Global Think Technology (Prueba Técnica Nest.js)

Problemática: manejar usuarios con sus perfiles

Solución: desarrollé una API REST utilizando NestJS con un enfoque modular y buenas prácticas de desarrollo backend.

Desarrollador: Héctor Frías

Estructura del proyecto

```
📂 src/
 ├── 📂 auth/            # Módulo de autenticación
 |    ├── decorators/     # Decoradores personalizados para roles/permisos
 |    ├── guards/         # Guards de seguridad (JWT, Roles, etc.)
 |    ├── strategies/     # Estrategias de autenticación (Local, JWT)
 |    ├── auth.controller.ts  # Controlador de autenticación
 |    ├── auth.module.ts      # Módulo de autenticación
 |    ├── auth.service.ts     # Lógica de autenticación
 |    ├── profile-permissions.ts # Permisos específicos de perfiles
 |
 ├── 📂 config/          # Configuración del entorno
 |    ├── env.config.ts    # Variables de entorno y configuración global
 |
 ├── 📂 data/            # Datos estáticos y servicios auxiliares
 |    ├── constants/      # Constantes usadas en toda la app
 |    ├── services/       # Servicios globales, como manejo de excepciones
 |
 ├── 📂 modules/         # Módulos específicos de la app
 |    ├── usersProfiles/   # Módulo de perfiles y usuarios
 |    |    ├── controllers/ # Controladores de perfiles
 |    |    ├── dto/        # DTOs para validaciones de datos
 |    |    ├── entities/   # Entidades TypeORM
 |    |    ├── services/   # Servicios para lógica de usuarios
 |    |    ├── users.module.ts  # Módulo de usuarios
 |
 ├── 📂 providers/       # Módulos principales de la app
 |    ├── app.controller.ts  # Controlador raíz
```

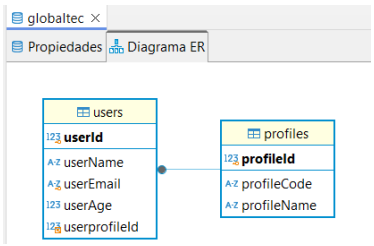
```
|   ├── app.module.ts      # Módulo raíz
|   ├── app.service.ts     # Servicio principal
```

Este proyecto es una aplicación desarrollada con NestJS que implementa autenticación, gestión de usuarios y perfiles. Se ha estructurado de manera modular para mejorar la escalabilidad y el mantenimiento.

Implementación

1. **Configuración del proyecto**
    - Creé un nuevo proyecto en **NestJS** con TypeScript.
    - Configuré las dependencias necesarias y el archivo tsconfig.json.
2. **Creación y conexión a la base de datos MySQL:**

- Se creó una base de datos en MySQL con el nombre globaltec, que contiene la tabla users-profiles. La conexión se estableció utilizando TypeORM, y las credenciales de acceso se gestionan a través del archivo .env para mayor seguridad y flexibilidad en la configuración.



3. **Diseño de la API REST**
    - Implementé las operaciones CRUD para la gestión de usuarios y perfiles.
    - Agregué un filtro por texto en la lista de usuarios.
4. **Modelo de Usuario y Perfil**
    - Definí entidades con los campos requeridos (id, nombre, correo electrónico, edad, perfil)-(código, nombre perfil).
    - **Validé** los datos con class-validator para asegurar formatos correctos y unicidad del correo electrónico.
5. **Manejo de errores y permisos**
    - Implementé respuestas JSON claras para errores.
    - se integró un log de errores.
    - Agregué un sistema de permisos para restringir accesos: En este punto Como no contaba con un sistema de autenticación

con usuario y contraseña, diseñé un servicio que genera un token utilizando el correo electrónico del usuario. Este token se utiliza para verificar si el usuario tiene permisos específicos para crear, leer, actualizar o eliminar registros.

6. **Documentación y pruebas**
    - **Swagger:** Generé documentación para visualizar y probar los endpoints(http://localhost:3000/docApi) solo implementado para entorno de desarrollo o testings.
    - **Pruebas unitarias:** Creé pruebas con Jest para asegurar la funcionalidad.

7. **Despliegue con Docker**

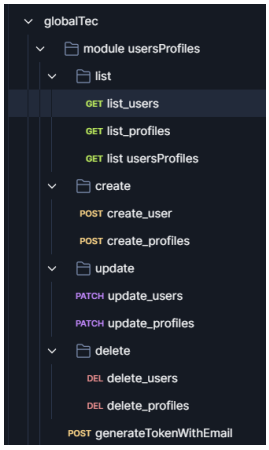
Este proyecto incluye archivos de configuración de Docker para facilitar la ejecución del backend en un entorno aislado, permitiendo una gestión eficiente de los servicios.

Archivos incluidos

- Dockerfile: Define la imagen del contenedor con todas las dependencias necesarias.
- docker-compose.yml: Orquesta los servicios.

Entrega

Se adjunto colecciones de postman.



El código fue subido a un repositorio público de GitHub para su revisión y descarga. https://github.com/Hector-Frias/globalTec