



Desarrollo de FullStack

AI07064821

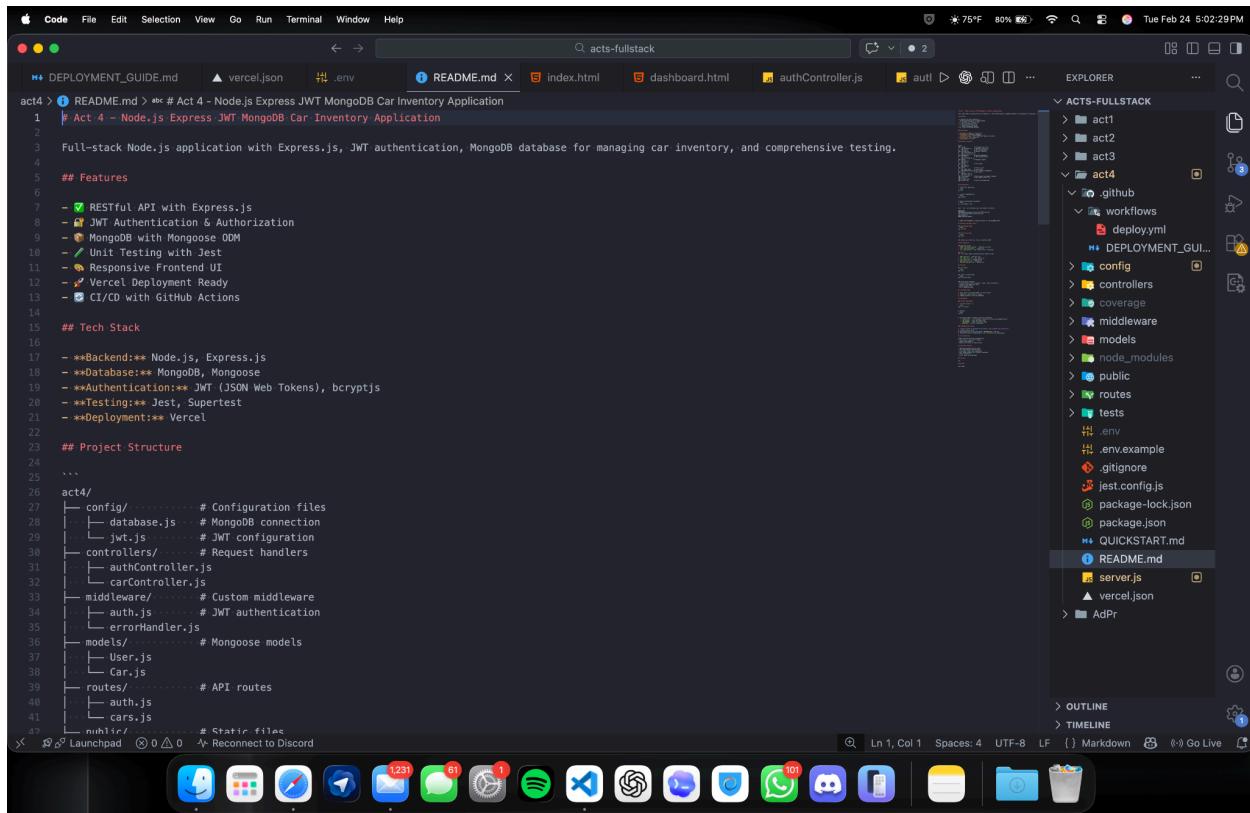
Diego Villarreal Martinez

Profesor: Francisco Gomez Rubio

Link del repo de
github:

[https://github.com/We
bbiestStew/acts-fullst
ack.git](https://github.com/WebbiestStew/acts-fullstack.git)

Primero se creo el [Readme.md](#) explicando las funciones de el codigo



```
# Act 4 - Node.js Express JWT MongoDB Car Inventory Application
# Act 4 - Node.js Express JWT MongoDB Car Inventory Application

Full-stack Node.js application with Express.js, JWT authentication, MongoDB database for managing car inventory, and comprehensive testing.

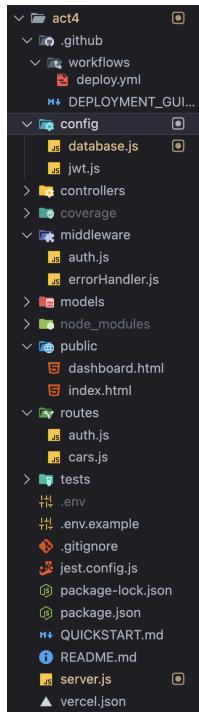
## Features
- ✓ RESTful API with Express.js
- ✓ JWT Authentication & Authorization
- ✓ MongoDB with Mongoose ODM
- ✓ Unit Testing with Jest
- ✓ Responsive Frontend UI
- ✓ Vercel Deployment Ready
- ✓ CI/CD with GitHub Actions

## Tech Stack
**Backend:** Node.js, Express.js
**Database:** MongoDB, Mongoose
**Authentication:** JWT (JSON Web Tokens), bcryptjs
**Testing:** Jest, Supertest
**Deployment:** Vercel

## Project Structure
```
act4/
 config/ # Configuration files
 database.js # MongoDB connection
 jwt.js # JWT configuration
 controllers/ # Request handlers
 authController.js
 carController.js
 middleware/ # Custom middleware
 auth.js # JWT authentication
 errorHandler.js
 models/ # Mongoose models
 User.js
 Car.js
 routes/ # API routes
 auth.js
 cars.js
 static/ # Static files
```

RECONNECT TO DISCORD
```

Despues se empezo el codigo usando [Node.js](#) como el principal, despues el framework de [Express.js](#) para el servidor y se usaron los middleware necesarios para la auth con JWT y se uso MongoDB para la base de datos



System Status: All Good
©2026 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Despues se creo el HTML con el login (se adjunta fotos del codigo y de su ejecucion)

Code File Edit Selection View Go Run Terminal Window Help

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no">
<title>Login - Car Inventory Management</title>
<style>
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

body {
    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, sans-serif;
    background: linear-gradient(135deg, #667eea 0%, #7e57c2 100%);
    min-height: 100vh;
    display: flex;
    justify-content: center;
    align-items: center;
    position: relative;
    overflow: hidden;
}

body::before {
    content: '';
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background:
        radial-gradient(circle at 20% 50%, rgba(120, 119, 117, 0.95)),
        radial-gradient(circle at 80% 80%, rgba(138, 43, 122, 0.95));
    pointer-events: none;
}

.container {
    background: #fff;
    backdrop-filter: blur(10px);
    padding: 3rem;
}
```

Login - Car Inventory Management

Login Register

Email

Password

LOGIN

ACTS-FULLSTACK

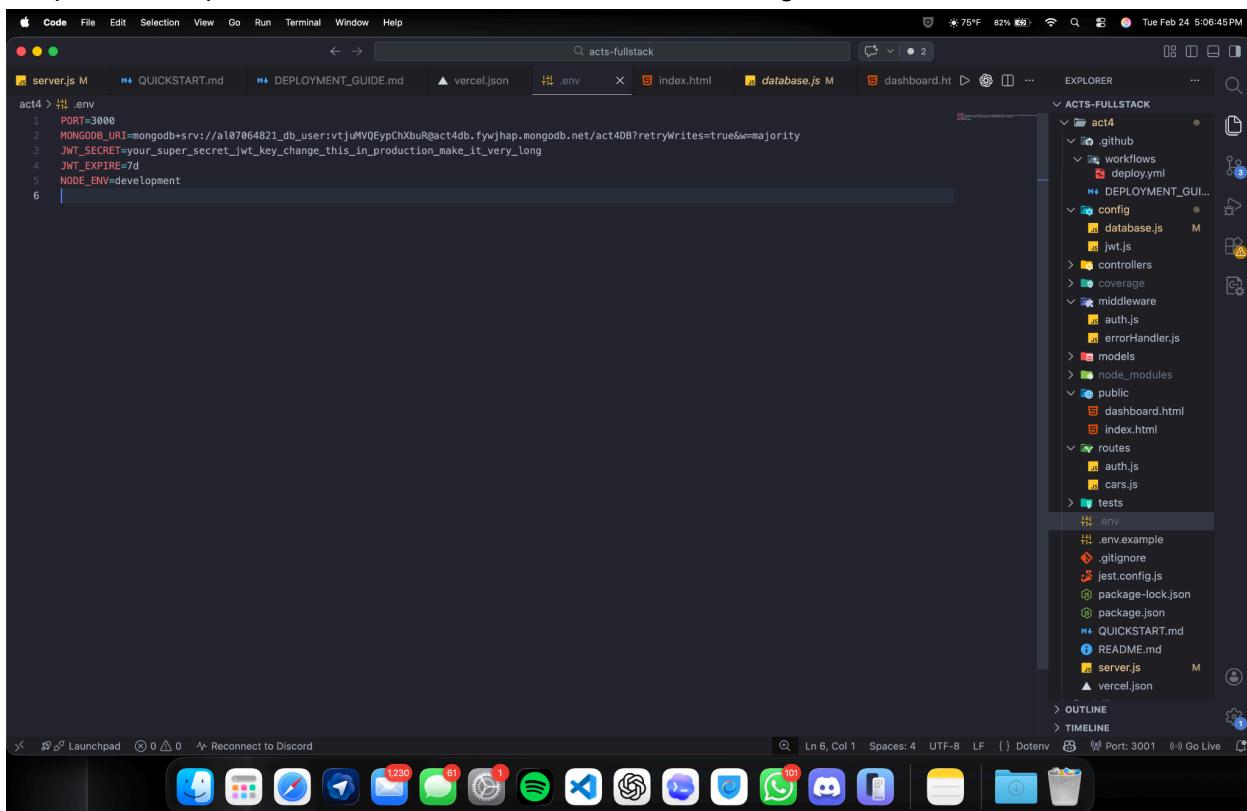
- act4
 - .github
 - workflows
 - deploy.yml
- DEPLOYMENT_GUIDE.md
- config
 - database.js
 - jwt.js
- controllers
- coverage
- middleware
 - auth.js
 - errorHandler.js
- models
- node_modules
- public
 - dashboard.html
 - index.html
- routes
 - auth.js
 - cars.js
- tests
 - .env.example
 - .gitignore
 - jest.config.js
- .env
- package-lock.json
- package.json
- QUICKSTART.md
- README.md
- server.js
- vercel.json

OUTLINE

TIMELINE

Port: 3001

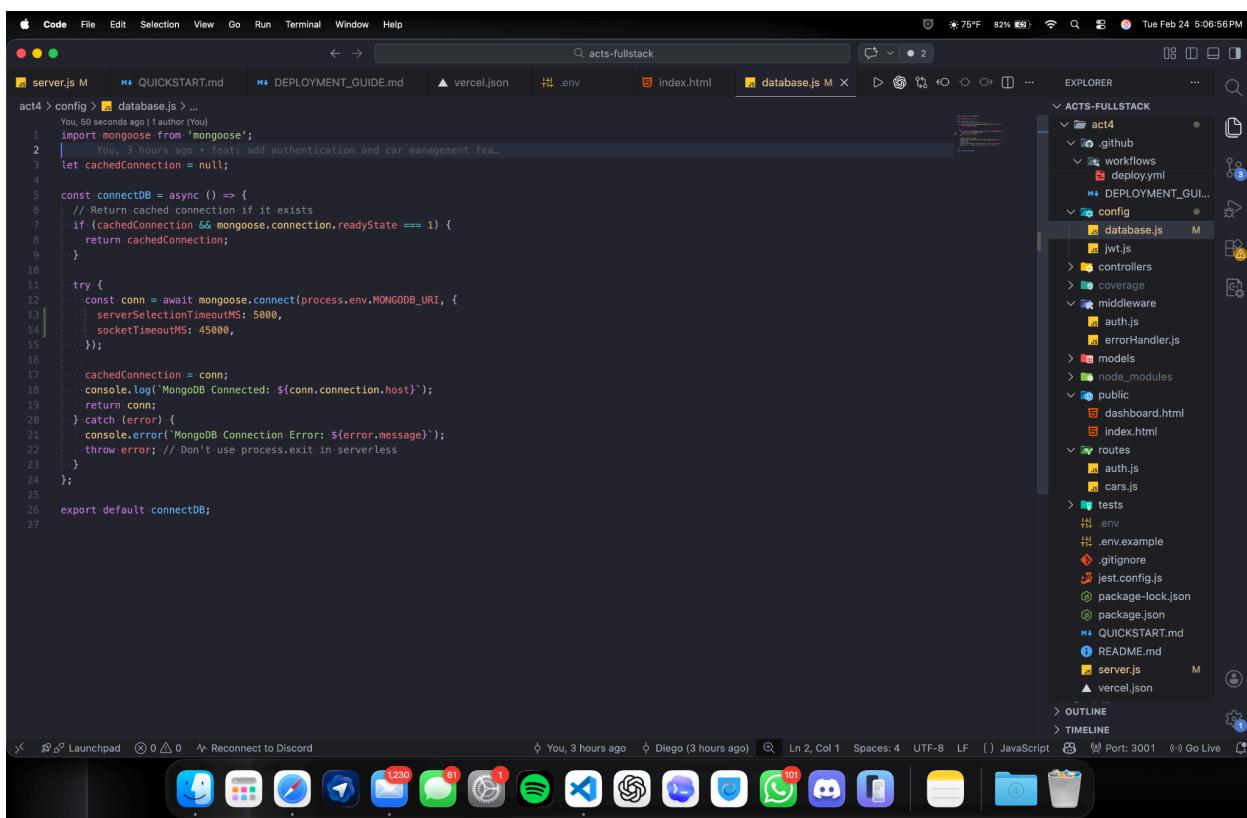
Despues se empezo la conexion a la base de datos de MongoDB



A screenshot of the VS Code interface. The left sidebar shows the project structure for 'ACTS-FULLSTACK' with files like 'act4', '.github', 'workflows', 'config', 'database.js', 'controllers', 'coverage', 'middleware', 'models', 'public', 'routes', 'tests', and various configuration files. The main editor tab is 'server.js' with the following code:

```
act4 > .env
1 PORT=3000
2 MONGODB_URI=mongodb+srv://al07064821_db_user:vtjuMVQEyPChxbuR@act4db.fywjh.mongodb.net/act4DB?retryWrites=true&w=majority
3 JWT_SECRET=your_super_secret_jwt_key_change_this_in_production_make_it_very_long
4 JWT_EXPIRE=7d
5 NODE_ENV=development
6
```

The status bar at the bottom shows 'Ln 6, Col 1' and 'Port: 3001'.



A screenshot of the VS Code interface. The left sidebar shows the project structure for 'ACTS-FULLSTACK' with files like 'act4', '.github', 'workflows', 'config', 'database.js', 'controllers', 'coverage', 'middleware', 'models', 'public', 'routes', 'tests', and various configuration files. The main editor tab is 'database.js' with the following code:

```
act4 > config > database.js > ...
1 You, 50 seconds ago | Author (You)
2 import mongoose from 'mongoose';
3 let cachedConnection = null;
4
5 const connectDB = async () => {
6   // Return cached connection if it exists
7   if (cachedConnection && mongoose.connection.readyState === 1) {
8     return cachedConnection;
9   }
10
11   try {
12     const conn = await mongoose.connect(process.env.MONGODB_URI, {
13       serverSelectionTimeoutMS: 5000,
14       socketTimeoutMS: 45000,
15     });
16
17     cachedConnection = conn;
18     console.log(`MongoDB Connected: ${conn.connection.host}`);
19     return conn;
20   } catch (error) {
21     console.error(`MongoDB Connection Error: ${error.message}`);
22     throw error; // Don't use process.exit in serverless
23   }
24 };
25
26 export default connectDB;
27
```

The status bar at the bottom shows 'Ln 2, Col 1' and 'Port: 3001'.

Y ya por ultimo se hicieron las pruebas Jest y sus respectivas ejecuciones para cer el correcto funcionamiento de todo el codigo:

The screenshot shows the VS Code interface with the following details:

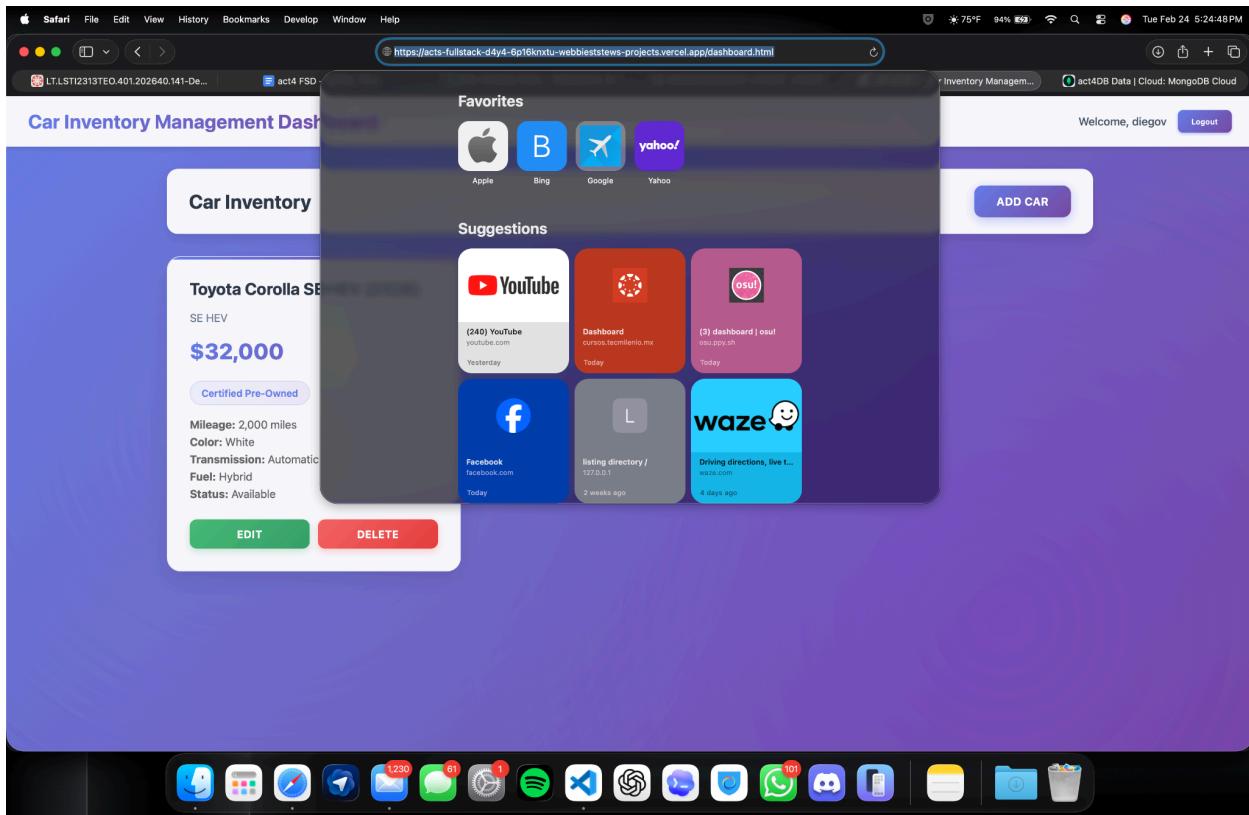
- Code Editor:** The main area displays `server.js` and a terminal window showing Jest test results.
- Terminal:**

```
diegovillarreal@Diegos-MacBook-Pro-3 act4 % npm test
  POST /api/auth/login
    ✓ should login user with correct credentials (114 ms)
    ✓ should login user with incorrect password (115 ms)
    ✓ should not login with non-existent email (60 ms)
    ✓ should not login without email or password (59 ms)
  GET /api/auth/me
    ✓ should get current user with valid token (60 ms)
    ✓ should not get user without token (59 ms)
    ✓ should not get user with invalid token (59 ms)
```
- File Explorer:** Shows the project structure for `ACTS-FULLSTACK`, including files like `act4`, `.github`, `config`, `database.js`, `jwt.js`, `controllers`, `coverage`, `middleware`, `auth.js`, `errorHandler.js`, `models`, `public`, `routes`, `index.html`, and `tests`.
- Output:** Shows coverage analysis for various files, such as controllers, authController.js, carController.js, middleware, auth.js, routes, authHandler.js, and cars.js.
- Terminal:** Shows the command `npm test` and its execution results.

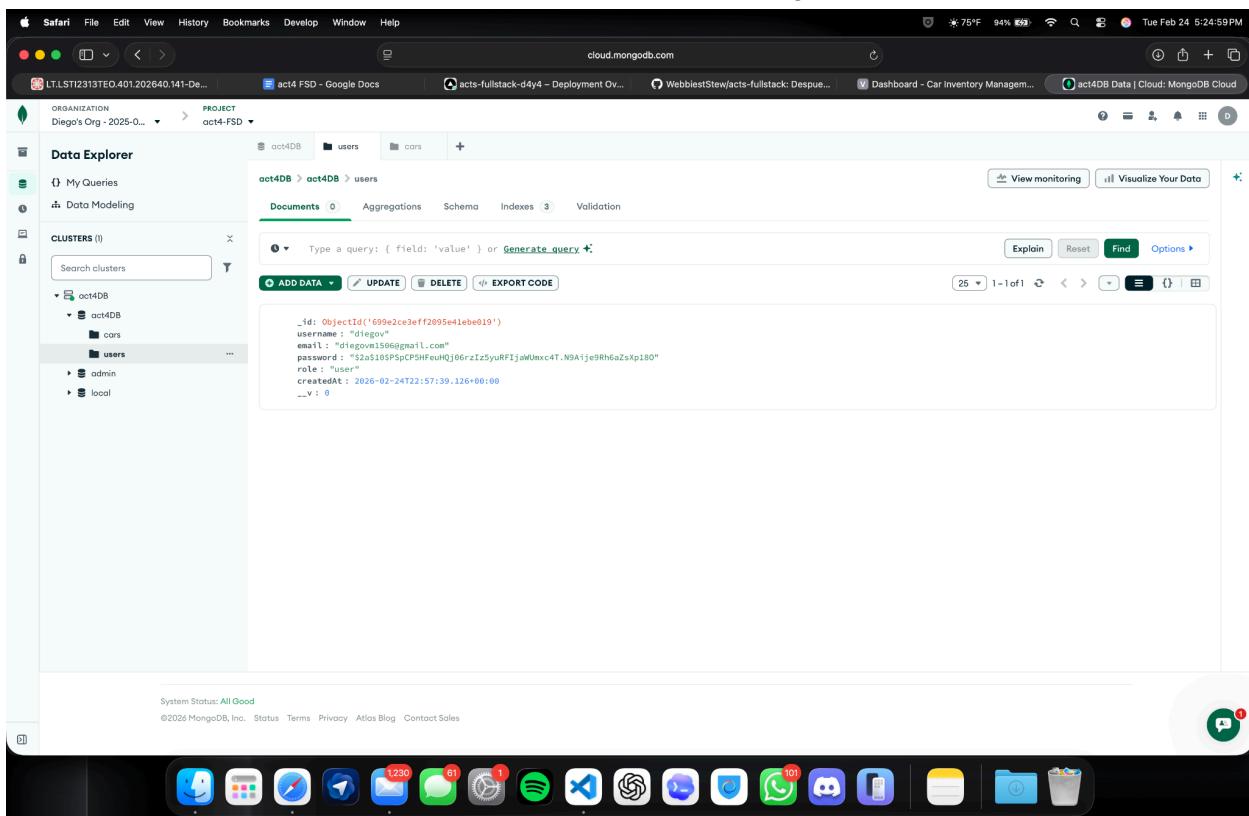
Y se despliega el codigo en Vercel:

The screenshot shows the Vercel deployment details page for a project named `act4`:

- Deployment Details:**
 - Created:** 3h ago
 - Status:** Ready Latest
 - Duration:** 18s
 - Environment:** Production
 - Domains:** `act4-fulstack-d4y4.VERCELAPP`
 - Source:** `main`
 - Build Logs:** 18s (2 errors)
 - Deployment Summary:** Resources
 - Deployment Checks:**
 - Assigning Custom Domains:**
- Runtime Logs:** View and debug runtime logs & errors
- Observability:** Monitor app health & performance
- Speed Insights:** Performance metrics from real users
- Web Analytics:** Analyze visitors & traffic in real-time



Y los usuarios son creados exitosamente con medidas de seguridad de encriptación



Link de la pagina de la act4:

<https://acts-fullstack-d4y4-6p16knxtu-webbieststews-projects.vercel.app/dashboard.html>