

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

Tema: “Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público ”

Trabajo de Integración Curricular previo a la obtención del
título de Ingenieros en Ciencias de la Computación

AUTOR: Pascal Chamba Nury Gabriela

Rosero Narváez Cristian Paul

TUTOR: Ing. Del Hierro Mosquera Milton Gabriel, MSc.

Tulcán, 2026.

CERTIFICADO DEL TUTOR

Certifico que los estudiantes Pascal Chamba Nury Gabriela y Rosero Narváez Cristian Paul con el número de cédula 0402174965 y 0401770342 respectivamente han desarrollado el Trabajo de Integración Curricular: "Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público"

Este trabajo se sujeta a las normas y metodología dispuesta en el Reglamento de la Unidad de Integración Curricular, Titulación e Incorporación de la UPEC, por lo tanto, autorizo la presentación de la sustentación para la calificación respectiva

Ing. Del Hierro Mosquera Milton Gabriel, MSc.

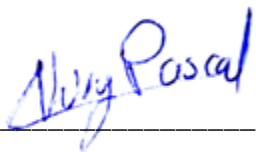
TUTOR

Tulcán, abril de 2026

AUTORÍA DE TRABAJO

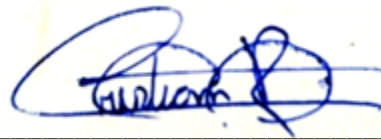
El presente Trabajo de Integración Curricular constituye un requisito previo para la obtención del título de Ingeniero en la Carrera de computación de la Facultad de Industrias Agropecuarias y Ciencias Ambientales

Nosotros, Pascal Chamba Nury Gabriela y Rosero Narváez Cristian Paul con cédula de identidad número 0402174965 y 0401770342 respectivamente declaramos que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que hemos llegado son de nuestra absoluta responsabilidad.



Pascal Chamba Nury Gabriela

AUTORA



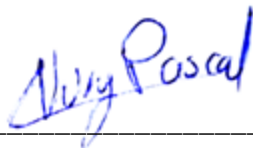
Rosero Narváez Cristian Paul

AUTOR

Tulcán, abril de 2026

ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Nosotros Pascal Chamba Nury Gabriela y Rosero Narvárez Cristian Paul declaramos ser autor de los criterios emitidos en el Trabajo de Integración Curricular: "Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público" y se exime expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes de posibles reclamos o acciones legales.



Pascal Chamba Nury Gabriela

AUTORA



Rosero Narvárez Cristian Paul

AUTOR

Tulcán, abril de 2026

AGRADECIMIENTO

Hoy quiero expresar mis más sinceros agradecimientos a Diós, por ser mi guía y mi fortaleza, por permitirme culminar este trabajo de titulación alcanzar este importante logro en mi carrera. Gracias por mantenerme con esperanza, paciencia y determinación de momentos desafiantes. A mi familia por su apoyo incondicional y constante. a mis padres Bairo Pascal y Martha Chamba les agradezco infinitamente por su amor, sacrificio y consejos que me han impulsado a seguir adelante. A mi hermana Zaira Pascal, mis hermanos Michael Pascal y Johnny Pascal, gracias por estar a mi lado, mis abuelitos, María Muepaz y Carlos Pascal gracias por su apoyo y bendiciones como también agradecer a mis tíos, tías, primas por sus palabras y aliento y consejos que me han motivado a no rendirme y a seguir estudiando.

Agradezco a mis compañeras de clase Tania, Amparito, Daniela, gracias por ser parte de esta aventura por las risas, los desvelos, el apoyo y la amistad cada momento vivido con ustedes, gracias por estar ahí en los buenos y malos momentos, aunque nuestros caminos se separen siempre las llevaré en mi corazón.

A ti Henry Reyes este trabajo también lleva algo de ti, que estuviste presente de manera cercana y significativa: quiero que sepas que desde luego también tiene algo tuyo gracias por estar cuando más lo necesitaba por tu compañía tu aliento y todo lo que compartiste conmigo en este camino, aunque los caminos cambian el agradecimiento permanece.

Mi más sincero agradecimiento al MSC. Milton de hierro, mi tutor de trabajo de titulación por su valiosa orientación y dedicación y apoyo a lo largo de este proceso su experiencia y conocimientos constructivos fueron fundamentales para el desarrollo y culminación de este trabajo, agradezco a todos los docentes que compartieron sus conocimientos y experiencias conmigo durante mi trayecto de estudio su pasión por la enseñanza y su compromiso con la formación de profesionales que han sido una inspiración para mí gracias por contribuir a mi conocimiento académico y profesional.

Nury Pascal

DEDICATORIA

El trabajo de titulación que presenté, titulado herramienta tecnológica para la gestión de talento humano y venta de boletos de transporte público, está dedicado principalmente a Dios, mi constante guía y fortaleza, quién me proporcionó la fe, la perseverancia y el aliento necesario en los momentos más difíciles, ayudándome a completar una etapa fundamental en mi desarrollo académico y personal su presencia fue la luz que iluminó cada paso de este camino.

Dedico este proyecto con profundo cariño y gratitud a mi familia, soporte esencial en mi vida. A mis padres quienes siempre estuvieron a mi lado con su apoyo incondicional enseñándome mediante su ejemplo la importancia del esfuerzo la responsabilidad y la perseverancia, agradezco su confianza su compañía en cada etapa y el amor y comprensión que me brindaron en los momentos más complejos de mis estudios. A mis hermanos, tíos y abuelos por sus palabras de aliento fueron cruciales para mantenerme firme en la culminación, cada gesto de apoyo fue un impulso para que este logro sea una realidad.

De forma especial dedico este trabajo a mi tutor magister Milton de hierro, por su invaluable orientación paciencia y dedicación a lo largo de este proceso investigativo. Su conocimiento profesionalismo y compromiso fueron pilares fundamentales para el desarrollo de este trabajo.

Así mismo, dedico este logro a la Universidad Politécnica estatal del Carchi, la institución que me brindó la oportunidad de crecer académicamente personal y profesional, horas sus autoridades y docentes de la carrera de computación quienes con su dedicación, experiencia y pasión por la enseñanza forjaron mis conocimientos y desarrollo permitiendo culminar esta etapa con satisfacción y orgullo.

A mis compañeras de clases amigos cómplices de esta aventura gracias por las risas los desvelos compartidos y el apoyo mutuo de cada momento difícil, este trabajo no es solo una herramienta tecnológica es el reflejo del amor esfuerzo y apoyo, que recibí de cada una de las personas que formaron parte de este camino.

Nury Pascal

AGRADECIMIENTO

Deseo expresar mi más profundo y sincero agradecimiento a todas las personas que, de una u otra manera, han contribuido al desarrollo y culminación de este trabajo de titulación, etapa que representa no solo un logro académico, sino también un crecimiento personal y profesional en mi vida.

En primer lugar, agradezco a mis padres, Chouenlay Rosero Reina y Silvia Yolanda Narváez Cevallos, quienes han sido el pilar fundamental de mi formación. Su esfuerzo, sacrificio y apoyo incondicional a lo largo de los años han sido la base que me permitió avanzar en cada etapa de mi vida académica. Gracias por inculcarme valores como la responsabilidad, la perseverancia y la honestidad, los cuales han guiado mi camino hasta alcanzar esta meta.

Expreso también mi sincero agradecimiento a mi tutor de tesis, MSc. Milton Del Hierro, por su orientación académica, sus valiosos conocimientos, su paciencia y su constante disposición para guiarme durante el desarrollo de este trabajo investigativo. Su experiencia y recomendaciones fueron fundamentales para fortalecer la calidad de esta investigación.

De igual manera, agradezco profundamente a mi pareja, Salomé Changuan, por su apoyo incondicional, comprensión y motivación durante todo este proceso. Su compañía y confianza en mí fueron una fuente constante de inspiración para continuar adelante incluso en los momentos más desafiantes.

Extiendo también mi gratitud a mis suegros, Ana Báez y Óscar Changuan, quienes me brindaron su apoyo, confianza y palabras de aliento durante este camino académico.

A todos ustedes, gracias por creer en mí, por acompañarme en este proceso y por ser parte fundamental de este logro que hoy culmina con la presentación de este trabajo.

Cristian Paul Rosero Narváez

DEDICATORIA

Dedico este logro, fruto de años de esfuerzo, dedicación y aprendizaje, a las personas más importantes de mi vida.

A mis padres, Chouenlay Rosero Reina y Silvia Yolanda Narváez Cevallos, quienes con su amor, sacrificio y ejemplo han sido siempre mi mayor inspiración. Gracias por enseñarme el valor del esfuerzo, la responsabilidad y la perseverancia, principios que me permitieron alcanzar esta meta.

A mi hija, Paula Rosero, quien representa la mayor motivación de mi vida y la razón que me impulsa a seguir superándome cada día. Este logro también es para ti, como ejemplo de que con dedicación y constancia es posible cumplir los sueños.

A mi hermano Mateo Rosero, por su apoyo, compañía y por ser parte importante de mi vida y de mi crecimiento personal.

Y de manera muy especial, en memoria de mi hermano Jhosember Rosero, quien, aunque ya no se encuentra físicamente entre nosotros, vive siempre en mi recuerdo y en mi corazón. Su memoria sigue siendo una fuente de inspiración para continuar adelante y luchar por cada meta que me propongo.

A todos ustedes, con profundo amor y gratitud, dedico este logro que representa una etapa importante en mi vida.

Cristian Paul Rosero Narváez

ÍNDICE

RESUMEN	16
ABSTRACT	17
INTRODUCCIÓN	18
I. EL PROBLEMA	19
1.1. PLANTEAMIENTO DEL PROBLEMA	19
1.2. FORMULACIÓN DEL PROBLEMA	20
1.3. JUSTIFICACIÓN	20
1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN	22
1.4.1. Objetivo General	22
1.4.2. Objetivos Específicos	22
1.4.3. Preguntas de Investigación	22
II. FUNDAMENTACIÓN TEÓRICA	23
2.1. ANTECEDENTES DE LA INVESTIGACIÓN	23
2.2. MARCO TEÓRICO	24
2.2.1. Software de gestión en el transporte	24
2.2.2. Herramientas de construcción de software	25
2.2.3. Sintaxis de programación	25
2.2.4. Framework	26
2.2.4.1. Flutter	26
2.2.4.2. Lenguaje Dart	26
2.2.5. Base de Datos	26
2.2.5.1. Tipos de base de datos	27
2.2.5.2. Ventajas de optimizar el rendimiento de una base de datos	27
2.2.6. Firebase	27
2.2.7. Ingeniería de Software	28
2.2.7.1. Calidad de Software	28
2.2.8. Metodologías de Desarrollo	28
2.2.9. Metodología Ágil	29

2.2.10. Estándar ISO/IEC/IEEE 29148:2018.....	30
2.2.11. Norma ISO/IEC 25010.....	30
2.2.12. Estructura del Organigrama.....	32
2.2.12.1. Gestión de Talento Humano	33
2.2.12.2. La Importancia de la gestión del talento humano	34
2.2.12.3. Estrategias para Gestionar el Talento	34
2.2.13. Digitalización y tecnología en talento humano.....	34
2.2.14. Control de Datos de Choferes en el Transporte Público	35
2.2.15. Compra de Boletos en Línea en el Transporte Público.....	35
2.2.16. Transporte Público en Zonas.....	35
2.2.17. Tecnología en la Gestión de Transporte	36
2.2.18. Adopción de Tecnología en Zonas Rurales.....	36
2.2.18.1. Comparativa de herramientas tecnológicas para la optimización de boletos	36
III. METODOLOGÍA.....	37
3.1. ENFOQUE METODOLÓGICO	37
3.1.1. Enfoque	37
3.1.2. Tipo de Investigación	37
3.1.2.1. Investigación Documental	37
3.1.2.2. Investigación de campo	38
3.2. IDEA A DEFENDER	38
3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES.....	38
3.3.1. Operacionalización de las variables	39
3.4. MÉTODOS UTILIZADOS	40
3.4.1. Métodos	40
3.4.2. Técnicas.....	40
3.4.2.1. Encuesta.....	41
3.4.2.2. Entrevista estructurada.	41

3.5. ANÁLISIS ESTADÍSTICO	41
3.5.1. Determinación del tamaño de la encuesta.....	42
IV. RESULTADOS Y DISCUSIÓN	44
4.1. RESULTADOS	44
4.1.1. Resultados de la encuesta.....	44
4.2. PROPUESTA	50
4.2.1. Fase de planificación.....	50
4.2.1.1. Levantamiento de procesos	51
4.2.1.2. Requerimientos funcionales	52
4.2.1.3. Requerimientos no funcionales.....	55
4.2.2. Fase de diseño	56
4.2.3. Fase de Codificación.....	66
4.2.3.1. Arquitectura del Sistema.....	66
4.2.3.2. Configuración de Firebase e Inicialización.....	67
4.2.3.3. Modelos de Datos y Entidades	68
4.2.3.4. Implementación del Sistema de Bloqueo de Asientos	68
4.2.3.5. Interfaz de Usuario - Módulo de Reserva.....	69
4.2.3.6. Autenticación y Gestión de Roles	70
4.2.3.7. Gestión de Encomiendas (RF004).....	70
4.2.3.8. Panel de Administración - Validación de Pagos (RF011).....	71
4.2.3.9. Notificaciones y Alertas (RF006, RF008)	72
4.2.3.10. Control de versiones.....	72
4.2.3.11. Módulo de gestión de talento humano.....	73
4.2.3.11.1. Funcionalidades implementadas.....	73
4.2.3.11.2. Gestión de roles y perfiles del personal	74
4.2.3.11.3. Síntesis del Módulo.....	75
4.2.4. Fase de pruebas.....	76
4.2.5. Fase de Retroalimentación	81

4.2.6. Fase de Lanzamiento	83
4.3. DISCUSIÓN.....	102
V. CONCLUSIONES Y RECOMENDACIONES.....	105
5.1.CONCLUSIONES	105
5.2.RECOMENDACIONES.....	106
VI. REFERENCIAS BIBLIOGRÁFICAS.....	107
VII. ANEXOS.....	110

ÍNDICE DE TABLAS

Tabla 1. Comparación de la metodología ágil y tradicional	29
Tabla 2. Metodología XP	30
Tabla 3. Roles	32
Tabla 4. Comparativa de herramientas	36
Tabla 5. Operacionalización de las variables independientes.....	39
Tabla 6. Operacionalización de las variables dependientes.....	40
Tabla 7. Historia del usuario.....	51
Tabla 8. Requerimientos Funcionales.....	52
Tabla 9. Requerimientos no Funcionales	55
Tabla 10. Roles del sistema Trans Doramald y permisos por perfil.....	75
Tabla 11. Tiempo promedio y máximo en pruebas de usabilidad.....	80
Tabla 12. Criterios de evaluación por tipo de usuario.....	80
Tabla 13. Resultados y análisis de las pruebas de usabilidad del sistema Trans Doramald	80
Tabla 14. Comparativa de antecedentes.....	104

ÍNDICE DE FIGURAS

Figura 1. Organigrama.....	33
Figura 2. Pregunta 1	44
Figura 3. Pregunta 2.....	45
Figura 4. Pregunta 3.....	46
Figura 5. Pregunta 4.....	46
Figura 6. Pregunta 5.....	47
Figura 7. Pregunta 6.....	48
Figura 8. Pregunta 7	48

Figura 9. Pregunta 8.....	49
Figura 10. Pregunta 9.....	49
Figura 11. Pregunta 10.....	50
Figura 12. Clases.....	57
Figura 13. Caso de uso – registro	58
Figura 14. Secuencia – registro.....	58
Figura 15. Caso de uso - inicio de sesión.....	59
Figura 16. Secuencia - inicio de sesión	59
Figura 17. Caso de uso - consulta de rutas	60
Figura 18. Secuencia - consulta de rutas.....	60
Figura 19. Caso de uso - reserva de asiento	61
Figura 20. Secuencia - reserva de asiento	62
Figura 21. Caso de uso - compra de boleto.....	62
Figura 22. Secuencia - compra de boleto	63
Figura 23. Caso de uso – encomienda.....	63
Figura 24. Secuencia – encomienda	64
Figura 25. Caso de uso - gestión administrativa.....	64
Figura 26. Secuencia - gestión administrativa	65
Figura 27. Conexión de la base de datos	66
Figura 28. Árbol de directorios.....	67
Figura 29. Inicialización de firebase.....	67
Figura 30. Lógica de reserva	68
Figura 31. Disponibilidad de asientos	69
Figura 32. Reserva de asiento.....	69
Figura 33. Autenticación dual.....	70
Figura 34. Gestión encomiendas.....	71
Figura 35. Validación de pagos.....	71
Figura 36. Sistema de notificaciones	72
Figura 37. Control de versiones	73
Figura 38. Entorno de pruebas (dispositivo gama media).....	76
Figura 39. Entorno de pruebas (dispositivo gama baja)	77
Figura 40. Información de compatibilidad de la aplicación	77
Figura 41. Resultados de las pruebas con obtención de pass (satisfactorio).....	77
Figura 42. Métricas de uso de cpu, ram y memoria.....	78
Figura 43. Pruebas de estrés y robo en la aplicación.....	78

Figura 44. Uso de cpu y memoria ram (Test Lab)	79
Figura 45. Resultados de rendimiento de la aplicación.....	79
Figura 46. Corrección de notificaciones	82
Figura 47. Corrección y simplificación de UX.....	82
Figura 48. Corrección del módulo de notificaciones push.....	83
Figura 49. Alerta de seguridad.....	83
Figura 50. Panel de información.....	84
Figura 51. Panel de misión y visión.....	85
Figura 52. Panel de Contactos.....	85
Figura 53. Menú de acceso.....	86
Figura 54. Panel de inicio de sesión.....	86
Figura 55. Panel de selección de cuenta (Google)	87
Figura 56. Panel de servicios.....	87
Figura 57. Panel de compra de boletos	88
Figura 58. Panel de buses disponibles.....	88
Figura 59. Panel de selección de parada.....	89
Figura 60. Panel de selección de asiento.....	89
Figura 61. Panel de información del pasajero.....	90
Figura 62. Panel de método de pago	90
Figura 63. Panel de historial de boletos	91
Figura 64. Panel de encomiendas.....	91
Figura 65. Panel de nueva encomienda (remitente)	92
Figura 66. Panel de nueva encomienda (destinatario).....	92
Figura 67. Panel de nueva encomienda (detalles del paquete).....	93
Figura 68. Ventana de solicitud registrada	93
Figura 69. Panel de notificaciones.....	94
Figura 70. Servicios disponibles (chofer)	94
Figura 71. Gestión de encomiendas (chofer).....	95
Figura 72. Panel de inicio de sesión (administrador).....	95
Figura 73. Código de verificación de administrador.....	96
Figura 74. Panel de control.....	96
Figura 75. Panel de gestión de rutas	97
Figura 76. Panel de nueva ruta.....	97
Figura 77. Panel de rutas registradas.....	98
Figura 78. Panel de gestión de encomiendas (administrador).....	98

Figura 79. Panel de tipos de encomiendas.....	99
Figura 80. Panel de enviar notificación	99
Figura 81. Panel de buses registrados	100
Figura 82. Panel de crear nueva salida de bus	100
Figura 83. Panel de galería de imágenes.....	101
Figura 84. Panel de gestión de pagos	101
Figura 85. Panel de gestión de datos.....	102

ÍNDICE DE ANEXOS

Anexo 1. Acta de la sustentación de Predefensa del TIC	110
Anexo 2. Certificado del abstract por parte de idiomas.....	112
Anexo 3. Código de la app Trans Doramald.....	114
Anexo 4. Manual De Usuario Del Sistema Trans Doramald	156

RESUMEN

El presente estudio titulado Herramienta tecnológica móvil para la gestión de talento humano y venta de boletos de transporte público, en la compañía Trans Doramald, ubicada en el cantón Tulcán. La investigación adoptó un enfoque mixto cuantitativo, mediante encuestas estructuradas a los 152 habitantes y cualitativo a través de entrevistas semiestructuradas al personal administrativo y conductores, con lo que se identificaron problemas operativos críticos como: la adquisición de boletos, la duplicidad de asientos, la ausencia de un sistema de comunicación oficial y los retrasos de las salidas en tiempo real. El desarrollo de la aplicación fue mediante la metodología ágil Extreme Programming (XP), empleado Flutter y Dart como framework multiplataformas y Firebase como gestor de base de datos y la plataforma de autenticación en tiempo real. Como resultado se obtuvo la aplicación móvil Trans Doramald integrada en un boletero digital con mapa de asientos interactivos y bloqueo transaccional en tiempo real, un módulo de registro y rastreo de encomiendas, un sistema de notificación push y un modelo de gestión de talento humano con cuatro perfiles diferenciados por roles: administrador, secretaria, conductor y pasajero. La implementación fortaleció las principales limitaciones del sistema manual: duplicidad de pagos, falta de control en tiempo real y desorganización operativa, digitalizar el control del personal y habilitar múltiples métodos de pago, la arquitectura resultó óptima para el contexto rural de Maldonado y El Chical logrando una experiencia positiva con los usuarios al utilizar la herramienta con un funcionamiento offline y una sincronización en tiempo real ante la conectividad intermitente. Se concluye que la digitalización del sistema mejoró significativamente la eficiencia operativa de la compañía siendo replicable en contextos rurales similares.

Palabras Claves: Aplicación Móvil, Gestión de talento humano, Venta de Boletos, Transporte Público Rural, Flutter, Firebase, Metodología XP.

ABSTRACT

This study, entitled "Mobile Technological Tool for Human Talent Management and Public Transportation Ticket Sales at Trans Doramald, located in the Tulcán canton," adopted a mixed quantitative approach, using structured surveys of 152 residents, and a qualitative approach through semi-structured interviews with administrative staff and drivers. This approach identified critical operational problems such as ticket acquisition, duplicate seats, the absence of an official communication system, and real-time departure delays. The application was developed using the Extreme Programming (XP) agile methodology, employing Flutter and Dart as cross-platform frameworks, and Firebase as the database management and real-time authentication platform. The result was the Trans Doramald mobile application, integrated with a digital ticketing system featuring interactive seat maps and real-time transactional blocking, a parcel registration and tracking module, a push notification system, and a human talent management model with four profiles differentiated by roles: administrator, secretary, driver, and passenger. The implementation addressed the main limitations of the manual system: duplicate payments, lack of real-time control, and operational disorganization. By digitizing personnel management and enabling multiple payment methods, the architecture proved optimal for the rural context of Maldonado and El Chical, achieving a positive user experience. Users were able to use the tool offline and synchronize in real time despite intermittent connectivity. It is concluded that the digitization of the system significantly improved the company's operational efficiency and is replicable in similar rural contexts.

Keywords: Mobile Application, Human Talent Management, Ticket Sales, Rural Public Transportation, Flutter, Firebase, XP Methodology.

INTRODUCCIÓN

El transporte público rural constituye un servicio esencial para las comunidades alejadas de los centros urbanos, siendo en muchos casos el único medio de movilización disponible para sus habitantes a nivel mundial la incorporación de tecnologías digitales en la gestión de transporte ha demostrado generar mejoras significativas en la eficiencia operativa la reducción de tiempos de espera y la satisfacción del usuario.

La presente investigación se justifica por la necesidad concreta y actual de mejorar el acceso, la eficiencia y la calidad del transporte público en las parroquias rurales El Chical y Maldonado, cantón Tulcán, Provincia del Carchi se ubica la compañía de transporte rural Trans Doramald, atendiendo las necesidades de la movilización de aproximadamente 2700 habitantes. A pesar de la importancia de su funcionamiento social, la compañía carece de herramientas digitales para gestionar sus operaciones la problemática afecta de manera directa tanto a los usuarios como el personal operativo de la compañía. Los pasajeros enfrentan dificultades para adquirir boletos, planificar sus viajes y acceder a información actualizada sobre rutas y horarios, viéndose obligados a trasladarse físicamente a los puntos de venta o realicé el pago directamente con el conductor.

El objetivo general de esta investigación es desarrollar una herramienta tecnológica móvil que integre la gestión de talento humano y la venta de boletos en una sola plataforma digital, adaptada a las condiciones de conectividad intermitente características de las zonas rurales. La solución busca modernizar los procesos internos de la compañía, mejorar la experiencia de los usuarios y generar un impacto positivo en la movilidad y calidad de vida de las comunidades.

La investigación adoptó un enfoque mixto, combinando métodos cuantitativos mediante encuestas estructuradas y métodos cualitativos a través de entrevistas semiestructuradas al personal de Trans Doramald. El desarrollo de la aplicación móvil siguió la metodología ágil Extreme Programming (XP), aplicando iteraciones cortas y retroalimentación continua. Se utilizaron Flutter y Dart como framework de desarrollo multiplataforma, firebase como una base de datos en tiempo real y Visual Studio Code como un entorno de programación.

I. EL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

A nivel global, Aleatica es una compañía dedicada a la operación de infraestructura de transporte con presencia en América Latina y Europa se han desarrollado varios proyectos orientados a la implementación de sistemas de transporte inteligentes, la compañía ha puesto en funcionamiento sistemas de cobro enfocados en mejorar la eficiencia, sostenibilidad y seguridad del transporte terrestre. Su objetivo es brindar soluciones de movilidad inteligente, segura y sostenible a largo plazo que impulsan el crecimiento (Hernández, 2023).

El transporte público a nivel nacional ha incorporado la telemática como herramienta vital para adaptarse a los cambios tecnológicos. En Ecuador, está concentrado principalmente en las ciudades de Quito y Guayaquil, que reúnen la mayor parte de la conectividad y transformación digital del país, los factores que han impulsado en el crecimiento de esta evolución permiten una gestión más eficiente y efectiva de las flotas de vehículos reducción de costos y un aumento de la productividad, en plataformas de comercio electrónico y aplicaciones móviles para eventos y pasajes, con alta penetración en las zonas urbanas de Pichincha y Guayas gracias a la incorporación de soluciones innovadoras y sofisticadas para mejorar la eficiencia de la logística y el transporte. Un software ayuda a ahorrar tiempo, es eficiente, moderno y permite centralizar la información del personal, por esto es imprescindible para las compañías contar con un software de Recursos Humanos en la nube.

El cantón Tulcán desde sus perspectivas los usuarios y trabajadores enfrentan dificultades con la tecnología, disminuyendo la calidad del servicio. Los pasajeros enfrentan dificultades para adquirir un boleto y planificar sus viajes, debido a las

deficiencias en el acceso a información sobre rutas y horarios. Los conductores y el personal administrativo carecen de una herramienta tecnológica para gestionar adecuadamente su servicio, lo que se traduce en largas esperas, baja seguridad laboral y una atención deficiente al cliente en los territorios de zonas rurales. Las parroquias El Chical y Maldonado son un claro ejemplo de los desafíos que enfrentan en el transporte rural, donde actualmente operan un servicio con múltiples dificultades que afectan la movilidad diaria.

El Chical y Maldonado son los principales lugares donde se evidencia la falta de una herramienta tecnológica para la gestión de boletos y talento humano. En estos lugares no existe una plataforma digital que facilite la compra de boletos en línea, la optimización de rutas, la gestión de horarios ni el control del número de pasajeros. Estas carencias limitan la toma de decisiones estratégicas en la compañía Trans Doramald.

Es momento de que las parroquias El Chical y Maldonado cuenten con una aplicación móvil que permita gestionar y optimizar el transporte de forma sencilla y segura. Esta herramienta facilitará a sus habitantes la reserva de boletos en línea, sin necesidad de trasladarse a los puntos de venta, ahorrando tiempo y garantizando el acceso a un asiento para realizar sus viajes.

1.2. FORMULACIÓN DEL PROBLEMA

¿Cómo una herramienta tecnológica contribuye a la mejora en el proceso de venta de boletos en el transporte público y la gestión de talento humano en la compañía Trans Doramald?

1.3. JUSTIFICACIÓN

La gestión de servicios de transporte en zonas rurales representa un desafío estructural que incide directamente en la calidad de vida de las comunidades. En lugares como la parroquia El Chical, la ausencia de herramientas tecnológicas para optimizar la operación del transporte público genera dificultades tanto para los usuarios como para los operadores del servicio. Esta problemática real justifica la necesidad de emprender una investigación que permita desarrollar soluciones innovadoras y adaptadas al contexto local.

La población de El Chical podría beneficiarse de esta herramienta tecnológica, ya sea como usuarios frecuentes del transporte público o como personas que dependen de forma ocasional. Esto incluye a estudiantes, trabajadores, personas mayores y cualquier habitante que necesite moverse dentro o fuera de la comunidad. Aunque no todos usan el servicio a diario, una mejora en el sistema de transporte impacta positivamente en toda la población, ya que facilita el acceso a servicios, educación, salud y oportunidades económicas.

Esta investigación es altamente conveniente, ya que aborda una necesidad concreta y actual: mejorar el acceso, la eficiencia y la calidad del transporte público en zonas rurales mediante el uso de herramientas digitales. Su utilidad se manifiesta en la posibilidad de implementar una aplicación tecnológica que facilite la compra de boletos en línea, la planificación de rutas y horarios y el bienestar del personal.

La presente investigación tiene como propósito contribuir a la sociedad mejorando el acceso al transporte. Estas acciones permiten reducir las dificultades que enfrentan las personas en zonas rurales, fomentar el crecimiento económico y generar un impacto positivo en la calidad del servicio con nuevas oportunidades.

Los principales beneficiarios de esta investigación serán los usuarios del servicio y el personal de Trans Doramald, incluyendo conductores y administradores. Todos podrán disfrutar de un servicio más eficiente, confiable y fácil de usar. Para ello, se combinarán métodos cualitativos y cuantitativos con el fin de obtener una visión integral del problema y encontrar soluciones efectivas, se espera que este servicio de transporte sea más organizado, predecible y accesible para todos.

La combinación de enfoques cualitativos y cuantitativos permitirá analizar los datos, la frecuencia de uso y los tiempos de espera, ofreciendo una visión clara de cómo debe funcionar el sistema. El análisis se centrará en las necesidades de todos los actores involucrados: usuarios, administradores y conductores, quienes deberán adaptarse a esta nueva modalidad tecnológica para lograr una solución efectiva.

De esta forma, se espera no solo optimizar la gestión interna de las empresas de transporte, sino también elevar significativamente la experiencia de los usuarios, este proyecto contribuirá a resolver un problema real y palpable en la vida cotidiana de miles de personas, convirtiéndose en un aporte significativo para la transformación tecnológica y social del transporte rural.

1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN

1.4.1. Objetivo General

Desarrollar una herramienta tecnológica que permita la gestión de talento humano y compra de boletos en línea

1.4.2. Objetivos Específicos

- Examinar la gestión del servicio de boletaje y talento humano en el transporte rural mediante la identificación de sus limitaciones actuales y el diseño de un organigrama funcional que delimite responsabilidades específicas, orientado a la mitigación de dichas dificultades.
- Comparar herramientas tecnológicas para la optimización de la compra de boletos y a gestión digital de talento humano.
- Establecer una solución tecnológica móvil para optimizar la compra de boletos con la gestión de talento humano, facilitando el acceso, la eficiencia del servicio y la experiencia del usuario.

1.4.3. Preguntas de Investigación

- ¿Cómo se estructura el organigrama de la compañía y qué funciones y responsabilidades se asignan para garantizar una distribución eficiente de actividades?
- ¿Qué características debe tener una solución tecnológica para mejorar el acceso, la eficiencia y la experiencia del usuario en la compra de boletos?
- ¿De qué forma una herramienta móvil puede mejorar la eficiencia operativa, la experiencia del usuario y el acceso al transporte en comunidades rurales

II. FUNDAMENTACIÓN TEÓRICA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

De acuerdo con Basantes (2023), la Operadora Interprovincial Riobamba enfrenta desafíos críticos en su gestión operativa debido a la ausencia de un sistema automatizado para el registro de afiliados y la administración de frecuencias, esta carencia tecnológica obliga a que procesos fundamentales, como la emisión de boletos y el control de inventario de asientos, se realicen de manera manual, lo que deriva una falta de sincronización de la información en tiempo real entre las distintas sucursales. Como consecuencia, la operadora experimenta problemas de ineficiencia y logística, tales como la duplicidad de pasajes y la incapacidad de los propietarios para monitorear la ubicación exacta de las unidades, comprometiendo tanto la calidad al usuario como la planificación de salidas vehiculares.

Carrasco Mena (2023) señaló en su trabajo "Diseño de una Aplicación para la gestión de pasajes de buses interprovinciales con el propósito de automatizar, sistematizar y optimizar el proceso de registro, venta y reserva de boletos, utilizando métodos de investigación descriptiva y técnicas de encuesta para recopilar datos y descubrir que la empresa no tiene un sistema de emisión de boletos, los usuarios hacen cola para obtener boletos y usan dispositivos móviles Android para las transacciones de emisión de boletos, porque el uso de la tecnología es beneficioso para mejorar la calidad de los procesos y servicios de la empresa.

Route (2023) señala que los avances tecnológicos han repercutido en el uso de las cooperativas de transporte, ya que dentro de las unidades se utiliza con mayor frecuencia Internet, lo que además de mantener informados a los usuarios, también mejora la gestión del personal. Asimismo, destaca que estas innovaciones han sido clave para reducir accidentes viales. El control electrónico de estabilidad ha disminuido los vuelcos en camiones al tomar curvas a alta velocidad. Además, los

sistemas de prevención de colisiones, que combinan radares, cámaras y sensores, permiten detectar y reaccionar ante peligros, mejorando la seguridad vial.

Díaz (2022) define una aplicación web que funciona en base a cliente y servidor y contiene tres tipos de niveles. La capa superior es la encargada de interactuar con el usuario (generalmente ubicada en el navegador), el servidor web, también conocido como servidor de aplicaciones, se considera la capa intermedia, encargada de procesar los datos, y finalmente la base de datos se considera la capa inferior, encargada de proporcionar los datos necesarios. Este tipo de aplicaciones funcionan mediante el protocolo de comunicación HTTP.

Un estudio realizado por Díaz Salinas (2024) sobre la modernización del transporte público en Ecuador, la digitalización del transporte público ha avanzado en los últimos años, especialmente en el ámbito interprovincial, con la implementación de plataformas en línea para la compra de boletos. Sin embargo, en el transporte urbano y rural, la adopción de estos sistemas aún es limitada debido a factores como la falta de infraestructura tecnológica, resistencia al cambio por parte de operadores y la brecha digital en ciertas regiones del país, que la implementación de plataformas digitales para la compra de boletos mejora significativamente la eficiencia del servicio, reduce los tiempos de espera y optimiza la planificación de rutas. A pesar de los avances en la actualidad muchas cooperativas siguen realizando el método de venta antiguo por una falta de inversión tecnológica, como también la desconfianza de los usuarios hacia los pagos digitales.

2.2. MARCO TEÓRICO

2.2.1. Software de gestión en el transporte.

Son herramientas clave para mejorar la eficiencia y la productividad de las empresas. Impulsan la innovación en las tareas diarias, permitiendo reducir costos y optimizar

recursos. Además, fomentan la colaboración, el intercambio de conocimientos y el aumento de la competitividad en el mercado (Nebreda, 2025).

Beneficios de las herramientas tecnológicas

- Permite procesar grandes volúmenes de datos y aprender de ellos.
- Ayuda a la toma de decisiones basadas en datos.
- Favorece la automatización de tareas complejas.
- Ahorra tiempo y reduce los errores humanos.
- Impulsa la personalización de la experiencia de los usuarios o clientes.
- Permite detectar fraudes y aumentar la ciberseguridad.
- Anticipa fallos o errores en sectores industriales.
- Mejora la atención al cliente y reduce los tiempos de espera.
- Incrementa la productividad.
- Mejora y agiliza los procesos de reclutamiento y selección de talento.

2.2.2. Herramientas de construcción de software

Las herramientas son clave para los profesionales que crean software sitios web o aplicaciones móviles, los asistentes inteligentes resuelven tareas complejas de forma rápida manteniendo una calidad y seguridad en el proyecto, el proceso automatiza tareas, escritura, pruebas de código, correcciones de errores. Esto agiliza el trabajo individual y facilita la colaboración entre equipos nos permite trabajar juntos de forma más eficiente nos dan más tiempo para enfocarnos en lo más importante (Vieira, 2024).

2.2.3. Sintaxis de programación

Toda regla permite comunicarnos con una computadora y darle instrucciones para que realice tareas específicas, ya que funcionan como un "puente" que facilita la comunicación entre los humanos y las máquinas, permitiendo así la creación de software, aplicaciones y sistemas que usamos a diario. Existen lenguajes de bajo nivel, muy cercanos a la forma en que la computadora procesa la información, y de alto nivel, que se parecen más al lenguaje humano y son más fáciles de usar. Básicamente, son la clave para construir toda la tecnología digital (Cimas Cuadrado, 2020).

2.2.4. Framework

Es una estructura predefinida con herramientas y componentes que facilitan el desarrollo del software. Sus principales ventajas son: ahorro de tiempo, organización de código y estabilidad. Para este proyecto, se utilizó para construir una aplicación móvil de compra y venta de boletos y gestión de talento humano, el objetivo es ofrecer un desarrollo enfocado en las funcionalidades específicas del proyecto para que el desarrollo sea eficiente, robusto y escalable, diseñado para trabajar de manera eficiente y con medidas de seguridad integrada (García de Zúñiga, 2025).

2.2.4.1. Flutter

Flutter es un framework construido sobre el lenguaje de Dart originado por Google aplicado en el desarrollo de aplicaciones móviles que ayuda a crear aplicaciones nativas, móviles y web. La combinación con Dart es un lenguaje de forma versátil y con un buen rendimiento al desarrollar una aplicación móvil o web. Al unir estas bases de código en el desarrollo ahorramos tiempo, costos, eficiencia y rápido (Cristancho, 2022).

2.2.4.2. Lenguaje Dart.

Dart es un lenguaje de programación de código abierto, creado por Google. Ofrece una sintaxis moderna orientada a objetos, permite detectar errores de forma anticipada y proporciona una experiencia de codificación eficiente y estructurada (Martínez Cordón, 2021).

2.2.5. Base de Datos

Es un modelo informático diseñado para guardar grandes volúmenes de información de manera organizada, haciendo que sea fácil de encontrar y usar en el futuro. Funciona como una biblioteca digital eficiente: permite almacenar datos de manera segura y organizada para facilitar su consulta futura. Estas arquitecturas no solo permiten el almacenamiento, sino que también transforma la forma en que se interactúa con los datos, permitiendo búsquedas y actualizaciones de forma rápida y sencilla (López, 2021).

2.2.5.1. Tipos de base de datos

Existen varios tipos de bases de datos, diseñados para almacenar y organizar información de maneras diferentes, adaptándose a distintas necesidades. Las bases de datos relacionales, que guardan la información en tablas conectadas entre sí, como hojas de cálculo vinculadas las más comunes son.

- Bases de datos relacionales
- Bases de datos NoSQL
- Bases de datos de objetos
- Bases de datos de grafos
- Bases de datos en memoria

2.2.5.2. Ventajas de optimizar el rendimiento de una base de datos.

Optimizar el rendimiento de una base de datos ofrece beneficios significativos esto se traduce en una mayor eficiencia y productividad, un aumento en la rentabilidad y una mejor experiencia para los usuarios al interactuar con las aplicaciones.

- Mayor velocidad y eficiencia.
- Mayor capacidad de respuesta.
- Mejora del rendimiento del sistema
- Reducción de costos.
- Mayor disponibilidad.

2.2.6. Firebase

Firebase es un conjunto de herramientas robustas que se encuentran en la nube que simplifica enormemente el desarrollo de aplicaciones web y móviles que permiten crear aplicaciones de alta calidad para iOS, Android y la web de forma más rápida, ha evolucionado para agrupar diversas funciones útiles en una sola plataforma. Su principal propósito es acelerar el desarrollo y mantenimiento de aplicaciones, liberando a los desarrolladores de la complejidad del backend (la parte del sistema que el usuario no ve directamente). Firebase organiza sus herramientas en cuatro grupos de desarrollo, crecimiento, monetización y análisis, todo diseñado para que trabajar en tus proyectos sea mucho más fácil y eficiente (López Mora, 2020).

2.2.7. Ingeniería de Software

La ingeniería de software consiste en resolver problemas complejos mediante principios técnicos y creatividad, con el objetivo de desarrollar software útil, confiable y de alta calidad. Combina la informática y la ciencia para construir sistemas digitales, siendo la base que sustenta el funcionamiento de computadoras, redes y aplicaciones modernas. Los ingenieros de software son como arquitectos digitales; aplican principios de ingeniería para diseñar y construir soluciones tecnológicas innovadoras, asegurándose de que el software sea confiable y eficiente (Ceballos, 2025).

2.2.7.1. Calidad de Software

La calidad en el software implica una planificación y control meticuloso del presupuesto, los tiempos y los riesgos, para garantizar que una solución tecnológica cumpla exactamente con las necesidades del usuario de forma consistente y confiable. No es resultado de la creatividad espontánea del desarrollador, sino de un proceso de ingeniería rigurosa, comparable a la construcción de un automóvil, que aborda los desafíos de un campo en constante evolución (Román, 2024).

2.2.8. Metodologías de Desarrollo

Permite a los equipos adaptarse rápidamente a los cambios de un proyecto, las metodologías ágiles son un enfoque de trabajo flexible que imagina que se está construyendo algo a mitad de camino, hay una forma mejor de hacerlo; las metodologías ágiles te permiten cambiar de rumbo sin mayores complicaciones, lo que a menudo lleva a reducir costos y aumentar la productividad. En lugar de seguir un plan rígido de principio a fin, estas metodologías promueven la entrega de valor en ciclos cortos y la mejora continua, fomentando la colaboración y la retroalimentación constante pueden responder de inmediato a las necesidades del entorno, asegurando que el producto final se adapte perfectamente a las expectativas (Garrido Sotomayor, 2024).

Tabla 1. Comparación de la metodología ágil y tradicional

Criterio	Metodología	Metodología Tradicional
Enfoque	Se construye paso a paso, entregando pequeñas mejoras continuamente.	Se sigue un camino fijo de principio a fin, fase por fase.
Flexibilidad	Cambiar de idea a mitad de camino es normal y bienvenido.	Cambiar algo después de arrancar suele ser caro y complicado.
Planificación	El plan vive y respira: se ajusta cada vez que cambia las necesidades	Se arma un gran plan al inicio y luego se procura no desviarse.
Tiempos de entrega	Las versiones se liberan rápido y a menudo; así ves resultados pronto.	El producto "terminado" llega hasta el final del proyecto, y puede tardar bastante.
Participación del cliente	El cliente está presente todo el tiempo, opinando y priorizando.	El cliente se asoma al principio (requisitos) y a final (entrega).
Gestión de riesgos	Los problemas se detectan y corrigen sobre la marcha, en cada iteración	Los riesgos se analizan a comienzo; si algo cambia luego, arreglarlo puede doler.
Documentación	Se escribe lo justo y necesario; la prioridad es que el software funcione.	Se documenta todo con mucho detalle antes y durante el desarrollo.
Control de calidad	Las pruebas se hacen en cada ciclo, no se dejan para el final.	Se testea el producto ya casi completo, en la recta final.
Ejemplos	Scrum, XP, Kanban, RAD	Cascada, Modelo en V, Espiral.

2.2.9. Metodología Ágil

La metodología ágil es una forma de trabajo que prioriza la flexibilidad, divide proyectos grandes en tareas más pequeñas y manejables, entregando resultados en ciclos de trabajo cortos. Así, la mejora es constante y el resultado final se alinea mucho mejor con lo que realmente se necesita, beneficiando a todos al lograr mejores productos y procesos más fluidos piensa como construir una casa por etapas: en lugar de esperar a que todo esté terminado para ver algo, con el enfoque ágil puedes ir viendo avances funcionales a medida que cada etapa se completa (Gilibets, 2023).

La metodología Kanban de gestión de proyectos que, visualizando el flujo de trabajo en tableros, optimiza la eficiencia y la transparencia. Permite a los equipos ágiles adaptar y mejorar continuamente sus procesos, siendo muy útil en el desarrollo de software y otras industrias (Gilibets, 2023).

La metodología Scrum divide los proyectos complejos en ciclos cortos llamados "sprints". Su meta es entregar valor rápidamente a través de incrementos funcionales, adaptándose continuamente a los comentarios del cliente. Esto permite una

colaboración y eficiencia excepcionales, ideal para proyectos que requieren flexibilidad y resultados constantes (Lomelí, 2023).

Extreme Programming (XP) es una metodología ágil que busca hacer el desarrollo de software más eficiente y de mayor calidad. Su clave es la adaptabilidad, asume que los requisitos de los clientes cambiarán. En lugar de intentar predecir todo desde el principio, XP se enfoca en responder rápidamente a esos cambios para entregar justo lo que el usuario necesita, en el momento justo. Es una estrategia que valora la flexibilidad por encima de un plan rígido (Bello, 2021).

Tabla 2. Metodología XP

Parámetro	Kanban	Scrum	Extreme Programming (XP)
Enfoque	Flujo de trabajo continuo y flexible	Iteraciones cortas y definidas (sprints)	Desarrollo iterativo e incremental con entrega continua de valor
Roles	No define roles específicos	Product Owner, Scrum Master, Development Team	Programador, Cliente, Coach
Iteraciones	No tiene iteraciones definidas	Sprints con duración fija (2-4 semanas)	Iteraciones cortas (1-4 semanas)
Planificación	Planificación continua y flexible	Planificación al inicio de cada sprint	Planificación continua
Priorización	Priorización continua	Priorización al inicio de cada sprint	Priorización continua
Entrega	Entrega continua	Entrega al final de cada sprint	Entrega continua
Mejora Continua	Retrospectiva y análisis de flujo	Retrospectiva al final de cada sprint	Retrospectiva y prácticas de mejora
Flexibilidad	Alta flexibilidad	Flexibilidad moderada	Alta flexibilidad
Énfasis	Flujo y entrega continua	Iteración y trabajo en equipo	Entrega de valor y mejora continua
Adecuación	Equipos con flujos de trabajo complejos	Equipos con objetivos claros y definidos	Equipos con alta flexibilidad y entrega continua de valor

2.2.10. Estándar ISO/IEC/IEEE 29148:2018.

El estándar ISO/IEC/IEEE 29148:2018 es un estándar internacional que establece una guía clara para la creación de la Especificación de Requisitos de Software. Se puede definir como un plano detallado que define con precisión qué debe hacer un sistema de software. El objetivo es asegurar que este documento sea completo, consistente, verificable y sin ambigüedades, para que todos los involucrados en un proyecto, desde los desarrolladores hasta los usuarios finales, tengan un entendimiento común y claro de lo que se va a construir (Rehman, 2024).

2.2.11. Norma ISO/IEC 25010

El propósito principal de la implementación de la norma ISO/IEC 25010 es servir como un marco de referencia para garantizar que el software proporcione valor real y evite

consecuencias negativas para todas las partes interesadas, como desarrolladores, usuarios y clientes y la integridad de los usuarios dependen directamente de la operatividad de un software de alta calidad permitiendo mitigar riesgos, desde desarrolladores hasta usuarios finales. Aplicando sus modelos de calidad y herramientas de comprobación, es posible establecer requisitos precisos y criterios de satisfacción que alineen la funcionalidad del software asegurando un producto robusto, seguro y eficiente (Paneiva, 2024).

Permite traducir las necesidades humanas y operativas en especificaciones técnicas precisas durante todo el ciclo de vida del desarrollo, desde la gestión del bienestar del talento humano hasta la agilidad en la venta de boletos sea plenamente comprensible, verificable y fiel a la visión original del servicio público.

Beneficios clave:

La documentación ayuda a garantizar un registro detallado que sirve como memoria técnica del proyecto, respetando cada requisito administrativo.

La documentación técnica asegura que cada proceso, desde la gestión de permisos del personal hasta el registro de encomiendas, está documentado de forma detallada y transparente.

La optimización actúa como un puente entre los desarrolladores, departamento de talento humano, usuarios finales, garantizando que todos compartan la misma visión del sistema.

El control de expectativas facilita la detección temprana de ajustes necesarios en el flujo de venta de boletos o servicios de transporte, evaluando su impacto en los recursos y el tiempo del proyecto.

La resolución de problemas permite identificar rápidamente el origen de cualquier problema técnico y verificar si se están cumpliendo las necesidades operativas de la entidad.

La satisfacción del usuario define los requisitos precisos, se asegurando la entrega de un producto de software robusto que responda eficazmente a las demandas del transporte público y la administración de talento humano.

2.2.12. Estructura del Organigrama

El organigrama funcional de la compañía de transporte rural Trans Doramald refleja la forma actual.

El gerente es responsable de la supervisión general de la empresa y debe trasladarse a las sucursales para verificar el cumplimiento de las actividades operativas y el manejo del servicio de transporte.

La administradora se encarga de la venta de boletos, realizando el proceso de forma manual, donde llena los datos del pasajero, elabora el listado de pasajeros por unidad y suma manualmente los valores recaudados posteriormente entrega a cada conductor el listado correspondiente junto con el dinero adecuado.

Los conductores, junto con el personal de cobro, tienen la responsabilidad de transportar a los pasajeros hasta sus paradas asignadas, verificando el abordaje conforme al listado entregado y realizando el cobro cuando corresponde.

Este organigrama permite evidenciar que la gestión del boletaje se realiza de manera manual, lo que genera limitaciones en el control de organización de la información, justificando la necesidad de mejorar el proceso mediante una herramienta tecnológica.

Tabla 3. Roles

Roles.	Función
Gerente	Supervisión general, control de rutas, visitas a sucursales, toma de decisiones.
Administrador	Venta de boletos, registro manual, listado de pasajeros, entrega de dinero.
Conductor	Conducción del bus, cumplimiento de rutas traslado de pasajeros.
Cobrador	Cobro de pasajes, control de pasajeros, apoyo al conductor
Usuarios	Uso del servicio, compra de boletos.

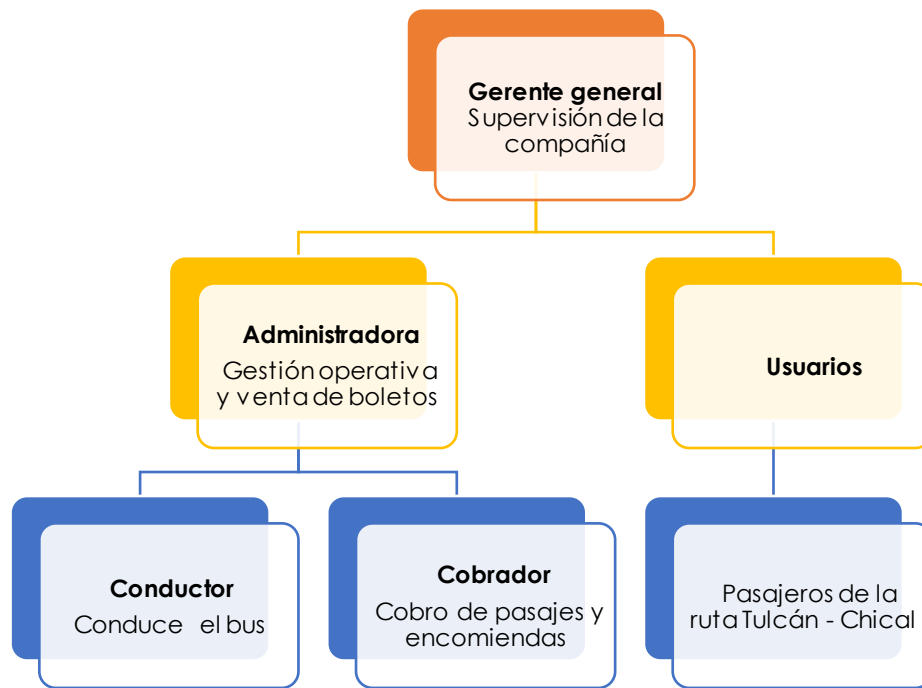


Figura 1. Organigrama

2.2.12.1. Gestión de Talento Humano

La gestión de talento humano se centra en identificar los empleados con mayores capacidades, desarrollar sus habilidades y asegurar su crecimiento y contribución a largo plazo dentro de la organización. Se basa en la idea de que invertir en las personas adecuadas puede transformar el rendimiento y la competitividad de un negocio (Aguilera, 2023).

Requisitos para gestionar un talento humano

- Ayuda a encontrar y atraer a las personas con las habilidades y conocimientos que mejor encajan con la cultura y los objetivos de la empresa.
- Potenciar las capacidades de los empleados a través de programas de capacitación aplicar su aprendizaje en las empresas adaptándose a las necesidades cambiantes del mercado.
- Crea un ambiente de trabajo positivo que fomente la satisfacción, la lealtad y el compromiso del usuario.
- Fortalecer el rendimiento general de la empresa para su desarrollando al mejor talento.

2.2.12.2. La Importancia de la gestión del talento humano

Un plan de gestión de talento humano sólido acompaña a los empleados en cada etapa de su desarrollo, asegurando que las personas correctas ocupen los roles correctos. Esto maximiza sus habilidades, impulsa la innovación y la productividad, y alinea las capacidades individuales con las metas de la organización, creando un mejor rendimiento y una ventaja competitiva clara.

Una estrategia de gestión del talento bien ejecutada también fortalece la reputación de la empresa como un lugar atractivo para trabajar, contribuyendo al éxito a largo plazo (Aguilera, 2023). Estas estrategias permiten a los empleados adquirir nuevas habilidades y mantenerse al día con las tendencias de la industria, lo que no solo aumenta su satisfacción sino también ayuda a la organización a adaptarse a los cambios del mercado y a los avances tecnológicos.

2.2.12.3. Estrategias para Gestionar el Talento

A continuación, se describen diversas tácticas para gestionar su talento:

- Se debe encontrar y atraer a los candidatos más adecuados para las evaluaciones exhaustivas y el uso.
- Implica establecer expectativas claras, retroalimentación constante y reconocer los logros del personal.
- Consiste en equipar al personal con las habilidades y conocimientos necesarios para ser efectivos en sus roles, adaptándose a las necesidades de cada industria.
- Busca crear un ambiente positivo que mejore la satisfacción y la lealtad y un buen equilibrio entre el trabajo y la vida personal y oportunidades de crecimiento.
- Promueve un ambiente de diferentes orígenes y experiencias valoradas, lo que enriquece la cultura empresarial y el crecimiento del equipo.

2.2.13. Digitalización y tecnología en talento humano

La digitalización de los Recursos Humanos se ha vuelto una necesidad urgente, no solo para que el departamento funcione de manera más eficaz, sino también para mejorar el rendimiento general de la innovación. El software de gestión empresarial impulsa la transformación de las organizaciones, convirtiendo procesos manuales en formatos electrónicos y automatizados (González Ortiz, 2023).

2.2.14. Control de Datos de Choferes en el Transporte Público

El control de los choferes permite verificar los antecedentes y habilidades de los conductores, asegurando que estén calificados y cumplan con los estándares de seguridad establecidos no solo promueve la tranquilidad de los pasajeros si no que contribuye un transporte más seguro y confiable garantizando la seguridad y confiabilidad del transporte público (Peña Cayturo, 2022).

La gestión centralizada y la verificación de antecedentes de los choferes mejoran la calidad del servicio de transporte público al permitir un control más eficiente y una supervisión en tiempo real de los conductores. Esto garantiza la idoneidad de los choferes y contribuye a la puntualidad, seguridad y confiabilidad del servicio, beneficiando a los usuarios.

2.2.15. Compra de Boletos en Línea en el Transporte Público

La compra de boletos en línea se ha convertido en un componente esencial para mejorar la accesibilidad y la comodidad en el transporte público. Este sistema ofrece a los pasajeros la flexibilidad de adquirir boletos desde cualquier ubicación, eliminando la necesidad de desplazarse a puntos de venta físicos. Esto no solo ahorra tiempo y esfuerzo, sino que también reduce la congestión en taquillas y estaciones, la compra en línea permite a los usuarios planificar sus viajes con anticipación, verificar horarios y disponibilidad de asientos, mejorando así la eficiencia y la comodidad. Esta herramienta revoluciona la experiencia del usuario al hacer que el transporte público sea más accesible y fácil de usar, contribuyendo a una movilidad más sostenible (Andreea, 2025).

2.2.16. Transporte Público en Zonas

El transporte público como un servicio esencial de infraestructura urbana que pone a disposición de la comunidad medios de transporte compartidos operados en horarios regulares, constituyendo una forma eficiente y asequible de conectar a las personas con sus centros de actividad económica, educativa y de salud, mientras contribuye a la reducción de la congestión vehicular y la sostenibilidad ambiental (López Mora, 2020).

El transporte público en zonas rurales enfrenta obstáculos estructurales derivados de la baja densidad poblacional y la infraestructura de conectividad limitada, lo que genera servicios económicamente inviables, rutas inflexibles y aislamiento social,

dificultando el acceso a servicios esenciales de salud y educación para poblaciones marginadas (Bello, 2021).

2.2.17. Tecnología en la Gestión de Transporte

La integración de sistemas de posicionamiento GPS y plataformas digitales de gestión ha revolucionado el control de flotas en transporte público, permitiendo la localización en tiempo real de las unidades, la optimización dinámica de rutas según la demanda y la reducción de tiempos de espera para los usuarios mediante información predictiva. La implementación de soluciones tecnológicas integrales en operadores de transporte público genera mejoras cuantificables en la eficiencia operativa, permitiendo la asignación dinámica de recursos, la reducción de tiempos muertos en terminales y la optimización de rutas basada en datos de demanda real, lo que se traduce en menores costos operativos y mayor satisfacción del usuario (Díaz Salinas, 2024).

2.2.18. Adopción de Tecnología en Zonas Rurales

La adopción de tecnología en las áreas rurales se ve obstaculizada por desafíos como la falta de infraestructura de conectividad, la falta de la digitalización, y la escasez de recursos financieros y técnicos. Estas barreras limitan el acceso y la implementación de soluciones tecnológicas, dificultando la mejora de servicios como el transporte público en zonas rurales. Enfrentado estos retos en la implementación de herramientas tecnológicas en zonas rurales, es esencial enfocarse en la expansión de la infraestructura de conectividad, brindar capacitación en habilidades digitales a la comunidad y colaborar con organizaciones y gobiernos locales para garantizar la inversión y el apoyo necesarios (Hinestroza, 2022).

2.2.18.1. Comparativa de herramientas tecnológicas para la optimización de boletos

Tabla 4. Comparativa de herramientas

Herramienta	Funcionalidad Principal
Lark	Integra tickets Center y Help desk en un solo espacio con chats, formularios y automatización en tiempo real.
Zendesk	Automatización avanzada, reglas de SLA y asistencia por IA de alto nivel.
Freshdesk	Interfaz muy intuitiva, configuración rápida y recomendaciones inteligentes de IA

III. METODOLOGÍA

3.1. ENFOQUE METODOLÓGICO

3.1.1. Enfoque

Se aplicó un enfoque mixto que combina elementos cualitativos y cuantitativos.

Enfoque cuantitativo: se implementó mediante la aplicación de encuestas estructuradas con preguntas de escala y selección múltiple, dirigidas a usuarios y personal. Este método permitió recolectar datos estadísticos para medir patrones de comportamiento y preferencias de manera numérica.

Enfoque cualitativo: se llevó a cabo a través de entrevistas en profundidad con choferes y personal administrativo. El propósito fue captar experiencias directas, opiniones subjetivas y percepciones detalladas sobre los problemas logísticos actuales lo cual no puede ser reducido únicamente en cifras.

3.1.2. Tipo de Investigación

Se proporcionó una base teórica y técnica necesaria para desarrollar la herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público. Mediante una revisión exhaustiva de libros, artículos científicos, de gestión de recursos humanos y tecnologías de pago electrónico pertinentes. Este procedimiento facilitó la identificación de prácticas óptimas en el diseño de software para la gestión de talento humano y la venta de boletos, y la construcción de una base metodológica sólida para desarrollar la herramienta tecnológica.

3.1.2.1. Investigación Documental

Se enfocó el análisis en sistemas de gestión de talento humano y venta de boletos electrónicos similares, junto con la normativa relacionada con la gestión de transporte público. Esto permitió comprender los requisitos técnicos y operativos que debe

cumplir la herramienta tecnológica. El análisis documental fue clave para acceder a datos históricos y estadísticas sobre sistemas de gestión de personal y venta de boletos existentes, lo que facilitó reconocer los problemas comunes y soluciones efectivas aplicadas en otros contextos.

3.1.2.2. Investigación de campo

Se facilitó la recopilación de datos en tiempo real en el entorno laboral, incluyendo observación directa de los procesos y entrevistas a empleados y usuarios. La metodología permitió comprender las limitaciones del entorno y los factores operativos que afectan la eficiencia. El trabajo con los actores involucrados proporcionó información valiosa sobre preferencias y requerimientos específicos que deben considerarse en el diseño de la herramienta tecnológica, asegurando que el sistema final sea tanto eficiente como fácil de usar.

3.2. IDEA A DEFENDER

Desarrollar una herramienta móvil para optimizar el talento humano y la venta de boletos, incrementando el rendimiento al transporte en el sector de El Chical y Maldonado que están ubicados en una zona rural.

3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES

VARIABLE INDEPENDIENTE:

- Herramienta tecnológica

Las herramientas tecnológicas optimizan el tiempo y la calidad en el trabajo, permitiendo una mejor gestión de recursos y resultados. También promueven el intercambio de conocimiento e investigación, impulsando la rentabilidad, innovación y la capacidad de las empresas para adaptarse a las demandas futuras del mercado y los consumidores (Nebreda, 2025).

VARIABLE DEPENDIENTE:

- Gestión de talento humano y venta de boletos del transporte público

La gestión del talento es un enfoque estratégico y humano que busca atraer, desarrollar y retener a los mejores profesionales, asegurando que su compromiso y crecimiento estén alineados con los objetivos de la organización. Es un proceso integral que acompaña a cada colaborador a lo largo de todo su ciclo de vida

dentro de la empresa, desde su ingreso hasta su eventual salida, buscando siempre satisfacer sus necesidades y potenciar su valor (Hastwell, 2023).

3.3.1. Operacionalización de las variables

Tabla 5. Operacionalización de las variables independientes

Variable	Definición	Dimensión	Indicadores	Técnica	Instrumento
Herramienta tecnológica	Conjunto de recursos digitales desarrollados mediante software que permiten automatizar y optimizar los procesos de gestión de talento humano y venta de boletos en el transporte público rural.	Funcionalidad del sistema	Existencia de módulos de gestión de personal	Entrevista	Guía de entrevista
		Accesibilidad	Registro digital de choferes y personal administrativo Acceso a la aplicación desde dispositivos móviles	Encuesta	Cuestionario
		Automatización de procesos	Disponibilidad del sistema en zonas rurales Reducción de procesos manuales	Observación	Ficha de observación
			Integración de compra de boletos en línea	Encuesta	Cuestionario
		Seguridad de la información	Control de acceso por roles	Entrevista	Guía de entrevista
			Protección de datos de usuarios y empleados	Observación	Lista de verificación

Tabla 6. Operacionalización de las variables dependientes

Variable	Definición	Dimensión	Indicadores	Técnica	Instrumento
Gestión de talento humano y venta de boletos del transporte público	Proceso de administración del personal de la compañía que permite organizar, controlar y optimizar el desempeño de choferes y personal administrativo.	Organización del personal	Registro actualizado del personal	Encuesta	Cuestionario
		Control administrativo	Control de roles y funciones	Entrevista	Guía de entrevista
	Seguimiento de desempeño laboral		Observación	Ficha de observación	
	Proceso de comercialización de pasajes del transporte público mediante medios digitales, garantizando rapidez, disponibilidad y control en tiempo real.	Eficiencia del proceso	Reducción de errores administrativos	Encuesta	Cuestionario
Satisfacción del usuario		Disminución del tiempo de compra	Observación	Lista de verificación	
	Eliminación de boletos duplicados	Encuesta	Cuestionario		
			Nivel de satisfacción del usuario		
			Facilidad de uso del sistema		

3.4. MÉTODOS UTILIZADOS

3.4.1. Métodos

Métodos inductivo y deductivo

Inductivo

Es un enfoque científico que te permite llegar a conclusiones generales partiendo de observaciones o hechos específicos. En pocas palabras, observas algo particular y, basándote en ello, infieres una regla o idea más amplia (Suárez, 2024).

Deductivo

Es un proceso lógico que, partiendo de una idea o premisa general, aplica la lógica para alcanzar una conclusión específica y rigurosa. A diferencia del inductivo, este método empieza con teorías amplias para luego deducir resultados particulares (Suárez, 2024).

3.4.2. Técnicas

Para la presente investigación, se utilizarán la entrevista y la encuesta como técnicas clave para la recopilación de datos cualitativos y cuantitativos.

3.4.2.1. Encuesta

La encuesta consiste en una serie de preguntas dirigidas a una población o a un segmento específico de esta, con el propósito de obtener la opinión sobre un tema determinado. Mediante un cuestionario estructurado, se busca recopilar información relacionada con las variables de estudio, lo que facilitará la evaluación de la viabilidad de implementar la propuesta formulada.

3.4.2.2. Entrevista estructurada.

La ejecución de entrevistas posibilitará la obtención de información relevante acerca de los mecanismos de pago, la gestión financiera, los procedimientos contables, la cantidad de usuarios y las rutas actualmente operadas por las empresas de transporte urbano. La recolección de datos será fundamental para integrar funcionalidades adicionales al proyecto de investigación.

3.5. ANÁLISIS ESTADÍSTICO

Para garantizar que el desarrollo de esta herramienta tecnológica se ajuste a las necesidades reales del sector, se trabajó con una población de 2.700 habitantes de la zona, logrando la participación directa de 152 personas. Se seleccionó de manera estratégica a quienes utilizan sus vehículos para el servicio de la zona rural, incluyendo la visión operativa de secretarios y choferes, así como la experiencia de los usuarios frecuentes, quienes son los que verdaderamente enfrentan las dificultades actuales. El apoyo de sus testimonios fue un aporte permitiendo diseñar una solución basada en vivencias reales y no solo en suposiciones técnicas.

Técnicas

Las técnicas de recopilación de datos empleadas en esta investigación fueron la encuesta y la entrevista semiestructurada, diseñadas para obtener información sobre el proceso de gestión de boletos y talento humano en la compañía Trans Doramald.

Entrevista Semiestructurada

Es la técnica de recolección de datos incluyeron entrevistas y encuestas, utilizando preguntas abiertas centradas en la aceptación para la compra y venta de boletos en Tulcán. La entrevista se llevó a cabo en el área de gestión de talento humano, respondida por el Ger. Gabriel Rosero, encargado del departamento.

Instrumentos

Encuesta: La encuesta se enfocó en conocer las necesidades y preferencias de los usuarios del servicio de transporte de la compañía Trans Doramald, especialmente en las parroquias El Chical y Maldonado. Se empleó una escala de Likert para evaluar las opiniones de los encuestados.

Entrevista

Participantes: 1

Entrevistadores: Nury Pascal, Cristian Rosero

Entrevistado: La entrevista se llevó a cabo en el área de gestión de talento humano, respondida por el Ing. Gabriel Rosero, Gerente de la compañía Trans Doramald S.A

Los conductores de transporte público rural entrevistados manifestaron que su experiencia laboral ha sido completamente manual. Aunque ellos no perciben mayores inconvenientes, reconocen que para los usuarios sí representan un desafío, ya que requiere un contacto directo con los propietarios de cada unidad. Proponen que una herramienta digital ayudaría a evitar estos inconvenientes.

Tanto de los usuarios como los administradores de Trans Doramald expresaron su interés en mejorar las condiciones laborales y están dispuestos a usar aplicaciones móviles para la venta de boletos. Son conscientes de que un cambio hacia lo digital generaría una mejora en la organización el control de cobros de pasajes, encomiendas, respetando la seguridad de cada cliente.

Es fundamental capacitar tanto a los usuarios como al personal de talento humano, para que la implementación del sistema sea efectiva y se logre una adecuada adaptación al cambio tecnológico, considerando que la digitalización avanza de manera acelerada en la actualidad. Los conductores se muestran abiertos a la mejora del servicio con la tecnología y las condiciones de infraestructuras viales y así brindar apoyo a todos los usuarios.

3.5.1. Determinación del tamaño de la encuesta

Para determinar el tamaño de la muestra de la población de El Chical y Maldonado está conformada por 2.700 habitantes, se aplicó la formula estándar para una población finita.

$$n = \frac{Z^2 * p * q * N}{e^2 * (N - 1) + Z^2 * p * q}$$

Donde:

N = Tamaño de la población = 2.700 habitantes

Z = Valor de la distribución normal estándar para el nivel de confianza del 95% = 1,96

p = Probabilidad de éxito (máxima variabilidad) = 0,50

q = Probabilidad de fracaso (q = 1-p) = 0,50

e = Margen de error máximo admisible = 0,08(8%)

Resultado final

$$n = \frac{(1,96)^2 * (0,50) * (0,50) * (2.700)}{(0,08)^2 * (2.700 - 1) + (1,96)^2 * (0,50) * (0,50)}$$

$$n = \frac{(3,8416) * (0,50) * (2.700)}{(17,2736) + (0,9604)}$$

$$n = \frac{2.593,08}{18,234}$$

n = 142 personas.

La fórmula estableció una muestra mínima requerida de 142 personas. Sin embargo, con el propósito de fortalecer la representatividad y la validez estadística del estudio, se decidió incrementar la muestra a 152 personas. Esta decisión reduce el margen de error efectivo a aproximadamente el 7,7%, lo que incrementa la precisión y confiabilidad de los resultados obtenidos.

IV. RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS

4.1.1. Resultados de la encuesta

El levantamiento de información se realizó mediante una encuesta con aproximadamente 152 respuestas, técnica fundamental para obtener información precisa y representativa de la población de las parroquias de El Chical y Maldonado.

Gracias a la herramienta, se logró alinear los objetivos en tiempo real, obteniendo respuestas sobre la realidad del sector, para identificar oportunidades de mejora, permitiendo que la información recopilada sirva de base sólida para la toma de decisiones en beneficio de la comunidad.

La falta de opciones digitales en el transporte Trans Doramald evidencia una oportunidad de modernización. Al ser un proceso presencial, los usuarios están sujetos a la disponibilidad física y tiempos de espera, lo que justifica la necesidad de implementar soluciones tecnológicas que faciliten el acceso al servicio.

A través de qué medio prefiere realizar la compra de sus boletos

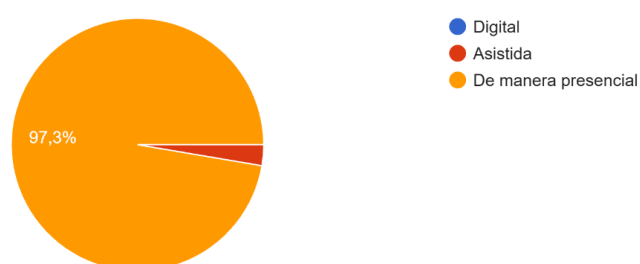


Figura 2. Pregunta 1

A través de una encuesta realizada a la Compañía Trans Doramald, se determinó que el sistema de transporte opera bajo una presencialidad absoluta, con los encuestados realizando la compra de boletos de forma física. Si bien la mayoría de los usuarios utiliza las oficinas de estación como punto principal de compra, mientras que un porcentaje menor opta por el pago directo en la unidad con el conductor lo que refleja la necesidad de soluciones que se adapten tanto a la infraestructura formal como a quienes abordan el transporte en puntos intermedios de la vía.

¿Cómo compra actualmente su boleto de transporte?

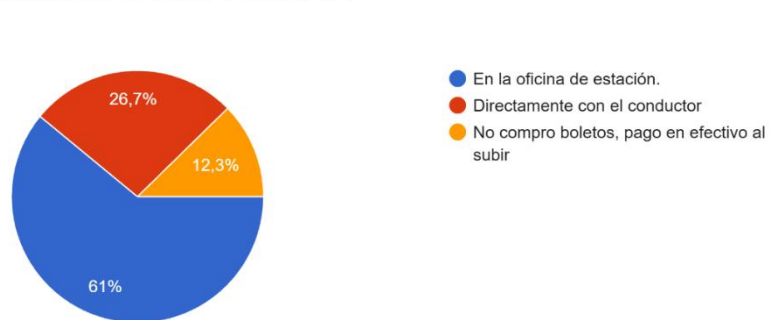


Figura 3. Pregunta 2

En las parroquias de El Chical y Maldonado, el autobús se consolida como el corazón de la movilidad para la gran mayoría de sus habitantes. Esta cifra demuestra que la comunidad depende vitalmente del transporte público para sus actividades diarias, dejando en segundo plano el uso del vehículo particular, en una medida mucho menor, el transporte informal. Estos resultados resaltan que el servicio de autobús no es solo una opción, sino el medio de confianza que mueve a casi tres cuartas partes de la población local.

¿Cuáles son los medios de transporte que usa con más frecuencia? (Puede elegir más de una opción)

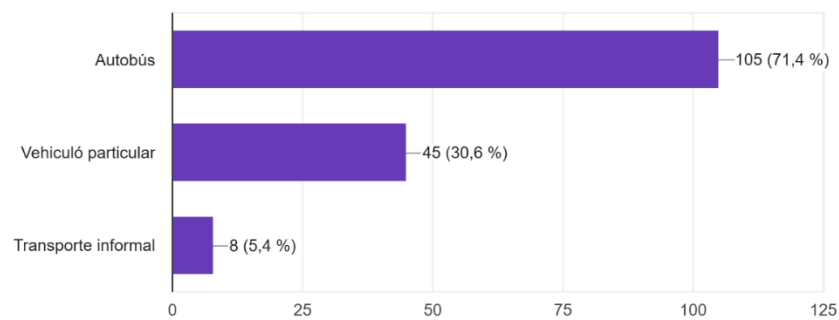


Figura 4. Pregunta 3

Los resultados de la encuesta muestran que la compañía "Trans Doramald" es una pieza clave en la rutina semanal de la comunidad, ya que más de la mitad de los usuarios utiliza sus servicios de 1 a 3 veces por semana. A este grupo se suma las personas que dependen del transporte de forma mensual, mientras que un pequeño sector lo utiliza diariamente o casi nunca. Estas cifras reflejan que el servicio no es solo un medio de transporte ocasional, sino una herramienta constante y esencial para que los habitantes de la zona puedan cumplir con sus actividades regulares, conectando sus hogares con sus destinos de trabajo, comercio o estudio de manera recurrente.

¿Con qué frecuencia utiliza el transporte de la cooperativa "TRANS DORAMALD"?

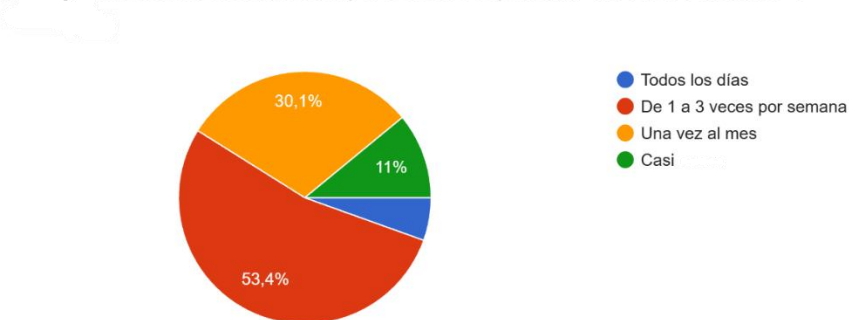


Figura 5. Pregunta 4

En cuanto a la percepción de calidad en las parroquias de El Chical y Maldonado, la mayoría de los usuarios se siente satisfecha y considera que las rutas y horarios actuales de la compañía "Trans Doramald" son adecuados. Sin embargo, existe un grupo importante que manifiesta inconformidad debido a la falta de horarios o rutas

disponibles, mientras que otros indican que su conformidad depende estrictamente del día y la demanda. Estos datos subrayan que, aunque el servicio cumple con las expectativas de más de la mitad de la población, todavía hay un margen de mejora necesario para cubrir las necesidades de movilidad de casi un tercio de la comunidad que demanda mayor disponibilidad.

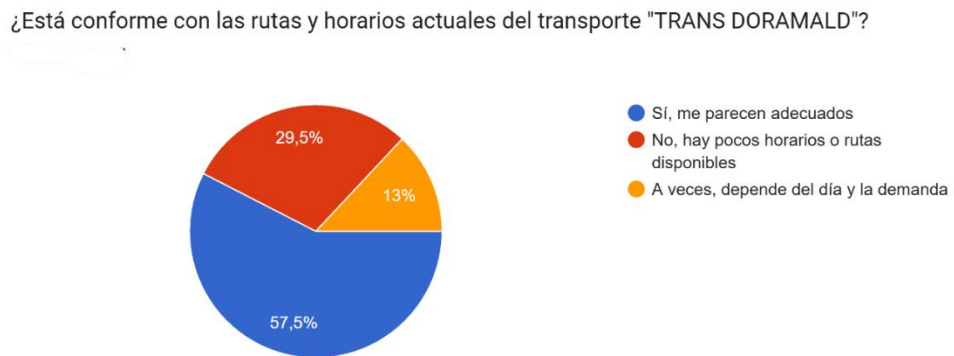


Figura 6. Pregunta 5

Al proyectar el futuro del servicio en El Chical y Maldonado, se percibe una clara voluntad de cambio por parte de los usuarios. Los encuestados manifiestan que les gustaría contar con la opción de comprar su boleto con anticipación a través de puntos de venta cercanos o personas encargadas, señalando que esto les ayudaría a organizar mejor sus viajes. Por otro lado, la población prefiere mantener la costumbre de pagar al momento de subir a la unidad, mientras que un pequeño porcentaje aún se muestra inseguro. Estos resultados son sumamente valiosos, ya que demuestran que la mayoría de la comunidad está abierta a nuevas formas de gestión que les brinden mayor orden y tranquilidad en sus traslados diarios.

Si tuviera la posibilidad de una herramienta tecnológica , ¿le gustaría comprar su boleto con anticipación ?

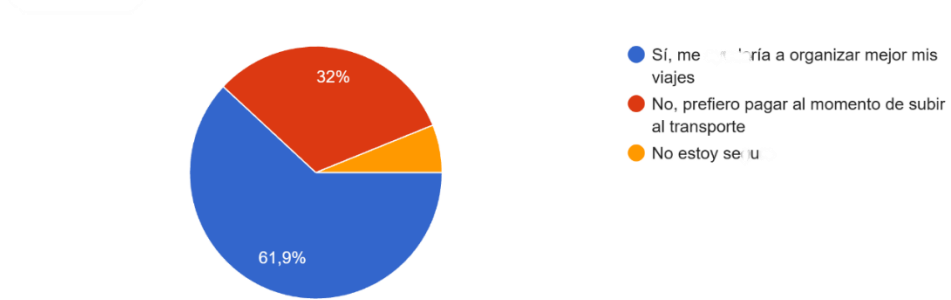


Figura 7. Pregunta 6

El estudio revela una importante apertura al cambio en El Chical y Maldonado, ya que los usuarios consideran que contar con una opción para reservar boletos a través de puntos físicos o encargados en la comunidad sería una excelente alternativa para asegurar su viaje con antelación. Aunque prefieren mantener el método actual de pago por costumbre, los datos demuestran que más de la mitad de la población valora la comodidad de organizar sus traslados desde su propio entorno sin depender de internet. Esta disposición mayoritaria sugiere que implementar puntos de reserva locales no solo facilitaría la vida del pasajero, sino que respondería directamente al deseo de modernizar el servicio de transporte de una manera accesible y humana.

¿Qué tan interesado estaría en utilizar una plataforma en línea para reservar sus boletos de forma anticipada?

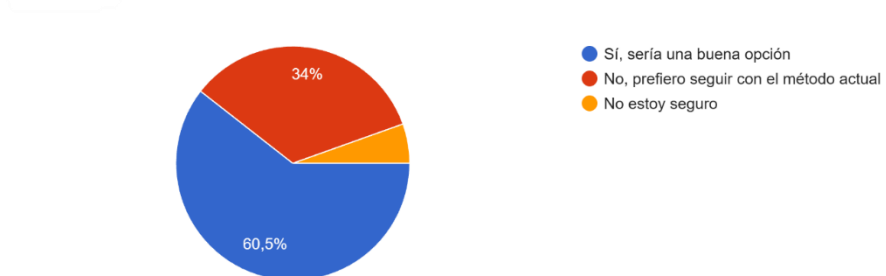


Figura 8. Pregunta 7

Las deficiencias en el servicio de transporte rural revelan que el problema más crítico es la dificultad para encontrar boletos disponibles y una desorganización logística en las cooperativas ya que valida la necesidad de implementar sistemas automatizados de venta y reserva de pasajes.

¿Cuáles son los principales problemas que ha tenido con el transporte rural?

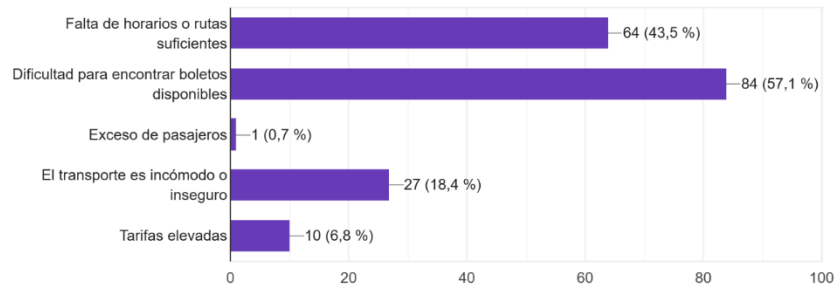


Figura 9. Pregunta 8

Para asegurar una comunicación efectiva con las personas que no cuentan con acceso a internet o telefonía móvil, busca reducir el estrés logístico mediante la automatización, valorando la inmediatez y la utilidad operativa por encima de las funciones transaccionales o promocionales, garantizando la retención del usuario, la estrategia de desarrollo de un sistema de notificaciones que anticipe las necesidades del viajero antes de que este tenga que buscarlas.

Si utilizara nuestra aplicación móvil, ¿Qué funciones y avisos de la compañía le resultarían más útiles para gestionar su viaje?

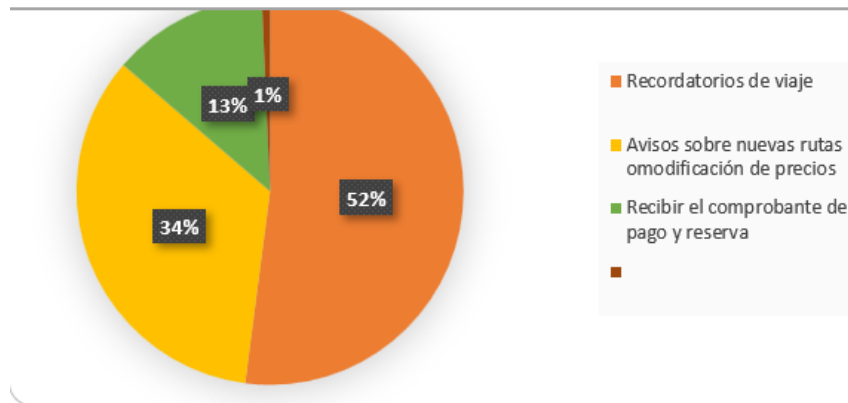


Figura 10. Pregunta 9

Sobre la posibilidad de implementar la venta de boletos en línea en la compañía, la encuesta aplicada a las personas revela una postura mayoritariamente favorable pero cautelosa, donde se considera que su utilidad dependería de la accesibilidad del sistema que agilice los procesos y reduciría errores. Estos datos sugieren que existe una oportunidad real para digitalizar el servicio, siempre que la transición incluya un enfoque humano basado en la facilidad de uso y la formación de la comunidad.

¿Considera que la venta de boletos en línea facilitaría su trabajo?

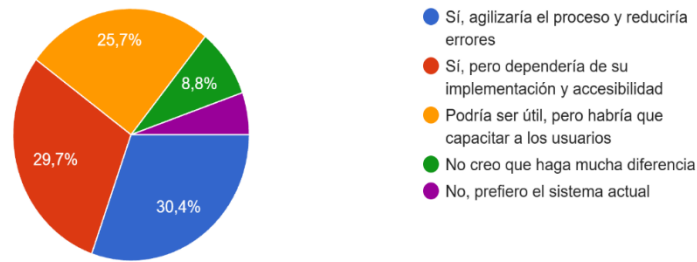


Figura 11. Pregunta 10

4.2. PROPUESTA

Desarrollar una aplicación móvil que centralice la administración del personal, conductores y administrativos en la venta de boletos y talento humano en la compañía Trans Doramald.

Extreme Programming (XP) se adoptó como marco metodológico por su énfasis en la adaptabilidad al cambio, la programación en pares y la retroalimentación continua con el cliente, elementos críticos para el desarrollo de una solución tecnológica. La metodología se estructuró en seis fases que combinan las prácticas técnicas de XP con las necesidades específicas de la compañía de transporte, manteniendo la entrega de valor funcional como eje central del proceso.

4.2.1. Fase de planificación

Durante la sesión de apertura del proyecto, se establecieron formalmente los requerimientos técnicos y funcionales que regirán el desarrollo del sistema se centró en diseñar una plataforma que no solo sea funcional y sencilla permitiendo alinear los objetivos estratégicos del departamento de talento humano con las necesidades operativas de la venta de boletos y gestión de encomiendas, asegurando que la herramienta tecnológica sea intuitiva, eficiente y totalmente adaptable a la normativa de la entidad pública.

El software está diseñado para facilitar la venta de boletos mediante rutas directas y la gestión de envíos con seguimiento en tiempo real, garantizando que tanto la administración del personal se gestione de forma eficiente y centralizada. De esta manera, se prioriza una herramienta capaz de modernizar tanto el control del personal interno como los servicios de transporte y encomiendas ofrecidos al público.

4.2.1.1. Levantamiento de procesos

Identificación de flujos operativos críticos mediante entrevistas estructuradas al personal de Trans Doramald, resultando en 8 historias de usuario priorizadas (Tabla 7), entre ellas la reserva de boletos, gestión de encomiendas y administración de talento humano.

Tabla 7. Historia del usuario

Historia de Usuario	
Número	1
Nombre Historia	Reserva y Pago de Boletos en Línea
Prioridad	Alta
Funcionalidad	Como usuario final, quiero seleccionar mi destino, asiento y método de pago para asegurar mi viaje de forma anticipada y sin errores.
Criterio de aceptación	El sistema debe permitir elegir origen, destino, fecha y hora del viaje. Debe mostrar un mapa visual de asientos para seleccionar disponibilidad en tiempo real.
Número	2
Nombre Historia	Registro y Seguimiento de Encomiendas
Prioridad	Alta
Funcionalidad	Como remitente, quiero registrar los datos de mi paquete y destinatario para gestionar el envío de mercancía a través de las rutas de la compañía.
Criterio de aceptación	El registro debe quedar vinculado a una ruta y bus específico para su logística. El sistema debe solicitar descripción del contenido, destino, monto pagado y datos de contacto de ambas partes
Número	3
Nombre Historia	Administración de Personal y Unidades
Prioridad	Alta
Funcionalidad	Como administrador, quiero gestionar la información de los choferes y buses para garantizar el control operativo y el bienestar del personal. El sistema debe permitir registrar y editar datos personales de los Choferes.
Criterio de aceptación	El administrador debe poder asignar choferes a unidades específicas el control de turnos
Número	4
Nombre Historia	Administración de trayectorias y paradas
Prioridad	Alta
Funcionalidad	Como administrador, quiero añadir, editar o eliminar trayectorias y optimizar el servicio según la demanda de las rutas en El Chical y Maldonado.
Criterio de aceptación	El sistema debe permitir ingresar nombre de la ruta, puntos de origen, destino y paradas intermedias.
Número	5
Nombre Historia	Emisión de avisos
Prioridad	Alta
Funcionalidad	Como administrador, quiero redactar y enviar avisos específicos e informar a los usuarios sobre cambios de horarios, rutas o confirmaciones de servicio.
Criterio de aceptación	Se debe permitir la selección de destinatarios Los mensajes deben aparecer en el panel de notificaciones de la aplicación del usuario en tiempo real
Número	6
Nombre Historia	Registro técnico de unidades y personal
Prioridad	Alta
Funcionalidad	Como administrador, quiero registrar los datos técnicos de los buses y datos de choferes para mantener un inventario actualizado

Criterio de aceptación	de la capacidad operativa El sistema debe solicitar número de unidad, placa, capacidad de asientos y nombres del chofer.
Número	7
Nombre Historia	Aprobación de transacciones digitales
Prioridad	Alta
Funcionalidad	Como encargado financiero, quiero verificar los comprobantes transferencia cargados para aprobar o rechazar la compra de boletos
Criterio de aceptación	El sistema debe listar los pagos pendientes con su respectiva imagen del comprobante.
Número	8
Nombre Historia	Consulta de historial y generación de reportes
Prioridad	Media
Funcionalidad	Como usuario o administrador, quiero consultar registros de viajes pasados para llevar un control de los servicios realizados.
Criterio de aceptación	Debe permitir filtrar la búsqueda por rangos de fechas e ID de usuario. El sistema debe permitir descargar los recibos o reportes en formato digital (PDF)

4.2.1.2. Requerimientos funcionales

Definición de 12 requerimientos funcionales (Tabla 8 — RF001 a RF012) que cubren autenticación, bloqueo transaccional de asientos, validación de pagos y gestión administrativa; complementados con 12 requerimientos no funcionales (Tabla 9 — RNF001 a RNF012) enfocados en usabilidad rural, operación offline (RNF005) e integridad de datos (RNF006).

Tabla 8. Requerimientos Funcionales

Requerimientos Funcionales	
Id requerimiento	RF001
Nombre	Registro de usuarios
Prioridad	Alta
Descripción	Permite crear cuentas con email y contraseña para que los usuarios de El Chical y Maldonado puedan acceder al sistema de gestión de transporte.
Entrada	Correo electrónico válido, contraseña nombres completos y número de identificación.
Procesos	Verificación de que el usuario no exista previamente en la base de datos. Notificación de éxito en
Salida	pantalla, correo de confirmación enviada al usuario y redirección al inicio de sesión
Id requerimiento	RF002
Nombre	Autenticación y seguridad
Prioridad	Alta
Descripción	Los usuarios deben registrarse mediante correo y contraseña o a través de cuentas de Google para garantizar un acceso seguro y rápido a la plataforma.
Entrada	Credenciales de usuario (email/password) o token de autenticación de Google.
Procesos	Validación de credenciales contra la base de datos encriptada
Salida	Acceso concedido al panel del usuario o mensaje de error en caso de credenciales incorrectas.
Id requerimiento	RF003
Nombre	Reserva de boleto

Prioridad	Alta
Descripción	El sistema permite seleccionar destino, bus, paradas y asientos. Incluye el registro del pasajero y la elección entre transferencia o pago en efectivo. Datos del viaje (origen/destino, fecha, hora), selección de asiento en mapa visual, datos del pasajero (Cédula/Nombre) y comprobante de pago (si es transferencia).
Entrada	Bloqueo temporal de asientos seleccionados para evitar duplicidad.
Procesos	Verificación de comprobante (para transferencias) o generación de orden de pago (para efectivo en punto físico).
Salida	Comprobante de reserva digital (PDF/PNG) de reserva para ser validado por el encargado físico o conductor
Id requerimiento	RF004
Nombre	Envío de Encomienda.
Prioridad	Alta
Descripción	El sistema permite registrar la descripción del paquete, definir el destino, registrar el monto pagado e ingresar los datos del remitente y destinatario. Descripción del contenido (tipo de encomienda), destino final, datos del destinatario (Nombres Completos), monto pagado y datos del remitente.
Entrada	Registro detallado de la encomienda en el sistema vinculado a una ruta espec
Procesos	Comprobante de envío físico o digital y etiquetas para el paquete.
Salida	
Id requerimiento	RF005
Nombre	Historial de boletos.
Prioridad	Media
Descripción	Permite al usuario consultar sus viajes pasados y futuros mediante la selección de fechas, con la opción de visualizar detalles y descargar el recibo de pago
Entrada	Rango de fechas de búsqueda e ID de usuario.
Procesos	Filtrado por estado (Pagado, Pendiente, Cancelado).
Salida	Listado de boletos en pantalla y archivo digital (recibo) descargable.
Id requerimiento	RF006
Nombre	Notificación
Prioridad	Alta
Descripción	El sistema permite observar y gestionar las notificaciones relacionadas con el estado de reservas, validación de comprobantes de pago, asignación de asientos y registro de encomiendas.
Entrada	Actualizaciones de estado en la base de datos (pago aprobado, reserva confirmada, encomienda en camino) y preferencias de alerta del usuario.
Procesos	Monitoreo en tiempo real de cambios en el estado de boletos y encomiendas.
Salida	Envío de avisos sobre cambios en horarios o rutas para mitigar la falta de información.
Id requerimiento	Alertas visuales dentro de la aplicación mensajes de confirmación y actualizaciones en el panel principal del usuario.
Nombre	RF007
Prioridad	Gestión de rutas. (Administrador.)
Descripción	Alta
Entrada	Permite al administrador del sistema añadir nuevas trayectorias, Eliminar rutas obsoletas o editar las existentes para optimizar el servicio.
Procesos	Nombre de la ruta, puntos de origen y destino, paradas intermedias. Registro de la disponibilidad de rutas para los usuarios.
Salida	Confirmación de ruta creada/editada exitosamente y actualización inmediata en el catálogo que ven los pasajeros.
Id requerimiento	RF008
Nombre	Crear notificaciones
Prioridad	Alta
Descripción	Permite al administrador redactar y enviar avisos masivos o específicos a todos los usuarios registrados sobre cambios en el servicio o confirmaciones.
Entrada	Avisos, selección de destinatarios (todos o grupos específicos) y programación de fecha/hora de envío.
Procesos	Redacción del contenido informativo sobre rutas, horarios o encomiendas

Salida	Mensaje emitido con éxito hacia los paneles de usuario y registro en la sección de avisos del sistema.
Id requerimiento	RF009
Nombre	Gestión de buses.
Prioridad	Alta
Descripción	Permite registrar y administrar las unidades de transporte, detallando su identificación, capacidad de asientos, información del conductor y logística de salida (hora y lugar).
Entrada	Número de unidad, capacidad total de asientos, nombres del chofer, hora de salida y lugar de partida.
Procesos	Vinculación del bus con una ruta específica gestionada en el RF007.
Salida	Unidad dada de alta en el sistema y disponible para que los usuarios que prefieren el autobús puedan realizar sus reservas.
Id requerimiento	RF0010
Nombre	Gestión de imágenes.
Prioridad	Media
Descripción	El sistema permite al administrador cargar, editar y actualizar las imágenes de la pantalla principal y de las secciones de información para mantener al día a los usuarios.
Entrada	Archivos de imagen (formatos JPG, PNG), etiquetas de descripción y selección de la sección de destino (Inicio, Info, Rutas).
Procesos	Reemplazo de imágenes antiguas por las versiones actualizadas.
Salida	Interfaz del sistema actualizada con la nueva información visual para que la población pueda consultarla.
Id requerimiento	RF011
Nombre	Verificación de pagos.
Prioridad	Alta
Descripción	Permite aprobar o rechazar la compra de Boletos. Visualizar pendientes aprobados y Rechazados. Visualización de reportes
Entrada	Comprobantes de transferencia cargados por usuarios, ID de transacción, y filtros de fecha para reportes.
Procesos	Revisión visual de comprobantes de pago versus el monto registrado.
Salida	Notificación al usuario sobre el estado de su pago y generación de documentos de reporte para la administración.
Id requerimiento	RF012
Nombre	Gestión de datos.
Prioridad	Alta
Descripción	Módulo de administración para el control detallado de la flota, incluyendo identificación de unidades, capacidad operativa, asignación de personal y logística de terminales.
Entrada	Placas de los vehículos, números de unidad, capacidad máxima de asientos, datos personales de choferes y catálogo de puntos de salida.
Procesos	Vinculación de choferes a unidades específicas para control de turnos.
Salida	Inventario actualizado de la flota y disponibilidad de recursos reflejada en el sistema de ventas.

4.2.1.3. Requerimientos no funcionales.

Tabla 9. Requerimientos no Funcionales

Id requerimiento	RNF001
Nombre	Usabilidad
Prioridad	Alta
Descripción	La interfaz debe ser fácil de usar para personas con poca experiencia digital.
Entrada	Interacción del usuario con botones y menús
Procesos	Simplificación de pasos para reducir la carga cognitiva.
Salida	Navegación fluida y sin errores de usuario.
Id requerimiento	RNF002
Nombre	Disponibilidad
Prioridad	Alta
Descripción	El sistema debe estar activo todo el tiempo para garantizar el servicio.
Entrada	Peticiones de acceso de los usuarios.
Procesos	Monitoreo constante del estado del servidor.
Salida	Acceso ininterrumpido a la plataforma
Id requerimiento	RNF003
Nombre	Seguridad
Prioridad	Alta
Descripción	Protección de datos personales y cifrado de contraseñas de los usuarios.
Entrada	Datos sensibles ingresados (email/password).
Procesos	Encriptación de datos mediante protocolos de seguridad.
Salida	Información protegida contra accesos no autorizados.
Id requerimiento	RNF004
Nombre	Rendimiento
Prioridad	Media
Descripción	Las consultas de rutas y horarios deben responder en menos de 2 segundos.
Entrada	Consultas de búsqueda de viajes.
Procesos	Optimización de peticiones a la base de datos.
Salida	Resultados de búsqueda mostrados de forma inmediata
Id requerimiento	RNF005
Nombre	Accesibilidad
Prioridad	Alta
Descripción	Optimización para funcionar en zonas con baja señal de internet (Chical/Maldonado).
Entrada	Carga de la aplicación en redes móviles lentas.
Procesos	Compresión de imágenes y recursos ligeros.
Salida	Visualización rápida del contenido en cualquier red.
Id requerimiento	RNF006
Nombre	Integridad
Prioridad	Alta
Descripción	Evitar la duplicidad de reservas en un mismo asiento y bus.
Entrada	Selección de asiento por múltiples usuarios.
Procesos	Bloqueo transaccional de registros en tiempo real.
Salida	Un solo boleto emitido por cada asiento físico.
Id requerimiento	RNF007
Nombre	Escalabilidad
Prioridad	Media
Descripción	Capacidad para soportar un aumento en el número de unidades y usuarios.
Entrada	Nuevos registros de buses y pasajeros.
Procesos	Estructura de base de datos.
Salida	Sistema estable ante el crecimiento de la compañía.
Id requerimiento	RNF008
Nombre	Portabilidad
Prioridad	Media

Descripción	El sistema debe ser compatible con navegadores web y dispositivos móviles.
Entrada	Acceso desde smartphones.
Procesos	Adaptación de la interfaz (Diseño Responsive).
Salida	Visualización correcta en cualquier pantalla.
Id requerimiento	RNF009
Nombre	Mantenibilidad
Prioridad	Media
Descripción	El código debe permitir actualizaciones fáciles de rutas y horarios.
Entrada	Cambios realizados por el administrador.
Procesos	Estructura de código modular y documentada.
Salida	Actualizaciones aplicadas sin afectar el sistema global.
Id requerimiento	RNF010
Nombre	Fiabilidad
Prioridad	Alta
Descripción	En errores de pago, el sistema debe asegurar que no se pierda la información.
Entrada	Intentos de pago fallidos o interrumpidos.
Procesos	Reversión automática de procesos incompletos.
Salida	Datos consistentes y sin transacciones fantasma.
Id requerimiento	RNF011
Nombre	Legibilidad
Prioridad	Alta
Descripción	Textos y avisos claros para cumplir con la preferencia de información física.
Entrada	Contenido de notificaciones y carteles informativos.
Procesos	Aplicación de contrastes altos y fuentes grandes.
Salida	Información fácil de leer para personas de toda edad.
Id requerimiento	RNF012
Nombre	Privacidad
Prioridad	Alta
Descripción	cumplimiento con normas legales de protección de datos de los pasajeros.
Entrada	Información de registro y documentos.
Procesos	Restricción de acceso a datos privados del usuario.
Salida	Confidencialidad total de la información personal.

4.2.2. Fase de diseño

Esta fase tradujo los requerimientos en modelos técnicos concretos, aplicando el principio XP de diseño simple y estableciendo la metáfora del sistema como "Boletero Digital Inteligente", facilitando la comunicación entre el equipo técnico y los usuarios no técnicos de la compañía.

Elaboración de diagramas de clases (Figura 12) que definen las entidades del sistema (Usuario, Bus, Reserva, Encomienda) y sus relaciones; diagramas de casos de uso que especifican las interacciones por rol (pasajero, conductor, administrador); y diagramas de secuencia que modelan flujos críticos como el bloqueo de asientos y la compra de boletos.

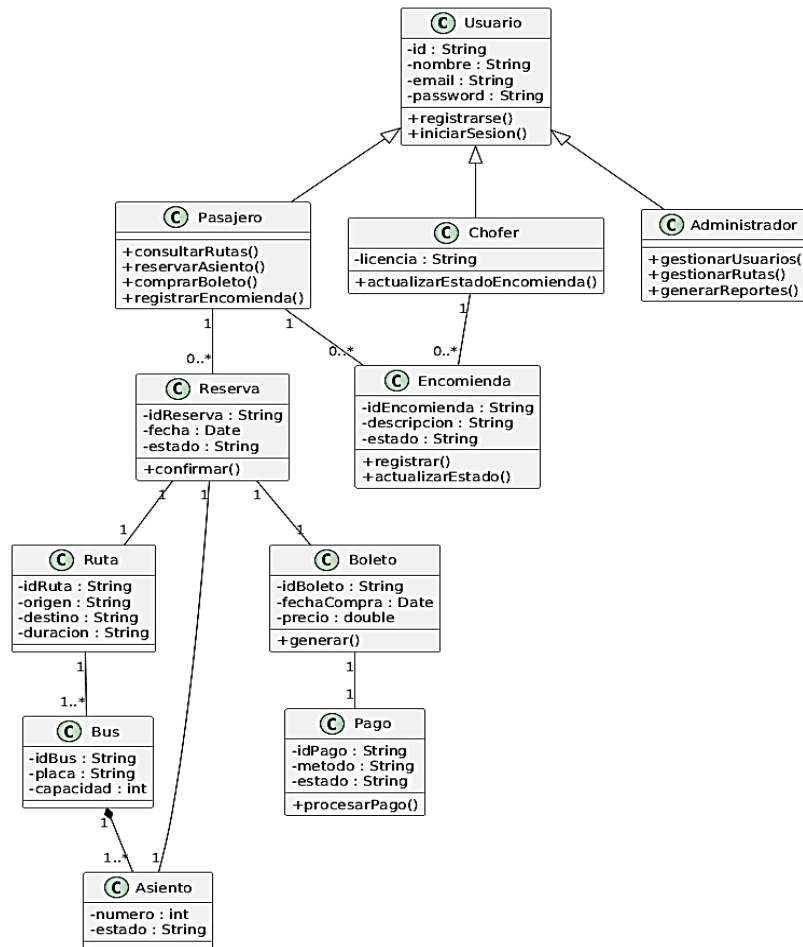


Figura 12. Clases

El diagrama de clases representa la estructura estática del sistema Trans Doramald, identificando las principales entidades del dominio, sus atributos, métodos y relaciones. Se emplea herencia para modelar los diferentes tipos de usuarios, y asociaciones para representar los procesos de reserva, compra de boletos y gestión de encomiendas. Este diagrama permite comprender la organización interna del sistema y sirve como base para la implementación.

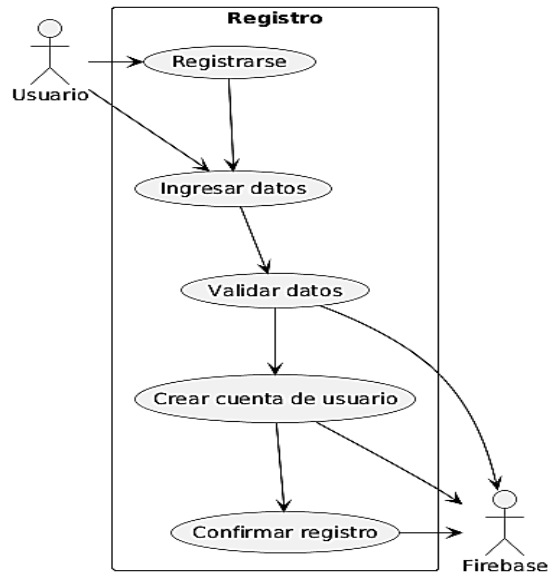


Figura 13. Caso de uso – registro

El diagrama de casos de uso del registro de usuario representa la interacción entre el usuario y el sistema durante el proceso de creación de una cuenta. El usuario inicia el proceso ingresando sus datos personales, los cuales son validados por el sistema antes de proceder a la creación de la cuenta. Para la gestión de autenticación y almacenamiento de la información, el sistema interactúa con el servicio externo Firebase, garantizando un manejo seguro y centralizado de los datos del usuario.

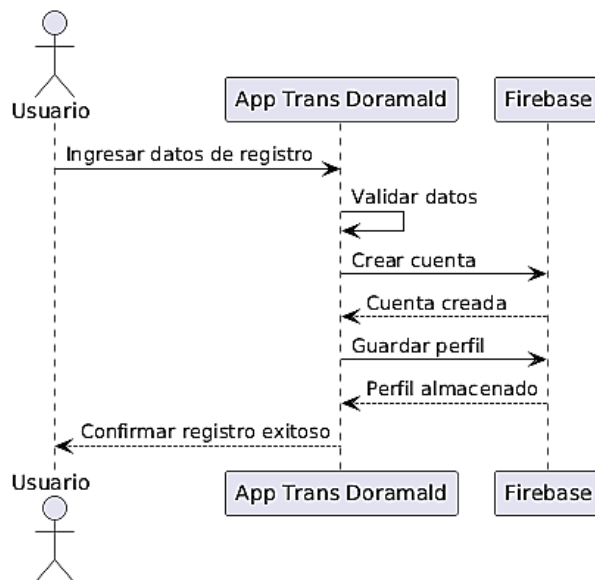


Figura 14. Secuencia – registro

El diagrama de secuencia del registro de usuario describe el flujo temporal de interacciones entre el usuario, el sistema y Firebase durante la creación de una

cuenta. El sistema valida los datos ingresados, solicita la creación de la cuenta al servicio externo y almacena el perfil del usuario. Finalmente, el sistema notifica al usuario sobre el registro exitoso.

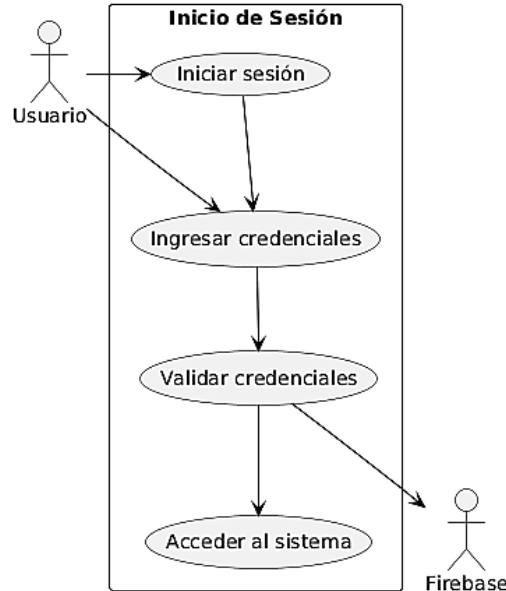


Figura 15. Caso de uso - inicio de sesión

Este diagrama de casos de uso describe el proceso de inicio de sesión del sistema. El usuario ingresa sus credenciales, las cuales son verificadas por el sistema mediante el servicio externo Firebase. Una vez autenticado correctamente, el sistema permite el acceso a las funcionalidades disponibles según el rol del usuario. Este diagrama delimita claramente la responsabilidad del sistema en la validación de credenciales y el uso de servicios externos para la autenticación.

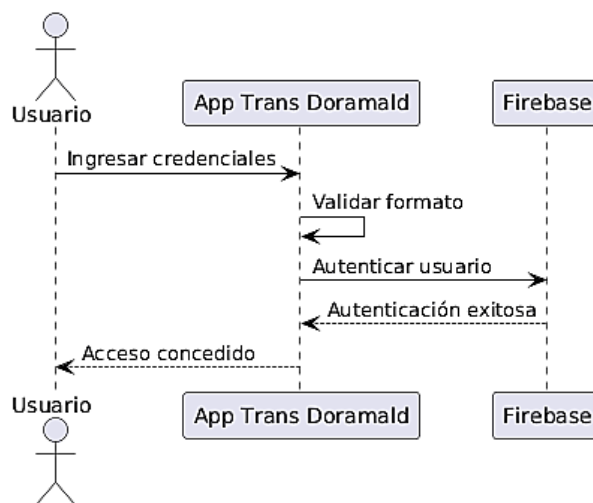


Figura 16. Secuencia - inicio de sesión

Este diagrama de secuencia muestra el proceso de autenticación de un usuario. El usuario envía sus credenciales al sistema, el cual valida el formato y delega la autenticación a Firebase. Una vez recibida la confirmación, el sistema concede el acceso correspondiente al usuario.

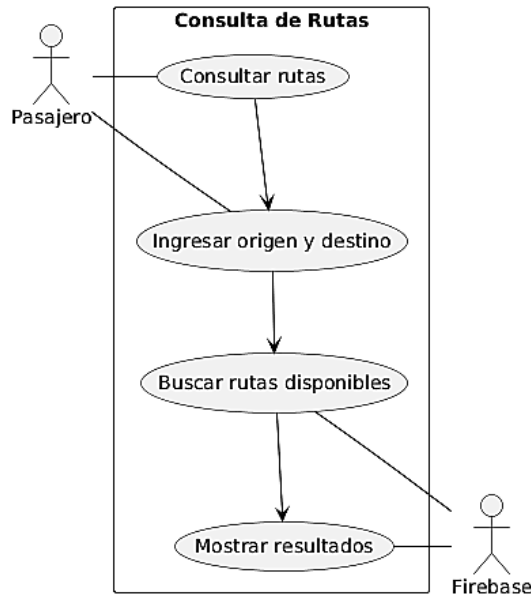


Figura 17. Caso de uso - consulta de rutas

El diagrama de casos de uso de consulta de rutas muestra cómo el pasajero interactúa con el sistema para obtener información sobre las rutas disponibles. El pasajero ingresa el origen y destino del viaje, y el sistema consulta la base de datos en Firebase para recuperar las rutas que coinciden con los criterios ingresados. Finalmente, el sistema presenta los resultados al pasajero, permitiendo la planificación del viaje.

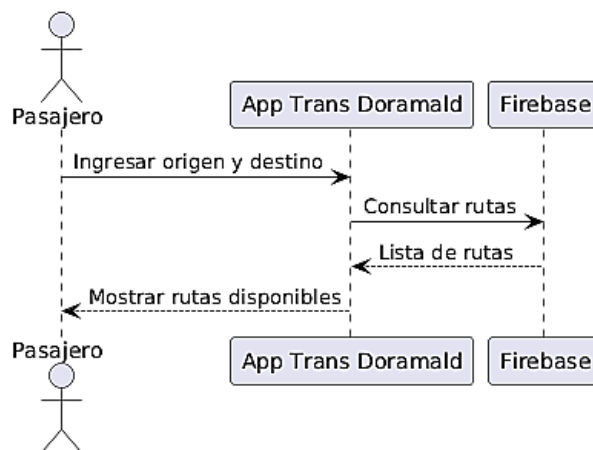


Figura 18. Secuencia - consulta de rutas

El diagrama de secuencia de consulta de rutas representa la interacción entre el pasajero, el sistema y Firebase para la obtención de rutas disponibles. El sistema actúa como intermediario, consultando la base de datos y devolviendo los resultados al pasajero de manera ordenada.

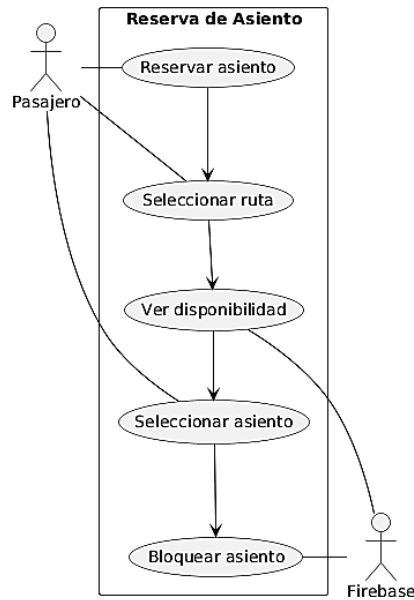


Figura 19. Caso de uso - reserva de asiento

Este diagrama de casos de uso representa el proceso de reserva de asientos. El pasajero selecciona una ruta y consulta la disponibilidad de asientos, información que es obtenida desde Firebase. Una vez seleccionado el asiento, el sistema procede a bloquearlo temporalmente para evitar conflictos con otros usuarios, asegurando así la integridad del proceso de reserva.

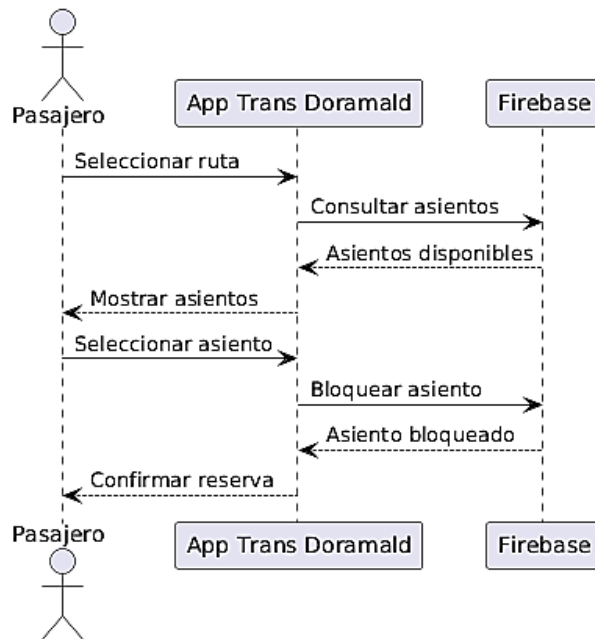


Figura 20. Secuencia - reserva de asiento

Este diagrama de secuencia describe el flujo de mensajes durante el proceso de reserva de un asiento. El sistema consulta la disponibilidad de asientos, permite la selección por parte del pasajero y bloquea el asiento seleccionado en la base de datos, asegurando la correcta gestión de la reserva.

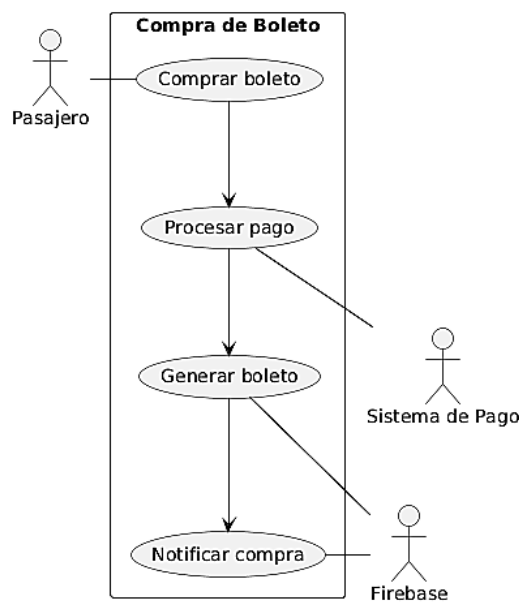


Figura 21. Caso de uso - compra de boleto

El diagrama de casos de uso de compra de boleto describe el proceso mediante el cual el pasajero confirma una reserva y realiza el pago correspondiente. El sistema gestiona la interacción con el sistema de pago externo para procesar la transacción

y, en caso de éxito, genera el boleto y registra la compra en la base de datos. Este diagrama permite visualizar claramente la integración de servicios externos dentro del flujo principal del sistema.

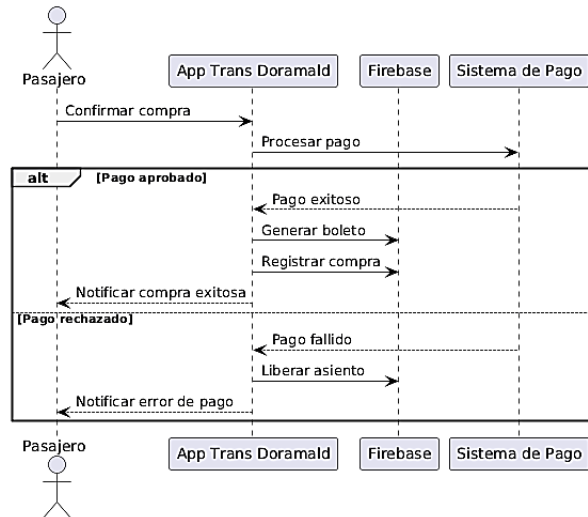


Figura 22. Secuencia - compra de boleto

El diagrama de secuencia de compra de boleto muestra el flujo completo de la transacción. El sistema gestiona el proceso de pago mediante un sistema externo y, dependiendo del resultado, genera el boleto y registra la compra o libera el asiento reservado. Este diagrama evidencia el manejo de escenarios alternos dentro del sistema.

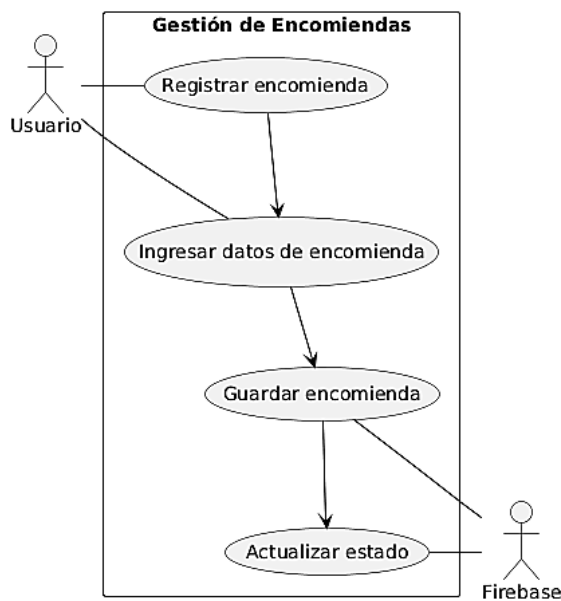


Figura 23. Caso de uso – encomienda

Este diagrama de casos de uso muestra el proceso de gestión de encomiendas dentro del sistema. El usuario es responsable de registrar la encomienda, mientras que el chofer interactúa posteriormente para actualizar el estado de la misma durante el transporte. El sistema utiliza Firebase para almacenar y actualizar la información de las encomiendas, garantizando su trazabilidad.

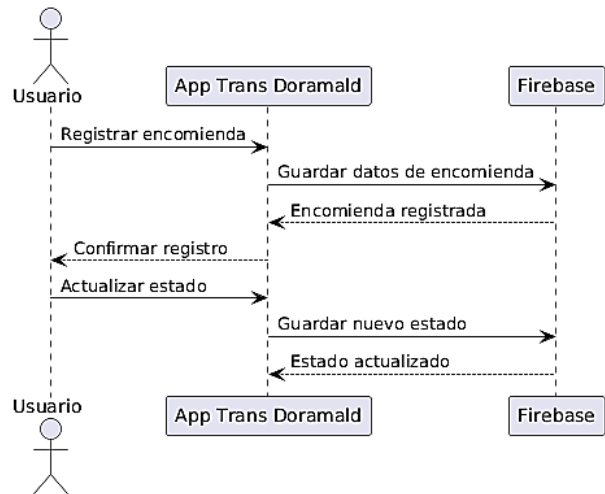


Figura 24. Secuencia – encomienda

Este diagrama de secuencia representa el proceso de registro y actualización de encomiendas. El usuario registra la encomienda y el sistema almacena la información en Firebase. Posteriormente, el chofer actualiza el estado de la encomienda, permitiendo un seguimiento continuo del envío.

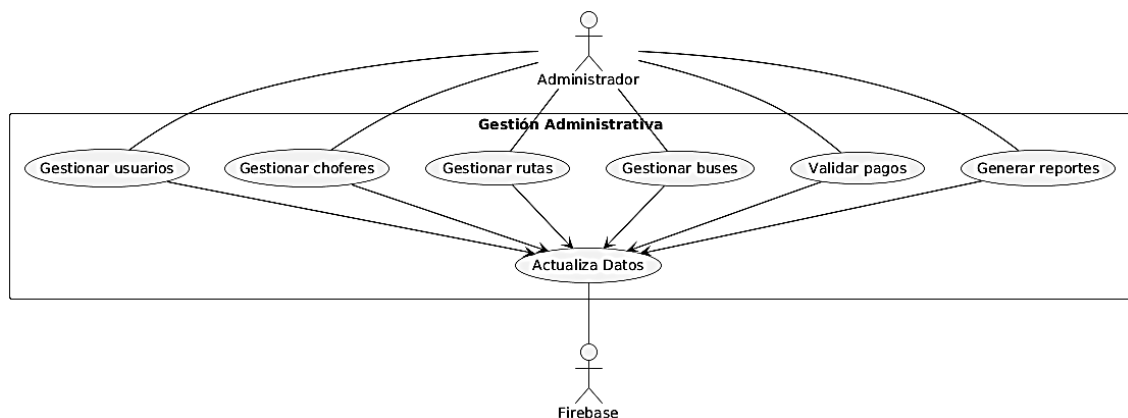


Figura 25. Caso de uso - gestión administrativa

El diagrama de casos de uso de gestión administrativa representa las funcionalidades disponibles para el administrador del sistema. Estas incluyen la gestión de usuarios, choferes, rutas, buses y pagos, así como la generación de reportes. El sistema

centraliza estas operaciones y utiliza Firebase para el acceso y manipulación de los datos necesarios para la toma de decisiones administrativas.

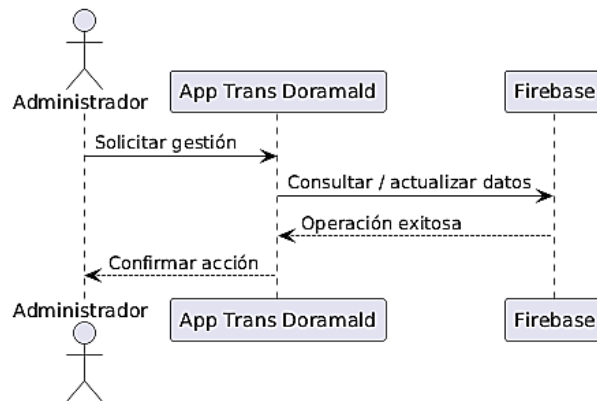


Figura 26. Secuencia - gestión administrativa

El diagrama de secuencia de gestión administrativa muestra la interacción entre el administrador, el sistema y la base de datos durante las operaciones de administración. El sistema procesa las solicitudes del administrador y realiza las consultas o actualizaciones necesarias, confirmando posteriormente la ejecución de las acciones.

Diseño de la base de datos NoSQL en Firebase Cloud Firestore (Figura 27), estructurada en colecciones jerárquicas que garantizan la sincronización offline y la integridad referencial mediante el UID del usuario.

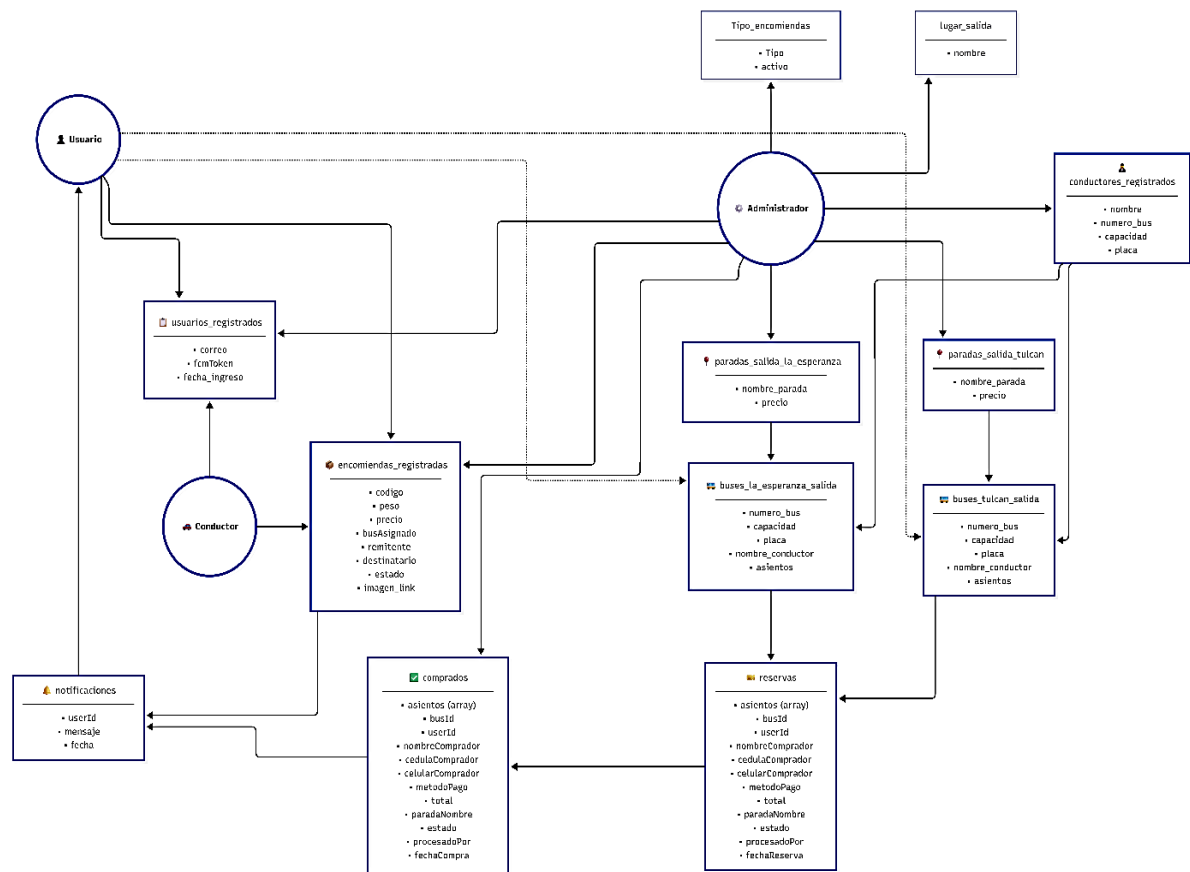


Figura 27. Conexión de la base de datos

La arquitectura de datos implementada en Firebase Cloud Firestore, una base de datos NoSQL documental. A diferencia de las bases relacionales tradicionales, la estructura se organiza en colecciones de documentos JSON jerárquicos. La colección usuarios registrados actúa como nodo central, vinculándose mediante el uid las subcolecciones. Esta arquitectura facilita la sincronización en tiempo real y el funcionamiento offline mediante el caché local del SDK de Flutter, característica crítica para operar en zonas rurales con conectividad intermitente.

4.2.3. Fase de Codificación

Esta fase es el núcleo constructivo del sistema, rigurosamente aplicando la práctica XP de pair programming: los desarrolladores trabajaron en estación compartida alternando roles de conductor y navegante, garantizando revisión de código inmediata y transferencia de conocimiento continua.

4.2.3.1. Arquitectura del Sistema

Se adoptó una arquitectura basada en Clean Architecture adaptada a Flutter, dividida en tres capas principales:

- Capa de Presentación: Widgets, pantallas y gestores de estado.
- Capa de Dominio: Lógica de negocio, entidades y casos de uso (validación de asientos, cálculo de tarifas).
- Capa de Datos: Repositorios, modelos y fuentes de datos (Firebase Firestore, caché local).

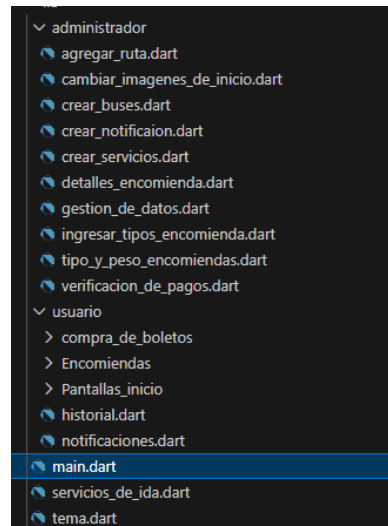


Figura 28. Árbol de directorios

4.2.3.2. Configuración de Firebase e Inicialización

El sistema requiere inicialización de Firebase para autenticación, base de datos en tiempo real (Firestore) y almacenamiento de comprobantes (Firebase Storage). La configuración principal se realiza en el archivo `main.dart`, implementando manejo de errores de conexión para garantizar disponibilidad en zonas rurales.

```
// ===== MAIN =====
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp();
  await initializeDateFormatting('es', null);

  FirebaseMessaging.onBackgroundMessage(firebaseMessagingBackgroundHandler);

  await requestPermissions();
  await initializeLocalNotifications();

  runApp(
    ChangeNotifierProvider(
      create: (_) => ThemeProvider(),
      child: const MyApp(),
    ),
  );
}
```

Figura 29. Inicialización de firebase.

4.2.3.3. Modelos de Datos y Entidades

Para garantizar la integridad referencial y evitar la duplicidad de boletos (RNF006), se implementaron modelos con validaciones transaccionales.

```
@override
Widget build(BuildContext context) {
  final db = FirebaseFirestore.instance;

  return StreamBuilder<DocumentSnapshot>{
    stream: db.collection('buses_tulcan_salida').doc(busId).snapshots(),
    builder: (context, snapshot) {
      int asientosOcupados = 0;

      if (snapshot.hasData && snapshot.data!.exists) {
        var data = snapshot.data!.data() as Map<String, dynamic>;
        if (data != null && data.containsKey('asientos')) {
          try {
            List<dynamic> asientosArray = data['asientos'];
            for (var asiento in asientosArray) {
              if (asiento is Map<String, dynamic> &&
                  asiento['estado'] == 'pagado') {
                asientosOcupados++;
              }
            }
          } catch (e) {
            print('Error al leer asientos: $e');
          }
        }
      }

      int asientosDisponibles = capacidad - asientosOcupados;
    }
  }
}
```

Figura 30. Lógica de reserva

4.2.3.4. Implementación del Sistema de Bloqueo de Asientos

Para cumplir con el requerimiento no funcional de integridad (RNF006), se desarrolló un servicio de transacciones en tiempo real. Este mecanismo garantiza que un asiento no pueda ser vendido simultáneamente a dos usuarios diferentes.

```

_reservaSubscription = db

var data = snapshot.data() as Map<String, dynamic>;
if (data == null) return;

List<Map<String, dynamic>> asientos =
    List<Map<String, dynamic>>.from(data['asientos'] ?? []);

bool hayaCambios = false;
for (var asiento in asientos) {
    int numero = asiento['numero'];
    var asientoCached = _asientosCache[numero];

    if (asientoCached == null ||
        asientoCached['estado'] != asiento['estado'] ||
        asientoCached['email'] != asiento['email']) {
        hayaCambios = true;
        _asientosCache[numero] = Map<String, dynamic>.from(asiento);
    }
}

if (!hayaCambios && _inicializado) return;

if (asientoSeleccionado != null && _inicializado) {
    var asiento = asientos.firstWhere(
        (a) => a['numero'] == asientoSeleccionado,
        orElse: () => {});
}

if (asiento.isNotEmpty &&
    asiento['email'] != null &&
    asiento['email'] != userEmail) {
    if (mounted && _pantallaActiva) {
        setState(() {
            asientoSeleccionado = null;
            asientoSeleccionadoUI = null;
        });
        _mostrarError('El asiento fue tomado por otro usuario');
    }
}
}

```

Figura 31. Disponibilidad de asientos

4.2.3.5. Interfaz de Usuario - Módulo de Reserva

La pantalla de selección de asientos se implementó utilizando GridView dinámico con actualización en tiempo real mediante StreamBuilder.

```

debugPrint('Actualizando asiento en bus...');

final busRef = db.collection('buses_tulcan_salida').doc(widget.busId);
final busDoc = await busRef.get();

if (!busDoc.exists) {
    throw Exception('Bus no encontrado en la base de datos');
}

var data = busDoc.data();
if (data == null) {
    throw Exception('Datos del bus no disponibles');
}

List<Map<String, dynamic>> asientos = [];
if (data['asientos'] != null) {
    asientos = List<Map<String, dynamic>>.from(data['asientos']);
} else {
    throw Exception('El bus no tiene asientos configurados');
}

final index =
    asientos.indexWhere((a) => a['numero'] == widget.asientoSeleccionado);

if (index == -1) {
    throw Exception(
        'Asiento ${widget.asientoSeleccionado} no encontrado en el bus');
}

// Verificar que el asiento esté en estado válido
final estadoActualAsiento = asientos[index]['estado'];
debugPrint('Estado actual del asiento: $estadoActualAsiento');

if (estadoActualAsiento != 'IntentandoReservar' &&
    estadoActualAsiento != 'disponible') {
    debugPrint('Asiento en estado: $estadoActualAsiento');
    // Continuar de todas formas si es el mismo usuario
    if (asientos[index]['email'] != widget.userEmail) {
        throw Exception('El asiento ya no está disponible');
    }
}
}

```

Figura 32. Reserva de asiento

4.2.3.6. Autenticación y Gestión de Roles

Se implementó autenticación dual: Google Sign-In para usuarios regulares (facilitando el acceso en zonas rurales donde memorizar contraseñas es problemático) y Email/Password con PIN para administradores (RF002).

```
void _signInWithEmail(BuildContext context) async {
  // ...
  final String password = _passwordController.text.trim();

  try {
    UserCredential userCredential = await _auth.signInWithEmailAndPassword(
      email: email,
      password: password,
    );

    await _firestore
      .collection('usuarios_registrados')
      .doc(userCredential.user!.uid)
      .set({
        'email': email,
        'createdAt': FieldValue.serverTimestamp(),
      }, SetOptions(merge: true));

    await _saveFCMTokenAfterLogin();
    _setupNotificationsStream();

    if (!mounted) return;

    Navigator.pop(context, true); // solo para admin

  } on FirebaseAuthException catch (e) {
    if (!mounted) return;

    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(
        content: Text('Error: ${e.message}'),
        backgroundColor: const Color(0xFFE44444),
      ),
    );
  }
}

void _signInWithGoogle() async {
  try {
    final GoogleSignInAccount? googleUser = await GoogleSignIn(
      scopes: ['email'],
    ).signIn();

    if (googleUser == null) return;

    final GoogleSignInAuthentication googleAuth =
      await googleUser.authentication;

    final AuthCredential credential = GoogleAuthProvider.credential(
      accessToken: googleAuth.accessToken,
      idToken: googleAuth.idToken,
    );

    final UserCredential userCredential =
      await _auth.signInWithCredential(credential);

    final User? user = userCredential.user;
    if (user != null) {
      await _firestore.collection('usuarios_registrados').doc(user.uid).set({
        // ...
      });
    }
  }
}
```

Figura 33. Autenticación dual

4.2.3.7. Gestión de Encomiendas (RF004)

El módulo de encomiendas requiere seguimiento de estados: Registrado, En tránsito, Entregado.

```

        _tipoEncomiendaSeleccionado = widget.encomiendaData.tipoEncomienda;
        _pesoController.text = widget.encomiendaData.rangoPeso ?? '';
        if (widget.encomiendaData.precioTipo != null) {
            _precioTipoController.text = widget.encomiendaData.precioTipo.toString();
        }
        _observacionesController.text = widget.encomiendaData.observaciones ?? '';
    }

    Future<void> _verificarUsuario() async {
        final user = FirebaseAuth.instance.currentUser;
        if (user != null) {
            setState(() {
                _isAdmin = user.email == 'admin@dominio.com';
            });
        }
    }

    Future<void> _cargarTiposEncomienda() async {
        setState(() => _isLoading = true);

        try {
            final tiposSnapshot = await FirebaseFirestore.instance
                .collection('Tipo_encomienda')
                .where('activo', isEqualTo: true)
                .get();

            _tiposEncomienda = tiposSnapshot.docs.map((doc) {
                final data = doc.data();
                return {
                    'id': doc.id,
                    'tipo': data['Tipo'] ?? data['tipo'] ?? 'Sin nombre',
                    'descripcion': data['descripcion'] ?? '',
                    'precio_adicional': 0.0,
                };
            }).toList();

            setState(() => _isLoading = false);
        } catch (e) {
            setState(() => _isLoading = false);
            _mostrarError('Error al cargar datos: $e');
        }
    }
}

```

Figura 34. Gestión encomiendas

4.2.3.8. Panel de Administración - Validación de Pagos (RF011)

La interfaz administrativa permite la gestión manual de comprobantes de transferencia bancaria.

```

class AdminVerificacionPagosScreen extends StatefulWidget {
    const AdminVerificacionPagosScreen({Key? key}) : super(key: key);

    @override
    _AdminVerificacionPagosScreenState createState() =>
        _AdminVerificacionPagosScreenState();
}

class _AdminVerificacionPagosScreenState
    extends State<AdminVerificacionPagosScreen> {
    final FirebaseFirestore db = FirebaseFirestore.instance;
    final FirebaseFunctions functions = FirebaseFunctions.instance;

    String? cooperativaSeleccionada;
    String? busSeleccionado;
    String filtroEstado = 'pendiente_verificacion';
    String coleccionActual = 'reservas';

    // Paleta empresarial
    final Color primaryNavy = const Color(0xFF1A2332);
    final Color darkGray = const Color(0xFF2D3748);
    final Color lightGray = const Color(0xFFF77FAFC);
    final Color mediumGray = const Color(0xFF718096);

    static const String FCM_SERVER_KEY = 'TU_SERVER_KEY_AQUI';

    // Obtener colección según cooperativa
    String get coleccionBuses {
        if (cooperativaSeleccionada == 'la_esperanza') {
            return 'buses_la_esperanza_salida';
        } else if (cooperativaSeleccionada == 'tulcan') {
            return 'buses_tulcan_salida';
        }
        return 'buses_tulcan_salida';
    }
}

```

Figura 35. Validación de pagos

4.2.3.9. Notificaciones y Alertas (RF006, RF008)

El sistema implementa notificaciones locales (para modo offline) y push notifications (cuando hay conexión) para informar cambios de horarios críticos en rutas rurales.

```
class NotificacionesScreen extends StatefulWidget {
  final String userEmail;

  const NotificacionesScreen({
    Key? key,
    required this.userEmail,
  }) : super(key: key);

  @override
  _NotificacionesScreenState createState() => _NotificacionesScreenState()
}

class _NotificacionesScreenState extends State<NotificacionesScreen>
  with SingleTickerProviderStateMixin {
  final FirebaseFirestore db = FirebaseFirestore.instance;

  late AnimationController _fadeController;
  late Animation<double> _fadeAnimation;

  // Paleta de colores inspirada en transporte de buses
  final Color primaryBusBlue = const Color.fromARGB(255, 243, 248, 255);
  final Color accentOrange = const Color(0xFFEA580C);
  final Color darkNavy = const Color(0xFF0F172A);
  final Color roadGray = const Color(0xFF334155);
  final Color lightBg = const Color(0xFFF1F5F9);
  final Color textGray = const Color(0xFF475569);
  final Color successGreen = const Color(0xFF059669);
  final Color accentBlue = const Color(0xFF1E40AF);
  final Color mainRed = const Color(0xFF940016);
  final Color unreadBg = const Color(0xFFEF3C7);
  final Color borderColor = const Color(0xFFE5E7EB);

  @override
  void initState() {
    super.initState();
    timeago.setLocaleMessages('es', timeago.EsMessages());

    _fadeController = AnimationController(
      vsync: this,
      duration: const Duration(milliseconds: 800),
    );
  }
}
```

Figura 36. Sistema de notificaciones

4.2.3.10. Control de versiones

El desarrollo colaborativo de la aplicación Trans Doramald requirió la implementación de un sistema de control de versiones distribuido (DVCS) mediante Git, alojado en un repositorio remoto (GitHub). Esta práctica fue fundamental para gestionar la evolución concurrente del código fuente en Flutter/Dart, manteniendo la integridad del sistema ante modificaciones simultáneas en módulos críticos.

El sistema de control de versiones permitió la trazabilidad completa de cada cambio mediante commits semánticos, facilitando la generación automática de changelogs y el cumplimiento del requerimiento no funcional RNF009 (Mantenibilidad), al documentar explícitamente cada modificación en la arquitectura de Firebase o en la interfaz de usuario.

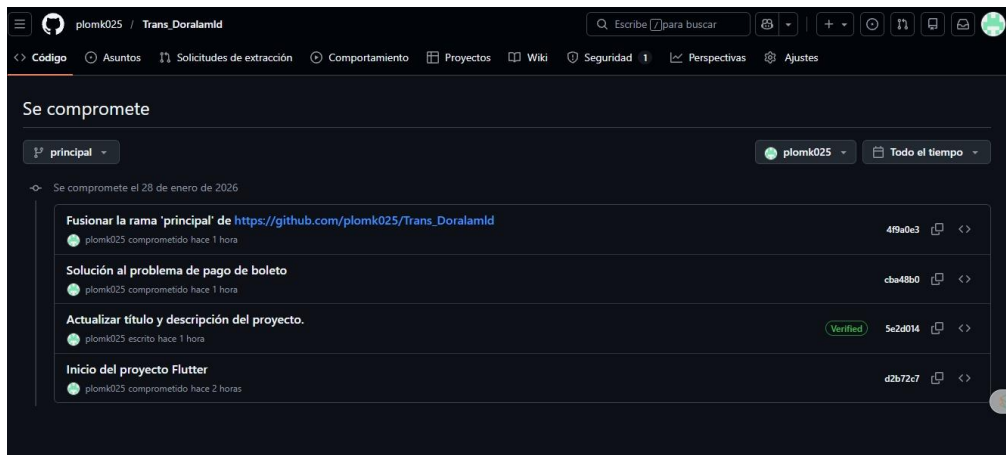


Figura 37. Control de versiones

4.2.3.11. Módulo de gestión de talento humano

El módulo de gestión de talento humano constituye uno de los componentes centrales de la herramienta Trans Doramald, siendo parte esencial del objetivo del proyecto tal como lo establece la historia de Usuario (Administración de Personal y Unidades) y el Requerimiento Funcional RF12 (Gestión de datos). Este módulo permite al administrador del sistema registrar, consultas, actualizar y gestionar la información del personal operativo de la compañía de transportes, garantizando un control centralizado de los conductores vinculados a cada unidad de la flota.

Previo al desarrollo de esta herramienta, la compañía Trans Doramald administraba la información de sus conductores de manera manual, utilizando registros físicos en papel y hojas de cálculo básicas sin sincronización entre las parroquias Maldonado y El Chical. Esta situación generaba inconsistencias en la asignación de choferes a unidades específicas, falta de control sobre la vigencia de licencias de conducción y dificultades para el seguimiento del personal operativo, problemáticas identificadas durante las entrevistas en la fase de levantamiento de requisitos.

La solución implementada centraliza toda la información del talento humano en Firebase, Firestore, integrándola directamente con los módulos de gestión de buses, reservas de boletos y encomiendas, de manera que cualquier cambio en los datos del personal se refleja en tiempo real en todo el sistema.

4.2.3.11.1. Funcionalidades implementadas

De acuerdo con los criterios de aceptación definidos en el RF012, el módulo de gestión de talento humano abarca las siguientes funcionalidades principales:

Registro de conductores: el administrador puede ingresar los datos Personales completos de cada chofer de la compañía, incluyendo nombres completos, número de cédula, Número de placa del vehículo, número de asientos.

Asignación de conductores a unidades: cada conductor puede ser asignado a un bus específico de la flota. Esta vinculación es bidireccional: al signar un chofer a un bus, el nombre del conductor que está asignado a la flota aparece y visible para los pasajeros la pantalla de selección de bus al momento de comprar su boleto.

Edición y actualización de datos: el administrador puede modificar cualquier campo de perfil del conductor en cualquier momento Los cambios se sincronizan en tiempo real con Firebase, Firestore.

Listado y consulta del personal activo: el módulo presenta un catálogo completo de todos los conductores registrados, con su estado (activo/ inactivo), la unidad asignada, la ruta que opera y los datos de contacto. Este catálogo es accesible exclusivamente desde el panel del administrador.

Desactivación de registros: en lugar de eliminar físicamente un registro del sistema, el administrador puede desactivar el perfil de un conductor que ya no opera en la compañía. Este mecanismo de eliminación preserva el historial operativo del conductor, manteniendo la trazabilidad de los viajes y encomiendas que gestionó.

4.2.3.11.2. Gestión de roles y perfiles del personal

Un elemento fundamental de la gestión del talento humano es el sistema de control de roles, que determinan funciones que puedan ejecutar cada tipo de usuario dentro de la aplicación. Se definieron tres roles de diferenciando, cada interfaz y un conjunto de permisos específicos:

Tabla 10. Roles del sistema Trans Doramald y permisos por perfil

Rol	Panel de Acceso	Funciones Disponibles en la App
Administrador	Panel de control exclusivo con autenticación dual (correo + PIN)	Gestión de choferes, buses, rutas, pagos, encomiendas, notificaciones, galería de imágenes y reportes. Acceso completo al módulo de talento humano.
Secretaria	Panel administrativo con autenticación por correo y contraseña institucional	Venta presencial de boletos en ventanilla, registro y recepción de comprobantes de pago por transferencia y gestión del historial de transacciones. No tiene acceso al módulo de talento humano ni a la configuración del sistema.
Chofer / Conductor	Dashboard simplificado de servicios activos (Figura 70)	Visualización de encomiendas asignadas a su unidad y actualización de estado de paquetes (Recogido, En tránsito, Entregado). No tiene acceso al panel administrativo.
Pasajero / Usuario final	Pantalla de servicios con autenticación Google.	Compra de boletos, historial de viajes, registro y seguimiento de encomiendas, recepción de notificaciones. Sin acceso a información del personal.

4.2.3.11.3. Síntesis del Módulo

La implementación del módulo de gestión de talento humano en la herramienta Trans Doramald resolvió directamente la problemática identificada en el diagnóstico inicial: la inexistencia de una plataforma digital que integre la administración del personal operativo con las operaciones de transporte en las parroquias rurales de Maldonado y El Chical. La centralización del registro de conductores en Firebase Firestore, combinada con la vinculación transaccional con la colección de buses y la implementación del sistema de roles, garantiza la coherencia de los datos en toda la plataforma.

El módulo cumple con todos los criterios de aceptación definidos en el RF012 y la Historia de Usuario, tal como se verificó en las pruebas de usabilidad realizadas con el personal administrativo de Trans Doramald. Los resultados evidenciaron que la tarea de "Registro conductor + vinculación unidad" fue completada, establecido para el perfil de administrador.

La transversalidad del módulo su integración con reservas, encomiendas, buses y notificaciones demuestra que la gestión digital del talento humano no es un componente aislado, sino el eje que articula la operación completa del servicio de

transporte en Trans Doramald, contribuyendo directamente al objetivo general de la investigación: desarrollar una herramienta tecnológica integral para la gestión de talento humano y la venta de boletos del transporte público.

4.2.4. Fase de pruebas

Esta fase implementó un plan de pruebas mixto que combina herramientas especializadas de métricas de rendimiento en dispositivos reales, validación técnica de backend y pruebas de usabilidad con usuarios finales, adaptándose a las limitaciones de infraestructura y diversidad de dispositivos de gama media/baja.

Se utilizó la herramienta Apptim (plataforma de mobile performance testing que permite medir métricas objetivas sin requerir scripting de automatización complejo) para evaluar el comportamiento real de la aplicación en condiciones de uso cotidiano:

Profiling de recursos: Medición objetiva del consumo de CPU, memoria RAM, batería y tráfico de red durante la ejecución de los flujos críticos (reserva de boleto, registro de encomienda, validación de pago), garantizando el cumplimiento de RNF004 (tiempos de respuesta < 2 segundos) en dispositivos con especificaciones limitadas.

Análisis de navegabilidad: Captura de métricas de user flows que identificaron cuellos de botella en la interfaz, tales como tiempos de carga excesivos en la carga de imágenes de comprobantes de pago, lo que motivó la optimización posterior mediante compresión (RNF005).

Validación de estabilidad: Detección de freezes (congelamientos de UI) y crashes durante la transición entre modo offline y online.

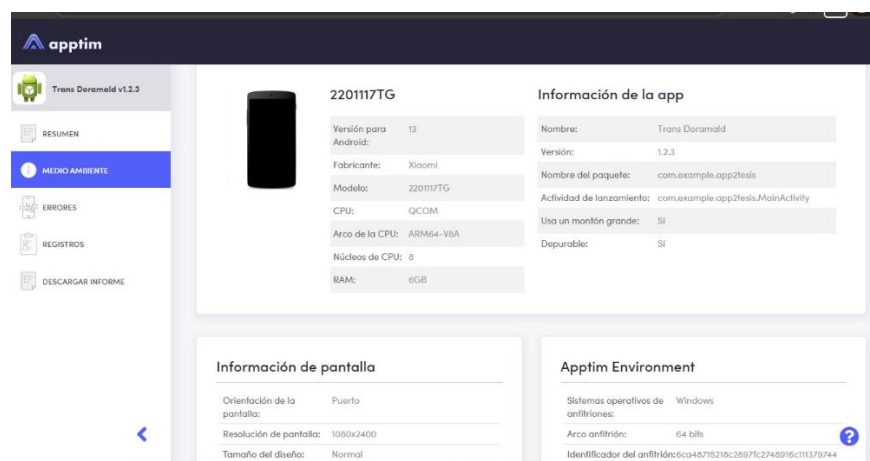


Figura 38. Entorno de pruebas (dispositivo gama media)

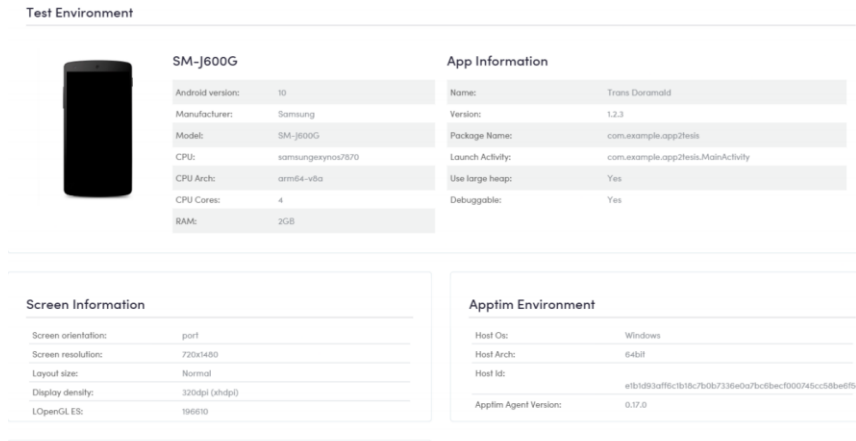


Figura 39. Entorno de pruebas (dispositivo gama baja)

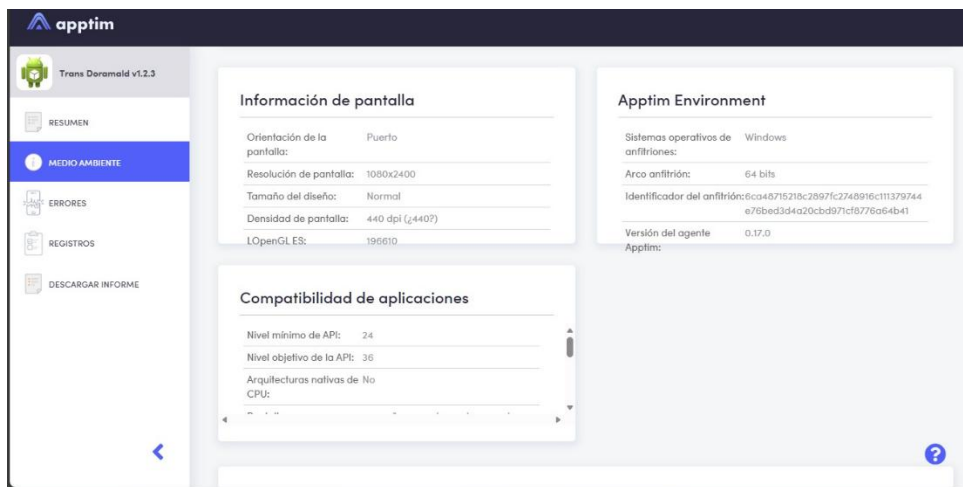


Figura 40. Información de compatibilidad de la aplicación

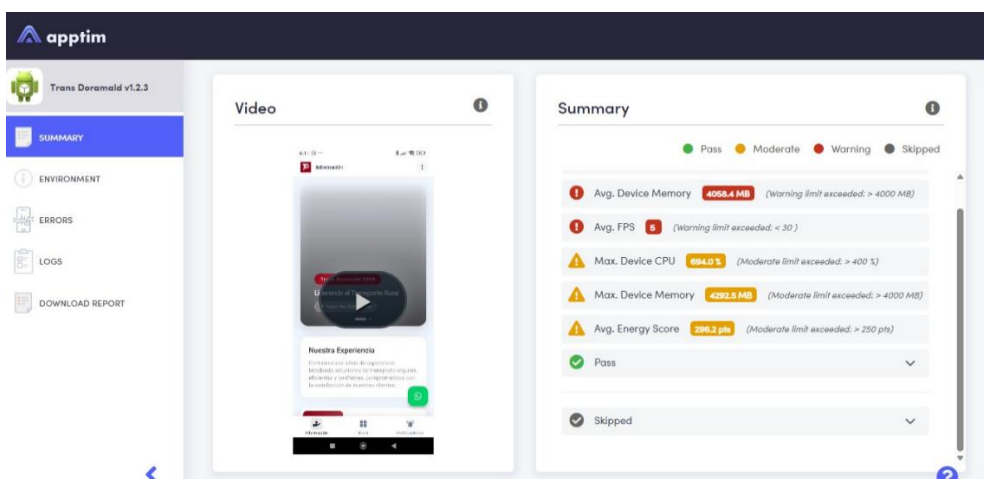


Figura 41. Resultados de las pruebas con obtención de pass (satisfactorio)



Figura 42. Métricas de uso de cpu, ram y memoria

Validación de Base de Datos y Reglas de Seguridad con Firebase Test Lab.

Se empleó Firebase Test Lab para la verificación automatizada de la capa de datos:

Pruebas de concurrencia y sincronización: Testeo de escenarios de múltiples usuarios simultáneos modificando documentos en Firestore, verificando que el mecanismo de bloqueo transaccional (RNF006) mantuviera la integridad referencial sin duplicidad de reservas.

Compatibilidad de dispositivos: Ejecución de la suite de pruebas en matriz de dispositivos físicos cloud (diferentes fabricantes, versiones Android 5.0-12, tamaños de pantalla variados) para garantizar RNF008 (portabilidad).

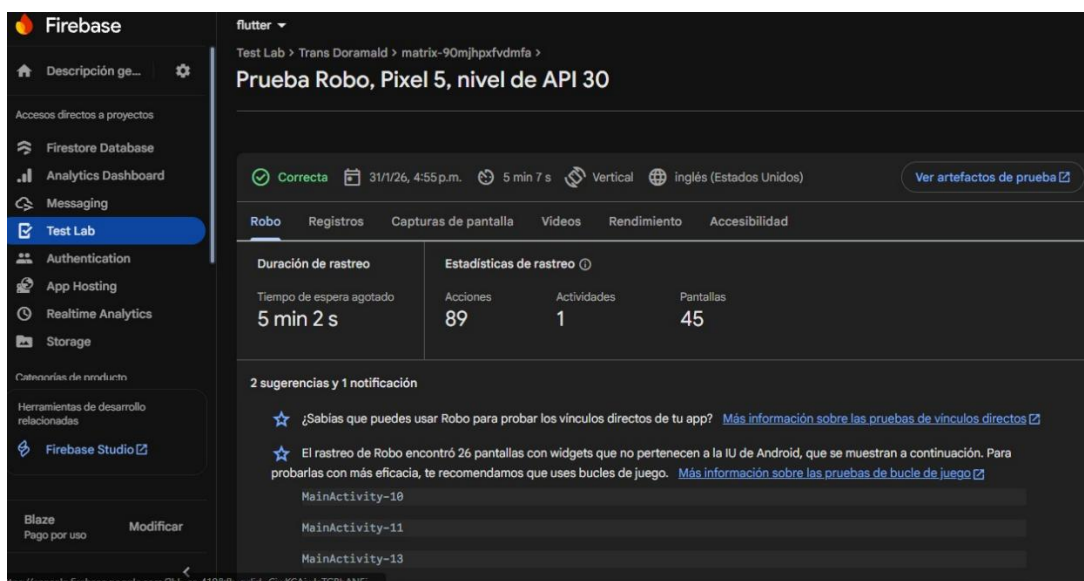


Figura 43. Pruebas de estrés y robo en la aplicación

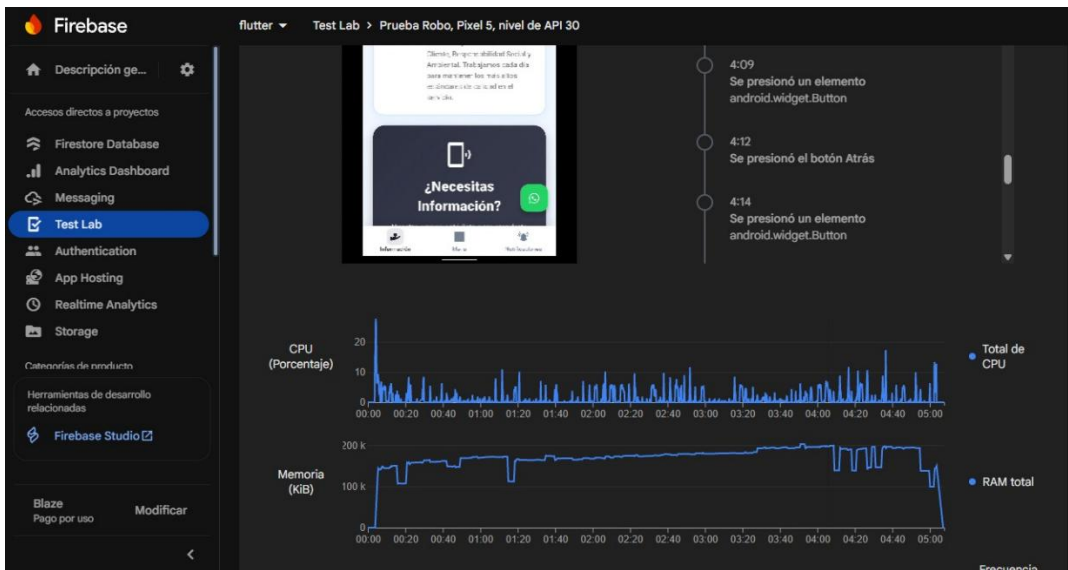


Figura 44. Uso de cpu y memoria ram (Test Lab)

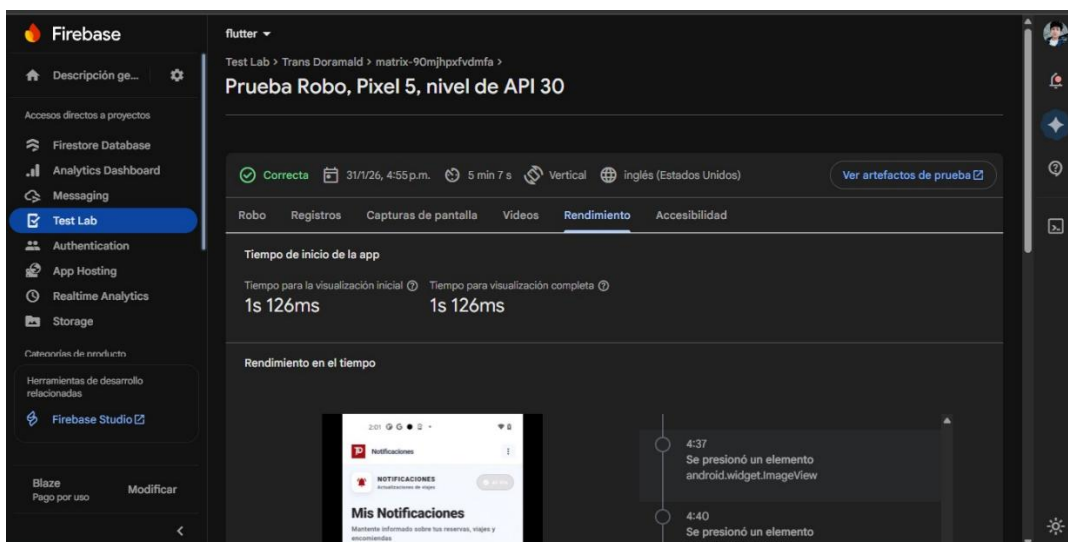


Figura 45. Resultados de rendimiento de la aplicación

Pruebas de Usabilidad Manuales con Usuarios Finales

Complementariamente, se ejecutaron pruebas manuales presenciales con 17 usuarios (Tabla 11), evaluando:

Tiempos de ejecución de tareas (Tabla 11): Registro (40 segundos), reserva completa (3 minutos), registro de encomiendas (2 minutos).

Criterios de aceptación por perfil (Tabla 12): Adultos mayores, usuarios ocasionales y conductores.

Tasas de éxito por funcionalidad (Tabla 13): Efectividad del 90-100% en procesos críticos, con detección de fricción en registro de encomiendas (80% inicial).

Tabla 11. Tiempo promedio y máximo en pruebas de usabilidad

Tarea	Tiempo Promedio	Máximo Aceptable
Registro/Login con Google	40 segundos	2 minutos
Reserva completa de boleto (selección, ruta, asiento, pago)	3 minutos	4 minutos
Registro de encomienda (3 pasos)	2 minutos	3 minutos
Creación de nueva ruta (admin)	3 minutos	5 minutos
Validación de pago transferencia	30 segundos	1 minuto
Actualización estado encomienda (conductor)	20 segundos	30 segundos
Lectura y comprensión notificación	10 segundos	20 segundos
Acceso offline a boleto (historial)	5 segundos	10 segundos
Registro conductor + vinculación unidad	2 minutos	4 minutos

Tabla 12. Criterios de evaluación por tipo de usuario

Perfil	Tiempo	Penalización (Fallo)
Usuario digital experto	60% del tiempo máximo	>Tiempo máximo
Usuario ocasional	80% del tiempo máximo	>Tiempo máximo
Usuario adulto mayor	Tiempo máximo aceptable	>Tiempo máximo + 20% tolerancia
Conductor	Estricto < 30 segundos	Cualquier tiempo que interfiera con conducción

Tabla 13. Resultados y análisis de las pruebas de usabilidad del sistema Trans Doramald

Funcionalidad evaluada	Usuarios participantes (n)	Resultado obtenido (%)	Análisis del resultado
Registro de usuario	10 pasajeros	90	Nueve de los diez participantes completaron el registro sin asistencia. Un usuario presentó dificultad inicial para identificar el botón de confirmación, lo que evidencia una curva de aprendizaje leve.
Inicio de sesión	10 pasajeros	100	Todos los usuarios lograron iniciar sesión correctamente en el primer intento, confirmando un flujo de autenticación claro y eficiente.
Consulta de rutas	10 pasajeros	90	La mayoría identificó rutas y horarios sin inconvenientes; un usuario requirió revisar nuevamente los filtros antes de completar la tarea.
Selección de asientos	10 pasajeros	100	El sistema bloqueó correctamente los asientos ocupados, evitando conflictos y garantizando la integridad del proceso de reserva.
Compra de boletos	10 pasajeros	90	Nueve usuarios completaron la compra sin inconvenientes; un participante revisó varias veces el resumen antes de confirmar, sin generar error funcional.
Historial de boletos	10 pasajeros	100	Todos los participantes accedieron correctamente al historial de viajes, demostrando consistencia en la persistencia de datos.
Registro de encomiendas	10 pasajeros	80	Ocho usuarios completaron el registro sin asistencia; dos requirieron aclaración en campos específicos, lo que sugiere oportunidades de mejora en la rotulación del formulario.
Gestión de encomiendas	4 choferes	75	Tres de los cuatro conductores completaron la tarea dentro del

Inicio de sesión administrativo	3 administradores	100	tiempo establecido; uno presentó adaptación inicial al flujo digital, sin afectar la operación general. Todos los administradores accedieron correctamente al sistema mediante doble autenticación, manteniendo un equilibrio entre seguridad y usabilidad.
Gestión de rutas	3 administradores	100	Las operaciones de creación y edición se realizaron sin errores, evidenciando una curva de aprendizaje baja del panel administrativo.
Validación de pagos	3 administradores	67	Dos administradores completaron la validación sin inconvenientes; uno requirió una segunda revisión del comprobante
Sistema de notificaciones	10 pasajeros	100	Todos los usuarios recibieron y comprendieron las notificaciones de forma oportuna, fortaleciendo la comunicación del sistema.

Los resultados de las pruebas de usabilidad muestran que el sistema Trans Doramald presenta un desempeño satisfactorio en escenarios reales de uso, aun considerando un tamaño de muestra reducido. La mayoría de las funcionalidades evaluadas alcanzaron porcentajes de éxito iguales o superiores al 90% en el perfil de pasajeros, lo que demuestra que los procesos principales registro, inicio de sesión, consulta de rutas, reserva y compra de boletos pueden ser completados de forma efectiva sin asistencia externa. Las tareas que presentaron porcentajes menores, como el registro de encomiendas (80%) y la gestión de encomiendas por parte de los conductores (75%), estuvieron asociadas principalmente a una adaptación inicial al uso de formularios digitales, sin que se registren errores críticos o fallos funcionales del sistema.

4.2.5. Fase de Retroalimentación

Análisis de métricas de rendimiento:

La evaluación de los reportes de Apptim reveló que la carga de imágenes en el módulo de validación de pagos consumía 18MB de memoria y generaba latencias de 3.2 segundos (incumpliendo RNF004). Esta métrica cuantitativa motivó la refactorización del componente ImagePicker, implementando compresión progresiva que redujo el consumo a 4MB y el tiempo a 0.8 segundos. Los commits correspondientes (optimize-image-loading, fix-memory-leak-validation) fueron integrados mediante pair programming y documentados en el historial de GitHub.

Documentación del ciclo de mejora:

Cada iteración de refinamiento quedó registrada en el repositorio Git mediante commits semánticos, estableciendo trazabilidad directa entre los hallazgos de las fases de prueba y las correcciones implementadas, evidenciando el cumplimiento del principio XP de "diseño emergente" y "mejora continua".

```
466 468 }
467 469
470 + // CREAR NOTIFICACIÓN EN FIRESTORE (sin mostrar nada en pantalla)
468 471 espera db.recogida ( 'notificaciones' ).agregar ({
469 472   'userId' : userIdFinal,
470 473   'correo electrónico' : widget.userEmail,
471 474
472 475   '@@ -498,7 +501,7 @@ clase _PagoScreen2State extiende State<PagoScreen2>
473 476   'fechaCompra' : FieldValue.serverTimestamp (),
474 477   });
475 478
476 479
477 480
478 481 - debugPrint ( ' Notificación creada exitosamente ' );
479 482 + debugPrint ( ' Notificación creada silenciosamente en Firestore ' );
480 483 } captura ( e ) {
481 484   debugPrint ( ' ✖ Error al crear notificación: $ e ' );
482 485 }
483 486
484 487
485 488
486 489
487 490
488 491
489 492
490 493
491 494
492 495
493 496
494 497
495 498
496 499
497 500
498 501
499 502
500 503
501 504
502 505
503 506
504 507
505 508
506 509
507 510
508 511
509 512
510 513
511 514
512 515
513 516
514 517
515 518
516 519
517 520
518 521
519 522
520 523
521 524
522 525
523 526
524 527
525 528
526 529
527 530
528 531
529 532
530 533
531 534
532 535
533 536
534 537
535 538
536 539
537 540
538 541
539 542
540 543
541 544
542 545
543 546
544 547
545 548
546 549
547 550
548 551
549 552
550 553
551 554
552 555
553 556
554 557
555 558
556 559
557 560
558 561
559 562
560 563
561 564
562 565
563 566
564 567
565 568
566 569
567 570
568 571
569 572
570 573
571 574
572 575
573 576
574 577
575 578
576 579
577 580
578 581
579 582
580 583
581 584
582 585
583 586
584 587
585 588
586 589
587 590
588 591
589 592
590 593
591 594
592 595
593 596
594 597
595 598
596 599
597 600
600 601
601 602
602 603
603 604
604 605
605 606
606 607
607 608
608 609
609 610
610 611
611 612
612 613
613 614
614 615
615 616
616 617
617 618
618 619
619 620
620 621
621 622
622 623
623 624
624 625
625 626
626 627
627 628
628 629
629 630
630 631
631 632
632 633
633 634
634 635
635 636
636 637
637 638
638 639
639 640
640 641
641 642
642 643
643 644
644 645
645 646
646 647
647 648
648 649
649 650
650 651
651 652
652 653
653 654
654 655
655 656
656 657
657 658
658 659
659 660
660 661
661 662
662 663
663 664
664 665
665 666
666 667
667 668
668 669
669 670
670 671
671 672
672 673
673 674
674 675
675 676
676 677
677 678
678 679
679 680
680 681
681 682
682 683
683 684
684 685
685 686
686 687
687 688
688 689
689 690
690 691
691 692
692 693
693 694
694 695
695 696
696 697
697 698
698 699
699 700
700 701
701 702
702 703
703 704
704 705
705 706
706 707
707 708
708 709
709 710
710 711
711 712
712 713
713 714
714 715
715 716
716 717
717 718
718 719
719 720
720 721
721 722
722 723
723 724
724 725
725 726
726 727
727 728
728 729
729 730
730 731
731 732
732 733
733 734
734 735
735 736
736 737
737 738
738 739
739 740
740 741
741 742
742 743
743 744
744 745
745 746
746 747
747 748
748 749
749 750
750 751
751 752
752 753
753 754
754 755
755 756
756 757
757 758
758 759
759 760
760 761
761 762
762 763
763 764
764 765
765 766
766 767
767 768
768 769
769 770
770 771
771 772
772 773
773 774
774 775
775 776
776 777
777 778
778 779
779 780
780 781
781 782
782 783
783 784
784 785
785 786
786 787
787 788
788 789
789 790
790 791
791 792
792 793
793 794
794 795
795 796
796 797
797 798
798 799
799 800
800 801
801 802
802 803
803 804
804 805
805 806
806 807
807 808
808 809
809 810
810 811
811 812
812 813
813 814
814 815
815 816
816 817
817 818
818 819
819 820
820 821
821 822
822 823
823 824
824 825
825 826
826 827
827 828
828 829
829 830
830 831
831 832
832 833
833 834
834 835
835 836
836 837
837 838
838 839
839 840
840 841
841 842
842 843
843 844
844 845
845 846
846 847
847 848
848 849
849 850
850 851
851 852
852 853
853 854
854 855
855 856
856 857
857 858
858 859
859 860
860 861
861 862
862 863
863 864
864 865
865 866
866 867
867 868
868 869
869 870
870 871
871 872
872 873
873 874
874 875
875 876
876 877
877 878
878 879
879 880
880 881
881 882
882 883
883 884
884 885
885 886
886 887
887 888
888 889
889 890
890 891
891 892
892 893
893 894
894 895
895 896
896 897
897 898
898 899
899 900
900 901
901 902
902 903
903 904
904 905
905 906
906 907
907 908
908 909
909 910
910 911
911 912
912 913
913 914
914 915
915 916
916 917
917 918
918 919
919 920
920 921
921 922
922 923
923 924
924 925
925 926
926 927
927 928
928 929
929 930
930 931
931 932
932 933
933 934
934 935
935 936
936 937
937 938
938 939
939 940
940 941
941 942
942 943
943 944
944 945
945 946
946 947
947 948
948 949
949 950
950 951
951 952
952 953
953 954
954 955
955 956
956 957
957 958
958 959
959 960
960 961
961 962
962 963
963 964
964 965
965 966
966 967
967 968
968 969
969 970
970 971
971 972
972 973
973 974
974 975
975 976
976 977
977 978
978 979
979 980
980 981
981 982
982 983
983 984
984 985
985 986
986 987
987 988
988 989
989 990
990 991
991 992
992 993
993 994
994 995
995 996
996 997
997 998
998 999
999 1000
```

Figura 46. Corrección de notificaciones

```
484 + destino finalCiudad = destinatario[ 'ciudad' ] ?? '' ;
482 485
483 486
484 487
485 488
486 489
487 490
488 491
489 492
490 493
491 494
492 495
493 496
494 497
495 498
496 499
497 500
498 501
499 502
500 503
501 504
502 505
503 506
504 507
505 508
506 509
507 510
508 511
509 512
510 513
511 514
512 515
513 516
514 517
515 518
516 519
517 520
518 521
519 522
520 523
521 524
522 525
523 526
524 527
525 528
526 529
527 530
528 531
529 532
530 533
531 534
532 535
533 536
534 537
535 538
536 539
537 540
538 541
539 542
540 543
541 544
542 545
543 546
544 547
545 548
546 549
547 550
548 551
549 552
550 553
551 554
552 555
553 556
554 557
555 558
556 559
557 560
558 561
559 562
560 563
561 564
562 565
563 566
564 567
565 568
566 569
567 570
568 571
569 572
570 573
571 574
572 575
573 576
574 577
575 578
576 579
577 580
578 581
579 582
580 583
581 584
582 585
583 586
584 587
585 588
586 589
587 590
588 591
589 592
590 593
591 594
592 595
593 596
594 597
595 598
596 599
597 600
600 601
601 602
602 603
603 604
604 605
605 606
606 607
607 608
608 609
609 610
610 611
611 612
612 613
613 614
614 615
615 616
616 617
617 618
618 619
619 620
620 621
621 622
622 623
623 624
624 625
625 626
626 627
627 628
628 629
629 630
630 631
631 632
632 633
633 634
634 635
635 636
636 637
637 638
638 639
639 640
640 641
641 642
642 643
643 644
644 645
645 646
646 647
647 648
648 649
649 650
650 651
651 652
652 653
653 654
654 655
655 656
656 657
657 658
658 659
659 660
660 661
661 662
662 663
663 664
664 665
665 666
666 667
667 668
668 669
669 670
670 671
671 672
672 673
673 674
674 675
675 676
676 677
677 678
678 679
679 680
680 681
681 682
682 683
683 684
684 685
685 686
686 687
687 688
688 689
689 690
690 691
691 692
692 693
693 694
694 695
695 696
696 697
697 698
698 699
699 700
700 701
701 702
702 703
703 704
704 705
705 706
706 707
707 708
708 709
709 710
710 711
711 712
712 713
713 714
714 715
715 716
716 717
717 718
718 719
719 720
720 721
721 722
722 723
723 724
724 725
725 726
726 727
727 728
728 729
729 730
730 731
731 732
732 733
733 734
734 735
735 736
736 737
737 738
738 739
739 740
740 741
741 742
742 743
743 744
744 745
745 746
746 747
747 748
748 749
749 750
750 751
751 752
752 753
753 754
754 755
755 756
756 757
757 758
758 759
759 760
760 761
761 762
762 763
763 764
764 765
765 766
766 767
767 768
768 769
769 770
770 771
771 772
772 773
773 774
774 775
775 776
776 777
777 778
778 779
779 780
780 781
781 782
782 783
783 784
784 785
785 786
786 787
787 788
788 789
789 790
790 791
791 792
792 793
793 794
794 795
795 796
796 797
797 798
798 799
799 800
800 801
801 802
802 803
803 804
804 805
805 806
806 807
807 808
808 809
809 810
810 811
811 812
812 813
813 814
814 815
815 816
816 817
817 818
818 819
819 820
820 821
821 822
822 823
823 824
824 825
825 826
826 827
827 828
828 829
829 830
830 831
831 832
832 833
833 834
834 835
835 836
836 837
837 838
838 839
839 840
840 841
841 842
842 843
843 844
844 845
845 846
846 847
847 848
848 849
849 850
850 851
851 852
852 853
853 854
854 855
855 856
856 857
857 858
858 859
859 860
860 861
861 862
862 863
863 864
864 865
865 866
866 867
867 868
868 869
869 870
870 871
871 872
872 873
873 874
874 875
875 876
876 877
877 878
878 879
879 880
880 881
881 882
882 883
883 884
884 885
885 886
886 887
887 888
888 889
889 890
890 891
891 892
892 893
893 894
894 895
895 896
896 897
897 898
898 899
899 900
900 901
901 902
902 903
903 904
904 905
905 906
906 907
907 908
908 909
909 910
910 911
911 912
912 913
913 914
914 915
915 916
916 917
917 918
918 919
919 920
920 921
921 922
922 923
923 924
924 925
925 926
926 927
927 928
928 929
929 930
930 931
931 932
932 933
933 934
934 935
935 936
936 937
937 938
938 939
939 940
940 941
941 942
942 943
943 944
944 945
945 946
946 947
947 948
948 949
949 950
950 951
951 952
952 953
953 954
954 955
955 956
956 957
957 958
958 959
959 960
960 961
961 962
962 963
963 964
964 965
965 966
966 967
967 968
968 969
969 970
970 971
971 972
972 973
973 974
974 975
975 976
976 977
977 978
978 979
979 980
980 981
981 982
982 983
983 984
984 985
985 986
986 987
987 988
988 989
989 990
990 991
991 992
992 993
993 994
994 995
995 996
996 997
997 998
998 999
999 1000
```

Figura 47. Corrección y simplificación de UX

```
464 + // ----- PASO 4: AGREGAR NUEVA FUNCIÓN PARA PUSH NOTIFICATIONS -----
465 + // Agrega esta función nueva después de .crearNotificación:
466 + Futuro <void> _enviarNotificaciónPush ({
467 +   Cadena de usuario requerida ,
468 +   Título de cadena requerida ,
469 +   Cadena de mensaje requerida ,
470 +   Cadena requerida uidRemitente,
471 +   Se requiere cadena codigoEncomienda,
472 +   Cadena requerida estado,
473 + }) asíncrono {
474 +   intentar {
475 +     // SOLO el dominio base
476 +     const String baseUrl = 'https://notificaciones-3hoa.onrender.com';
477 +
478 +     // Punto final correcto (tal como esté en Flask)
479 +     final Uri url = Uri .parse (
480 +       '$ baseUrl /api/notificaciones/enviar-al-usuario' ,
481 +     );
482 +
483 +     respuesta final = esperar http.post (
484 +       URL,
485 +       encabezados : { 'Tipo de contenido' : 'application/json' },
486 +       cuerpo : json.codificar ( {
487 +         'userId' : uidRemitente,
488 +         'titulo' : titulo,
489 +         'cuerpo' : mensaje,
490 +         'datos' : {
491 +           'tipo' : 'encomienda' ,
492 +           'codigo' : codigoEncomienda,
493 +           'estado' : estado,
494 +           'accion' : 'cambio_estado' ,
```

Figura 48. Corrección del módulo de notificaciones push

Esta fase garantizó que el sistema no solo cumpliera con los requerimientos técnicos documentados, sino que se adaptara efectivamente a las capacidades reales de los dispositivos y las condiciones operativas de la compañía, cerrando el ciclo de mejora continua propio de XP.

4.2.6. Fase de Lanzamiento

Aplicación **Trans Doramald**

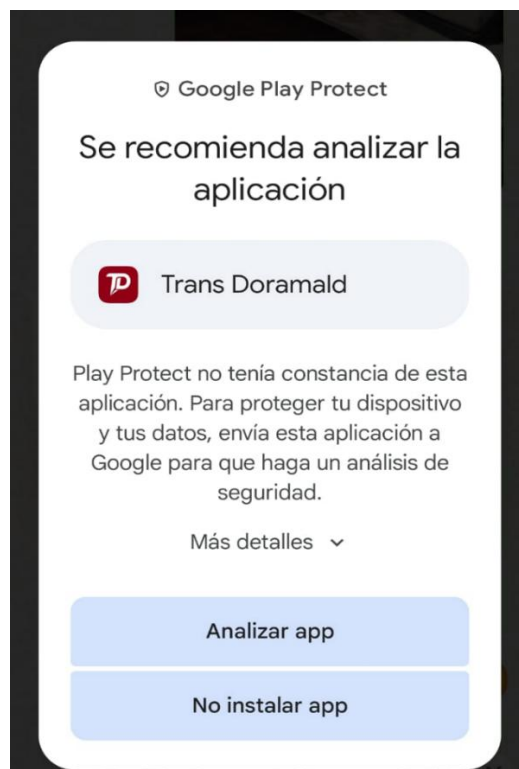


Figura 49. Alerta de seguridad

Interfaz de seguridad del sistema operativo Android (Google Play Protect) que verifica la integridad del archivo APK de la aplicación "Trans Doramald" antes de su instalación. La ventana modal muestra el escaneo de seguridad confirmando que la aplicación no contiene malware, garantizando al usuario final la confianza necesaria para instalar software de fuentes externas a las tiendas oficiales, aspecto crítico en zonas rurales donde la distribución puede ser mediante instaladores offline (APK).

Información Trans Doramald 2026



Figura 50. Panel de información

Pantalla de bienvenida principal de la aplicación móvil que funciona como punto de entrada al sistema. Presenta una interfaz gráfica con identidad corporativa de la compañía (colores institucionales), imagen distintiva de una unidad de transporte y el eslogan "Seguridad Garantizada". Incluye navegación inferior tipo bottom navigation bar con tres accesos principales: Información (actual), Menú de servicios y Notificaciones. Esta pantalla describe la experiencia de usuario en el sector transporte rural, estableciendo la propuesta de valor de la compañía.

Información que detalla su misión



Figura 51. Panel de misión y visión

Módulo informativo organizacional que presenta los postulados institucionales de la compañía "Trans Doramald". La interfaz divide verticalmente el contenido en dos secciones diferenciadas: Misión (propósito actual y compromiso con la comunidad de Chical y Maldonado) y Visión (proyección a futuro del servicio).

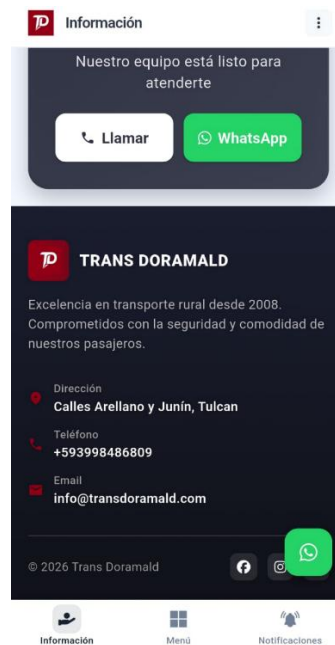


Figura 52. Panel de Contactos

Interfaz de comunicación directa que centraliza los canales de atención al cliente. Presenta los datos de contacto institucional (teléfono móvil de la compañía) y

botones de acción rápida (call-to-action) para iniciar comunicación inmediata vía WhatsApp Business. Incluye además el correo electrónico institucional para consultas formales.

Sistema de registro y login (Usuario-Administrador)

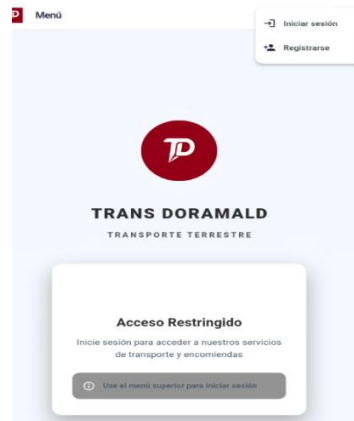


Figura 53. Menú de acceso

Pantalla de navegación inicial que separa el acceso según el perfil de usuario. Presenta dos opciones distintivas visualmente: Acceso para usuarios regulares (pasajeros) mediante autenticación Google OAuth 2.0, y portal restringido para administradores con credenciales específicas. El diseño considera la brecha digital de usuarios al priorizar el login social sobre el registro tradicional con contraseña.

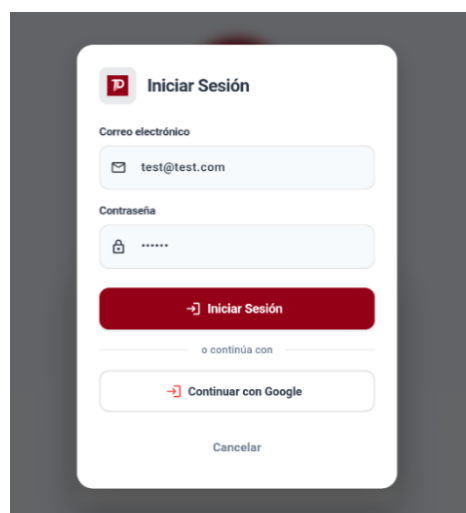


Figura 54. Panel de inicio de sesión

Interfaz de autenticación para usuarios finales que implementa el estándar OAuth 2.0 mediante Google Sign-In. La pantalla muestra campos de email/contraseña para usuarios con registro previo, pero destaca visualmente el botón de acceso rápido con

Google, minimizando la fricción de entrada al sistema y reduciendo el abandono por olvido de credenciales.

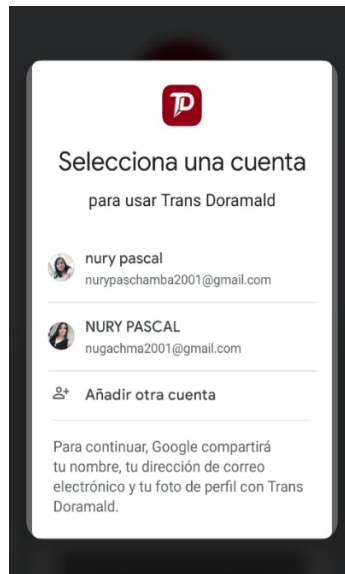


Figura 55. Panel de selección de cuenta (Google)

Diálogo nativo de Android para selección de cuentas Google disponibles en el dispositivo. Muestra las cuentas previamente sincronizadas en el smartphone, permitiendo al usuario seleccionar con un solo toque la identidad digital que utilizará en la aplicación, implementando así el requerimiento funcional RF002 (Autenticación y seguridad) mediante tokens de autenticación federada.

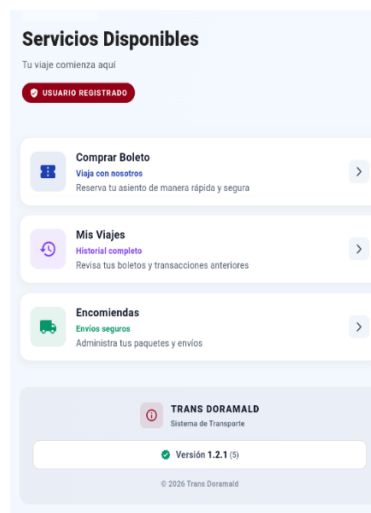


Figura 56. Panel de servicios

Dashboard central del usuario autenticado que actúa como hub de navegación hacia los tres servicios principales: "Comprar Boleto" con iconografía intuitiva de

boletería, "Mis Viajes" para gestión de historial y reservas activas, y "Encomiendas" para el servicio de paquetería.

Pantalla de Compra de Boletos

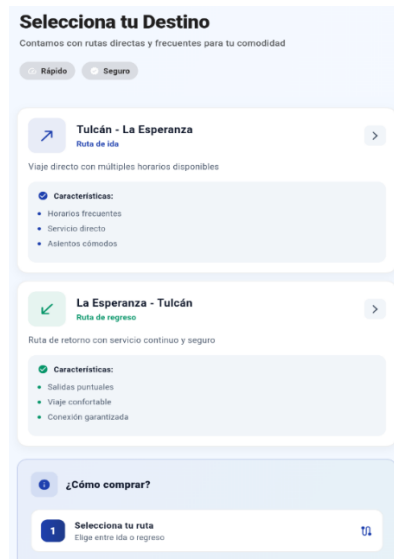


Figura 57. Panel de compra de boletos

Interfaz del proceso de reserva (RF003) que presenta el catálogo de rutas disponibles. Muestra tarjetas informativas por cada trayecto (Tulcán-La Esperanza y viceversa) con indicadores visuales de disponibilidad horaria. Cada ruta incluye metadata sobre frecuencia de salidas y estado operativo actual, permitiendo al usuario seleccionar su itinerario preferido mediante interacción táctil directa.

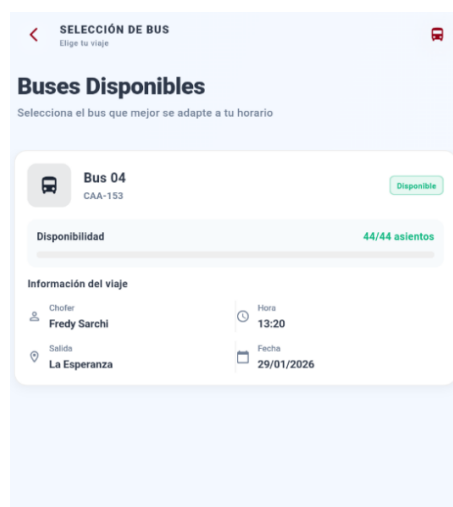


Figura 58. Panel de buses disponibles

Lista las unidades activas para la ruta seleccionada previamente, mostrando información crítica: número de unidad, horario de salida programada, cantidad de asientos disponibles en tiempo real (integración con sistema de bloqueo de asientos), y estado del bus (activo/cupo lleno). Implementa el requerimiento no funcional RNF006 (Integridad) al mostrar datos sincronizados con la base de datos Firestore.

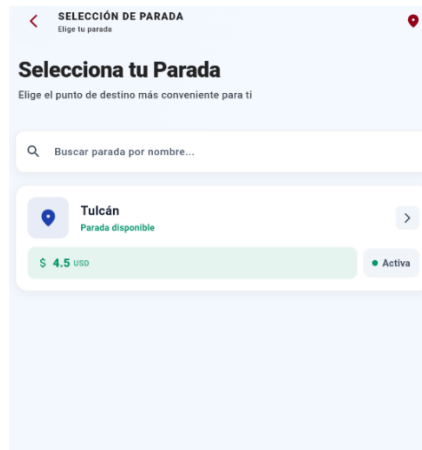


Figura 59. Panel de selección de parada

Módulo de paradas intermedias que permite al usuario seleccionar puntos de embarque o desembarque específicos dentro de la trayectoria rural (parroquias Maldonado, El Chical, etc.). Presenta un listado scrollable con nombres de paradas reconocidas localmente y tiempos estimados de llegada, facilitando la planificación del viaje para usuarios ubicados en puntos intermedios no terminales.

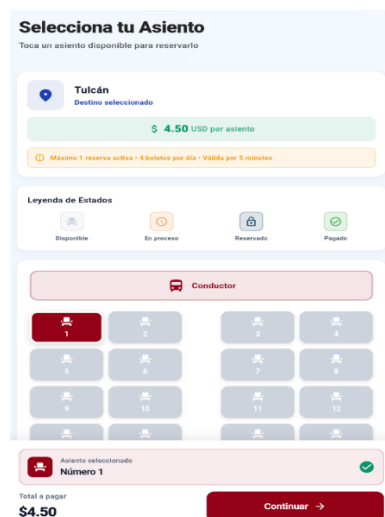


Figura 60. Panel de selección de asiento

Representación gráfica de la distribución física de asientos del bus (diagrama de planta) implementando un sistema de selección visual tipo "mapa de calor": asientos disponibles en verde, ocupados en rojo, amarillo reservado. La interfaz utiliza un GridView adaptativo que simula la disposición real de la unidad (asientos por fila, pasillo central), permitiendo al usuario elegir ubicación específica (ventana/pasillo) según preferencia.

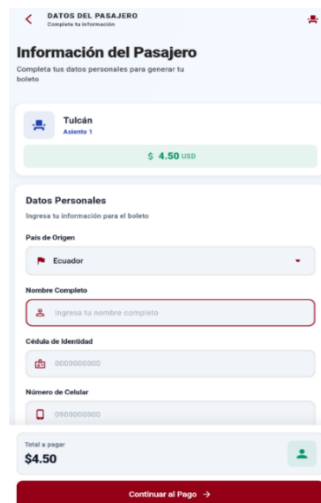


Figura 61. Panel de información del pasajero

Formulario de captura de datos del viajero para la reservación. Campos obligatorios incluyen: número de cédula de identidad (validación formatos Ecuador), nombres completos y número de contacto telefónico. La interfaz implementa validación en tiempo real (input validation) para asegurar la integridad de los datos antes del proceso de pago, cumpliendo con normativas de transporte para control de pasajeros.

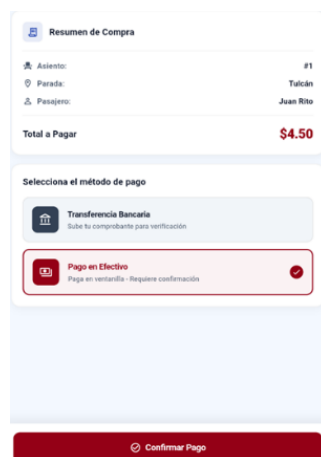


Figura 62. Panel de método de pago

Interfaz de selección de forma de pago que ofrece dualidad entre métodos tradicionales (efectivo en punto físico) y digitales (transferencia bancaria). Para pagos electrónicos, incluye visualización de datos bancarios de la compañía y botón para adjuntar comprobante de transferencia (upload de imagen).

Pantalla de Historial de Boletos Comprados

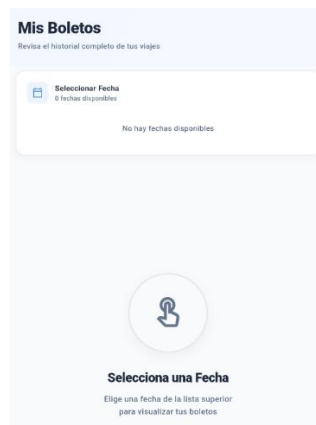


Figura 63. Panel de historial de boletos

Módulo de consulta de transacciones (RF005) que presenta listado cronológico de viajes realizados y reservas activas. Cada ítem del historial funciona como entrada desplegable (accordion) que muestra detalles completos: fecha, hora, asiento asignado, estado del boleto (pagado/pendiente/cancelado) y opción de descarga de comprobante digital en formato PDF. Incluye filtros por rango de fechas para facilitar la auditoría personal.

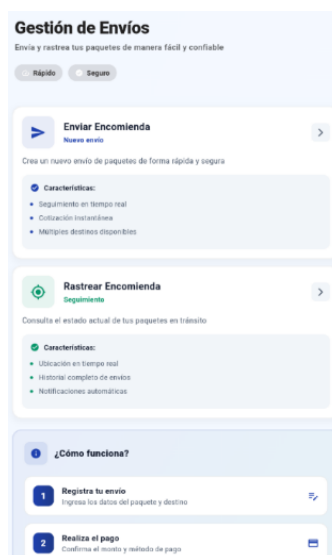


Figura 64. Panel de encomiendas

Portal de acceso al servicio de paquetería (RF004) con descripción del alcance del servicio (envíos seguros entre Tulcán y comunidades rurales). Presenta estadísticas de envío (paquetes entregados, en tránsito) y botón prominente para iniciar nuevo registro de encomienda. El diseño incluye iconografía distintiva de paquetería y colores diferenciados del módulo de pasajeros para evitar confusiones de contexto.

The screenshot shows a mobile application interface for creating a new shipment. The title is 'NUEVA ENCOMIENDA' with a sub-header 'Código: ENC-0004'. Below this is the section 'Datos del Remitente' (Sender Information) with the subtitle 'Información de la persona que envía el paquete'. There are two tabs: 'Seguro' (selected) and 'Privado'. A progress indicator shows three steps: 'Remitente' (1, active), 'Destinatario' (2), and 'Envío' (3). A message says 'Completa todos los campos marcados con *'. The form fields are: 'Nombre completo *' (Example: Juan Pérez García), 'Cédula *' (Example: 1234567890), 'Teléfono *' (Example: 0999999999), 'Correo electrónico *' (Example: test@test.com), and 'Lugar de salida *' (Dropdown menu: 'Seleccione el lugar de salida'). A red button at the bottom says 'Continuar al Destinatario' with a right arrow.

Figura 65. Panel de nueva encomienda (remitente)

Primer paso del registro de paquetería. Formulario estructurado para captura de datos del emisor: nombres completos, número de teléfono de contacto y dirección de origen completa. Incluye validación de campos obligatorios (*) y teclado optimizado para entrada numérica en campos telefónicos, reduciendo errores de tipeo en dispositivos móviles.

The screenshot shows the second step of the shipment registration process. The title is 'NUEVA ENCOMIENDA' with a sub-header 'Código: ENC-0004'. Below this is the section 'Datos del Destinatario' (Recipient Information) with the subtitle 'Información completa de quien recibirá el paquete'. There are two tabs: 'Seguro' and 'Privado' (selected). A progress indicator shows three steps: 'Remitente' (1, completed), 'Destinatario' (2, active), and 'Envío' (3). A message says 'Completa todos los campos requeridos (*)'. The form fields are: 'Nombre completo *' (Example: María González López), 'Cédula *' (Example: 1234567890), 'Teléfono *' (Example: 0999999999), and 'Correo electrónico' (Example: correo@ejemplo.com). A red button at the bottom says 'Continuar al Destinatario' with a right arrow.

Figura 66. Panel de nueva encomienda (destinatario)

Segundo paso del flujo de paquetería. Captura de información del receptor con campos adicionales de ubicación específica (dirección de destino rural, referencias geográficas locales reconocibles). Incluye campo de número de cédula para validación de identidad en la entrega, asegurando la cadena de custodia del envío.

Figura 67. Panel de nueva encomienda (detalles del paquete)

Tercera etapa del registro de encomiendas. Formulario técnico para especificación del contenido: descripción detallada del objeto, clasificación por tipo (frágil, perecedero, documentos), categoría de peso/volumen y observaciones especiales. Incluye selector de valor declarado para fines de seguro de transporte y estimación automática de tarifa según peso y distancia.

Figura 68. Ventana de solicitud registrada

Pantalla de confirmación de transacción (feedback positivo) que muestra la encomienda registrada con código de seguimiento único (tracking number).

Pantalla de Mis Notificaciones

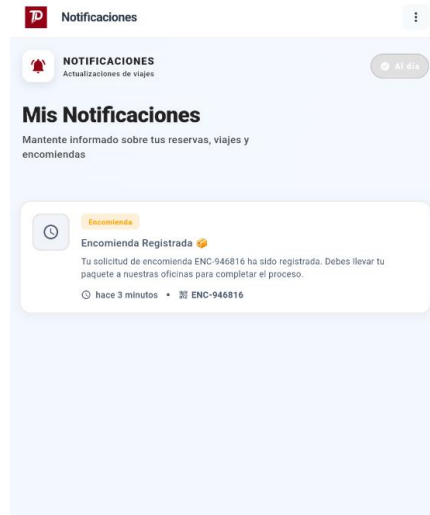


Figura 69. Panel de notificaciones

Centro de mensajes integrado (RF006, RF008) que centraliza alertas del sistema: confirmaciones de reserva, aprobaciones de pago, cambios de horario de último momento, y avisos masivos de la compañía. La interfaz utiliza badges (contadores) en el icono de campana del menú principal y diferencia visual entre notificaciones leídas y pendientes mediante opacidad diferenciada.

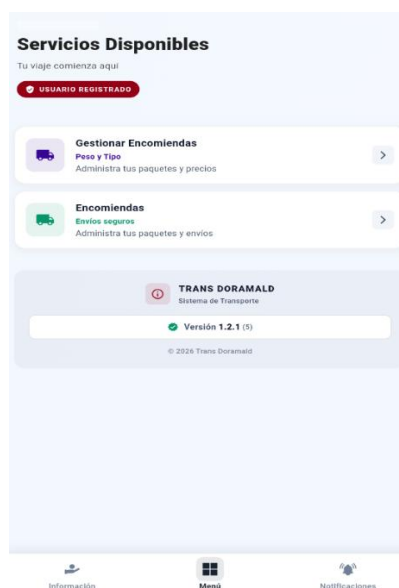


Figura 70. Servicios disponibles (chofer)

Dashboard específico para conductores con privilegios restringidos. Muestra resumen de servicios activos asignados al chofer autenticado, incluyendo acceso al módulo de encomiendas asignadas a su unidad. Diseño simplificado para uso durante operación.

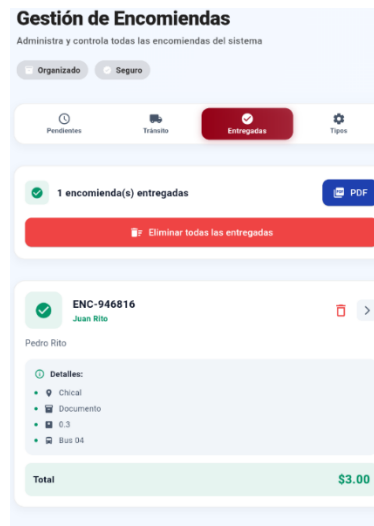


Figura 71. Gestión de encomiendas (chofer)

Interfaz operativa para conductores que permite la actualización de estado de paquetes en ruta. Lista las encomiendas asignadas a la unidad del conductor con opciones de cambio de estado: "Recogido", "En tránsito" y "Entregado". Incluye función de escaneo o ingreso manual de códigos de tracking para confirmación de entrega en destino, sincronizando en tiempo real con el panel del administrador.

Desarrollo de la parte administrativa.

Acceso del administrador

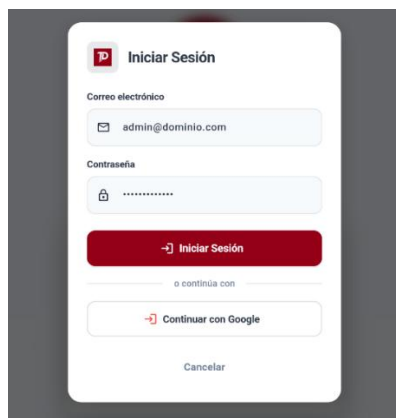


Figura 72. Panel de inicio de sesión (administrador)

Portal de acceso restringido para personal autorizado (gerentes y secretaría). Diferente a la interfaz de usuarios regulares, requiere autenticación dual: correo electrónico institucional y contraseña segura. Incluye mensaje de advertencia sobre acceso restringido, cumpliendo con RNF003 (Seguridad) y RNF012 (Privacidad).

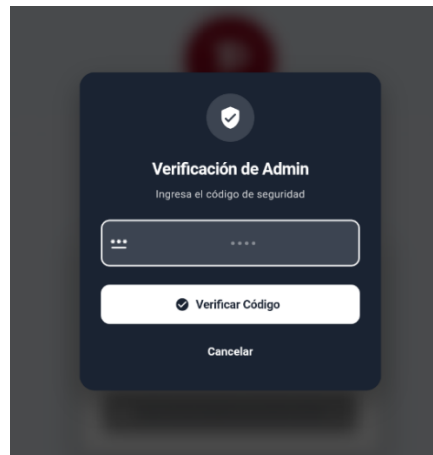


Figura 73. Código de verificación de administrador

Sistema de autenticación multifactor (MFA) para capa adicional de seguridad en funciones críticas. Tras el login inicial, el sistema pide un PIN numérico. La interfaz presenta teclado numérico virtual para ingreso del código, implementando RF002 con estándares de seguridad empresarial.

Ventanas del administrador

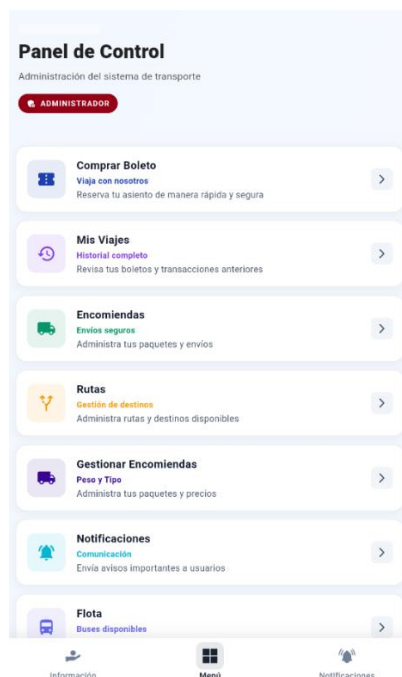


Figura 74. Panel de control

Panel central de gestión operativa que integra todas las funciones administrativas. Presenta widgets resumidos: estadísticas de ventas diarias, ocupación de buses en tiempo real, encomiendas pendientes de entrega, y gráficos de tendencia de pasajeros. Incluye acceso a submódulos: Rutas, Buses, Pagos, Reportes, Notificaciones y Gestión de Datos.

Administración de rutas.

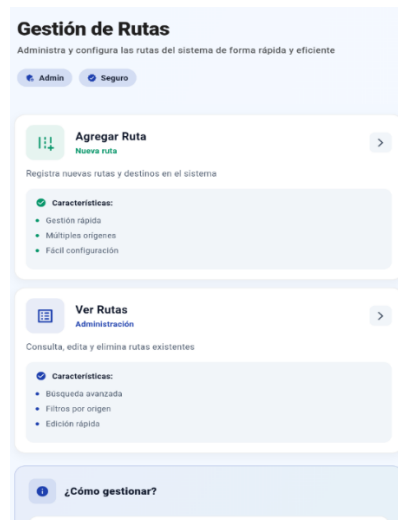


Figura 75. Panel de gestión de rutas

Interfaz de administración de trayectorias (RF007) que lista las rutas activas del sistema (Tulcán-Maldonado, Tulcán-Chical). Permite operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre trayectos, mostrando metadata de cada ruta: origen, destino, paradas intermedias y tarifa base. Incluye indicadores de estado (activa/inactiva) para control de visibilidad en la app de usuarios.

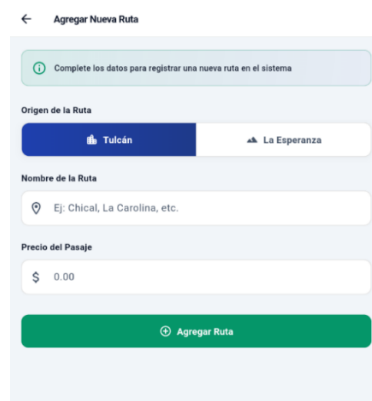


Figura 76. Panel de nueva ruta

Formulario de alta de trayectos con campos estructurados: nombre identificativo de la ruta, selección de origen/destino desde catálogo geográfico, y campo dinámico para agregar paradas intermedias. Incluye selector de horarios de operación y asignación preliminar de tarifas, validando que no exista duplicidad de rutas en el sistema. Rutas actuales registradas

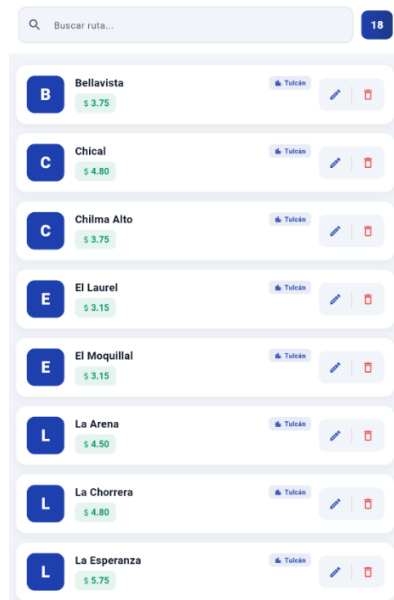


Figura 77. Panel de rutas registradas

Vista de administración de catálogo de rutas existentes. Presenta tabla/listado con 18 rutas activas identificadas, mostrando código de ruta, trayecto completo (origen-destino), y acciones de edición/eliminación. Incluye función de búsqueda y filtros por estado operativo, facilitando la gestión de la oferta de servicios según demanda estacional.

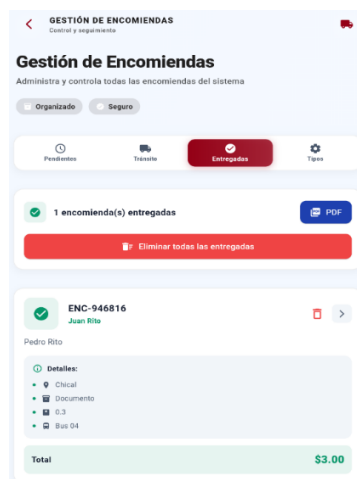


Figura 78. Panel de gestión de encomiendas (administrador)

Panel de supervisión del servicio de paquetería que presenta tabla de registros con filtros avanzados (por estado, fecha, ruta, chofer asignado). Permite reasignación manual de paquetes entre unidades en caso de contingencias operativas y visualización de detalles completos del envío (remitente, destinatario, contenido, valor declarado).

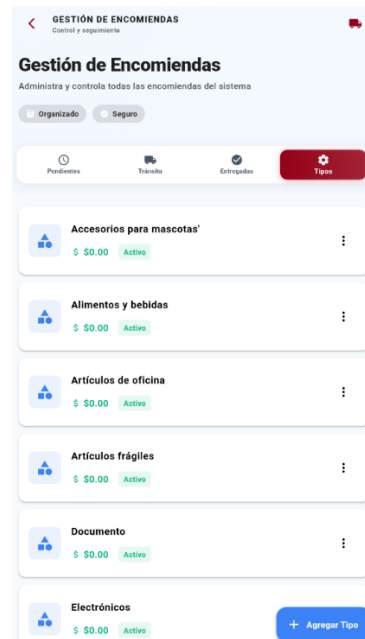


Figura 79. Panel de tipos de encomiendas

Interfaz de configuración de categorías de paquetería para clasificación logística. Permite definir tipos de envío (documentos, carga frágil, perecederos, gran volumen) con sus respectivas tarifas base y restricciones de transporte. Incluye campos para especificaciones de embalaje requerido y advertencias de manipulación especial.

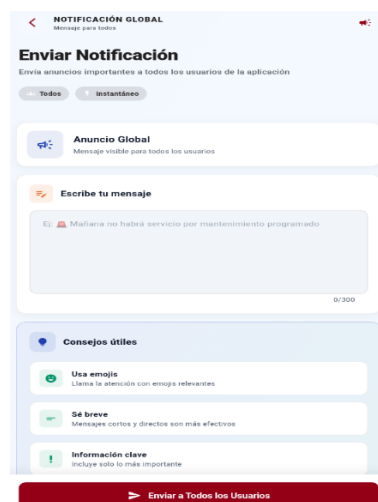


Figura 80. Panel de enviar notificación

Módulo de comunicación masiva (RF008) para emisión de avisos oficiales. Editor de texto enriquecido para redacción de comunicados, implementando el sistema de notificaciones push para información crítica sobre cambios de horarios o emergencias viales.



Figura 81. Panel de buses registrados

Inventario digital de la flota vehicular (RF009, RF012) que muestra tarjetas de unidad con información completa: número de unidad, placa, capacidad de asientos, nombre del conductor asignado, horario de salida programada en tiempo real.

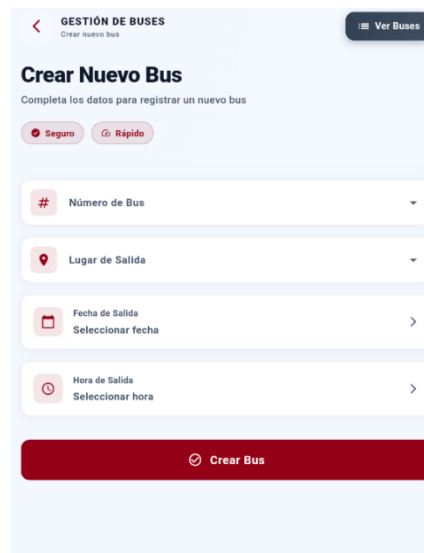


Figura 82. Panel de crear nueva salida de bus

Formulario de programación de salidas específicas que vincula un bus físico con una ruta y horario determinado. Campos: selección de unidad desde inventario, asignación de conductor, fecha y hora de salida, punto de origen. Valida disponibilidad de la unidad para evitar sobreasignación en horarios coincidentes.

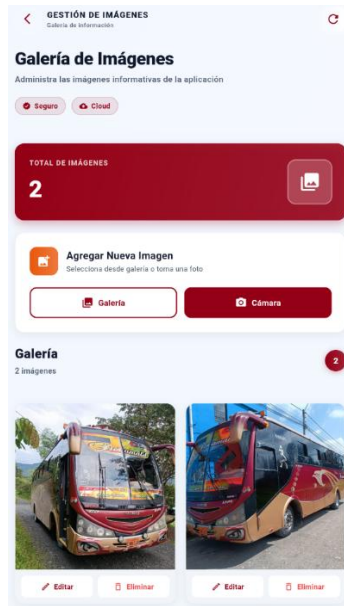


Figura 83. Panel de galería de imágenes

Gestor de contenido multimedia (RF010) para mantenimiento de imagen institucional. Permite carga, previsualización y eliminación de fotografías de la flota, terminales y equipo humano que se muestran en la sección "Información" de la app usuario. Incluye compresión automática de imágenes para optimización de rendimiento (RNF005 - Accesibilidad en zonas de baja conectividad).

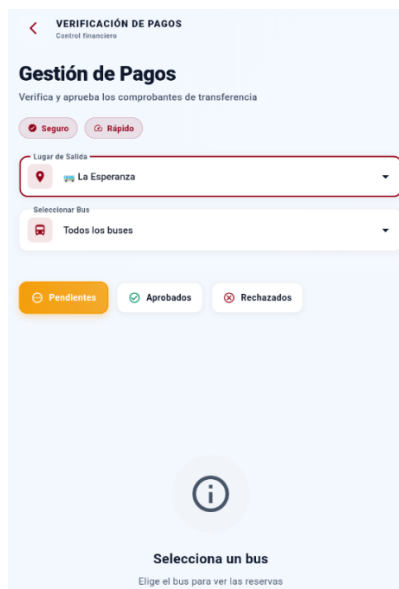


Figura 84. Panel de gestión de pagos

Consola de validación financiera (RF011) que presenta listado de transacciones pendientes de verificación. Muestra miniatura del comprobante de transferencia bancaria cargado por el usuario, monto declarado, datos del pagador.

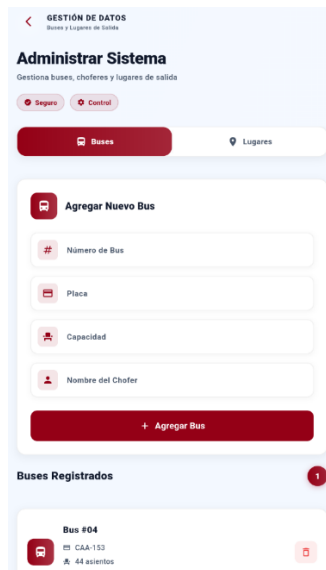


Figura 85. Panel de gestión de datos

Módulo de administración maestra de información operativa (RF012) que integra: Catálogo de conductores con datos personales, Registro de unidades con documentación técnica (matrícula), y Puntos de salida/terminales georreferenciados. Funciona como repositorio central de datos maestros para toda la operación del sistema.

4.3. DISCUSIÓN

La problemática descrita en los antecedentes sobre la falta de un sistema de registro de afiliados, la emisión manual de boletos y la desorganización operativa respalda directamente el enfoque de esta investigación, ya que evidencia la necesidad de una solución tecnológica. El planteamiento es aplicable debido a que la propuesta aborda estos mismos problemas, integrando la gestión del talento humano, lo que amplía el alcance y permite un control más eficiente del servicio. Por tanto, se demuestra una relación directa con el estudio de esta investigación, aportando antecedentes reales del sector del transporte público que refuerzan la pertinencia de implementar sistemas tecnológicos como los que se proponen en este trabajo.

El estudio de (Basantés, 2023) es relevante para esta investigación ya que evidencia los problemas operativos derivados de la ausencia de un sistema automatizado para el registro de afiliados y control de boletos donde la emisión manual genera duplicidad de pasajes y la falta de sincronización entre sucursales. Esto se relaciona directamente con la problemática identificada en la compañía Trans Doramald que enfrentaba los mismos desafíos de desorganización operativa.

El trabajo citado de (Carrasco Mena, 2023) es relevante para esta investigación porque demuestra cómo la automatización y sistematización del proceso de venta de boletos mediante una aplicación móvil mejora la eficiencia del servicio en terminales de transporte. Su estudio identificó que la falta de un sistema tecnológico obligaba a los usuarios a realizar filas, retrasando el proceso. Esto se relaciona directamente con el tema de investigación, ya que también se propone una herramienta tecnológica que no solo optimiza la venta de boletos, sino que incluye la gestión del talento humano, ofreciendo un enfoque más integral. Por tanto, este antecedente respalda la investigación al evidenciar la utilidad del uso de tecnologías móviles en el sector del transporte público.

Asimismo, el estudio (Route, 2023) respalda la presente investigación al demostrar que los avances tecnológicos impactan positivamente en las cooperativas, mejorando la gestión del personal y la seguridad vial. El análisis sobre el uso de sistemas inteligentes refuerza la importancia de incorporar tecnología en este sector. Esto se relaciona directamente con la propuesta de implementar una herramienta tecnológica que optimice tanto la gestión del talento humano como la venta de boletos, demostrando que la innovación mejora los procesos internos y la calidad del servicio.

Por otro lado, el aporte de (Díaz, 2022) es pertinente, ya que explica la estructura funcional de una aplicación web y móvil basada en una arquitectura en tres capas: interfaz de usuario, servidor y base de datos. Este enfoque técnico es aplicable a la propuesta de herramienta tecnológica para la gestión de talento humano y venta de boletos, ya que una solución eficiente requiere una arquitectura robusta que garantice una experiencia del usuario óptima y un manejo seguro de datos. Por tanto, este antecedente fortalece el sustento técnico de la presente investigación.

Finalmente, el estudio de (Díaz Salinas, 2024) resulta altamente relevante, ya que evidencia tanto los avances y limitaciones en la digitalización del transporte público. Si bien se ha progresado en el ámbito interprovincial, en el transporte urbano y rural aún persisten brechas por falta de infraestructura y confianza en la tecnología. Esta realidad justifica la necesidad de la propuesta, ya que una herramienta tecnológica orientada a la gestión del talento humano y la venta de boletos contribuye a cerrar dicha brecha digital y optimizar los procesos internos de las cooperativas.

Tabla 14. Comparativa de antecedentes

Antecedente	Tecnología y Enfoque del Antecedente	Investigación
Basantes (2023)	Enfoque: Registro de afiliados y control manual de boletos. Tecnología: No especifica una arquitectura avanzada, se centra en el problema de la duplicidad	Tecnología: Flutter y Firebase. Aporte: Implementa un bloqueo transaccional en tiempo real que impide físicamente que dos personas reserven el mismo asiento
Carrasco Mena (2023)	Enfoque: Venta en línea para terminales terrestres urbanos (Guayaquil). Tecnología: Aplicación nativa Android	Tecnología: Multiplataforma (Dart/Flutter). Aporte: Diseñada para baja conectividad en zonas rurales (Maldonado/Chical) mediante compresión de recursos y notificaciones locales
Díaz (2022)	Enfoque: Arquitectura tradicional cliente-servidor en tres capas (Navegador, Servidor, DB). Tecnología: Protocolo HTTP estándar.	Tecnología: Serverless (Firebase). Aporte: Elimina la necesidad de un servidor físico costoso. Usa Firestore para bases de datos NoSQL y Firebase Storage para comprobantes de pago.
Route (2023)	Enfoque: Uso de Internet en unidades y sistemas de seguridad vial (radares/sensores). Tecnología: Hardware preventivo de colisiones	Tecnología: Gestión Digital del Talento Humano. Aporte: Integra el control de datos de choferes, roles administrativos y desempeño, vinculando la tecnología con el bienestar del personal.
Díaz Salinas (2024)	Enfoque: Digitalización del transporte público en Ecuador, con avance en el ámbito interprovincial. Tecnología: Plataformas en línea para compra de boletos. Limitación por falta de infraestructura y desconfianza hacia pagos digitales en zonas rurales.	Tecnología: Aplicación móvil con Flutter y Firebase. Aporte: Diseñada específicamente para zonas rurales con conectividad intermitente. Pagos mixtos transferencia y efectivo, superar la desconfianza hacia pagos digitales.

V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Se logró optimizar la gestión del servicio de boletaje mediante la identificación exhaustiva de las limitaciones del sistema manual (duplicidad de reservas, falta de control en tiempo real y desorganización operativa), implementándose un organigrama funcional digital que centraliza las responsabilidades del personal administrativo y operativo, reduciendo los errores en la emisión de boletos y estableciendo protocolos claros de validación de pagos. Paralelamente, se diseñó el organigrama funcional digital que delimita los roles del administrador, secretaria, conductores y pasajeros, respondiendo también a las limitaciones de gestión del talento humano identificadas en el diagnóstico inicial

La comparación sistemática de herramientas tecnológicas determinó que la arquitectura basada en el framework Flutter y la plataforma Firebase constituye la solución óptima para el contexto rural de las parroquias Maldonado y El Chical, dado que permite el funcionamiento offline, sincronización en tiempo real y compatibilidad multiplataforma, superando las limitaciones de conectividad intermitente características de la zona.

La solución tecnológica desarrollada (Aplicación Móvil Trans Doramald) cumplió satisfactoriamente con el objetivo de optimizar la compra de boletos, facilitando el acceso digital mediante reservas anticipadas, bloqueo transaccional de asientos en tiempo real y múltiples métodos de pago (transferencia/efectivo), incrementando significativamente la eficiencia operativa de la compañía. Esta propuesta responde directamente a la demanda identificada en la encuesta inicial, donde los usuarios manifestaron interés en contar con opciones de compra anticipada de boletos.

En cumplimiento del objetivo específico relacionado con la gestión del talento humano, se implementó un módulo de administración centralizada del personal

operativo de Trans Doramald, que permite registrar, actualizar y asignar conductores a unidades específicas mediante una estructura de datos en Firebase, Firestore con vinculación entre las colecciones /choferes y /buses. El sistema diferencia cuatro perfiles de acceso: administrador, secretaria, conductor y pasajero, cada uno con una interfaz y permisos específicos, lo que digitaliza el control del personal y elimina la dependencia de registros manuales en papel que caracterizaba a la compañía antes de la implementación.

5.2. RECOMENDACIONES

Implementar de manera integral el organigrama funcional digital diseñado, capacitando al personal administrativo y conductores sobre sus responsabilidades específicas dentro del nuevo sistema, estableciendo un manual de procedimientos claros para la gestión de boletos y definiendo roles de validación de pagos que eviten duplicidades y garanticen la trazabilidad de cada transacción.

Establecer convenios de colaboración con pequeños comercios locales (tiendas, farmacias o emprendimientos) ubicados estratégicamente en las parroquias Maldonado y El Chical, para que funcionen como agentes de venta autorizados utilizando la aplicación en modo "auxiliar", permitiendo a habitantes sin acceso a smartphones o internet realizar sus reservas y pagos en estos puntos cercanos a sus domicilios, ampliando así la cobertura del sistema digital sin requerir inversión en infraestructura adicional por parte de la compañía.

Diseñar e implementar un programa de capacitación continua para usuarios finales y personal operativo, utilizando metodologías de aprendizaje presencial y demostraciones prácticas en puntos de encuentro comunitarios, que fortalezcan las competencias digitales de la población rural y maximicen la apropiación efectiva de la herramienta tecnológica en el mediano plazo.

VI. REFERENCIAS BIBLIOGRÁFICAS



- Andreea. (13 de 02 de 2025). *littlepay*. Recuperado el 11 de 09 de 2025, de Cómo las soluciones de pago móvil están transformando el transporte público: <https://littlepay.com/resource-hub/blog/articles/mobile-payment-solutions-are-transforming-public-transportation/>
- Aguilera, C. (2023, 28 de junio). ¿Qué es la gestión de talento humano en una empresa? *ispring*. <https://www.ispring.es/blog/gestion-talento-humano>
- Basantes, S. J.-J. (13 de 06 de 2023). “PROPUESTA DEL MODELO DE GESTIÓN PARA MEJORAR LA CALIDAD DEL SERVICIO DE TRANSPORTE DE LA OPERADORA INTERPROVINCIAL RIOBAMBA”. Recuperado el 21 de 02 de 2025, de <https://dspace.esPOCH.edu.ec:8080/server/api/core/bitstreams/c1b388c0-477e-4235-8852-4f62d336db2f/content>
- Bello, E. (2021, 28 de abril). Descubre qué es el Extreme Programming y sus características. *IEBS*. <https://www.iebschool.com/hub/que-es-el-xp-programming-agile-scrum/>
- Carrasco Mena, V. F. (2023). *Desarrollo de una aplicación móvil para la gestión de pasajes de buses interprovinciales*. Quito : EPN, 2023. Recuperado el 17 de 09 de 2024, de <http://bibdigital.epn.edu.ec/handle/15000/24474>
- Ceballo, M. (7 de 05 de 2025). *areandina*. Recuperado el 23 de 07 de 2025, de ¿Qué es la ingeniería de software?: <https://www.areandina.edu.co/blogs/que-es-la-ingenieria-de-software>
- Cimas Cuadrado, G. (2020, 16 de julio). Qué es un lenguaje de programación: Guía para principiantes. *OpenWebinars*. <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>
- Cristancho, F. (2022, 26 de julio). ¿Qué es Flutter? *talently*. <https://talently.tech/blog/que-es-flutter/>
- Díaz, P. D.-M. (01 de 04 de 2022). *Herramientas tecnológicas utilizadas para la optimización de la gestión de transporte*. Recuperado el 05 de 03 de 2025, de Polo Conocimiento: <https://www.polodelconocimiento.com/ojs/index.php/es/article/view/3809/html>

- Díaz Salinas, J. S. (2024). *Formulación del plan de negocios para la aplicación Gobus Co móvil de transporte público urbano en el municipio de Neiva (Huila)* [Tesis doctoral, Corporación Universitaria Minuto de Dios-UNIMINUTO].
- García de Zúñiga, F. (2025, 3 de febrero). ¿Qué es un framework en programación y para qué sirve? ARSYS. <https://www.arsys.es/blog/que-es-un-framework-en-programacion-y-para-que-sirve>
- Garrido Sotomayor, S. (2024, 9 de noviembre). Metodologías ágiles: ¿Qué son y cuáles son más utilizadas? IEBS. <https://www.iebschool.com/hub/que-son-metodologias-agiles-agile-scrum/>
- Gilibets, L. (2023, 12 de enero). Qué es la metodología Kanban y cómo utilizarla. IEBS. <https://www.iebschool.com/hub/metodologia-kanban-agile-scrum/>
- González Ortiz, M. (2023, 31 de octubre). Digitalización en Recursos Humanos: Qué es y qué beneficios aporta. OpenWebinars. <https://openwebinars.net/blog/digitalizacion-en-recursos-humanos/>
- Hastwell, C. (2023, 20 de abril). ¿Qué es la gestión del talento? Definición, Estrategia, Procesos y Modelos. Great Place To Work. <https://blog.greatplacetowork.com.ar/blog/que-es-la-gestion-del-talento>
- Hernández, L. (14 de 06 de 2023). *Aleatica Smart y Sustainable Infrastructure*. Recuperado el 15 de 02 de 2023, de ¿Qué tipo de tecnologías aplican los sistemas de transporte inteligentes?: <https://www.aleatica.com/tecnologias-sistemas-de-transporte-inteligentes/>
- Hinestroza-C, A., Sanabria, Ó., & Beltrán-G., J. A. (2022). Conociendo el perfil socioeconómico de los productores para promover la adopción de tecnología. **Palmas: Revista de la Sociedad Colombiana de Palmicultores*, *43*(2), 15-25.*
- Lomelí, L. (2023, 26 de mayo). Metodología Scrum: Roles, Procesos y Artefactos. INNEVO. <https://innevo.com/blog/metodologia-scrum>
- López, M. (2021, 7 de octubre). ¿Qué es una base de datos? *Immune Institute*. <https://immune.institute/blog/base-de-datos-que-es-usos/>
- López Mora, S. (2020, 17 de mayo). Firebase: qué es, para qué sirve, funcionalidades y ventajas. DIGITAL55. <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>
- Martínez Córdón, M. (2021, 30 de junio). ¿Qué es el lenguaje de programación Dart? *hiberus Blog*. <https://www.hiberus.com/crecemos-contigo/que-es-el-lenguaje-de-programacion-dart/>
- Nebreda, M. (2025, 8 de abril). ¿Qué son las herramientas tecnológicas? *Campus Training*. <https://www.campustraining.es/noticias/que-son-herramientas-tecnologicas/>

- Paneiva, B. (2024, 27 de mayo). Checklist ISO 25010 de Requisitos de Calidad de Sistemas y Software. *Lumiform*. <https://lumiformapp.com/es/checklists-recursos/certificacion-iso-25010>
- Peña Cayturo, S. (2022). Aplicación Web para mejorar el monitoreo de los vehículos de Transporte Público en Abancay, 2018 [Tesis de maestría en ciencias de la computación, Universidad Nacional Micaela Bastidas de Apurímac]. Repositorio UNAMBA. <https://repositorio.unamba.edu.pe/handle/UNAMBA/1116>
- Rehman, A. (2024, 22 de agosto). Estándar IEEE para especificaciones de requisitos de software (IEEE 830-1998). *Medium*. https://medium.com/@abdul.rehman_84899/ieee-standard-for-software-requirements-specifications-ieee-830-1998-0395f1da639a
- Román, E. (2024, 30 de agosto). Entendiendo la Calidad del Software: qué es, qué la compone y estándares principales. *INNEVO*. <https://innevo.com/blog/calidad-en-el-desarrollo-de-software>
- Route, S. (12 de 11 de 2023). *simpliroute*. Recuperado el 23 de 02 de 2026, de Tecnología en el Transporte: Factores Clave a Tomar en Cuenta: <https://simpliroute.com/es/blog/tecnologia-en-el-transporte>
- Suárez, E. (2024, 9 de febrero). Método inductivo y deductivo. *Experto Universitario*. <https://expertouniversitario.es/blog/metodo-inductivo-y-deductivo/>
- Vieira, D. (2024, 27 de marzo). Herramientas de desarrollo: las mejores opciones. *HostGator*. <https://www.hostgator.net.ec/blog/herramientas-desarrollo/>

VII. ANEXOS

Anexo 1. Acta de la sustentación de Predefensa del TIC

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

ACTA

DE LA SUSTENTACIÓN ORAL DE LA PREDEFENSA DEL TRABAJO DE INTEGRACIÓN CURRICULAR


ESTUDIANTE: Pascal Chamba Nury Gabriela	CÉDULA DE IDENTIDAD: 0402174965
PERIODO ACADÉMICO: 2026 A	DOCENTE TUTOR: MSC. Milton Gabriel del Hierro Mosquera
PRESIDENTE TRIBUNAL: MSC. Luis Patiño Hernández	
DOCENTE: Ph.D. Samuel Lascano Rivera	
TEMA DEL TIC: "Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público"	

No.	CATEGORÍA	Evaluación cuantitativa	OBSERVACIONES Y RECOMENDACIONES
1	PROBLEMA - OBJETIVOS	10,00	
2	FUNDAMENTACIÓN TEÓRICA	10,00	
3	METODOLOGÍA	10,00	
4	RESULTADOS	10,00	
5	DISCUSIÓN	10,00	
6	CONCLUSIONES Y RECOMENDACIONES	10,00	
7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	9,67	
8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	9,00	Revisar la redacción y ortografía del documento

Obteniendo una nota de: **9,67** Por lo tanto, **APRUEBA** ; debiendo el o los investigadores acatar el siguiente artículo:

Art. 36.- De los estudiantes que aprueban el Informe final del TIC con observaciones.- Los estudiantes tendrán el plazo de 10 días para proceder a corregir su informe final del TIC de conformidad a las observaciones y recomendaciones realizadas por los miembros del Tribunal de sustentación de la pre-defensa.


Para constancia del presente, firman en la ciudad de Tulcán el **miércoles, 18 de febrero de 2026**



MSC. Luis Patiño Hernández
PRESIDENTE TRIBUNAL



MSC. Milton Gabriel del Hierro Mosquera
DOCENTE TUTOR



Ph.D. Samuel Lascano Rivera
DOCENTE



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

ACTA

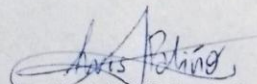
DE LA SUSTENTACIÓN ORAL DE LA PREDEFENSA DEL TRABAJO DE INTEGRACIÓN CURRICULAR

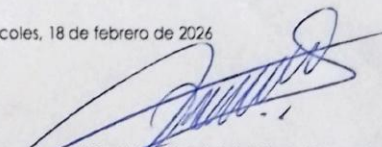
ESTUDIANTE:	Rosero Narváez Crislian Paul	CÉDULA DE IDENTIDAD:	0401770342
PERIODO ACADÉMICO:	2026 A		
PRESIDENTE TRIBUNAL:	Msc. Luis Patiño Hernández	DOCENTE TUTOR:	MSc. Milton Gabriel del Hierro Mosquera
DOCENTE:	PhD. Samuel Lascano Rivera		
TEMA DEL TIC:	Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público		
No.	CATEGORÍA	Evaluación cuantitativa	OBSERVACIONES Y RECOMENDACIONES
1	PROBLEMA - OBJETIVOS	10,00	
2	FUNDAMENTACIÓN TEÓRICA	10,00	
3	METODOLOGÍA	10,00	
4	RESULTADOS	10,00	
5	DISCUSIÓN	10,00	
6	CONCLUSIONES Y RECOMENDACIONES	10,00	
7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	10,00	
8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	9,00	Revisar la redacción y ortografía del documento

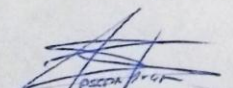
Obteniendo una nota de: **9,70** Por lo tanto, **APRUEBA** : debiendo el o los investigadores acatar el siguiente artículo:

Art. 36.- De los estudiantes que aprueban el informe final del TIC con observaciones.- Los estudiantes tendrán el plazo de 10 días para proceder a corregir su informe final del TIC de conformidad a las observaciones y recomendaciones realizadas por los miembros del Tribunal de sustentación de la pre-defensa.

Para constancia del presente, firman en la ciudad de Tulcán el **miércoles, 18 de febrero de 2026**


Msc. Luis Patiño Hernández
PRESIDENTE TRIBUNAL


MSc. Milton Gabriel del Hierro Mosquera
DOCENTE TUTOR


PhD. Samuel Lascano Rivera
DOCENTE

Anexo 2. Certificado del abstract por parte de idiomas



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI FOREIGN
AND NATIVE LANGUAGES CENTER

ABSTRACT- EVALUATION SHEET				
NAME: Cristian Paul Rosero Narvaez y Nury Gabriela Pascal Chamba				
DATE: Jueves, 5 de marzo de 2026				
Topic: "Herramienta tecnológica para la gestión de talento humano y venta de boletos del transporte público"				
MARKS AWARDED		QUANTITATIVE AND QUALITATIVE		
VOCABULARY AND WORD USE	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
WRITING COHESION	Clear and logical progression of ideas and supporting paragraphs.	Adequate progression of ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate ideas and supporting paragraphs.
De	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
ARGUMENT	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
CREATIVITY	Outstanding flow of ideas and events	Good flow of ideas and events	Average flow of ideas and events	Poor flow of ideas and events
	EXCELLENT: 2 <input type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
SCIENTIFIC SUSTAINABILITY	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
TOTAL/AVERAGE	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED	TOTAL 9		



**UNIVERSIDAD POLITÉCNICA ESTATAL DEL
CARCHI- FOREIGN AND NATIVE LANGUAGES
CENTER**

**Informe sobre el Abstract de Artículo Científico
o Investigación.**

Autor: Cristian Paul Rosero Narvaez y Nury Gabriela Pascal Chamba

Fecha de recepción del abstract: Jueves, 5 de marzo de 2026

Fecha de entrega del informe: Jueves, 5 de marzo de 2026

El presente informe validará la traducción del idioma español al inglés si alcanza un porcentaje de: 9 – 10 Excelente.

Si la traducción no está dentro de los parámetros de 9 – 10, el autor deberá realizar las observaciones presentadas en el ABSTRACT, para su posterior presentación y aprobación.

Observaciones:

Después de realizar la revisión del presente abstract, éste presenta una apropiada traducción sobre el tema planteado en el idioma Inglés. Según la rúbrica de evaluación de la traducción en Inglés, ésta alcanza un valor de 9; por lo cual se valida dicho trabajo.

Atentamente



MA. Martha Viveros
RESPONSABLE CIDEN

Anexo 3. Código de la app Trans Doramald

```
agregar_ruta.dart | main.dart 1 | verificacion_de_pagos.dart 5 | crear_notificaiion.dart 1 | iniciarsesion.dart 1
lib > administrador > agregar_ruta.dart > _AgregarNombrePageState > build
13 class _AgregarNombrePageState extends State<AgregarNombrePage> {
23   Widget build(BuildContext context) {
34     padding: const EdgeInsets.all(16),
35     sliver: SliverList(
36       delegate: SliverChildListDelegate([
37         _buildOptionCard(
38           context: context,
39           icon: Icons.add_road_rounded,
40           iconColor: successGreen,
41           title: 'Agregar Ruta',
42           subtitle: 'Nueva ruta',
43           description: 'Registra nuevas rutas y destinos en el sistema',
44           features: [
45             'Gestión rápida',
46             'Múltiples orígenes',
47             'Fácil configuración',
48           ],
49           onTap: () {
50             Navigator.push(
51               context,
52               MaterialPageRoute(
53                 builder: (context) => const AgregarRutaScreen(),
54               ), // MaterialPageRoute
55             );
56           },
57           isTablet: isTablet,
58         ),
59         const SizedBox(height: 12),
60         _buildOptionCard(
61           context: context,
62           icon: Icons.list_alt_rounded,
63           iconColor: primaryBusBlue,
64           title: 'Ver Rutas',
65           subtitle: 'Administración'
```

```
agregar_ruta.dart x main.dart 1 verificacion_de_pagos.dart 5 crear_notificaion.dart 1 iniciarsesio

lib > administrador > agregar_ruta.dart > _AgregarNombrePageState > build
10 class _AgregarNombrePageState extends State<AgregarNombrePage> {
548 }
549
550 // ===== PANTALLA AGREGAR RUTA =====
551 class AgregarRutaScreen extends StatefulWidget {
552   const AgregarRutaScreen({super.key});
553
554   @override
555   _AgregarRutaScreenState createState() => _AgregarRutaScreenState();
556 }
557
558 class _AgregarRutaScreenState extends State<AgregarRutaScreen> {
559   final TextEditingController _nameController = TextEditingController();
560   final TextEditingController _priceController = TextEditingController();
561   final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
562   bool _isLoading = false;
563
564   String _origenSeleccionado = 'Tulcán';
565
566   static const Color primaryBusBlue = Color(0xFF1E40AF);
567   static const Color darkNavy = Color(0xFF0F172A);
568   static const Color lightBg = Color(0xFFF1F5F9);
569   static const Color textGray = Color(0xFF475569);
570   static const Color successGreen = Color(0xFF059669);
571
572   @override
573   void dispose() {
574     _nameController.dispose();
575     _priceController.dispose();
576     super.dispose();
577   }
578
579   Future<void> _agregarRuta() async {
580     if (!_formKey.currentState!.validate()) return;
581
```

```
cambiar_imagenes_de_inicio.dart 2 x
lib > administrador > cambiar_imagenes_de_inicio.dart > _ImageManagementPageState
7 class ImageManagementPage extends StatefulWidget {
10   @override
11   ImageManagementPageState createState() => ImageManagementPageState();
12 }
13
14 class ImageManagementPageState extends State<ImageManagementPage>
15   with SingleTickerProviderStateMixin {
16   final ImagePicker _picker = ImagePicker();
17   File? _selectedImage;
18   List<String> imageUrls = [];
19   bool _isLoading = false;
20   bool _isUploading = false;
21   late AnimationController _animationController;
22
23   // Paleta de colores moderna
24   static const Color primaryBusBlue = Color(0xFF940016);
25   static const Color accentOrange = Color(0xFFEA580C);
26   static const Color darkNavy = Color(0xFF0F172A);
27   static const Color roadGray = Color(0xFF334155);
28   static const Color lightBg = Color(0xFFFF1F5F9);
29   static const Color textGray = Color(0xFF475569);
30   static const Color successGreen = Color(0xFF059669);
31   static const Color accentRed = Color(0xFFEF4444);
32
33   @override
34   void initState() {
35     super.initState();
36     _animationController = AnimationController(
37       vsync: this,
38       duration: const Duration(milliseconds: 300),
39     ); // AnimationController
40     fetchImageUrls();
41   }
42
43   @override
```

```

cambiar_imagenes_de_inicio.dart 2 x
b > administrador > cambiar_imagenes_de_inicio.dart > _ImageManagementPageState
14 class _ImageManagementPageState extends State<ImageManagementPage>
60 Future<void> _takePhoto() async {
69 }
70
71 Future<void> _uploadImage() async {
72   if (_selectedImage == null) {
73     _showSnackBar('Por favor selecciona una imagen primero', ■accentOrange);
74     return;
75   }
76
77   setState(() {
78     _isUploading = true;
79   });
80
81   final storageRef = FirebaseStorage.instance.ref();
82   final imageRef = storageRef
83     .child('informacion/${DateTime.now().millisecondsSinceEpoch}.jpg');
84
85   try {
86     await imageRef.putFile(_selectedImage!);
87     final imageUrl = await imageRef.getDownloadURL();
88     setState(() {
89       imageUrls.insert(0, imageUrl);
90       _selectedImage = null;
91       _isUploading = false;
92     });
93     _animationController.reverse();
94     _showSnackBar('Imagen subida exitosamente!', ■successGreen);
95   } catch (e) {
96     setState(() {
97       _isUploading = false;
98     });
99     _showSnackBar('Error al subir la imagen', ■accentRed);
100    print('Error al subir la imagen: $e');

```

```
crear_buses.dart 2 x
lib > administrador > crear_buses.dart > _CrearBusScreenState > _buildBusCard
7 class CrearBusScreen extends StatefulWidget {
9   _CrearBusScreenState createState() => _CrearBusScreenState();
10 }
11
12 class _CrearBusScreenState extends State<CrearBusScreen> {
13   final _formKey = GlobalKey<FormState>();
14
15   // Variables para los selectores
16   String? _numeroSeleccionado;
17   String? _placaSeleccionada;
18   String? _capacidadSeleccionada;
19   String? _choferSeleccionado;
20   String? _lugarSeleccionado;
21   DateTime? _fechaSalida;
22   TimeOfDay? _horaSalida;
23
24   String _normalizarTexto(String texto) {
25     return texto
26       .toLowerCase()
27       .replaceAll('á', 'a')
28       .replaceAll('é', 'e')
29       .replaceAll('í', 'i')
30       .replaceAll('ó', 'o')
31       .replaceAll('ú', 'u')
32       .replaceAll('ñ', 'n')
33       .trim();
34   }
35
36   // Listas para las opciones
37   List<Map<String, dynamic>> _buses = [];
38   List<String> _lugares = [];
39
40   final db = FirebaseFirestore.instance;
41
```

```

lib > administrador > crear_buses.dart > _CrearBusScreenState > _buildBusCard
12 class _CrearBusScreenState extends State<CrearBusScreen> {
62
63 // =====
64 // ♦ CARGAR OPCIONES DESDE FIREBASE
65 // =====
66 Future<void> _cargarOpcionesDesdeFirebase() async {
67   setState(() {
68     _cargandoDatos = true;
69   });
70
71   try {
72     final busesSnapshot =
73       await db.collection('conductores_registrados').get();
74     _buses = busesSnapshot.docs.map((doc) {
75       var data = doc.data();
76       data['id'] = doc.id;
77       return data;
78     }).toList();
79
80     final lugaresSnapshot = await db.collection('lugares_salida').get();
81     _lugares =
82       lugaresSnapshot.docs.map((doc) => doc['lugar'].toString()).toList();
83
84     setState(() {
85       _cargandoDatos = false;
86     });
87   } catch (e) {
88     print('Error al cargar opciones: $e');
89     setState(() {
90       _cargandoDatos = false;
91     });
92
93     if (mounted) {
94       _mostrarSnackBar('Error al cargar opciones: $e', esError: true);
95     }
96   }
97 }

```

```
crear_buses.dart 2  crear_notificaion.dart 1 x
> administrador > crear_notificaion.dart > _CrearNotificacionScreenState > _enviarNotificacionPushATodos
16 class _CrearNotificacionScreenState extends State<CrearNotificacionScreen>
59 @override
60 void dispose() {
61   _fadeController.dispose();
62   _mensajeController.dispose();
63   super.dispose();
64 }
65
66 /// 📢 Envía notificación push a TODOS los usuarios (versión corregida)
67 Future<Map<String, dynamic>> _enviarNotificacionPushATodos({
68   required String titulo,
69   required String mensaje,
70 }) async {
71   try {
72     // ⚠️ IMPORTANTE: Cambia esta URL por la tuya de Render
73     const String baseUrl = 'https://notificaciones-1hoa.onrender.com';
74
75     print('📢 Enviando notificación a TODOS los usuarios...');
76     print('🔗 URL: $baseUrl/api/notifications/send-to-all');
77
78     final response = await http
79       .post(
80         Uri.parse('$baseUrl/api/notifications/send-to-all'),
81         headers: {'Content-Type': 'application/json'},
82         body: json.encode({
83           'title': titulo,
84           'body': mensaje,
85           'data': {
86             'tipo': 'anuncio_global',
87             'timestamp': DateTime.now().toIso8601String(),
88             'route': '/notificaciones', // Ruta opcional en tu app
89           },
90         })),
91   )

```

```

crear_notificaion.dart 1 x
> administrador > crear_notificaion.dart > _CrearNotificacionScreenState > _enviarNotificacionPushATodos
16 class _CrearNotificacionScreenState extends State<CrearNotificacionScreen>
17   Future<Map<String, dynamic>> _enviarNotificacionPushATodos({
18     print(' 📄 Respuesta del servidor: ${response.statusCode} ');
19
20     if (response.statusCode == 200) {
21       final responseData = json.decode(response.body);
22
23       print(' ✅ Notificaciones enviadas exitosamente');
24       print(' 📄 Total usuarios: ${responseData['totalUsers']}');
25       print(' ✓ Exitosas: ${responseData['successCount']}');
26       print(' ✗ Fallidas: ${responseData['failureCount']}');
27
28       return {
29         'success': true,
30         'totalUsers': responseData['totalUsers'] ?? 0,
31         'successCount': responseData['successCount'] ?? 0,
32         'failureCount': responseData['failureCount'] ?? 0,
33       };
34     } else {
35       print(' ⚠️ Error del servidor: ${response.statusCode}');
36       print(' 📄 Respuesta: ${response.body}');
37
38       return {
39         'success': false,
40         'error': 'Error del servidor: ${response.statusCode}',
41       };
42     }
43   } catch (e) {
44     print(' ❌ Error al enviar notificaciones push: $e');
45
46     return {
47       'success': false,
48       'error': e.toString(),
49     };
50   }

```

```
lib > administrador > detalles_encomienda.dart > _DetalleEncomiendaScreenState > _crearNotificacion
```

```
12 class DetalleEncomiendaScreen extends StatefulWidget {
13   final String codigo;
14   final Map<String, dynamic> data;
15   final String estado;
16
17   const DetalleEncomiendaScreen({
18     super.key,
19     required this.codigo,
20     required this.data,
21     required this.estado,
22   });
23
24   @override
25   State<DetalleEncomiendaScreen> createState() =>
26     DetalleEncomiendaScreenState();
27 }
28
29 class DetalleEncomiendaScreenState extends State<DetalleEncomiendaScreen> {
30   static const Color darkGray = Color(0xFF2D3748);
31   static const Color lightGray = Color(0xFFFF7F4FC);
32   static const Color mediumGray = Color(0xFF718096);
33   static const Color accentBlue = Color(0xFF3B82F6);
34   static const Color successGreen = Color(0xFF10B981);
35   static const Color warningRed = Color(0xFFEF4444);
36   static const Color accentOrange = Color(0xFFFF59E0B);
37
38   final TextEditingController _precioTipoController = TextEditingController();
39   final TextEditingController _pesoController =
40     TextEditingController(); //  NUEVO
41   XFile? _imagenTransito;
42   XFile? _imagenEntregada;
43   String? _busSeleccionado;
44   List<Map<String, dynamic>> _busesDisponibles = [];
```

```
lib > administrador > detalles_encomienda.dart > _DetalleEncomiendaScreenState > _crearNotificacion
29 class _DetalleEncomiendaScreenState extends State<DetalleEncomiendaScreen> {
67 Future<void> _cargarBuses() async {
68   print('🔵 ===== INICIANDO CARGA DE BUSES =====');
69   print('🔵 Cargando TODOS los buses de ambas colecciones...');
70
71   try {
72     List<Map<String, dynamic>> buses = [];
73
74     // 🔵 CARGAR TODOS LOS BUSES DE COLECCIÓN "buses" (Tulcán)
75     print('\n🔵 Cargando todos los buses de colección "buses"...');
76     final busesSnapshot = FirebaseFirestore.instance
77       .collection('buses_tulcan_salida')
78       .get();
79
80     print('📊 Total en "buses": ${busesSnapshot.docs.length}');
81
82     for (var doc in busesSnapshot.docs) {
83       final data = doc.data();
84
85       print('🚌 Bus agregado:');
86       print('   - ID: ${doc.id}');
87       print('   - Número: ${data['numero']}');
88       print('   - Lugar salida: ${data['lugar_salida']}');
89       print('   - Hora: ${data['hora_salida']}');
90
91       buses.add({
92         'id': doc.id,
93         'coleccion': 'buses_tulcan_salida',
94         'numero': data['numero']?.toString() ?? 'S/N',
95         'lugar_salida': data['lugar_salida']?.toString() ?? '',
96         'fecha_salida': data['fecha_salida'],
97         'hora_salida': data['hora_salida']?.toString() ?? '',
98         'ruta': data['ruta']?.toString() ?? '',
99       });

```

```
lib > administrador > gestion_de_datos.dart > _GestionDatosScreenState > _buildHeader
6 class GestionDatosScreen extends StatefulWidget {
7   @override
8   GestionDatosScreenState createState() => _GestionDatosScreenState();
9 }
10
11 class _GestionDatosScreenState extends State<GestionDatosScreen> {
12   final db = FirebaseFirestore.instance;
13
14   // Controladores
15   final _numeroCtrl = TextEditingController();
16   final _placaCtrl = TextEditingController();
17   final _capacidadCtrl = TextEditingController();
18   final _choferCtrl = TextEditingController();
19   final _lugarCtrl = TextEditingController();
20
21   // FocusNodes
22   final _numeroFocus = FocusNode();
23   final _placaFocus = FocusNode();
24   final _capacidadFocus = FocusNode();
25   final _choferFocus = FocusNode();
26   final _lugarFocus = FocusNode();
27
28   // Paleta de colores moderna
29   static const Color primaryBusBlue = Color(0xFF940016);
30   static const Color accentOrange = Color(0xFFEA580C);
31   static const Color darkNavy = Color(0xFF0F172A);
32   static const Color roadGray = Color(0xFF334155);
33   static const Color lightBg = Color(0xFFF1F5F9);
34   static const Color textGray = Color(0xFF475569);
35   static const Color successGreen = Color(0xFF940016);
36   static const Color accentRed = Color(0xFFEF4444);
37
38   int _currentTab = 0;
39 }
```

```
lib > administrador > gestion_de_datos.dart > _GestionDatosScreenState > _buildHeader
11 class _GestionDatosScreenState extends State<GestionDatosScreen> {
140 Future<void> _agregarLugar() async {
144     }
145
146     try {
147         final existente = await db
148             .collection('lugares_salida')
149             .where('lugar', isEqualTo: _lugarCtrl.text.trim())
150             .get();
151
152         if (existente.docs.isNotEmpty) {
153             _mostrarSnackBar(
154                 'El lugar "${_lugarCtrl.text.trim()}" ya existe',
155                 accentRed,
156             );
157             return;
158         }
159
160         await db.collection('lugares_salida').add({
161             'lugar': _lugarCtrl.text.trim(),
162             'fecha_creacion': Timestamp.now(),
163         });
164
165         _mostrarSnackBar('✓ Lugar agregado correctamente', successGreen);
166
167         _lugarCtrl.clear();
168         FocusScope.of(context).requestFocus(_lugarFocus);
169     } catch (e) {
170         _mostrarSnackBar('Error al agregar: $e', accentRed);
171     }
172 }
173
174 Future<void> _eliminarBus(String id, String numero) async {
175     bool? confirmar = await showDialog<bool>(  

```

```
lib > administrador > Gestion_encomiendas.dart > ...
17 class ConfiguracionEncomiendas extends StatefulWidget {
18   const ConfiguracionEncomiendas({super.key});
19
20   @override
21   State<ConfiguracionEncomiendas> createState() =>
22     _ConfiguracionEncomiendasState();
23 }
24
25 const String NOTIFICATIONS_SERVER_URL =
26   'https://tu-servidor-render.onrender.com';
27
28 class _ConfiguracionEncomiendasState extends State<ConfiguracionEncomiendas>
29   with SingleTickerProviderStateMixin {
30   // Paleta de colores (misma que MenuOpcionesScreen)
31   static const Color primaryBusBlue = Color(0xFF1E40AF);
32   static const Color accentOrange = Color(0xFFEA580C);
33   static const Color darkNavy = Color(0xFF0F172A);
34   static const Color roadGray = Color(0xFF334155);
35   static const Color lightBg = Color(0xFFF1F5F9);
36   static const Color textGray = Color(0xFF475569);
37   static const Color successGreen = Color(0xFF059669);
38   static const Color errorRed = Color(0xFFEF4444);
39   static const Color warningYellow = Color(0xFFFF590B);
40   static const Color mainRed = Color(0xFF940016);
41
42   late TabController _tabController;
43
44   @override
45   void initState() {
46     super.initState();
47     _tabController = TabController(length: 4, vsync: this);
48     _procesarImagenesPendientes();
49   }
50 }
```

```
Gestion_encomiendas.dart 9+ x
lib > administrador > Gestion_encomiendas.dart > ...
28 class ConfiguracionEncomiendasState extends State<ConfiguracionEncomiendas>
51 Future<void> _procesarImágenesPendientes() async {
70   await FirebaseFirestore.instance
71     .collection('encomiendas_registradas')
72     .doc(data['codigo_envio'])
73     .update({data['campo']: url});
74
75   print('✅ Imagen subida: ${data['campo']}');
76
77   if (data['campo'] == 'imagenes.entregada' &&
78       data['uid_remitente'] != null) {
79     final encomiendaDoc = await FirebaseFirestore.instance
80       .collection('encomiendas_registradas')
81       .doc(data['codigo_envio'])
82       .get();
83
84     final encomiendaData = encomiendaDoc.data();
85     final correo = encomiendaData?['remitente']?['correo'] ?? '';
86     final nombre = encomiendaData?['remitente']?['nombre'] ?? '';
87
88     await FirebaseFirestore.instance
89       .collection('notificaciones')
90       .add({
91         'uid': data['uid_remitente'],
92         'correo': correo,
93         'nombre_remitente': nombre,
94         'titulo': 'Encomienda Entregada ✅',
95         'mensaje':
96           'Tu encomienda ${data['codigo_envio']} ha sido entregada exitosamente.',
97         'codigo_encomienda': data['codigo_envio'],
98         'estado': 'entregado',
99         'leida': false,
100        'fecha': FieldValue.serverTimestamp(),
101        'tipo': 'encomienda'
```

```
Gestion_encomiendas.dart 9+  ingresar_tipos_encomienda.dart x
lib > administrador > ingresar_tipos_encomienda.dart > _TiposEncomiendaTabState > build
9  class TiposEncomiendaTab extends StatefulWidget {
10     const TiposEncomiendaTab({super.key});
11     @override
12     State<TiposEncomiendaTab> createState() => _TiposEncomiendaTabState();
13 }
14
15 class _TiposEncomiendaTabState extends State<TiposEncomiendaTab> {
16     static const Color lightGray = Color(0xFF77FAFC);
17     static const Color mediumGray = Color(0xFF718096);
18     static const Color accentBlue = Color(0xFF3B82F6);
19     static const Color successGreen = Color(0xFF10B981);
20     static const Color warningRed = Color(0xFFEF4444);
21     void _mostrarDialogoAgregar() {
22         final tipoController = TextEditingController();
23         final descripcionController = TextEditingController();
24         final precioController = TextEditingController();
25         showDialog(
26             context: context,
27             builder: (context) => AlertDialog(
28                 shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(16)),
29                 title: Row(
30                     children: [
31                         Container(
32                             padding: const EdgeInsets.all(8),
33                             decoration: BoxDecoration(
34                                 color: accentBlue.withOpacity(0.1),
35                                 borderRadius: BorderRadius.circular(8),
36                             ), // BoxDecoration
37                             child: const Icon(Icons.add_box, color: accentBlue, size: 24),
38                         ), // Container
39                         const SizedBox(width: 12),
40                         const Text(
41                             'Nuevo Tipo de Encomienda',
42                             style: TextStyle(fontSize: 18, fontWeight: FontWeight.w700),
```

```
Gestion_encomiendas.dart 9+  ingresar_tipos_encomienda.dart x
ib > administrador > ingresar_tipos_encomienda.dart > _TiposEncomiendaTabState > build
15 class _TiposEncomiendaTabState extends State<TiposEncomiendaTab> {
21 void _mostrarDialogoAgregar() {
107     await Firestore.instance
108         .collection('Tipo_encomienda')
109         .add({
110             'Tipo': tipoController.text.trim(),
111             'descripcion': descripcionController.text.trim(),
112             'precio_adicional': double.parse(precioController.text),
113             'activo': true,
114             'fecha_creacion': FieldValue.serverTimestamp(),
115         });
116
117     if (mounted) {
118         Navigator.pop(context);
119         ScaffoldMessenger.of(context).showSnackBar(
120             const SnackBar(
121                 content: Text('Tipo de encomienda agregado exitosamente'),
122                 backgroundColor: successGreen,
123             ), // SnackBar
124         );
125     }
126 } catch (e) {
127     ScaffoldMessenger.of(context).showSnackBar(
128         SnackBar(
129             content: Text('Error: $e'),
130             backgroundColor: warningRed,
131         ), // SnackBar
132     );
133 }
134 },
135 style: ElevatedButton.styleFrom(
136     backgroundColor: accentBlue,
137     shape: RoundedRectangleBorder(
138         borderRadius: BorderRadius.circular(10)), // RoundedRectangleBorder
```

```

verificacion_de_pagos.dart 5
lib > administrador > verificacion_de_pagos.dart > _AdminVerificacionPagosScreenState > _enviarNotificacionPush
14 class AdminVerificacionPagosScreen extends StatefulWidget {
15   const AdminVerificacionPagosScreen({Key? key}) : super(key: key);
16
17   @override
18   AdminVerificacionPagosScreenState createState() =>
19     AdminVerificacionPagosScreenState();
20 }
21
22 class AdminVerificacionPagosScreenState
23   extends State<AdminVerificacionPagosScreen> {
24   final FirebaseFirestore db = FirebaseFirestore.instance;
25   final FirebaseFunctions functions = FirebaseFunctions.instance;
26
27   String? cooperativaSeleccionada;
28   String? busSeleccionado;
29   String filtroEstado = 'pendiente_verificacion';
30   String coleccionActual = 'reservas';
31
32   // Paleta de colores moderna
33   static const Color primaryBusBlue = Color(0xFF940016);
34   static const Color accentOrange = Color(0xFFEA580C);
35   static const Color darkNavy = Color(0xFF0F172A);
36   static const Color roadGray = Color(0xFF334155);
37   static const Color lightBg = Color(0xFFF1F5F9);
38   static const Color textGray = Color(0xFF475569);
39   static const Color successGreen = Color(0xFF059669);
40   static const Color accentRed = Color(0xFF940016);
41   static const Color warningYellow = Color(0xFFF59E0B);
42
43   String get coleccionBuses {
44     if (cooperativaSeleccionada == 'la_esperanza') {
45       return 'buses_la_esperanza_salida';
46     } else if (cooperativaSeleccionada == 'tulcan') {
47       return 'buses_tulcan_salida';

```

```
verificacion_de_pagos.dart 5 x
lib > administrador > verificacion_de_pagos.dart > AdminVerificacionPagosScreenState > _enviarNotificacionPush
22 class AdminVerificacionPagosScreenState
73   Widget _buildHeader(BuildContext context) {
186     ), // InkWell
187   ],
188   ), // Row
189   const SizedBox(height: 28),
190   Text(
191     'Gestión de Pagos',
192     style: TextStyle(
193       fontSize: 30,
194       fontWeight: FontWeight.w900,
195       color: darkNavy,
196       height: 1.1,
197       letterSpacing: -0.5,
198     ), // TextStyle
199   ), // Text
200   const SizedBox(height: 8),
201   Text(
202     'Verifica y aprueba los comprobantes de transferencia',
203     style: TextStyle(
204       fontSize: 14,
205       color: textGray,
206       fontWeight: FontWeight.w500,
207       height: 1.4,
208     ), // TextStyle
209   ), // Text
210   const SizedBox(height: 20),
211   Row(
212     children: [
213       _buildStatChip(Icons.verified_rounded, 'Seguro'),
214       const SizedBox(width: 10),
215       _buildStatChip(Icons.speed_rounded, 'Rápido'),
216     ],
217   ), // Row
```

usuario > compra_de_boletos > De_ida > Datos_del_comprador.dart > _DatosCompradorScreenState > _get

```
27
28 class _DatosCompradorScreenState extends State<DatosCompradorScreen>
29   with SingleTickerProviderStateMixin {
30   final _formKey = GlobalKey<FormState>();
31   final TextEditingController _nombreController = TextEditingController();
32   final TextEditingController _cedulaController = TextEditingController();
33   final TextEditingController _celularController = TextEditingController();
34
35   final FocusNode _nombreFocus = FocusNode();
36   final FocusNode _cedulaFocus = FocusNode();
37   final FocusNode _celularFocus = FocusNode();
38
39   late AnimationController _fadeController;
40   late Animation<double> _fadeAnimation;
41
42   bool _isLoading = false;
43   String _paisSeleccionado = 'Ecuador'; // País por defecto
44
45   // Lista de países disponibles
46   final List<String> _paises = [
47     'Ecuador',
48     'Colombia',
49     'Perú',
50     'Venezuela',
51     'Otro',
52   ];
53
54   // Paleta de colores inspirada en transporte de buses
55   final Color primaryBusBlue = Color.fromARGB(255, 243, 248, 255);
56   final Color accentOrange = Color(0xFFEA580C);
57   final Color darkNavy = Color(0xFF0F172A);
58   final Color roadGray = Color(0xFF334155);
59   final Color lightBg = Color(0xFFF1F5F9);
60   final Color textGray = Color(0xFF475569);
```

```

Datos_del_comprador.dart x
lib > usuario > compra_de_boletos > De_ida > Datos_del_comprador.dart > _DatosCompradorScreenState > _getConfiguracion
28 class _DatosCompradorScreenState extends State<DatosCompradorScreen>
204 widget _buildHeader() {
259     Text(
260         'DATOS DEL PASAJERO',
261         style: TextStyle(
262             fontSize: 14,
263             fontWeight: FontWeight.w900,
264             color: Color.fromARGB(255, 36, 35, 35),
265             letterSpacing: 1,
266         ), // TextStyle
267     ), // Text
268     Text(
269         'Completa tu información',
270         style: TextStyle(
271             fontSize: 10,
272             fontWeight: FontWeight.w500,
273             color: Color.fromARGB(255, 38, 38, 39),
274             letterSpacing: 0.5,
275         ), // TextStyle
276     ), // Text
277 ],
278 ), // Column
279 ],
280 ), // Row
281 InkWell(
282     borderRadius: BorderRadius.circular(12),
283     onTap: () {},
284     child: Container(
285         padding: const EdgeInsets.all(10),
286         decoration: BoxDecoration(
287             color: const Color.fromARGB(0, 255, 255, 255),
288             borderRadius: BorderRadius.circular(12),
289             boxShadow: [
290                 BoxShadow(

```

```
Datos_del_comprador.dart  Lista_de_buses_de_ida.dart x
lib > usuario > compra_de_boletos > De_ida > Lista_de_buses_de_ida.dart > _ListaBusesScreenState > build
8
9 class ListaBusesScreen extends StatefulWidget {
10   const ListaBusesScreen({Key? key}) : super(key: key);
11
12   @override
13   _ListaBusesScreenState createState() => _ListaBusesScreenState();
14 }
15
16 class _ListaBusesScreenState extends State<ListaBusesScreen>
17   with SingleTickerProviderStateMixin {
18   // Paleta empresarial
19   final Color primaryNavy = const Color(0xFF1A2332);
20   final Color darkGray = const Color(0xFF2D3748);
21   final Color lightGray = const Color(0xFFF7FAFC);
22   final Color mediumGray = const Color(0xFF718096);
23   final Color successGreen = const Color(0xFF10B981);
24   final Color warningOrange = const Color(0xFFF59E0B);
25   final Color errorRed = const Color(0xFFE44444);
26   final Color accentBlue = const Color(0xFF3B82F6);
27
28   late AnimationController _animationController;
29   late Animation<double> _fadeAnimation;
30
31   @override
32   void initState() {
33     super.initState();
34     _animationController = AnimationController(
35       duration: const Duration(milliseconds: 800),
36       vsync: this,
37     ); // AnimationController
38     _fadeAnimation = Tween<double>(begin: 0.0, end: 1.0).animate(
39       CurvedAnimation(
40         parent: _animationController,
41         curve: Curves.easeOut,
42       ),
43     );
44   }
45 }
```

```
Datos_del_comprador.dart  Lista_de_buses_de_ida.dart x
b > usuario > compra_de_boletos > De_ida > Lista_de_buses_de_ida.dart > _ListaBusesScreenState > build
16 class _ListaBusesScreenState extends State<ListaBusesScreen>
102 Widget build(BuildContext context) {
16   child: StreamBuilder<QuerySnapshot>(
17     stream: db.collection('buses_tulcan_salida').snapshots(),
18     builder: (context, snapshot) {
19       if (snapshot.connectionState == ConnectionState.waiting) {
20         return _buildLoadingState(isTablet);
21       }
22
23       if (snapshot.hasError) {
24         return _buildErrorState(isTablet);
25       }
26
27       if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {
28         return _buildEmptyState(isTablet);
29       }
30
31       var buses = snapshot.data!.docs;
32       var busesDisponibles = buses.where((busDoc) {
33         var busData = busDoc.data() as Map<String, dynamic>;
34         var fechaSalida = busData['fecha_salida'] != null
35           ? (busData['fecha_salida'] as Timestamp).toDate()
36           : (busData['fechaSalida'] != null
37             ? (busData['fechaSalida'] as Timestamp).toDate()
38             : null);
39
40         if (fechaSalida == null) return false;
41
42         String horaSalida =
43           busData['hora_salida'] ?? busData['horaSalida'] ?? '';
44         if (horaSalida.isEmpty) return false;
45
46         DateTime? fechaHoraSalida =
47           parsearHoraSalida(horaSalida, fechaSalida);
```

```
pagos_desde_tulcan.dart x
lib > usuario > compra_de_boletos > De_ida > pagos_desde_tulcan.dart > _PagoScreenState > _crearNotificacionUsuario
37 class _PagoScreenState extends State<PagoScreen>
411 Future<String?> _subirComprobanteFirebase() async {
415     String fileName =
416         'comprobantes/${widget.userId}_${DateTime.now().millisecondsSinceEpoch}.jpg';
417     Reference ref = storage.ref().child(fileName);
418     UploadTask uploadTask = ref.putFile(_imagenComprobante!);
419     TaskSnapshot snapshot = await uploadTask;
420     String downloadUrl = await snapshot.ref.getDownloadURL();
421     return downloadUrl;
422 } catch (e) {
423     debugPrint('✘ Error al subir comprobante: $e');
424     return null;
425 }
426 }
427
428 Future<void> _crearNotificacionUsuario(
429     String tipo, String mensaje, String reservaId, String userIdFinal) async {
430     try {
431         if (userIdFinal.isEmpty) {
432             debugPrint('⚠ No se puede crear notificación: userId vacío');
433             return;
434         }
435
436         // Obtener datos del bus - Buscar en ambas colecciones
437         String numeroBus = 'N/A';
438         String fechaSalida = '';
439         String horaSalida = '';
440         String lugarSalida = '';
441
442         // Intentar en colección 'buses_tulcan_salida'
443         var busDoc =
444             await db.collection('buses_tulcan_salida').doc(widget.busId).get();
445
446         // Si no existe, intentar en 'buses_la_esperanza_salida'
```

```
pagos_desde_tulcan.dart x
ib > usuario > compra_de_boletos > De_ida > pagos_desde_tulcan.dart > _PagoScreenState > _crearNotificacionUsuario
37 class _PagoScreenState extends State<PagoScreen>
505
506 Future<void> _procesarPago() async {
507   // Validar método de pago seleccionado
508   if (metodoPagoSeleccionado == null) {
509     ScaffoldMessenger.of(context).showSnackBar(
510       SnackBar(
511         content: const Text('Selecciona un método de pago'),
512         backgroundColor: accentYellow,
513         behavior: SnackBarBehavior.floating,
514         shape:
515           RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),
516         margin: const EdgeInsets.all(16),
517       ), // SnackBar
518     );
519     return;
520   }
521
522   // Validar comprobante si es transferencia
523   if (metodoPagoSeleccionado == 'transferencia' &&
524     _imagenComprobante == null) {
525     ScaffoldMessenger.of(context).showSnackBar(
526       SnackBar(
527         content: const Text('Debes subir el comprobante de pago'),
528         backgroundColor: accentYellow,
529         behavior: SnackBarBehavior.floating,
530         shape:
531           RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),
532         margin: const EdgeInsets.all(16),
533       ), // SnackBar
534     );
535     return;
536   }
537
```

```
pagos_desde_tulcan.dart | seleccion_de_asientos.dart 8 x
b > usuario > compra_de_boletos > De_ida > seleccion_de_asientos.dart > ColoresAsientos
7 // ===== ENUMS Y CONSTANTES =====
8 enum EstadoAsiento {
9   disponible,
10  intentandoReservar,
11  reservado,
12  pagado,
13 }
14
15 class ColoresAsientos {
16   static const Color disponible = Color(0xFFCDD3DD);
17   static const Color intentandoReservar = Color(0xFFFF4A259);
18   static const Color reservado = Color(0xFF1B4965);
19   static const Color pagado = Color(0xFF4CAF50);
20   static const Color seleccionado = Color(0xFF940016);
21 }
22
23 class LimitesReserva {
24   static const int LIMITE DIARIO TOTAL =
25     | 4; // Límite combinado de reservas + compras
26   static const int MAX_RESERVAS ACTIVAS =
27     | 4; // Mantener 1 reserva activa a la vez
28 }
29
30 // ===== PANTALLA PRINCIPAL DE ASIENTOS =====
31 class AsientosScreen extends StatefulWidget {
32   final String busId;
33   final String paradaNombre;
34   final double paradaPrecio;
35   final String userId;
36
37   const AsientosScreen({
38     Key? key,
39     required this.busId,
40     required this.paradaNombre,
```

```
seleccion_de_asientos.dart 8 x
lib > usuario > compra_de_boletos > De_ida > seleccion_de_asientos.dart > ColoresAsientos
49 class _AsientosScreenState extends State<AsientosScreen>
110
117 Future<void> _inicializar() async {
118   await _limpiarReservasAbandonadas();
119   _iniciarHeartbeat();
120   _iniciarLimpiezaPeriodica();
121   _escucharCambiosReserva();
122 }
123
124 @override
125 void dispose() {
126   _pantallaActiva = false;
127   _fadeController.dispose();
128   _limpiarRecursos();
129   super.dispose();
130 }
131
132 void _limpiarRecursos() {
133   _heartbeatTimer?.cancel();
134   _limpiezaTimer?.cancel();
135   _reservaSubscription?.cancel();
136 }
137
138 // ===== LIMPIEZA PERIÓDICA AUTOMÁTICA =====
139 void _iniciarLimpiezaPeriodica() {
140   _limpiezaTimer =
141     Timer.periodic(Duration(seconds: LIMPIEZA_INTERVAL), (timer) {
142       if (_pantallaActiva) {
143         _limpiarAsientosPorTimeout();
144       }
145     }); // Timer.periodic
146 }
147
148 Future<void> _limpiarAsientosPorTimeout() async {
149   await

```

```
seleccion_de_asientos.dart 8  Seleccion_de_parada_desde_Tulcan.dart x
b > usuario > compra_de_boletos > De_jda > Seleccion_de_parada_desde_Tulcan.dart > _ParadasScreenState
5  class ParadasScreen extends StatefulWidget {
10  ParadasScreenState createState() => ParadasScreenState();
11  }
12
13  class ParadasScreenState extends State<ParadasScreen>
14  | with SingleTickerProviderStateMixin {
15  |   late AnimationController _fadeController;
16  |   late Animation<double> _fadeAnimation;
17  |   String _searchQuery = '';
18  |
19  |   // Paleta de colores inspirada en transporte de buses
20  |   final Color primaryBusBlue = Color.fromARGB(255, 243, 248, 255);
21  |   final Color accentOrange = Color(0xFFEA580C);
22  |   final Color darkNavy = Color(0xFF0F172A);
23  |   final Color roadGray = Color(0xFF334155);
24  |   final Color lightBg = Color(0xFFFF1F5F9);
25  |   final Color textGray = Color(0xFF475569);
26  |   final Color successGreen = Color(0xFF059669);
27  |   final Color accentBlue = Color(0xFF1E40AF);
28  |
29  |   @override
30  |   void initState() {
31  |     super.initState();
32  |     _fadeController = AnimationController(
33  |       vsync: this,
34  |       duration: const Duration(milliseconds: 800),
35  |     ); // AnimationController
36  |     _fadeAnimation = Tween<double>(begin: 0.0, end: 1.0).animate(
37  |       CurvedAnimation(
38  |         parent: _fadeController,
39  |         curve: Curves.easeOut,
40  |       ),
41  |     );
42  |     _fadeController.forward();
```

```

> usuario > Encomiendas > datos_encomienda.dart > _EnvioScreenState > accentOrange
124 class _EnvioScreenState extends State<EnvioScreen> {
161   Future<void> _verificarUsuario() async {
166     },
167   }
168 }
169
170 Future<void> _cargarTiposEncomienda() async {
171   setState(() => _isLoading = true);
172
173   try {
174     final tiposSnapshot = await Firestore.instance
175       .collection('Tipo_encomienda')
176       .where('activo', isEqualTo: true)
177       .get();
178
179     _tiposEncomienda = tiposSnapshot.docs.map((doc) {
180       final data = doc.data();
181       return {
182         'id': doc.id,
183         'tipo': data['Tipo'] ?? data['tipo'] ?? 'Sin nombre',
184         'descripcion': data['descripcion'] ?? '',
185         'precio_adicional': 0.0,
186       };
187     }).toList();
188
189     setState(() => _isLoading = false);
190   } catch (e) {
191     setState(() => _isLoading = false);
192     _mostrarError('Error al cargar datos: $e');
193   }
194 }
195
196 Future<void> _seleccionarImagen() async {
197   try {

```

```
datos_del_destinatario.dart 1 | datos_encomienda.dart 3 x
o > usuario > Encomiendas > datos_encomienda.dart > _EnvioScreenState > accentOrange
124 class _EnvioScreenState extends State<EnvioScreen> {
125
126   Future<void> _seleccionarImagen() async {
127     try {
128       final ImageSource? source = await showModalBottomSheet<ImageSource>(
129         context: context,
130         shape: const RoundedRectangleBorder(
131           borderRadius: BorderRadius.vertical(top: Radius.circular(20)),
132         ), // RoundedRectangleBorder
133         builder: (context) => Container(
134           padding: const EdgeInsets.all(20),
135           child: Column(
136             mainAxisAlignment: MainAxisAlignment.min,
137             children: [
138               Container(
139                 width: 40,
140                 height: 4,
141                 margin: const EdgeInsets.only(bottom: 20),
142                 decoration: BoxDecoration(
143                   color: Colors.grey[300],
144                   borderRadius: BorderRadius.circular(2),
145                 ), // BoxDecoration
146               ), // Container
147               const Text(
148                 'Seleccionar foto del paquete',
149                 style: TextStyle(
150                   fontSize: 18,
151                   fontWeight: FontWeight.w700,
152                   color: darkNavy,
153                 ), // TextStyle
154               ), // Text
155               const SizedBox(height: 20),
156               ListTile(
157                 leading: Container(
```

```

datos_remitente.dart 2 x
b > usuario > Encomiendas > datos_remitente.dart > _RemitenteScreenState > primaryBusBlue
17 class _RemitenteScreenState extends State<RemitenteScreen> {
62   Future<void> _cargarDatosUsuario() async {
69     // Intentar cargar datos adicionales desde Firestore
70     final userDoc = await Firestore.instance
71       .collection('usuarios_registrados')
72       .doc(user.uid)
73       .get();
74
75     if (userDoc.exists) {
76       final data = userDoc.data();
77       if (data != null) {
78         _nombreController.text = data['nombre'] ?? '';
79         _cedulaController.text = data['cedula'] ?? '';
80         _telefonoController.text = data['telefono'] ?? '';
81         _lugarSalidaController.text = data['direccion'] ?? '';
82       }
83     }
84   }
85   } catch (e) {
86     print('Error al cargar datos del usuario: $e');
87   } finally {
88     setState(() {
89       _isLoadingUserData = false;
90     });
91   }
92 }
93
94 String _generarCodigoEnvio() {
95   final random = Random();
96   final numero = random.nextInt(999999).toString().padLeft(6, '0');
97   return 'ENC-$numero';
98 }
99
100 bool validarFormulario() {

```

```

datos_remitente.dart 2 x
b > usuario > Encomiendas > datos_remitente.dart > _RemitenteScreenState > primaryBusBlue
17 class _RemitenteScreenState extends State<RemitenteScreen> {
99
00   bool _validarFormulario() {
01     if (_nombreController.text.trim().isEmpty) {
02       _mostrarError('Ingrese el nombre completo');
03       return false;
04     }
05     if (_cedulaController.text.trim().isEmpty) {
06       _mostrarError('Ingrese la cédula');
07       return false;
08     }
09     if (_cedulaController.text.trim().length != 10) {
10       _mostrarError('La cédula debe tener 10 dígitos');
11       return false;
12     }
13     if (_telefonoController.text.trim().isEmpty) {
14       _mostrarError('Ingrese el teléfono');
15       return false;
16     }
17     if (_telefonoController.text.trim().length != 10) {
18       _mostrarError('El teléfono debe tener 10 dígitos');
19       return false;
20     }
21     if (_correoController.text.trim().isEmpty) {
22       _mostrarError('Ingrese el correo electrónico');
23       return false;
24     }
25     if (!_correoController.text.contains('@')) {
26       _mostrarError('Ingrese un correo válido');
27       return false;
28     }
29     if (_lugarSalidaController.text.trim().isEmpty) {
30       _mostrarError('Ingrese el lugar de salida');
31       return false;
32     }
33   }

```

```
datos_del_destinatario.dart 1 x
b > usuario > Encomiendas > datos_del_destinatario.dart > _DestinatarioScreenState
14 class _DestinatarioScreenState extends State<DestinatarioScreen> {
51 }
52
53 bool _validarFormulario() {
54   if (_nombreController.text.trim().isEmpty) {
55     _mostrarError('Ingrese el nombre completo del destinatario');
56     return false;
57   }
58   if (_cedulaController.text.trim().isEmpty) {
59     _mostrarError('Ingrese la cédula del destinatario');
60     return false;
61   }
62   if (_cedulaController.text.trim().length != 10) {
63     _mostrarError('La cédula debe tener 10 dígitos');
64     return false;
65   }
66   if (_telefonoController.text.trim().isEmpty) {
67     _mostrarError('Ingrese el teléfono del destinatario');
68     return false;
69   }
70   if (_telefonoController.text.trim().length != 10) {
71     _mostrarError('El teléfono debe tener 10 dígitos');
72     return false;
73   }
74   if (_ciudadController.text.trim().isEmpty) {
75     _mostrarError('Ingrese la destino');
76     return false;
77   }
78   if (_direccionController.text.trim().isEmpty) {
79     _mostrarError('Ingrese la dirección de entrega');
80     return false;
81   }
82   return true;
83 }
```

```
datos_del_destinatario.dart 1 x
b > usuario > Encomiendas > datos_del_destinatario.dart > _DestinatarioScreenState
14 class _DestinatarioScreenState extends State<DestinatarioScreen> {
172   Widget _buildHeader(BuildContext context) {
261     ],
262     ), // Row
263     const SizedBox(height: 28),
264     const Text(
265       'Datos del Destinatario',
266       style: TextStyle(
267         fontSize: 30,
268         fontWeight: FontWeight.w900,
269         color: Color.fromARGB(255, 32, 31, 31),
270         height: 1.1,
271         letterSpacing: -0.5,
272       ), // TextStyle
273     ), // Text
274     const SizedBox(height: 8),
275     const Text(
276       'Información completa de quien recibirá el paquete',
277       style: TextStyle(
278         fontSize: 14,
279         color: Color.fromARGB(255, 84, 86, 88),
280         fontWeight: FontWeight.w500,
281         height: 1.4,
282       ), // TextStyle
283     ), // Text
284     const SizedBox(height: 20),
285     Row(
286       children: [
287         _buildStatChip(Icons.location_on_rounded, 'Destino'),
288         const SizedBox(width: 10),
289         _buildStatChip(Icons.verified_rounded, 'Verificado'),
290       ],
291     ), // Row

```

```
datos_remitente.dart 2  rastreo.dart x
lib > usuario > Encomiendas > rastreo.dart > _RastreoEncomiendaScreenState > _buildInfoCard
12  class _RastreoEncomiendaScreenState extends State<RastreoEncomiendaScreen> {
934  Widget _buildEntregaCard(Map<String, dynamic> entrega) {
1009      child: Column(
1010          children: [
1011              buildInfoRow(
1012                  'Fecha de Entrega',
1013                  _formatTimestamp(entrega['fecha_entrega']),
1014                  isBold: true,
1015              ),
1016              if (entrega['recibido_por'] != null)
1017                  _buildInfoRow('Recibido por', entrega['recibido_por']),
1018              if (entrega['cedula_receptor'] != null)
1019                  buildInfoRow(
1020                      'Cédula Receptor', entrega['cedula_receptor']),
1021              if (entrega['parentesco'] != null)
1022                  _buildInfoRow('Parentesco', entrega['parentesco']),
1023              if (entrega['observaciones'] != null &&
1024                  entrega['observaciones'].toString().isNotEmpty)
1025                  _buildInfoRow('Observaciones', entrega['observaciones']),
1026              if (entrega['entregado_por'] != null)
1027                  _buildInfoRow('Entregado por', entrega['entregado_por']),
1028          ],
1029      ), // Column
1030  ), // Container
1031  ],
1032  ), // Column
1033  ), // Padding
1034  ); // Container
1035 }
```

```

datos_remitente.dart 2  rastreo.dart x
b > usuario > Encomiendas > rastreo.dart > _RastreoEncomiendaScreenState > _buildInfoCard
12  class _RastreoEncomiendaScreenState extends State<RastreoEncomiendaScreen> {
1334  Widget _buildCostosCard(
1387      Container(
1388          padding: const EdgeInsets.all(14),
1389          decoration: BoxDecoration(
1390              color: Colors.white,
1391              borderRadius: BorderRadius.circular(12),
1392              border: Border.all(color: Colors.grey.shade200),
1393          ), // BoxDecoration
1394          child: Column(
1395              children: [
1396                  _buildInfoRow(
1397                      'Peso',
1398                      '${envio['rango_peso'] ?? 0} kg',
1399                  ),
1400                  _buildInfoRow(
1401                      'Precio encomienda',
1402                      '\${costos['precio_tipo'] ?? 0}',
1403                  ),
1404                  const Padding(
1405                      padding: EdgeInsets.symmetric(vertical: 8),
1406                      child: Divider(thickness: 1),
1407                  ), // Padding
1408                  _buildInfoRow(
1409                      'TOTAL',
1410                      '\${costos['total'] ?? 0}',
1411                      isBold: true,
1412                  ),
1413              ],
1414          ), // Column
1415      ), // Container
1416  ],

```

```
informacion.dart x
lib > usuario > Pantallas_inicio > informacion.dart > _TransDoramaldHomeScreenState
15 class _TransDoramaldHomeScreenState extends State<TransDoramaldHomeScreen>
58 Future<void> _loadImagesFromFirebase() async {
59   if (!mounted) return;
60
61   try {
62     setState(() => isLoadingImages = true);
63
64     final storageRef = FirebaseStorage.instance.ref().child('informacion');
65     final listResult = await storageRef.listAll();
66
67     List<String> urls = [];
68     for (var item in listResult.items) {
69       final ext = item.name.toLowerCase();
70       if (ext.endsWith('.jpg') ||
71           ext.endsWith('.jpeg') ||
72           ext.endsWith('.png') ||
73           ext.endsWith('.webp')) {
74         final url = await item.getDownloadURL();
75         urls.add(url);
76       }
77     }
78
79     if (!mounted) return;
80     setState(() {
81       imgList = urls;
82       isLoadingImages = false;
83     });
84   } catch (e) {
85     debugPrint('Error al cargar imágenes: $e');
86     if (!mounted) return;
87     setState(() => isLoadingImages = false);
88   }
89 }
90
```

```
informacion.dart x iniciarsesion.dart 1
ib > usuario > Pantallas_inicio > informacion.dart > _TransDoramaldHomeScreenState
15 class _TransDoramaldHomeScreenState extends State<TransDoramaldHomeScreen
127 Widget _buildHeroCarousel() {
179   BoxDecoration(
180     borderRadius: BorderRadius.circular(20),
181   ), // BoxDecoration
182   child: const Text(
183     'Trans Doramald 2026',
184     style: TextStyle(
185       color: Colors.white,
186       fontSize: 13,
187       fontWeight: FontWeight.w700,
188       letterSpacing: 1.5,
189     ), // TextStyle
190   ), // Text
191   const SizedBox(height: 12),
192   const Text(
193     'Liderando el Transporte Rural ',
194     style: TextStyle(
195       color: Colors.white,
196       fontSize: 20,
197       fontWeight: FontWeight.w900,
198       height: 1.1,
199       letterSpacing: -1,
200     ), // TextStyle
201   ), // Text
202   const SizedBox(height: 12),
203   Row(
204     children: [
205       buildCarouselBadge(
206         Icons.shield_rounded,
207         'Seguridad Garantizada',
208       ),
209     ],
210   ), // Row
211 }
```

```
informacion.dart x iniciarsesion.dart 1 x
lib > usuario > Pantallas_inicio > iniciarsesion.dart > _HomePageState > _saveFCMTokenAfterLogin
18 class _HomePageState extends State<HomePage> {
158 Future<void> _saveFCMTokenAfterLogin() async {
169     await _firestore.collection('usuarios_registrados').doc(user.uid).set({
170         'userId': user.uid, // Agregado userId
171         'email': user.email,
172         'fcmToken': token,
173         'lastTokenUpdate': FieldValue.serverTimestamp(),
174     }, SetOptions(merge: true));
175
176     print('✅ FCM Token guardado correctamente');
177     print('📄 Token: ${token.substring(0, 30)}...');
178     print('👤 UserId: ${user.uid}'); // Log del userId
179 } else {
180     print('⚠️ No se pudo obtener el FCM token');
181 }
182
183 FirebaseMessaging.instance.onTokenRefresh.listen((newToken) {
184     _firestore.collection('usuarios_registrados').doc(user.uid).update({
185         'fcmToken': newToken,
186         'lastTokenUpdate': FieldValue.serverTimestamp(),
187     });
188     print('🔄 FCM Token actualizado');
189 });
190 } catch (e) {
191     print('❌ Error al guardar FCM token: $e');
192 }
193 }
194
195 @override
196 Widget build(BuildContext context) {
197     return LayoutBuilder(
198         builder: (context, constraints) {
199             final isSmallScreen = constraints.maxWidth < 600;
200             final isTablet =
```

```
informacion.dart  iniciarsesion.dart 1 x
lib > usuario > Pantallas_inicio > iniciarsesion.dart > _HomePageState > _saveFCMTokenAfterLogin
18  class _HomePageState extends State<HomePage> {
449  Widget _buildBottomNavigationBar(
507      // MENÚ
508      BottomNavigationBarItem(
509          icon: Container(
510              padding: EdgeInsets.all(isSmall ? 6 : 8),
511              decoration: BoxDecoration(
512                  color: _currentPageIndex == 1
513                      ? primaryNavy.withOpacity(0.1)
514                      : Colors.transparent,
515                  borderRadius: BorderRadius.circular(10),
516              ), // BoxDecoration
517              child: Image.asset(
518                  'assets/menu.png',
519                  width: iconSize,
520                  height: iconSize,
521                  color: _currentPageIndex == 1 ? primaryNavy : mediumGray,
522              ), // Image.asset
523          ), // Container
524          label: 'Menú',
525      ), // BottomNavigationBarItem
526
527      // 📢 NOTIFICACIONES CON BADGE
528      BottomNavigationBarItem(
529          icon: StreamBuilder<int>(
530              stream: _unreadNotificationsStream,
531              initialData: 0,
532              builder: (context, snapshot) {
533                  final count = snapshot.data ?? 0;
534
535                  return _buildNotificationBadge(
536                      count: count,
537                      isSelected: _currentPageIndex == 2,
538                      child: Container(
```

```
informacion.dart  menu.dart 3, M x  iniciarsesion.dart 1
b > usuario > Pantallas_inicio > menu.dart > ...
29 class _MenuCuadrosScreenState extends State<MenuCuadrosScreen>
133 List<Map<String, dynamic>> getServicios() {
138   return [
139     {
140       'titulo': 'Gestionar Encomiendas',
141       'subtitulo': 'Peso y Tipo',
142       'descripcion': 'Administra tus paquetes y precios',
143       'ruta': ConfiguracionEncomiendas(),
144       'icon': Icons.local_shipping,
145       'color': const Color.fromARGB(255, 56, 5, 150),
146     },
147     {
148       'titulo': 'Encomiendas',
149       'subtitulo': 'Envíos seguros',
150       'descripcion': 'Administra tus paquetes y envíos',
151       'ruta': EncomiendasScreen(),
152       'icon': Icons.local_shipping,
153       'color': const Color(0xFF059669),
154     },
155   ];
156 }
157 List<Map<String, dynamic>> serviciosBase = [
158   {
159     'titulo': 'Comprar Boleto',
160     'subtitulo': 'Viaja con nosotros',
161     'descripcion': 'Reserva tu asiento de manera rápida y segura',
162     'ruta': MenuOpcionesScreen(),
163     'icon': Icons.confirmation_number,
164     'color': const Color(0xFF1E40AF),
165   },
166   {
167     'titulo': 'Mis Viajes',
168     'subtitulo': 'Historial completo',
169     'descripcion': 'Revisa tus boletos y transacciones anteriores',
170     'ruta': HistorialComprasScreen()
```

```
menu.dart 3, M x
lib > usuario > Pantallas_inicio > menu.dart > ...
29 class _MenuCuadrosScreenState extends State<MenuCuadrosScreen>
291 widget buildMensajeIniciarSesion() {
305   child: SafeArea(
306     child: Center(
307       child: SingleChildScrollView(
308         padding: const EdgeInsets.all(24),
309         child: FadeTransition(
310           opacity: _fadeAnimation,
311           child: Column(
312             mainAxisAlignment: MainAxisAlignment.center,
313             children: [
314               Container(
315                 padding: const EdgeInsets.all(24),
316                 decoration: BoxDecoration(
317                   color: Colors.white.withOpacity(0.15),
318                   shape: BoxShape.circle,
319                 ), // BoxDecoration
320                 child: Container(
321                   padding: const EdgeInsets.all(20),
322                   decoration: const BoxDecoration(
323                     color: const Color(0xFF940016),
324                     shape: BoxShape.circle,
325                   ), // BoxDecoration
326                   child: Image(
327                     image: AssetImage('assets/icon2.png'),
328                     width: 80,
329                   ), // Image
330                 ), // Container
331               ), // Container
332               const SizedBox(height: 20),
333               const Text(
334                 'TRANS DORAMALD',
335                 style: TextStyle(
336                   fontSize: 28,
```

```
menu.dart 3, M  historial.dart 1 x
ib > usuario > historial.dart > _HistorialComprasScreenState > _buildFiltroFechas
17  class _HistorialComprasScreenState extends State<HistorialComprasScreen>
82  Widget _buildModernAppBar() {
136      crossAxisAlignment: CrossAxisAlignment.start,
137      children: [
138          Text(
139              'HISTORIAL DE BOLETOS',
140              style: TextStyle(
141                  fontSize: 14,
142                  fontWeight: FontWeight.w900,
143                  color: textDark,
144                  letterSpacing: 1,
145              ), // TextStyle
146          ), // Text
147          Text(
148              'Gestión de viajes',
149              style: TextStyle(
150                  fontSize: 10,
151                  fontWeight: FontWeight.w500,
152                  color: textGray,
153                  letterSpacing: 0.5,
154              ), // TextStyle
155          ), // Text
156      ],
157  ), // Column
158  ],
159  ), // Row
160  InkWell(
161      borderRadius: BorderRadius.circular(12),
162      onTap: () {},
163      child: Container(
164          padding: const EdgeInsets.all(10),
165          decoration: BoxDecoration(
166              color:  const Color.fromARGB(0, 255, 255, 255),
167              borderRadius: BorderRadius.circular(12),
```

Anexo 4. Manual De Usuario Del Sistema Trans Doramald

1. INTRODUCCIÓN

1.1. Propósito

El presente manual tiene como objetivo proporcionar las instrucciones detalladas para la instalación, configuración y operación del sistema móvil "Trans Doramald", desarrollado para la gestión integral de talento humano, venta de boletos y envío de encomiendas de la compañía de transporte público Trans Doramald, operante en las parroquias de Maldonado y El Chical, Cantón Tulcán.

1.2. Audiencia Objetivo

Este manual está dirigido a:

- **Usuarios pasajeros:** Ciudadanos que utilizan el servicio de transporte y necesitan comprar boletos o enviar encomiendas. **Conductores:** Operadores de las unidades de transporte que gestionan encomiendas en ruta.
- **Administradores:** Personal de la compañía encargado de la gestión de rutas, buses, pagos y talento humano.

1.3. Alcance del Sistema

La aplicación contempla tres perfiles de acceso diferenciados:

- **Módulo Cliente:** Reserva de boletos, seguimiento de encomiendas, historial de viajes y notificaciones.
- **Módulo Conductor:** Gestión de encomiendas asignadas y confirmación de entregas.
- **Módulo Administrador:** Control total del sistema incluyendo gestión de rutas, validación de pagos, administración de flota y personal.

2. ESPECIFICACIONES TÉCNICAS Y REQUISITOS DEL SISTEMA

2.1. Requisitos de Hardware

- **Dispositivo:** Smartphone o tablet Android (versión 5.0 o superior)
- **Conectividad:** Wi-Fi o datos móviles (3G/4G/5G)
- **Almacenamiento:** Mínimo 150 MB de espacio libre

- **Cámara:** Requerida para escaneo de códigos de tracking (módulo conductor), fotos de transacciones bancarias y paquetes.

2.2. Requisitos de Software

- **Sistema Operativo:** Android 5.0+
- **Permisos necesarios:**
 - Acceso a internet
 - Notificaciones push
 - Almacenamiento (para descarga de comprobantes PDF)
 - Cámara (para carga de comprobantes de pago)

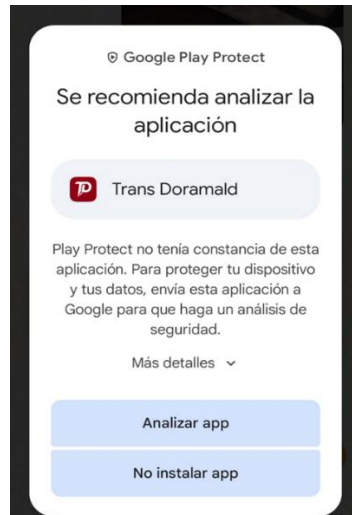
3. INSTALACIÓN Y CONFIGURACIÓN INICIAL

3.1. Descarga e Instalación

La aplicación se distribuye mediante archivo APK (Android) o mediante instalador oficial. Dado que la aplicación está destinada a zonas rurales con limitada conectividad, se permite la instalación offline.

Procedimiento:

1. Descargue el archivo TransDoramald_vX.X.apk desde el enlace proporcionado por la compañía.
2. Abra el archivo descargado en su dispositivo Android.
3. Si aparece una advertencia de seguridad, seleccione "Permitir instalación de fuentes desconocidas" en la configuración de seguridad.
4. El sistema Google Play Protect realizará un escaneo de seguridad (*Ver Figura 23 - Alerta de seguridad*).
5. Una vez verificada la integridad, pulse "Instalar".
6. Espere la finalización del proceso y presione "Abrir".



3.2. Primer Inicio y Permisos

Al iniciar la aplicación por primera vez:

1. Se presentará la pantalla de información institucional (Ver Figura 24 - Panel de información).
2. Deslice hacia la izquierda para conocer la Misión y Visión de la compañía (Ver Figura 25).
3. Acceda a la sección de Contactos para verificar los datos de comunicación (Ver Figura 26).
4. El sistema solicitará permisos de notificaciones; recomendamos aceptar para recibir alertas de sus viajes.

4. MÓDULO DE USUARIO (PASAJEROS)

4.1. Acceso al Sistema

El sistema ofrece dos métodos de autenticación según el requerimiento funcional RF002:

4.1.1. Opción A: Acceso con Google (Recomendado)

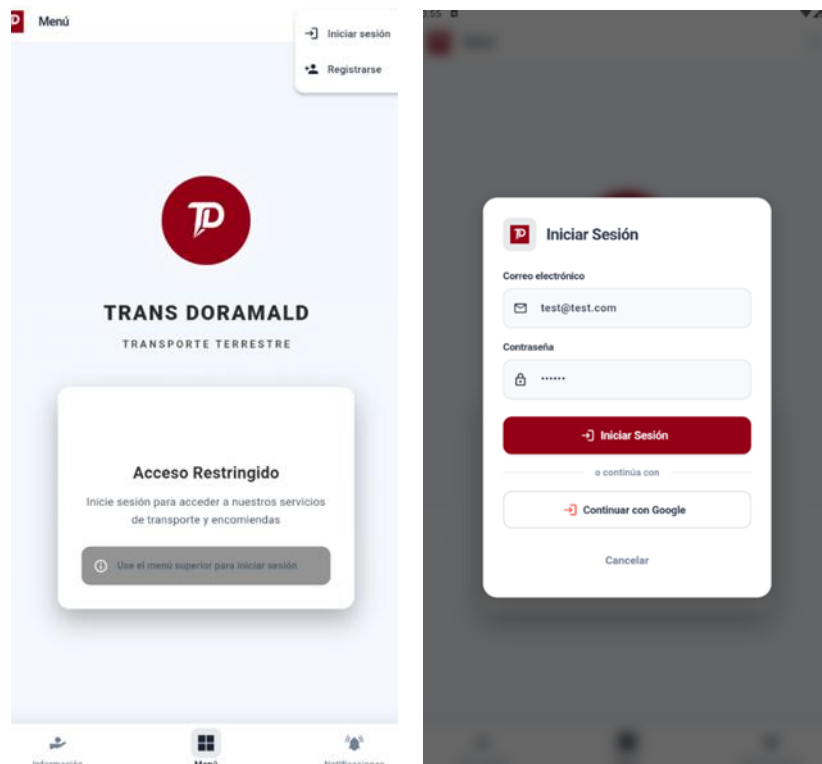
1. En la pantalla de Menú de Acceso (Ver Figura 27), seleccione "Acceder con Google".
2. Elija la cuenta de Google que desea vincular (Ver Figura 29 - Panel de selección de cuenta).

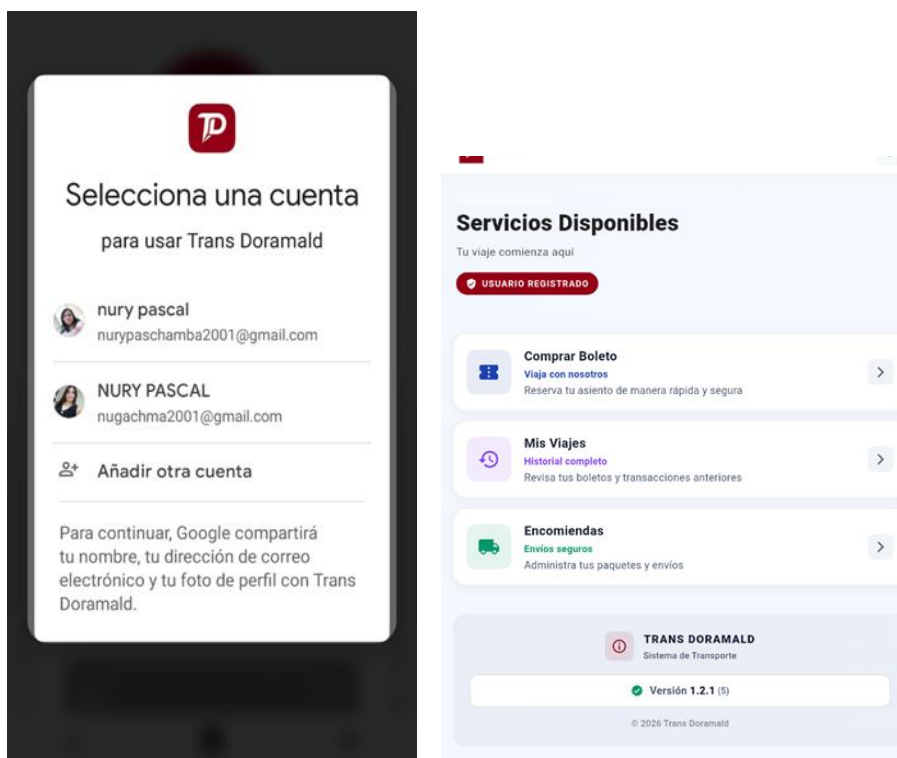
3. El sistema validará automáticamente sus credenciales y creará un perfil vinculado a su correo.
4. Será redirigido al Panel de Servicios (Ver Figura 30).

4.1.2. Opción B: Acceso con Correo y Contraseña

1. Seleccione "Acceder con Correo" en el Menú de Acceso.
 2. Si ya tiene cuenta, ingrese su email y contraseña en el Panel de Inicio de Sesión (Ver Figura 28).
-
3. Si es nuevo usuario, seleccione "Crear cuenta" y complete:
 - Nombres completos
 - Número de cédula de identidad
 - Correo electrónico válido
 - Contraseña segura (mínimo 6 caracteres)

Nota importante: Para usuarios en zonas rurales sin acceso regular a internet, se recomienda el acceso con Google para evitar el olvido de contraseñas.





4.2. Compra de Boletos (RF003)

4.2.1. Selección de Ruta y Horario

1. Desde el Panel de Servicios, pulse "Comprar Boleto".
2. Seleccione su ruta preferida entre las opciones disponibles (ej: Tulcán - La Esperanza) (Ver Figura 31 - Panel de compra de boletos).
3. Elija la unidad (bus) disponible según horario y asientos libres (Ver Figura 32 - Panel de buses disponibles).
 - Observe el número de asientos disponibles en tiempo real.
 - Verifique el horario de salida programado.
4. Seleccione su parada de origen y destino desde el listado de paradas intermedias (Ver Figura 33).
 - El sistema mostrará paradas como: Maldonado, El Chical, Tulcán Centro, etc.
 - Se calcula automáticamente la tarifa según distancia.

4.2.2. Selección de Asiento

1. Visualice el mapa de asientos de la unidad (Ver Figura 34).

- **Color Rojo:** Asiento disponible
 - **Color Verde:** Asiento ocupado/bloqueado
2. Pulse sobre el asiento deseado (ventana o pasillo).
 3. El sistema bloqueará temporalmente el asiento durante 10 minutos mientras completa el proceso (RNF006 - Integridad).

4.2.3. Datos del Pasajero y Pago

1. Ingrese los datos del pasajero (*Ver Figura 35*):
 - Número de cédula (validación formato Ecuador)
 - Nombres completos
 - Teléfono de contacto
2. Seleccione método de pago (*Ver Figura 36*):
 - **Efectivo:** Genere un código QR o número de referencia para pagar en puntos físicos de la compañía.
 - **Transferencia:** Visualice los datos bancarios de la compañía y adjunte el comprobante de pago (fotografía del voucher).
3. Para transferencias, el sistema validará el pago en un máximo de 30 minutos (RF011).

4.2.4. Confirmación y Recibo

1. Una vez validado el pago, recibirá su comprobante digital en formato PDF/PNG.
2. Acceda al "Historial de Boletos" (*Ver Figura 37*) para:
 - Visualizar viajes pasados y futuros
 - Descargar nuevamente sus recibos
 - Cancelar reservas (sujeto a políticas de la compañía)

Selecciona tu Destino

Contamos con rutas directas y frecuentes para tu comodidad

Rápido Seguro

Tulcán - La Esperanza
Ruta de ida

Viaje directo con múltiples horarios disponibles

Características:

- Horarios frecuentes
- Servicio directo
- Asientos cómodos

La Esperanza - Tulcán
Ruta de regreso

Ruta de retorno con servicio continuo y seguro

Características:

- Salidas puntuales
- Viaje confortable
- Conexión garantizada

¿Cómo comprar?

- 1 **Selecciona tu ruta**
Elige entre ida o regreso

SELECCIÓN DE BUS

Elige tu viaje

Buses Disponibles

Selecciona el bus que mejor se adapte a tu horario

Bus 04
CAA-153 Disponible

Disponibilidad **44/44 asientos**

Información del viaje

Chofer: **Fredy Sarchi** Hora: **13:20**

Salida: **La Esperanza** Fecha: **29/01/2026**

SELECCIÓN DE PARADA

Elige tu parada

Selecciona tu Parada

Elige el punto de destino más conveniente para ti

Buscar parada por nombre...

Tulcán
Parada disponible

\$ 4.5 USD Activa

SELECCIÓN DE ASIENTO

Elige tu asiento

Selecciona tu Asiento

Toca un asiento disponible para reservarlo

Tulcán
Destino seleccionado

\$ 4.50 USD por asiento

Máximo 1 reserva activa - 4 boletos por día - Válido por 5 minutos

Leyenda de Estados

Disponible En proceso Reservado Pagado

Conductor

1	2	3	4
5	6	7	8
9	10	11	12

Asiento seleccionado **Número 1**

Total a pagar **\$4.50** Continuar →

DATOS DEL PASAJERO
Completa tu información

Información del Pasajero

Completa tus datos personales para generar tu boleto

Tulcán
Asiento 1

\$ 4.50 USD

Datos Personales
Ingresa tu información para el boleto

Pais de Origen
Ecuador

Nombre Completo
Ingresa tu nombre completo

Cédula de Identidad
0000000000

Número de Celular
0900000000

Total a pagar
\$4.50

Continuar al Pago →

MÉTODO DE PAGO
Finaliza tu compra

Método de Pago

Selecciona cómo deseas realizar el pago de tu boleto

Resumen de Compra

Asiento: #1
Parada: Tulcán
Pasajero: Juan Rito

Total a Pagar \$4.50

Selecciona el método de pago

Transferencia Bancaria
Sube tu comprobante para verificación

Pago en Efectivo
Paga en ventanilla - Requiere confirmación

Confirmar Pago

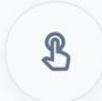
HISTORIAL DE BOLETOS
Gestiona tus viajes

Mis Boletos

Revisa el historial completo de tus viajes

Seleccionar Fecha
0 fechas disponibles

No hay fechas disponibles



Selecciona una Fecha
Elige una fecha de la lista superior para visualizar tus boletos

4.3. Envío de Encomiendas (RF004)

4.3.1. Registro de Nueva Encomienda

1. Desde el Panel de Servicios, seleccione "Encomiendas" (Ver Figura 38).
2. Pulse "Nueva Encomienda".

3. Complete los datos en tres pasos:

Paso 1 - Datos del Remitente (Ver Figura 39):

- Nombres completos
- Teléfono de contacto
- Dirección completa de origen

Paso 2 - Datos del Destinatario (Ver Figura 40):

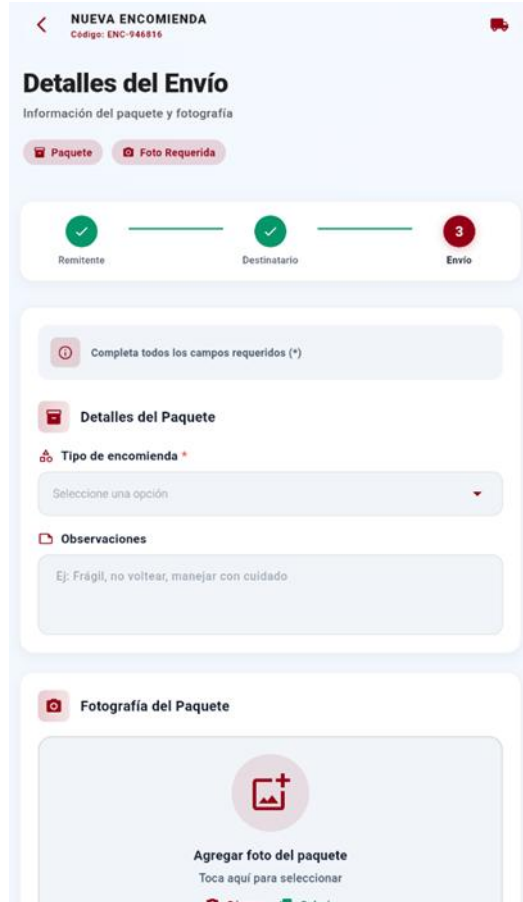
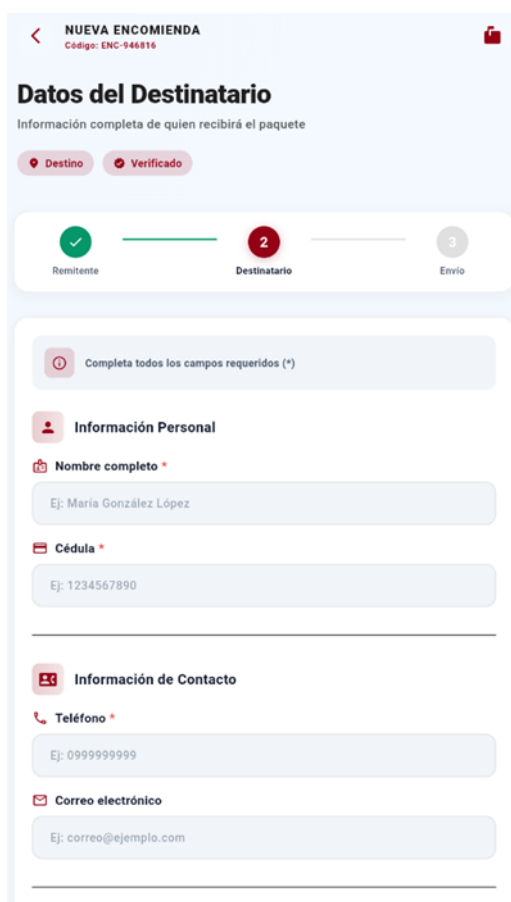
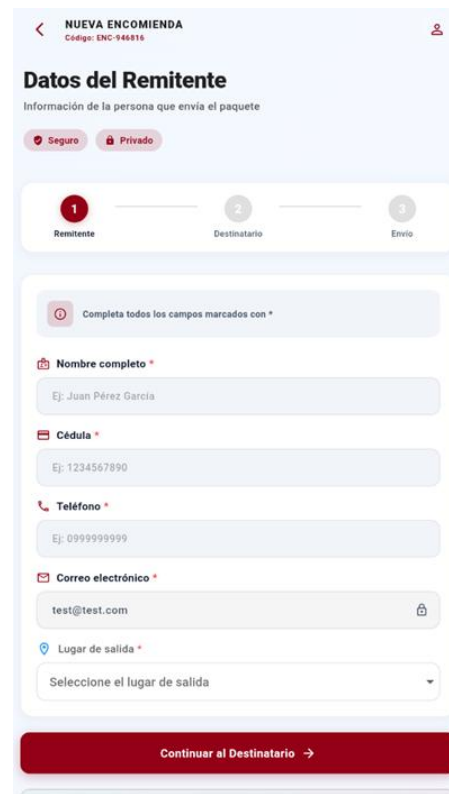
- Nombres completos
- Número de cédula (para verificación en entrega)
- Dirección exacta de destino (incluya referencias geográficas locales)

Paso 3 - Detalles del Paquete (Ver Figura 41):

- Descripción detallada del contenido
- Tipo de encomienda (Documentos, Frágil, Perecedero, Otros)
- Peso aproximado
- Valor declarado (para seguro)
- Observaciones especiales (ej: "No exponer al sol", "Frágil")

4.3.2. Pago y Seguimiento

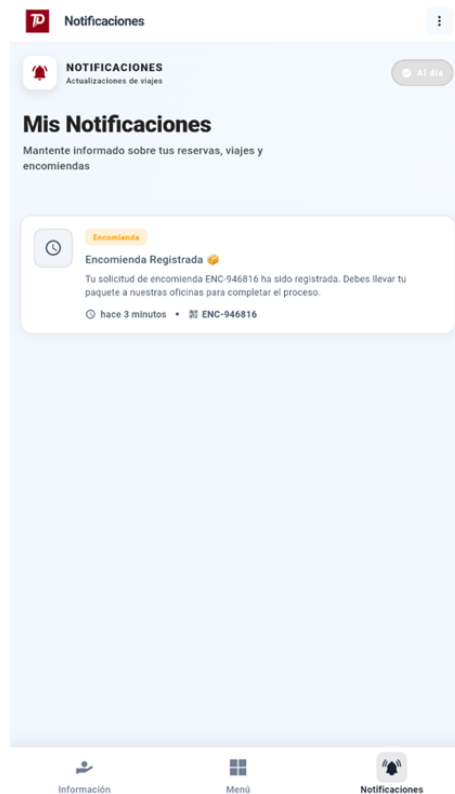
1. Realice el pago según tarifa calculada (transferencia o efectivo).
2. Obtenga su código de tracking único (Ver Figura 42 - Ventana de solicitud registrada).
3. Guarde el comprobante digital o físico.
4. Rastree su encomienda desde el mismo módulo, visualizando estados: "Registrado", "En tránsito", "Entregado".





4.4. Notificaciones y Comunicación

1. Acceda al "Panel de Notificaciones" desde el menú inferior (Ver Figura 43).
2. Visualice alertas sobre:
 - Confirmación de reservas
 - Aprobación de pagos
 - Cambios de horario o ruta (crítico en temporada lluviosa)
 - Recordatorios de viaje (2 horas antes de la salida)
3. Las notificaciones funcionan en modo local (sin internet) y se sincronizan cuando hay conectividad (RNF005).



5. MÓDULO DE CONDUCTOR

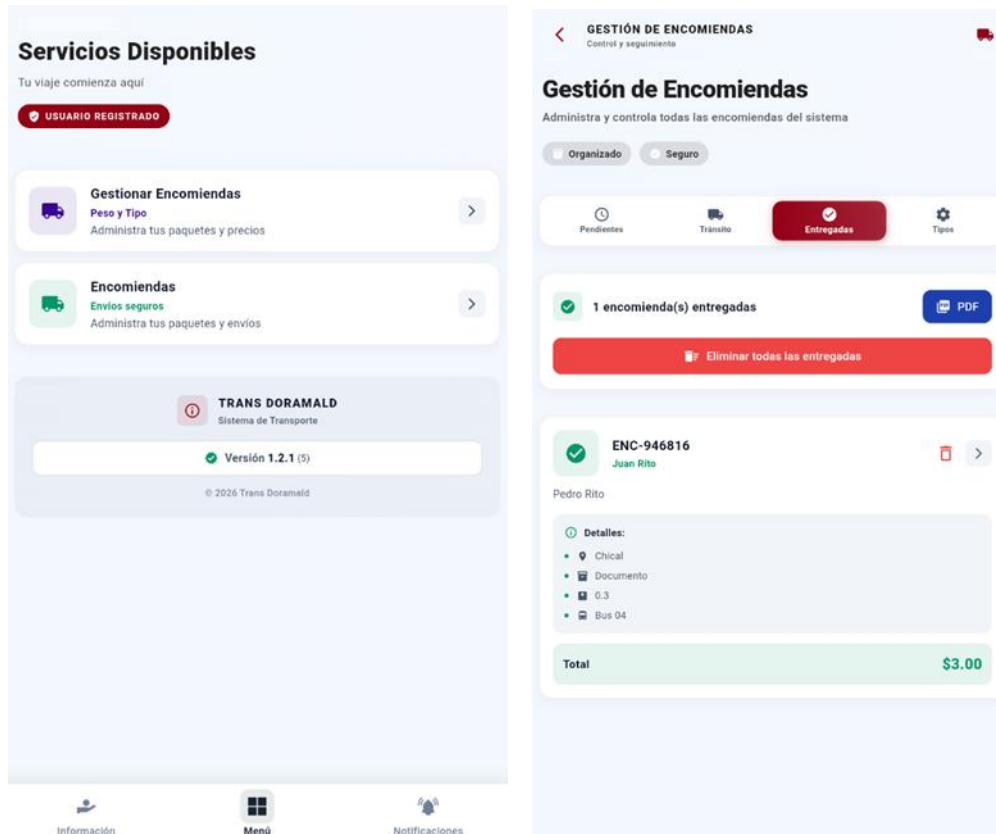
5.1. Acceso y Servicios

1. Inicie sesión con las credenciales específicas de conductor proporcionadas por la administración.
2. El dashboard simplificado muestra (Ver Figura 44):
 - Servicios activos asignados
 - Acceso directo a gestión de encomiendas
 - Estado de la unidad

5.2. Gestión de Encomiendas en Ruta

1. Seleccione "Gestión de Encomiendas" (Ver Figura 45).
2. Visualice la lista de paquetes asignados a su unidad.
3. Actualice el estado de cada envío:
 - **Recogido:** Confirma recepción en origen
 - **En tránsito:** Indica que el paquete viaja en la unidad

- **Entregado:** Confirma entrega al destinatario (requiere ingreso de código de tracking o escaneo QR)
4. Para entregas, solicite al destinatario su cédula de identidad y verifique contra el sistema.

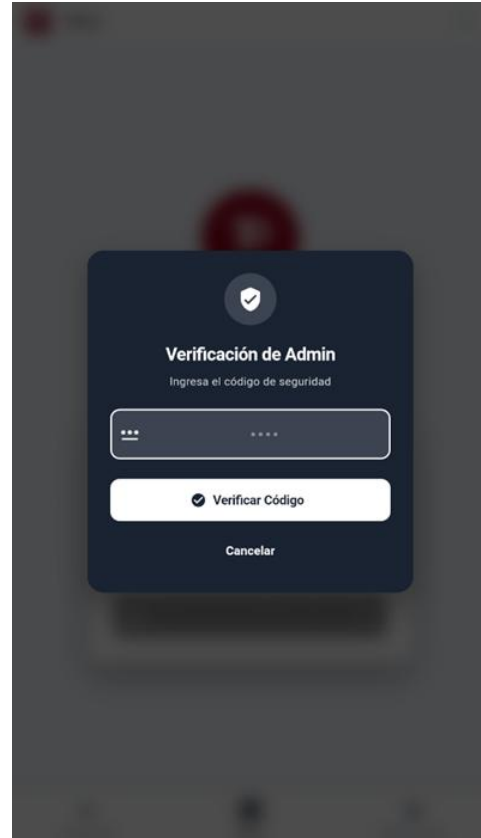
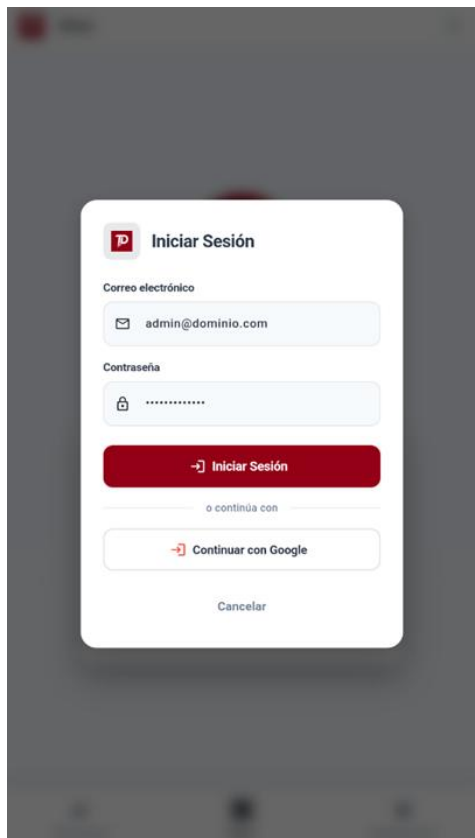


6. MÓDULO DE ADMINISTRADOR

6.1. Acceso Seguro (RF002)

El acceso administrativo requiere autenticación dual por seguridad (RNF003, RNF012):

1. En el Menú de Acceso, seleccione "Acceso Administrador" (Ver Figura 46).
2. Ingrese correo institucional y contraseña segura.
3. Proporcione el código de verificación numérico (PIN) de doble factor (Ver Figura 47).
4. Accederá al Panel de Control (Ver Figura 48).



6.2. Gestión de Rutas (RF007)

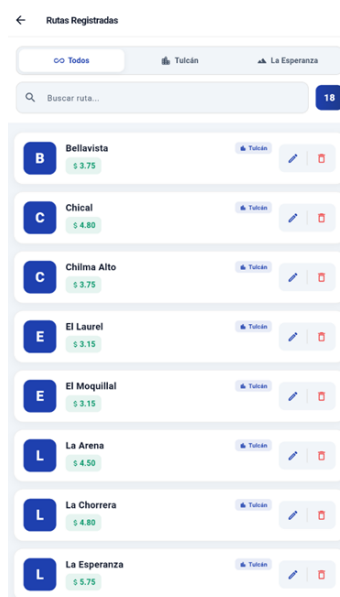
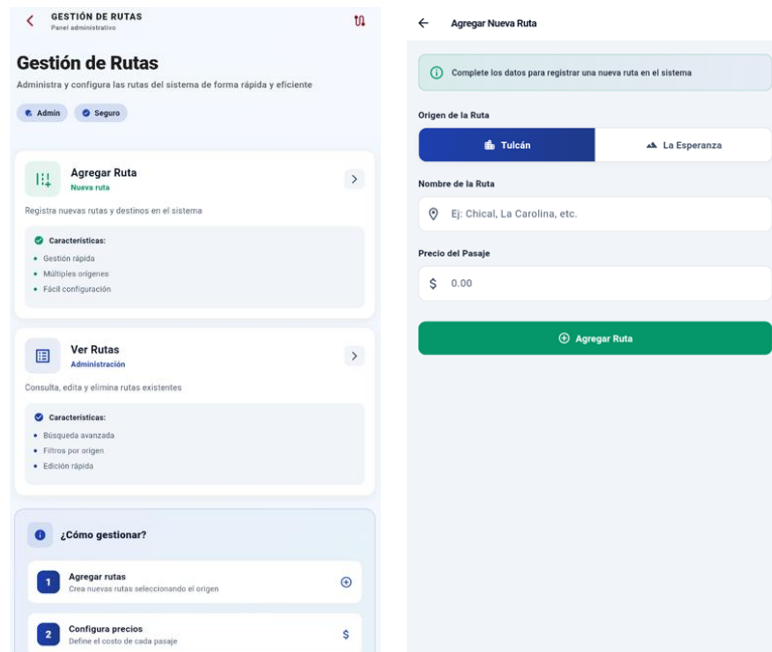
1. Desde el Panel de Control, seleccione "Gestión de Rutas" (Ver Figura 49).

2. Para crear nueva ruta:

- Pulse "Nueva Ruta" (Ver Figura 50)
- Ingrese: Nombre de ruta, Origen, Destino, Paradas intermedias
- Defina tarifa base y horarios de operación

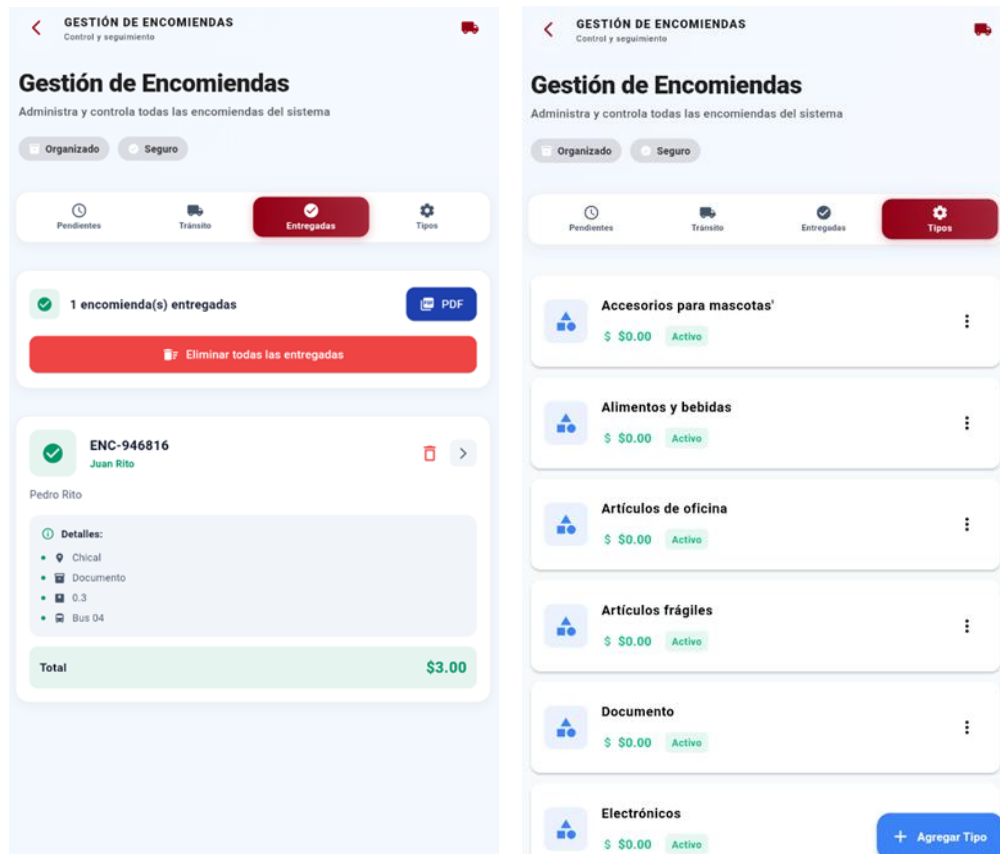
3. Visualice rutas existentes en el catálogo (Ver Figura 51).

4. Edite o desactive rutas según demanda estacional.



6.3. Gestión de Encomiendas

1. Panel de supervisión general (Ver Figura 52):
 - Filtros por estado, fecha, ruta, conductor
 - Reasignación manual entre unidades en caso de contingencias
 - Visualización de detalles completos (remitente, destinatario, valor)
2. Configuración de tipos de encomiendas (Ver Figura 53):
 - Defina categorías: Documentos, Carga frágil, Perecederos, Gran volumen
 - Establezca tarifas base por categoría
 - Especifique requisitos de embalaje



6.4. Comunicación Masiva (RF008)

1. Acceda a "Enviar Notificación" (Ver Figura 55).
2. Redacte el contenido del aviso (cambios de horario, emergencias viales, promociones).

3. Seleccione destinatarios:

- Todos los usuarios
- Grupos específicos (por ruta, por fecha de viaje)
- Usuarios individuales

4. Programe fecha/hora de envío o envío inmediato.

5. El sistema enviará push notifications y almacenará copia en el panel de cada usuario.

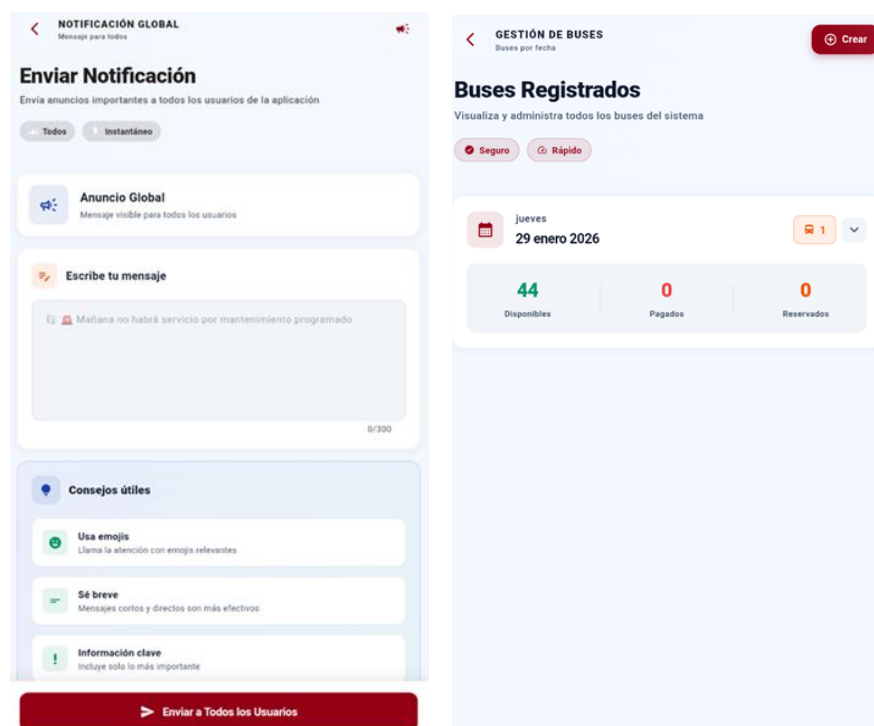
6.5. Gestión de Flota Vehicular (RF009, RF012)

1. Inventario de buses (Ver Figura 56):

- Registre nuevas unidades: Número, placa, capacidad, conductor asignado
- Visualice estado de cada unidad (activo/inactivo)

2. Programación de salidas (Ver Figura 57):

- Asigne bus a ruta específica
- Defina horario de salida y punto de origen
- Valide disponibilidad para evitar sobreasignación



The screenshot shows a mobile application interface for bus management. At the top, there is a header with a back arrow, the text 'GESTIÓN DE BUSES', and a sub-header 'Crear nuevo bus'. A 'Ver Buses' button is located in the top right corner. The main title is 'Crear Nuevo Bus', followed by the instruction 'Completa los datos para registrar un nuevo bus'. Below this are two toggle options: 'Seguro' (checked) and 'Rápido'. The form consists of four input fields: 'Número de Bus' (with a '#' icon), 'Lugar de Salida' (with a location pin icon), 'Fecha de Salida' (with a calendar icon and the text 'Seleccionar fecha'), and 'Hora de Salida' (with a clock icon and the text 'Seleccionar hora'). At the bottom of the form is a large red button labeled 'Crear Bus' with a checkmark icon.

6.6. Gestión de Contenido (RF010)

1. Acceda a "Galería de Imágenes" (Ver Figura 58).
2. Actualice fotografías de:
 - Pantalla principal (banners informativos)
 - Sección "Información" (fotos de flota, instalaciones)
 - Equipo humano (fotos oficiales del personal)
3. El sistema comprime automáticamente las imágenes para optimizar carga en zonas de baja conectividad (RNF005).

6.7. Validación Financiera (RF011)

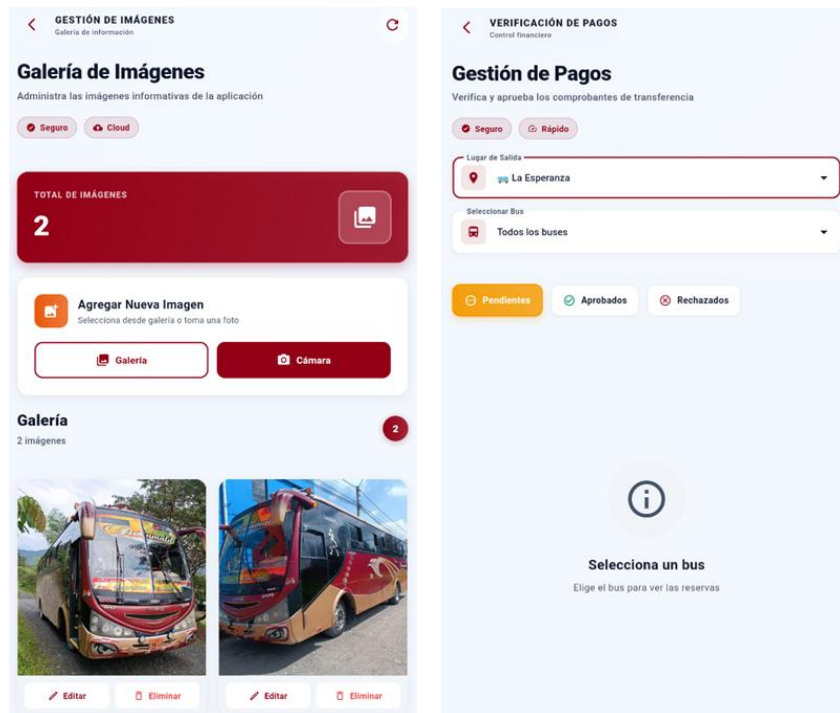
1. Panel de Gestión de Pagos (Ver Figura 59):
 - Visualice transacciones pendientes de verificación
 - Revise miniatura del comprobante de transferencia cargado por usuario
 - Compare monto declarado vs. monto en imagen
 - Acciones: "Aprobar" o "Rechazar" (con motivo)

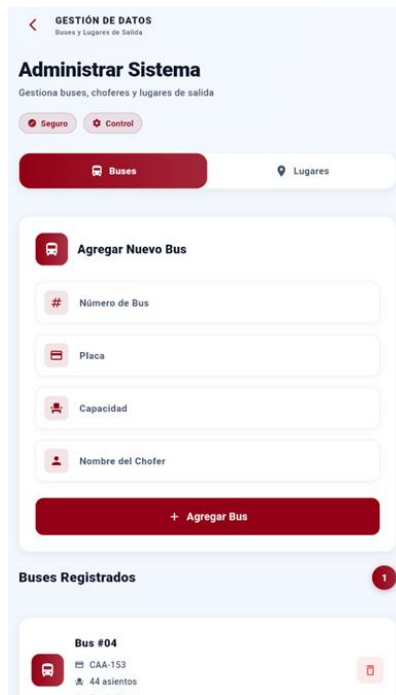
2. Genere reportes de ingresos por período, ruta o método de pago.

6.8. Gestión de Datos Maestros (RF012)

1. Panel de Gestión de Datos (Ver Figura 60):

- **Conductores:** Registro de datos personales, licencias, certificados
- **Unidades:** Documentación técnica, revisiones vehiculares, matrículas
- **Puntos de salida:** Georreferenciación de terminales y paradas autorizadas





7. SEGURIDAD Y RESOLUCIÓN DE PROBLEMAS

7.1. Recomendaciones de Seguridad

- **Cambio de contraseñas:** Realice cambios periódicos cada 3 meses (administradores).
- **Bloqueo de pantalla:** Mantenga activa la función de bloqueo automático en su dispositivo.
- **Cierre de sesión:** Cierre siempre la sesión al terminar, especialmente en dispositivos compartidos.
- **Verificación de pagos:** Nunca apruebe un pago sin verificar el comprobante bancario contra el monto declarado.

7.2. Operación Offline (Sin Internet)

El sistema está diseñado para funcionar en zonas rurales con conectividad intermitente:

- Las reservas se guardan localmente y sincronizan cuando hay señal.
- Los comprobantes pueden descargarse previamente para mostrarlos al conductor sin necesidad de datos móviles.

- Las notificaciones se almacenan y se entregan cuando el dispositivo recupera conexión.

7.3. Solución de Problemas Comunes

Problema	Causa Probable	Solución
Error al instalar el APK	Protección de Google Play Protect	Verifique que la fuente sea confiable (oficina de la compañía) y pulse "Instalar de todos modos"
No llegan notificaciones	Permisos desactivados	Verifique en Configuración > Aplicaciones > Trans Doramald > Notificaciones (deben estar activas)
Asiento no se puede seleccionar	Bloqueo temporal por otro usuario	Espere 10 minutos o seleccione otro asiento. Si persiste, contacte a administración
Pago no se refleja	Validación pendiente	Espere 30 minutos. Si no se aprueba, verifique que el comprobante sea legible y reenvíe
Aplicación lenta	Caché saturado	Vaya a Configuración > Almacenamiento > Limpiar caché de la aplicación
Error "Sin conexión"	Red inestable	Active el modo avión 5 segundos y desactívelo. Reinicie la app