

Este proyecto es una **API RESTful** construida con **Node.js** y **Express**, cuyo propósito principal es gestionar autenticación y autorización de usuarios utilizando **JSON Web Tokens (JWT)**. Está diseñado para ser el backend de una aplicación cliente, facilitando el acceso seguro a recursos protegidos.

Estructura Principal del Proyecto

- **index.js**
Archivo principal que inicia el servidor Express y configura las rutas y middleware necesarios. Aquí se definen los endpoints iniciales y se levanta el servidor.
- **config/**
Carpeta que contiene configuraciones globales, como:
 - **config.js**: Define valores clave como la clave secreta para el JWT, puerto, etc.
- **routes/**
Incluye las rutas de la API.
 - **auth.routes.js**: Maneja rutas relacionadas con login, registro y verificación de tokens.
 - **user.routes.js**: Contiene rutas protegidas, accesibles solo con un token válido.
- **controllers/**
Define la lógica que se ejecuta cuando se accede a cada ruta.
 - **auth.controller.js**: Controla el inicio de sesión y la creación de tokens.
 - **user.controller.js**: Devuelve datos protegidos al usuario autenticado.
- **middlewares/**
Contiene funciones que se ejecutan antes de acceder a ciertas rutas.
 - **auth.jwt.js**: Middleware que valida el JWT en las solicitudes.
- **models/**
Define la estructura de los datos. En este caso, simula una base de datos o usa una estructura para usuarios.
- **package.json**
Archivo que especifica las dependencias del proyecto como express, jsonwebtoken, body-parser, etc., y scripts para ejecutar el servidor.

Funcionamiento General

1. **Registro/Login:**

Un usuario se registra o inicia sesión a través de los endpoints correspondientes (/api/auth/login o /api/auth/signup).

2. **Generación de Token:**

Al hacer login, el servidor genera un token JWT que se devuelve al cliente.

3. **Acceso a Rutas Protegidas:**

El cliente usa ese token en futuras peticiones para acceder a rutas que requieren autenticación (por ejemplo, /api/user/data).

4. **Validación del Token:**

El middleware verifica que el token sea válido antes de permitir el acceso a esas rutas.