



## Creando un asistente para la educación:

### Objetivos:

Aprender cómo incorporar diferentes tecnologías para tener un ecosistema eficiente y descentralizado, que se pueda acoplar a diferentes sistemas por medio de APIs.

**Nombre del Proyecto:** "Sistema de Gestión de Usuarios y Perfiles"

**Descripción del Proyecto:** El proyecto consiste en desarrollar un sistema web que permita a los usuarios registrarse, iniciar sesión y gestionar sus perfiles. Los usuarios pueden tener roles diferentes, como "Usuario normal" y "Administrador". Además, el sistema debe almacenar información de usuarios en una base de datos MongoDB y utilizar una base de datos relacional (por ejemplo, MySQL o PostgreSQL) para almacenar información adicional relacionada con los perfiles de usuario.

### Funcionalidades Principales:

1. **Registro de Usuarios:** Los usuarios pueden registrarse proporcionando su nombre de usuario, correo electrónico y contraseña. La contraseña debe ser cifrada antes de almacenarse en la base de datos.
2. **Inicio de Sesión:** Los usuarios registrados pueden iniciar sesión utilizando su correo electrónico y contraseña.
3. **Gestión de Perfiles:** Los usuarios pueden completar su perfil, proporcionando información adicional como nombre completo, fecha de nacimiento, foto de perfil, etc. Esta información se almacena en la base de datos relacional.
4. **Roles de Usuario:** Los usuarios pueden tener roles diferentes, como "Usuario normal" y "Administrador". Los administradores tienen permisos adicionales, como la capacidad de ver y editar perfiles de otros usuarios.
5. **Autenticación JWT:** Implementa la autenticación basada en JWT (JSON Web Tokens) para permitir que los usuarios inicien sesión de forma segura y accedan a rutas protegidas.

### Tecnologías Requeridas:



## Desarrollo Web, Proyecto Final

1. Frontend: Utiliza HTML, CSS y JavaScript (puedes usar un framework como React o Vue.js si lo prefieres).
2. Backend: Utiliza Node.js y Express.js para crear el servidor.
3. Base de datos MongoDB para el almacenamiento de datos de usuario.
4. Base de datos relacional (MySQL, PostgreSQL, etc.) para el almacenamiento de datos de perfil de usuario.
5. Implementa la autenticación con JWT para proteger las rutas relevantes.
6. Utiliza bibliotecas de validación de datos en el backend (por ejemplo, `express-validator` para Express.js) para validar datos de entrada.
7. Investigue la implementación de redis en base de datos en memoria para evitar usar cookies para tener información de las sesiones para administrar las sesiones de usuario.