Assignment No : B1

Test Scenario: Gmail Login Page Functionality

Name : Prachi Sachin Said

Roll No : 41471

Objective

Verify the login functionality of the Gmail login page to ensure that users can successfully sign in with valid credentials and are prevented from logging in with invalid ones.

Preconditions

- Gmail login page URL is accessible (e.g., https://mail.google.com).
- User has valid and invalid test credentials.

Test Cases

Test Case 1: Successful Login with Valid Credentials

Description: Ensure the user can successfully log in using a valid email and password.

Step 1: Navigate to Gmail login page.

Step 2: Enter a valid email address.

Step 3: Enter a valid password.

Step 4: Click Next/Sign in.

Expected Result: User is redirected to the Gmail inbox.

Test Case 2: Login Attempt with Invalid Email Format

Description: Verify that users cannot proceed with an invalid email format.

Step 1: Navigate to Gmail login page.

Step 2: Enter an invalid email format (e.g., testemail@.com).

Step 3: Click Next.

Expected Result: Error message is displayed indicating "Enter a valid email".

Test Case 3: Login with Non-Registered Email

Description: Ensure that the user cannot log in with an email address that is not registered in the system.

Step 1: Enter a non-registered email address.

Step 2: Click Next.

Expected Result: Error message is displayed: "Couldn't find your Google Account."

Test Case 4: Login Attempt with Incorrect Password

Description: Verify that the user is prevented from logging in when an incorrect password is used. Step 1: Enter a valid email.

Step 2: Enter an incorrect password. Step 3:

Click Next/Sign in.

Expected Result: Error message is displayed: “Wrong password. Try again.”

Test Case 5: Login with Empty Email Field

Description: Verify that the email field cannot be left blank. Step 1: Open Gmail login page.

Step 2: Leave the email field empty. Step 3:

Click Next.

Expected Result: Error message is displayed: “Enter an email or phone number.”

Test Case 6: Login with Empty Password Field

Description: Ensure the user cannot log in with an empty password field. Step 1: Enter a valid email address.

Step 2: Leave the password field empty. Step 3:

Click Next/Sign in.

Expected Result: Error message is displayed: “Enter your password.”

Test Case 7: Verify “Forgot Password” Functionality

Description: Ensure the user can access the “Forgot Password” flow. Step 1: On the login page, click the Forgot password? link.

Step 2: Follow the steps to reset the password.

Expected Result: User is redirected to the password recovery flow.

Test Case 8: Ensure CAPTCHA is Triggered After Multiple Failed Attempts

Description: Verify that CAPTCHA is triggered after several unsuccessful login attempts. Step 1: Enter a valid email and an incorrect password multiple times (e.g., 5 times).

Expected Result: CAPTCHA is displayed to prevent further login attempts.

Post-conditions

- Ensure test accounts are logged out and data is reset where applicable.

Expected Result:

Upon clicking ”Settle Up,” the group expenses should be calculated and settled automatically. The application should update the balances of all group members, reflecting the amount they owe or are owed. The user should see a success message indicating that the settlement was successful.

Actual Result:

Instead of settling the expenses, the application returns a ”500 Internal Server Error,” and the balances remain unchanged. The crash interrupts the process, making it impossible to complete the settlement.

Attachments:

The following items are attached to help with troubleshooting:

- **Screenshot:** A screenshot showing the ”500 Internal Server Error” message displayed in the browser.
- **Console Logs:** The browser’s console log output at the time of the crash.
- **Backend Logs:** Server logs showing the exact exception that caused the error.

Possible Cause (If Known):

The error is likely due to an unhandled exception in the backend code. It may be caused by missing or incorrect validation of input data during the settlement process, or an issue with the database query that calculates the user balances.

Suggested Fix (Optional):

- Review the backend error logs to identify the root cause of the exception.
- Ensure that all input data passed to the settlement function is properly validated.
- Verify that the database queries related to the calculation of group balances are correctly structured and handled.

Assigned To:

Backend Developer – Responsible for investigating and resolving the issue.

Status:

Open – The defect has been logged and is pending investigation.

Comments:

This issue impacts a critical feature of the application and needs to be addressed as a priority. Users are currently unable to settle their group expenses, which could lead to frustration and abandonment of the app if not fixed quickl