Started web application using react, node.js, mongodb, express, tailwindcss  
  
npm create vite@latest client

Framework: React

Variant: JS+SWC

Here we will have all dependencies: npm i

Search for tailwind vite and start following steps from documentation

DO GIT COMMIT

05/04/2025

Npm I react-router-dom

Check how many pages are need while creating website, here we need home,about,sign-in,sign-up, then create router with these pages

Make sure to check router is connecting with all pages or not

DO GIT COMMIT

Next is to create component folder as we checked in original website we have header for every pages, so we create header component

So we add them in APP.jsx after browserroute

When added tailwind, if there is not happening css in browser responsive, you can check in google developer tool (responsive browser)

DO GIT COMMIT

Create backend folder name as “api”

Install express, nodemon

Add type:”module” as it is ES6 module we need to add while using import express from “express”-----

  "name": "api",

  "version": "1.0.0",

  "description": "",

  "main": "index.js",

  "type":"module”

Delete test text in scripts in package.json and add dev, start   
then only we add npm run dev

DO GIT COMMIT

As we created git in client we need to move to api also

mv .git ../

created mongodb database -project and cluster, added in env (mongourl)

installed dotenv and mongoose

DO GIT COMMIT

We need to create test api route in backend by creating folders of router, controller and add json file in controller whether it is working or not. Check as we are using ES module, we need to import

DO GIT COMMIT

When create auth router check all slashes,

While req.body from controller we need to make sure to use

app.use(express.json()); add in server.js

we need to save data in db so we deconstruction the body (postman) and save in user schema which we have created then we need to use “user.save()”, as we know for every computer there will be change of time in delay to save in database, so we use await then async

Next we need to check for password as we save directly in database we need to encrypt it using bcryptsjs and add

const hashedPassword = bcrypt.hashSync(password, 8); after deconstruction

to get error visible in postman we use try and catch method

res.status(500).json(error.message)

but above error message is correct but the way of keeping is different so we will next commit

DO GIT COMMIT