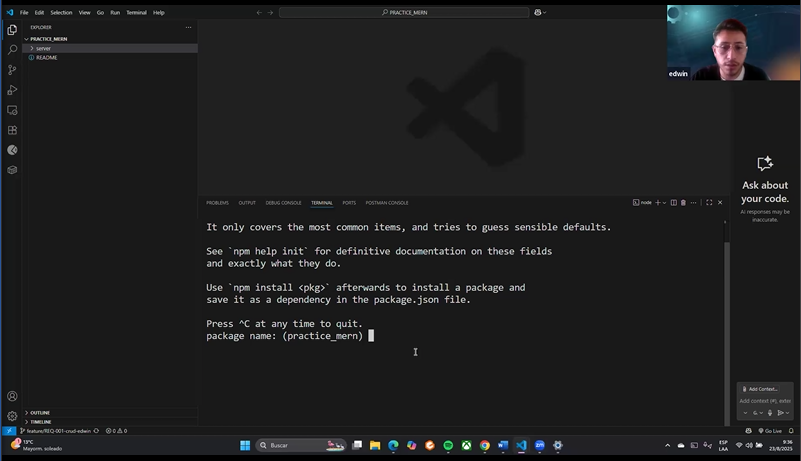
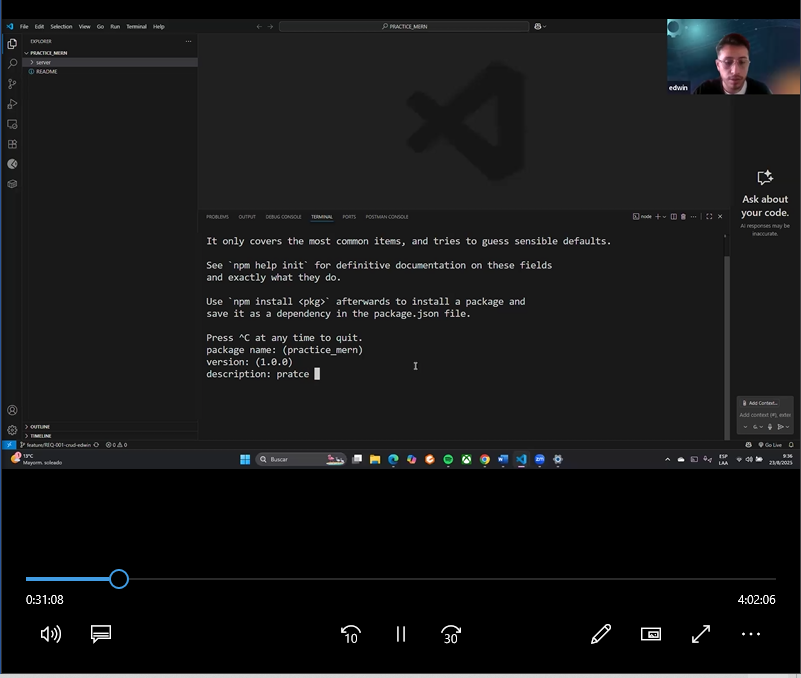
1. npm init

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm init

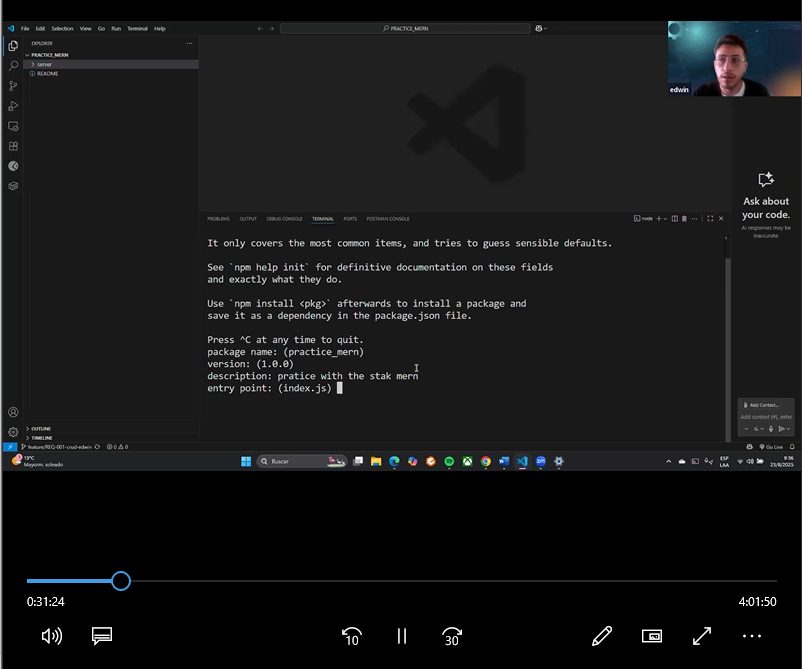
Nombre del paquete:



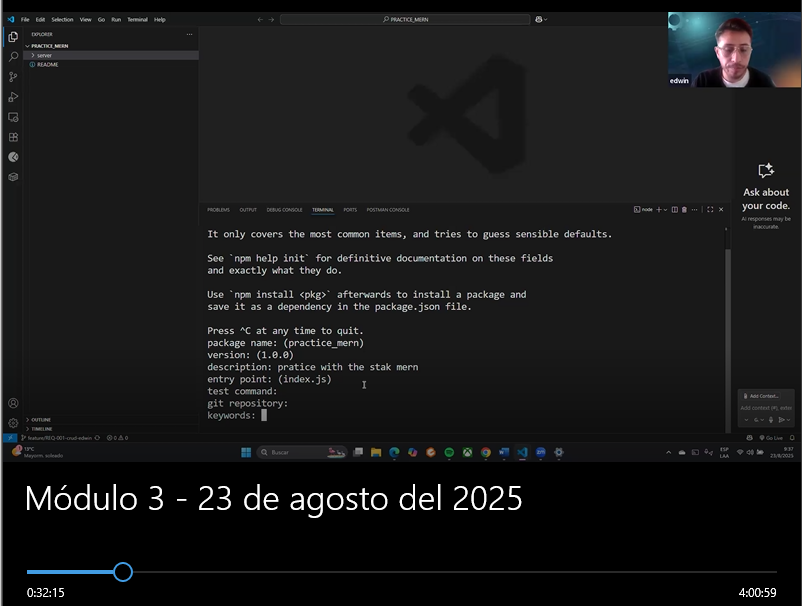
Versión a usar:



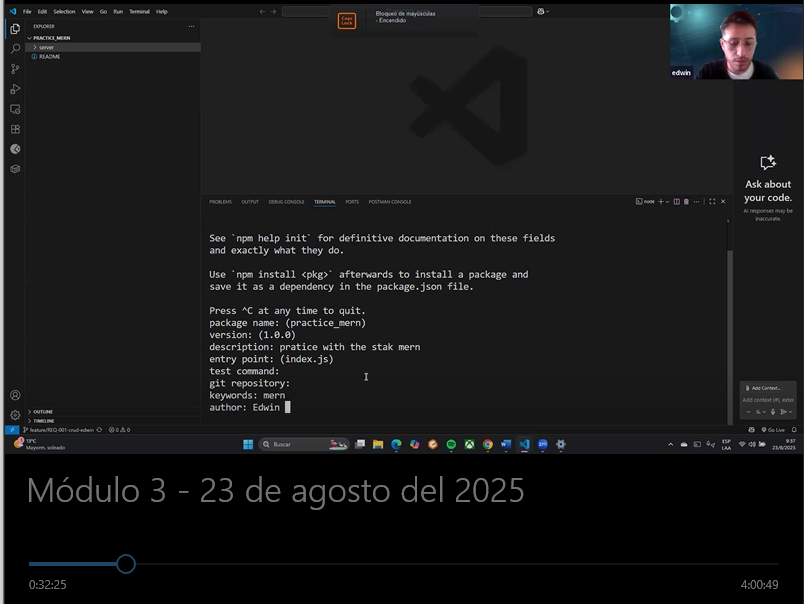
Descripción del proyecto y entry point:

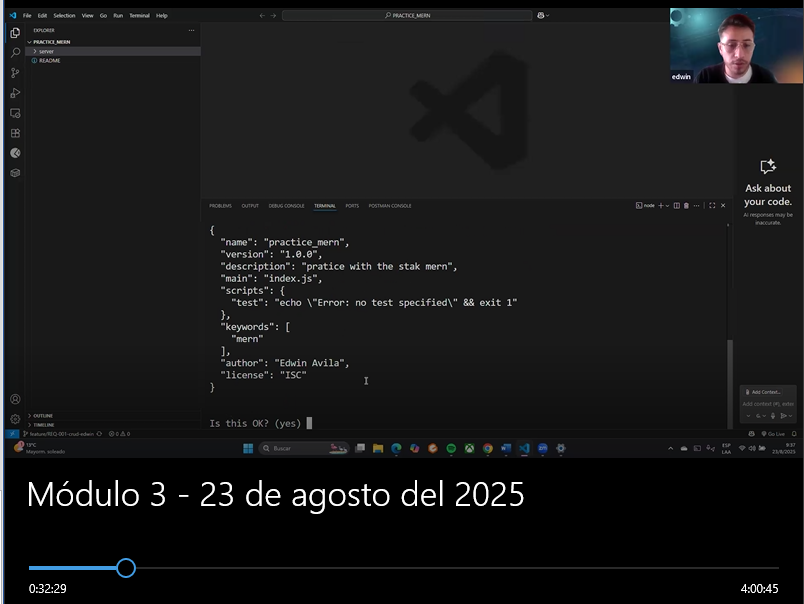


Test comand, git repository y keywords opcional



Autor y licencia





1. npm i dotenv

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i dotenv

1. npm i body-parser

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i body-parser

1. npm i express

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i express

1. npm i mongoose

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i mongoose

1. npm i save-dev nodemon

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i save-dev nodemon

1. npm install dotenv jsonwebtoken bcryptjs

D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server>npm install dotenv jsonwebtoken bcryptjs

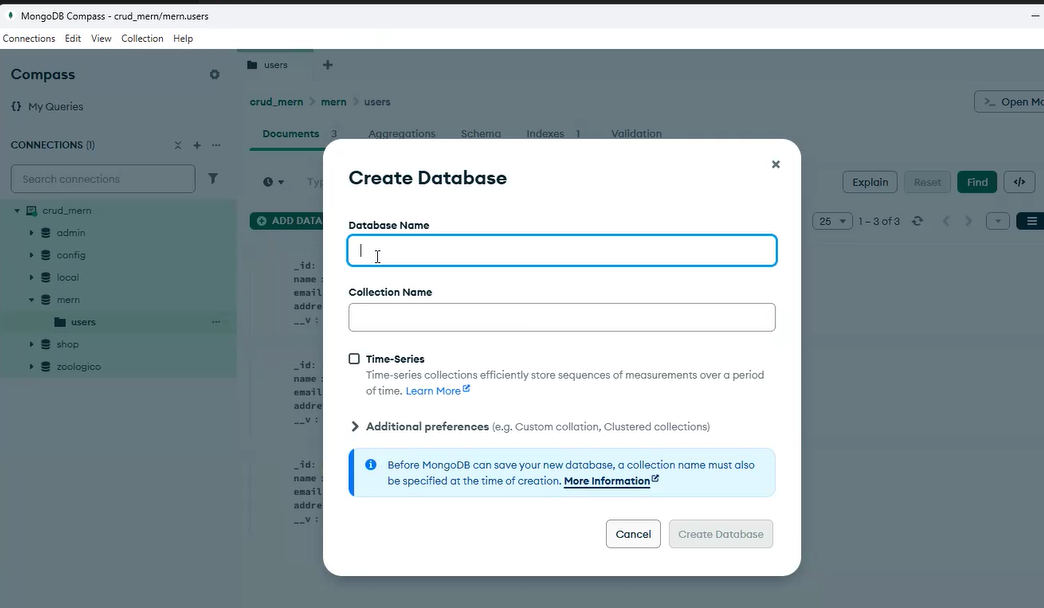
En mongo db compass hacer una nueva conexión:

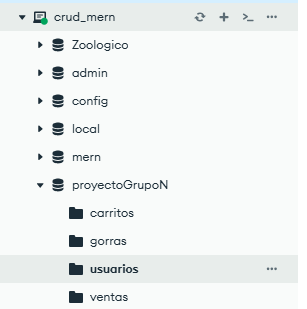
URL: mongodb://localhost:27017

Nombre de conexión: crud\_mern

Nombre de base de datos: proyectoGrupoN

Collection Name: usuarios, carritos, gorras, ventas

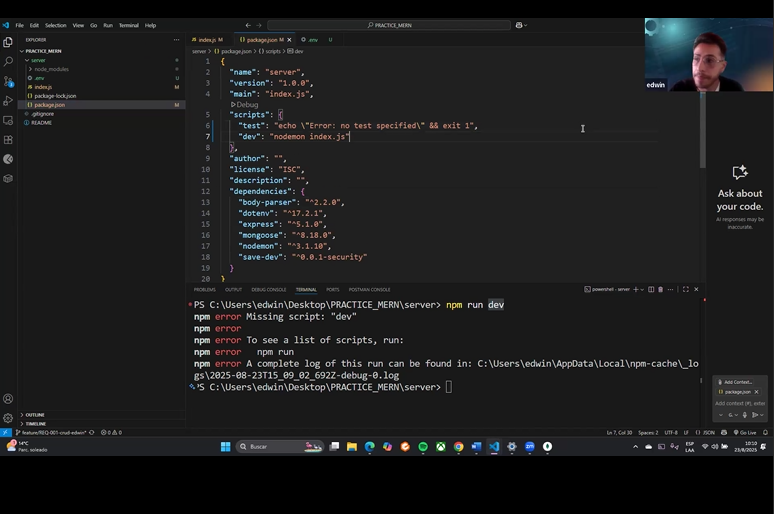




1. npm i cors

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm i cors

en el caso de:



1. npm run dev

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm run dev

1. npm install multer

D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server>npm install multer

1. npx create-react-app client

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npx create-react-app client

1. npm start

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm start

1. npm i bootstrap

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm i bootstrap

1. npm i react-router-dom

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm i react-router-dom

1. npm i react-hot-toast

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm i react-hot-toast

1. npm install react-bootstrap bootstrap

D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client>npm install react-bootstrap bootstrap

1. npm i font-awesome

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm i font-awesome

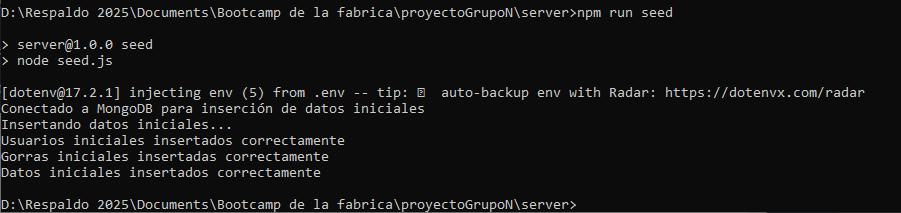
1. npm i axios

PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client> npm i axios

1. npm install chart.js react-chartjs-2

D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client>npm install chart.js react-chartjs-2

1. Cargar datos iniciales: D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server>npm run seed



1. Para DESARROLLO tener ambas terminales activas:

Para frontend: PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\client>npm start

Para backend: PS D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server> npm run dev

Resumen:

Instalación de paquetes:

1. npm init
2. npm i dotenv
3. npm i body-parser
4. npm i express
5. npm i mongoose
6. npm install --save-dev nodemon
7. npm i cors
8. npm run dev
9. npx create-react-app client
10. npm start
11. npm i bootstrap
12. npm i react-router-dom
13. npm i react-hot-toast
14. npm i font-awesome
15. npm i axios
16. para frontend: D:\Respaldo 2025\Documents\Bootcamp de la fabrica\modulo3\PRACTICE\_MERN\client>npm start
17. para backend: D:\Respaldo 2025\Documents\Bootcamp de la fabrica\modulo3\PRACTICE\_MERN\server> npm run dev

OPCIONAL:

Generación de JWT\_SECRET - Paso a paso

1. Crear el archivo generateSecret.js

Crea el archivo generateSecret.js en la **raíz de la carpeta**server (el mismo nivel que server.js):

**Ubicación:**

D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server\generateSecret.js

**Contenido del archivo:**

// generateSecret.js

const crypto = require('crypto');

const secret = crypto.randomBytes(64).toString('hex');

console.log('JWT\_SECRET=', secret);

2. Ejecutar el script

Abre tu terminal/consola y navega a la carpeta del servidor:

cd "D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server"

Luego ejecuta:

node generateSecret.js

3. Copiar el resultado

Verás una salida similar a esto en tu terminal:

JWT\_SECRET= 4f7d9a2c8b3e6f1a9c5d2e8b7a4f9c1d3e6f8a2b5c9d7e1f4a8b3c6d9e2f5a7b9c4d6e8f1a3b5c7d9e2f4a6b8c1d3e5f7a9b2c4d6e8f1a3b5

4. Actualizar el archivo .env

Abre tu archivo .env y reemplaza la línea del JWT\_SECRET con la clave generada:

**Archivo:** D:\Respaldo 2025\Documents\Bootcamp de la fabrica\proyectoGrupoN\server\.env

PORT=5001

MONGO\_URI=mongodb://localhost:27017/proyectoGrupoN

JWT\_SECRET=4f7d9a2c8b3e6f1a9c5d2e8b7a4f9c1d3e6f8a2b5c9d7e1f4a8b3c6d9e2f5a7b9c4d6e8f1a3b5c7d9e2f4a6b8c1d3e5f7a9b2c4d6e8f1a3b5

JWT\_EXPIRE=30d

NODE\_ENV=development

5. (Opcional) Eliminar el archivo generateSecret.js

Una vez que hayas copiado el JWT\_SECRET a tu archivo .env, puedes eliminar el archivo generateSecret.js por seguridad:

# Desde la carpeta server/

del generateSecret.js