

Anteproyecto – Eduardo Merino Fernández

SkyStay – Gestión de Hoteles

1. INTRODUCCIÓN

Este proyecto va a tratar de una plataforma web (SkyStay) que se encarga de gestionar reservas para una cadena de hoteles. Va a constar de 2 partes, una parte de Front que será toda la interfaz web a la que puede acceder el usuario y realizar las funciones y otra parte Back encargada de realizar operaciones y comunicarse con la base de datos SQL. Habrá un apartado para los diferentes administradores de los hoteles que puedan realizar diferentes gestiones aparte de las gestiones que puedan realizar los clientes.

2. UBICACIÓN Y DESTINATARIOS

La aplicación va dirigida a empresas hoteleras que requieran un sistema de reservas eficiente y una interfaz para el usuario clara y con numerosas funcionalidades como el sistema de puntos de fidelización. El sistema está diseñado para una cadena de hoteles, permitiendo la gestión de reservas de habitaciones y servicios adicionales como desayunos o comidas, a parte de esta gestión la aplicación será usada por los clientes para que puedan reservar online su habitación y esta reserva será registrada en la base de datos del hotel.

Supuesto Práctico Real: Una cadena hotelera con presencia en varias ciudades necesita una solución centralizada para gestionar las reservas de sus habitaciones. El sistema permitirá que los clientes a través de la web realicen reservas en línea y que los administradores del hotel puedan gestionar la disponibilidad de las habitaciones, modificar reservas y generar informes...

3. OBJETIVOS

1. Desarrollar una API REST en Spring Boot para la gestión de reservas hoteleras.
2. Implementar un sistema de autenticación y autorización para administradores y clientes.
3. Permitir a los clientes consultar disponibilidad y realizar reservas en tiempo real y poder canjear sus puntos por recompensas.
4. Desarrollar un frontend intuitivo que se conecte con la API REST.
5. Gestionar cancelaciones y modificaciones de reservas.
6. Sistema de envío de correos al crear una nueva cuenta y cuando se canjea una recompensa.
7. Implementar un sistema de notificación para confirmar reservas y recordar check-in/check-out (por correo).
8. Garantizar la seguridad de los datos de los usuarios y transacciones.
9. Proporcionar reportes y estadísticas a los administradores sobre la ocupación del hotel.

4. ESTUDIO DE VIABILIDAD.

A) REQUISITOS HW y SW

- **Software:**
 - Framework Back - Spring Boot 3.x (Backend – API REST)
 - IDE – IntelliJ y Visual Studio Code
 - Base de datos - MariaDB
 - Framework Front - Angular
 - Despliegue - Docker
 - Pruebas API - Postman
 - Documentación API -Swagger
 - Control de versiones - Git y GitHub
 - Pruebas Unitarias - JUnit
- **Hardware:**
 - Servidores con arquitectura Windows
 - CPU: 4 cores o superior
 - RAM: 8GB o más
 - Espacio en disco: 50GB SSD (para base de datos)

B) TEMPORALIZACIÓN PLANIFICACIÓN

Se planifica el desarrollo en 12 semanas dividiendo las tareas en fases.

Proceso Aplicación SkyStay

Gráfico de Gantt

PROCESO	PLANIFICACIÓN			DESARROLLO				DESPLIEGUE				
	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10	Semana 11	Semana 12
Análisis de requisitos	■	■										
Diseño de arquitectura		■	■	■								
Desarrollo del backend				■	■	■	■					
Desarrollo del frontend						■	■					
Integración API – Frontend							■	■				
Pruebas y depuración									■	■		
Despliegue en entorno real										■	■	
Documentación y preparación de la presentación											■	■

Semana	Tarea
1	Análisis de requisitos y diseño de arquitectura
2-4	Desarrollo del backend (API REST en Spring Boot)
5-6	Desarrollo del frontend (Angular)
7-8	Integración API – Frontend
9-10	Pruebas y depuración
11	Despliegue en entorno real
12	Documentación y preparación de la presentación

5. INTEGRANTES DEL PROYECTO Y REPARTO DE FUNCIONES

Nombre	Funciones
Eduardo Merino	Desarrollo del backend (Spring Boot, API REST)
	Desarrollo del frontend (React.js o Angular)
	Gestion de la base de datos y despliegue
	Pruebas, documentación y control de versiones.