API SENA - Documentación del Proyecto

# Introducción

Este proyecto corresponde a la entrega del API desarrollado en Node.js con Express, MongoDB y autenticación JWT. Incluye el CRUD de usuarios y productos, así como el mecanismo de login seguro con tokens.

# Estructura de Archivos

El proyecto se organiza de la siguiente manera:

- server.js → Archivo principal del servidor Express.  
- config/db.js → Conexión con la base de datos MongoDB.  
- models/Usuario.js → Modelo de usuario con contraseña encriptada.  
- models/Producto.js → Modelo de producto con nombre, precio y descripción.  
- routes/usuarios.js → Rutas CRUD para usuarios.  
- routes/productos.js → Rutas CRUD para productos (protegidas por JWT).  
- routes/auth.js → Ruta de login que genera el token JWT.  
- middleware/verificarToken.js → Middleware para validar el token antes de acceder a rutas protegidas.  
- .env.example → Archivo de ejemplo con variables de entorno.  
- package.json → Dependencias y scripts del proyecto.  
- README.md → Instrucciones para ejecutar el proyecto.

# Modelos

## Usuario.js

Define la estructura de un usuario con los campos: nombre, correo, edad y contraseña. La contraseña se guarda encriptada con bcrypt.

## Producto.js

Define la estructura de un producto con los campos: nombre, precio y descripción.

# Rutas

## Usuarios

- POST /usuarios → Registrar nuevo usuario (contraseña encriptada).  
- GET /usuarios → Listar todos los usuarios.  
- GET /usuarios/:id → Obtener un usuario por ID.  
- PUT /usuarios/:id → Actualizar un usuario.  
- DELETE /usuarios/:id → Eliminar un usuario.

## Productos

Rutas protegidas con JWT:  
- POST /productos → Crear producto.  
- GET /productos → Listar productos.  
- PUT /productos/:id → Actualizar producto.  
- DELETE /productos/:id → Eliminar producto.

## Auth

- POST /login → Verifica credenciales y devuelve un token JWT válido por 1 hora.

# Middleware

verificarToken.js es un middleware que valida el token JWT. Si el token no existe o es inválido, bloquea el acceso a la ruta.

# Instrucciones de Uso

1. Clonar el proyecto o descomprimir el archivo ZIP.  
2. Crear un archivo .env con las variables:  
 - MONGO\_URI → cadena de conexión a MongoDB.  
 - JWT\_SECRET → clave secreta para firmar los tokens.  
3. Instalar dependencias con: npm install  
4. Iniciar el servidor con: npm run dev  
5. Probar rutas en Postman o Insomnia.