

[Assignment # 2]

[Visual studio tool]



October 22, 2025

[Areeba usman]

[49039]

# In-Class Activity Report

Course: Generative AI in Software Development

Week:8 — AI in Front-End and Back-End Development

Activity Title: Using LLMs to Build and Test Full-Stack Components

Student Name: Areeba Usman

Date:10/22/2025

Total Marks: 10

## 1. Objective

The purpose of this activity was to practically explore how Large Language Models (LLMs) such as ChatGPT or Copilot can assist developers in generating, validating, and testing full-stack code. The activity focused on creating a login interface (front-end) and implementing corresponding backend logic.

## 2. Tools & Technologies Used

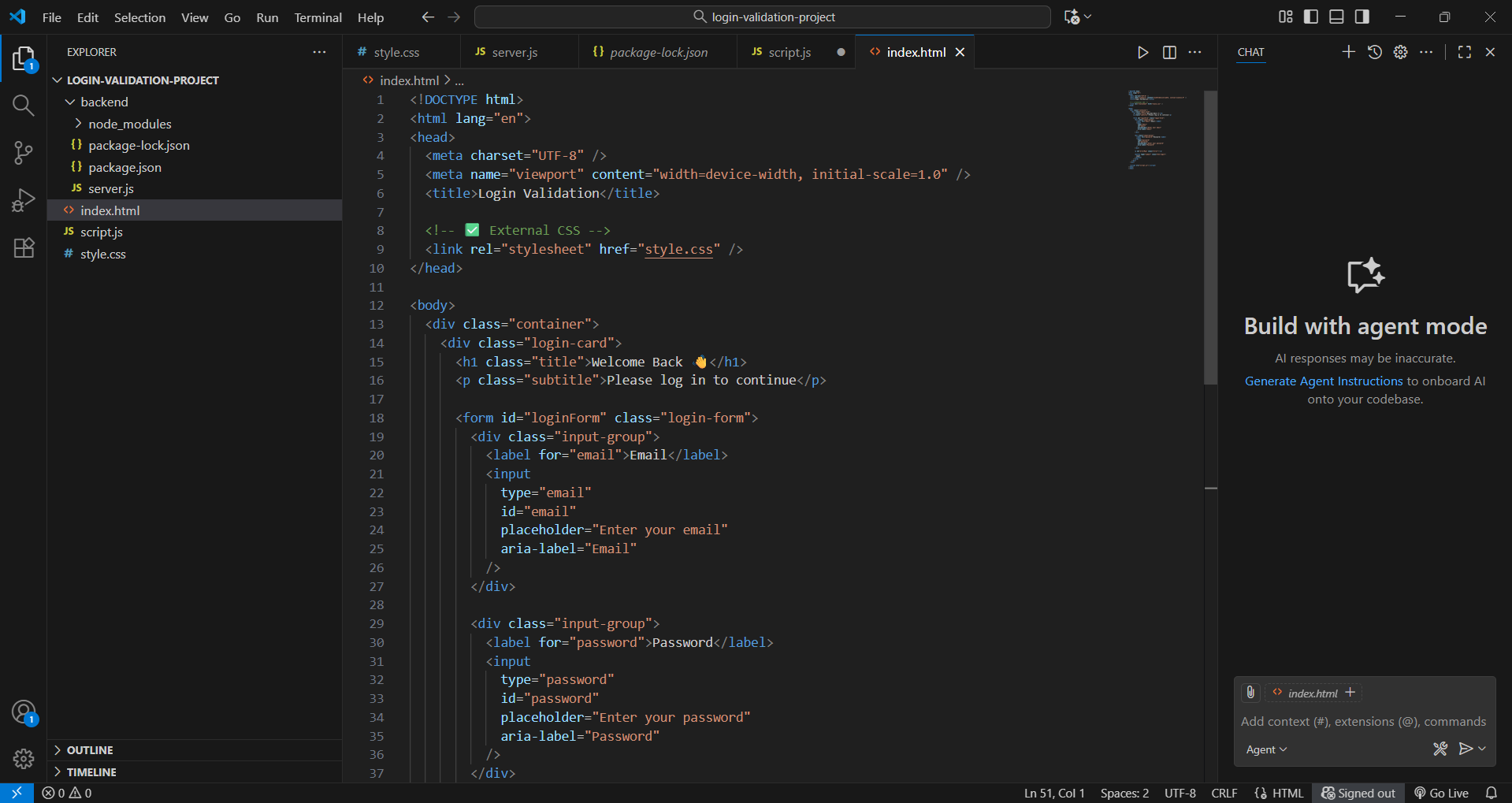
|  |  |
| --- | --- |
| **Category** | **Tools / Technologies Used** |
| Front-End | HTML, CSS, JavaScript |
| Back-End | Node.js, Express |
| AI Tool | ChatGPT (GPT-5) |

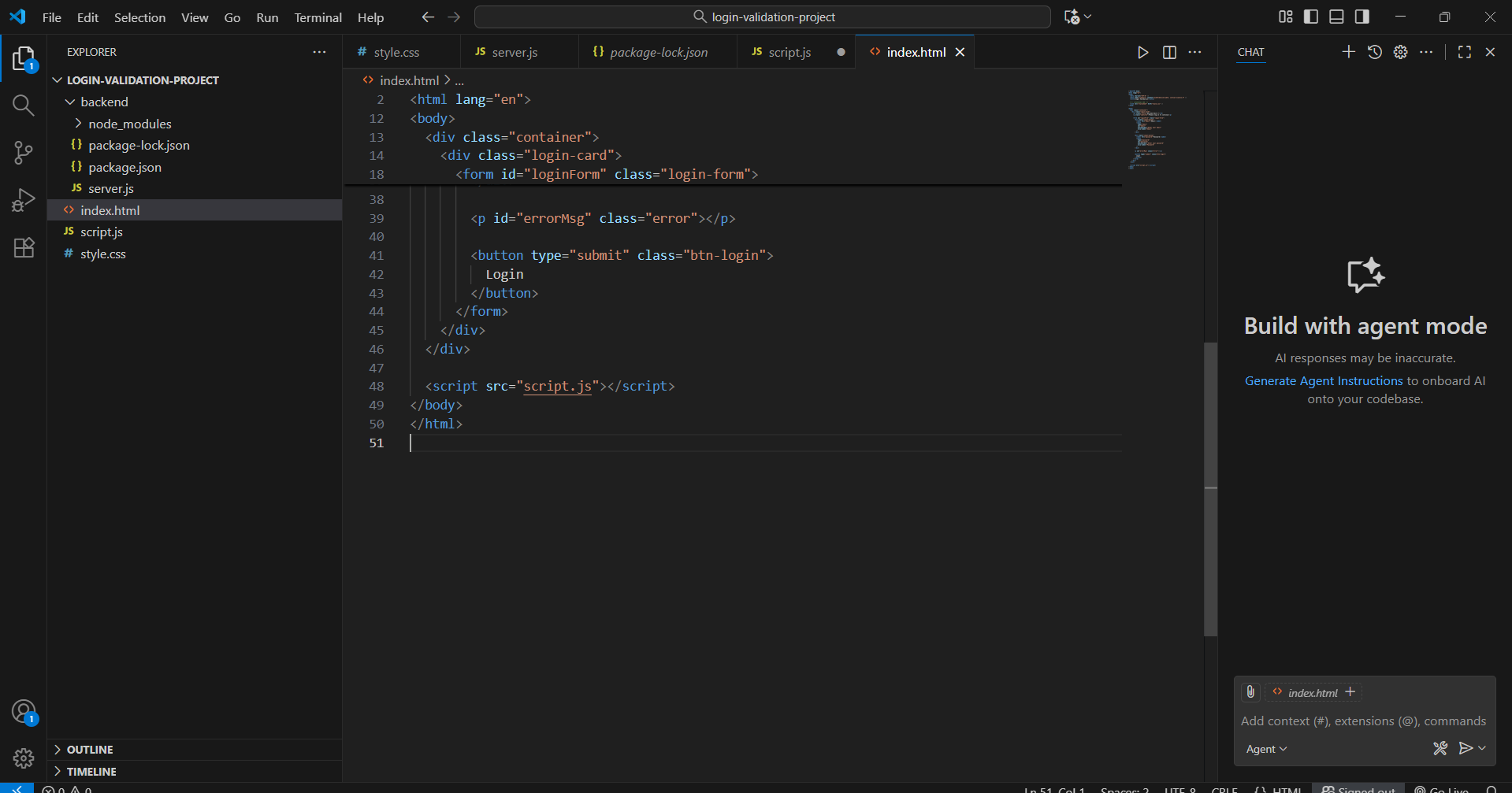
## 3. Part A – Front-End (5 Marks)

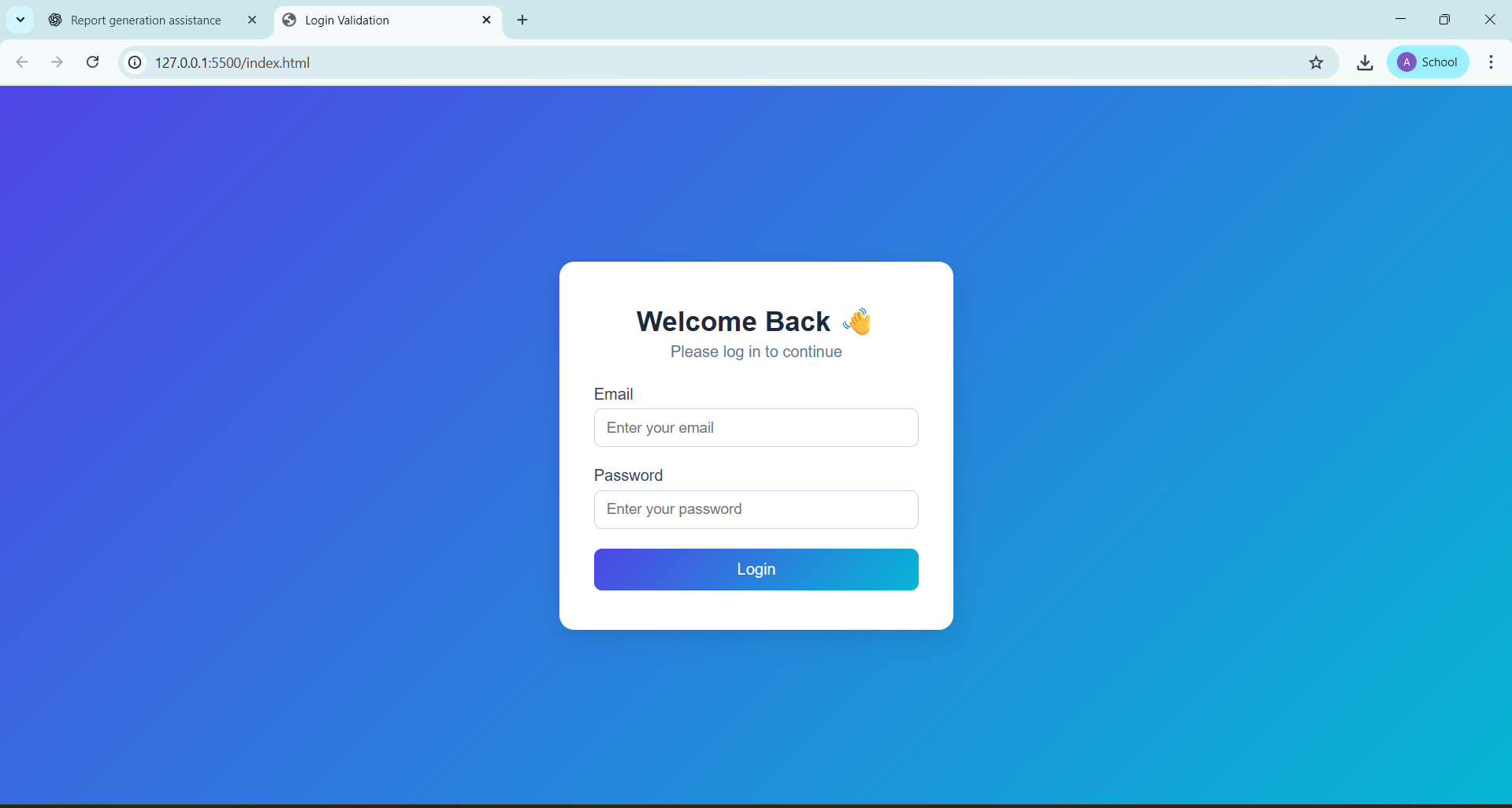
Using an LLM to generate a responsive login form with:  
- Email and password fields  
- Submit button  
- Client-side validation  
- Accessibility attributes

### Generated Code (HTML, CSS, JS)

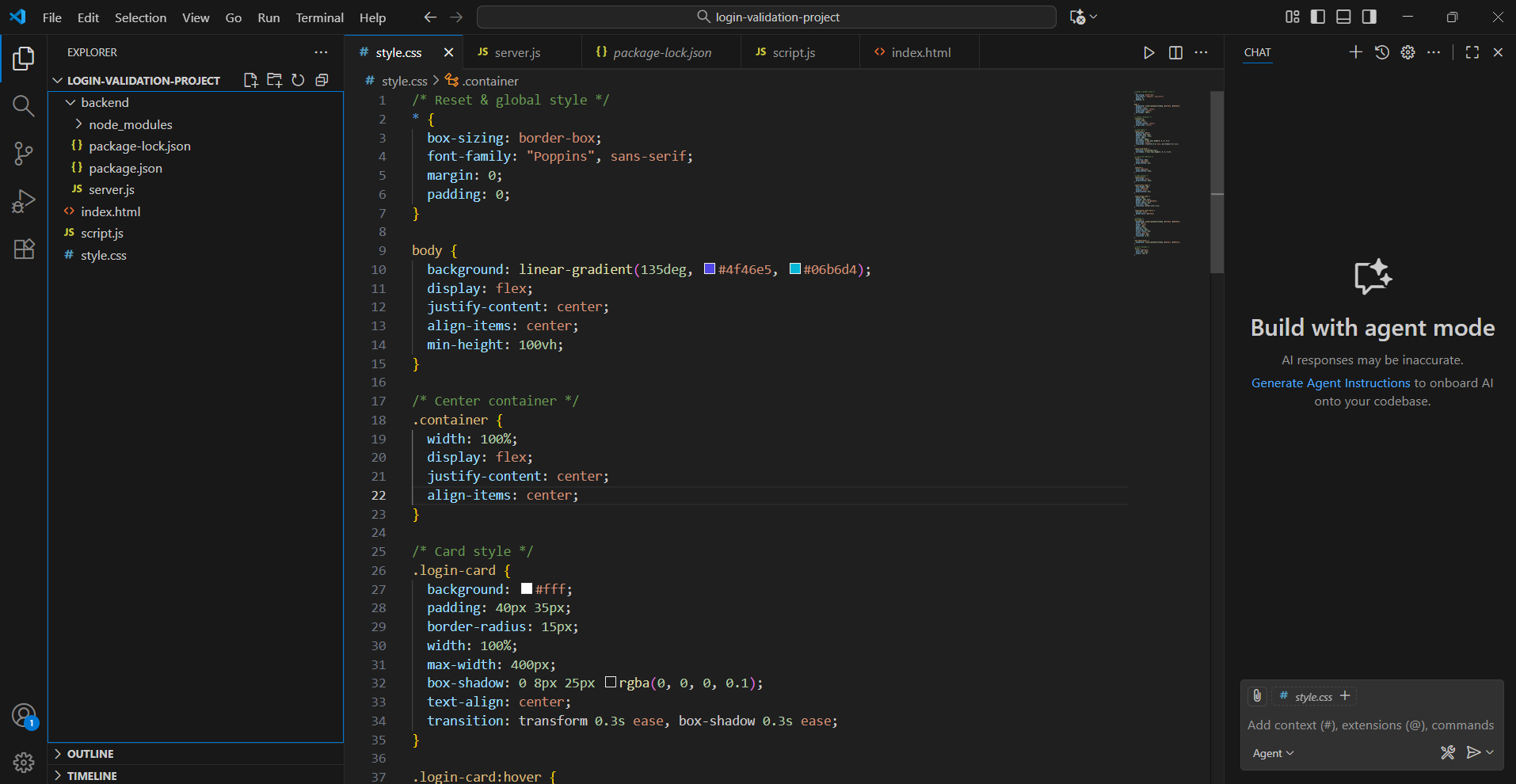
Index.html file:

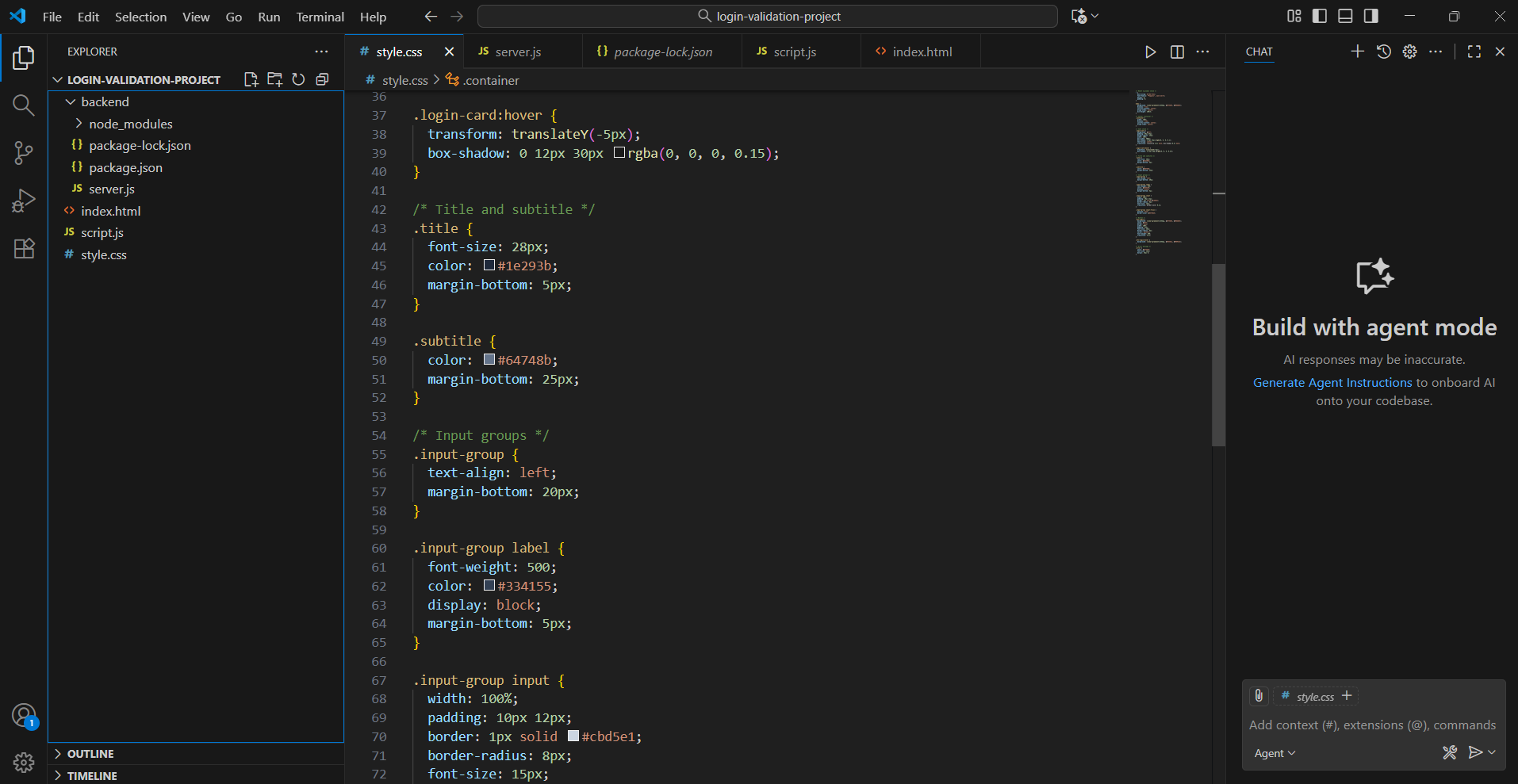


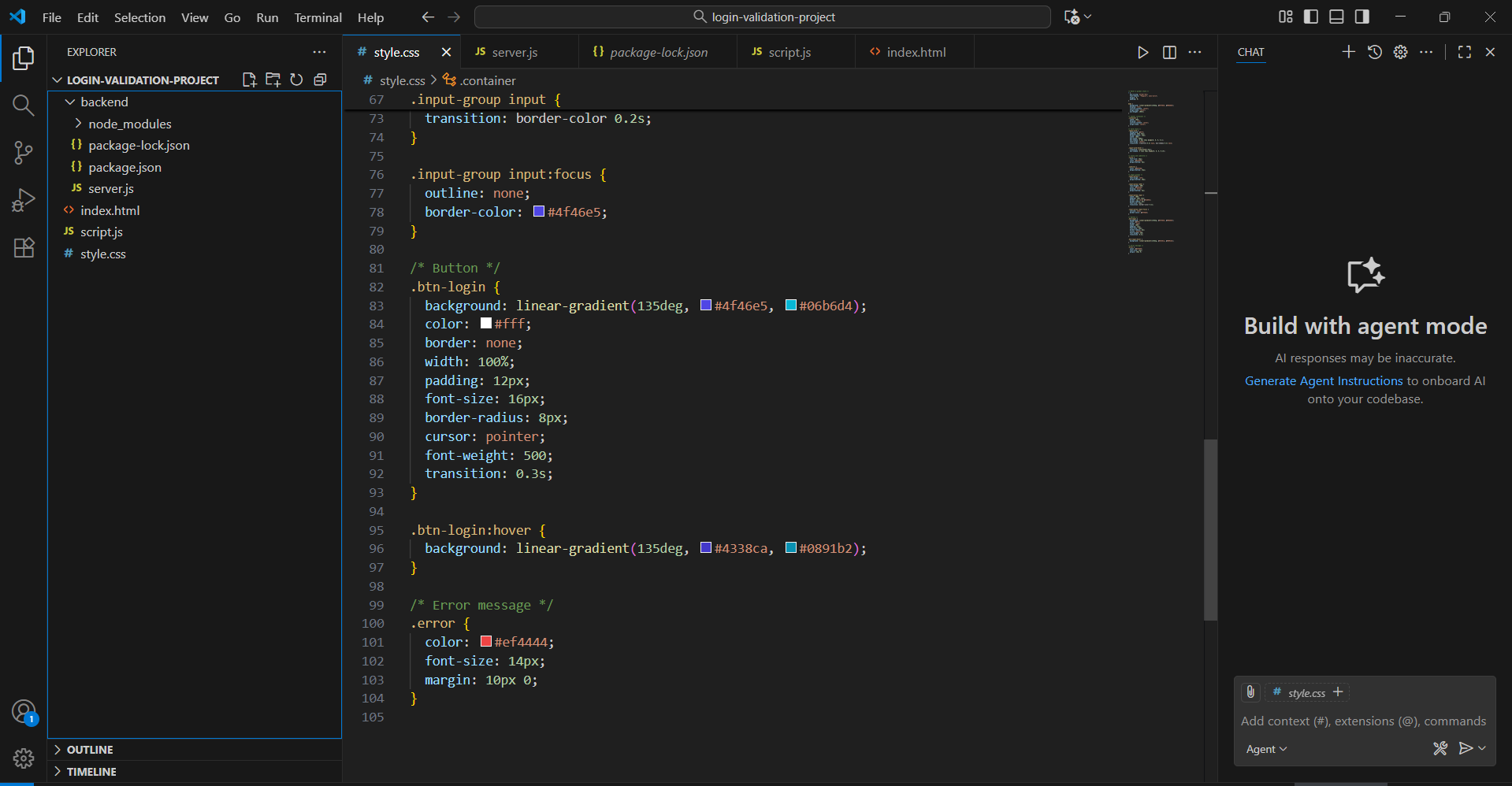




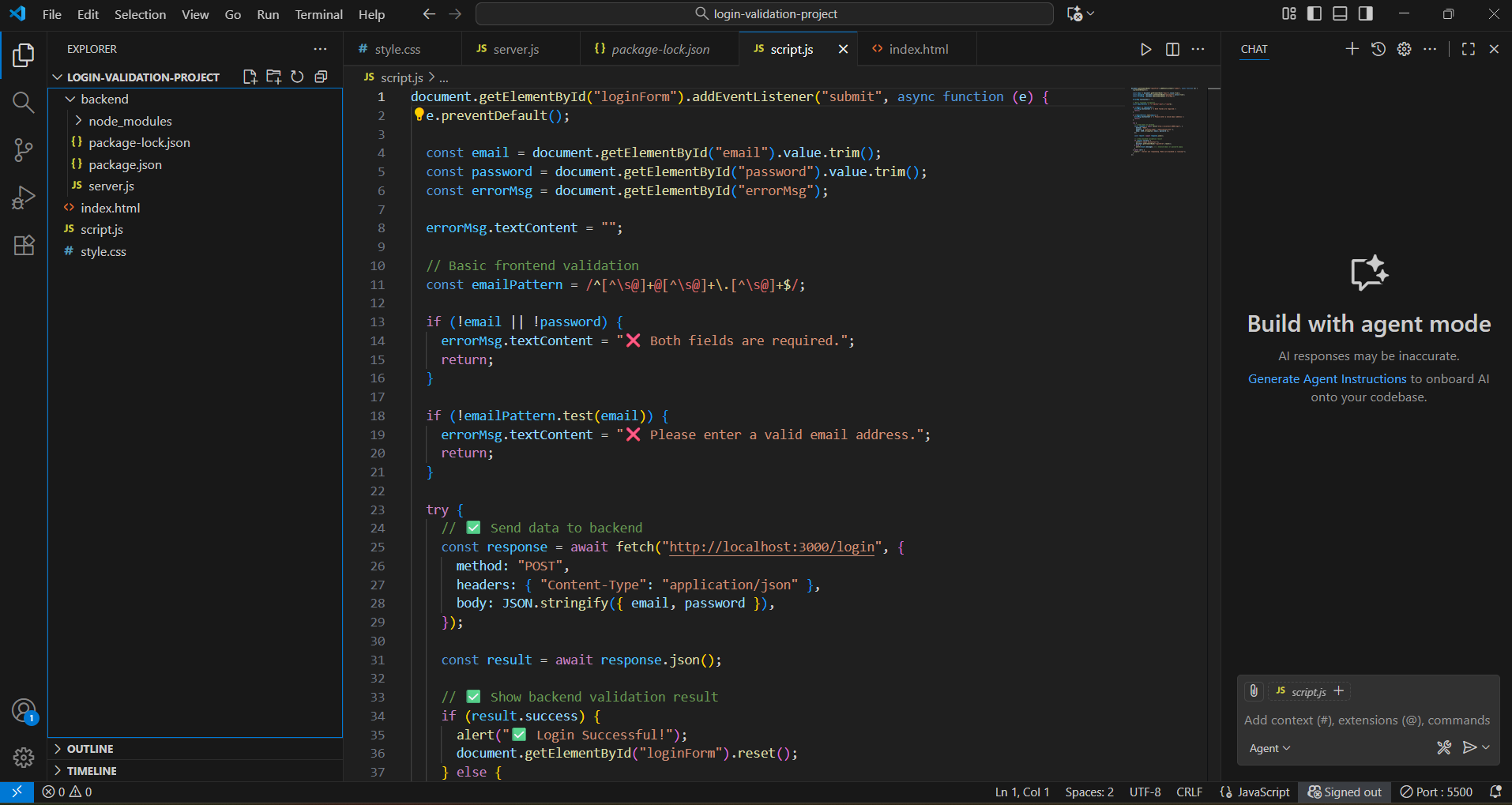
Style.css file:

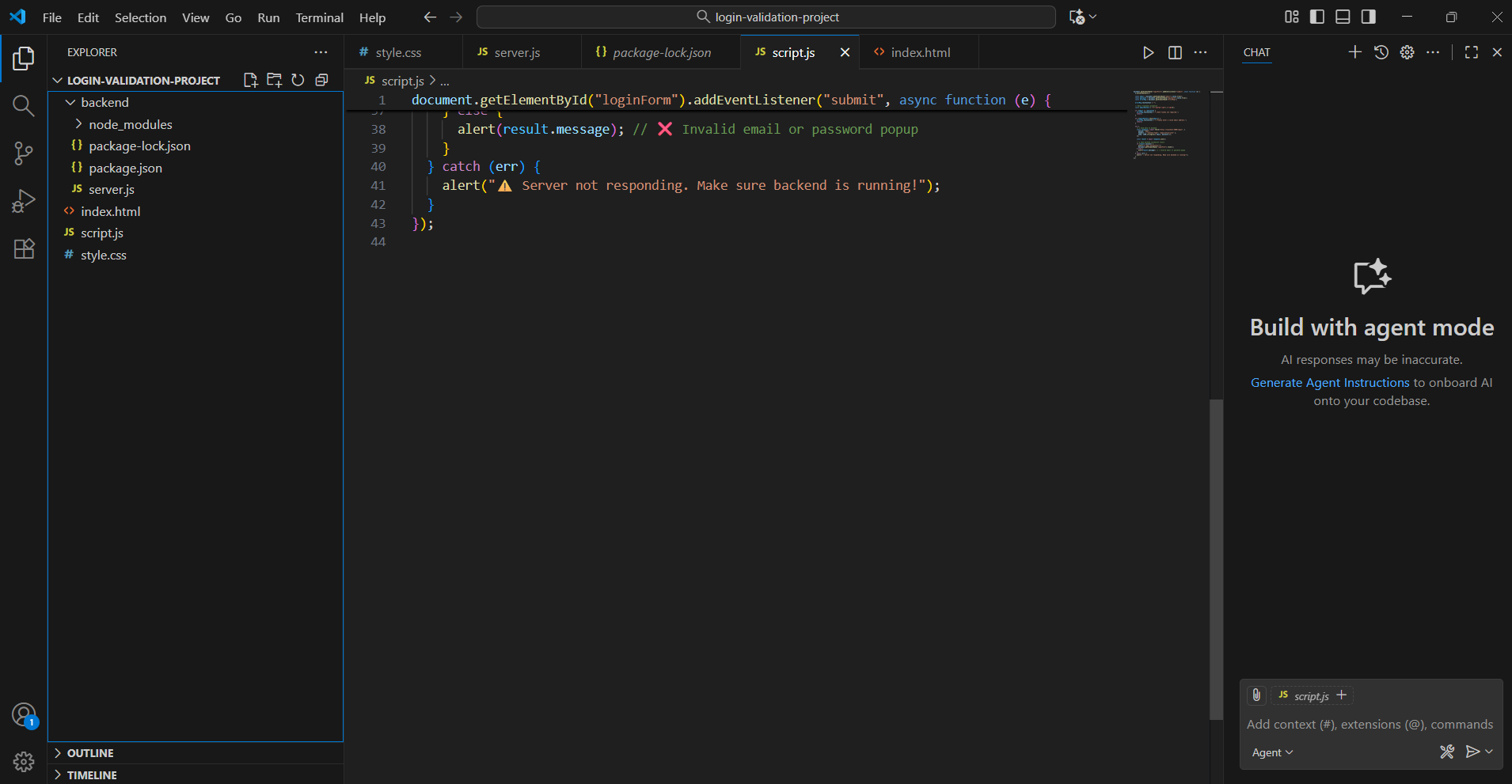






Script.js





Server.js file:

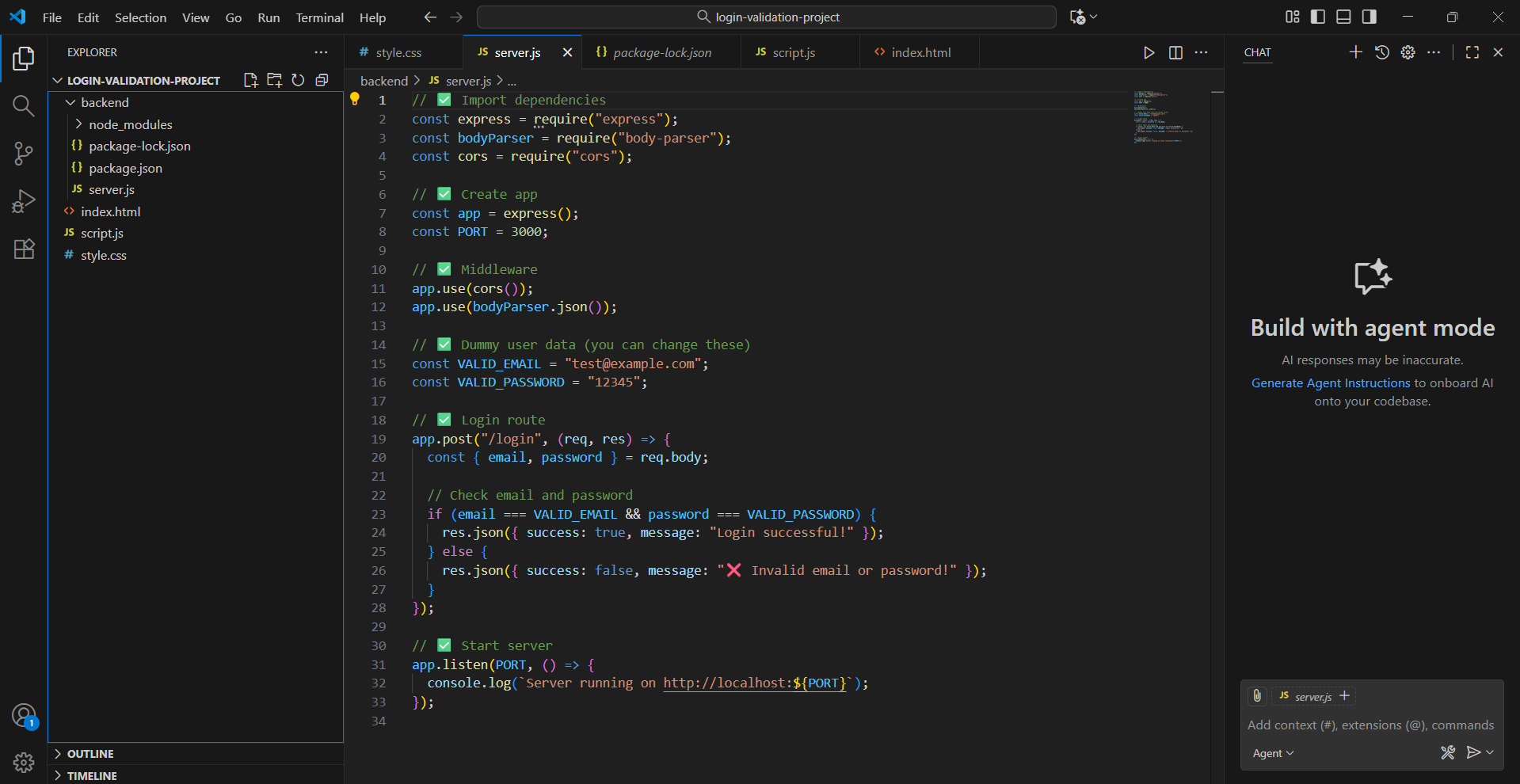
### Reflection

AI helped by quickly generating a responsive and accessible login form with validation. I manually improved the CSS design for better alignment and added a custom validation message. Additionally, I ensured proper ARIA labels and form semantics for accessibility.

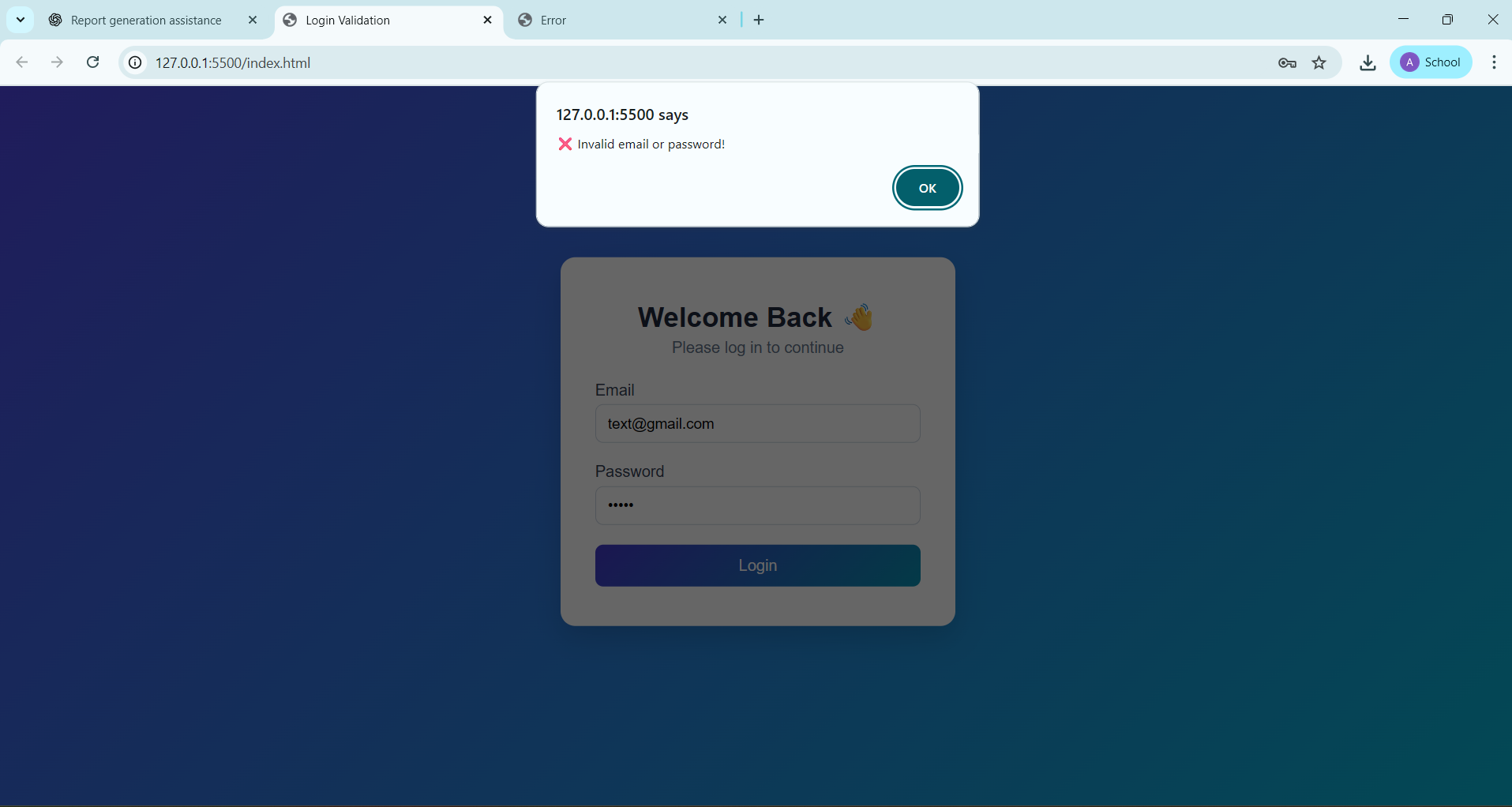
## 4. Part B – Back-End (5 Marks)

Use an LLM to generate a backend Express route to:  
- Accept { email, password }  
- Validate input  
- Return success/failure responses

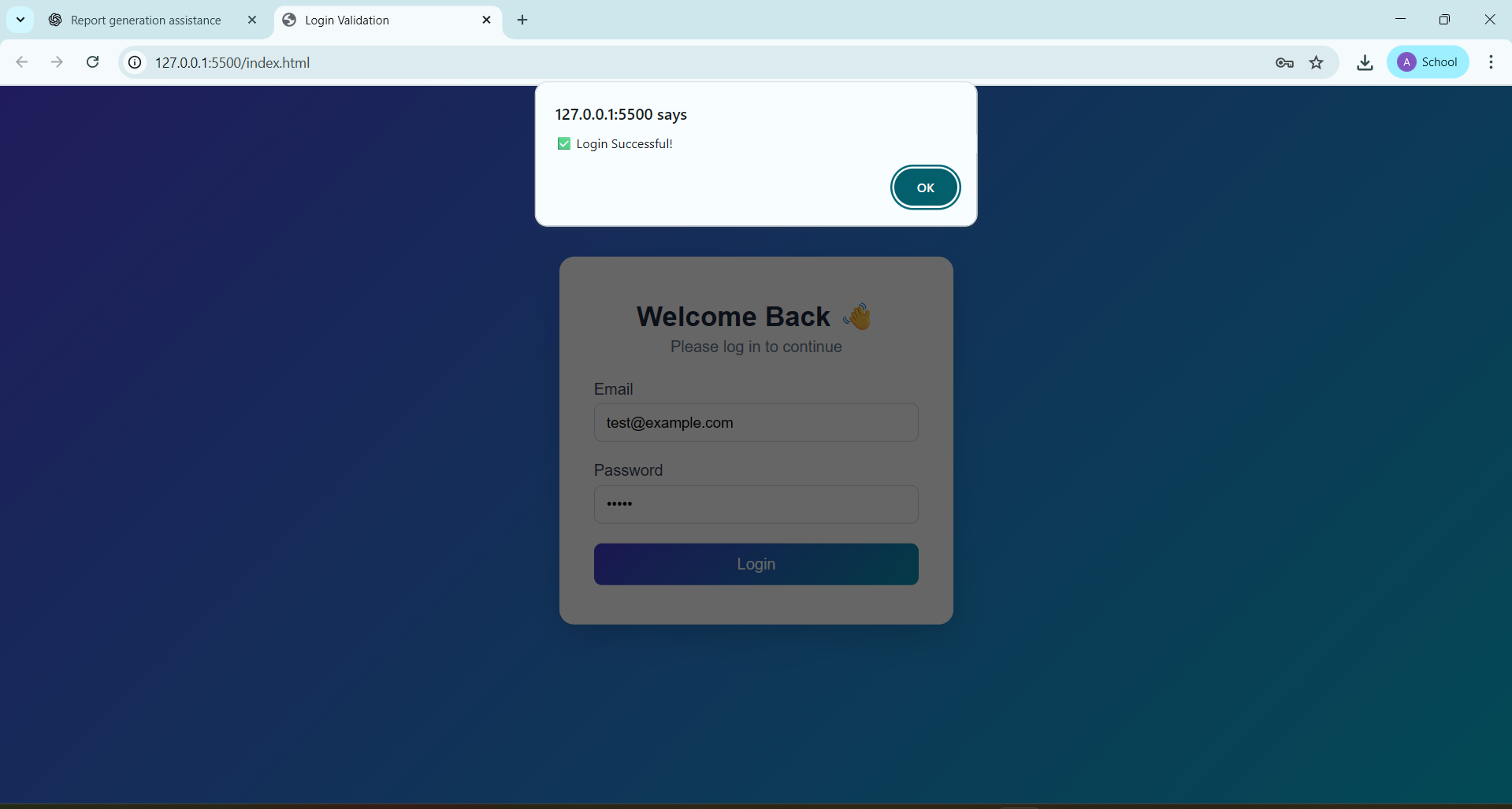
### Generated Code (Node.js + jason)



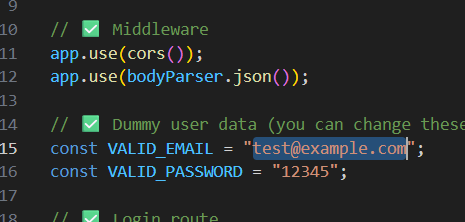
Invalidity: As password or email is wrong.



Validity: As format is correct and conditions are full filled.



As;



### Explanation of Validation

The AI used regular expressions to validate the email format and checked the password length for security. It also returned proper HTTP status codes (200 and 400) for success or failure.

### Suggested Improvement

I would implement password hashing (using bcrypt) and integrate a database for secure authentication instead of hardcoding credentials.

## 5. Conclusion

This activity demonstrated how AI tools can assist in generating both front-end and back-end components efficiently. Using LLMs accelerated the development process while allowing room for critical thinking to improve and secure the generated code.