### **Tareas del Proyecto:**

#### **Task 1: Implementar y Asegurar un Endpoint de API**

**Complejidad**: Media

**Descripción**: Implemente el endpoint de COTIZACIONES, para la **estructura de datos para objeto**, que se encuentra descrito en el documento **Manual para Conexión API´s Affinities.** adjunto al correo. Asegure el endpoint con autenticación de token JWT y valide los datos usando Pydantic.

**Requisitos**:

* Crear un llamado al endpoint que soporte operaciones Get. (completado)
* Asegurar el endpoint con autenticación de token JWT. (completado)
* Validar los datos entrantes usando modelos Pydantic. (completado)
* Usar SQLModel para interactuar con la base de datos PostgreSQL. (completado)

#### **Task 2: Implementar y Asegurar un Endpoint de API**

**Complejidad**: Media

**Descripción**: Implemente un endpoint para crear, recuperar, actualizar y eliminar objetos "Plan". Asegure el endpoint con autenticación de token JWT y valide los datos usando Pydantic.

**Requisitos**:

* Crear un endpoint API /api/plans que soporte operaciones CRUD. (completado)
* Asegurar el endpoint con autenticación de token JWT.(completado)
* Validar los datos entrantes usando modelos Pydantic.(completado)
* Usar SQLModel para interactuar con la base de datos PostgreSQL.(completado)

#### **Task 3: Desarrollar una Interfaz Frontend para Gestión de Planes**

**Complejidad**: Alta

**Descripción**: Desarrolle una interfaz frontend basada en la plantilla React utilizando TypeScript y Chakra UI que permita a los usuarios agregar, modificar y eliminar objetos "Plan".

**Requisitos**:

* Crear un componente React para la gestión de planes.(completado Solo Crear plan)
* Usar Chakra UI para diseñar la interfaz, incluyendo elementos de formulario y botones. (Completado pero con Css puro, para demostrar mi manejo)
* Integrar el frontend con la API backend para realizar operaciones CRUD. (completado solo Crear Plan)

#### **Task 5: Configurar un Pipeline de CI/CD para Pruebas y Despliegue Automático** Desafortunadamente no me alcanzo el tiempo, pero adjunto te agrego un ejercicio practico de devops aplicado a mis servidores On-premise con Jenkins completamente documentado

#### <https://devops.chomelo.org/devops> Instrucciones despliegue local

dato importante la DB: debe estar previamente configurada en el gestor es por eso que en el repositorio encontraras un documento llamado DBEXPORT.pgsql donde podras importar las tablas, como en el esquema de la imagen db.png

Para el backend:  
  
…\SoftgicTest  
  
commando docker-compose up -d  
  
para el frontend:

…\SoftgicTest\frontend

npm install

npm run dev