FACULTAD DE INGENIERÍA
DEPARTAMENTO DE TECNOLOGÍA DE INFORMACIÓN Y COMUNICACIONES

Taller: Backend - NodeJS
Docente: Leonardo Bustamante – lfbustamante@icesi.edu.co
Entrega: Miércoles, Noviembre 20 de 2024
Objetivo:
El objetivo de este taller es traducir una aplicación existente que utiliza una API REST a una que utilice GraphQL. La nueva aplicación debe mantener todas las funcionalidades de la original, pero aprovechar las ventajas de GraphQL para mejorar la eficiencia y flexibilidad en la gestión de datos.

Requisitos de la Asignación:

Descripción:

Tomar una aplicación backend existente que utiliza una API REST para la gestión de usuarios y comentarios, y traducirla a una API GraphQL utilizando Node.js y TypeScript. La nueva API debe permitir a los usuarios realizar operaciones CRUD sobre los usuarios y los comentarios, manejar la autenticación, y gestionar roles específicos que controlen las acciones permitidas.

Requisitos Funcionales:

Gestión de Usuarios:

- Los usuarios con rol de superadmin pueden crear, modificar y eliminar usuarios.
- Implementar roles de usuario: superadmin, usuario regular.
- Los usuarios autenticados pueden ver la lista de otros usuarios, pero sólo el superadmin puede modificar o eliminar usuarios.

Módulo 1 – Gestión de Proyectos (o cualquier entidad principal):

- **Operaciones CRUD:**
    - Los usuarios autenticados pueden crear, visualizar, modificar y eliminar sus propios proyectos.
    - Los administradores podrán gestionar proyectos de cualquier usuario.
- **Relación con Otros Módulos: Los proyectos deben poder vincularse a**

elementos del módulo 2 (por ejemplo, tareas, actividades, usuarios, etc.).

Autenticación y Autorización:

- Implementar un sistema de registro y autenticación de usuarios utilizando JWT.
- Proteger las rutas CRUD con middleware de autenticación y autorización.
- Los usuarios deben autenticarse para realizar cualquier operación relacionada con usuarios, comentarios, o reacciones.

Entrega y Presentación:

- El código fuente debe estar en un repositorio de GitHub, con un README claro sobre cómo configurar y ejecutar el proyecto, además de una descripción de la funcionalidad del mismo, que elementos no alcanzaron a ser desarrollados o dificultades encontradas.

- El proyecto debe ser desplegado en nube.

- Se debe incluir un archivo que permita revisar las funcionalidades implementadas.

Criterios de Evaluación:

- **Implementación de GraphQL (50%)**:
    - **Consultas y Mutaciones (30%)**: Implementación correcta de consultas y mutaciones para todas las operaciones CRUD.
    - **Fragments (10%)**: Uso efectivo de fragments para reutilizar partes de las consultas y evitar duplicación de código.
    - **Manejo de Errores (10%)**: Manejo efectivo de errores en las consultas y mutaciones.
- **Calidad del Código y Uso de TypeScript (15%)**: Código bien organizado, tipado fuertemente y comentado.
- **Funcionalidad y Validaciones (15%)**: Cumplimiento de todas las funcionalidades, incluyendo validaciones de entrada.
- **Seguridad en Autenticación y Autorización (10%)**: Seguridad en el acceso

a rutas y operaciones CRUD.

- **Documentación y Presentación (10%)**: Documentación completa (README de ejecución, endpoints y tipos de documentos que reciben) y presentación clara y detallada.