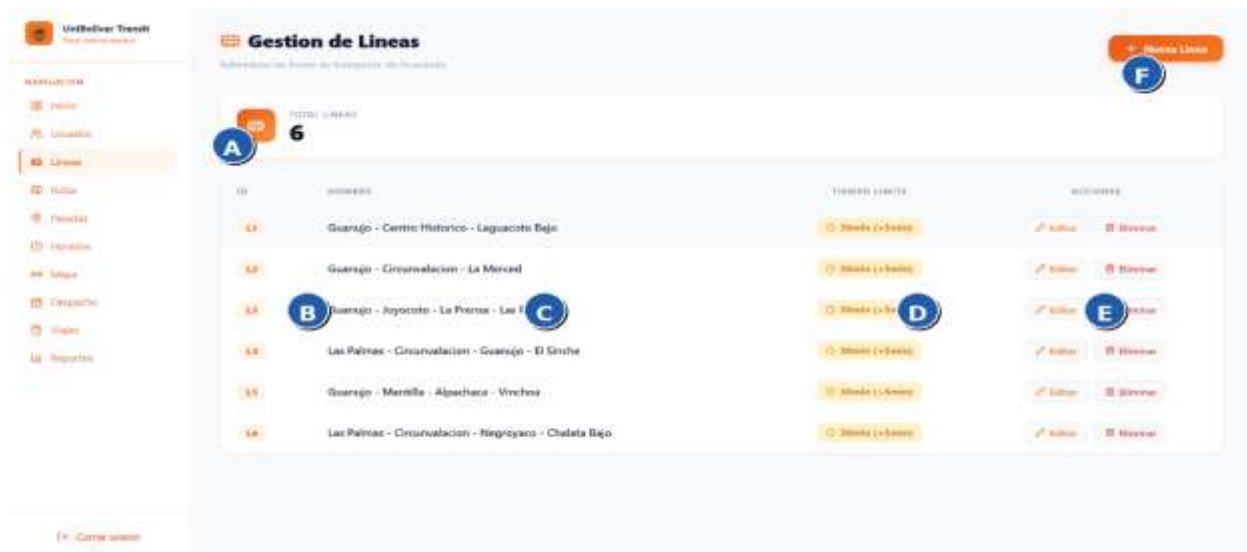
1. Complete Nombre, Email y Contraseña.

2. Seleccione el Rol correspondiente.
3. Seleccione el Número de Unidad cuando el usuario sea un chofer y la interfaz lo requiera.
4. Corrija cualquier mensaje de validación mostrado en rojo.
5. Pulse “Crear” para guardar o “Cancelar” para cerrar sin registrar cambios.

6. Gestión de Líneas

Objetivo: Consultar y mantener la información general de las líneas de transporte registradas.

Ilustración 68 Gestión de Líneas



Elementos principales

- A. Total de líneas: muestra la cantidad registrada.
- B. ID: identifica cada línea mediante códigos L1, L2, etc.
- C. Nombre: describe el recorrido principal.
- D. Tiempo límite: muestra el tiempo configurado para el viaje.
- E. Acciones: permite editar o eliminar una línea.

F. Nueva Línea: abre el formulario de creación.

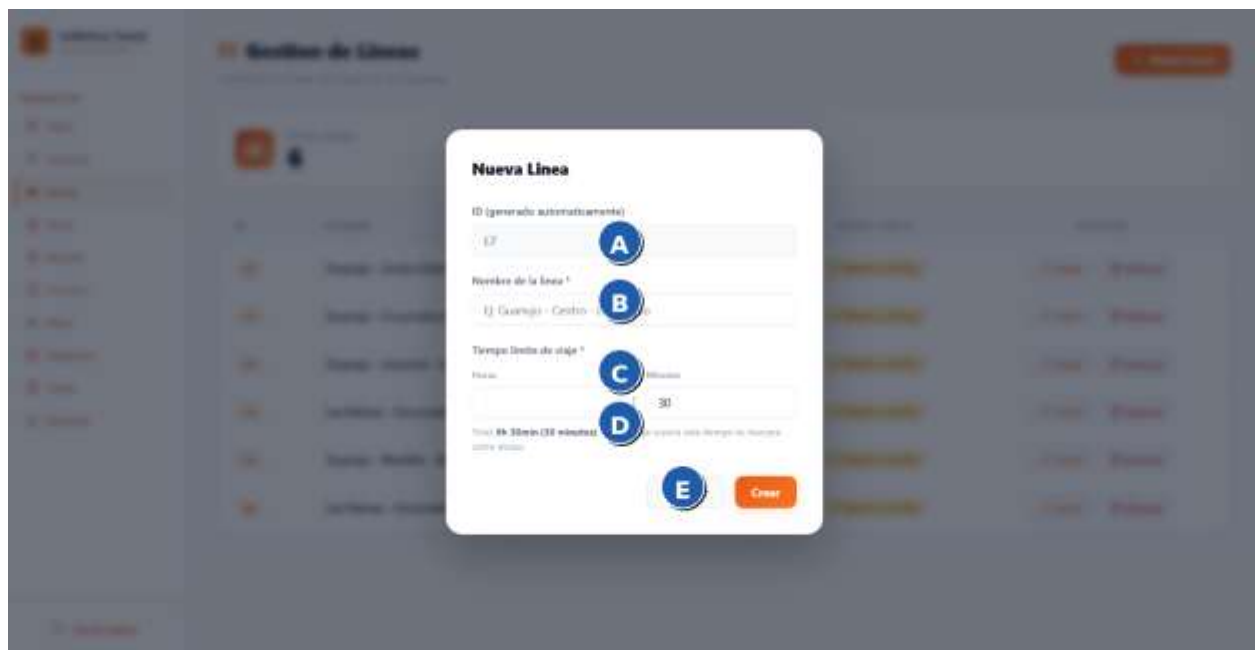
Procedimiento de uso

1. Revise la lista de líneas y sus tiempos límite.
2. Pulse “Editar” para modificar una línea existente.
3. Pulse “Eliminar” para retirar una línea cuando corresponda.
4. Use “Nueva Línea” para registrar una nueva configuración.

7. Modal “Nueva Línea”

Objetivo: Crear una nueva línea de transporte con su nombre y tiempo límite de viaje.

Ilustración 69 Modal “Nueva Línea”



Elementos principales

A. ID: se genera automáticamente y no requiere edición.

B. Nombre de la línea: permite describir el recorrido.

C. Horas y Minutos: configuran el tiempo límite de viaje.

D. Total calculado: resume el tiempo definido.

E. Crear/Cancelar: guarda o descarta el registro.

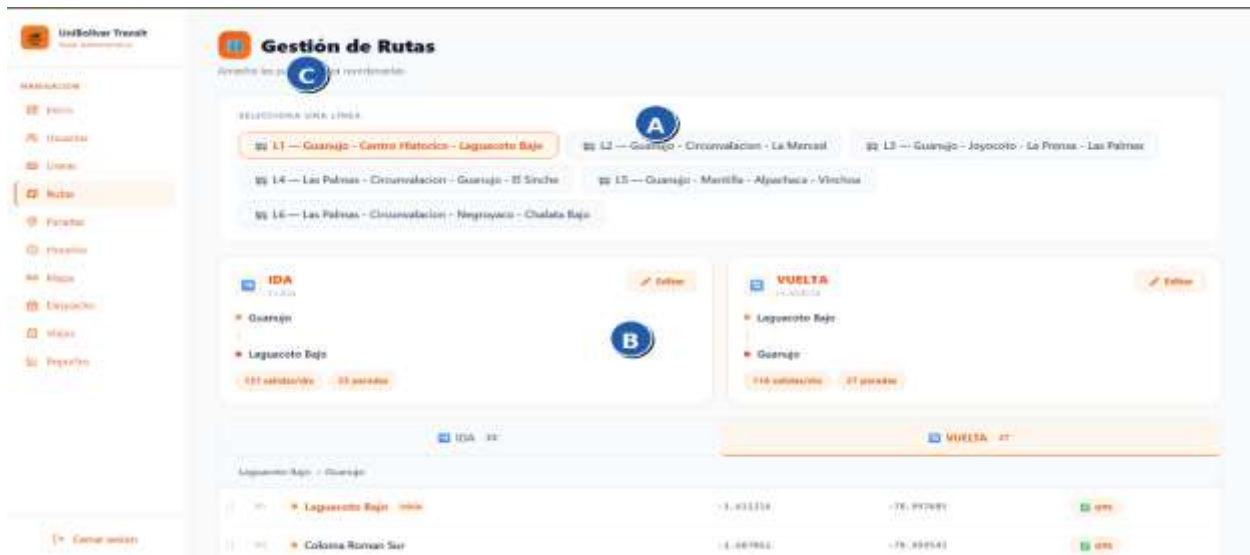
Procedimiento de uso

1. Verifique el ID generado automáticamente.
2. Escriba el nombre de la línea.
3. Configure las horas y los minutos del tiempo límite.
4. Revise el total calculado.
5. Pulse “Crear” para registrar la línea o “Cancelar” para salir sin guardar.

8. Gestión de Rutas - Selección de línea

Objetivo: Seleccionar una línea antes de revisar o administrar sus rutas de ida y vuelta.

Ilustración 70 Gestión de Rutas - Selección de línea



Elementos principales

- A. Selector de líneas: muestra las líneas L1 a L6 disponibles.
- B. Área de trabajo: permanece vacía hasta seleccionar una línea.
- C. Mensaje de selección: orienta al administrador para iniciar la consulta.

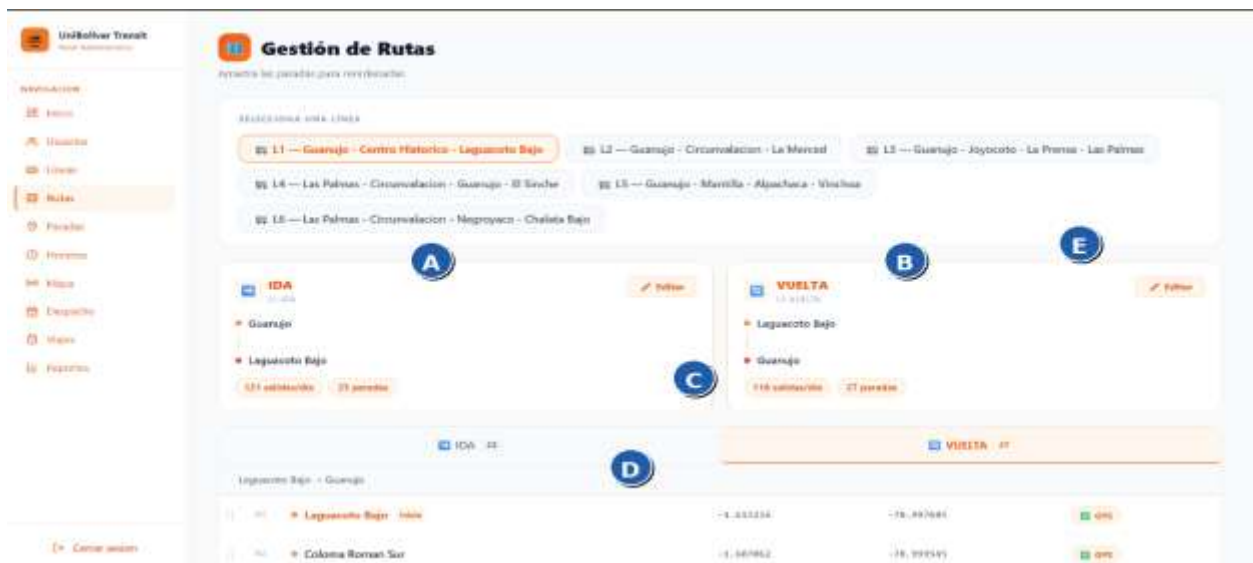
Procedimiento de uso

1. Ubique la línea que desea administrar.
2. Haga clic sobre la línea correspondiente.
3. El sistema cargará el detalle de sus recorridos IDA y VUELTA.

9. Gestión de Rutas - Detalle IDA/VUELTA

Objetivo: Visualizar la estructura de las rutas en ambos sentidos y la secuencia de sus paradas.

Ilustración 71 Gestión de Rutas - Detalle IDA/VUELTA



Elementos principales

- A. Resumen IDA: muestra origen, destino, salidas y número de paradas.
- B. Resumen VUELTA: presenta la información equivalente para el regreso.
- C. Pestañas IDA/VUELTA: permiten cambiar el sentido visualizado.
- D. Listado de paradas: muestra el orden, coordenadas y estado GPS.
- E. Editar: permite modificar información asociada al recorrido o sus paradas.

Procedimiento de uso

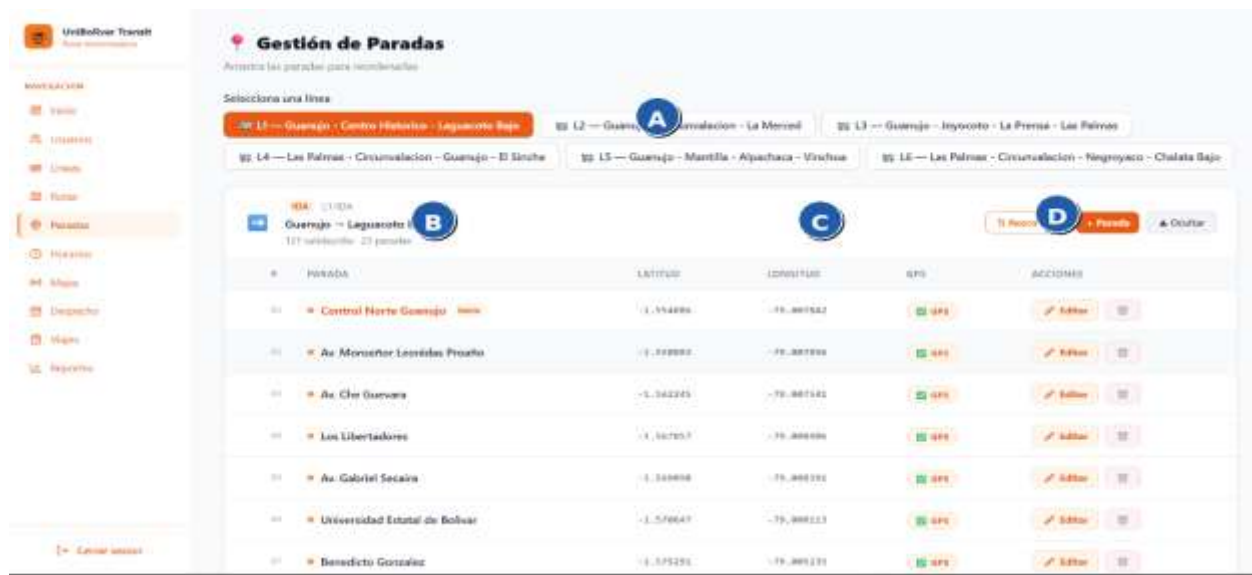
1. Seleccione la línea en la franja superior.
2. Revise los resúmenes de IDA y VUELTA.
3. Cambie de pestaña para consultar el sentido requerido.

4. Revise el orden de las paradas y sus coordenadas.
5. Utilice “Editar” cuando necesite actualizar la configuración disponible.

10. Gestión de Paradas - Selección de sentido

Objetivo: Elegir una línea y el sentido del recorrido antes de administrar sus paradas.

Ilustración 72 Gestión de Paradas - Selección de sentido



Elementos principales

- A. Selector de líneas: permite escoger L1 a L6.
- B. Tarjeta IDA: identifica el recorrido de salida.
- C. Tarjeta VUELTA: identifica el recorrido de retorno.
- D. Ver paradas: abre el listado del sentido seleccionado.

Procedimiento de uso

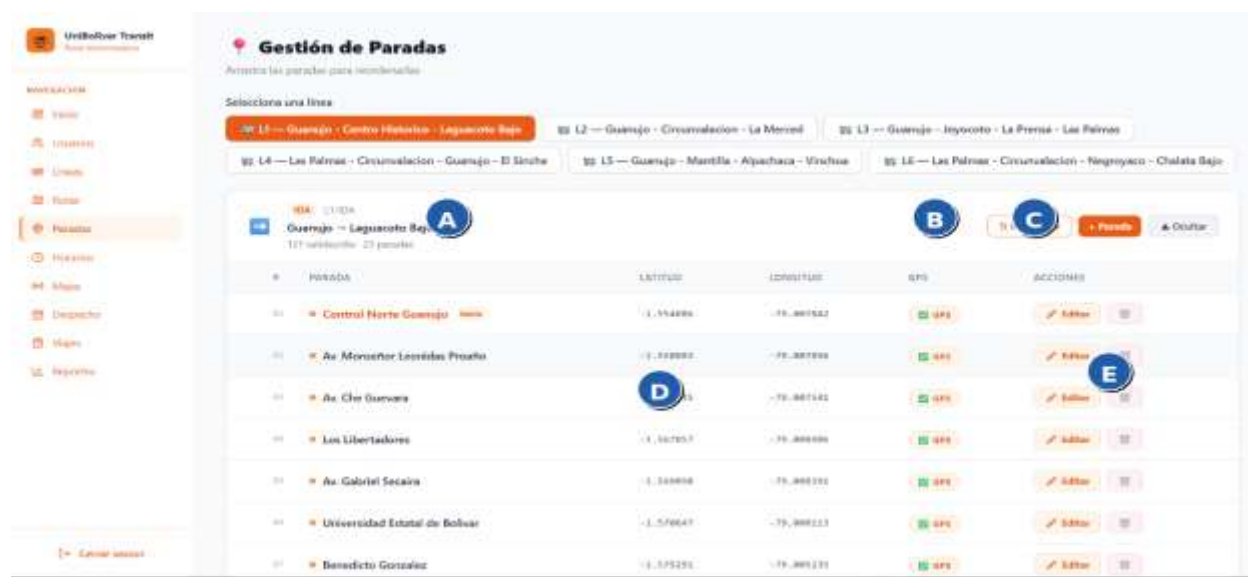
1. Seleccione primero la línea que desea consultar.
2. Identifique si trabajará con el sentido IDA o VUELTA.

3. Pulse “Ver paradas” en la tarjeta correspondiente.

11. Gestión de Paradas - Listado y acciones

Objetivo: Consultar y administrar la secuencia de paradas de un recorrido.

Ilustración 73 Gestión de Paradas - Listado y acciones



Elementos principales

- A. Información del recorrido: identifica la línea y el sentido seleccionados.
- B. Reordenar: permite cambiar la secuencia de las paradas.
- C. Parada: permite agregar una nueva parada.
- D. Tabla de paradas: muestra número, nombre, latitud, longitud y GPS.
- E. Acciones: permite editar o eliminar cada parada.

Procedimiento de uso

1. Verifique que la línea y el sentido seleccionados sean correctos.
2. Para agregar una parada, pulse “+ Parada”.
3. Para modificar datos, use “Editar” en la fila correspondiente.
4. Para cambiar el orden, pulse “Reordenar” y siga el flujo mostrado por el sistema.
5. Utilice la acción de eliminación solo cuando el registro deba retirarse.

12. Gestión de Horarios - Selección

Objetivo: Seleccionar la línea y el sentido antes de consultar o administrar sus horarios.

Ilustración 74 Gestión de Horarios - Selección

The screenshot displays the 'Gestion de Horarios' (Schedule Management) interface. It features a sidebar on the left with navigation options like 'Inicio', 'Consultas', 'Horarios', and 'Reportes'. The main area is titled 'Gestion de Horarios' and includes a 'Nuevo Horario' button. The interface is divided into two main steps: 'PASO 1 - SELECCIONA UNA LINEA' (Step 1: Select a line) and 'PASO 2 - SELECCIONA EL SENTIDO' (Step 2: Select the direction). In Step 1, several route options are listed, including 'L1 - Guarema - Centro Histórico - Lagunetas Bajo', 'L2 - ...', 'L3 - Guarema - Joyocoto - La Prensa - Las Palmas', 'L4 - Las Palmas - Circunvalación - Guarema - El Símbolo', 'L5 - Guarema - Mantilla - Alpacera - Virehes', and 'L6 - Las Palmas - Circunvalación - Neguaco - Chabela Bajo'. In Step 2, the user can select the direction: 'IDA' (Outbound) or 'VUELTA' (Return). Below these steps, a table shows the selected route 'L1-IDA: Guarema - Lagunetas Bajo' with 271 total hours. At the bottom, a grid of time slots is displayed, ranging from 06:12 to 08:14, each with a red 'X' icon indicating a selection or deletion action.

Elementos principales

- A. Paso 1: selección de línea L1 a L6.
- B. Paso 2: selección del sentido IDA o VUELTA.
- C. Área de horarios: se habilita después de completar la selección.

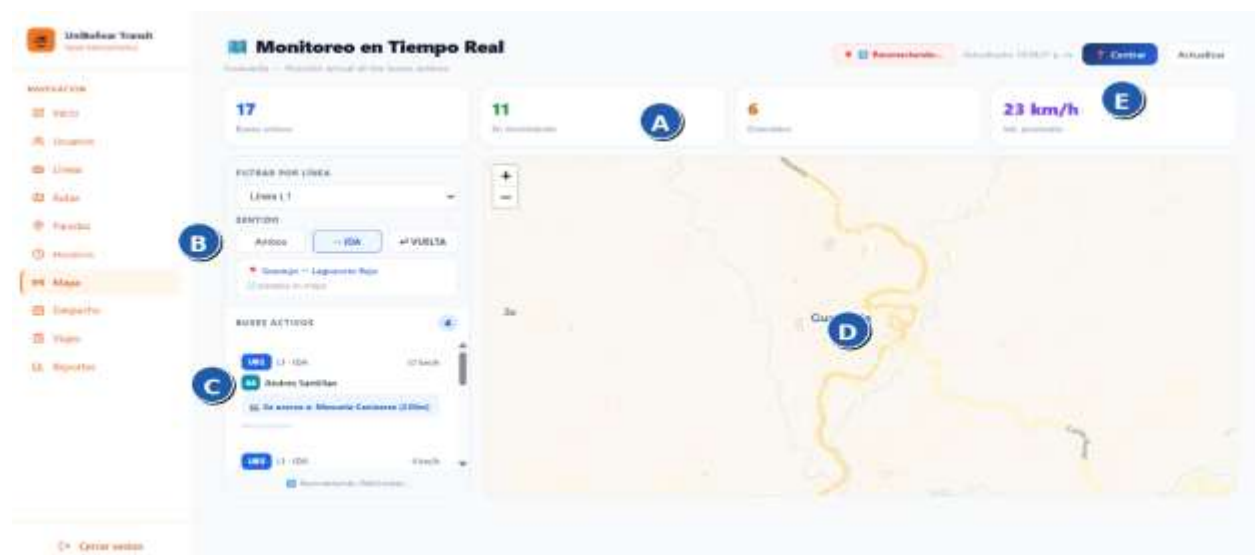
Procedimiento de uso

1. Seleccione la línea correspondiente.
2. Seleccione el sentido IDA o VUELTA.
3. Espere a que el sistema cargue los horarios asociados.

14. Monitoreo en Tiempo Real

Objetivo: Supervisar la ubicación y el estado de las unidades activas sobre el mapa.

Ilustración 75 Monitoreo en Tiempo Real



Elementos principales

- A. Indicadores superiores: buses activos, en movimiento, detenidos y velocidad promedio.
- B. Filtro por línea: permite limitar los buses mostrados.
- C. Lista de unidades: presenta las unidades activas y su información básica.
- D. Mapa: muestra los marcadores de los buses en tiempo real.
- E. Centrar/Actualizar: permite ajustar la vista o refrescar la información.

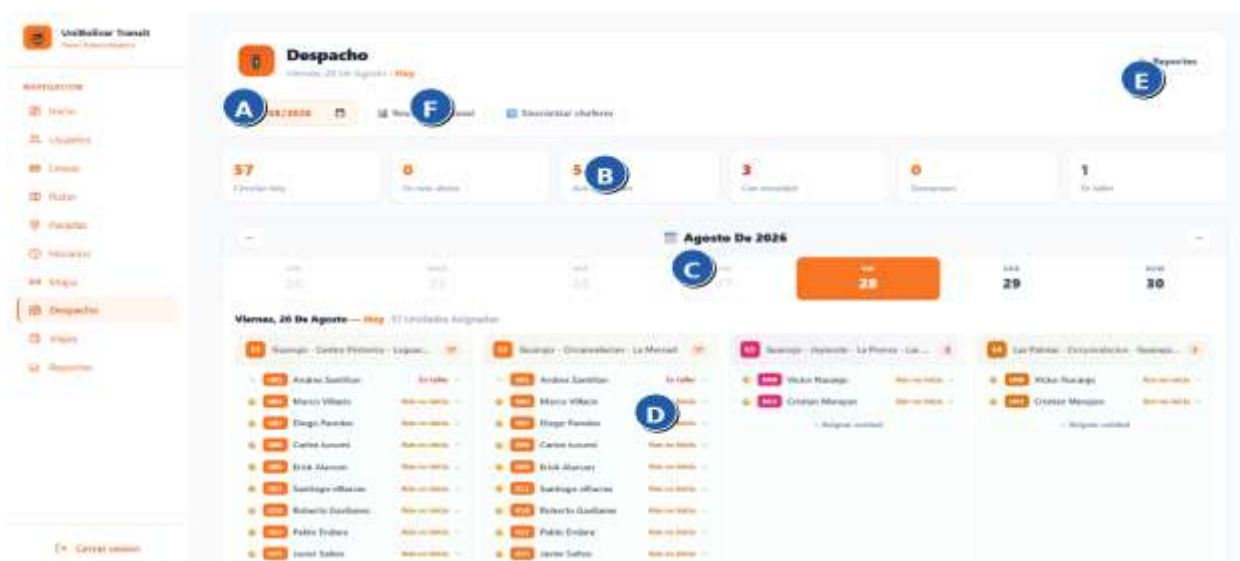
Procedimiento de uso

1. Revise los indicadores operativos de la parte superior.
2. Seleccione una línea en el filtro si desea reducir la visualización.
3. Seleccione una unidad de la lista para identificarla en el mapa.
4. Utilice “Centrar” para ajustar la vista y “Actualizar” cuando requiera refrescar la información.

15. Despacho - Programación diaria

Objetivo: Organizar la asignación diaria de choferes y unidades por línea.

Ilustración 76 Despacho - Programación diaria



Elementos principales

- A. Selector de fecha: permite consultar el día de trabajo.
- B. Resumen superior: muestra indicadores de la programación.
- C. Calendario: permite moverse entre los días disponibles.
- D. Columnas por línea: muestran las unidades y choferes asignados.

- E. Panel de reportes: presenta novedades registradas por los choferes.
- F. Sincronizar choferes/Resumen semanal: acciones de apoyo para la programación.

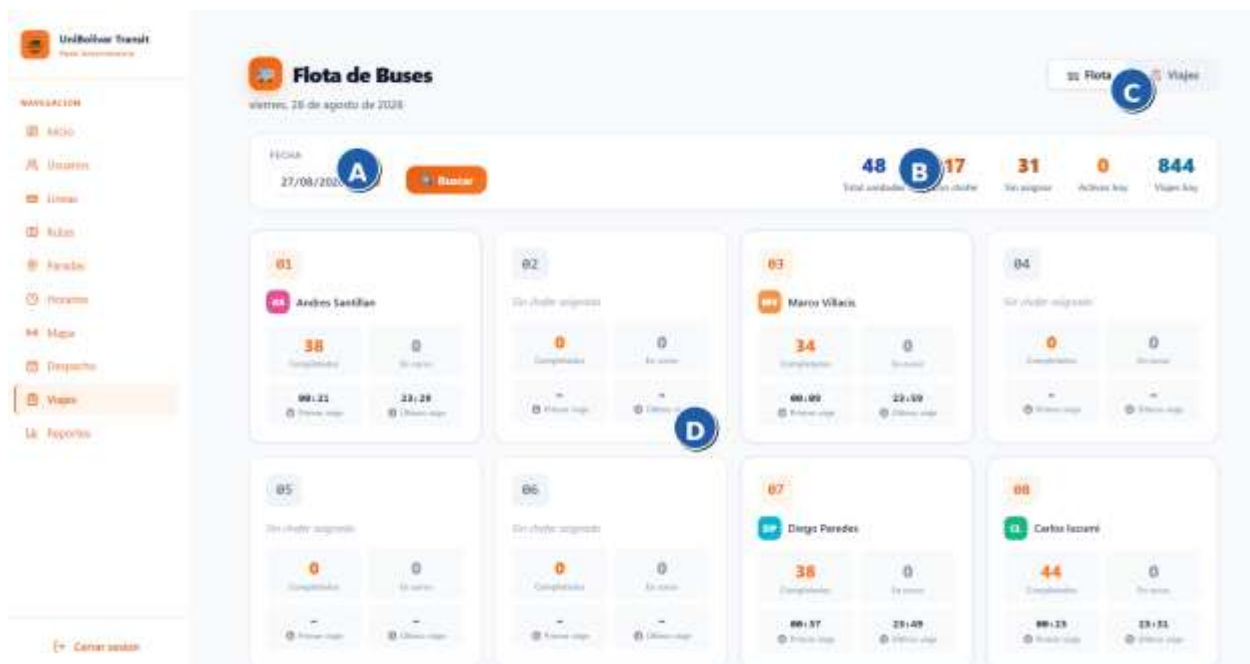
Procedimiento de uso

- A. Seleccione la fecha que desea revisar.
- B. Compruebe los indicadores de la jornada.
- C. Revise las asignaciones organizadas por línea.
- D. Consulte el panel de reportes para identificar novedades.
- E. Utilice las acciones disponibles para sincronizar o revisar la programación semanal cuando sea necesario.

17. Flota de Buses

Objetivo: Consultar el estado general de las unidades, los choferes asignados y la actividad de viajes del día.

Ilustración 77 Flota de Buses



Elementos principales

- A. Fecha y Buscar: permiten consultar la jornada seleccionada.
- B. Resumen: muestra total de unidades, con chofer, sin asignar, activas y viajes.
- C. Alternador Flota/Viajes: cambia la vista de información.
- D. Tarjetas de unidad: muestran número de bus, chofer y métricas de actividad.

Procedimiento de uso

1. Seleccione la fecha que desea consultar y pulse “Buscar”.
2. Revise los totales del resumen superior.
3. Use el alternador Flota/Viajes según la información que necesite.
4. Revise las tarjetas de cada unidad para comparar su actividad.

18. Reporte de Viajes

Objetivo: Consultar los viajes registrados aplicando filtros y revisar el detalle de cada recorrido.

Elementos principales

Ilustración 78 Reporte de Viajes

A	UNIDAD	CHOFER	RUTA	HORA SALIDA	HORA LLEGADA	DURACIÓN	VAL. MÉR.	ESTADO	DETALLE
001	0004	140 Fernando Moreta	1.2-MB126	09:39:44 a. m. 26 ago	En ruta...		0	En camino	Ver más
001	0004	06 Roberto Gavilanes	1.2-MB126	09:39:55 a. m. 26 ago	En ruta...		0	En camino	Ver más
001	0005	26 Javier Salazar	1.2-MB	09:39:48 a. m. 26 ago	En ruta...		0	En camino	Ver más
001	0005	08 Pablo Endara	1.2-MB126	09:39:52 a. m. 26 ago	En ruta...		0	En camino	Ver más
001	0005	16 Erick Alarcon	1.2-MB126	09:32:55 a. m. 26 ago	En ruta...		0	En camino	Ver más

- A. Filtros: Fecha, Estado, Línea, Sentido y Chofer.
- B. Buscar/Limpiar: ejecutan o reinician la consulta.
- C. Tabla: muestra unidad, chofer, ruta, salida, llegada, duración, velocidad máxima y estado.
- D. Ver más: abre el detalle disponible del viaje seleccionado.

Procedimiento de uso

1. Defina uno o varios filtros de búsqueda.
2. Pulse “Buscar” para aplicar los criterios.
3. Revise los resultados de la tabla.
4. Pulse “Ver más” cuando requiera consultar el detalle de un viaje.
5. Use “Limpiar” para restablecer los filtros.

19. Reportes por Unidad

Objetivo: Consultar el desempeño de una unidad dentro de un rango de fechas y generar el reporte disponible.

Ilustración 79 Reportes por Unidad



Elementos principales

- A. Rango de fechas: delimita el período a consultar.
- B. Buscador de unidad: ayuda a localizar un bus específico.
- C. Cuadrícula de unidades: permite seleccionar la unidad de interés.
- D. Resumen inferior: muestra indicadores de viajes y puntualidad.
- E. Ver reporte: abre el detalle de la unidad seleccionada.
- F. Descargar PDF: genera el reporte en el formato disponible.

Procedimiento de uso

1. Seleccione el rango de fechas.
2. Localice la unidad mediante el buscador o la cuadrícula.
3. Haga clic sobre la unidad para seleccionarla.
4. Revise los indicadores de viajes y el índice de puntualidad.
5. Pulse “Ver reporte” para consultar el detalle o “Descargar PDF” para generar el documento.

ANEXO 6

Manual de Usuario - Aplicación Móvil del Pasajero

Objetivo del manual

Este manual describe el uso de las principales interfaces de la aplicación móvil del pasajero de UniBolívar Transit. Las pantallas se presentan en el mismo orden del prototipo proporcionado y explican su finalidad, los elementos principales y el procedimiento de uso.

Convención de lectura

En cada pantalla se emplean letras (A, B, C, etc.) para identificar grupos funcionales en la explicación. Estas letras corresponden a los marcadores incorporados en las capturas de interfaz.

Orden de interfaces

Pantalla de bienvenida y carga (Splash)

Mapa principal (pestaña Rutas)

Lista de líneas / Rutas

Paradas y tiempos de llegada

Horarios por línea, sentido y día

Mis alertas activas

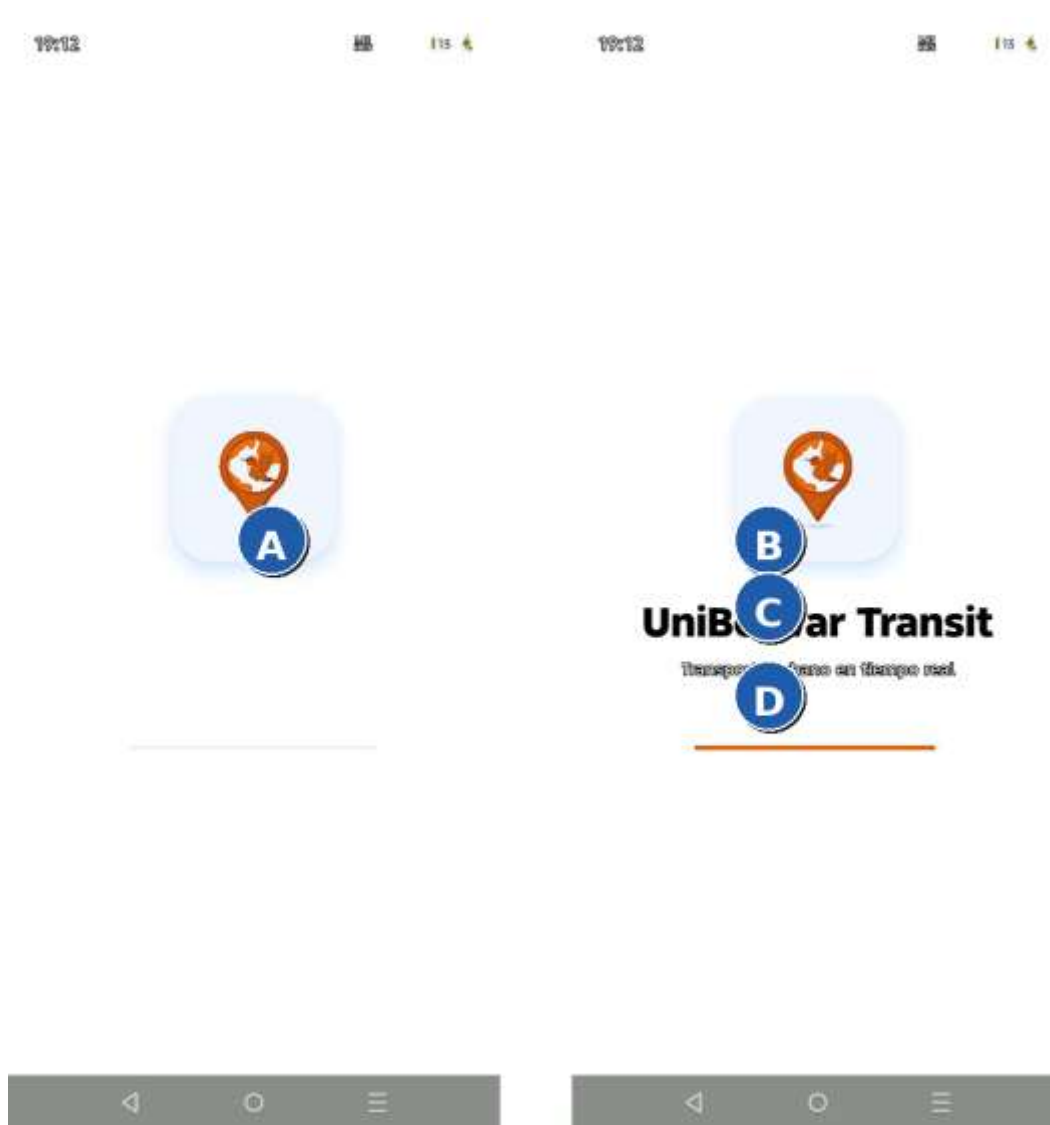
Alertas activas con contexto en el mapa

Mapa completo del recorrido

1. Bienvenida y carga (Splash)

Objetivo: Presentar la identidad de UniBolívar Transit mientras la aplicación realiza la carga inicial antes de mostrar la interfaz principal.

Ilustración 80 Bienvenida y carga (Splash)



Elementos principales

- A. Logotipo de UniBolívar Transit. En la primera vista aparece como elemento principal y en la segunda se mantiene sobre la identificación de la aplicación.
- B. Nombre de la aplicación: “UniBolívar Transit”.

- C. Subtítulo “Transporte urbano en tiempo real”, que identifica el propósito general de la aplicación.
- D. Barra de progreso utilizada durante la carga inicial.

Procedimiento de uso

1. Abra la aplicación desde el dispositivo móvil.
2. Espere mientras se visualiza el logotipo y la barra de progreso.
3. No es necesario pulsar ningún control; al finalizar la carga, la aplicación continúa hacia la pantalla principal.

2. Mapa principal (pestaña Rutas)

Objetivo: Permitir al pasajero visualizar el contexto geográfico del recorrido, su ubicación y acceder a las funciones principales de la aplicación.

Ilustración 81 Mapa principal (pestaña Rutas)



Elementos principales

- A. Mapa principal en modo oscuro, donde se visualizan vías, referencias y elementos asociados al transporte.
- B. Indicador de buses en ruta, que muestra la presencia de unidades en servicio.
- C. Botón de recentrado del mapa para volver a ubicar la vista en la posición de referencia.
- D. Pestaña Rutas.
- E. Pestaña Paradas.
- F. Pestaña Horarios.
- G. Pestaña Alertas.

Procedimiento de uso

1. Revise el mapa para identificar el área mostrada y los elementos visibles del servicio.
2. Utilice el botón de recentrado cuando desee regresar a la ubicación de referencia.
3. Use la barra inferior para cambiar entre Rutas, Paradas, Horarios y Alertas.

3. Lista de líneas / Rutas

Objetivo: Permitir al pasajero revisar las líneas disponibles, reconocer su estado de servicio y seleccionar el recorrido que desea consultar.

Ilustración 82 Lista de líneas / Rutas



Objetivo: Permitir al pasajero revisar las líneas disponibles, reconocer su estado de servicio y seleccionar el recorrido que desea consultar.

Elementos principales

- A. Barra de navegación principal de la aplicación.
- B. Resumen de líneas y direcciones disponibles.
- C. Tarjeta de una línea, identificada por color y nombre del recorrido.
- D. Datos secundarios de la línea, como número de paradas y salidas.
- E. Indicadores generales de unidades o servicio en vivo.

Procedimiento de uso

1. Ingrese a la pestaña Rutas desde la navegación inferior.
2. Desplácese verticalmente para revisar las líneas disponibles.
3. Identifique la línea deseada por su código, color y nombre del recorrido.
4. Toque la tarjeta o su control de expansión para consultar la información asociada a esa línea.

4. Paradas y tiempos de llegada

Objetivo: Mostrar la secuencia de paradas de la ruta seleccionada, la ubicación del pasajero y la información disponible sobre el estado del servicio.

Ilustración 83 Paradas y tiempos de llegada



Elementos principales

- A. Ruta seleccionada y distancia de referencia mostrada en la cabecera.
- B. Barra de navegación con la pestaña Paradas activa.
- C. Sentido del recorrido y tramo seleccionado.
- D. Aviso de estado del servicio y tarjeta de parada más cercana.
- E. Secuencia vertical de paradas del recorrido.
- F. Estado de la unidad o referencia de bus respecto de la parada.

Procedimiento de uso

1. Seleccione previamente una línea desde la pestaña Rutas.
2. Abra la pestaña Paradas.
3. Revise el sentido del recorrido y la parada más cercana indicada por la aplicación.
4. Desplácese por la lista para consultar las siguientes paradas y su estado.
5. Observe los avisos del sistema cuando no existan buses en servicio o cuando una unidad se encuentre detenida.

5. Horarios por línea, sentido y día

Objetivo: Permitir al pasajero consultar la programación del transporte filtrando la información por línea, sentido y tipo de día.

Ilustración 84 Horarios por línea, sentido y día



Elementos principales

- A. Ruta o línea seleccionada en la parte superior.
- B. Pestaña Horarios activa dentro de la navegación principal.
- C. Selector de línea.
- D. Selector de sentido IDA o VUELTA.
- E. Selector de día: Lun-Vie, Sábado o Domingo.
- F. Listado de franjas horarias y frecuencia de salida correspondiente.

Procedimiento de uso

1. Abra la pestaña Horarios.
2. Seleccione la línea que desea consultar.
3. Elija el sentido de viaje: IDA o VUELTA.
4. Seleccione el tipo de día correspondiente.
5. Revise las franjas horarias y la frecuencia mostrada para la combinación elegida.

6. Mis alertas activas

Objetivo: Permitir al pasajero consultar las alertas que tiene activas para distintas paradas y cancelar aquellas que ya no necesite.

Ilustración 85 Mis alertas activas



Elementos principales

- A. Ruta seleccionada mostrada en la cabecera.
- B. Barra de navegación con la pestaña Alertas activa.
- C. Contador total de alertas activas.
- D. Tarjeta individual de una parada con alerta.
- E. Control de cierre o eliminación de la alerta correspondiente.

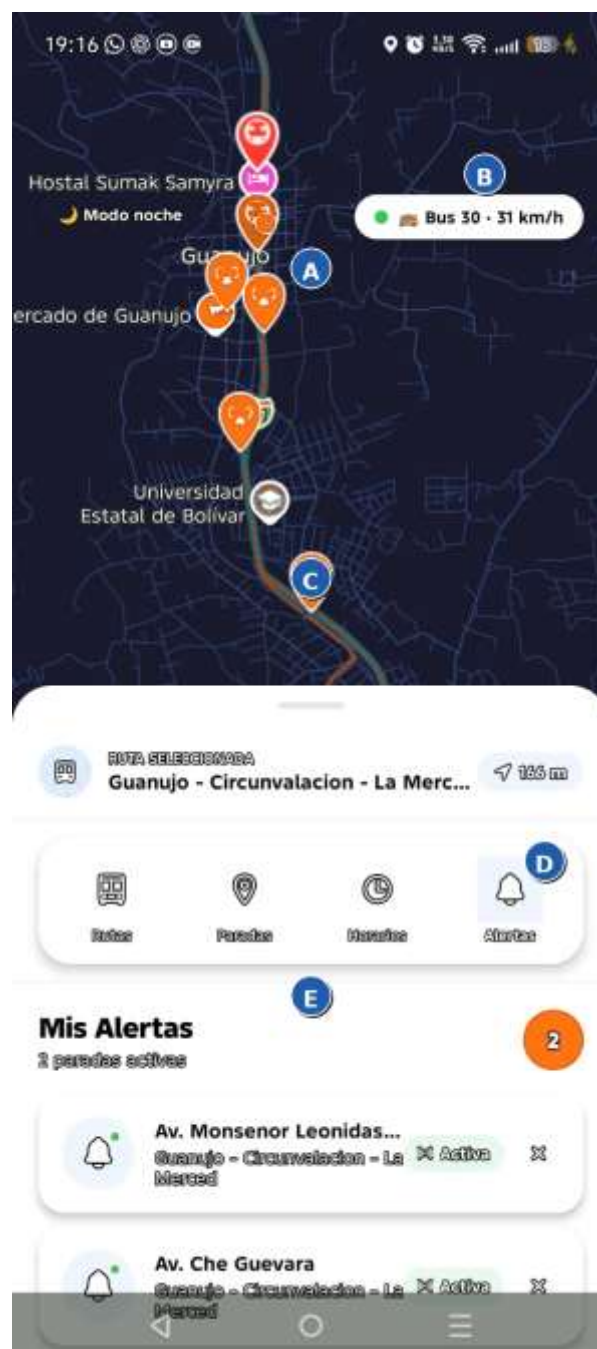
Procedimiento de uso

1. Ingrese a la pestaña Alertas.
2. Revise el número de alertas activas indicado en el contador.
3. Consulte cada tarjeta para identificar la parada y el recorrido asociado.
4. Si desea cancelar una alerta, utilice el control de cierre ubicado en la tarjeta correspondiente.

7. Alertas activas con contexto en el mapa

Objetivo: Relacionar las alertas activas con la ubicación de las paradas y el recorrido visualizado en el mapa.

Ilustración 86 Alertas activas con contexto en el mapa



Elementos principales

- A. Mapa del recorrido con marcadores de paradas.
- B. Chip de la unidad en servicio con información visible de operación.
- C. Panel de la ruta seleccionada.
- D. Contador de alertas activas.
- E. Listado de alertas asociadas a paradas del recorrido.

Procedimiento de uso

1. Abra la pestaña Alertas.
2. Observe en la parte superior el mapa y la distribución de las paradas del recorrido.
3. Revise el panel de la ruta seleccionada y el contador de alertas.
4. Desplácese por la lista de alertas para relacionar cada parada con el contexto mostrado en el mapa.

8. Mapa completo del recorrido

Objetivo: Ofrecer una vista amplia del trayecto seleccionado para identificar visualmente las paradas y la extensión general del recorrido.

Ilustración 87 Mapa completo del recorrido



Elementos principales

- A. Mapa completo del trayecto con trazado vial y recorrido visible.
- B. Marcadores de paradas distribuidos a lo largo del recorrido.
- C. Botón de recentrado del mapa.
- D. Panel inferior que mantiene identificada la ruta seleccionada y el acceso a la navegación.

Procedimiento de uso

1. Seleccione una ruta desde la pestaña Rutas.
2. Visualice el recorrido completo y la distribución de sus paradas en el mapa.
3. Desplace o amplíe el mapa con los gestos habituales del dispositivo cuando necesite revisar un sector específico.
4. Use el botón de recentrado para recuperar la vista de referencia.
5. Utilice la navegación inferior para continuar hacia Paradas, Horarios o Alertas.

Manual de Usuario - Aplicación Móvil del Chofer

Objetivo del manual

Este manual describe el uso de las principales interfaces de la aplicación móvil del chofer de UniBolívar Transit. Las instrucciones se presentan en el mismo orden de las pantallas proporcionadas, con el propósito de orientar al conductor durante el acceso, selección de ruta, operación del servicio, reporte del estado de la unidad y consulta del historial de recorridos.

Convención de lectura

En cada pantalla se emplean letras (A, B, C, etc.) para identificar grupos funcionales en la explicación. Estas letras corresponden a los marcadores colocados directamente sobre las capturas cuando es necesario señalar un control o una zona específica de la interfaz.

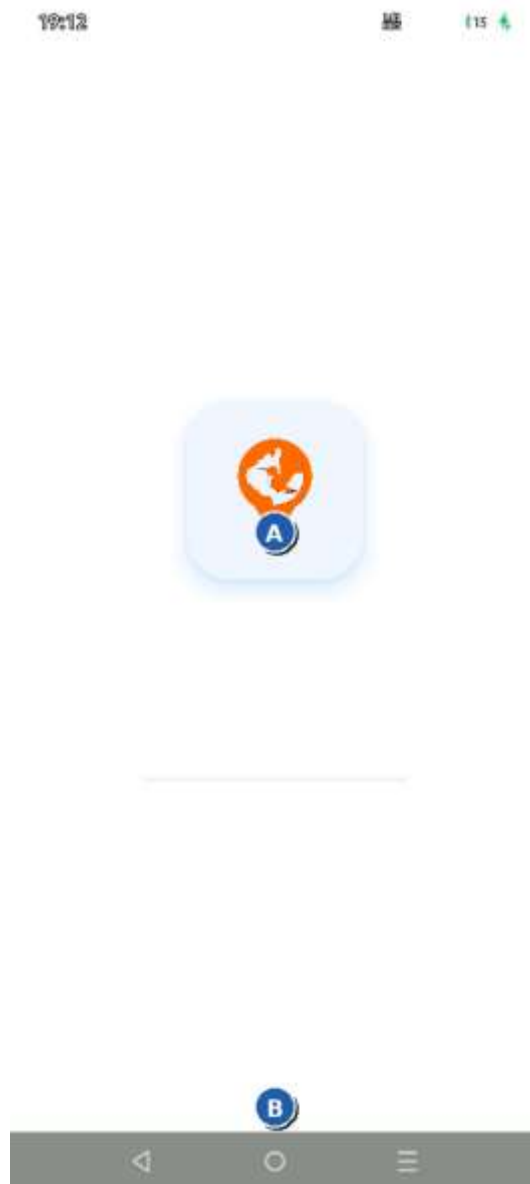
Orden de interfaces

1. Pantalla de bienvenida y carga (Splash)
2. Inicio de sesión
3. Selección de ruta y sentido
4. Pantalla «En servicio»
5. Estado de la unidad
6. Mi historial

1. Bienvenida y carga (Splash)

Objetivo: Presentar la identidad de UniBolívar Transit mientras la aplicación realiza la carga inicial antes de mostrar la pantalla de acceso del conductor.

Ilustración 88 Bienvenida y carga (Splash)



Elementos principales

- A. Logotipo de UniBolivar Transit. Identifica visualmente la aplicación durante el proceso de inicio.
- B. Barra de navegación del sistema Android. Corresponde a los controles propios del dispositivo móvil.

Procedimiento de uso

1. Abra la aplicación UniBolivar Transit desde el teléfono.
2. Espere mientras se muestra el logotipo en el centro de la pantalla.
3. No es necesario pulsar ningún control; al finalizar la carga, la aplicación continúa automáticamente hacia el inicio de sesión.

2. Inicio de sesión

Objetivo: Permitir que el chofer ingrese de forma segura a la aplicación mediante las credenciales asignadas por la cooperativa.

Ilustración 89 Inicio de sesión



Elementos principales

- A. Identificación de UniBolívar Transit y del Panel del Chofer.
- B. Campo Correo electrónico. Permite ingresar el correo institucional asignado al conductor.
- C. Campo Contraseña. Permite escribir la clave de acceso del chofer.
- D. Control de visualización de contraseña. Permite mostrar u ocultar los caracteres ingresados.
- E. Botón Ingresar. Valida las credenciales y permite acceder a la aplicación.

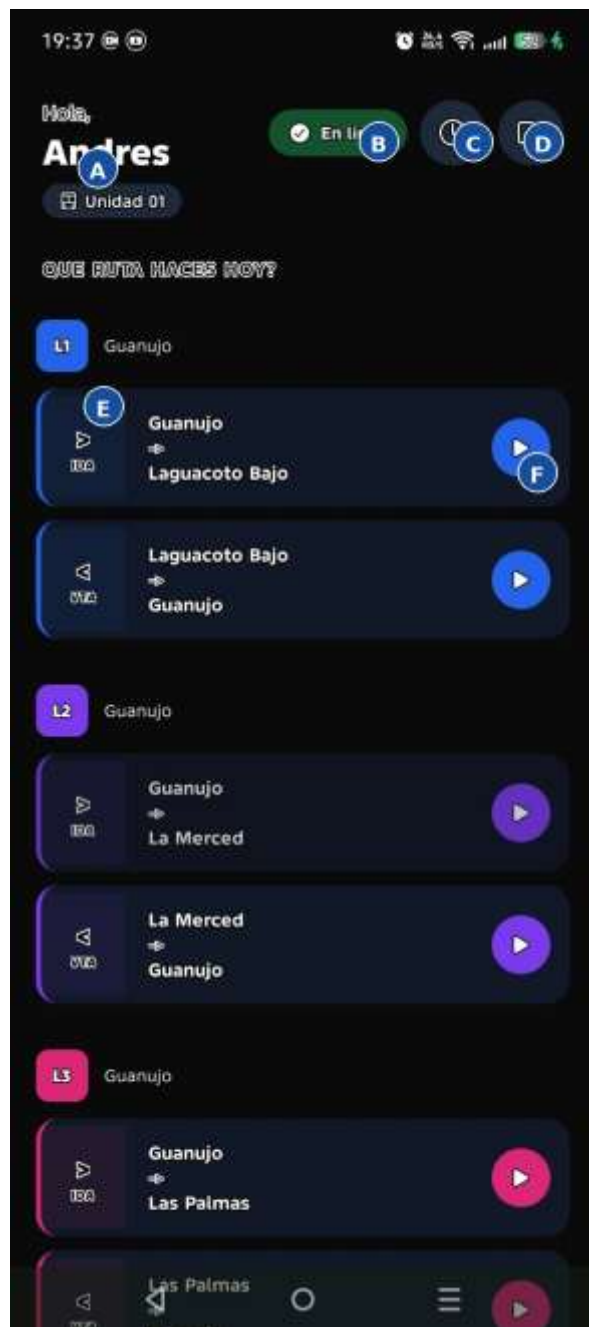
Procedimiento de uso

1. Ingrese el correo electrónico entregado por el administrador.
2. Escriba la contraseña correspondiente.
3. Si necesita comprobar la contraseña, utilice el ícono de visualización señalado con la letra D.
4. Pulse Ingresar.
5. Si las credenciales son correctas, la aplicación muestra la pantalla de selección de ruta y sentido.

3. Selección de ruta y sentido

Objetivo: Permitir al chofer elegir la línea y el sentido de recorrido que realizará antes de iniciar el servicio.

Ilustración 90 Selección de ruta y sentido



Elementos principales

- A. Datos del chofer y unidad asignada. Muestran el nombre del conductor y el número de unidad.
- B. Estado En línea. Indica que la aplicación mantiene conexión con el sistema.
- C. Acceso al historial. Abre la consulta de recorridos registrados.
- D. Opción de salida. Permite cerrar la sesión del chofer.
- E. Listado de líneas y sentidos. Organiza las rutas disponibles por línea, origen y destino.
- F. Botón de inicio de ruta. Permite seleccionar el sentido correspondiente e iniciar el recorrido.

Procedimiento de uso

1. Verifique que su nombre y unidad asignada sean correctos.
2. Revise que el estado indique «En línea».
3. Desplácese por el listado hasta localizar la línea que debe operar.
4. Compruebe el origen y el destino para seleccionar correctamente IDA o VUELTA.
5. Pulse el botón de inicio ubicado a la derecha de la ruta seleccionada.
6. La aplicación pasa a la pantalla En servicio y comienza el registro del recorrido.

4. Pantalla En servicio

Objetivo: Mostrar al chofer el estado del recorrido activo, el tiempo transcurrido y las métricas principales mientras la aplicación mantiene el servicio en ejecución.

Ilustración 91 Pantalla En servicio



Elementos principales

- A. Estado EN SERVICIO. Confirma que existe un recorrido activo.
- B. Estado de conexión En línea. Indica comunicación activa con el sistema.
- C. Tarjeta de ruta activa. Muestra la línea, el sentido y el recorrido actualmente seleccionado.
- D. Cronómetro de servicio. Registra el tiempo transcurrido desde el inicio del recorrido.
- E. Panel de métricas. Presenta información como distancia recorrida y velocidad máxima.
- F. Botón Finalizar turno. Permite cerrar el recorrido cuando el servicio haya concluido.

Procedimiento de uso

1. Confirme que el estado superior indique EN SERVICIO y En línea.
2. Revise que la tarjeta de ruta corresponda al recorrido seleccionado.
3. Durante el viaje, consulte el cronómetro y las métricas únicamente cuando sea seguro hacerlo.
4. Mantenga la aplicación activa durante el servicio para conservar el registro operativo.
5. Cuando termine el recorrido, pulse Finalizar turno y confirme la acción si la aplicación lo solicita.

5. Estado de la unidad

Objetivo: Permitir al chofer informar al sistema el estado operativo de la unidad y registrar una observación cuando sea necesario.

Ilustración 92 Estado de la unidad

19:38

EN SERVICIO En línea

L1 → ID2
Guanaju - Lagunacoto Bajo

Estado de la unidad

☒ **En línea**
Operativo y listo para trabajar

☐ **En mantenimiento**
El bus necesita revisión o está en el taller

☒ **Avería**
No puedo continuar el recorrido

Me choque |

Confirmar

Elementos principales

- A. Título Estado de la unidad» e identificador de la unidad seleccionada.
- B. Opción En línea. Indica que la unidad se encuentra operativa y disponible para trabajar.
- C. Opción En mantenimiento. Se utiliza cuando la unidad requiere revisión o permanece en taller.
- D. Opción Avería. Registra una falla que impide continuar normalmente con el servicio.
- E. Campo de observación. Permite escribir una breve descripción relacionada con el estado reportado.
- F. Botón Confirmar. Guarda el estado seleccionado y la observación registrada.

Procedimiento de uso

1. Abra la opción de estado de la unidad desde la pantalla de servicio.
2. Seleccione En línea, En mantenimiento o Avería, según corresponda.
3. Si es necesario, escriba una observación breve explicando la situación de la unidad.
4. Revise que la opción seleccionada sea la correcta.
5. Pulse «Confirmar» para registrar el cambio de estado.

1. Mi historial

Objetivo: Permitir al chofer consultar los recorridos registrados, revisar su estado y filtrar la información por fecha o período disponible en la aplicación.

Ilustración 93 Mi historial



Elementos principales

- A. Botón de regreso. Permite volver a la pantalla anterior.
- B. Navegación por mes. Permite cambiar el período consultado.
- C. Resumen mensual. Presenta el total de recorridos, completados y en curso.
- D. Filtros por fecha. Facilitan la consulta de recorridos correspondientes a días específicos.
- E. Tarjetas de viaje. Muestran línea, sentido, horario, duración y estado de cada recorrido.

Procedimiento de uso

1. Ingrese al historial desde el acceso correspondiente en la pantalla principal del chofer.
2. Utilice los controles del mes para seleccionar el período que desea consultar.
3. Revise el resumen mensual para conocer el total de recorridos y sus estados.
4. Use los filtros de fecha para limitar la información mostrada.
5. Desplácese por la lista y consulte los datos de cada viaje.
6. Pulse el botón de regreso cuando desee volver a la pantalla anterior

Anexo 6.

MANUAL DE DESCARGA E INSTALACIÓN EN TELÉFONO

UniBolívar Transit

App Pasajero y App Chofer

Guía práctica para la descarga, instalación y primera habilitación de permisos en dispositivos Android.

Ilustración 94 Pantalla principal del portal web de UniBolívar Transit para seleccionar la App Pasajero o la App Chofer.



1. Objetivo del manual

Este manual describe el procedimiento para descargar e instalar la aplicación UniBolívar Transit en un teléfono Android, tanto en su versión para pasajeros como en su versión para choferes, utilizando las evidencias visuales proporcionadas.

2. Requisitos previos

- Disponer de un teléfono Android con acceso a Internet.
- Tener habilitado un navegador web, por ejemplo, Google Chrome.
- Contar con espacio de almacenamiento suficiente para instalar la aplicación.
- Aceptar los permisos de instalación, ubicación y notificaciones cuando el sistema los solicite.

3. Ingreso al sitio oficial y selección de la aplicación

1	Buscar el sistema en Google Abrir el navegador del teléfono y escribir “UniBolívar Transit” para localizar el sitio oficial del sistema.
---	--

Ilustración 95 Resultado de búsqueda del sitio oficial de UniBolívar Transit.



Ingresar al portal web oficial

2

Seleccionar el resultado correspondiente a unibolivartransit.com. En el portal se visualizarán las opciones App Pasajero, App Chofer e Iniciar sesión.

Ilustración 96 Portal web oficial



Elegir la aplicación que se desea instalar

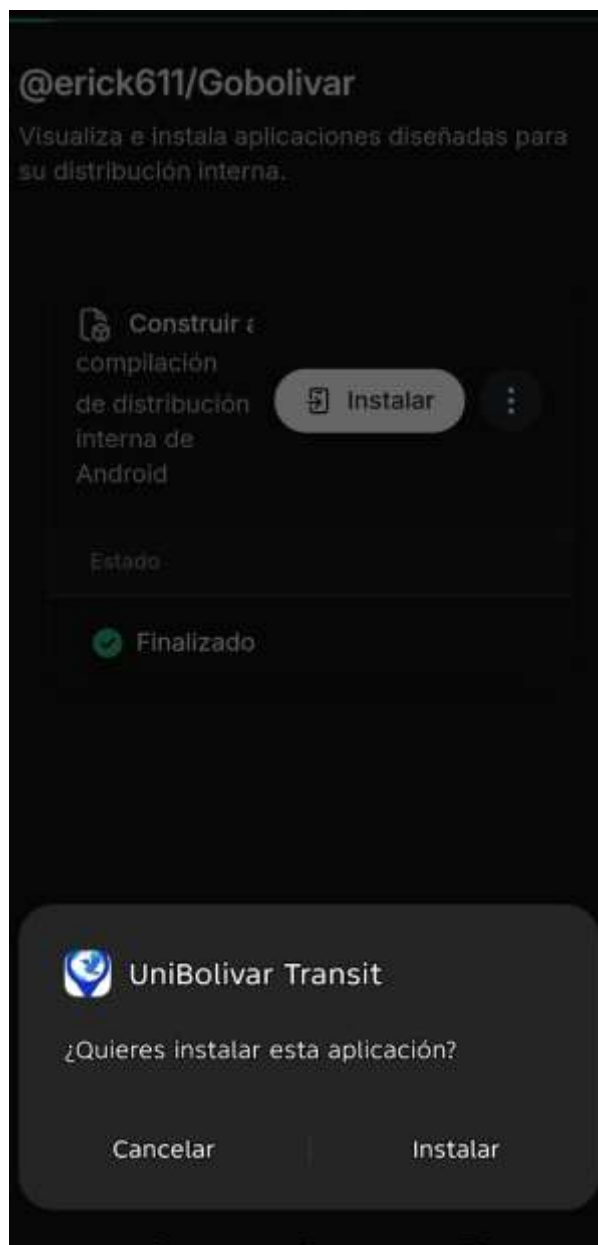
3

Desde el portal, pulsar “App Pasajero” si la instalación es para usuarios/pasajeros, o “App Chofer” si la instalación corresponde al conductor.

4. Instalación de la App Pasajero (usuario)

4	Confirmar la instalación del archivo Al seleccionar la App Pasajero, el navegador mostrará una ventana de confirmación. Se debe pulsar “Instalar” para continuar con el proceso.
---	--

Ilustración 97 Confirmación de instalación de la App Pasajero.

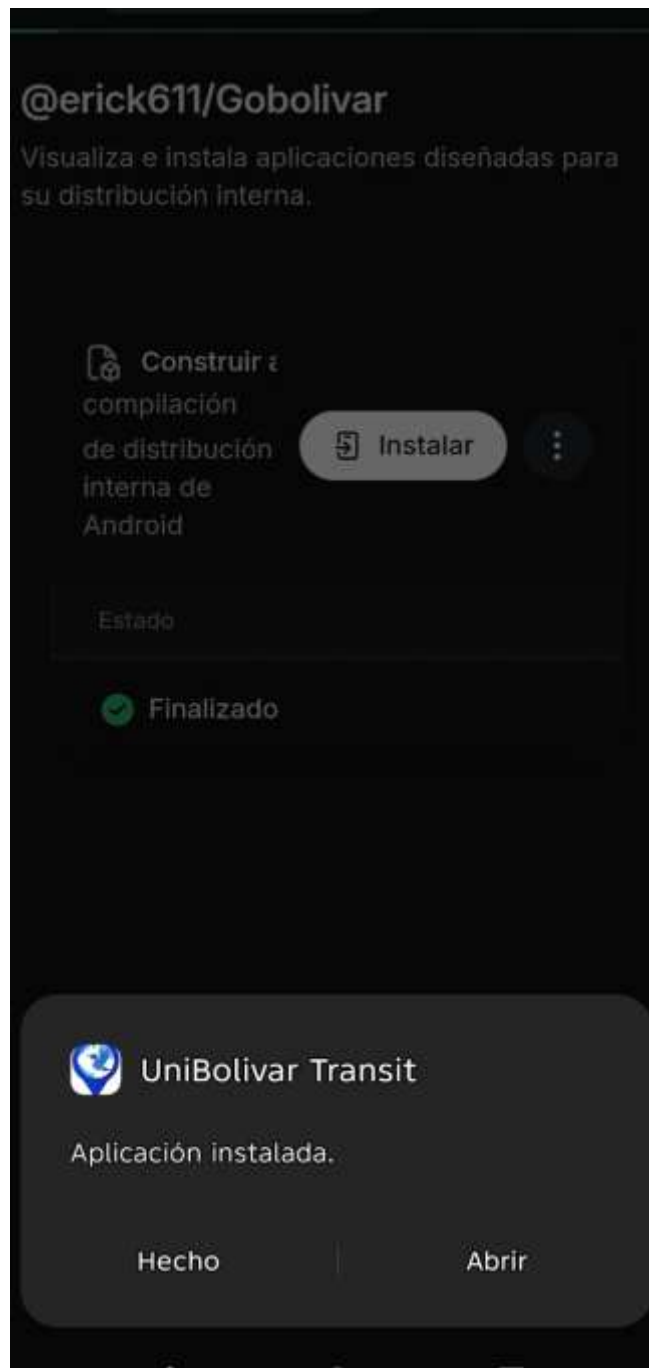


Finalizar la instalación y abrir la aplicación

5

Una vez completada la descarga e instalación, el sistema mostrará el mensaje de aplicación instalada. Se puede pulsar “Abrir” para ingresar inmediatamente.

Ilustración 98 Mensaje de instalación completada de la App Pasajero.

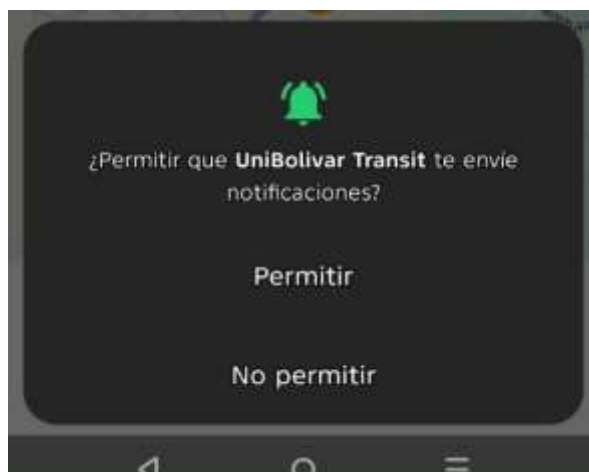


Permitir el envío de notificaciones

6

La aplicación puede solicitar permiso para enviar notificaciones. Se recomienda pulsar “Permitir” para recibir avisos del sistema, alertas y novedades del servicio.

Ilustración 99 Solicitud de permiso de notificaciones



Conceder acceso a la ubicación

7

La aplicación también puede solicitar acceso a la ubicación del dispositivo. Se recomienda seleccionar “Mientras se usa la aplicación” para permitir el funcionamiento adecuado de los mapas y la localización.

Ilustración 100 Solicitud de permiso de ubicación.



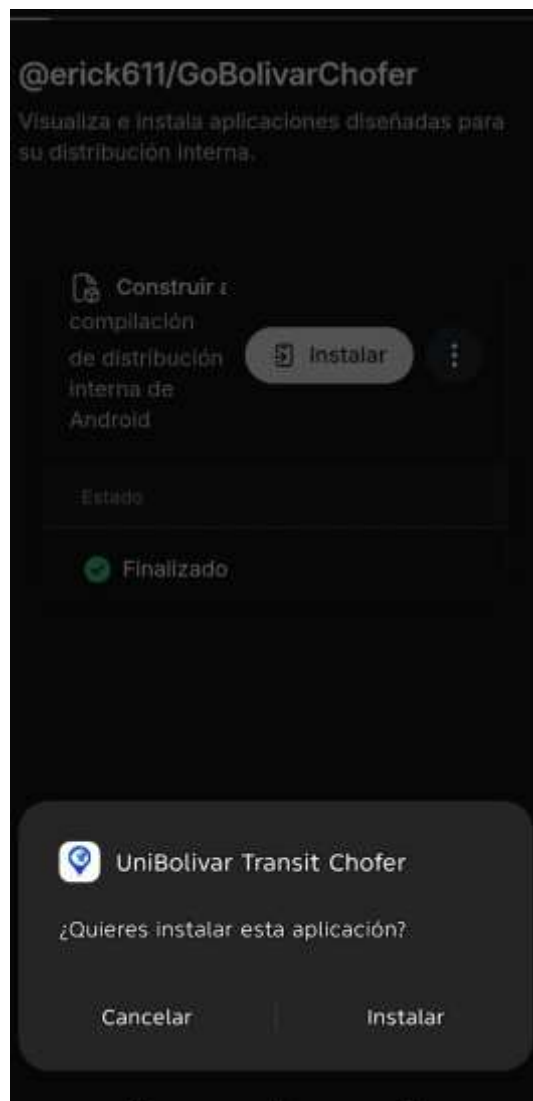
1. Instalación de la App Chofer

Confirmar la instalación del archivo

8

Después de pulsar la opción “App Chofer” en el portal web, el navegador solicitará confirmación. Se debe pulsar “Instalar” para iniciar la instalación.

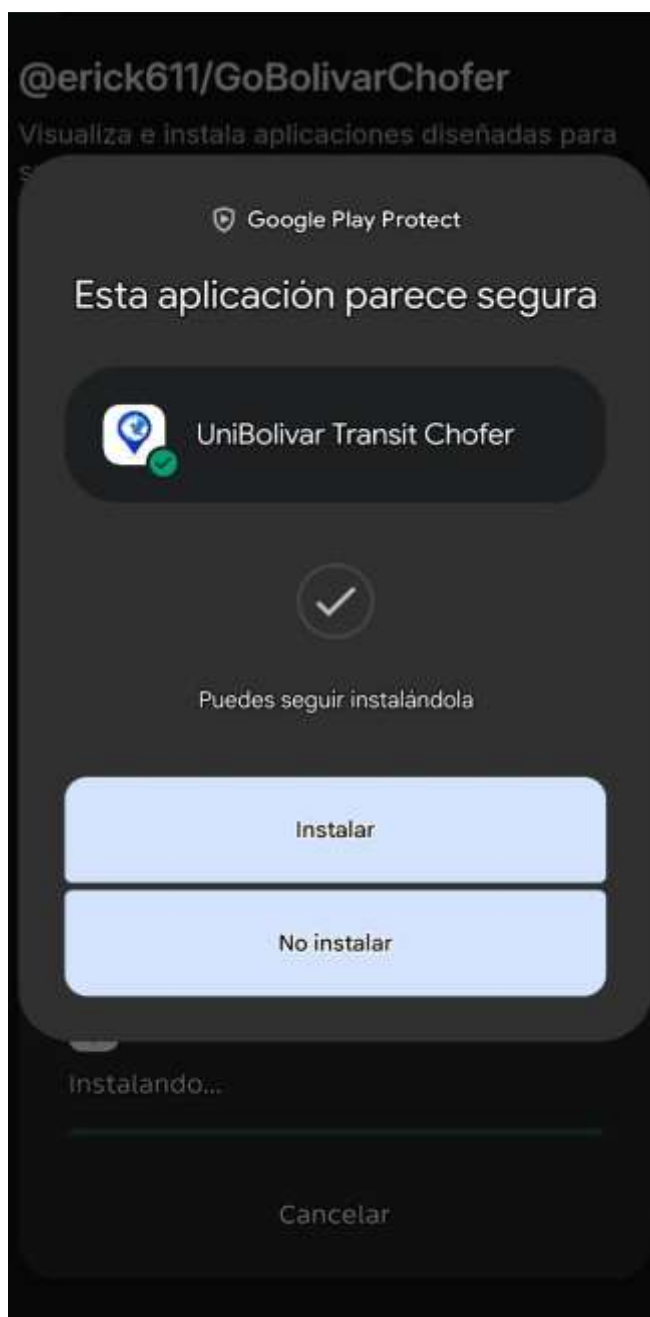
Ilustración 101 Confirmación de instalación de la App Chofer.



Aceptar la verificación de seguridad si aparece Google Play Protect

En algunos dispositivos Android puede mostrarse una pantalla de verificación de Google Play Protect. Si la aplicación se reconoce como segura, se debe pulsar “Instalar” para continuar.

Ilustración 102 Validación de seguridad de la App Chofer mediante Google Play Protect.

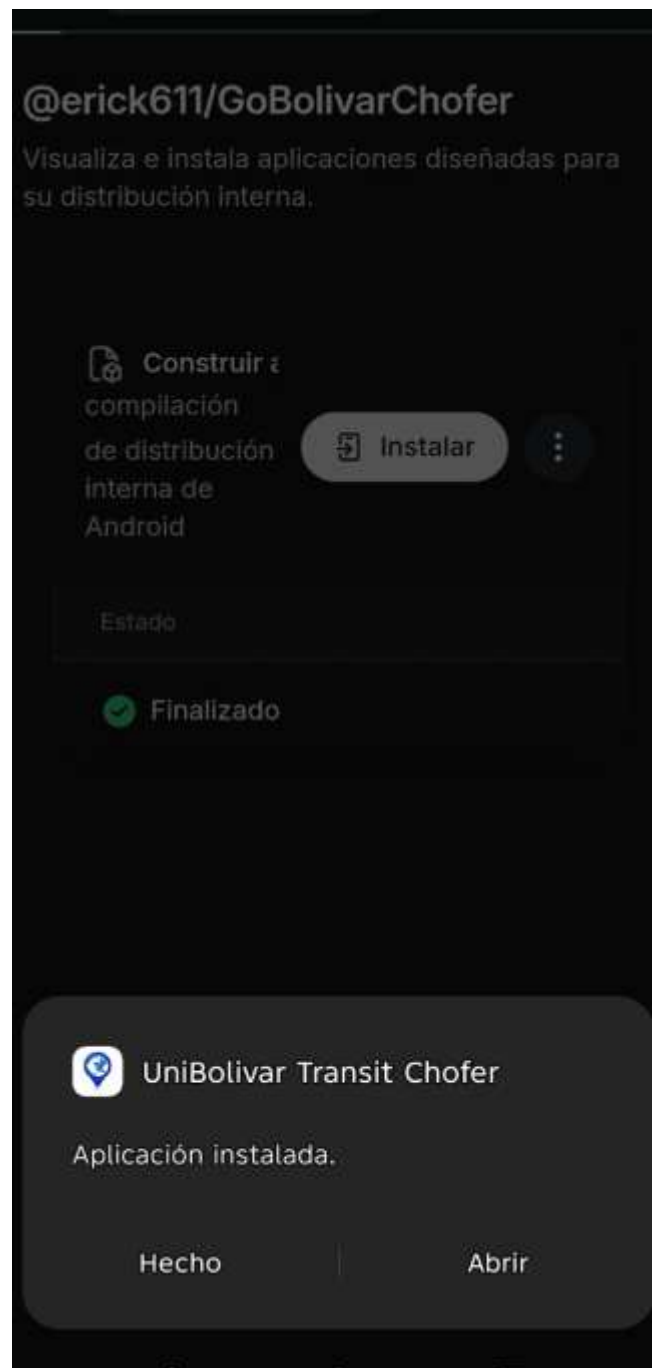


10

Completar la instalación y abrir la aplicación

Una vez concluida la instalación, el sistema mostrará el mensaje correspondiente. Se puede pulsar “Abrir” para iniciar el uso de la App Chofer.

Ilustración 103 Mensaje de instalación completada de la App Chofer.



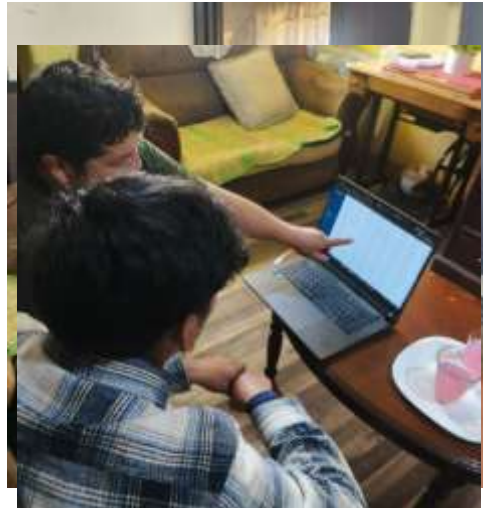
6. Recomendaciones finales

- Verificar que la conexión a Internet sea estable durante la descarga.
- Mantener habilitados los permisos de notificaciones y ubicación para aprovechar todas las funcionalidades del sistema.
- Si el navegador o el sistema Android solicitan permisos adicionales, revisarlos antes de continuar con la instalación.
- En caso de que la aplicación no se abra correctamente, cerrar el proceso y repetir la instalación desde el sitio oficial.

Anexo 7

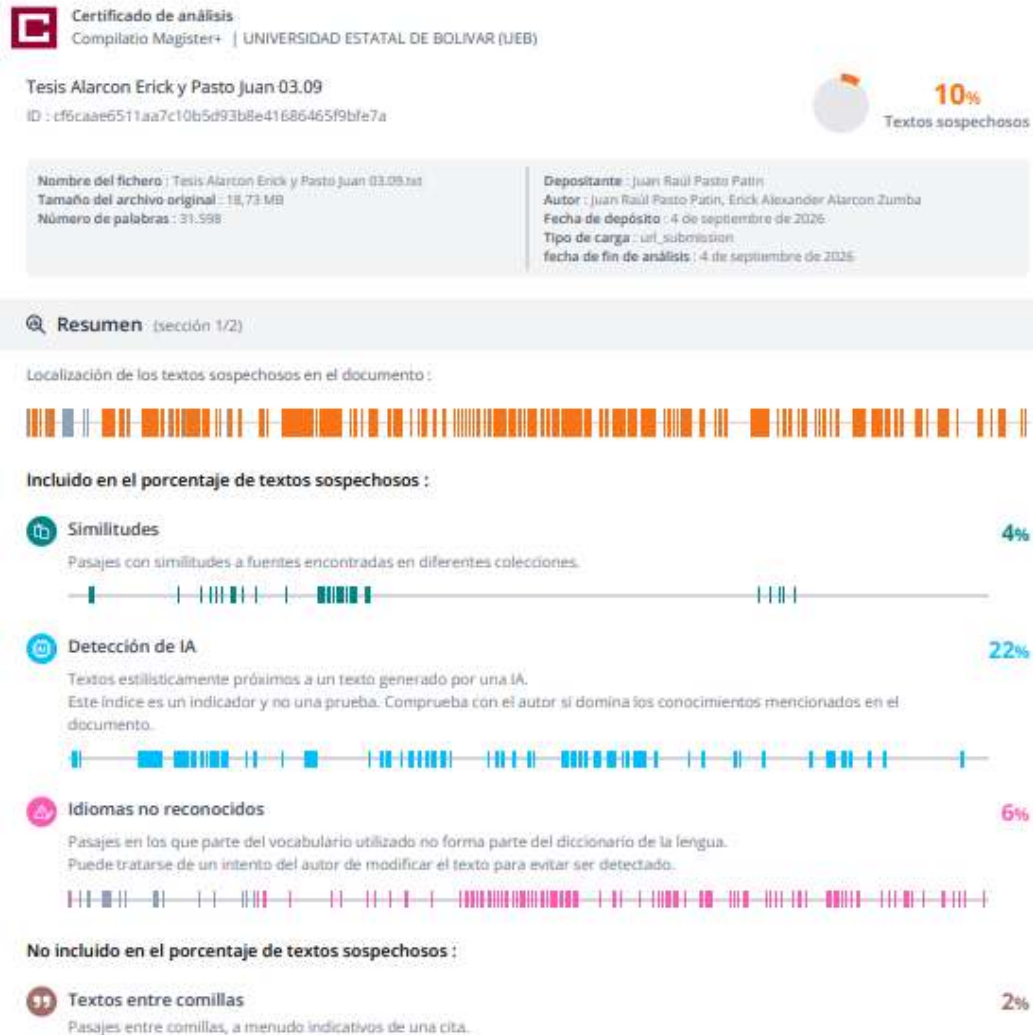
Fotografías y evidencias

Ilustración 104 fotografías de reuniones mantenidas con el gerente de la Cooperativa Universidad de Bolívar



Anexo 8

Certificado Anti plagio



Fuentes de similitudes (sección 2/2)

Similitudes

4%

Pasajes con similitudes a fuentes encontradas en diferentes colecciones.



Fuente principal detectada

N°	Descripciones	Similitudes	Ubicaciones
1	Propuesta del plan tarifario para el transporte público... dspace.esPOCH.edu.ec/bitstream/123456789/11459/3/11270104.pdf.txt	<1%	
2	dspace.ueb.edu.ec dspace.ueb.edu.ec/server/api/core/bitstreams/29489e2a-c85b-4e55-a50c-...	<1%	
3	Metodologías Ágiles Para El Desarrollo De Soft - Revisión de... www.studocu.com/co/document/universidad-icesi/planeacion-de-las-ope...	<1%	
4	repositorio.uteq.edu.ec repositorio.uteq.edu.ec/bitstream/43000/5049/1/T-UTEQ-0084.pdf	<1%	
5	Presentacion grupo 3 PPT es.slideshare.net/slideshow/presentacion-grupo-3-29084354/29084354	<1%	
6	dspace.ueb.edu.ec dspace.ueb.edu.ec/server/api/core/bitstreams/589cb552-6a85-4bd0-a5e2-...	<1%	
7	La Facultad Tributaria del Gobierno Municipal de Latacunga en... www.dspace.uce.edu.ec/handle/25000/26653	<1%	
8	www.ueb.edu.ec www.ueb.edu.ec/images/Biblioteca/Derechos_de_Autor.docx	<1%	
9	Documento de otro usuario #021eaf Vine de de otro grupo	<1%	
10	La regulación administrativa del transporte terrestre, a la luz de l... dspace.ucuenca.edu.ec/bitstream/123456789/20325/3/TESIS.pdf.txt	<1%	

Fuente mencionada (sin similitudes detectadas)

N°	Descripciones
1	https://www.asambleanacional.gob.ec/documentos/Constitucion-2008.pdf
2	https://www.imit.gob.ec/wp-content/uploads/downloads/2021/08/LOTAIP_6_Ley-Organica-de-Transporte-...
3	https://www.telecomunicaciones.gob.ec/wp-content/uploads/2021/06/Ley-Organica-de-Datos-Personales...
4	http://repositorio.undac.edu.pe/bitstream/undac/1005/4/T026_72367456_T.pdf
5	https://www.telecomunicaciones.gob.ec/wp-content/uploads/downloads/2012/11/Ley-de-Comercio-Electr...

N°		Descripciones
6		http://dspace.ups.edu.ec/handle/123456789/22102
7		https://www.researchgate.net/publication/228389461_SISTEMA_DE_POSICIONAMIENTO_GLOBAL
8		https://www.guaranda.gob.ec/newsiteCMT/feue-bolivar-y-transportistas-de-la-cooperativa-universidad-de...
9		https://repositorio.upse.edu.ec/server/api/core/bitstreams/25cdcb1b-81dd-4701-85c1-7547ba7086eb/con...
10		https://www.redalyc.org/journal/5604/560467941011/560467941011.pdf
11		http://repositorio.uta.edu.ec/handle/123456789/27421
12		https://nachoiborraies.github.io/nodejs/md/es/01b.pdf
13		https://repositorio.puce.edu.ec/server/api/core/bitstreams/b4c177c8-5ced-45af-98f8-159f1d1b86c6/conte...
14		https://dialnet.unirioja.es/descarga/articulo/6581740.pdf
15		https://repositorio.usc.edu.co/server/api/core/bitstreams/fb7c125a-ce8f-482a-932c-dc7c6a5909f3/content
16		https://doi.org/10.15665/rp.v11i2.36
17		https://upcommons.upc.edu/bitstream/handle/2117/113040/Degree_thesis.pdf?sequence=1
18		https://www.oas.org/juridico/pdfs/mesicic4_ecu_regla2.pdf
19		https://uvadoc.uva.es/bitstream/handle/10324/13538/TFG-B.796.pdf;jsessionid=4884EFEF32A8C7F44A91E...
20		https://sga.unemi.edu.ec/media/recursosotema/Documento_20201120171118.pdf
21		https://github.com/alarconerick612/unibolivar-backend
22		https://github.com/alarconerick612/unibolivar-admin
23		https://github.com/alarconerick612/GoBolivar
24		https://github.com/alarconerick612/GoBolivarChofer
25		https://unibolivartransit.com