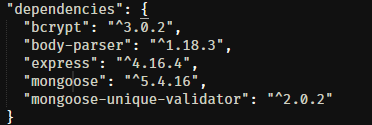
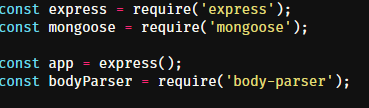
Creación de Rest Server

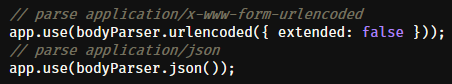
1. Para ello principalmente necesitaremos de las dependencias:



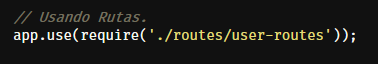
1. Importación de dependecias:



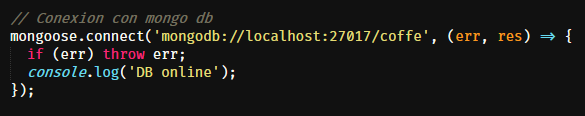
1. Obtener los elementos del body:



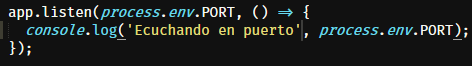
1. Importación de las rutas:



1. Connexión a la base de datos:



1. Iniciando el servidor:

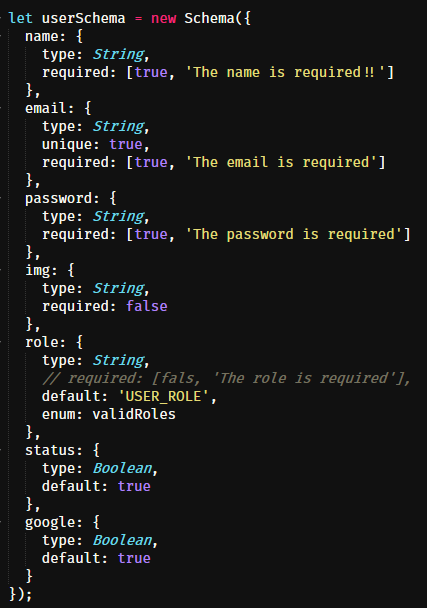


1. Configuración de modelos para la validación de datos en mongo db:
   1. Importación de dependencias:

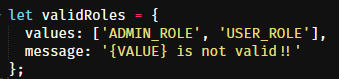


* 1. Uso esquema:





* 1. Validación de roles de usuario:



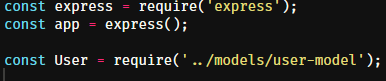
* 1. Validador de llave única:



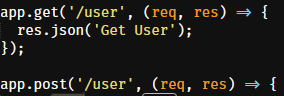
* 1. Exportación de modulo:



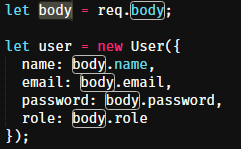
1. Configuración de rutas para el server:
   1. Importación de dependecias y modelos:



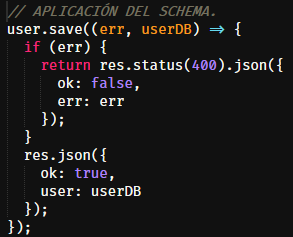
* 1. Ejemplo de usuario:



* 1. Toma de petición:



* 1. Guardando en bd por medio de Schema:



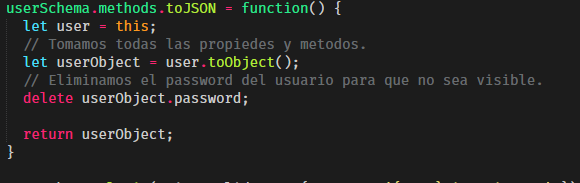
1. Encriptando el password:
   1. Instalar dependencia:



* 1. Importación del módulo:



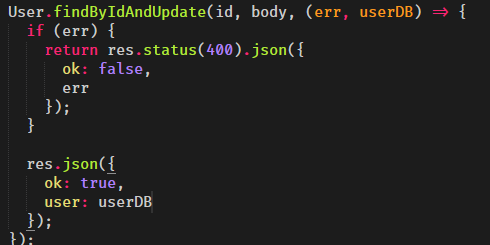
* 1. Eliminado el password del JSON:



1. Actualizando en la base de datos:
   1. Recibir parámetros y body:

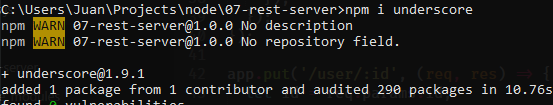


* 1. Guardando en BD:



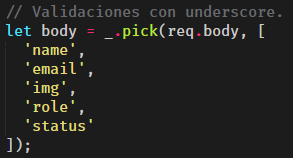
* 1. Obteniendo los datos actualizados:



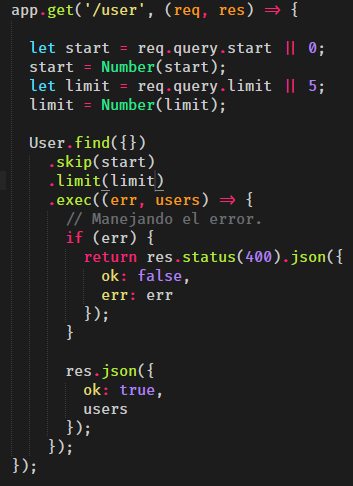
1. Validaciones adicionales en el PUT:
   1. Instalar dependencia “underscore”:
   2. Importar dependencia:



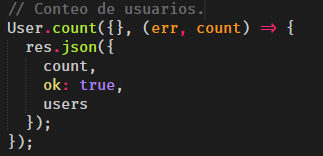
* 1. Crear validaciones:



1. Obtener todos los datos de forma paginada:



1. Retornar numero total de registros en una colección:



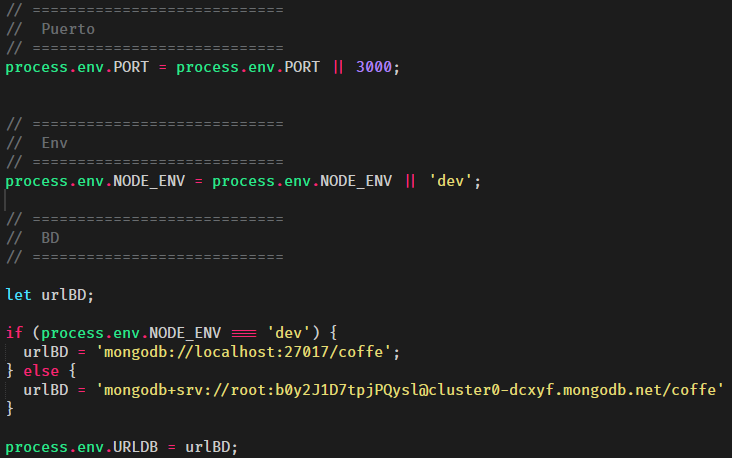
1. Filtrando los campos de los resultados de un get:

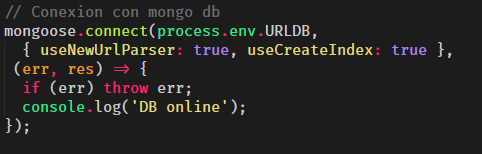


1. Delete: Borrando un usuario de la base de datos:

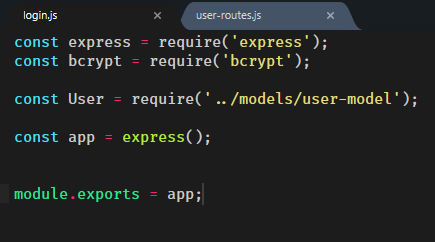


1. conectado con MongoAtlas:

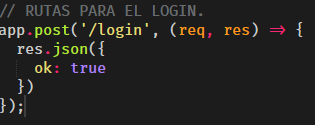




1. JWT Ordenado rutas del proyecto:
   1. Crear login.js



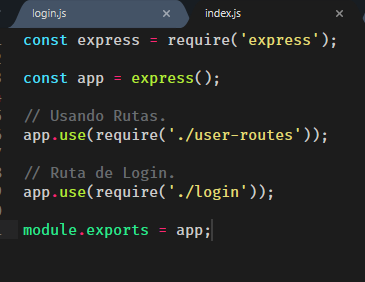
* 1. Crear ruta de login en login.js:



* 1. Declarar la ruta en server.js:



* 1. Crear index.js de rutas y hacer uso de las rutas en un solo archivo:



* 1. Server.js

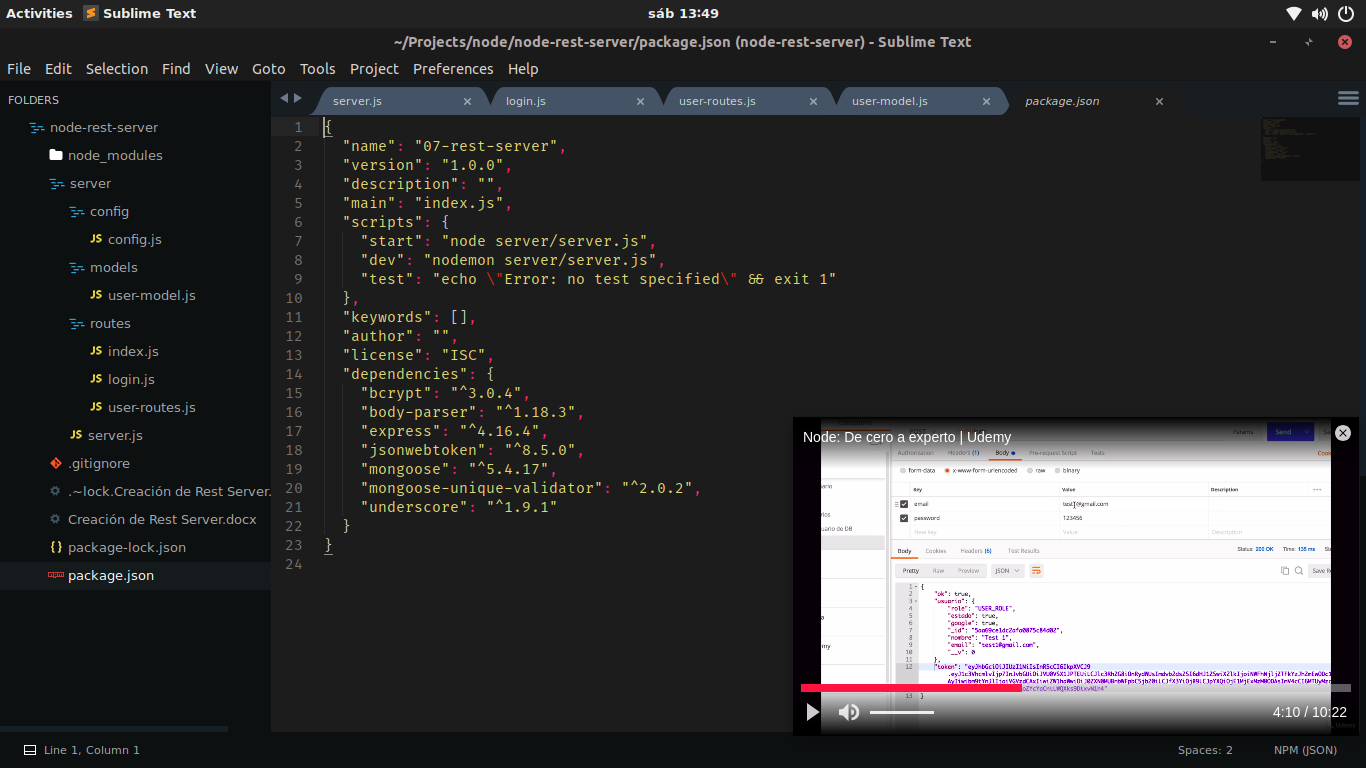


1. Login de usuario
   1. Buscar y validar un usuario mediante su email y su contraseña:



19. Crear un JTW:

a. Instalar dependencia:



b. importar dependecia:

// JTW.

**const jwt = require('jsonwebtoken');**

c. Gererar el Token:

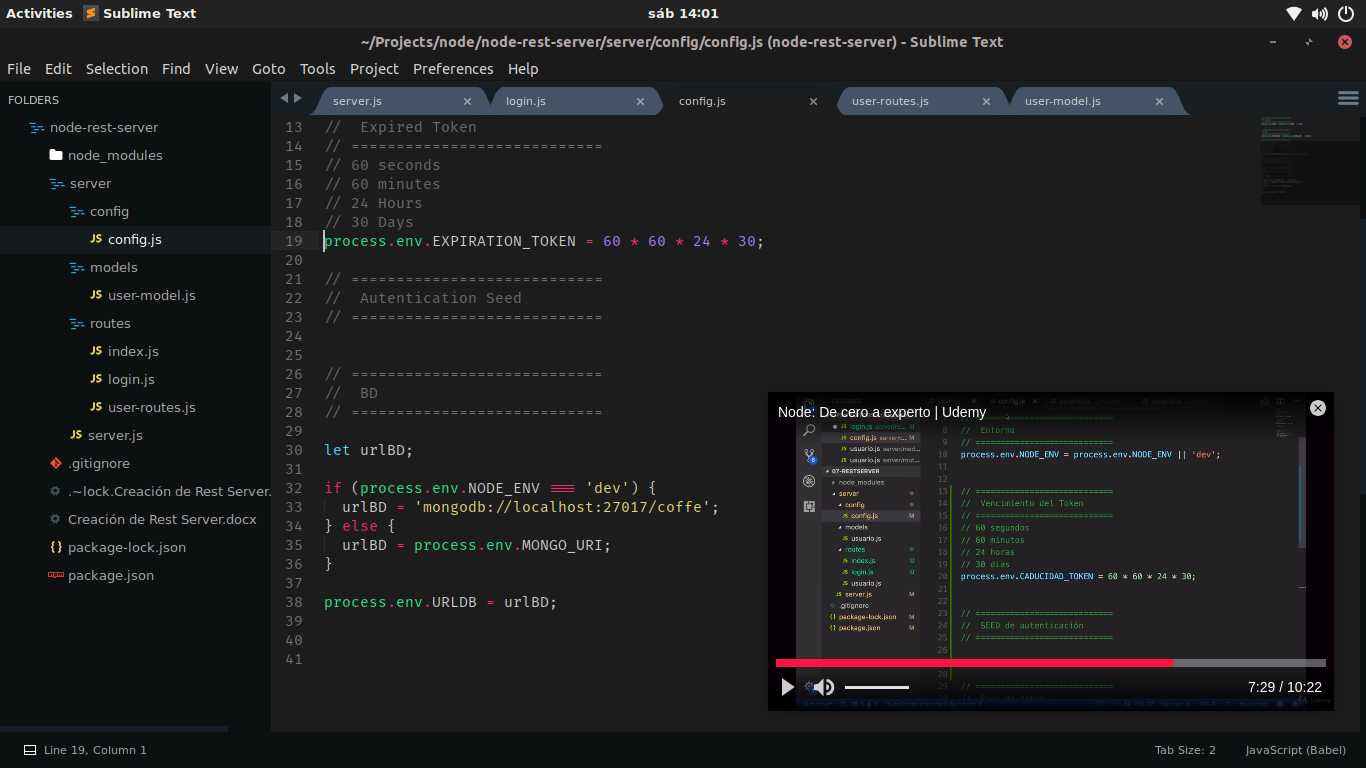
**// Generando Token.**

**let token = jwt.sign({**

**user: userDB,**

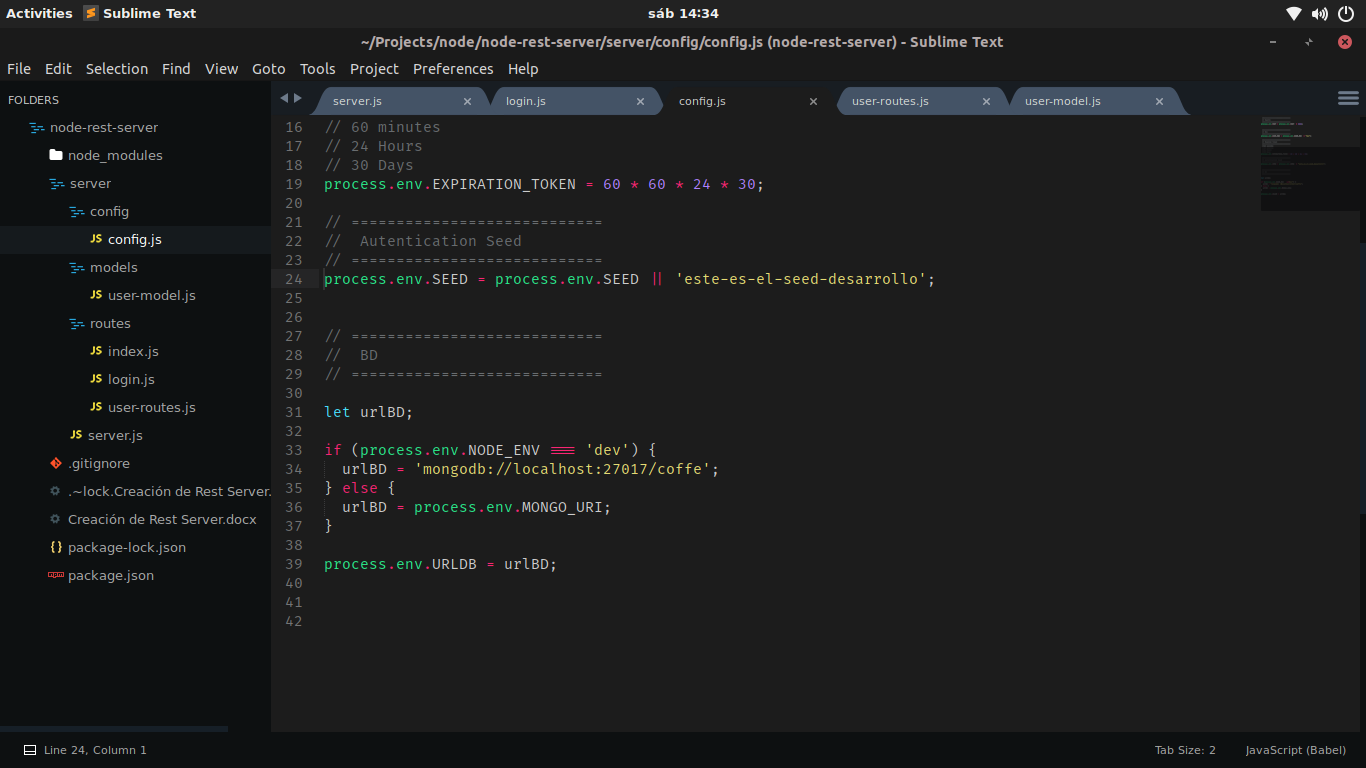
**}, 'este-es-el-seed-desarrollo', { expiresIn: 60 \* 60 \* 24 \* 30 } );**

d. Configurar las variables de entorno para la expiración del token:



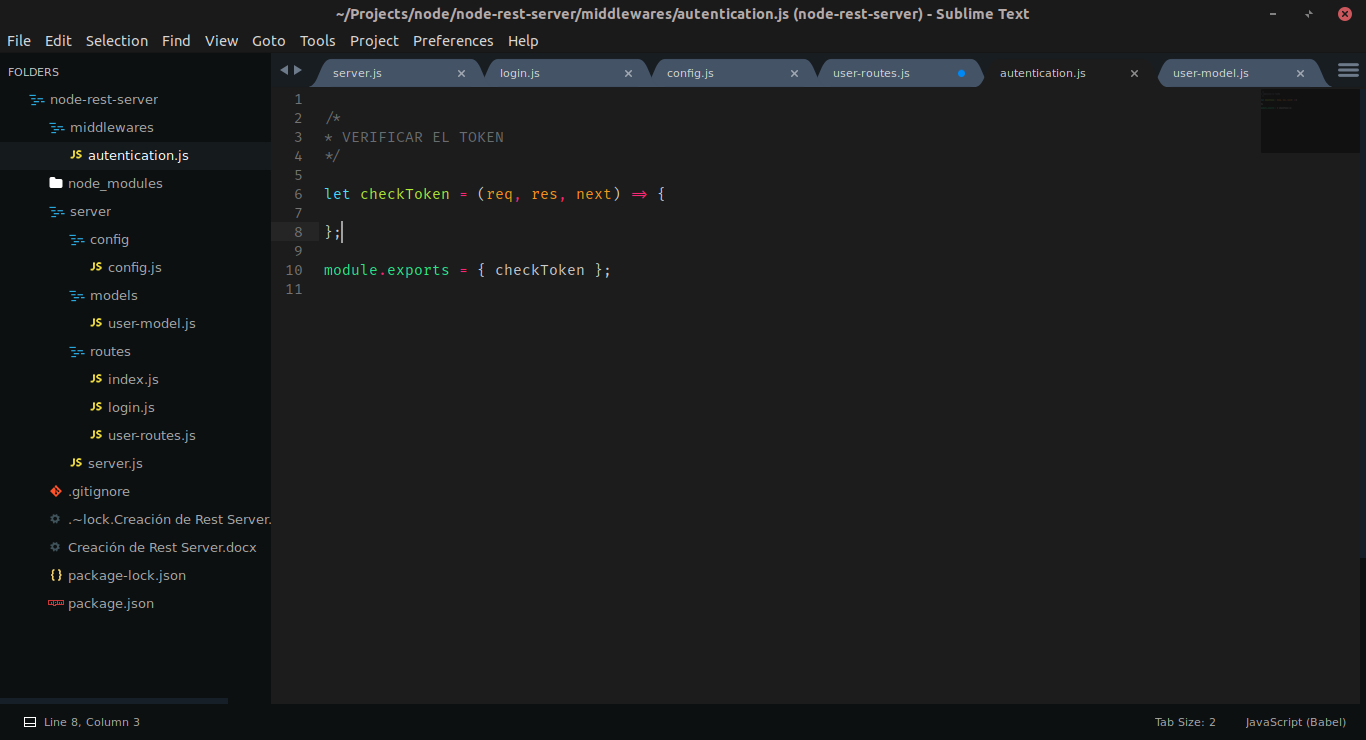
e. Configurara las variables de entorno para el seed:

Se recomienda crear una varible de entorno en heroku.

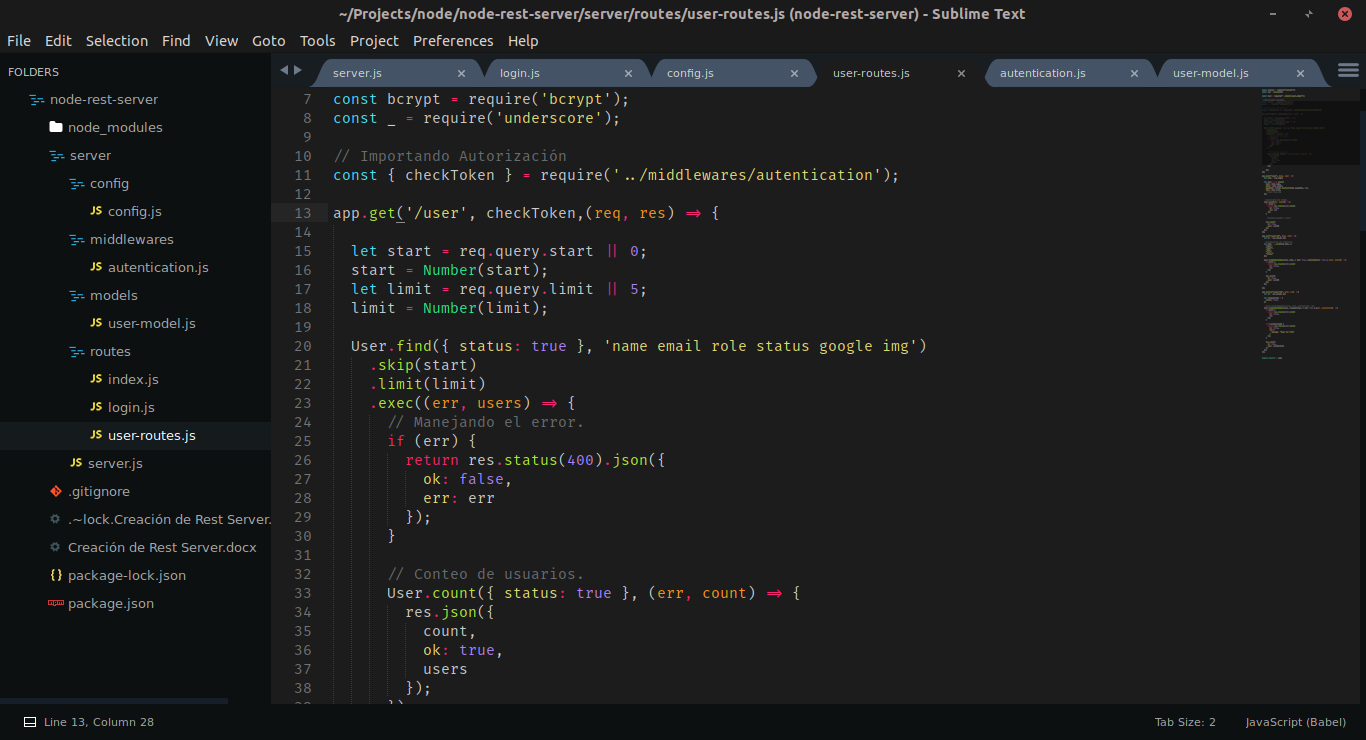


20. Proteger rutas mediante el uso de Token – Middlewares:

a. Crear Directorio de middlewares: dentro el archivp autentication:



b. Importar y utilizar el middleware en las rutas del usuario:



c. Vericando el token en autentication.js:

