

Aplicaciones Informáticas 2

- Nombre: Jennifer Yambay – 6916
- Fecha: 25 – 05 – 2025
- Docente: Ing. Julio Santillán

Tecnologías

Lenguajes de Programación

• Frontend:

Se ha optado por **React.js** como el principal framework para el desarrollo del frontend, gracias a sus características modernas y su capacidad para cubrir de manera efectiva los requerimientos del proyecto.

- Permite la construcción de interfaces de usuario altamente interactivas.
- Fomenta el desarrollo de código modular, escalable y fácil de mantener.
- Es un sistema robusto que puede integrarse sin problemas con diversos frameworks y bibliotecas.

• Backend:

Para el desarrollo del backend, se seleccionó **Django REST Framework (DRF)** por su solidez y su capacidad de gestionar eficientemente los requerimientos funcionales del sistema.

- Proporciona utilidades integradas para la serialización, validación y manejo de datos, acelerando el desarrollo sin comprometer la calidad del producto.
- Django, mediante su ORM, facilita la definición y manipulación de relaciones complejas entre entidades de la base de datos de forma declarativa.
- Al estar basado en Python, DRF permite una integración sencilla con bibliotecas de Machine Learning como TensorFlow, PyTorch o Scikit-learn, lo cual resulta útil para implementar funciones avanzadas como análisis predictivo, clasificación o generación de informes.

• PostgreSQL:

Este sistema está diseñado para gestionar grandes volúmenes de datos de forma eficiente. Soporta consultas complejas, índices avanzados y una gestión óptima de la concurrencia, convirtiéndolo en una solución ideal para proyectos con alta demanda de almacenamiento y procesamiento de datos.

- Su compatibilidad con el tipo de datos JSONB permite almacenar y consultar información semiestructurada, ideal para configuraciones flexibles o datos que no se ajustan a un modelo relacional tradicional.
- Facilita búsquedas eficientes en campos JSON mediante índices especializados.
- Proporciona flexibilidad para manejar información adicional relacionada con procesos de Machine Learning o análisis estadístico.

- Se integra con herramientas de visualización y análisis como Metabase, Tableau, Power BI, así como con bibliotecas de Python como Pandas y SQLAlchemy.
- Ofrece soporte nativo para replicación en tiempo real y asíncrona, lo que garantiza disponibilidad y consistencia en entornos distribuidos.

• Servidor:

Se utiliza un servidor virtual privado (VPS) en **AWS**, lo cual permite una configuración personalizada del entorno operativo. Esto asegura que los recursos como CPU, memoria y almacenamiento se adapten a las necesidades específicas del sistema, optimizando el rendimiento para tareas como procesamiento de datos y servicios REST.

• Clientes:

El acceso al sistema se realiza mediante navegadores web modernos, que son completamente compatibles con las tecnologías empleadas en el frontend, como React.js.

Protocolos de Comunicación

- Se emplea el protocolo **HTTPS** para asegurar la transmisión de datos, protegiendo la información frente a ataques del tipo *Man-in-the-Middle* (MitM).
- Para funcionalidades en tiempo real, se integran **WebSockets**, los cuales permiten una comunicación bidireccional continua, disminuyendo la latencia y reduciendo el consumo de recursos del servidor, a diferencia de las solicitudes HTTP tradicionales basadas en polling.

https://github.com/JenniferYambay/Apli_Proyecto.git