

Proyecto Integrador: Kingdom Barber

Integrantes:

Alejandro Urrego

Andrés Vallejo

Juan Manuel Rivera

Institución:

Institución Técnica CESDE

Fecha:

2025

Tabla de Contenido

1. Introducción y Visión General
 - 1.1. Resumen del Proyecto
 - 1.2. Planteamiento del Problema
 - 1.3. Solución Propuesta
2. Objetivos del Proyecto
 - 2.1. Objetivo General
 - 2.2. Objetivos Específicos
3. Arquitectura General del Sistema
 - 3.1. Diseño de la Arquitectura
 - 3.2. Flujo de Datos
4. Módulos del Proyecto
 - 4.1. Módulo 1: Backend y API Central (PI_MOVIL2)
 - 4.2. Módulo 2: Plataforma Web para Clientes (PI_WEB2)
 - 4.3. Módulo 3: Panel de Gestión y Análisis de Datos (PI_NTP)
5. Stack Tecnológico Consolidado
6. Conclusiones del Proyecto

1. Introducción y Visión General

1.1. Resumen del Proyecto

En el competitivo sector del cuidado personal, la experiencia del cliente y la eficiencia operativa son diferenciadores clave. El proyecto Kingdom Barber nace como una plataforma digital integral, diseñada para transformar y modernizar la gestión de barberías a través de la tecnología. Su objetivo es ofrecer una solución robusta que no solo simplifique la administración de citas, sino que también enriquezca la interacción con el cliente y proporcione herramientas avanzadas para la toma de decisiones estratégicas.

La solución se compone de tres sistemas interconectados que funcionan de manera sinérgica:

- Una **Plataforma Web** orientada al cliente, desarrollada con tecnologías modernas como Next.js, que garantiza una experiencia de usuario fluida e intuitiva para la reserva y personalización de servicios.
- Un **Backend robusto y centralizado**, construido sobre Spring Boot (Java), que actúa como el cerebro de la operación, gestionando la lógica de negocio, la seguridad y la persistencia de los datos a través de una API RESTful.
- Un **Panel de Gestión y Análisis de Datos**, desarrollado en Python con Streamlit, destinado a los administradores para ofrecer una visión profunda del rendimiento del negocio, visualizar métricas clave e integrar un innovador Asistente de Inteligencia Artificial.

1.2. Planteamiento del Problema

La gestión tradicional de barberías enfrenta desafíos significativos en la era digital. Existe una necesidad clara de una solución digital que permita a las barberías gestionar sus operaciones de manera más eficiente, ofrecer una experiencia personalizada a los clientes y mantenerse competitivas en un mercado en constante evolución. La dependencia de procesos manuales para la reserva de citas, la comunicación ineficiente y la falta de análisis de datos resultan en una menor eficiencia operativa y una experiencia que no cumple con las expectativas actuales. La acelerada adopción digital, impulsada por factores como la pandemia de COVID-19, ha subrayado la necesidad crítica de implementar soluciones que garanticen seguridad, comodidad y personalización.

1.3. Solución Propuesta

Kingdom Barber aborda estos desafíos mediante una plataforma unificada que digitaliza y optimiza las operaciones. La solución propuesta incluirá funcionalidades clave como la gestión de reservas, la personalización de servicios, la comunicación directa con los clientes y la venta de productos. Adicionalmente, el panel administrativo empodera a los dueños del

negocio con herramientas de análisis de datos e inteligencia artificial para identificar patrones, detectar oportunidades y automatizar tareas de marketing, llevando la gestión a un nivel superior.

2. Objetivos del Proyecto

2.1. Objetivo General

Mejorar la eficiencia operativa de las barberías mediante la digitalización de la gestión de reservas y la comunicación con los clientes, ofreciendo una experiencia personalizada y aumentando la satisfacción del cliente.

2.2. Objetivos Específicos

- **Seguridad y Acceso:** Desarrollar un sistema de autenticación y seguridad robusto que incluya inicio de sesión, registro de nuevos usuarios y recuperación de contraseñas.
- **Usabilidad y Navegación:** Crear una interfaz de usuario intuitiva con menús desplegables, barras de búsqueda y navegación por categorías para garantizar una experiencia fluida.
- **Gestión de Reservas:** Implementar un sistema de gestión de citas completo que permita la selección de servicios, visualización de un calendario interactivo con horarios, selección de cortes favoritos y el envío de recordatorios de citas.
- **Experiencia Personalizada:** Ofrecer opciones de personalización que permitan a los usuarios ajustar el tema, configurar notificaciones y explorar un catálogo detallado de servicios, productos y sedes.
- **Comunicación Directa:** Facilitar la interacción a través de chat en vivo, soporte por correo electrónico, acceso a redes sociales y números de contacto.
- **Venta y Promoción:** Integrar funcionalidades de comercio electrónico, incluyendo un carrito de compras, wishlist, reseñas de productos y gestión de ofertas.

3. Arquitectura General del Sistema

3.1. Diseño de la Arquitectura

El proyecto está diseñado bajo una arquitectura de componentes desacoplados, donde cada módulo tiene una responsabilidad única y se comunica a través de una API RESTful. Este enfoque garantiza escalabilidad, mantenibilidad y flexibilidad.

- **Backend (Núcleo del Sistema - PI_MOVIL2):** Es el componente central desarrollado en Spring Boot (Java). Aloja la lógica de negocio y gestiona los datos de usuarios, citas y servicios. Utiliza un patrón MVC adaptado para una API REST y se estructura en

controladores (endpoints), modelos (entidades JPA) y repositorios (operaciones CRUD).

- **Frontend (Plataforma del Cliente - PI_WEB2):** Es la capa de presentación desarrollada con Next.js (React y TypeScript). Consume los datos del backend para renderizar una interfaz dinámica. Su estructura modular se basa en el App Router de Next.js, separando las páginas, componentes reutilizables y la lógica de la API.
- **Panel de Administración (Inteligencia de Negocio - PI_NTP):** Es una aplicación web independiente en Python y Streamlit. Se conecta a la misma fuente de datos para realizar análisis, generar visualizaciones y ejecutar modelos de IA, sin afectar el rendimiento de la aplicación principal. Su estructura separa las páginas (Dashboard, Gestión de Citas, Asistente IA) de los módulos de lógica de datos y generación de reportes.

3.2. Flujo de Datos

El flujo de datos sigue un patrón cliente-servidor estándar. Por ejemplo, al crear una cita:

1. **Cliente (Frontend):** El usuario rellena el formulario de reserva en la interfaz de Next.js y hace clic en "Agendar".
2. **Petición:** El frontend empaqueta los datos en un objeto JSON y envía una petición POST a la API del backend (/api/citas) usando la Fetch API.
3. **Controlador (Backend):** El CitaController.java en Spring Boot recibe la petición. El cuerpo JSON se convierte automáticamente en un objeto Java (Cita).
4. **Lógica y Persistencia:** El controlador invoca al CitaRepository, que utiliza Spring Data JPA para insertar una nueva fila en la tabla de citas de la base de datos.
5. **Respuesta:** El backend devuelve una respuesta JSON al frontend con un código de estado 201 CREATED, confirmando la creación de la cita.
6. **Actualización de UI:** El frontend recibe la confirmación y actualiza la interfaz para mostrar al usuario que su cita ha sido agendada con éxito.

4. Módulos del Proyecto

4.1. Módulo 1: Backend y API Central (PI_MOVIL2)

- **Descripción:** Es el corazón del sistema, desarrollado con Spring Boot y Java. Gestiona toda la lógica, los datos y la seguridad a través de una API REST.
- **Funcionalidades Principales:**
 - **Endpoints RESTful:** Expone un conjunto de endpoints para gestionar clientes, barberos, sedes, servicios y citas.
 - **Portal del Cliente:** Ofrece la lógica para reservar, visualizar, modificar y cancelar citas.
 - **Portal del Barbero:** Proporciona un panel donde los barberos pueden iniciar sesión por sede y visualizar su agenda de forma ordenada.

4.2. Módulo 2: Plataforma Web para Clientes (PI_WEB2)

- **Descripción:** Es la cara visible del proyecto, construida con Next.js, React y TypeScript para ofrecer una experiencia de usuario excepcional.
- **Funcionalidades Principales:**
 - **Sistema de Reservas Avanzado:** Incluye un calendario interactivo para seleccionar sede, barbero y servicios.
 - **Exploración de Servicios y Sedes:** Ofrece un catálogo detallado de servicios y un mapa interactivo para geolocalización.
 - **Panel para Barberos:** Un dashboard exclusivo donde los barberos pueden gestionar su agenda y publicar una galería de sus trabajos.
 - **Autenticación y Comunicación:** Incluye sistema de registro e inicio de sesión seguro y un formulario de contacto.

4.3. Módulo 3: Panel de Gestión y Análisis de Datos (PI_NTP)

- **Descripción:** Herramienta estratégica en Python con Streamlit, diseñada para el análisis de datos y la administración del negocio.
- **Funcionalidades Principales:**
 - **Dashboard Interactivo:** Ofrece una vista panorámica del rendimiento con KPIs (Ingresos Totales, Total de Citas, etc.) y filtros avanzados en tiempo real.
 - **Gestión de Citas:** Permite la búsqueda y consulta detallada del historial de citas con filtros combinables.
 - **Asistente de Inteligencia Artificial:** Un centro de análisis avanzado que integra un LLM con varias capacidades:
 - **Generador de Reportes:** Crea informes ejecutivos en PDF con análisis generados por IA.
 - **Analista de Datos Interactivo:** Un chatbot que responde preguntas en lenguaje natural sobre los datos.
 - **Asistente de Marketing:** Genera textos persuasivos para campañas de marketing.
 - **Detector de Oportunidades:** Analiza patrones para identificar proactivamente oportunidades de negocio.

5. Stack Tecnológico Consolidado

Capa	Tecnología	Propósito
Backend	Java, Spring Boot, Spring Data JPA, Hibernate	Lógica de negocio, API RESTful, persistencia de datos
Frontend	Next.js, React, TypeScript,	Interfaz de usuario,

	Tailwind CSS	experiencia del cliente
Análisis de Datos	Python, Streamlit, Pandas, Plotly	Panel de administración, visualización, IA
Base de Datos	H2 (Desarrollo), SQL (Producción)	Almacenamiento de datos

6. Conclusiones del Proyecto

A lo largo del desarrollo del proyecto Kingdom Barber, hemos logrado avances significativos y, aunque nos encontramos con algunos desafíos, pudimos superarlos gracias a la colaboración constante y el enfoque ágil que adoptamos.

- **Evolución del diseño:** Desde el inicio, nos centramos en crear una plataforma fácil de usar y visualmente atractiva. A medida que avanzamos, fuimos ajustando la maquetación y mejorando la experiencia del usuario para asegurar una navegación intuitiva.
- **Importancia de la comunicación:** Las reuniones diarias y las sesiones de revisión fueron clave para mantenernos alineados. La comunicación fluida nos permitió resolver problemas rápidamente y tomar decisiones informadas.
- **Adaptabilidad ante los cambios:** Durante el proceso, tuvimos que adaptarnos a nuevas propuestas y cambios. Esta flexibilidad fue esencial para simplificar partes del proyecto y mejorar la experiencia final.
- **Desarrollo de funcionalidades clave:** Nos enfocamos primero en las funcionalidades esenciales como el sistema de reservas y el inicio de sesión. Con el tiempo, fuimos añadiendo más características que dieron forma a la plataforma.
- **Lecciones aprendidas:** Hubo momentos en los que nos sentimos estancados, pero supimos afrontarlo en equipo. Cada uno aportó su perspectiva, lo que ayudó a encontrar soluciones eficaces. Aprendimos a no subestimar el tiempo que algunas tareas pueden tomar y a ser más realistas en nuestras expectativas.

El proyecto no solo ha entregado una solución tecnológica funcional, sino que también ha sentado las bases para futuras expansiones, asegurando que Kingdom Barber pueda seguir evolucionando.