Environment:

- **Node.js Version**: v22.11.0
- **NPM Version**: 10.9.0
- **Browser Used**: Safari and Chrome (tested on both).
- **Operating System**: macOS Ventura (Version 13.6.9).
- **Editor**: Visual Studio Code (with live server for testing frontend).

Testing:

Backend API Testing (using Postman)

- **Postman**:
    - Tested the **/send-email** route by providing email data (To, CC, BCC, Subject, Body) and verified the email was sent successfully.
    - Tested the **/mailboxes** route to fetch the list of mailboxes and confirmed it returned correct data.
    - Tested the **/mailboxes/:mailbox** route for specific mailboxes, including empty ones, to verify the correct behavior and error handling.

Frontend Functionality Testing

- **Send Email**:
    - Entered values in the "Compose Email" form.
    - Verified that the email was successfully sent by receiving it in the recipient's inbox.
    - Tested edge cases such as leaving some fields empty (CC or BCC).
- **View Mailboxes**:
    - Loaded the page and verified that the list of mailboxes (ex: Inbox, Drafts) was dynamically fetched from the backend.
- **View Inbox**:
    - Clicked on a mailbox to view the messages.
    - Verified the correct emails were displayed in the inbox section.
- **Toggle Mailbox Messages**:
    - Clicked on the same mailbox again to hide its messages, confirming that the toggle functionality works.
- **Error Handling**:
    - Tested scenarios like:
        - Clicking on an empty mailbox (ex: Drafts).
        - Server issues (shutting down the backend when mailbox is empty).
    - Verified that user-friendly error messages were displayed.

Benefits of AJAX:

This project uses AJAX via the Fetch API, which provides some benefits:

- **No Page Reloads**:
  - Data is fetched and displayed dynamically without reloading the page, creating a seamless user experience.
- **Faster Performance**:
  - Only the required data is sent and received, minimizing network usage and improving performance.
- **Better User Interactions**:
  - Features like toggling mailbox messages and displaying success/error messages make the application more interactive.

Frontend Functionality

The webmail client provides the following functionality:

1. **Send Email**:
   - A form allows the user to compose and send an email, with fields for To, CC, BCC, Subject, and Message.
   - On submission, the form sends the email data to the backend and displays a success or error message.
2. **View Mailboxes**:
   - Fetches a list of mailboxes from the backend and displays them dynamically in the Mailboxes section.
3. **View Messages in a Mailbox**:
   - When a mailbox is clicked, it fetches the last 10 messages from that mailbox and displays them in the Inbox section.
   - If the mailbox is empty, it displays a message: No messages in this mailbox.
4. **Toggle Messages**:
   - Clicking on the same mailbox again hides its messages, providing a clean interface.
5. **Error Handling**:
   - Displays user-friendly alerts if there are issues with the backend or if no messages are found.