

**PT-PP-01. PLANTEAMIENTO DEL PROBLEMA Y PLAN DE PROYECTO**

**Booking Plane**

**Versión 1.0**

**Historial de revisión**

| **Versión** | **Fecha elaboración** | **Responsable Elaboración** | **Fecha Aprobación** | **Responsable Aprobación** |
| --- | --- | --- | --- | --- |
| **1.0** | **21/10/2025** | **Andres Cebay, Christian de Jesus** | **Pendiente** | **Por definir** |
|  |  |  |  |  |

**Cambios frente a la última versión**

| **Versión** | **Modificación respecto a versión anterior** |
| --- | --- |
| **1.0** | **Creación del documento** |

1. **Nombre del proyecto**

**Booking Plane** – Sistema de Gestión de reservas aéreas

1. **Descripción general del sistema**

Booking plane es un sistema desarrollado para gestionar de forma eficiente las reservas de vuelos y administración de vuelos en aeropuertos. El sistema permitirá que los clientes consulten vuelos disponibles, reserven asientos y gestionan sus pagos, mientras que los administradores podrán gestionar vuelos y aviones.

1. **Problema o necesidad**

A pesar de la alta demanda de transacciones en el sector aéreo, la problemática principal en los sistemas actuales de venta de tiquetes se centra en dos áreas críticas: la integridad de los datos de reserva y la eficiencia del proceso de compra. Los sistemas que no están diseñados con mecanismos robustos de control de versiones y transacciones en tiempo real corren un riesgo crítico de sobrevender (*overbooking*) el inventario. Esto ocurre cuando dos o más usuarios intentan reservar simultáneamente el mismo asiento, lo que resulta en una violación de la capacidad física del vuelo y genera problemas logísticos, multas y una severa pérdida de credibilidad.

1. **Objetivo**

Desarrollar un sistema web de reservas de vuelos que permita la gestión eficiente de usuarios, reservar, vuelos y pagos dentro de un entorno web estructurado con Laravel.

1. **Alcance**

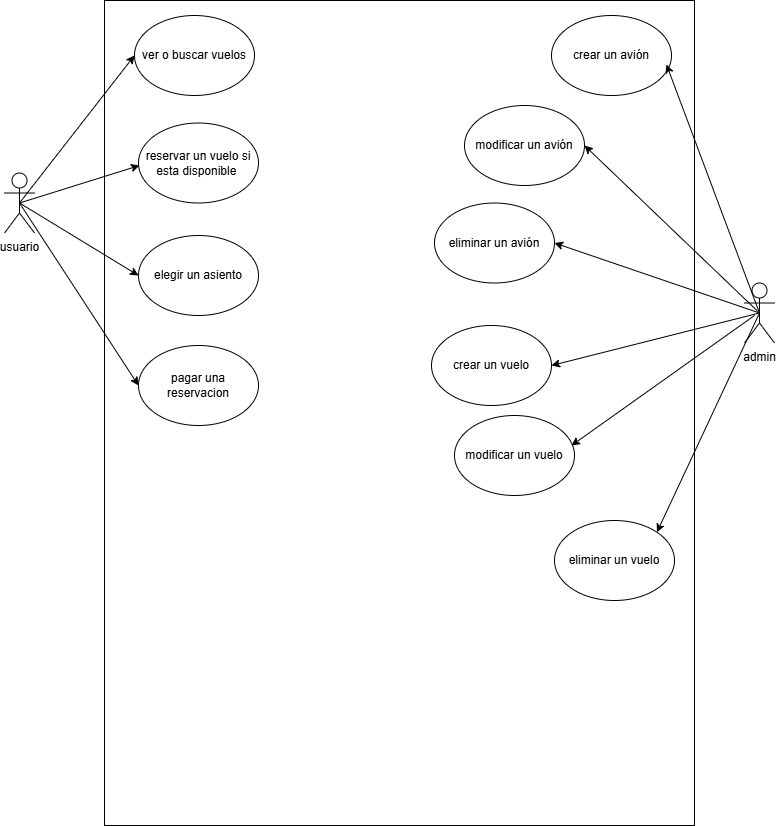
El sistema incluirá en esta primera versión:

* Módulo de reservas con selección de vuelo y asiento.
* Módulo administrativo con gestión de vuelos, usuarios y reportes.
* Validaciones de datos y control en autenticación
* Diseño responsivo con Tailwind CSS

1. **Características de usuario**

| **Rol** | **Descripción** | **Funcionalidades** |
| --- | --- | --- |
| Administrador | Usuario encargado de la gestión completa del sistema. | Crear, editar, y eliminar vuelos y aviones |
| Cliente | Usuario que realiza las reservas | Consulta vuelos, reserva asientos, y realiza pagos |

1. **Diagramas de casos de uso**

****

1. **Tecnologías a usar**

**Backend:** Flask(Python)

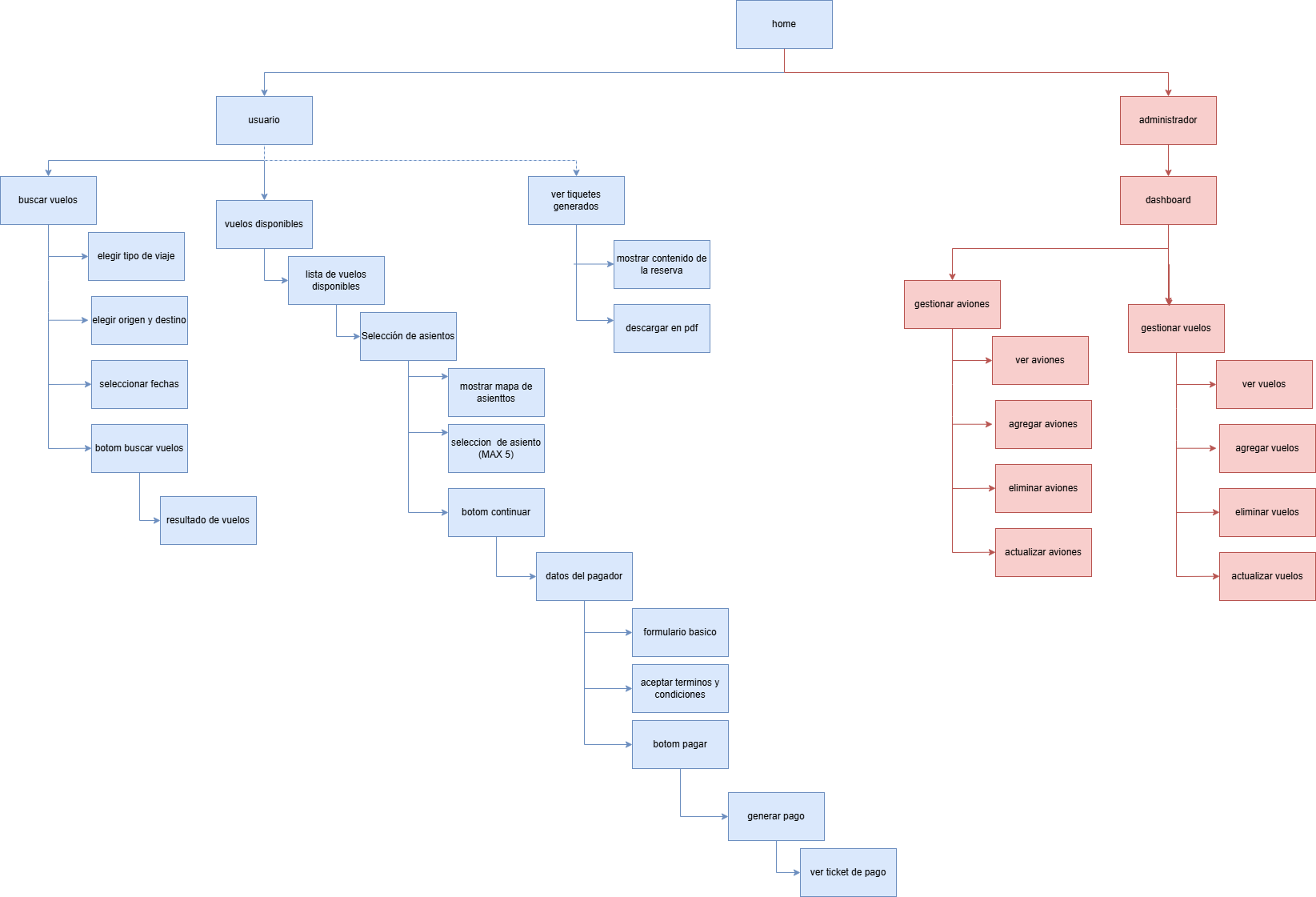
Se eligió Fask ya que es ligero, permite la modularidad, Permite construir APIs limpias y orientadas a microservicios, ideal para este sistema que es tan demandante. A diferencia de frameworks más pesados, Flask solo incorpora lo necesario, optimizando recursos y tiempos de respuesta.

**Fronted:** Construido con HTML5 y Tailwind CSS para obtener una interfaz limpia y rápida sin sobrecargar al navegador**.** Además que este permite interfaces responsivas.

**Base de datos:** Para garantizar integridad de datos se seleccionó MySQL

**Control de versiones:** Github se integra como herramienta de control de versiones y colaboración, Permitiéndonos gestionar cambios y versiones de una manera más sencilla.

1. **Mapa de navegación**

****

1. **Requerimientos**
   1. **Requerimientos funcionales**

|  | | **SENA**  **SISTEMA PARA COMPRA DE TIQUETES AÉREOS No 001**    **No 001** |  |
| --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Prioridad** |
| **0** | **iniciar sesion** | permite a los visitantes de la página iniciar sesión de modo que tendrán funciones adicionales. como ver el historial de las reservas. | **Media** |
| **1** | **Búsqueda de vuelos** | Permitir la búsqueda de vuelos según origen, destino, fechas válidas (día actual hasta 2 meses adelante) y tipo de viaje (ida o ida y regreso). | **Alta** |
| **2** | **Selección de vuelo y asientos** | Mostrar vuelos disponibles y controlar la capacidad de cada avión, evitando que se reserven asientos duplicados | **Alta** |
| **3** | **Registro de pasajeros** | Permitir registrar hasta 5 pasajeros por compra con todos los datos requeridos (nombres, apellidos, documento, fecha de nacimiento, etc.). | **Alta** |
| **4** | **Registro del pagador** | Capturar los datos del comprador responsable del pago. | **Alta** |
| **5** | **Pago** | proceso de pago mediante tarjeta de crédito, débito o PSE, mostrando visualmente si fue exitoso o rechazado. | **Alta** |
| **6** | **Generación de tiquetes electrónicos** | Generar un archivo PDF con la información del vuelo, los pasajeros y un código de reserva único. | **Alta** |
| **7** | **Confirmación de reserva** | Mostrar resumen completo de la reserva y permitir su visualización o descarga. | **Alta** |
| **8** | **Modulo administrativo** | Permitir a un administrador visualizar vuelos, reservas y pasajeros registrados. | **Alta** |
| **9** | **Gestionar aviones** | Permitir a un administrador Crear modificar y eliminar aviones | **Alta** |
| **10** | **Gestionar vuelo** | Permitir a un administrador Crear modificar y eliminar vuelos | **Alta** |

* 1. **Requerimientos no funcionales**

|  | | **SENA**  **SISTEMA PARA COMPRA DE TIQUETES AÉREOS No 001**    **No 001** |  |
| --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Prioridad** |
| **1** | **Rendimiento** | El sistema debe procesar una búsqueda de vuelos en menos de 5 segundos. | **Alta** |
| **2** | **Disponibilidad** | El sistema debe estar disponible la mayoria del tiempo | **Alta** |
| **3** | **Seguridad** | Toda la información sensible (datos de usuarios, pagos) debe estar cifrada | **Alta** |
| **4** | **Validación** | Los formularios deben validar datos antes de enviarse al servidor | **Alta** |
| **5** | **Accesibilidad** | El sistema debe ser accesible desde navegadores modernos y móviles. | **Media** |
| **6** | **Compatibilidad** | Generar un archivo PDF con la información del vuelo, los pasajeros y un código de reserva único. | **c** |