

# E-Commerce Web App — AI-Assisted Development

## Tools Used

- ChatGPT
- GitHub Copilot

## Project Overview

This project focuses on developing the **login and signup feature** for an E-Commerce web application using **React**, **Bootstrap**, **Express**, and **MongoDB**. AI tools provided backend logic, CSS styling improvements, and Git-related guidance, while the UI layout was manually built using Bootstrap for greater design control.

## AI Interaction & Prompts

1. **Initial Request:** Asked ChatGPT to create a login/signup page. The output placed both forms on one page without routing.
2. **Layout Adjustment:** Requested layout changes from AI, but results did not match the intended design, so a manual Bootstrap layout was created.
3. **CSS Enhancement:** Asked AI for styling improvements, receiving suggestions for colors, shadow effects, and alignment.
4. **Backend Setup:** Requested authentication logic; AI provided Express routes with JWT authentication and bcrypt password hashing.
5. **Git Troubleshooting:** Sought guidance on cleaning large files from the repository and updating `.gitignore`.

## AI's Role

- **Frontend:** Provided example components and CSS enhancements.
- **Backend:** Suggested Express-based authentication with JWT and bcrypt.

- **Git:** Explained feature branch workflow, `.gitignore` configuration, and repository cleanup.

## My Implementation

- Built **SignupForm.jsx** with a custom Bootstrap layout.
- Applied AI-suggested CSS improvements for a modern and responsive look.
- Implemented **signup/login API endpoints** connected to MongoDB.
- Integrated Axios for form submissions.
- Fixed Git push issues by cleaning the repository and refining `.gitignore`.

## Code Evolution

### AI's Initial Version:

- Both forms on the same page.
- Minimal styling.
- No routing separation.

### Final Version:

- Separate components for **signup** and **login**.
- Responsive, Bootstrap-based design.
- Axios integration for API calls.
- Navigation handled via `react-router-dom`.

## Component Working

- Uses **React useState** to manage form input fields.
- On submission, sends a **POST** request to the backend using Axios.

- Bootstrap ensures proper alignment, spacing, and responsiveness.
- Link from `react-router-dom` enables smooth navigation.

## Advantages

- Clean and responsive UI.
- Easy to maintain and reuse.
- Backend-ready with secure API integration.
- Quick CSS enhancements from AI.

## Limitations

- No client-side form validation.
- No error or loading state handling.
- API endpoint currently hardcoded.
- Accessibility optimizations not implemented.

## Original AI Project Plan

- **Week 1:** Authentication setup (login/signup UI and backend).
- **Week 2:** Product listing & cart.
- **Week 3:** Checkout & payment integration.
- **Week 4:** Azure deployment & testing.

## Reflection

AI was most effective for **backend code samples**, **CSS styling suggestions**, and **Git troubleshooting**. Manual work proved better for **UI layout, structure**, and **routing setup**. The hybrid approach allowed for rapid development while maintaining creative control over the user interface.

