

Documento de Requisitos del Producto

Sistema de Gestión de Citas - Hospital San Fermín

Versión: 1.0

Autor: Alan Flores

Fecha: 26 de septiembre de 2025

1. Resumen Ejecutivo

El Sistema de Gestión de Citas es una API RESTful backend diseñada para administrar las operaciones fundamentales de una clínica. Su objetivo principal es proveer una base de software robusta y escalable para la creación de pacientes, la gestión de doctores y la programación de citas. Este sistema no incluye una interfaz de usuario (UI), sino que expone una serie de endpoints seguros y bien definidos para que futuras aplicaciones (web, móvil) puedan consumirlos.

Este proyecto sirve como una demostración de competencia en tecnologías de backend modernas, aplicando las mejores prácticas de arquitectura de software, como el diseño en capas y la programación orientada a eventos.

2. Objetivos del Producto

O1

Centralizar la Lógica de Negocio:
Crear una única fuente de verdad
para la gestión de pacientes,
doctores y citas.

O2

Proveer una API RESTful Coherente:
Exponer endpoints claros,
predecibles y bien documentados
para facilitar su integración con otros
sistemas.

O3

Garantizar la Integridad de los Datos:
Asegurar que los datos almacenados
sean válidos y consistentes mediante
reglas de negocio y validaciones.

O4

Demostrar Excelencia Técnica: Servir
como un proyecto de portafolio que
evidencia el dominio de Java, Spring
Boot y principios de arquitectura de
software avanzada.

3. Alcance del Proyecto

✓ Funcionalidades Incluidas (In-Scope)

- Gestión de Pacientes:** Registro de nuevos pacientes y consulta del historial de citas.
- Gestión de Doctores:** Población inicial de la base de datos y consulta de doctores disponibles.
- Gestión de Citas:** Agendamiento y cancelación de citas.
- Sistema de Notificaciones:** Publicación de eventos asíncronos y simulación de confirmaciones.

X Funcionalidades Excluidas (Out-of-Scope)

- Interfaz de usuario (Frontend)
- Autenticación y autorización de usuarios
- Gestión de historial clínico, recetas o facturación
- Integración real con servicios de email o SMS
- Panel de administración para CRUD completo

4. Requisitos Funcionales

Módulo	Feature ID	Descripción de la Funcionalidad	Endpoint de API	Método HTTP
Pacientes	F-PA-01	Registrar un nuevo paciente con sus datos personales.	/api/pacientes	POST
Pacientes	F-PA-02	Consultar la lista de todas las citas de un paciente.	/api/pacientes/{numeroPaciente}/historial-citas	GET
Doctores	F-DO-01	Obtener una lista de todos los doctores disponibles.	/api/pacientes/doctores-disponibles	GET
Citas	F-CI-01	Agendar una nueva cita para un paciente con un doctor.	/api/pacientes/citas	POST
Citas	F-CI-02	Eliminar una cita agendada.	/api/pacientes/{numeroPaciente}/citas/{citaId}	DELETE
Notificaciones	F-NO-01	Al ejecutarse F-CI-01, el sistema debe imprimir en consola una confirmación para el paciente y una notificación para el doctor.	N/A (Proceso interno)	N/A

5. Requisitos No Funcionales

- Tecnología:** El sistema está construido con Java (versión 17+) y el framework Spring Boot.
- Base de Datos:** Utiliza una base de datos en memoria (H2) para facilitar el desarrollo y las pruebas, pero está diseñado para ser compatible con bases de datos relacionales como PostgreSQL.
- Arquitectura:** Sigue un patrón de diseño en capas (Controlador, Servicio, Repositorio) para una clara separación de responsabilidades.
- Mantenibilidad:** El código está desacoplado mediante el uso de un bus de eventos para las notificaciones, permitiendo añadir nuevas reacciones a eventos sin modificar el código existente.

6. Criterios de Éxito

- ✓ Todos los endpoints definidos en los Requisitos Funcionales responden correctamente con los códigos de estado HTTP apropiados.
- ✓ Las reglas de negocio (ej: un paciente no puede tener citas duplicadas a la misma hora) se aplican correctamente.
- ✓ El sistema de eventos de notificación se activa de forma fiable al crear una cita.
- ✓ El código fuente está limpio, bien organizado y sigue las convenciones de Java y Spring Boot.

