Manual Testing?

-What are the requirements?

-How did the software deviate from the requirements (bugs)?

-Main goal is to ensure the software is bug free

-Write test cases:

--guide the tester through a sequence of steps to test functions and different scenarios within the software application. Writing good test cases is important as it makes executing the tests go smoothly and ensures good test coverage.

--Good test cases should be repeatable.

-Conduct the tests

--Using TestLodge, you can follow the test cases and mark each test as passed, failed, or skipped. When doing manual testing, it’s important to keep notes on what happens when a test fails.

--testlodge makes it easy to add in these details directly during the test execution

-Log good bug reports

--as you encounter bugs, you need to log information for the development team about the defect. Writing good bug reports helps.

TEST CASES:

-Guide the tester through a sequence of steps to validate whether a software application is free of bugs and working as required by the end user.

-Use a strong title

-Include a strong description

-include assumptions and preconditions

--you should include any assumptions that apply to the test and any preconditions that must be met prior to the test being executed. Information can include which page the user should start the test on, dependencies on the test environment, and any special setup requirements that must be done before running the test.

-Keep the steps clear and concise.

-Include the expected result

-Make it reusable

**Test Case 1**

**Title**: Employee Signup – Authenticate Successfully

**Description:**A new user should be able to successfully signup

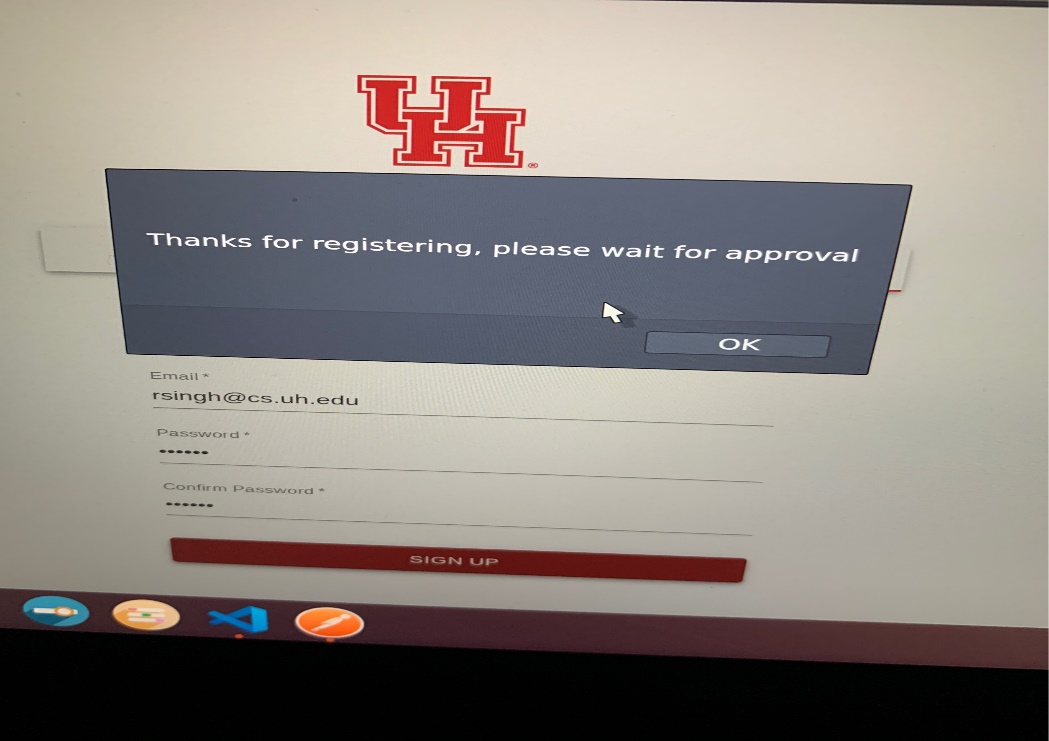
**Precondition:** the user must not have already signed up with the same credentials

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to login page
2. Enter Employee ID of the registered user
3. Enter the password of the registered user
4. Click ‘Sign In’

**Expected Result:**A page displaying the user has registered



**Test Case 2**

**Title**: Employee Login – Authenticate Successfully

**Description:**A registered user should be able to successfully login

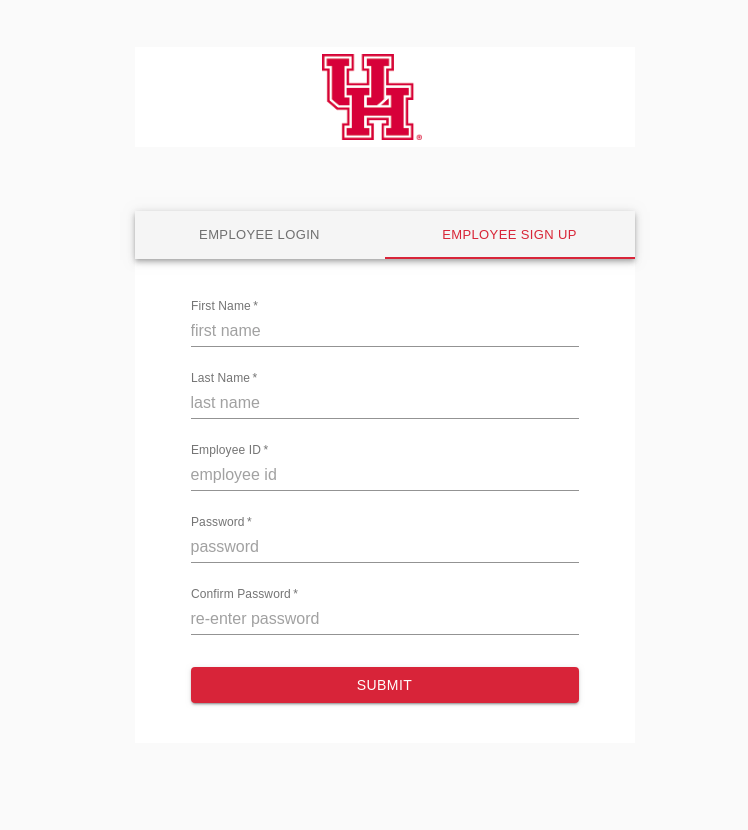
**Precondition:** the user must already be registered with an employee id, name, and password

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to login page
2. Enter the Employee ID of the registered user
3. Enter the password of the registered user
4. Click ‘Sign In’

**Expected Result:**A page displaying the user has signed in



**Test Case 3**

**Title**: Logout – Logout Successfully

**Description:**A registered user should be able to successfully logout

**Precondition:** the user must already be logged in

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to logout button
2. Click the ‘Logout’ button.

**Expected Result:**A page displaying the user has successfully logged out

**Test Case 4**

**Title**: Create/remove admin – Authenticate Successfully

**Description:**A super admin can do this

**Precondition:** The user must be a super admin.

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to create/remove admin button
2. In the text field, detail which one is being created/removed
3. Click ok

**Expected Result:**A page displaying an admin was successfully created or removed

**Test Case 5**

**Title**: Approve/Deny admin– Authenticate Successfully

**Description:**A Super Admin can do this

**Precondition:** the user must be a super admin

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to Approve/Deny Admin Button
2. Click if a request is approved or denied
3. Click ‘Sign In’

**Expected Result:**A page displaying the approval or denial was successful or not

**Test Case 6**

**Title**: Update Admin Role

**Description:**A registered user should be able to successfully login

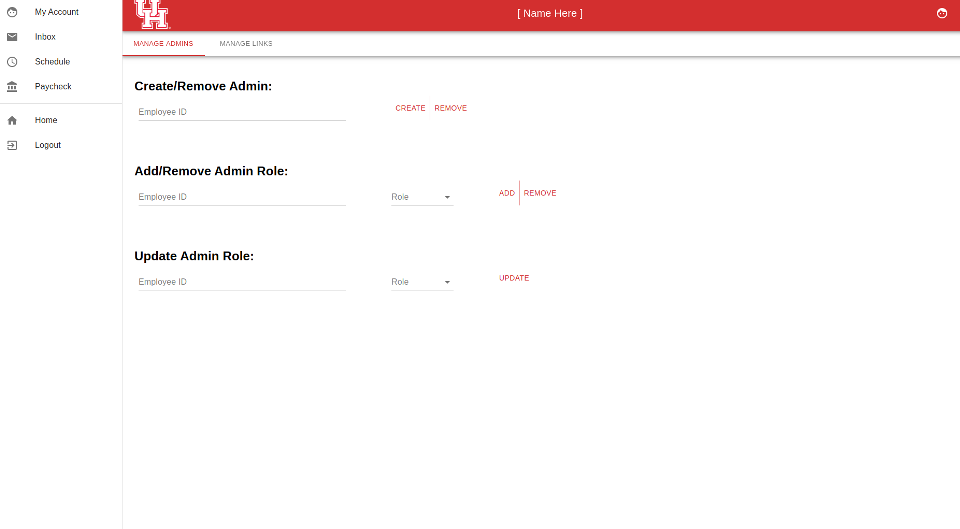
**Precondition:** the user must already be registered with an email address and password.

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to login page
2. In the ’email’ field, enter the email address of the registered user.
3. Click the ‘Next’ button.
4. Enter the password of the registered user
5. Click ‘Sign In’

**Expected Result:**A page displaying the gmail user’s inbox should load, showing any new message at the top of the page.



**Test Case 8**

**Title**: Add Link

**Description:**A super admin can ad a link

**Precondition:** the user must be a super admin

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to Add links page
2. Select a column to add a link
3. Click the ‘add link’ button

**Expected Result:**A page displaying the link was successfully added is displayed

**Test Case 9**

**Title**: Remove Link

**Description:**A super admin can remove a link

**Precondition:** the user must be registered as a super admin

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to links Page
2. Navigate to the remove link column
3. Click the button for the corresponding link to be removed

**Expected Result:**A page displaying the link was successfully removed

**Test Case 10**

**Title**: Update Link Role

**Description:**A super admin can update the link role

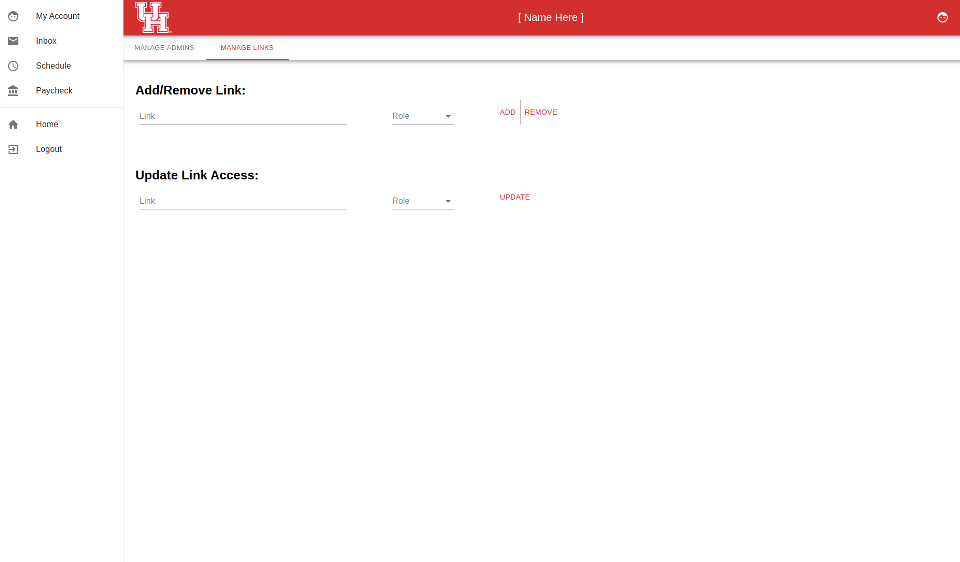
**Precondition:** the user attempting this must be registered as a super admin

**Assumption:** a supported browser is being used.

**Test Steps:**

1. Navigate to links page
2. Navigate to the Update Link Role column
3. Find the corresponding link role to be updated
4. Update the link role

**Expected Result:**A page displaying the changing of the link role was successful



-Bug reports:

--Should have a strong title, steps to replicate the bug (often the test case steps), an expected and actual result, and any relevant attachments that will help the team understand the issue (screenshots, screen recordings, export files, etc.)

**Bug Report 1:**

**Title**: Login Page

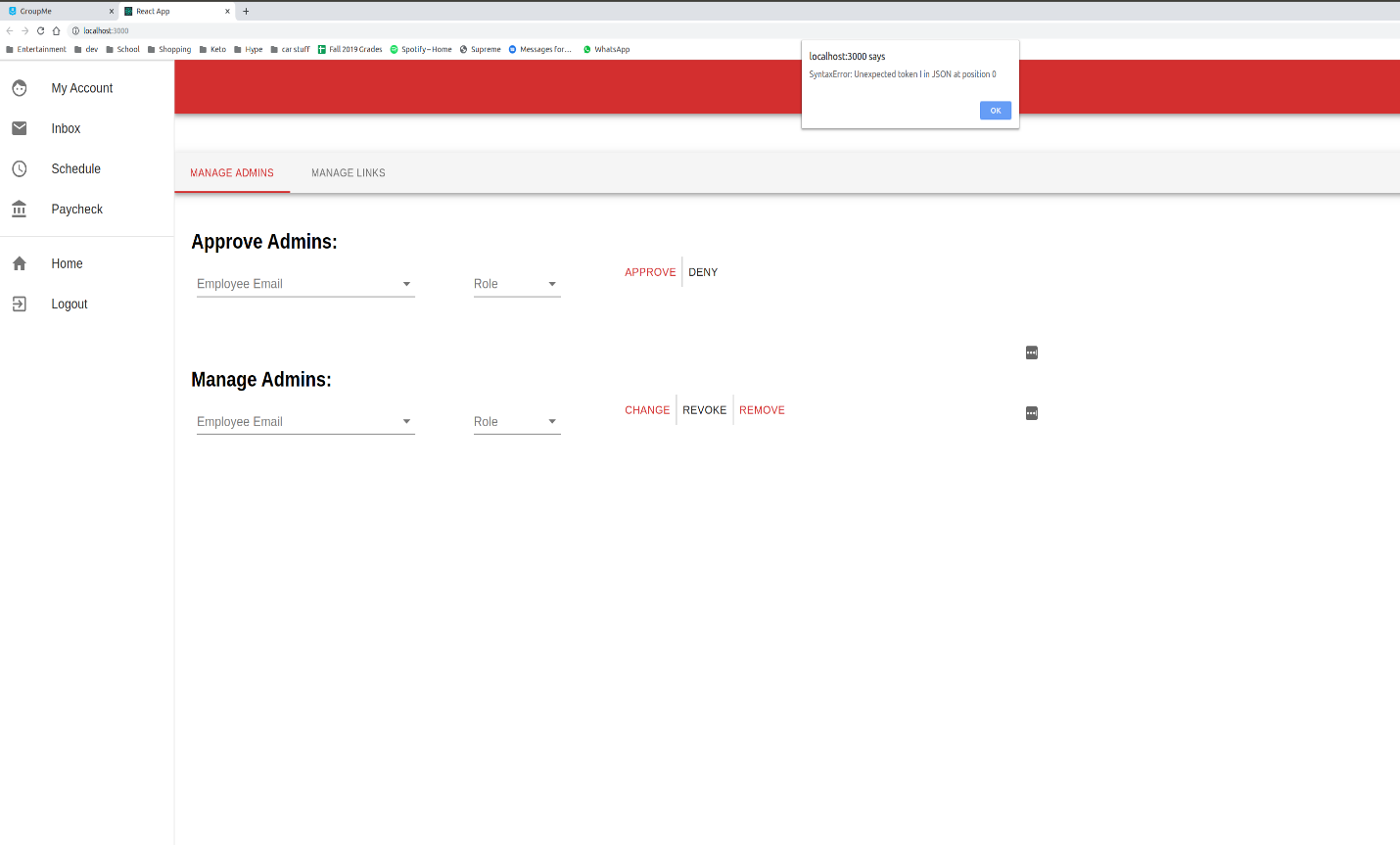
**Description:**Login Page Does not load the links properly on first login

**Precondition:** the user is attempting to login for the first time

**Assumption:** a supported browser is being used.

**Fix:** Manual refresh of the page

**Expected Result:**successful login with all links displayed

**Bug Report 2:**

**Title**: Initial Login

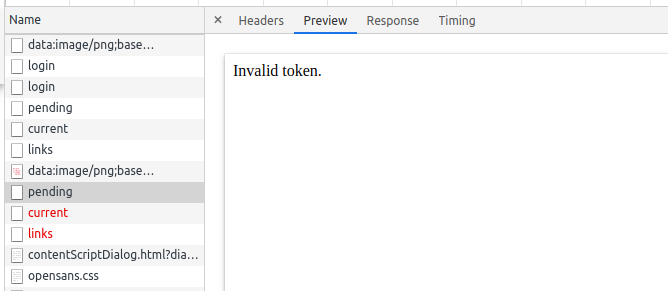
**Description:**Invalid token for three requests on initial login

**Precondition:** the user is attempting to login for the first time

**Assumption:** a supported browser is being used.

**Fix:** Fixed the error, it was caused because the object fetchOptions being called before outside of component hook. Just moved them inside the component function and that fixed the error, them being declared outside made the x-auth-token equal to null, but inside it sets

**Expected Result:**successful login with all links displayed



**Bug Report 3:**

**Title**: Login

**Description:**Login crash when using invalid credentials

**Precondition:** the user is attempting to login for the first time

**Assumption:** a supported browser is being used.

**Fix:** login is fixed when using invalid credentials

**Expected Result:**successful login with all links displayed

**Bug Report 4:**

**Title**: Signup Error after fixing login error

**Description:**signup throws this error: ReferenceError: Cannot access 'res' before initialization on even a valid signup. although it did register the user it threw that message up

**Precondition:** the user is attempting to login for the first time

**Assumption:** a supported browser is being used.

**Fix:** Move the if statement above the const res, then throw new Error(‘invalid register’)

**Expected Result:**successful login with all links displayed

