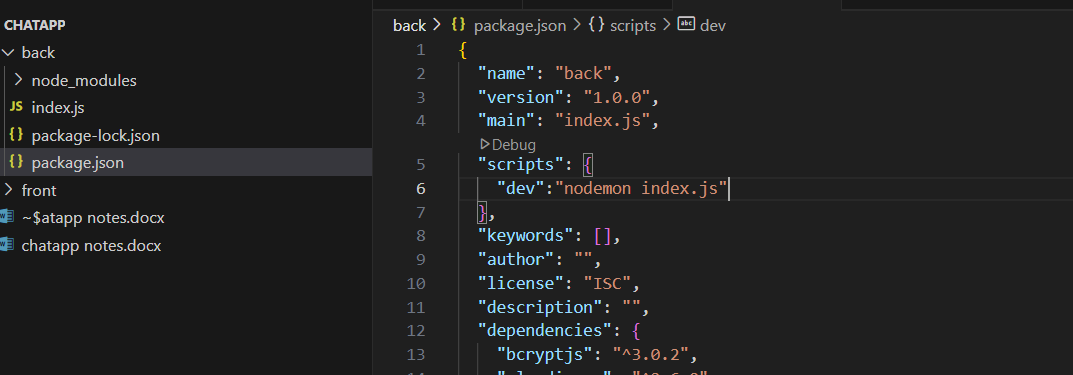
Npm create vite->react folders

Npm install-> node modules

Npm init -y=> package-json for backend

Npm i nodemon -D

Add this in package.json



App.use(“/api/auth”,”AuthRoutes”)

If users visit /api/auth authroutes is performed

Make authroutes.js router file

const router=express.Router();

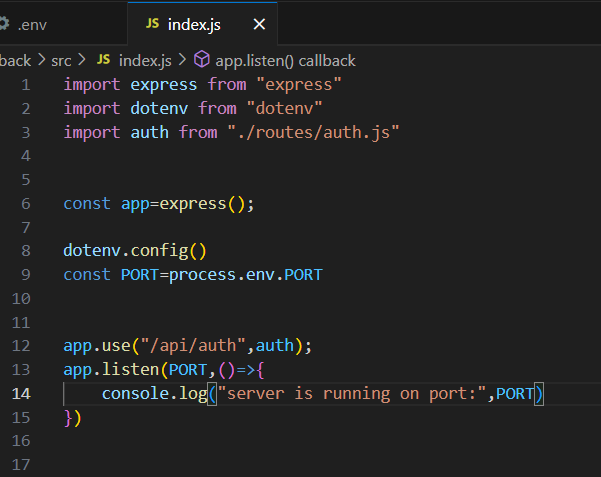
router.post("/signup",(req,res)=>{

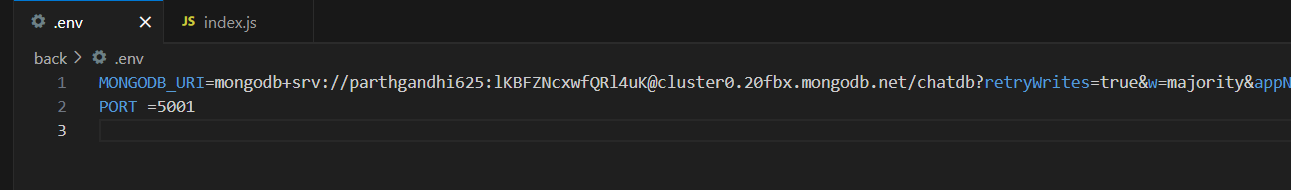
    res.send("signup")

})

If user visits /api/auth/signup he gets signup

Dotenv use:





Before ? add database name

App.use(express.json)=> data fetched is converted to json data that can be used in req.data

Else it is undefined

const salt=await bcrypt.genSalt(10);

const hashedpwd=await bcrypt.hash(pwd,salt);

decrypting the pwd using bcrypt

npm i react-router-dom react-hot-toast

wrap app with browser router import { BrowserRouter }from "react-router-dom"

import {Route,Routes} from "react-router-dom"

import axios from "axios";

export const axiosInstance=axios.create({

    baseURL:"http://localhost:5001/api",

    withCredentials:true,            //sending the cookie from the server

})

//Creating an axios instance (axios.create()) is useful when you need to:

//Set a Default baseURL (Avoid repeating the base API URL in every request)

//Enable withCredentials Globally (For authentication & cookies)

Zustand is a lightweight, fast, and scalable state management library for React. It helps you manage global and local state without the complexity of Redux.

Global state management refers to managing and sharing data (state) across multiple components in a React application **without passing props manually**. It helps maintain consistency and synchronization across the app.

