confguracion de poryecto de node:

npm int -y

instalar nodemon //para refrscar el servidor caundo se hace cambios

"scripts": {

"start": "node index.js",

"dev": "nodemon index.js"

},

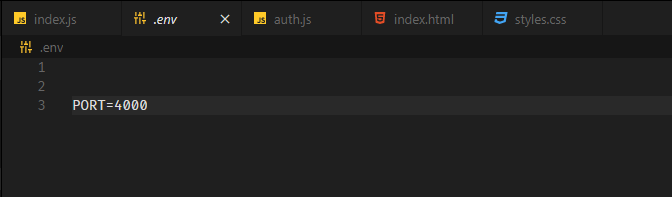
npn run dev (modo desarroloo)

condifguracion de express:

npm i express

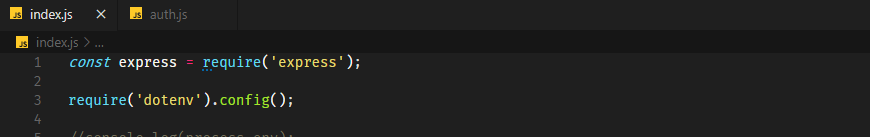
..configurar servdor y la primera rutas

CONFIGURAR LAS VARIABLES DE ENTORNO



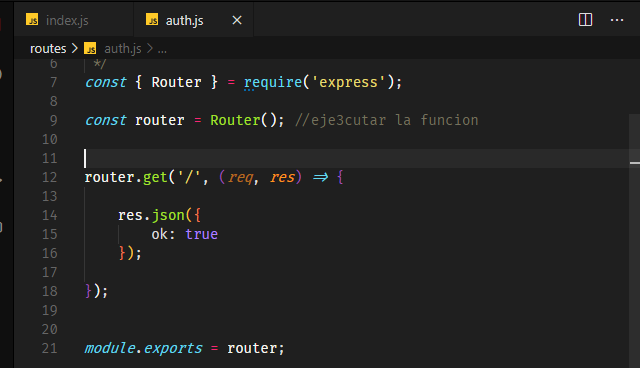
\*\*crear el archvo de entorno con la extension .env

\*\* instalar paquete .... npm i dotenv

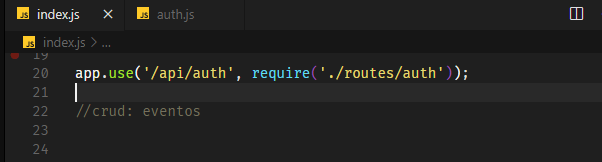


**creacion de rutas para los usuarios**

se crea las rutas de usuarios(auth.js)



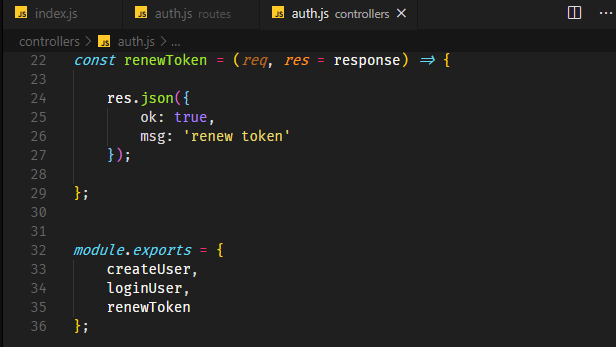
**//**se ussa la ruta creado en router/auth.js (http://localhost:4000/api/auth)



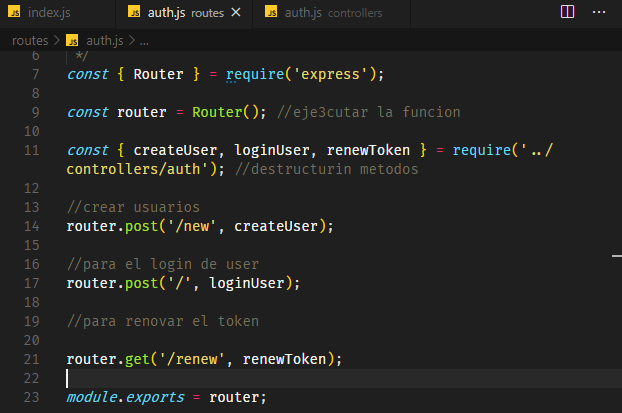
**endpoint remover, CREAR, LOGIN (separar la logica en controladores)**

**\*\*archivo controller (auth.js)**

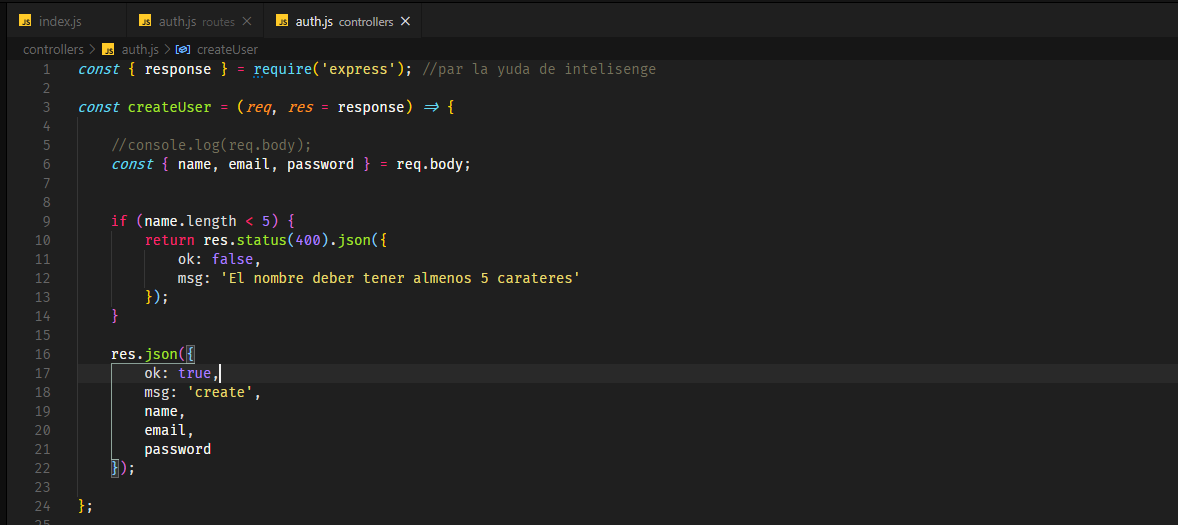


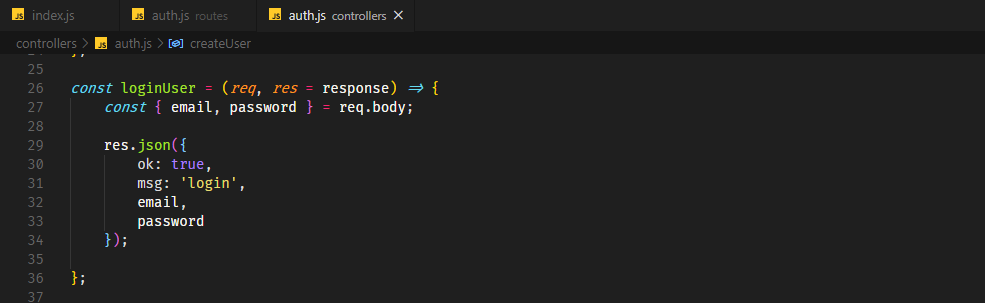


**\*\*\*routes (auth.js)**



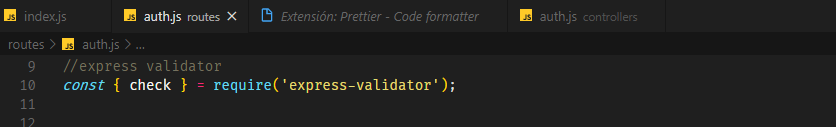
**recuéracion informacion de endpoint (post..crear usuario)**

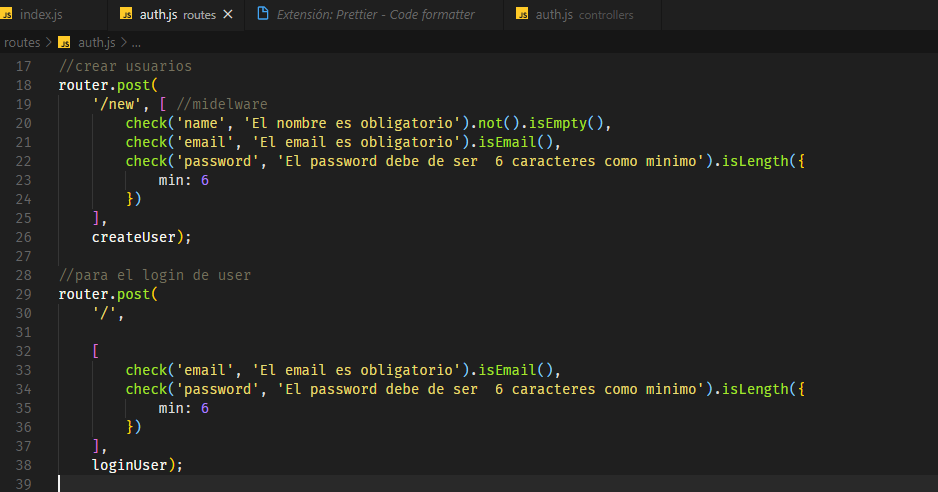


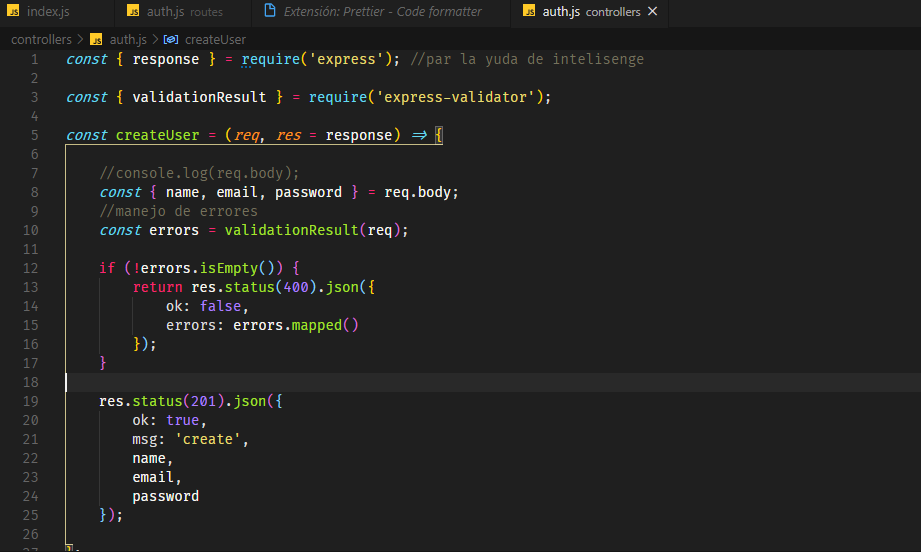


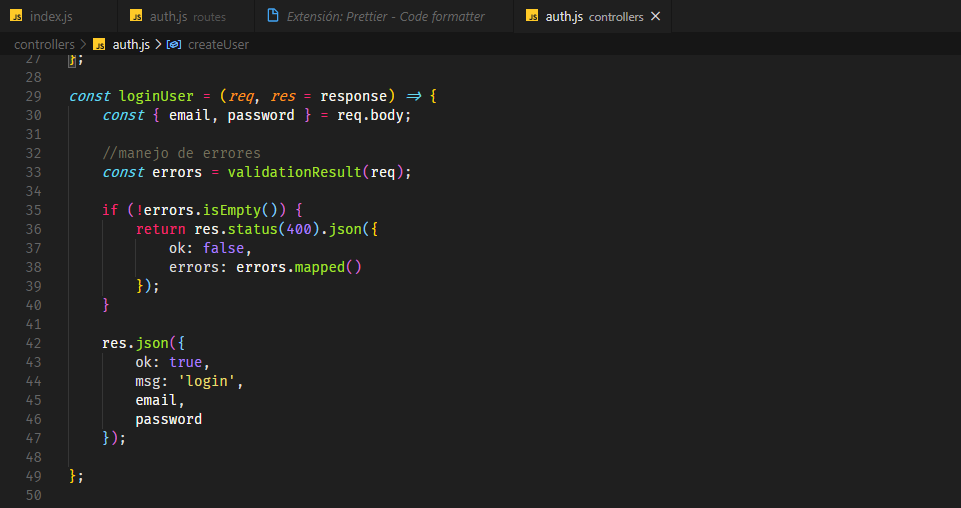
**express validator**

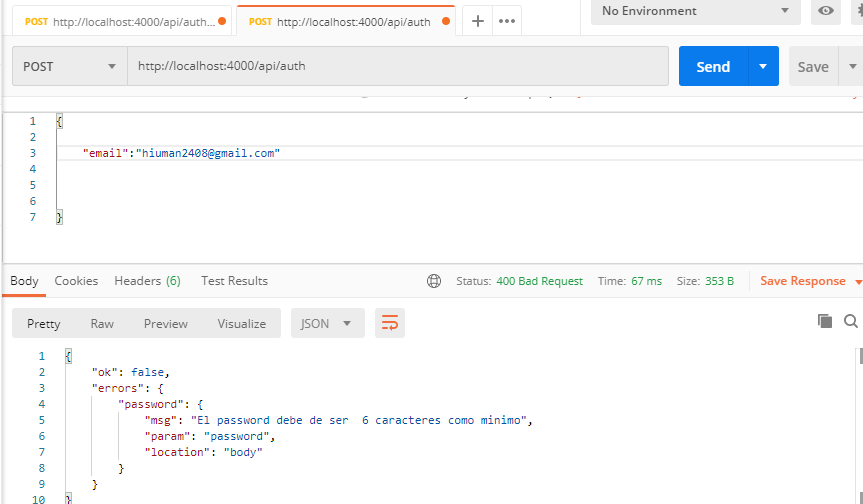
npm install --save express-validator



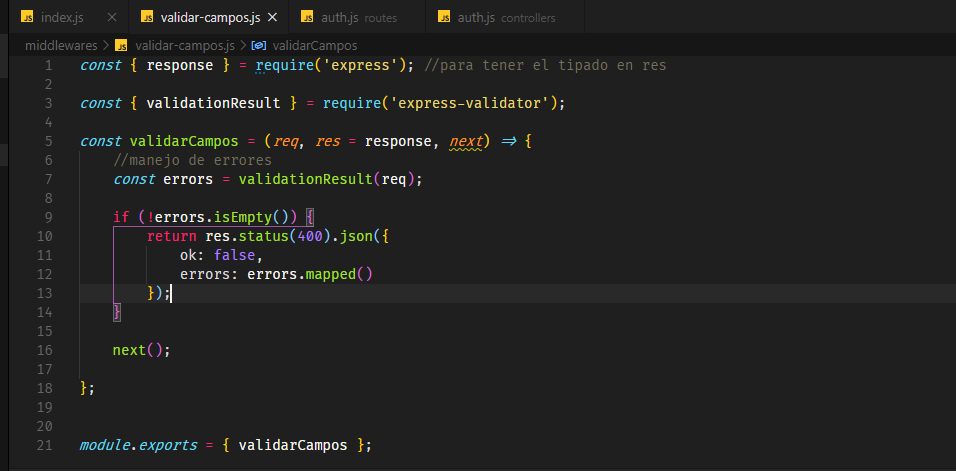


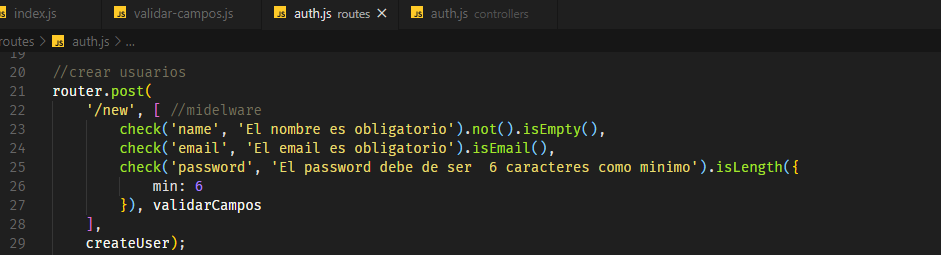


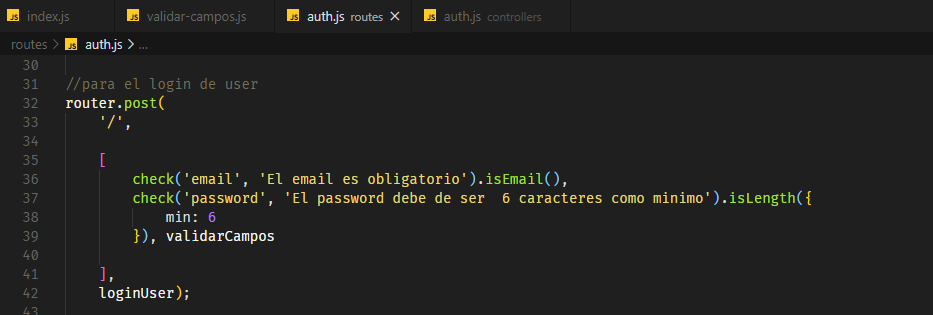




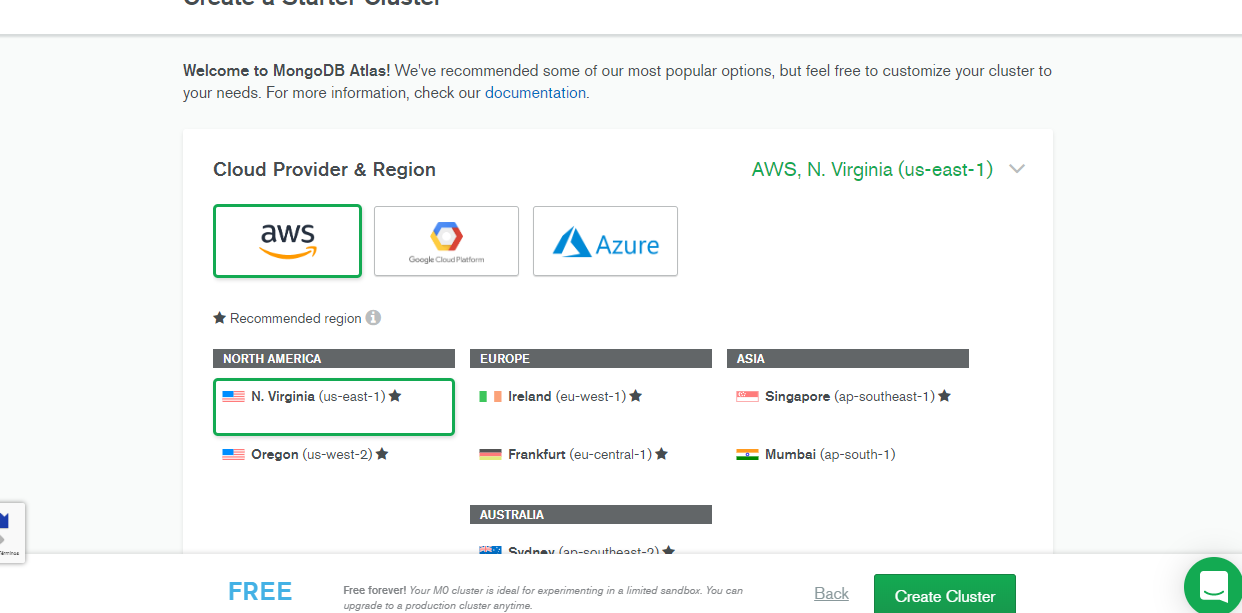
**modelware perdsonalizado**







**CONFIGURACION DE LA BASE DE DATOS (MONGODBaTLAS)**

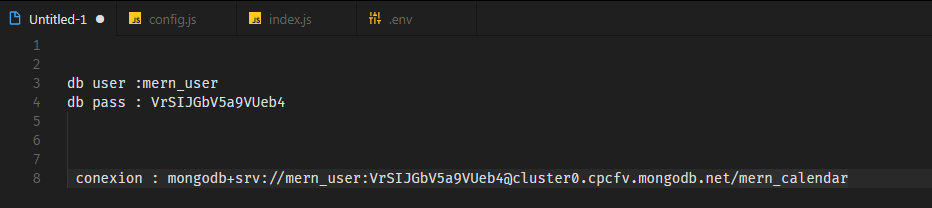


Conexion par el mongoCompas

db user :mern\_user

db pass : VrSIJGbV5a9VUeb4

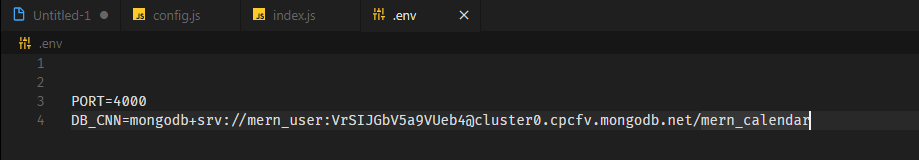
conexion : mongodb+srv://mern\_user:<password>@cluster0.cpcfv.mongodb.net

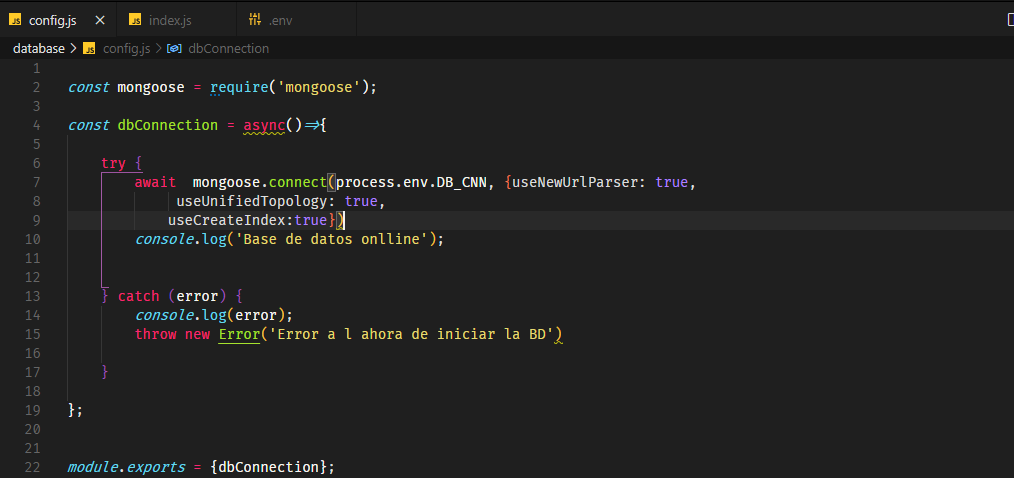


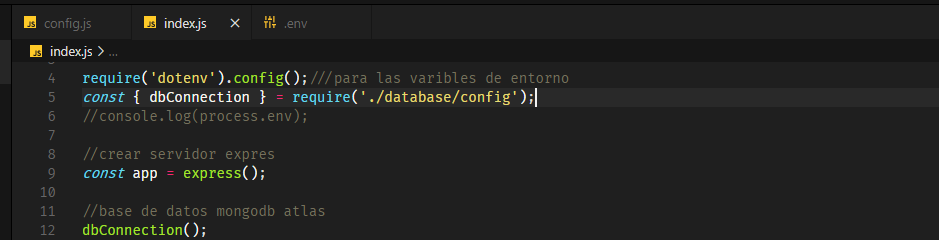
**CONECTAR NODEJS A MONGODB ATLAS**

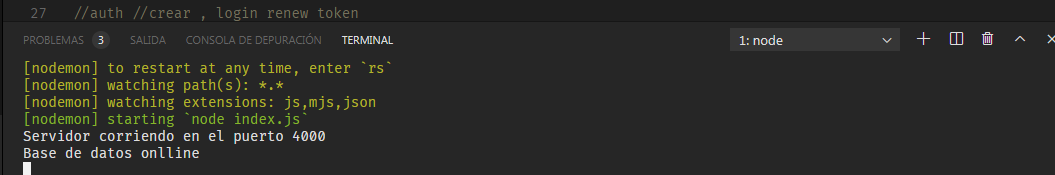
**Configurar moogosse**

npm install mongoose



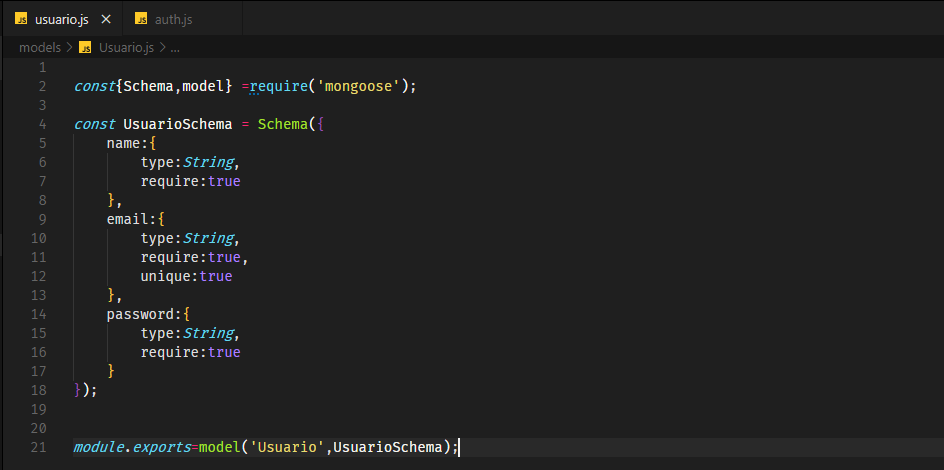




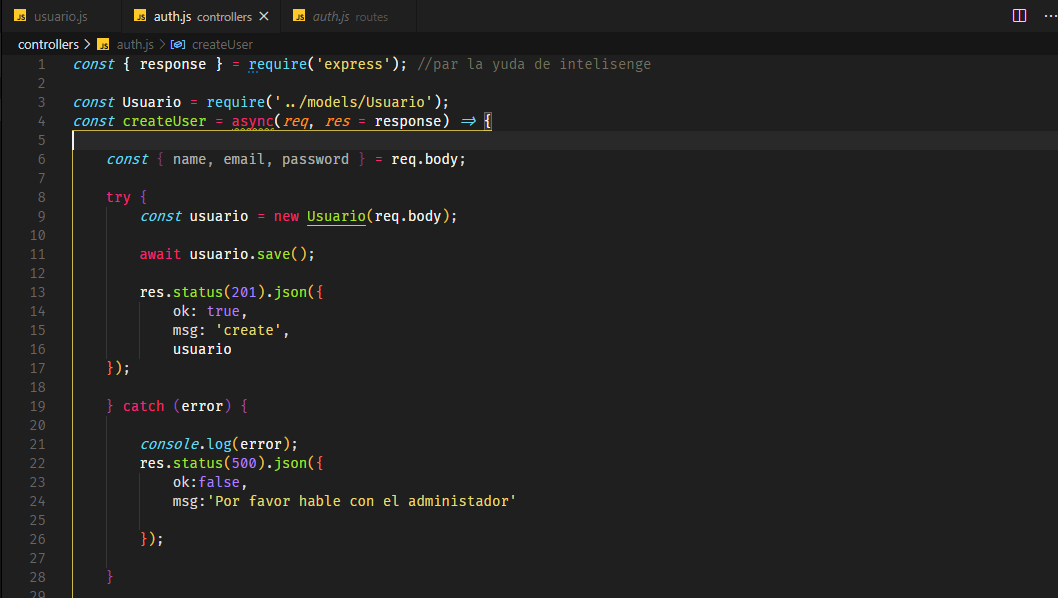


**CREAR USUARIO EN LA BASE DE DATOS**

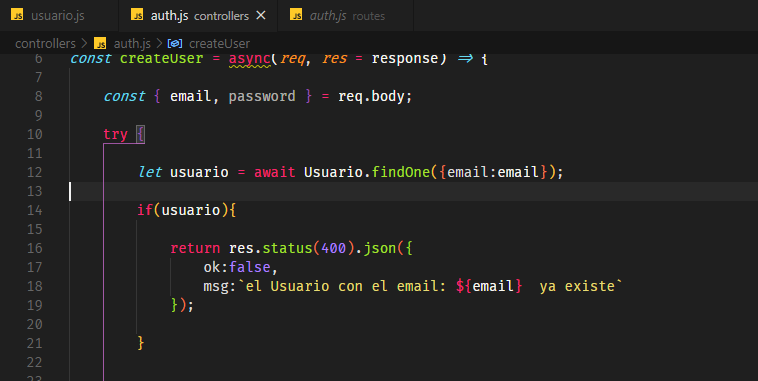
**\*\*\*crear un modelo Usuario.js**

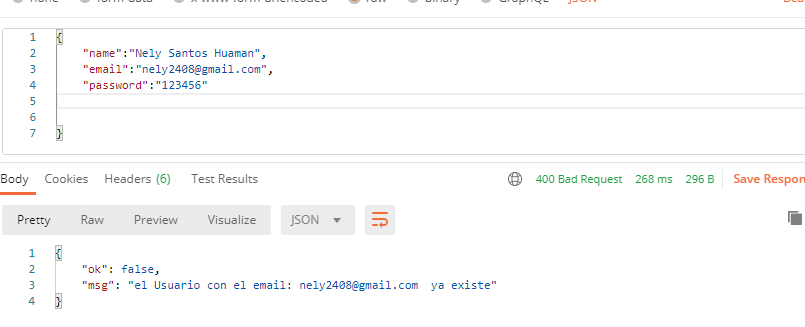


**\*\*auth.js (controler)**



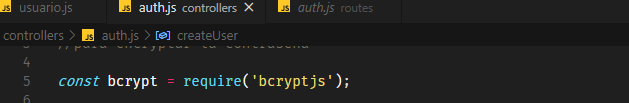
**VALIDACIONES DE USUARIOS CREADOS ( que el email sea único)**

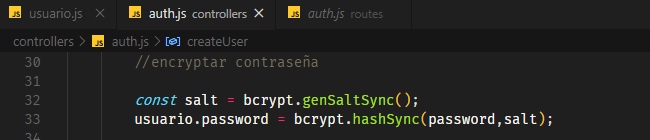


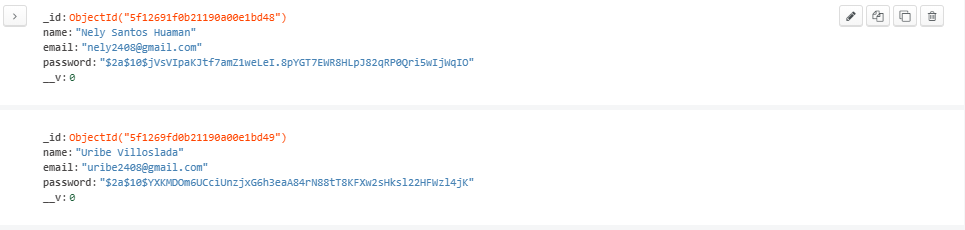


**ENCRYPTAR LA CONTRASEÑA**

npm install bcryptjs

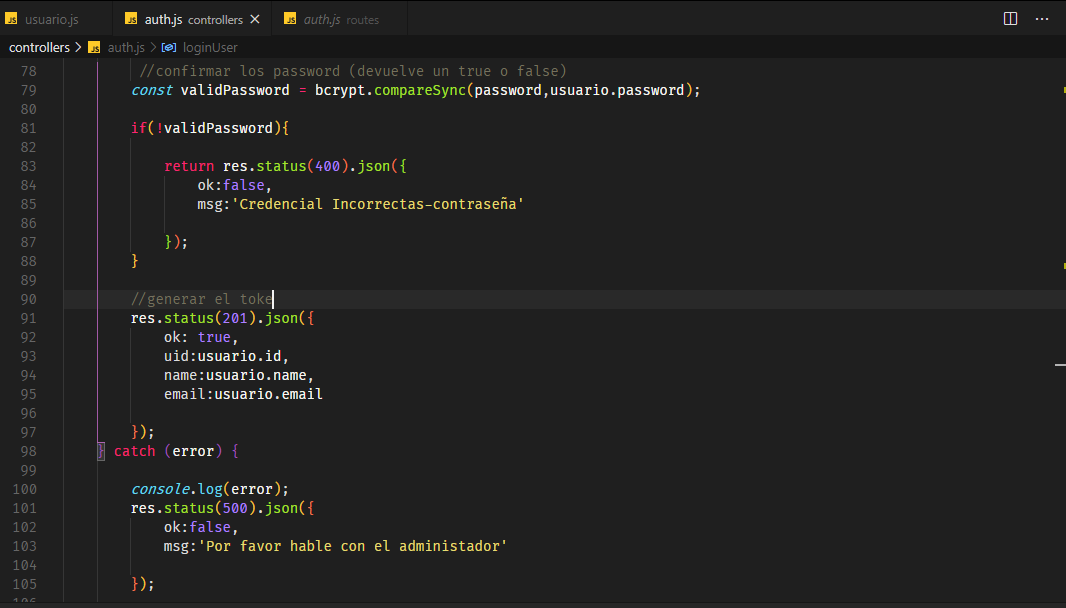






**LOGIN DE USUARIO**

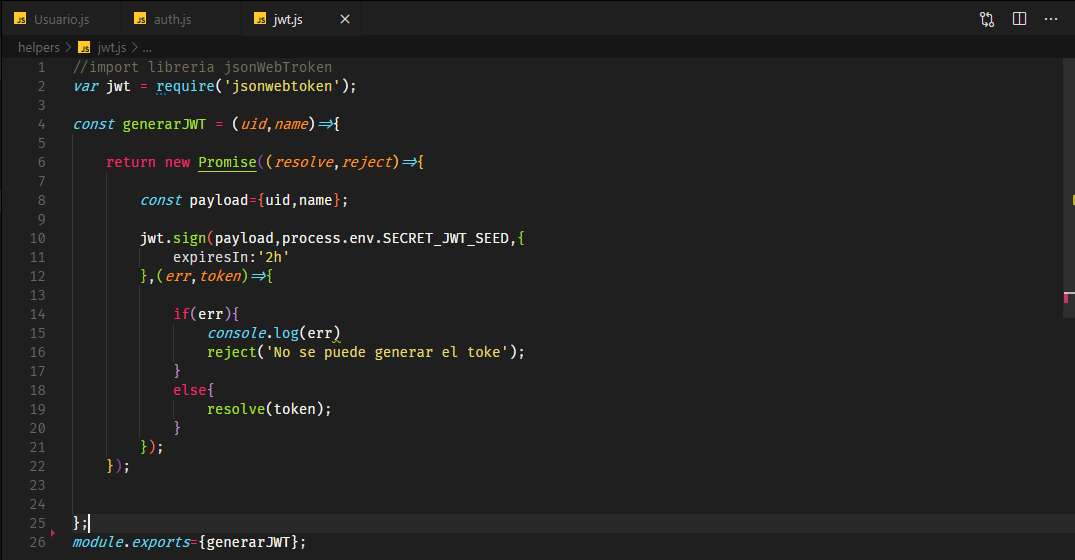


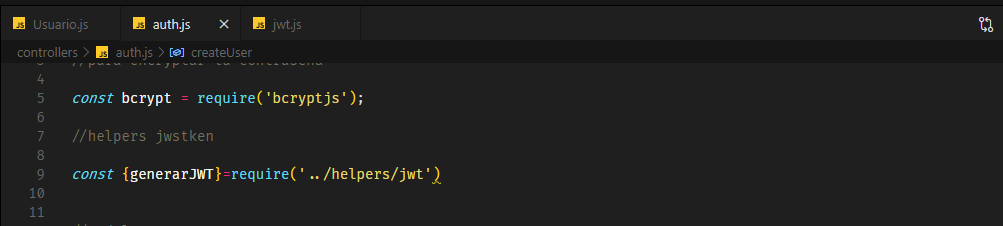


**GENERAR EL TOKEN (jsonwebtoken)**

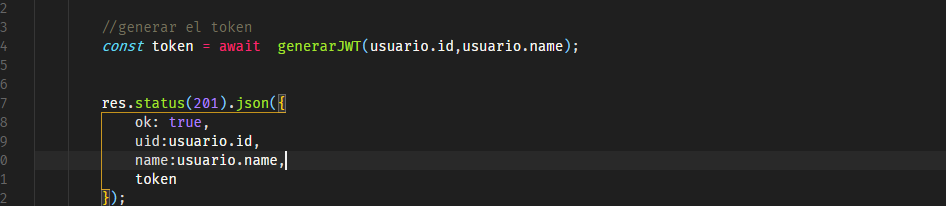
npm install jsonwebtoken

**\*\*crado un helpers (jwt)**

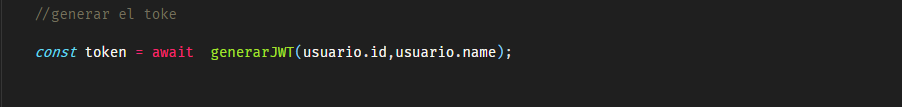


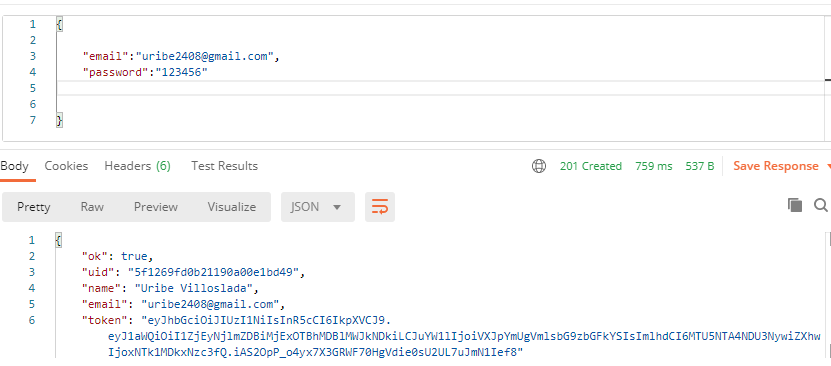


**\*\*crear usuario**



**\*\*\*login**

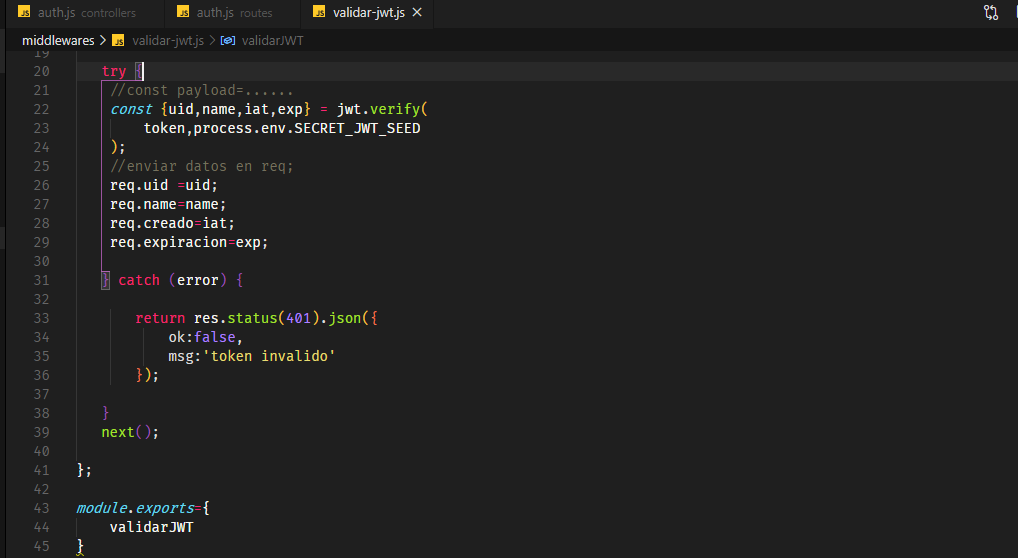




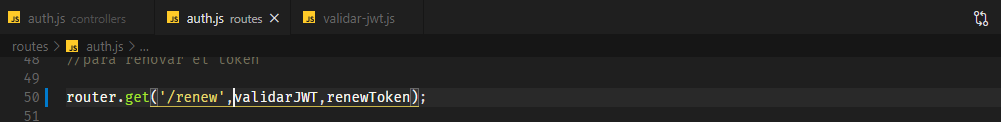
**RENOVAR TOKEN**

**\*\*\*crramos un midelware (revalidar-jwt.js)**

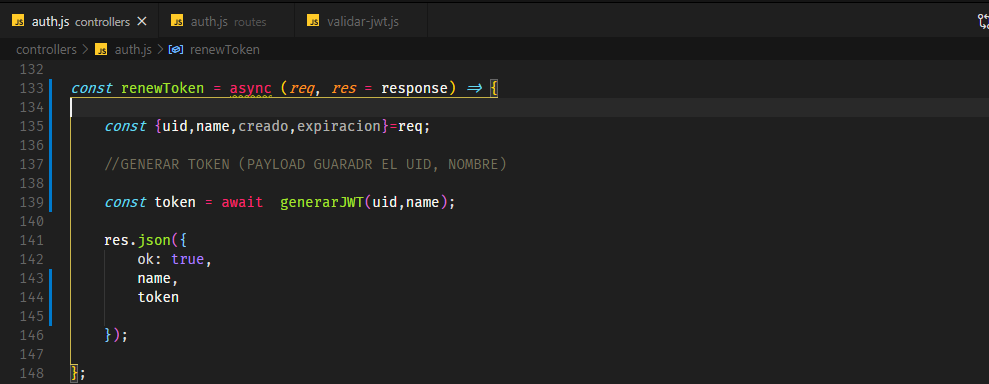




**\*\*\*los utilizmos en las routes**

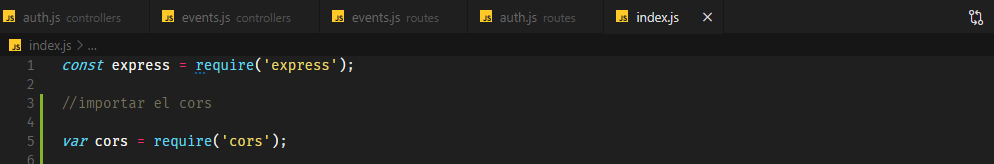


**\*\*validaos el token y generamos nuevo token en el controlador**



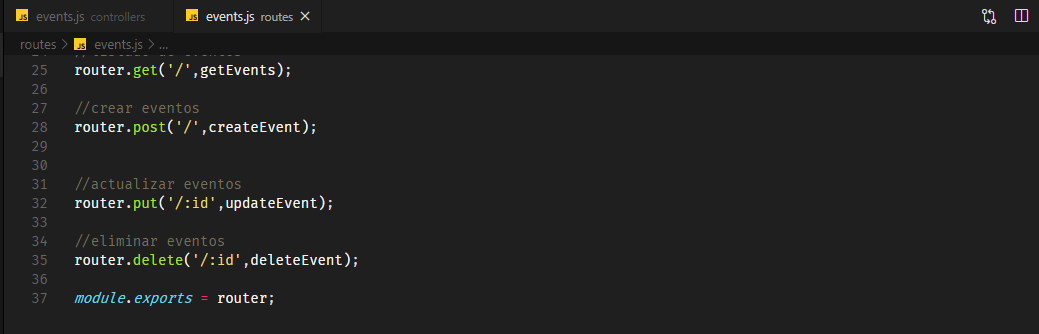
**CONFIGUAR CORS**

npm install cors

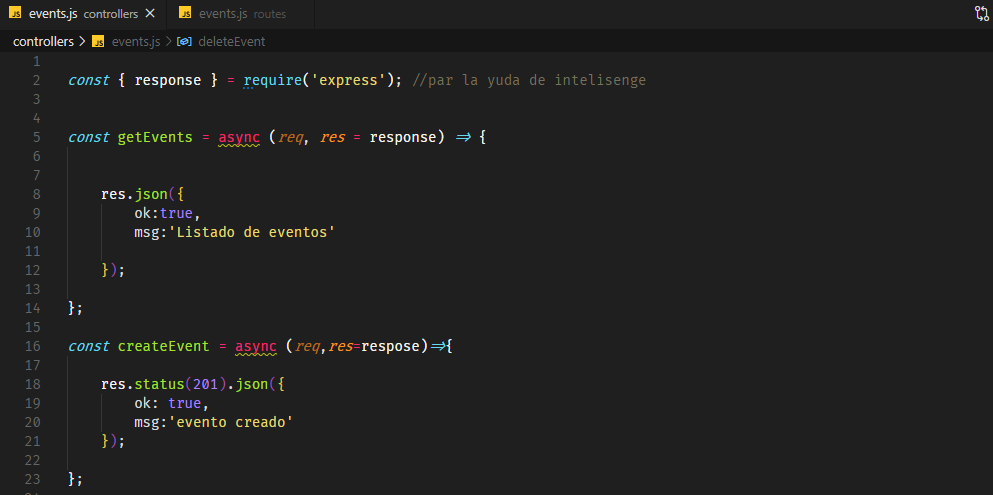


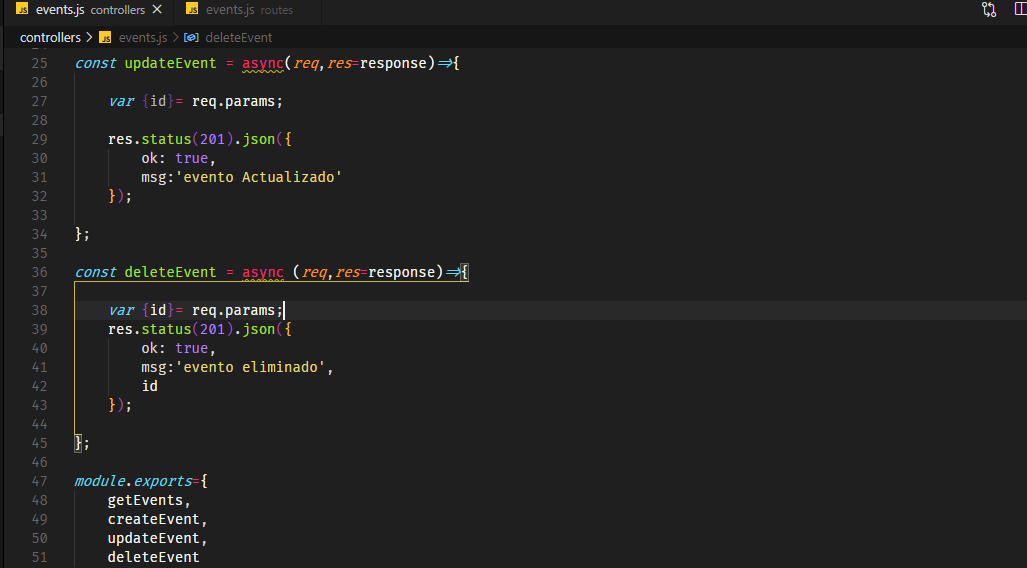
**CREAR ENDPOINT EVENTS (ROUTES)**



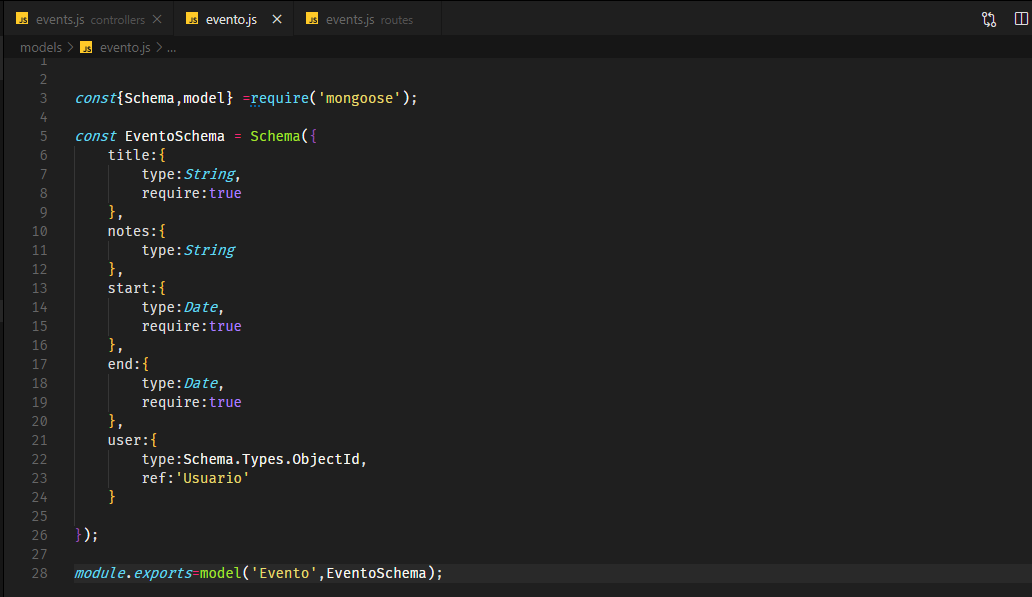


**CREAR CONTOLADOR EVENTS (SIMPLE)**



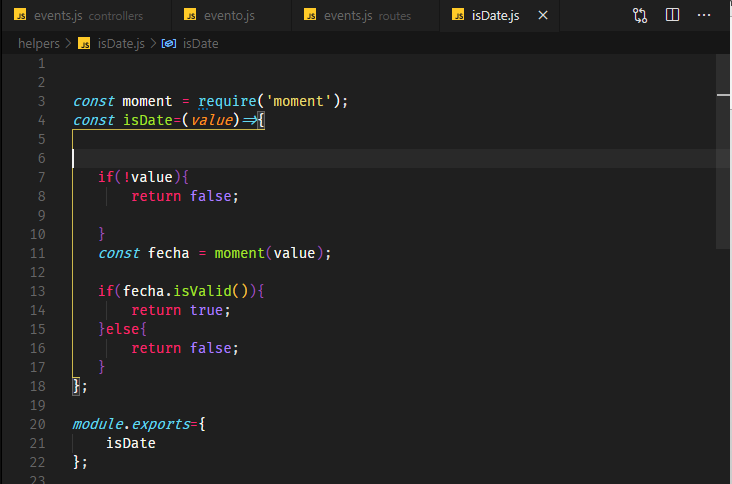


**CREAR EL MODEL EVENTS**

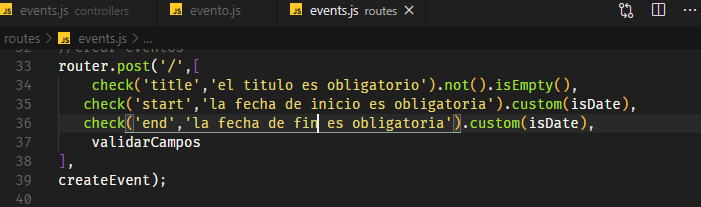


**VALIDAR CAMPOS NECESARIOS CREAR EVENTOS**

**\*\*creación helpers Isdate, para validar si es una fecha correcta**

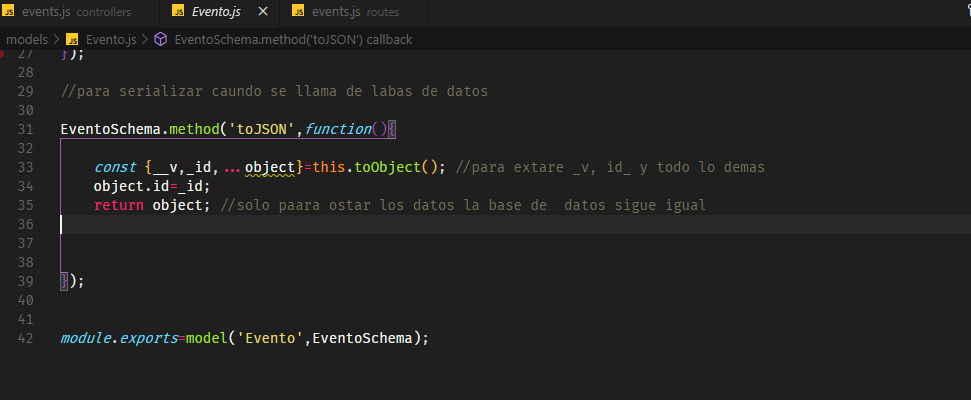


**\*\*configuar en la ruta crear evento**

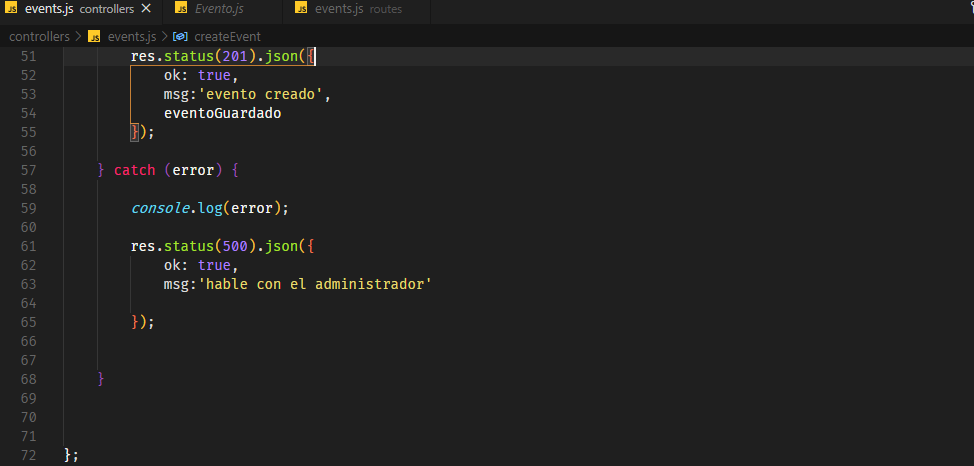


**GUARDAR EVENTO EN LA BASE DE DATOS**

**\*\*cambos en el modelo evento**



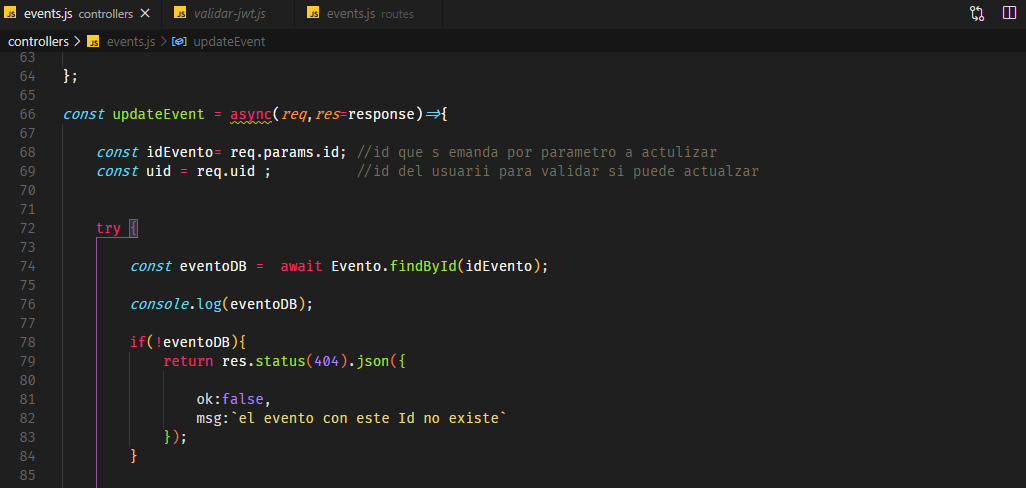


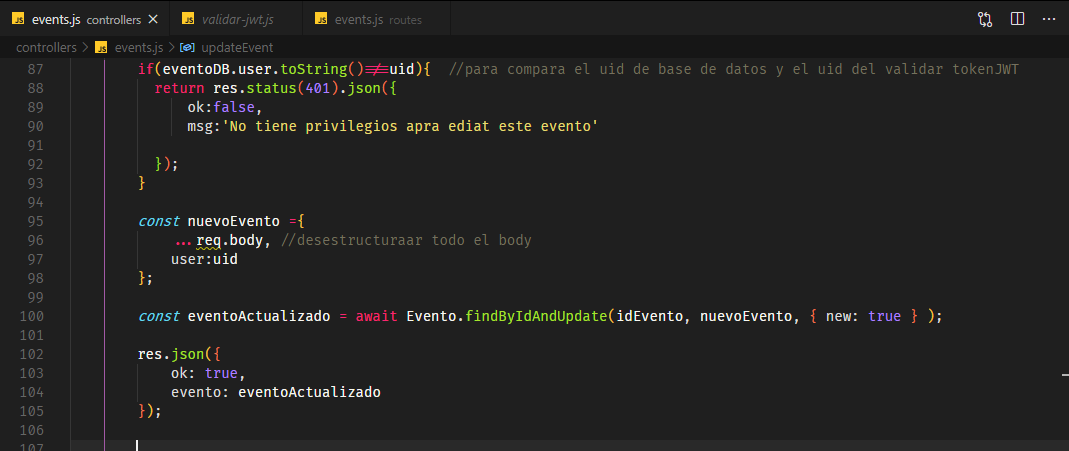


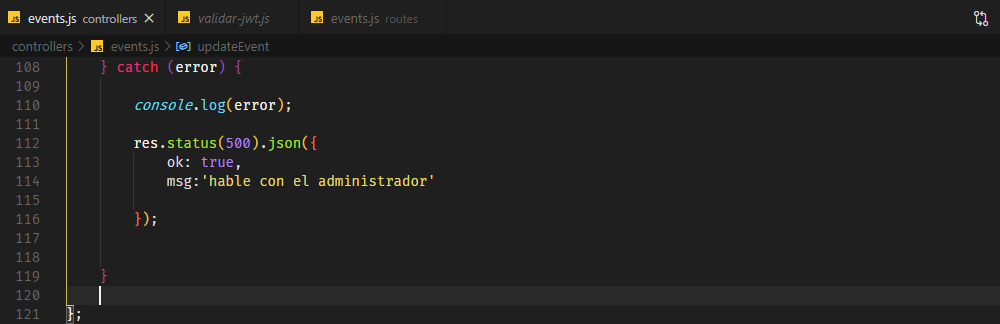
**LISTAR EVENTOS**



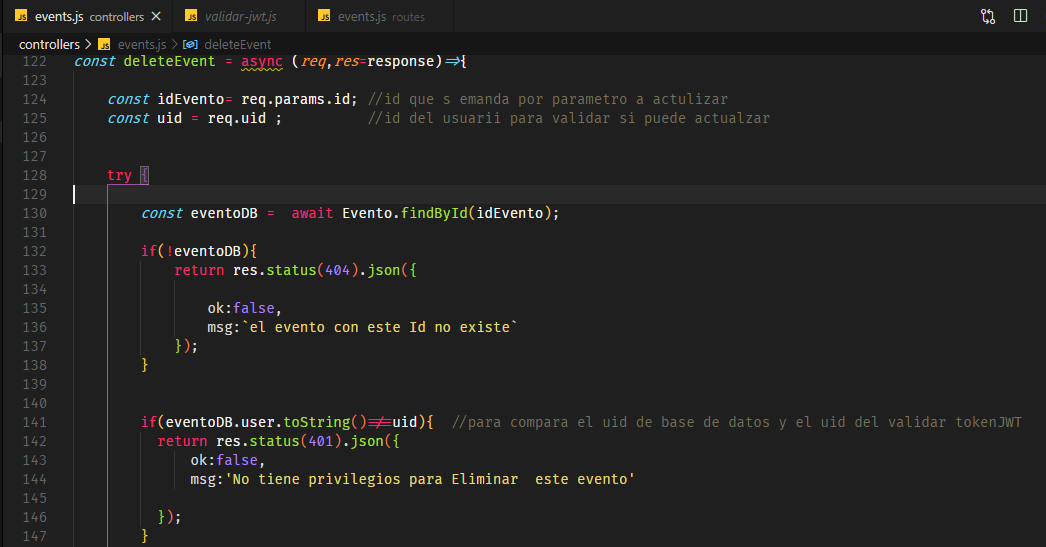
**ACTUALIZAR EL EVENTO**

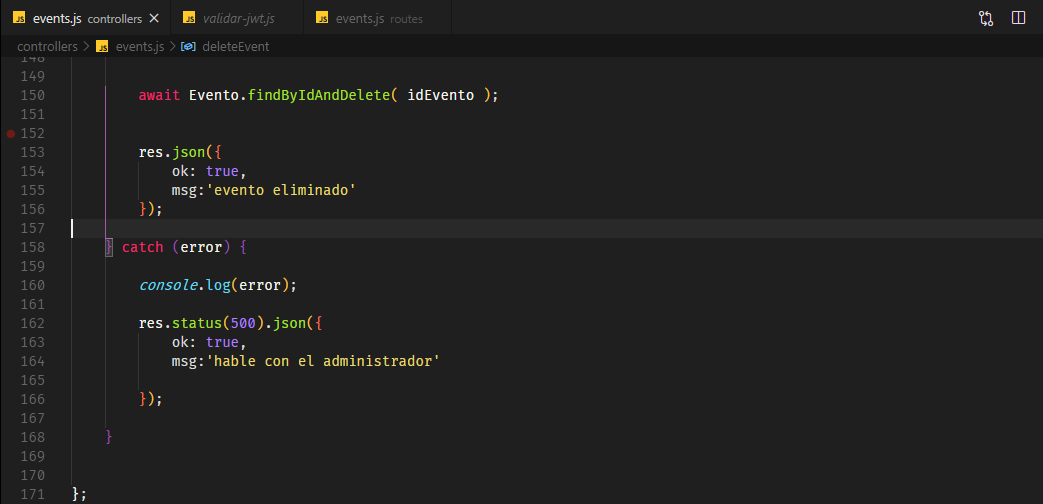




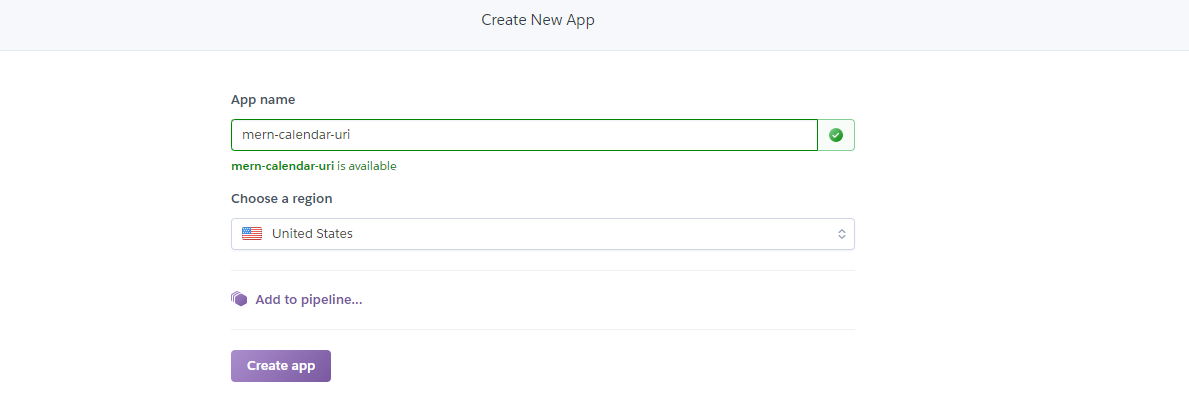


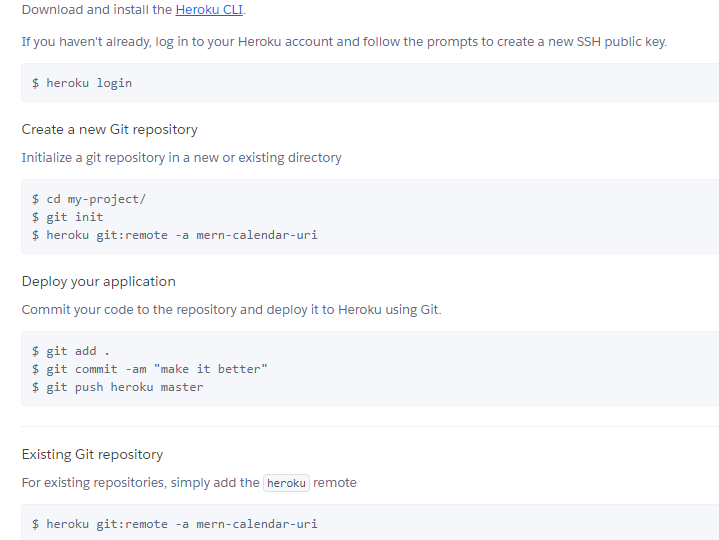
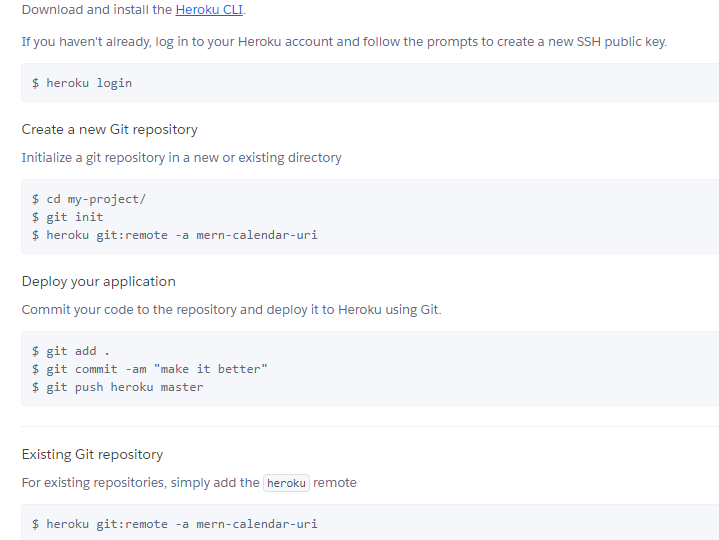
**ELIMINAR EVENTOS**





**DESPLEGAR BACKEND SERVER EN HEROKU**





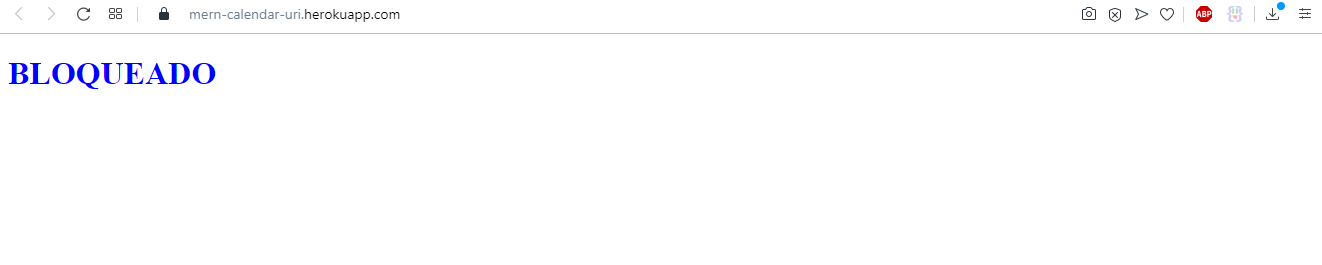
https://git.heroku.com/mern-calendar-uri.git

**si es que se quiere subir a heroku**

**git push heroku master**

**\*\*\*desplegando la aplicación en heroku**

**https://mern-calendar-uri.herokuapp.com**



**\*\*probando en postman desde heroku**