

## 1. Vista general

### Grupo3:

**Nombre:** Geny Marcela Vargas Suarez

**Nombre:** Santiago Martínez Yara

**Nombre:** Miguel Ángel Salazar Jaramillo

**Nombre:** Angela Consuelo torres Trujillo

**Nombre:** Esteban Henao Sánchez

### 1.1 Nombre del Proyecto: Estramipyme

**1.2 Descripción:** Estramipyme es una plataforma web que tiene como objetivo facilitar a emprendedores la tarea de identificar y trabajar en las áreas de mejora de sus negocios mediante una prueba de diagnóstico y seguimiento posterior a esta, cuyos resultados resaltan las fortalezas y puntos débiles, permitiendo una mejora en las decisiones y gestión empresarial. También permite separar la posibilidad de toma de acciones en función del tipo de usuario que accede a la plataforma, distinguiendo entre Administradores y Usuarios.

## 2. Objetivos del Producto Mínimo Viable (PMV):

**2.1 Diagnóstico Empresarial:** ofrece una prueba que permita a los emprendedores obtener una evaluación inicial de sus fortalezas y áreas de mejora.

**2.2 Gestión de Usuarios y Roles:** un sistema de roles para administradores y usuarios, permitiendo interacción según permisos.

**2.3 Seguimiento de Usuarios:** proporciona datos estadísticos acerca de los usuarios registrados, pruebas realizadas y otras acciones tomadas por ellos dentro de la plataforma.

## 3. Funcionalidades Clave del PMV

**3.1 Registro e Inicio de Sesión:** proporciona registro e inicio de sesión, distinguiendo entre Usuarios y Administradores.

**3.2 Prueba de Diagnóstico:** examen cualitativo que permite destacar los aspectos que se deben mejorar para hacer más eficientes los procesos dentro de la empresa, y que permite guiar a los usuarios por distintos planes de aprendizaje de acuerdo a sus resultados.

**3.3 Seguimiento de Progreso:** contador de usuarios que completan la prueba.

**3.4 Creación de Profesores y Estudiantes:** por la parte del Back End, proporciona la funcionalidad de crear Profesores y Estudiantes y modificar la información de estos, en caso de ser necesario.

## 4. Flujo de Peticiones y Arquitectura de Spring Boot

Estramipyme utiliza una de tipo cadena para gestionar y procesar las peticiones que se realizan a la plataforma, empleando los principios de CRUD (Crear, Leer, Actualizar, Eliminar) en cada interacción con la base de datos.

**4.1 Controladores:** los controladores son el punto de entrada de las peticiones HTTP. Estos puntos de acceso reciben las solicitudes y delegan el procesamiento al nivel de servicios. Cada controlador se asocia con una ruta y ofrece métodos específicos para cada operación CRUD, permitiendo a los usuarios realizar tareas como:

- o Crear nuevos usuarios, proyectos y registros de pruebas.

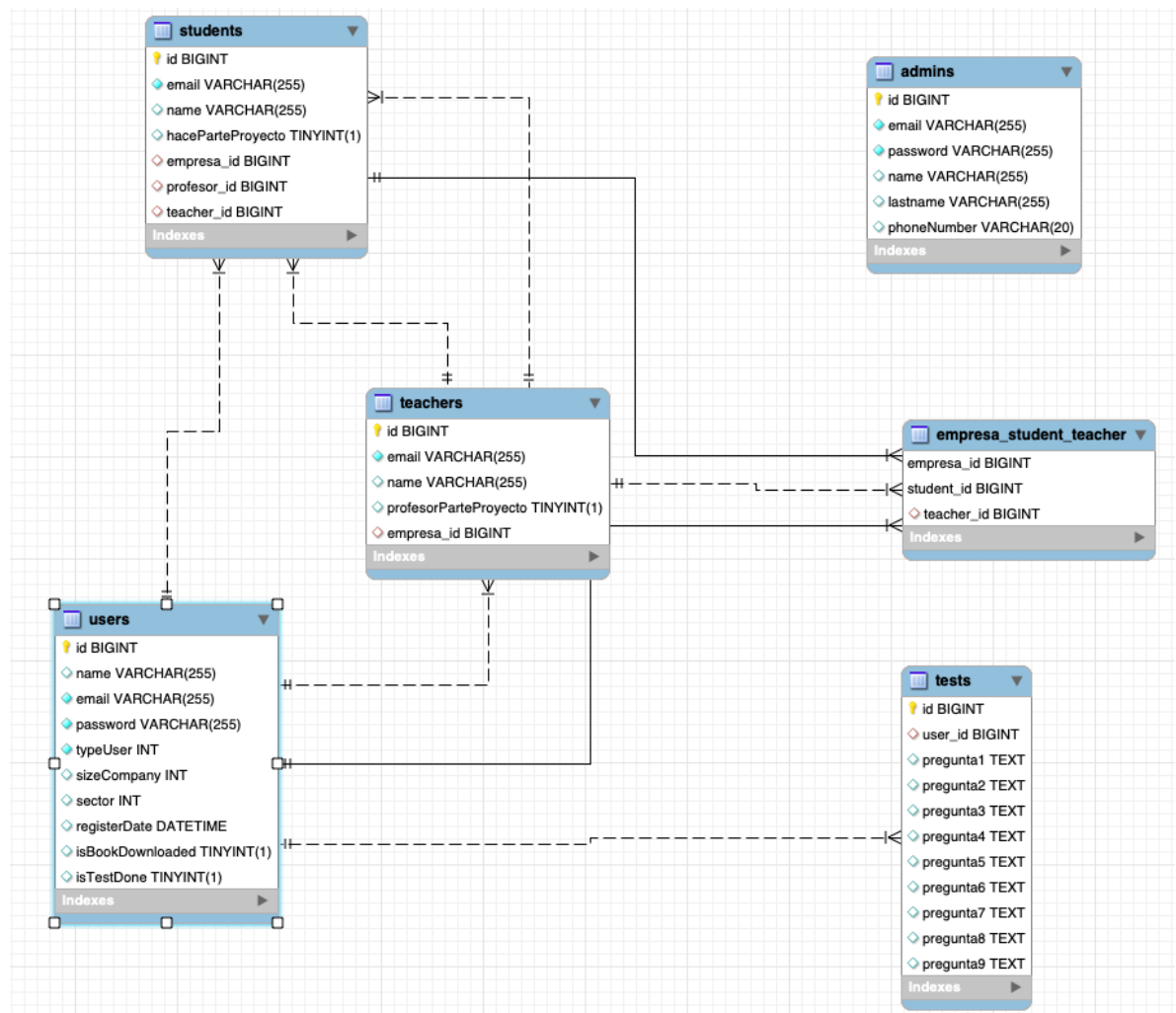
- o Leer datos sobre proyectos asignados, resultados de pruebas y el perfil de usuario.

- o Actualizar información de usuarios o asignaciones de proyecto.
- o Eliminar registros según permisos de usuario.

**4.2 Servicios:** la lógica de negocio principal reside en los servicios que controlan las operaciones CRUD, llaman a los repositorios para acceder o modificar datos en la Base de Datos e implementan verificaciones. Los servicios reciben las peticiones desde los controladores y ejecutan la lógica necesaria para completar la operación requerida.

**4.3 Modelos:** los modelos representan las entidades del sistema, como Usuario, Proyecto y Prueba. Cada modelo se encuentra mapeado a una tabla específica en la base de datos MySQL mediante anotaciones JPA (Java Persistence API), lo que facilita la persistencia de datos.

**4.4 Repositorios:** los repositorios son interfaces que gestionan las operaciones CRUD directamente con la base de datos. Utilizan JPA y las herramientas de Spring Data para simplificar las consultas y transacciones con MySQL, permitiendo un acceso y manipulación eficiente de los datos.



## 5. Base de Datos

**5.1 Gestión de Datos con MySQL:** la base de datos se almacena y gestiona en MySQL, en donde cada entidad de la plataforma (como Usuarios, Proyectos y Pruebas) tiene su tabla correspondiente.

## 6. Tecnologías Utilizadas

**6.1 Front End:** las interfaces de usuario se construyeron sobre Angular 18 con TypeScript como lenguaje de programación, Tailwind CSS como librería de estilos, SCSS como pre-procesador para los estilos, ChartJS para las gráficas dinámicas y HTML5 para la maquetación.

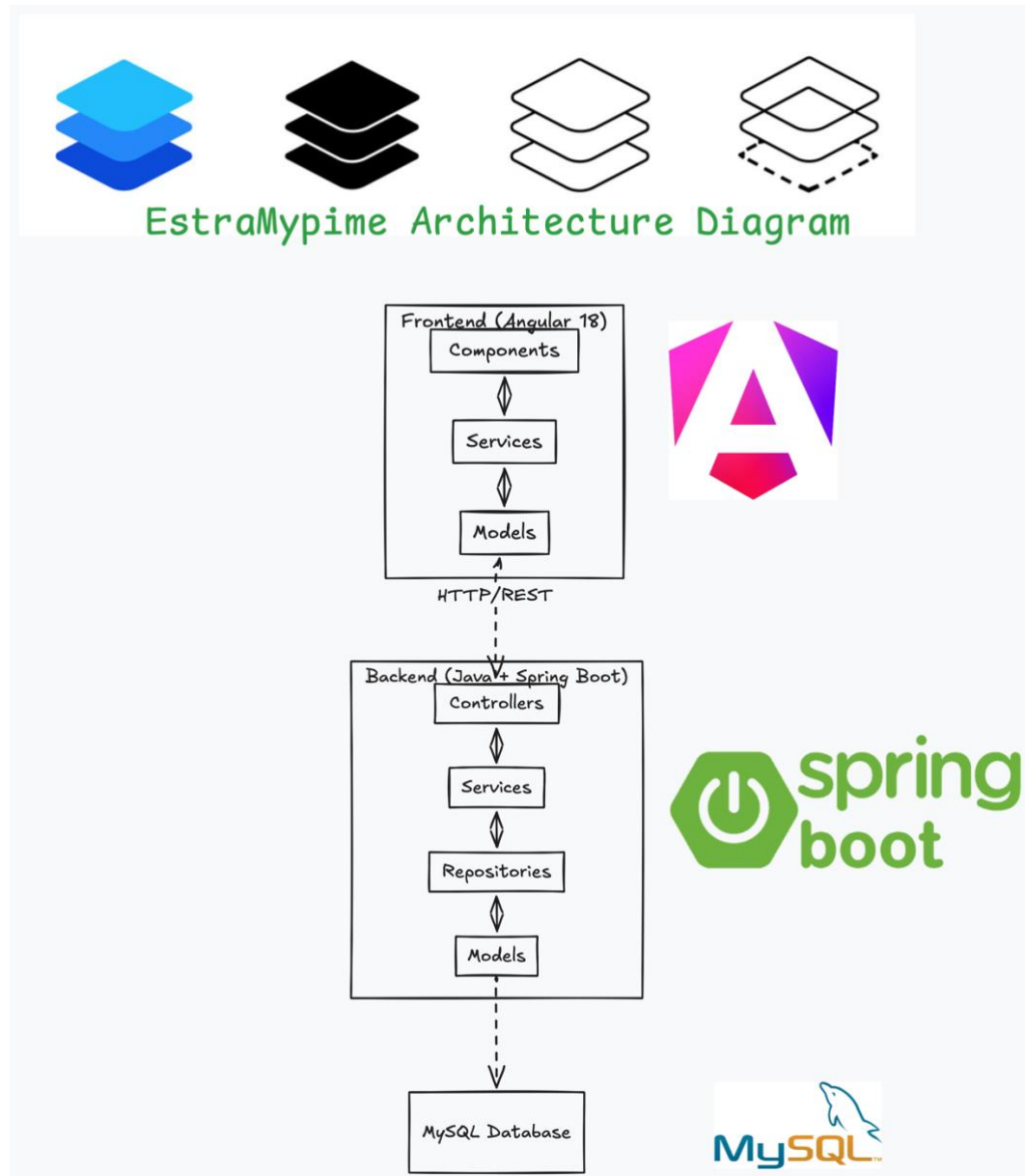
**6.2 Back End:** se empleó Java 17 como lenguaje de programación con el Framework Spring Boot para hacer una aplicación más robusta y mejorar la velocidad del desarrollo, además del uso de las dependencias Lombok para evitar la repetición de código, JPA y MySQL Driver para la conexión y manejo de datos con una base de datos MySQL.

**6.3 Control de Versiones:** se empleó Git como tecnología para el manejo de versiones y GitHub como repositorio.

## Colecciones postman

<https://www.postman.com/gold-escape-545722/estramypimegrupo3/folder/jzukjtx/teachers?action=share&creator=28945019&ctx=documentation>

## Diagrama de arquitectura



## Diagrama de flujo detallado.

