

# Universidad Don Bosco



**Asignatura:** Desarrollo de Aplicaciones con Web Frameworks

**Actividad:** Fase I

**Docente:** Mario Alvarado

**Github:** <https://github.com/pgap22/dwf-catedra>

Carné	Estudiantes	Roles
BS242634	Gerardo Rafael Bonilla Saz	DBA
JA242663	Fernando Alexander Jiménez Artiga	Frontend
PA242675	Rodrigo Daniel Pineda Ardón	Documentador
VE242685	Leandro Alberto Valencia Escobar	Backend

Soyapango, 30 de agosto de 2025.

# Índice

<b>Índice</b>	<b>2</b>
<b>3. Descripción del Proyecto</b>	<b>3</b>
a) Objetivo General	3
b) Objetivos Específicos	3
<b>4. Gestión Integral del Proyecto</b>	<b>4</b>
4.1. Gestión de la Integración	4
4.2. Gestión del Alcance	5
Límites del proyecto en esta fase:	5
4.3 Gestión del Cronograma	6
4.4. Gestión de Costos	8
4.5. Gestión de la Calidad	9
Prácticas y Estándares de Desarrollo	9
4.6. Gestión de los Recursos Humanos	10
4.7. Gestión de la Comunicación	11
a) Informes de Progreso:	11
b) Canales de Resolución de Dudas:	11
4.8. Gestión de Adquisición y Cierre	11
a) Documentación Final:	11
<b>5. Descripción Técnica</b>	<b>12</b>
5.1. Prototipo de FrontEnd	12
5.2. Gestión de la Base de Datos	25
a) Estructura de la Base de Datos: Modelo Relacional	25
b) Diccionario de datos	31
<b>6. Resultados y Conclusiones</b>	<b>43</b>
a) Resultados obtenidos	43
b) Conclusiones	43
<b>7. Anexos y Documentación de Apoyo</b>	<b>44</b>
7.1. Configuraciones Avanzadas	44
7.2. Ejemplos de Código	44
7.2. Documentación Técnica Adicional	

### **3. Descripción del Proyecto**

#### **a) Objetivo General**

Desarrollar un sistema de gestión de boletos aéreos basado en una API REST con Java Spring Boot, que permita automatizar y optimizar los procesos de registro de vuelos, aerolíneas, pasajeros y reservaciones, garantizando integridad de datos, trazabilidad de transacciones y disponibilidad de información en tiempo real para los usuarios y administradores del sistema.

#### **b) Objetivos Específicos**

- Diseñar la estructura de la base de datos relacional que soporte los procesos de aerolíneas, vuelos, pasajeros, reservaciones, pagos y cancelaciones.
- Definir la arquitectura de la API REST con endpoints claros para cada operación de negocio.
- Crear prototipos de interfaz de usuario que permitan visualizar la interacción entre el usuario y el sistema.

## 4. Gestión Integral del Proyecto

### 4.1. Gestión de la Integración

La gestión de la integración del proyecto se llevará a cabo mediante un enfoque colaborativo apoyado en herramientas digitales y metodologías de trabajo adaptadas al contexto del equipo. Para el control de versiones se empleará **GitHub**, donde cada integrante mantendrá una **rama propia** asociada a sus tareas específicas. De esta forma, se asegura la independencia en el desarrollo y se facilita la posterior integración del código mediante *merge requests*.

El **seguimiento de tareas y hitos** se organizará en **Excel**, lo que permitirá mantener un control accesible y flexible de las actividades, asignaciones y avances. Esta herramienta funcionará como repositorio de cronogramas, responsables y estados de cumplimiento.

La **metodología de trabajo** seleccionada es **Design Thinking**, debido a que promueve la creatividad, la iteración constante y la orientación hacia la resolución de necesidades reales del usuario. El equipo desarrollará el proyecto en fases de ideación, prototipado y validación, favoreciendo la retroalimentación continua.

Los **roles del equipo** se definen de la siguiente manera:

- **Rodrigo Pineda**: responsable de documentación y control de entregables.
- **Fernando Jiménez**: encargado de desarrollo **Frontend**.
- **Gerardo Saz**: responsable de la **Base de Datos**.
- **Leandro Valencia**: encargado del **Backend**.

De esta manera, se busca una integración coherente entre los distintos componentes del sistema, garantizando trazabilidad, organización y eficiencia en la colaboración.

## 4.2. Gestión del Alcance

El alcance del proyecto se centra en el desarrollo de un sistema web para la gestión integral de boletos aéreos, con el objetivo de automatizar y optimizar los procesos principales asociados a la venta, administración y seguimiento de vuelos. La aplicación estará compuesta por un backend desarrollado en Java con Spring Boot, consumido por un frontend construido con React y Tailwind, y respaldado por una base de datos relacional.

Las funcionalidades mínimas que el sistema debe contemplar incluyen:

- **Gestión de aerolíneas:** Registro, actualización y eliminación de aerolíneas, garantizando la integridad de los datos.
- **Administración de vuelos:** Creación y modificación de vuelos, rutas, horarios y tarifas asociadas.
- **Gestión de aviones y tripulación:** Registro y asignación de aeronaves y personal a los vuelos correspondientes.
- **Registro de pasajeros:** Captura de información básica y relevante de los pasajeros, incluyendo datos personales y documentación.
- **Reservaciones y pagos:** Proceso de reserva de boletos desde la selección de vuelo hasta la confirmación. En esta fase únicamente se documentará y diseñará la lógica de pago, sin conectarla a una pasarela real.
- **Cancelaciones y modificaciones:** Gestión de cambios o cancelaciones de reservas siguiendo políticas definidas.
- **Procesamiento de reclamos:** Registro y seguimiento de incidencias relacionadas con reservas o vuelos.
- **Generación de estadísticas básicas:** Reportes sobre número de reservas, cancelaciones y métricas relevantes para la gestión de la aerolínea

### Límites del proyecto en esta fase:

Dado que esta entrega corresponde a la primera fase, se concentrará en la **documentación técnica, planificación, prototipos de interfaz y diseño de base de datos**, sin implementar aún la funcionalidad completa del sistema.

Un límite importante será la **implementación de la pasarela de pagos**, ya que por cuestiones de tiempo no se desarrollará ni se conectará con un servicio de terceros en esta etapa. En su lugar, se establecerá una estructura simulada para contemplar la validación de pagos, la cual se integrará de forma completa en la segunda fase.

## 4.3 Gestión del Cronograma

El cronograma del proyecto se ha organizado considerando la Fase I, comprendida entre el **16 y el 29 de agosto de 2025**, con el objetivo de garantizar el cumplimiento de los hitos establecidos y la entrega del documento técnico de manera ordenada y eficiente. Para una gestión visual y colaborativa del tiempo, se mantendrá un **cronograma dinámico en Google Sheets**, donde se actualizarán las actividades, responsables y fechas de revisión de manera continua.

### Distribución semanal de actividades:

#### Semana 1 (16 al 22 de agosto 2025)

- Reunión inicial del equipo para asignación de roles y responsabilidades.
- Evaluación y selección de tecnologías a utilizar: backend en Spring Boot, frontend con React y Tailwind, y base de datos relacional (MySQL/PostgreSQL).
- Definición de la estructura general del proyecto y organización de carpetas.
- Diseño preliminar del modelo de base de datos y determinación de entidades principales.

#### Semana 2 (23 al 29 de agosto 2025)

- Revisión y ajuste del modelo de base de datos para asegurar consistencia y viabilidad.
- Planeación de la futura integración de la API REST con el frontend, considerando la forma en que los endpoints serán consumidos desde React.
- Desarrollo de prototipos de interfaz de usuario, con énfasis en la navegación y flujo de la aplicación.
- Reuniones de seguimiento para consolidar la documentación técnica del proyecto y validar avances.
- Preparación y entrega del documento técnico en formato PDF, cumpliendo con los estándares de presentación establecidos.

### Hitos clave:

- Reuniones de coordinación y discusión tecnológica.
- Revisión y validación del modelo de base de datos.
- Definición de la estructura del proyecto y organización de carpetas.
- Elaboración y revisión de prototipos de interfaz.
- Entrega del documento técnico final.

El seguimiento continuo de este cronograma permitirá al equipo mantener un control preciso sobre el progreso de las actividades y garantizar que cada fase se cumpla en los plazos establecidos.

Actividad	Responsable	Fecha de inicio	Fecha de fin	Estado
Reunión inicial del equipo y asignación de roles	Rodrigo Pineda (Documentación)	16/08/2025	16/08/2025	Completada
Evaluación y selección de tecnologías (Spring Boot, React, Tailwind, DB)	Todos	16/08/2025	17/08/2025	Completada
Definición de la estructura general del proyecto y organización de carpetas	Leandro Valencia (Backend) & Fernando Jiménez (Frontend)	17/08/2025	18/08/2025	Completada
Diseño preliminar del modelo de base de datos	Gerardo Saz (Base de datos)	17/08/2025	19/08/2025	Completada
Revisión y ajuste del modelo de base de datos	Gerardo Saz & Rodrigo Pineda	20/08/2025	21/08/2025	Completada
Planeación de integración de API REST con frontend React	Leandro Valencia & Fernando Jiménez	20/08/2025	22/08/2025	Completada
Desarrollo de prototipos de interfaz de usuario (mínimo 10 maquetas)	Fernando Jiménez	21/08/2025	25/08/2025	Completada
Reuniones de seguimiento y consolidación de documentación técnica	Todos	23/08/2025	28/08/2025	Completada
Revisión final del documento técnico	Rodrigo Pineda	28/08/2025	29/08/2025	Completada
Entrega del documento técnico PDF	Rodrigo Pineda	29/08/2025	29/08/2025	Completada

#### 4.4. Gestión de Costos

Con el objetivo de garantizar una planificación financiera realista para la primera fase del proyecto, se realizó un análisis detallado de los recursos y herramientas requeridas para el desarrollo de la solución. Se

contemplaron tanto las licencias de software como los entornos de prueba y herramientas de diseño necesarias para la correcta implementación del sistema.

Recurso / Herramienta	Propósito	Costo estimado
Visual Studio Code	Desarrollo de código frontend y backend	Gratis
IntelliJ IDEA (cuenta universitaria)	Desarrollo backend en Java	Gratis
Figma	Diseño de prototipos de interfaz	Gratis
PHPMyAdmin	Gestión de base de datos	Gratis
Postman	Pruebas de endpoints y API REST	Gratis
Google Docs / Sheets	Gestión del cronograma y planificación	Gratis

#### Resumen de costos:

Todas las herramientas y recursos seleccionados para esta fase del proyecto son gratuitos o proporcionados mediante licencias académicas, lo que permite optimizar los recursos disponibles y concentrar los esfuerzos en el desarrollo, documentación y pruebas del sistema. Este enfoque garantiza un uso eficiente de los recursos y asegura que el proyecto se ejecute dentro de un marco presupuestario controlado y sostenible.

## 4.5. Gestión de la Calidad

### Prácticas y Estándares de Desarrollo

Para garantizar la calidad del código y la funcionalidad de la aplicación, el equipo implementará una serie de prácticas y estándares durante el desarrollo:

#### **a) Revisión de Código**

Las revisiones de código se llevarán a cabo de manera periódica en GitHub mediante *pull requests*. Cada integrante trabajará en su propia rama antes de integrar cambios al proyecto principal. Estas revisiones permitirán detectar errores, inconsistencias y oportunidades de mejora de forma temprana, promoviendo un código limpio, seguro y funcional.

#### **b) Estándares de Codificación**

Se seguirán buenas prácticas de programación, incluyendo:

- Uso de **camelCase** para la nomenclatura de variables y funciones.
- Convenciones claras y consistentes en nombres de archivos, clases y métodos.
- Comentarios breves y precisos en fragmentos complejos de código.
- Cumplimiento de guías de estilo previamente definidas por el equipo.

#### **c) Pruebas y Verificación**

- Se implementarán **pruebas unitarias y de integración** para validar las funcionalidades críticas del sistema.
- La validación de módulos se realizará de forma colaborativa en reuniones de equipo, asegurando que cada componente cumpla con los criterios de diseño definidos y se integre correctamente en el sistema global.

#### **d) Cultura de Calidad**

Cada miembro del equipo es responsable de la calidad de su trabajo y de contribuir a mantener los estándares generales. Se fomentará:

- La revisión continua.
- La escritura de código testeable y mantenable.
- La comunicación efectiva para resolver problemas de manera oportuna.

## **4.6. Gestión de los Recursos Humanos**

El proyecto se organiza estableciendo roles claros y específicos para cada integrante del equipo, asegurando una distribución eficiente de las responsabilidades y fomentando la colaboración efectiva:

- **Rodrigo Pineda:** Encargado de la documentación técnica y coordinación general del proyecto.

- **Fernando Jiménez:** Responsable del desarrollo del FrontEnd, utilizando React y Tailwind.
- **Gerardo Saz:** Responsable del diseño y administración de la base de datos mediante PhpMyAdmin.
- **Leandro Valencia:** Responsable del BackEnd, estructuración de los endpoints y lógica del sistema.

Cada miembro trabajará en su propia rama dentro del repositorio de GitHub, permitiendo un control de versiones ordenado y seguro. Las integraciones a la rama principal se realizarán mediante pull requests, asegurando revisiones previas y manteniendo la consistencia del código.

Para facilitar la comunicación y coordinación, se utilizarán los siguientes canales:

- **Discord:** Reuniones cada tres días para evaluar avances, definir estrategias y resolver cuestiones técnicas.
- **WhatsApp:** Comunicación inmediata para consultas puntuales y coordinación rápida.
- **Correo electrónico y Aula Virtual:** Envío de reportes formales correspondientes a la Fase I y Fase Final.

Esta estructura de roles y canales de comunicación garantiza claridad en las responsabilidades, continuidad en el seguimiento del proyecto y un flujo de información constante y ordenado entre los integrantes del equipo.

## 4.7. Gestión de la Comunicación

La comunicación dentro del equipo se estructurará para garantizar un flujo constante de información, permitiendo la resolución oportuna de dudas y el seguimiento adecuado del proyecto.

### a) Informes de Progreso:

El estado del proyecto se actualizará de manera continua durante las reuniones de equipo, que se realizarán cada tres días a través de Discord. En estas sesiones se dialogará y socializará el avance de las tareas, discutiendo problemas, soluciones y próximos pasos.

b) Canales de Resolución de Dudas:

Para consultas relacionadas con el proyecto, el equipo utilizará los medios oficiales establecidos por la institución, incluyendo el Aula Virtual y el correo electrónico. Adicionalmente, se podrán plantear dudas durante los horarios de clase, asegurando así una comunicación clara y eficiente con el docente.

## 4.8. Gestión de Adquisición y Cierre

En esta fase, se establece la entrega formal de la documentación correspondiente a la primera etapa del proyecto. Se garantiza que todos los elementos técnicos y de gestión estén completos, revisados y disponibles para su evaluación.

a) Documentación Final:

La documentación se presentará en formato PDF y se complementará con un archivo README en el repositorio de GitHub. Cada miembro del equipo trabajará en su propia rama, permitiendo una gestión organizada de los avances y evitando conflictos de integración. La documentación incluirá:

- Nombres y roles de los integrantes del equipo.
- Registro de reuniones y discusiones sobre el proyecto, realizadas mediante Discord y WhatsApp, para seguimiento de decisiones y coordinación.

Este enfoque asegura que toda la información relevante esté disponible de manera clara, estructurada y fácilmente accesible, facilitando tanto la revisión académica como la transición a la fase final del proyecto.

## 5. Descripción Técnica

### 5.1. Prototipo de FrontEnd

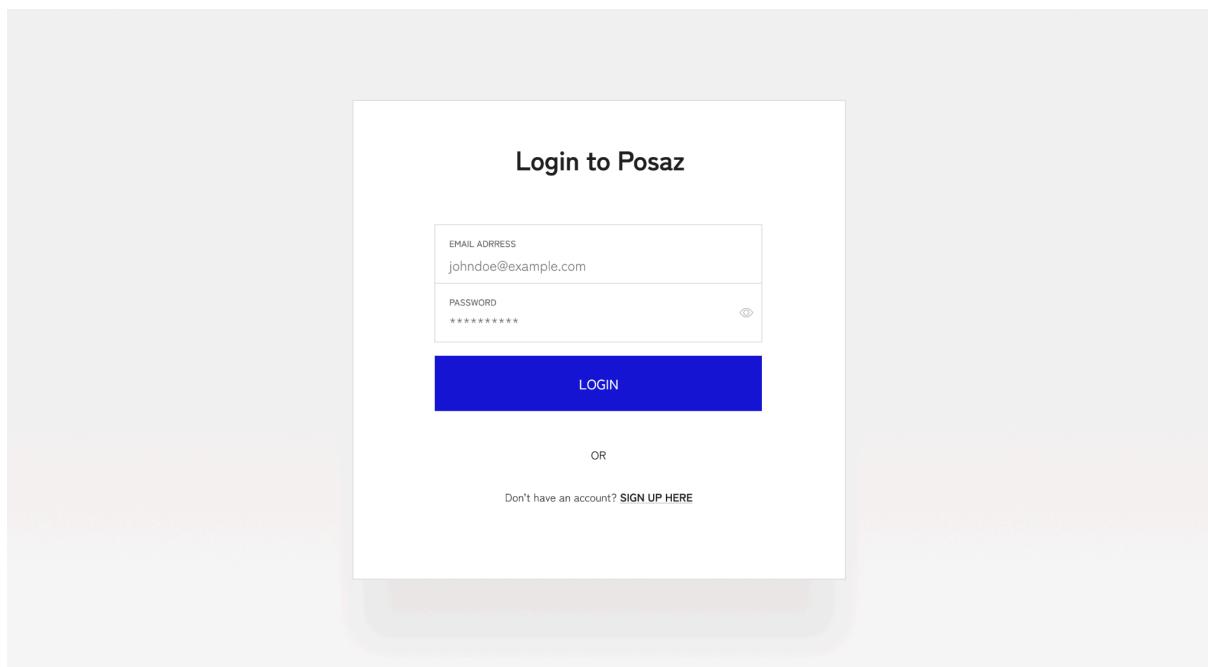
#### Registro

The screenshot shows a registration form titled "Sign up to Posaz". The form has three input fields: "FIRST NAME" containing "John", "EMAIL ADDRESS" containing "johndoe@example.com", and "PASSWORD". There is also a "Forgot password?" link and a "Privacy Policy" link at the bottom right.

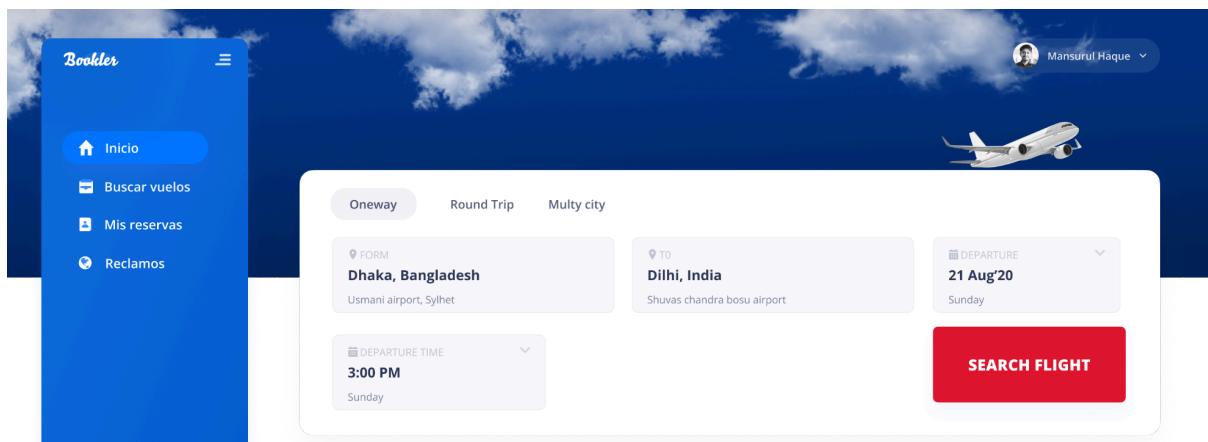
FIRST NAME	John
EMAIL ADDRESS	johndoe@example.com
PASSWORD	[REDACTED]

Forgot password? | Privacy Policy

## Inicio de sesión



## Página de inicio



The image displays the homepage of the Bookler flight booking platform. On the left, there is a vertical sidebar with a blue header containing the "Bookler" logo and a three-dot menu icon. Below the header, the sidebar features four buttons: "Inicio" (highlighted in blue), "Buscar vuelos", "Mis reservas", and "Reclamos". The main content area has a dark blue header with a white airplane silhouette and the name "Mansurul Haque". Below the header, there is a search form with options for "Oneway", "Round Trip", and "Multi city". The "FROM" field is set to "Dhaka, Bangladesh" (Usmani airport, Sylhet). The "TO" field is set to "Delhi, India" (Shuvas chandra bosu airport). The "DEPARTURE" date is "21 Aug'20" (Sunday). A large red "SEARCH FLIGHT" button is located at the bottom right of the search form. At the very bottom of the page, there is a "Special Offer" section with a "View all" link.

## Buscar vuelos

The screenshot displays the Bookler mobile application's flight search interface. At the top, there is a dark header bar with the Bookler logo on the left and a user profile icon on the right. Below the header, the main content area has a blue header titled "Buscar vuelos". On the left side, a vertical navigation menu is visible with options: "Inicio" (selected), "Buscar vuelos" (highlighted in blue), "Mis reservas", and "Reclamos". The main search form consists of several input fields: "TYPE" (Select), "FORM" (Select), "TO" (Select), and "DEPARTURE - RETURN" (Select). To the right of these fields is a large red button labeled "Search Flight". Below the search form, there is a graphic of a globe with a stylized airplane flying over it. At the bottom of the screen, a message reads "No has buscado un vuelo".

The screenshot shows the Booker flight search interface. The left sidebar has links for Inicio, Buscar vuelos, Mis reservas, and Reclamos. The main search bar specifies Onway, Dhaka,Bangladesh as the origin, Delhi, India as the destination, and the date range 21 Aug'19 - 29 Aug'19. The search button is labeled "Search Flight". Below the search bar, the results are displayed under the heading "Flights | Total 126 result". There are five flight options listed:

- SkyHigh Airlines** Boeing 737 MAX 9. Departure: 6:00 AM Los Angeles. Arrival: 9:00 PM New York. Duration: 3 hours. Direct. Facilities: Puedes cancelar, Puedes modificar reserva. [Ver detalles](#)
- FlyFast Airways** Boeing 787 Dreamliner. Departure: 8:00 AM Los Angeles. Arrival: 11:30 AM New York. Duration: 3 h 30 m. Escala CDMX. Facilities: Puedes cancelar, Puedes modificar reserva. [Ver detalles](#)
- AeroJet** Boeing 787 Dreamliner. Departure: 10:00 AM Los Angeles. Arrival: 12:45 PM New York. Duration: 2 h 45 m. Escala SAL. Facilities: No puedes cancelar, No puedes modificar reserva. [Ver detalles](#)
- JetStream Aviation** Boeing 737 MAX 9. Departure: 10:30 AM Los Angeles. Arrival: 1:00 PM New York. Duration: 2 h 30 m. Directo. Facilities: No puedes cancelar, Puedes modificar reserva. [Ver detalles](#)
- Nimbus Airlines** Boeing 737 MAX 9. Departure: 11:00 AM Los Angeles. Arrival: 2:00 PM New York. Duration: 3 hours. Escala CHI. Facilities: Puedes cancelar, No puedes modificar reserva. [Ver detalles](#)

(Ver detalles)

This is a detailed view of a flight from SkyHigh Airlines. The flight departs at 10:00 AM from Los Angeles and arrives at 1:00 PM in New York, with a duration of 3 hours. It is a direct flight. The facilities available are Puedes cancelar and Puedes modificar reserva. The flight is identified by the code SHI234 and is in the Económico class.

**SkyHigh Airlines**  
Boeing 737 MAX 9

10:00 AM  
Los Angeles

1:00 PM  
New York

3 hours • Directo

Facilities: Puedes cancelar, Puedes modificar reserva

[Cerrar detalles](#)

**SkyHigh Airlines**  
SHI234 Económico

\$350 /pax

[Reservar vuelo](#)

**SkyHigh Airlines**  
SHI234 Primera clase

\$550 /pax

[Reservar vuelo](#)

**Información del viajero**

La información del viajero que aparece aquí debe coincidir exactamente con la información que se encuentra en la identificación con foto emitida por el gobierno que el viajero presente en el aeropuerto.

**Viajero 1:**

Nombres (obligatorio)	Apellidos (obligatorio)
Leandro Alberto	Valencia Escobar
Fecha de nacimiento	Número de pasaporte
24/03/1999	+8801 4454 4544

**Viaje ida**

San Salvador SAL a Los Ángeles LAX	
17 de octubre De 2:00 a.m. a 6:19 a.m.	
> Mostrar detalles	
<b>Tarifa</b>	
Clase:	Primera clase
2 adultos mayores de 18 años	USD 541.00/perso
<b>Costo total</b>	
USD 1,451.54	

**Viajero 2:**

Nombres (obligatorio)	Apellidos (obligatorio)
Rodrigo Daniel	Pineda Ardón
Fecha de nacimiento (obligatorio)	Número de pasaporte (obligatorio)
24/03/1999	+8801 4454 4544

**Viaje ida**

San Salvador SAL a Los Ángeles LAX	
17 de octubre De 2:00 a.m. a 6:19 a.m.	
Aerolinea: United Airlines	
Duración: 5 h 19 m	
Número de vuelo: UA 1670	
Tipo de aeronave: Boeing 737 MAX 9	
Ruta: Escala en Ciudad de México CDMX	
> Ocultar detalles	
<b>Tarifa</b>	
Clase:	Primera clase
2 adultos mayores de 18 años	USD 541.00/perso
<b>Costo total</b>	
USD 1,451.54	

**Continúe para seleccionar asientos**

**+ Añadir**

**Seleccionar asientos**

Aerolinea: United Airlines Clase: Económica Fecha: vie, 17 de oct de 2025 Vuelo: UA 1670 Aeroav: Boeing 737 MAX 9 Duración: 5 h 19 m

Seleccionar asientos para el vuelo de San Salvador a Los Ángeles

**Descripción**

- Disponible (Blue square)
- Deshabilitado (Grey X)
- Reservado (Light Blue)

**131 asientos disponibles**

A	B	C	D	E	F
X			9		X
X			10		X
			11		
X			12	X	
			13	X	
X			14	X	X
(I) Exit row					

**Booker**

**Buscar vuelos**

**Mis reservas**

**Reclamos**

**TYPE** Onway **FORM** San Salvador, SV **TO** Los Angeles, CA, US **DEPARTURE - RETURN** 21 Aug'19 - 29 Aug'19 **Search Flight**

### Seleccionar asientos

Aerolínea: United Airlines Clase: Primera clase Fecha: vie, 17 de oct de 2025 Vuelo: UA 1670 Aeronave: Boeing 737 MAX 9 Duración: 5 h 19 m

Seleccionar asientos para el vuelo de San Salvador a Los Ángeles

A	B	C	D	E	F
1	Leandro Valencia	...	1	X	X
2	Rodrigo Pineda	...	2	X	X
3			3	X	X
4			4	X	
5			5	X	X
6			6		X
7			7	X	
8			8	X	X

131 asientos disponibles

Descripción

- Disponible
- Deshabilitado
- Reservado

Continuar para hacer check-out

**Booker**

**Buscar vuelos**

**Mis reservas**

**Reclamos**

**TYPE** Onway **FORM** Dhaka,Bangladesh **TO** Delhi, India **DEPARTURE - RETURN** 21 Aug'19 - 29 Aug'19 **Search Flight**

### Hacer el check-out

The session will expire in: 26min5sec

Vuelo seleccionado  Detalles de vuelo  Pago

#### Método de pago

Tarjeta crédito/débito  Transferencia

Número de tarjeta  
8893-2834-2424-5353

Nombre del titular

The screenshot shows the Bookler flight booking interface. On the left, a sidebar menu includes 'Inicio', 'Buscar vuelos' (selected), 'Mis reservas', 'Historial viajes', and 'Reclamos'. The main content area displays a flight search form with 'TYPE: Onway', 'FROM: Dhaka, Bangladesh', 'TO: Delhi, India', 'DEPARTURE - RETURN: 21 Aug'19 - 29 Aug'19', and a 'Search Flight' button. A central modal window titled 'Hacer el check-out' (Check-out) shows a success message: 'Your order has been placed!' with a small airplane icon above it. Below the message is a blue 'Thanks!' button. At the bottom of the modal, there are dropdown menus for 'Fecha de caducidad' (July (Month)), 'Año de caducidad' (2020 (Year)), and 'Código CVV' (343483 (CVV)). A checkbox for accepting terms and conditions is checked, and a note states 'The session will expire in: 26min5sec'. The total price '5,000 USD' and a 'Comprar' (Buy) button are also visible.

## Reservas

The screenshot shows the 'Mis viajes' (My trips) section of the Bookler website. The sidebar menu is identical to the previous screen. The main content lists three travel reservations:

- Paris → New York**  
CloudNine Airlines | Departure: 9:00 AM (CDG) July 1, 2028 | Arrival: 12:00 PM (JFK) July 1, 2028 | Tarifa: Clase económica \$350 | [Ver detalles](#)
- Frankfurt → Bangkok**  
SkyHigh Airlines | Departure: 7:00 AM (FRA) July 20, 2028 | Arrival: 3:00 PM (BKK) July 20, 2028 | Tarifa: Primero clase \$550 | [Ver detalles](#)
- London → New York**  
[Information partially cut off] | Tarifa: [Information partially cut off] | [Cancelar](#) | Booking ID: FF-CD5678

**Booker**

Regresar a Mis Viajes  
Los Angeles → New York  
Boeing 787 Dreamliner \*

SkyHigh Airlines SH-ZY6789 (Próximo)

Fecha July 1, 2028 Pago Tarjeta Débito \$350 /pax Modificar Cancelar vuelo

6:00 AM 15 Jul 2028 Los Angeles Los Angeles International Airport (LAX)

13 hours

9:00 PM 15 Jul 2028 New York John F. Kennedy International Airport (JFK)

**Viajero 1**  
Leandro Alberto Valencia Escobar  
24/03/1999 +8801 4454 4544 Asiento D18

**Viajero 2**  
Rodrigo Daniel Pineda Ardón  
24/03/1999 +8801 4454 4544 Asiento E16

Puedes modificar y cancelar

Agregar Reclamo !

Mansurul Haque

**Booker**

Regresar a Mis Viajes  
Los Angeles → New York  
Boeing 787 Dreamliner \*

SkyHigh Airlines SH-ZY6789 (Próximo)

Fecha July 1, 2028 Pago Transferencia \$580 /pax Cancelar vuelo

6:00 AM 15 Jul 2028 Los Angeles Los Angeles International Airport (LAX)

13 hours

**Viajero 1**  
Leandro Alberto Valencia Escobar  
24/03/1999 +8801 4454 4544 Asiento D18

Puedes cancelar

Mansurul Haque

## Reclamos

The screenshot shows the Bookler mobile application interface. On the left is a vertical navigation sidebar with a blue background, featuring icons for 'Inicio' (Home), 'Buscar vuelos' (Search flights), 'Mis viajes' (My trips), and 'Reclamos' (Claims). The 'Reclamos' icon is highlighted with a blue background. The main content area has a dark header with a user profile picture and the name 'Mansurul Haque'. Below the header is a button labeled '+ Nuevo reclamo'. The main section is titled 'Reclamos' and displays a table of claims with columns: Código reserva/vuelo, Ruta, Fecha, Tarifa, Estado, and Acción. There are three entries in the table:

Código reserva/vuelo	Ruta	Fecha	Tarifa	Estado	Acción
CN-KL2345	Frankfurt to Bangkok	July 20, 2028	\$500.00	Resuelto	Ver detalles
CN-KL2345	Frankfurt to Bangkok	July 20, 2028	\$500.00	En proceso	Ver detalles
CN-KL2345	Frankfurt to Bangkok	July 20, 2028	\$500.00	Rechazado	Ver detalles

## Modales

A modal window titled 'Nuevo reclamo' (New claim) is displayed. At the top right is a close button (X). The form contains the following fields:

- A required field labeled 'Seleccionar vuelo/reserva \*' with a dropdown menu showing 'Selected item'.
- A text input field labeled 'Descripción' with a placeholder 'Hint text'.

## Reclamo CN-KL2345 (Frankfurt to Bangkok)

Commodo eget a et dignissim dignissim morbi vitae, mi. Mi aliquam sit ultrices enim cursus. Leo sapien, pretium duis est eu volutpat interdum eu non. Odio eget nullam elit laoreet. Libero at felis nam at orci venenatis rutrum nunc. Etiam mattis ornare pellentesque iaculis enim.

Felis eu non in aliquam egestas placerat. Eget maecenas ornare venenatis lacus nunc, sit arcu. Nam pharetra faucibus eget facilisis pulvinar eu sapien turpis at. Nec aliquam aliquam blandit eu ipsum.

Cerrar

## Panel de administración

**Booker**

Mansurul Haque

### Dashboard

Número de reservas 125	Cancelaciones 25	Número de aviones 92	Ingresos totales \$15,000
Número de vuelos 67	Reclamos 12	Número de tripulantes 456	Aerolíneas 23

### Reclamos

Código reserva/vuelo	Ruta	Fecha	Cliente	Estado	Opción
CN-KL2345	Frankfurt to Bangkok	July 20, 2028	Rodrigo Daniel	Resuelto	Ver detalles
CN-KL2345	Frankfurt to Bangkok	July 20, 2028	Leandro Valencia	En proceso	Ver detalles

**Reservaciones**

Todas    Confirmadas    Canceladas

Flight No	Aerolinea	Ruta	Salida	Llegada	Fecha	Pasajero	Estado
CN-KL2345	CloudNine Airlines	CDG-JFK	9:00 AM	12:00 PM	2028-07-01	Rodrigo Pineda	Confirmed
QW-MN6789	QuickWing Air	HKG-LAX	11:00 PM	8:00 AM	2028-07-01	Rodrigo Pineda	Cancelled
SH-OP3456	SkyHigh Airlines	FRA-BKK	7:00 AM	3:00 PM	2028-07-01	Rodrigo Pineda	Confirmed
FF-QR7890	FlyFast Airways	LAX-HND	5:00 PM	10:00 PM	2028-07-01	Rodrigo Pineda	Cancelled
AJ-ST0123	AeroJet	SIN-LHR	6:00 AM	12:00 PM	2028-07-01	Rodrigo Pineda	Confirmed
SH-AB1234	SkyHigh Airlines	NYC-LAX	10:00 AM	1:00 PM	2028-07-01	Rodrigo Pineda	Confirmed
FF-CD5678	FlyFast Airways	IHR-JFK	8:00 AM	11:00 AM	2028-07-01	Rodrigo Pineda	Confirmed
AJ-EF9101	AeroJet	HND-SFO	2:00 PM	8:00 AM	2028-07-01	Rodrigo Pineda	Confirmed
NA-GH2345	Nimbus Airlines	SYD-SIN	6:00 PM	12:00 AM	2028-07-01	Rodrigo Pineda	Cancelled
JS-U6789	JetStream Aviation	DXB-LHR	11:00 PM	6:00 AM	2028-07-01	Rodrigo Pineda	Confirmed
CN-FG2987	CloudNine Airlines	HND-DXB	8:00 AM	5:00 PM	2028-07-01	Rodrigo Pineda	Confirmed
NA-VW3456	Nimbus Airlines	SIN-JFK	11:00 AM	10:00 PM	2028-07-01	Rodrigo Pineda	Cancelled
AJ-MN8201	AeroJet	LAX-SFO	6:00 PM	7:00 AM	2028-07-01	Rodrigo Pineda	Cancelled
QW-PQ4567	QuickWing Air	IHR-JFK	5:30 AM	3:00 PM	2028-07-01	Rodrigo Pineda	Confirmed

Showing 1-14 of 567

Prev 1 2 3 ... 8 Next

**Reservaciones**

Regresar a Reservaciones  
Los Angeles → New York  
Boeing 787 Dreamliner \*

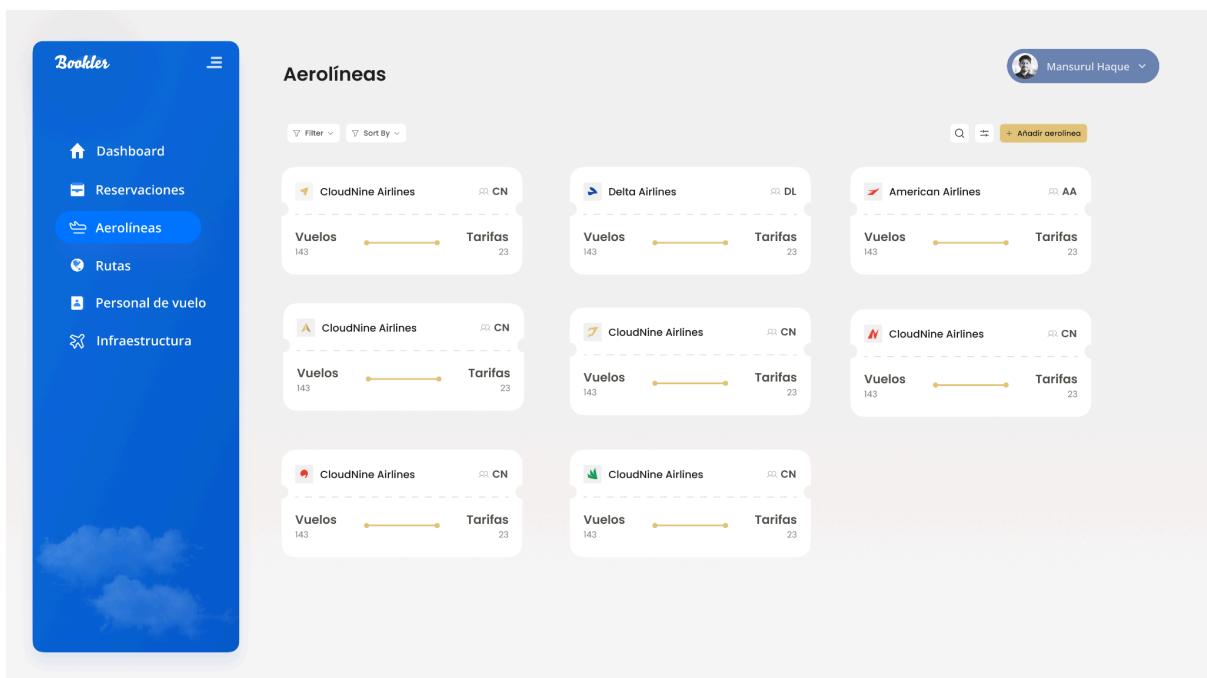
**SkyHigh Airlines**  
SH-ZY6789 Cancelado

Fecha July 1, 2028 Pago Tarjeta Débito \$350 /pax **1 reclamo** !

6:00 AM 15 JUL 2028 • Los Angeles  
Los Angeles International Airport (LAX)

Viajero 1  
Leandro Alberto Valencia Escobar  
24/03/1999  
+8801 4454 4544 Asiento D16

Viajero 2



**Aerolíneas**

This screenshot shows the Aerolíneas section of the Booker app. It displays a grid of airline cards. Each card includes the airline's logo, name, and two metrics: 'Vuelos' (Flights) and 'Tarifas' (Fares). The cards are arranged in three rows and four columns.

Airline	Vuelos	Tarifas
CloudNine Airlines (CN)	143	23
Delta Airlines (DL)	143	23
American Airlines (AA)	143	23
CloudNine Airlines (CN)	143	23
CloudNine Airlines (CN)	143	23
CloudNine Airlines (CN)	143	23
CloudNine Airlines (CN)	143	23
CloudNine Airlines (CN)	143	23

**Rutas**

This screenshot shows the Rutas section of the Booker app. It displays a grid of route cards. Each card shows a flight path from one airport to another, including the departure and arrival cities, IATA codes, duration, and an 'Editar ruta' (Edit route) button.

Route	Duration	Action
London (LHR) to Oslo (OSL)	14 hours	Editar ruta
Dublin (DUB) to Tokyo (NRT)	14 hours	Editar ruta
Brisbane (BNE) to Buenos A. (EZE)	14 hours	Editar ruta
Cairo (CAI) to Vienna (VIE)	14 hours	Editar ruta
London (LHR) to Oslo (OSL)	14 hours	Editar ruta
Dublin (DUB) to Tokyo (NRT)	14 hours	Editar ruta

**Booker**

- Dashboard
- Reservaciones
- Aerolineas
- Rutas**
- Personal de vuelo
- Infraestructura



## Rutas

Filter Sort By

Search: + Añadir ruta

Aeropuerto	Location	Aeropuerto	Location			
Heathrow Airport	London UK	LHR	Oslo Norway	Duration: 14 hours	OSL	<a href="#">Editor ruta</a>
Dublin Airport	Dublin Ireland	DUB	Tokyo Japan	Duration: 14 hours	NRT	<a href="#">Editor ruta</a>
Brisbane Airport	Brisbane Australia	BNE	Buenos A. Argentina	Duration: 14 hours	EZE	<a href="#">Editor ruta</a>
Cairo Airport	Cairo	CAI	Vienna Austria	Duration: 14 hours	VIE	<a href="#">Editor ruta</a>
Heathrow Airport	London UK	UHR	Oslo Norway	Duration: 14 hours	OSL	<a href="#">Editor ruta</a>
Dublin Airport	Dublin Ireland	DUB	Tokyo Japan	Duration: 14 hours	NRT	<a href="#">Editor ruta</a>

## 5.2. Gestión de la Base de Datos

### a) Estructura de la Base de Datos: Modelo Relacional

La base de datos propuesta para el sistema de gestión de vuelos está diseñada bajo un modelo relacional, donde se representan las entidades principales del dominio, así como sus relaciones. A continuación, se detalla la estructura de las tablas y claves:

#### Tabla: USUARIOS

- **id\_usuario** (PK) – Identificador único del usuario.
- **nombre** – Nombre del usuario.
- **correo** – Correo electrónico (único).
- **clave\_hash** – Contraseña encriptada.
- **rol** – Define el rol (Administrador, Cliente, Agente).

#### Tabla: AEROLINEAS

- **id\_aerolinea** (PK).
- **nombre**.
- **codigo\_iata** (único, 2-3 caracteres).

#### Tabla: AEROPUERTOS

- **id\_aeropuerto** (PK).
- **codigo\_iata** (único, 3 caracteres).
- **nombre**.
- **ciudad**.
- **pais**.

#### Tabla: RUTAS

- **id\_ruta** (PK).
- **origen\_id** (FK → AEROPUERTOS.id\_aeropuerto).

- **destino\_id** (FK → AEROPUERTOS.id\_aeropuerto).
- **distancia\_km**.

Tabla: AVIONES

- **id\_avion** (PK).
- **matricula** (única).
- **modelo**.
- **capacidad\_total**.

Tabla: CLASES

- **id\_clase** (PK).
- **nombre** (Económica, Business, Primera).
- **descripcion**.

Tabla: ASIENTOS\_AVION

- **id\_asiento** (PK).
- **avion\_id** (FK → AVIONES.id\_avion).
- **codigo\_asiento** (Ej: 12A, 14B).
- **clase\_id** (FK → CLASES.id\_clase).

Tabla: TRIPULANTES

- **id\_tripulante** (PK).
- **nombre**.
- **tipo** (Piloto, Copiloto, Azafata).

Tabla: VUELO

- **id\_vuelo** (PK).
- **numero\_vuelo** (único, ej: AV123).

- **aerolinea\_id** (FK → AEROLINEAS.id\_aerolinea).
- **ruta\_id** (FK → RUTAS.id\_ruta).
- **duracion\_min.**

Tabla: OPERACION\_VUELO

- **id\_operacion** (PK).
- **vuelo\_id** (FK → VUELO.id\_vuelo).
- **avion\_id** (FK → AVIONES.id\_avion).
- **fecha\_salida.**
- **fecha\_llegada.**
- **estado** (Programado, En curso, Finalizado, Cancelado).

Tabla: OPERACION\_TRIPULACION

- **id\_operacion\_trip** (PK).
- **operacion\_id** (FK → OPERACION\_VUELO.id\_operacion).
- **tripulante\_id** (FK → TRIPULANTES.id\_tripulante).
- **rol\_en\_vuelo** (Piloto principal, Azafata).

Tabla: TARIFA

- **id\_tarifa** (PK).
- **codigo** (Ej: ECO, BUS).
- **clase\_id** (FK → CLASES.id\_clase).
- **reembolsable** (boolean).

Tabla: TARIFA\_OPERACION

- **id\_tarifa\_oper** (PK).
- **operacion\_id** (FK → OPERACION\_VUELO.id\_operacion).

- **tarifa\_id** (FK → TARIFA.id\_tarifa).
- **precio.**
- **asientos\_disponibles.**

Tabla: PASAJEROS

- **id\_pasajero** (PK).
- **nombre\_completo.**
- **fecha\_nacimiento.**
- **nro\_pasaporte** (único).

Tabla: RESERVAS

- **id\_reserva** (PK).
- **codigo\_reserva** (único).
- **usuario\_id** (FK → USUARIOS.id\_usuario).
- **fecha\_reserva.**
- **estado** (Activa, Cancelada, Finalizada).
- **total.**

Tabla: RESERVA\_ASIENTO

- **id\_reserva\_asiento** (PK).
- **reserva\_id** (FK → RESERVAS.id\_reserva).
- **pasajero\_id** (FK → PASAJEROS.id\_pasajero).
- **operacion\_id** (FK → OPERACION\_VUELO.id\_operacion).
- **asiento\_avion\_id** (FK → ASIENTOS\_AVION.id\_asiento).
- **tarifa\_oper\_id** (FK → TARIFA\_OPERACION.id\_tarifa\_oper).
- **precio\_pagado.**

Tabla: PAGOS

- **id\_pago** (PK).
- **reserva\_id** (FK → RESERVAS.id\_reserva).
- **fecha\_pago**.
- **metodo\_pago** (Tarjeta, Transferencia).
- **monto**.

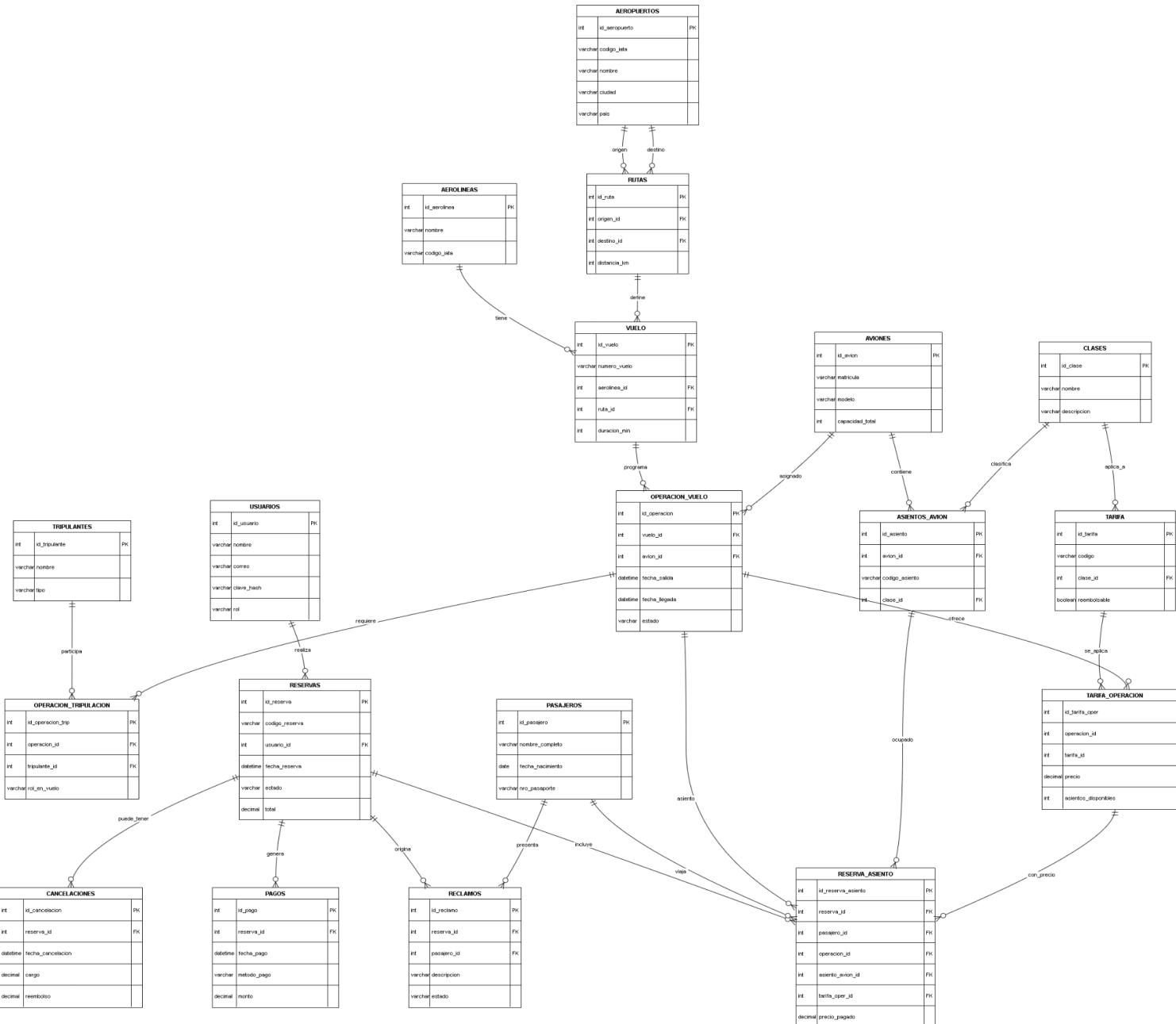
Tabla: CANCELACIONES

- **id\_cancelacion** (PK).
- **reserva\_id** (FK → RESERVAS.id\_reserva).
- **fecha\_cancelacion**.
- **cargo**.
- **reembolso**.

Tabla: RECLAMOS

- **id\_reclamo** (PK).
- **reserva\_id** (FK → RESERVAS.id\_reserva).
- **pasajero\_id** (FK → PASAJEROS.id\_pasajero).
- **descripcion**.
- **estado** (Pendiente, En revisión, Resuelto).

## Modelo Relacional



b) Diccionario de datos

1	USUARIOS	id_usuario	INT	Clave primaria que identifica a cada usuario.	-	Números enteros	PK, único, no nulo	-
		nombre	VARCHAR	Nombre completo del usuario.	-	Texto	No nulo	-
		correo	VARCHAR	Dirección de correo electrónico del usuario.	-	Formato de correo válido	No nulo, único	-
		clave_hash	VARCHAR	Contraseña del usuario cifrada.	-	Texto (hash)	No nulo	-
		rol	VARCHAR	Rol o tipo de usuario (ej. cliente, admin).	-	(Administrador, Cliente, Agente)	No nulo	-
2	AEROLINEAS	Tipo de Dato						Restricciones
		Campo	Dato	Descripción	Tamaño	Valores Permitidos		Relaciones
		id_aeropuerto	INT	Clave primaria para cada aeropuerto.	-	Números enteros	PK, único, no nulo	-
		codigo_iata	VARCHAR	Código IATA de 3 letras del aeropuerto.	3	Código de tres letras	No nulo, único	-
		nombre	VARCHAR	Nombre completo del aeropuerto.	-	Texto	No nulo	-
3	AEROPUERTOS	ciudad	VARCHAR	Ciudad donde se ubica el aeropuerto.	-	Texto	No nulo	-
		Tipo de Dato						Restricciones
		Campo	Dato	Descripción	Tamaño	Valores Permitidos		Relaciones
		id_ciudad	INT	Clave primaria para cada ciudad.	-	Números enteros	PK, único, no nulo	-
		nombre_ciudad	VARCHAR	Nombre completo de la ciudad.	-	Texto	No nulo	-

		pais	VARCHAR	País donde se ubica el aeropuerto.	-	Texto	No nulo	-
3 AEROPUERTOS	Tipo de Dato		Descripción		Tamaño	Valores Permitidos		Restricciones
	Campo	INT	Clave primaria para cada aeropuerto.	-	Números enteros		PK, único, no nulo	-
	codigo_iata	VARCHAR	Código IATA de 3 letras del aeropuerto.	3	Código de tres letras		No nulo, único	-
	nombre	VARCHAR	Nombre completo del aeropuerto.	-	Texto		No nulo	-
	ciudad	VARCHAR	Ciudad donde se ubica el aeropuerto.	-	Texto		No nulo	-
	pais	VARCHAR	País donde se ubica el aeropuerto.	-	Texto		No nulo	-
4 RUTAS	Tipo de Dato		Descripción		Tamaño	Valores Permitidos		Restricciones
	Campo	INT	Clave primaria que identifica una ruta de vuelo.	-	Números enteros		PK, único, no nulo	-
	origen_id	INT	Referencia al aeropuerto de origen.	-	AEROPUERTOS.id_aeropuerto		FK, no nulo	AEROPUERTOS (relación origen)

		destino_id	INT	Referencia al aeropuerto de destino.	-	AEROPUERTOS.id_aeropuerto	FK, no nulo	AEROPUERTOS (relación destino)
		distancia_km	INT	Distancia entre los aeropuertos en kilómetros.	-	Números enteros	-	-
5	VUELO	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_vuelo	INT	Clave primaria para cada vuelo programado.	-	Números enteros	PK, único, no nulo	-
		numero_vuelo	VARCHAR	Número único que identifica el vuelo.	-	Texto	No nulo	-
		aerolinea_id	INT	Referencia a la aerolínea que opera el vuelo.	-	AEROLINEAS.id_aerolinea	FK, no nulo	AEROLINEAS
		ruta_id	INT	Referencia a la ruta del vuelo.	-	RUTAS.id_ruta	FK, no nulo	RUTAS
		duracion_min	INT	Duración estimada del vuelo en minutos.	-	Números enteros	No nulo	-
6	OPERACION_VUELO	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones

		<b>id_operacion</b>	INT	Clave primaria para una instancia específica de un vuelo.	-	Números enteros	PK, único, no nulo	-
		<b>vuelo_id</b>	INT	Referencia al vuelo programado.	-	VUELO.id_vuelo	FK, no nulo	VUELO
		<b>avion_id</b>	INT	Referencia al avión asignado a la operación.	-	AVIONES.id_avion	FK, no nulo	AVIONES
		<b>fecha_salida</b>	DATETIME	Fecha y hora de salida programada/real.	-	Fecha y hora	No nulo	-
		<b>fecha_llegada</b>	DATETIME	Fecha y hora de llegada programada/real.	-	Fecha y hora	No nulo	-
		<b>estado</b>	VARCHAR	Estado actual de la operación de vuelo (ej. a tiempo, retrasado).	-	(Programado, En curso, Finalizado, Cancelado)	No nulo	-
7 ON OPERACION _TRIPULACI	<b>Tipo de Campo</b>	<b>Dato</b>	<b>Descripción</b>	<b>Tamaño</b>	<b>Valores Permitidos</b>	<b>Restriccione s</b>	<b>Relaciones</b>	
	<b>id_operacion_ trip</b>	INT	Clave primaria para la asignación de tripulación.	-	Números enteros	PK, único, no nulo	-	
	<b>operacion_id</b>	INT	Referencia a la operación de vuelo.	-	OPERACION_VUELO.id_operacion	FK, no nulo	OPERACION_VUELO	
	<b>tripulante_id</b>	INT	Referencia al tripulante asignado.	-	TRIPULANTES.id_tripulante	FK, no nulo	TRIPULANTES	

		rol_en_vuelo	VARCHAR	Rol del tripulante en este vuelo (ej. Capitán).	-	Texto	No nulo	-
8	TARIFA	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_tarifa	INT	Clave primaria para cada tipo de tarifa.	-	Números enteros	PK, único, no nulo	-
		codigo	VARCHAR	Código de la tarifa (ej. Y, J).	-	Texto	No nulo, único	-
		clase_id	INT	Referencia a la clase a la que se aplica la tarifa.	-	CLASES.id_clase	FK, no nulo	CLASES
		reembolsable	BOOLEAN	Indica si la tarifa es reembolsable.	-	TRUE, FALSE	No nulo	-
9	TARIFA_OPE RATION	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_tarifa_oper	INT	Clave primaria para una tarifa específica de una operación de vuelo.	-	Números enteros	PK, único, no nulo	-
		operacion_id	INT	Referencia a la operación de vuelo.	-	OPERACION_VUELO.id_operacion	FK, no nulo	OPERACION_VUELO
		tarifa_id	INT	Referencia al tipo de tarifa.	-	TARIFA.id_tarifa	FK, no nulo	TARIFA

		precio	DECIMAL	Precio de la tarifa para esta operación.	-	Números decimales	No nulo	-
		asientos_disponibles	INT	Cantidad de asientos disponibles para esta tarifa.	-	Números enteros	No nulo	-
10	PASAJEROS	Tipo de Campo						
		Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones	
		INT	Clave primaria para cada pasajero.	-	Números enteros	PK, único, no nulo	-	
		VARCHAR	Nombre completo del pasajero.	-	Texto	No nulo	-	
		DATE	Fecha de nacimiento del pasajero.	-	Fecha	No nulo	-	
		VARCHAR	Número de pasaporte del pasajero.	-	Texto	No nulo, único	-	
11	RESERVAS	Tipo de Campo						
		Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones	
		INT	Clave primaria para cada reserva.	-	Números enteros	PK, único, no nulo	-	
		VARCHAR	Código único de la reserva.	-	Texto	No nulo, único	-	

		usuario_id	INT	Referencia al usuario que realiza la reserva.	-	USUARIOS.id_usuario	FK, no nulo	USUARIOS
		fecha_reserva	DATETIME	Fecha y hora en que se hizo la reserva.	-	Fecha y hora	No nulo	-
		estado	VARCHAR	Estado de la reserva (ej. confirmada, cancelada).	-	(Activa, Cancelada, Finalizada)	No nulo	-
		total	DECIMAL	Monto total de la reserva.	-	Números decimales	No nulo	-
RESERVA_ASIENTO	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones	
12	id_asiento	INT	Clave primaria para cada asiento reservado.	-	Números enteros	PK, único, no nulo	-	
	reserva_id	INT	Referencia a la reserva a la que pertenece el asiento.	-	RESERVAS.id_reserva	FK, no nulo	RESERVAS	
	pasajero_id	INT	Referencia al pasajero que ocupa el asiento.	-	PASAJEROS.id_pasajero	FK, no nulo	PASAJEROS	
	operacion_id	INT	Referencia a la operación de vuelo.	-	OPERACION_VUELO.id_operacion	FK, no nulo	OPERACION_VUELO	
	asiento_avion_id	INT	Referencia al asiento físico del avión.	-	ASIENTOS_AVION.id_asiento	FK, no nulo	ASIENTOS_AVION	
	tarifa_oper_id	INT	Referencia a la tarifa aplicada.	-	TARIFA_OPERACION.id_tarifa_oper	FK, no nulo	TARIFA_OPERACION	

		precio_pagado	DECIMAL	Precio pagado por este asiento.	-	Números decimales	No nulo	-
13	PAGOS	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_pago	INT	Clave primaria para cada pago.	-	Números enteros	PK, único, no nulo	-
		reserva_id	INT	Referencia a la reserva a la que corresponde el pago.	-	RESERVAS.id_reserva	FK, no nulo	RESERVAS
		fecha_pago	DATETIME	Fecha y hora en que se realizó el pago.	-	Fecha y hora	No nulo	-
		metodo_pago	VARCHAR	Método de pago utilizado (ej. Tarjeta de crédito).	-	(Tarjeta, Transferencia)	No nulo	-
		monto	DECIMAL	Monto del pago.	-	Números decimales	No nulo	-
14	CANCELACIONES	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_cancelacion	INT	Clave primaria para cada cancelación.	-	Números enteros	PK, único, no nulo	-
		reserva_id	INT	Referencia a la reserva cancelada.	-	RESERVAS.id_reserva	FK, no nulo	RESERVAS

		fecha_cancelacion	DATETIME	Fecha y hora de la cancelación.	-	Fecha y hora	No nulo	-
		cargo	DECIMAL	Monto del cargo por cancelación.	-	Números decimales	-	-
		reembolso	DECIMAL	Monto del reembolso al cliente.	-	Números decimales	-	-
15	RECLAMOS	Campo	Tipo de Dato	Descripción	Tamaño	Valores Permitidos	Restricciones	Relaciones
		id_reclamo	INT	Clave primaria para cada reclamo.	-	Números enteros	PK, único, no nulo	-
		reserva_id	INT	Referencia a la reserva relacionada con el reclamo.	-	RESERVAS.id_reserva	FK, no nulo	RESERVAS
		pasajero_id	INT	Referencia al pasajero que presenta el reclamo.	-	PASAJEROS.id_pasajero	FK, no nulo	PASAJEROS
		descripcion	VARCHAR	Descripción detallada del reclamo.	-	Texto	No nulo	-
		estado	VARCHAR	Estado del reclamo (ej. abierto, en revisión, cerrado).	-	Texto	No nulo	-

		Tipo de Campo Dato Descripción Tamaño Valores Permitidos Restriccione s Relaciones					
16	AVIONES						
		id_avion	INT	Clave primaria para cada avión.	-	Números enteros	PK, único, no nulo -
		matricula	VARCHAR	Identificador único del avión.	-	Texto	No nulo, único -
		modelo	VARCHAR	Modelo del avión (ej. Boeing 737).	-	Texto	No nulo -
		capacidad_total	INT	Número total de asientos en el avión.	-	Números enteros	No nulo -
17	CLASES	Tipo de Campo Dato Descripción Tamaño Valores Permitidos Restriccione s Relaciones					
17	CLASES						
		id_clase	INT	Clave primaria que identifica una clase de asiento.	-	Números enteros	PK, único, no nulo -
		nombre	VARCHAR	Nombre de la clase (ej. Economía, Negocios).	-	(Económica, Business, Primera)	No nulo -
		descripcion	VARCHAR	Descripción detallada de los servicios de la clase.	-	Texto	- -
18	ASIENTOS_AVION	Tipo de Campo Dato Descripción Tamaño Valores Permitidos Restriccione s Relaciones					
18	ASIENTOS_AVION						

		<b>id_asiento</b>	INT	Clave primaria para cada asiento de un avión.	-	Números enteros	PK, único, no nulo	-
		<b>avion_id</b>	INT	Referencia al avión al que pertenece el asiento.	-	AVIONES.id_avion	FK, no nulo	AVIONES
		<b>codigo_asiento</b>	VARCHAR	Código del asiento (ej. 14A, 25B).	-	Texto	No nulo	-
		<b>clase_id</b>	INT	Referencia a la clase a la que pertenece el asiento.	-	CLASES.id_clase	FK, no nulo	CLASES
<b>19 S</b>	<b>TRIPULANTE</b>	<b>Tipo de Dato</b>					<b>Restricciones</b>	
		<b>Campo</b>		<b>Descripción</b>	<b>Tamaño</b>	<b>Valores Permitidos</b>		<b>Relaciones</b>
		<b>id_tripulante</b>	INT	Clave primaria para cada miembro de la tripulación.	-	Números enteros	PK, único, no nulo	-
		<b>nombre</b>	VARCHAR	Nombre completo del tripulante.	-	Texto	No nulo	-
		<b>tipo</b>	VARCHAR	Tipo de tripulante (ej. Piloto, Copiloto, Azafata).	-	(Piloto, Copiloto, Azafata)	No nulo	-

## 6. Resultados y Conclusiones

La primera fase del proyecto de gestión de boletos aéreos se centró en la **planificación, documentación y diseño**, estableciendo los cimientos necesarios para el desarrollo del sistema. A continuación, se presentan los principales resultados alcanzados y las conclusiones derivadas del proceso.

### a) Resultados obtenidos

#### **Diseño de la Base de Datos:**

Se definió un modelo relacional completo que respalda los procesos de gestión de aerolíneas, vuelos, pasajeros, reservaciones y pagos. La propuesta incluye la definición de tablas, claves primarias y foráneas, tipos de datos y restricciones, quedando lista para su implementación en fases posteriores.

#### **Prototipos de Interfaz de Usuario:**

Se diseñaron prototipos iniciales de la interfaz, que permiten visualizar el flujo de interacción del usuario con el sistema. Estos modelos serán la guía para la construcción del frontend en React con Tailwind.

#### **Planificación del Proyecto:**

Se elaboró un plan integral que abarca alcance, cronograma, recursos humanos, costos, calidad y comunicación. Además, se definieron roles y responsabilidades específicas para cada integrante, lo que contribuyó a una mejor organización y coordinación del equipo.

### b) Conclusiones

La experiencia en esta primera etapa resultó enriquecedora para el equipo, no solo por los productos generados, sino también por el proceso de trabajo colaborativo. La aplicación de **Design Thinking** permitió orientar el diseño hacia las necesidades reales de los usuarios, promoviendo la creatividad y la iteración de ideas. Asimismo, el uso de herramientas colaborativas como GitHub, Google Docs y Sheets facilitó la trazabilidad y organización de las tareas.

#### **Desafíos enfrentados y aprendizajes:**

Uno de los retos más importantes fue delimitar con precisión el alcance del proyecto en sus primeras fases. Esto requirió mantener una comunicación constante entre los integrantes y afinar los criterios técnicos y funcionales. De esta experiencia se aprendió la relevancia de una planificación cuidadosa para evitar retrabajos en etapas posteriores. La asignación clara de funciones por áreas (backend, frontend, base de datos, documentación) contribuyó a la eficiencia y calidad del trabajo realizado.

En síntesis, el trabajo realizado en esta primera etapa no solo permitió generar entregables concretos, sino que también consolidó las bases metodológicas y organizativas necesarias para el desarrollo del sistema. Con estos cimientos, el equipo se encuentra en condiciones de avanzar con mayor confianza hacia la implementación y validación del proyecto.

## 7. Anexos y Documentación de Apoyo

Con el fin de complementar la fase de planificación y diseño del sistema de gestión de boletos aéreos, se incluyen a continuación configuraciones avanzadas, ejemplos de código y documentación técnica adicional que servirán como referencia para la futura implementación del sistema.

### 7.1. Configuraciones Avanzadas

#### Entorno de Desarrollo:

- **Backend:** Java 17 + Spring Boot 3.5.5.
- **Frontend:** React 18 con TailwindCSS.
- **Base de Datos:** MySQL 8.0.

**Gestión de Versiones:** Cada miembro del equipo debe trabajar en su propia rama de GitHub y enviar las actualizaciones a través de un "pull request" a la rama principal ("main").

#### Buenas Prácticas de Seguridad:

- Implementación de JWT para la autenticación de usuarios.
- Manejo de errores y excepciones con controladores globales (@ControllerAdvice).
- Sanitización de datos en entradas SQL para prevenir inyección.

### 7.2. Ejemplos de Código

#### Ejemplo de Endpoint REST (Spring Boot):

```
@RestController
@RequestMapping("/api/vuelos")
public class VueloController {

    @Autowired
    private VueloService vueloService;

    @GetMapping("/{id}")
    public ResponseEntity<Vuelo> getVueloById(@PathVariable Long id) {
        Vuelo vuelo = vueloService.findById(id);
        return ResponseEntity.ok(vuelo);
    }

    @PostMapping
    public ResponseEntity<Vuelo> createVuelo(@RequestBody Vuelo vuelo) {
        Vuelo nuevoVuelo = vueloService.save(vuelo);
        return ResponseEntity.status(HttpStatus.CREATED).body(nuevoVuelo);
    }
}
```

Este fragmento ejemplifica cómo se manejarán las operaciones básicas de consulta y registro de vuelos en la API.

#### Ejemplo de Consulta SQL:

```
-- Consulta de vuelos disponibles con sus aerolíneas y rutas
SELECT v.numero_vuelo, a.nombre AS aerolinea, r.origen_id, r.destino_id, op.fecha_salida, op.fecha_llegada
FROM VUELO v
JOIN AEROLINEAS a ON v.aerolinea_id = a.id_aerolinea
JOIN RUTAS r ON v.ruta_id = r.id_ruta
JOIN OPERACION_VUELO op ON v.id_vuelo = op.vuelo_id
WHERE op.estado = 'Programado';
```

#### Ejemplo de Prueba Unitaria con JUnit y MockMvc:

```
import org.junit.jupiter.api.Test;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.context.SpringBootTest;

import static org.junit.jupiter.api.Assertions.*;

@SpringBootTest
class VueloServiceTest {

    @Autowired
    private VueloService vueloService;

    @Test
    void testCalcularDuracionVuelo() {
        int duracion = vueloService.calcularDuracion(14, 16);
        assertEquals(120, duracion, "La duración del vuelo debe ser de 120 minutos");
    }
}
```

## 7.2. Documentación Técnica Adicional

**Modelo Relacional** (ver *diagrama en la página 31 del documento*): Representa de forma integral la interconexión de las tablas principales: usuarios, vuelos, reservas, pagos y reclamos. Esto asegura consistencia en la gestión de datos y facilita la implementación.

**Diccionario de Datos** (páginas 32–42): Proporciona una descripción detallada de cada campo, tipo de dato, restricciones y relaciones. Este material resulta esencial como referencia en la programación de la capa de persistencia.

#### Herramientas de Apoyo:

- **Postman:** Para pruebas de endpoints y validación de la API.
- **Figma:** Utilizado en la construcción de los prototipos del frontend, documentados en las páginas 12–24.
- **PhpMyAdmin:** Para gestión visual de la base de datos durante las primeras fases de desarrollo.