Environment Details

a. Node.js Version

- **Version**: v20.18.0

b. NPM version

- **Version**: 10.8.2

b. Browser Details

- **Browser**: Google Chrome
- **Version**: Version 131.0.6778.86 (Official Build) (64-bit)

c. Operating System

- **OS Name**: Microsoft Windows 11 Pro
- **OS Version**: 10.0.22631 Build 22631

d. Computer Architecture

- **Architecture**: x64
- **Processor**: Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz
- **RAM**: 16.0 GB (15.8 GB usable)

e. IMAP and SMTP libraries: Custom implementation or third-party libraries

f. Testing Tool: cURL for making HTTP requests

g. Express version: 4.21.1

Test Plan:

http://localhost/mailboxes:

get a list of mailboxes from Gmail from the left

http://localhost/messages/INBOX/

get all messages in the inbox

GET - http://localhost/messages/INBOX/1 [should work for all messages in the INBOX based on ID]

get a particular message from the INBOX

DELETE - http://localhost/messages/INBOX/1 [should work for all messages in the INBOX based on ID]

Delete a particular message from the INBOX

http://localhost/contacts

get the list of contacts

POST - Add a Contact http://localhost/contacts

Should allow addition of any contact

DELETE – delete http://localhost/contacts/1

Should allow the deletion of the first contact

PUT – update http://localhost/contacts

Should allow the updation of name and email ID of the selected contact, it should show on the list on the right

short description of how AJAX helps your web application

Real-Time Data Fetching: AJAX enables the fetching of emails, contacts, and other mailbox data from the server (such as Gmail IMAP/SMTP) without requiring a page reload. For example, when a user opens their mailbox or inbox, AJAX ensures that the data is retrieved in the background and displayed immediately on the page. This results in a seamless experience without any interruptions.

Efficient User Interactions: Actions like deleting or updating emails and contacts are handled asynchronously, allowing the user interface to remain responsive while waiting for the server's response. This ensures that the user can continue interacting with the application without delays, even as backend operations are being processed.

Reduced Latency: By allowing communication between the client and server without full page reloads, AJAX minimizes the time users wait for updates. This reduces perceived latency, making interactions feel faster and more responsive, which is essential for a smooth user experience in real-time applications.

Enhanced User Experience: The dynamic nature of AJAX enables a more interactive and fluid user experience. For instance, actions like adding, updating, or deleting contacts are performed instantly and reflected on the interface without requiring a page refresh. This ensures that the user experience remains smooth and uninterrupted, increasing user satisfaction.