

Ingeniería de Software, 5 de noviembre de 2024

EJERCICIO SOBRE EL DESARROLLO DE SOFTWARE

Martínez Buenrostro Jorge Rafael

correo, molap96@gmail.com

Universidad Autónoma Metropolitana
Unidad Iztapalapa, México

Descripción del problema

Dentro del mundo de los gimnasios podemos encontrar dos roles: los entrenadores o asesores, y los que acuden al gimnasio en búsqueda de un cambio físico. Es muy común que los segundos busquen entrenadores para obtener rutinas y consejos nutricionales para lograr los resultados que quieren de manera más efectiva.

Sin embargo, la mayoría de los entrenadores trabajan independientemente del gimnasio por lo que no cuentan con un sistema para llevar la gestión de sus clientes. Por lo general su contacto es por WhatsApp; las rutinas y los planes alimenticios suelen ser creadas en las notas del teléfono del entrenador, y en algunos casos cuentan con plantillas en Word que mandan a sus clientes.

Esto provoca que los entrenadores no tengan una gestión eficiente de sus clientes. Así como una personalización de las rutinas y planes alimenticios asignados. Otra problemática es que los entrenadores no tienen forma de llevar registro del progreso de sus clientes a menos que se tengan un contacto constante con ellos.

Diagrama de la arquitectura

Para este desarrollo se usó la arquitectura de tres capas (*3-layer*) es una estructura de diseño de software que organiza una aplicación en tres niveles distintos, cada uno con responsabilidades específicas. Aquí te explico cada una de las capas:

1. **Capa de Presentación** - Es la interfaz de usuario de la aplicación. Aquí es donde los usuarios interactúan con el sistema.
2. **Capa de Aplicación** - Conocida como capa lógica o de negocio, es donde se procesan los datos y se aplican las reglas de negocio.
3. **Capa de Datos** - Se encarga del almacenamiento y gestión de datos. Aquí es donde se accede a las bases de datos y se realizan las operaciones CRUD (*Create, Read, Update, Delete*).

Beneficios de la Arquitectura de Tres Capas

1. **Separación de responsabilidades**
2. **Desarrollo en paralelo**
3. **Reutilización de componentes**
4. **Escalabilidad**

Stack Tecnológico

La razón principal para la selección de este stack tecnológico fue el interés por aprender nuevas tecnologías. Esta será una aplicación web por lo que se usaron las siguientes:

- **Django** - Framework de desarrollo web de alto nivel escrito en Python. Se utiliza principalmente para crear aplicaciones web de manera rápida y eficiente.
- **PostgreSQL** - Es un sistema de gestión de bases de datos relacionales de código abierto, conocido por su robustez y flexibilidad.