

# TRABAJO PRACTICO INTEGRADOR

# NOTIFICACIONES

Grupo 8

# INTEGRANTES

- **Coordinación:** Todos los integrantes.
- **Backend:** Agustín Gómez y Sebastian Scarcelli.
- **Frontend + Documentación:** Rocio Cappuccio.
- **Pruebas:** Todos los integrantes.

# OBJETIVO DEL PROYECTO

Desarrollar un sistema que permita crear plantillas reutilizables y enviar notificaciones personalizadas a los usuarios

## TECNOLOGÍAS

- Java 17
- Spring Boot
- Maven
- MySQL
- Postman

# FUNCIONALIDADES

01

## Funcionalidad 01

---

Crear y administrar plantillas de mensajes

02

## Funcionalidad 02

---

Generar notificaciones con variables dinámicas

03

## Funcionalidad 03

---

Almacenar y consultar las notificaciones enviadas

# ARQUITECTURA DEL SISTEMA

El sistema está diseñado con una arquitectura en capas:

**ENTITY:** Plantilla,  
Notificación

**REPOSITORY:** Interfaces que extienden  
JpaRepository para acceso a datos

**SERVICE:** Lógica de negocio para crear,  
actualizar y consultar.

**CONTROLLER:** Endpoints REST para interactuar  
con la API

Ejemplo: una plantilla de correo con variables dinámicas como {{nombre}} y {{fecha}}, que se reemplazan al generar la notificación

# INSTALACIÓN Y CONFIGURACIÓN

## Requisitos previos

Java 17

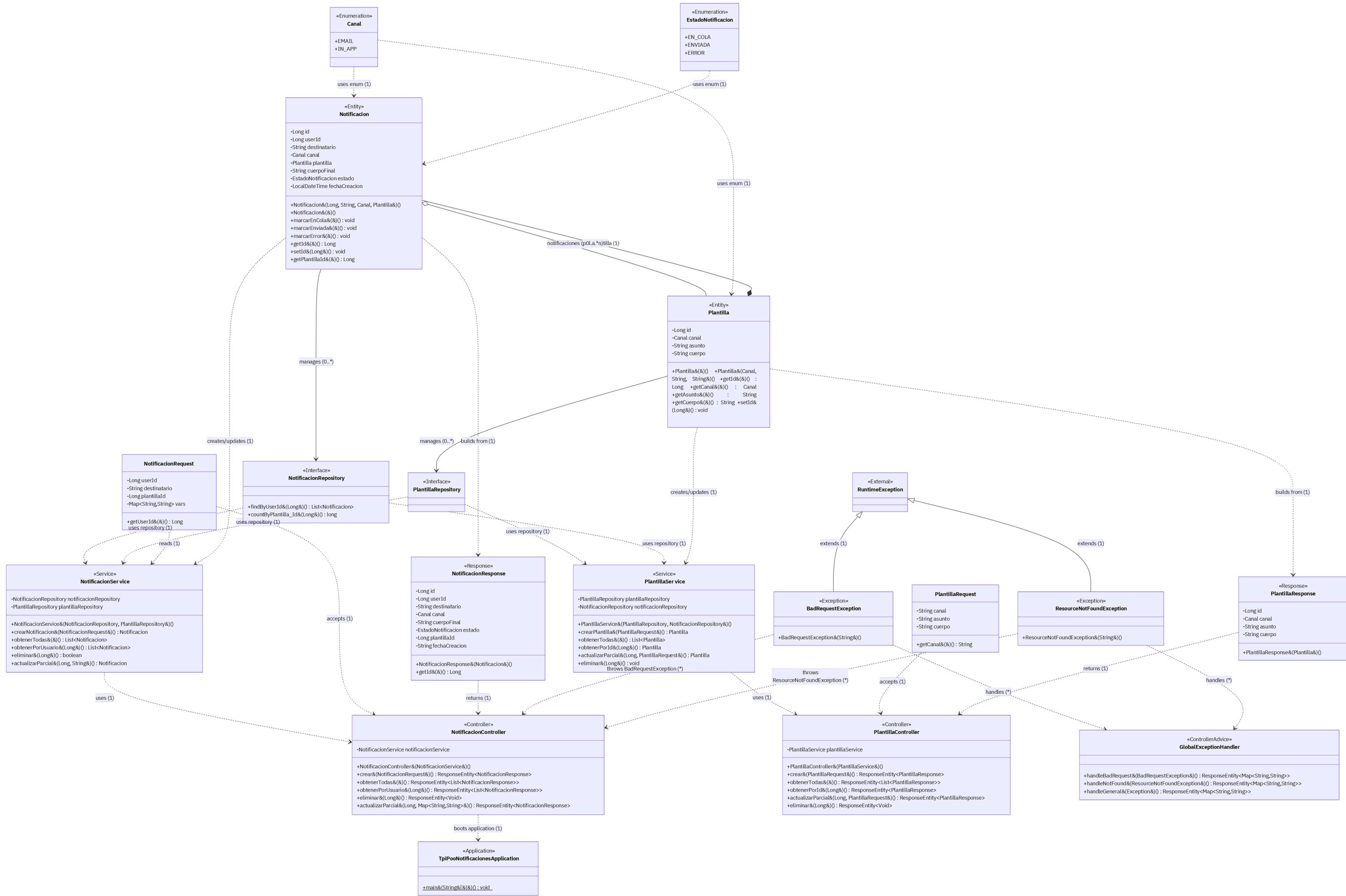
Maven

MySQL

## Pasos de instalación

1. Clonar el repositorio desde GitHub
2. Configurar la base de datos en application.properties
3. Ejecutar el proyecto con: mvn spring-boot:run

# DIAGRAMA UML



# CASOS DE PRUEBA

- 1. Crear una plantilla**
- 2. Listar todas las plantillas**
- 3. Crear una notificación con una plantilla existente**
- 4. Consultar notificaciones enviadas.**
- 5. Probar errores (ejemplo: intentar usar una plantilla inexistente)**

MUCHAS  
GRACIAS

