



Documento de Alcance

Integrantes:

Mathías Fernández
Eduardo Flores
Federico Acosta
Gabriel Suárez
Pablo Perdomo
Alexander Rodriguez

Tutor:

Fernando Arrieta

Índice

Índice.....	2
1. Introducción.....	3
2. Resultados esperados.....	3
3. Requisitos.....	4
4. Hitos.....	5
5. Supuestos.....	5
6. Restricciones.....	6
7. Funcionalidades fuera del alcance.....	7
8. Funcionalidades futuras (post-producción).....	7
9. Partes interesadas.....	9
10. Planificación por iteraciones.....	9
11. Dedicación horaria.....	11
12. Distribución de roles.....	12

1. Introducción

El objetivo principal del proyecto es desarrollar un prototipo de un sistema digital integral que facilite la venta de pasajes de ómnibus de larga distancia en Uruguay. Este sistema debe incluir tanto una aplicación web como una aplicación móvil, adaptadas a los distintos perfiles de usuario: administrador, vendedor y cliente.

En síntesis, el proyecto tiene como meta final la entrega de un prototipo funcional que cumpla con los requisitos establecidos, acompañado de la documentación técnica correspondiente.

2. Resultados esperados

Como resultado del presente proyecto, se espera la entrega de un sistema funcional compuesto por tres componentes principales y su documentación técnica completa, que refleje todas las etapas del desarrollo.

- **Aplicación web:** Destinada a usuarios con perfil de administrador y vendedor, permitirá la gestión de usuarios, localidades, ómnibus, viajes, ventas y visualización de estadísticas. También ofrecerá funcionalidades básicas para el cliente.
- **Aplicación móvil:** Desarrollada en React Native y orientada exclusivamente al cliente final, permitirá registrarse, iniciar sesión, buscar y comprar pasajes, consultar historial de viajes, calificarlos y recibir notificaciones.
- **Backend:** Desarrollado en Java con Spring Boot, actuará como capa de negocio y de servicios, conectando ambas interfaces (web y móvil) con la base de datos PostgreSQL.
- **Documentación técnica:** Se incluirá toda la documentación requerida:
 - Documento de estado del arte.
 - Documento de pruebas de concepto (PoC).
 - Documento de alcance.
 - Documento de requerimientos.
 - Documento de casos de uso y diagramas.

- Modelo de dominio.
- Documento de arquitectura.
- Documento de diseño con mockups.
- Modelo de datos.
- Documento de verificación.
- Informe final integrador.

3. Requisitos

El sistema a desarrollar deberá cumplir con una serie de requisitos que aseguren su funcionalidad, rendimiento, compatibilidad, usabilidad y seguridad. Estos requisitos se dividen en tres grandes categorías: técnicos, funcionales y no funcionales. Su cumplimiento será verificado durante las etapas de desarrollo, pruebas y validación final.

Desde el punto de vista técnico, el proyecto debe implementarse utilizando tecnologías específicas previamente aprobadas. El backend será desarrollado en lenguaje Java utilizando el framework Spring Boot, lo cual permitirá una arquitectura modular, escalable y orientada a servicios. La interfaz web estará construida con React, una biblioteca moderna y eficiente que facilita el desarrollo de componentes reutilizables y una experiencia de usuario fluida. La aplicación móvil, por su parte, será desarrollada en React Native, permitiendo la creación de una aplicación nativa para Android, cumpliendo con los requerimientos de accesibilidad y portabilidad. La persistencia de datos estará respaldada por una base de datos PostgreSQL, elegida por su robustez, capacidad de manejar estructuras complejas y compatibilidad con entornos productivos. El sistema estará además contenedorizado mediante Docker para facilitar su despliegue y garantizar la consistencia entre ambientes.

4. Hitos

El proyecto está estructurado en torno a una serie de hitos que marcan los momentos clave dentro de su ciclo de vida. Estos hitos representan puntos de control esenciales para validar el cumplimiento del cronograma, medir el avance real frente a la planificación inicial y permitir ajustes oportunos.

1. El primer hito importante ocurre durante el mes de marzo, cuando se definen los roles del equipo, se instala el entorno de desarrollo y se elaboran las pruebas de concepto.
2. A continuación, en abril, se alcanzará el segundo hito con la presentación del documento de alcance, el relevamiento de requerimientos y la elaboración de los casos de uso principales.
3. En mayo, el proyecto atraviesa un momento crítico: se espera la finalización del análisis y diseño del sistema, incluyendo la arquitectura general, el modelo de datos, el modelo de dominio y los primeros mockups. Este mes también contempla la ejecución de la primera iteración de implementación, centrada en los casos de uso más relevantes.
4. El siguiente hito se fija para mediados de junio, donde se completa la segunda iteración de implementación, abarcando el resto de funcionalidades.
5. Finalmente, en julio, se entregará el informe final, se ejecutarán las pruebas definitivas del sistema, se preparará la defensa oral y se presentarán todos los entregables.

Estos hitos no sólo ordenan el desarrollo, sino que también constituyen instancias de evaluación parcial clave para el tutor.

5. Supuestos

El desarrollo de este proyecto se basa en una serie de supuestos que, aunque no están garantizados explícitamente, se consideran como verdaderos para la planificación y ejecución del mismo.

Se asume que cada integrante del equipo dispone de aproximadamente 20 horas semanales para trabajar de forma efectiva en las distintas tareas asignadas, lo cual se corresponde con la carga horaria esperada por los 20 créditos académicos del

curso. También se da por hecho que todos los estudiantes tienen acceso a una computadora personal con los requisitos técnicos adecuados, incluyendo capacidad para ejecutar entornos de desarrollo, herramientas de contenedorización como Docker y emuladores móviles para pruebas.

Asimismo, se presume que el equipo tiene conocimientos suficientes, o los adquirirá oportunamente, en las tecnologías seleccionadas: Java, Spring Boot, React, React Native, PostgreSQL, Docker, y herramientas de versionado como Git. También se considera que las reuniones quincenales con el tutor se mantendrán como espacio regular de revisión y orientación, y que la comunicación dentro del equipo será fluida y constante. Finalmente, se parte del supuesto de que los entornos de desarrollo y despliegue podrán configurarse sin grandes impedimentos técnicos, y que las dependencias externas utilizadas (frameworks, librerías, plataformas de prueba) estarán disponibles y operativas durante el ciclo de vida del proyecto.

6. Restricciones

El desarrollo del proyecto está sujeto a una serie de restricciones que condicionan su alcance, ejecución y posibles decisiones técnicas. Una de las principales limitaciones es el tiempo disponible: el trabajo debe completarse en un semestre académico, lo que implica aproximadamente cuatro meses desde el inicio hasta la entrega final. Dentro de este período deben cumplirse todas las etapas del proyecto (análisis, diseño, implementación, pruebas y documentación), respetando los plazos establecidos por el cronograma. Ésta restricción temporal exige una planificación precisa, una distribución adecuada de tareas entre los integrantes y un monitoreo constante del avance.

Otra limitación importante es la disponibilidad de recursos. El proyecto no cuenta con un presupuesto monetario, por lo que todas las herramientas, entornos y tecnologías empleadas deben ser gratuitas o de código abierto. Asimismo, el proyecto está condicionado por decisiones pedagógicas previas, como la obligatoriedad de utilizar Java en el backend.

Finalmente, se identifican como restricciones externas los posibles cambios en las versiones de librerías, fallas en plataformas de terceros o dificultades técnicas para configurar entornos, factores que podrían afectar el ritmo de desarrollo y deben ser considerados en la gestión de riesgos.

7. Funcionalidades fuera del alcance

Durante las etapas de análisis y diseño inicial del sistema se identificaron diversas funcionalidades potenciales que fueron deliberadamente excluidas de esta primera versión. La decisión respondió a criterios de complejidad técnica, acotamiento del tiempo disponible y alineación con los objetivos formativos del proyecto.

- Se resolvió implementar exclusivamente pagos en línea a través de una plataforma digital como lo es Stripe, descartando cualquier tipo de gestión de pagos presenciales o en puntos de venta físicos.
- No se contemplará la selección de viajes intermedios, por lo que el sistema estará limitado a la venta de pasajes exclusivamente entre una localidad de origen y una de destino, decisión tomada con el objetivo de acotar la complejidad lógica de esta primera versión del software.

8. Funcionalidades futuras (post-producción)

Una vez que el sistema se encuentre en producción y haya sido validado por los usuarios finales, se prevé la incorporación progresiva de nuevas funcionalidades orientadas a ampliar la cobertura operativa, optimizar la experiencia de uso y dar respuesta a necesidades avanzadas del negocio. Entre las funcionalidades proyectadas se destacan:

- Implementación de un sistema de fidelización que recompense la recurrencia de los clientes mediante la acumulación de puntos o beneficios.
- Incorporación de notificaciones por SMS como canal complementario a los actuales medios de notificación (correo electrónico y notificaciones push).
- Desarrollo de una lógica de precios dinámicos basada en variables como demanda, estacionalidad u otros criterios definidos por la empresa.

- Funcionalidad para la gestión de convenios institucionales con entidades educativas o empresas, que permita configurar condiciones especiales para compras grupales o masivas.
- Integración de múltiples empresas de transporte dentro de una misma plataforma, posibilitando una gestión multiempresa con control independiente de flotas, destinos y tarifas.
- Adición de un módulo completo para la gestión de encomiendas, con el fin de extender el alcance comercial del sistema más allá del transporte de pasajeros.
- Automatización del proceso de alta de usuarios institucionales bajo el dominio “@tecnobus.uy”, mediante la búsqueda inteligente en la base de datos de combinaciones posibles entre nombre y apellido. Se propondrán las tres combinaciones más simples que no estén registradas para que el usuario seleccione la más adecuada (ej.: nombre.apellido, nombre.apellido1.apellido2, nombre1.nombre2.apellido).
- Desarrollo de un sistema avanzado de soporte y gestión de reclamos, apoyado en tecnologías de inteligencia artificial, que permita registrar incidencias, realizar seguimientos y brindar respuestas multicanal de forma eficiente.
- Creación de una API pública que permita a otras empresas del sector integrar el sistema de Tecnobus en sus propias plataformas, facilitando la interoperabilidad comercial y la ampliación del alcance del servicio.
- Simplificación del inicio de sesión para administradores y vendedores, permitiendo autenticar únicamente con el nombre de usuario en lugar de ingresar el correo electrónico completo.

Estas funcionalidades se abordarán en futuras versiones del sistema, priorizando su desarrollo en función del feedback de los usuarios y de la evolución técnica y organizacional del proyecto.

9. Partes interesadas

Las partes interesadas del proyecto son todas aquellas personas o entidades que tienen un impacto directo o indirecto en el desarrollo del sistema, así como también quienes se verán afectados por sus resultados.

- En primer lugar, se encuentra el equipo de estudiantes desarrolladores, responsables de planificar, diseñar, implementar, testear y documentar el sistema en su totalidad. Su interés principal radica en lograr una solución funcional, técnicamente sólida y debidamente documentada que cumpla con los criterios de evaluación académica.
- El tutor académico constituye otra figura clave dentro de las partes interesadas. Su rol es guiar al equipo, proporcionar retroalimentación técnica y metodológica, validar decisiones estratégicas y supervisar el cumplimiento del cronograma.
- A su vez, el tribunal docente encargado de evaluar el proyecto en su fase final es una parte interesada crítica, ya que validará tanto el resultado funcional del sistema como la calidad del proceso de desarrollo y de la documentación entregada.
- Por último, aunque el sistema no será utilizado por una empresa real, se simula la existencia de usuarios finales (administradores, vendedores y clientes) como actores representativos.

10. Planificación por iteraciones

A continuación se detalla la distribución estimada de funcionalidades a desarrollar en cada una de las dos iteraciones de implementación. Estas iteraciones tienen una duración de dos semanas cada una, y están organizadas de acuerdo a la criticidad y prioridad de los casos de uso definidos en el análisis previo.

En el primer sprint se priorizan funciones críticas y en el segundo se completa la funcionalidad general con flujos secundarios y/o mejoras.

Iteración	Semana	Funcionalidades
Sprint 1 - Casos	1	<ul style="list-style-type: none"> • Inicio de sesión.

críticos		<ul style="list-style-type: none"> • Cierre de sesión. • Alta administrador. • Alta vendedor. • Alta cliente. • Alta ciudad. (individual y masivo) • Alta ómnibus. (individual y masivo) • Alta viaje.
	2	<ul style="list-style-type: none"> • Edición perfil. • Cambio contraseña. • Recuperación de contraseña. • Baja/deshabilitar usuarios • Compra pasajes • Venta pasajes • Reasignación de bus a un viaje • Cierre de ventas de pasajes para un viaje. • Marcado de un ómnibus como activo/inactivo. • Nuevo ómnibus para viaje existente
Sprint 1 - Casos complementarios	3	<ul style="list-style-type: none"> • Listado histórico de pasajes que el cliente haya comprado para distintos viajes. • Calificar viaje • Listado y búsqueda de viajes asignados a un ómnibus. • Listado y búsqueda de usuarios. • Listado y búsqueda de ómnibus. • Listado y búsqueda de viajes. • Listado de pasajes vendidos para un viaje.
	4	<ul style="list-style-type: none"> • Devolución de un pasaje previamente comprado por un cliente que decide no viajar. • Estadísticas de viajes, ómnibus y pasajes. • Estadísticas de usuarios. • Baja de viaje. • Baja de ómnibus • Baja de localidad. • Push notification.

11. Dedicación horaria

A continuación se detalla un estimativo de la dedicación horaria total del equipo.

Actividad	Horas Estimadas	Horas Reales Total
Monitoreos con tutor	6	6
Coordinaciones grupales	35	60
Configuración de entornos de desarrollo	20	10
Documento de Estado del Arte	35	25
Documento de Pruebas de Concepto	90	120
Documento de Alcance	20	25
Documento de Requerimientos	30	20
Documento de Casos de Usos	60	110
Documento de Modelos de Dominio	15	20
Glosario	5	5
Documento de Arquitectura	20	20
Documento de Diseño	50	75
Documento de Modelo de Datos	40	45
Documento de Verificación	20	35
Implementación del Prototipo Funcional	400	550
Testing	80	130
Informe Final	50	60
Preparación y Defensa final	5	10
TOTAL	981	1326

12. Distribución de roles

Con el fin de organizar de forma eficiente el trabajo y asegurar una cobertura integral de todas las áreas involucradas en el desarrollo del sistema, se definieron distintos roles dentro del equipo. Cada integrante asume uno o más roles en función de sus habilidades, buscando complementariedad técnica y operativa a lo largo del proyecto.

	Eduardo Flores	Federico Acosta	Gabriel Suárez	Mathias Fernández	Pablo Perdomo	Alexander Rodríguez
Backend	✓				✓	
Frontend (Frontoffice)		✓				
Frontend (Backoffice)			✓		✓	
Mobile				✓		
DevOps					✓	
DBA	✓					
Testing		✓				✓
Analista Funcional			✓			✓
Coordinador				✓		