Technical Assessment

# 1.2. Login Functionality

### Acceptance Criteria:

Scenario1: Valid Credentials

Given a registered user with valid credentials get it from AD (active directory),

When the user enters the correct username and password,

And he clicks on the button sign in.

Then the user should be successfully logged in.

And the application should navigate to the home page.

Scenario2: Invalid Username

Given a registered user,

When the user enters an incorrect username,

And he clicks on the button sign in,

Then the user should not be logged in.

And an appropriate error message should be displayed.

Scenario3: Invalid Password

Given a registered user,

When the user enters an incorrect password,

And he clicks on the button sign in,

Then the user should not be logged in.

And an appropriate error message should be displayed.

Scenario4: Empty Fields

Given a registered user,

When the user submits the login form with empty fields,

And he clicks on the button sign in,

Then the user should not be logged in.

And an appropriate error message should be displayed.

Scenario5: Account Lockout

Given repeated unsuccessful login attempts for a user,

When the maximum allowed (3 times) attempts are reached,

Then the fields username , password and the button “sign in” are disable.

And the user should receive a notification about the lockout (“your account is temporarily locked ,please wait 5 seconds and repeat again).

When the user wait 5 seconds,

Then the fields username , password and the button “sign in” are enable.

And the user can log in .

### Test Plan and test strategy

#### 1. Objective:

Ensure the secure and successful functioning of the login feature with a focus on user Valid Credentials.

#### 2. Resources:

* Test Environments: List of Environments example local environment, qs, dev instances ..
* Test Data:
* Test data for valid (example username= rdahs password= password )
* Test data for invalid credentials that not from Active directory (example username =unknown, password= unknown)

#### 3.Test Cases:

A. Successful Login:

Precondition:

* Ensure the application is accessible.
* The test data valid (username and password) get from Active directory

Test Steps:

| Step | Action | Expected Result |
| --- | --- | --- |
| 1 | Navigate to the login page. | Login page is displayed. |
| 2 | Enter a valid username (rdahs) | - |
| 3 | Enter a valid password (password) | - |
| 4 | Click the "Sign in" button. | User is logged in.  User is redirected to the home page. |

B. Unsuccessful Login Attempts (Invalid Credentials):

Precondition:

* Ensure the application is accessible.
* The test data invalid (username and password) is not from Active directory

Test Steps:

| Step | Action | Expected Result |
| --- | --- | --- |
| 1 | Navigate to the login page. | Login page is displayed. |
| 2 | Enter an invalid username (unknown) | - |
| 3 | Enter a valid password (password) | - |
| 4 | Click the "sign in" button. | Error message is displayed.  User not able to logged in |
| 5 | Enter a valid username (rdahs) | - |
| 6 | Enter an invalid password (unknown) | - |
| 7 | Click the "sign in" button. | Error message is displayed.  User not able to logged in |
| 5 | Keep username empty | - |
| 6 | keep password empty | - |
| 7 | Click the "sign in" button. | Error message is displayed.  User not able to logged in |

C. Security (Account Lockout):

Precondition:

* Ensure the application is accessible.

Test Steps:

| Step | Action | Expected Result |
| --- | --- | --- |
| 1 | Perform a 3 times of unsuccessful login attempts. | User receives a notification about the account lockout : “your account is temporarily locked ,please wait 5 seconds and repeat again”  The following fields are disable:  Username , password , sign in button |
| 2 | Refresh the page | The following fields are disable:  Username , password , sign in |
| 3 | Wait 5 seconds. | * The following fields return enable:   Username , password , sign in button   * User can put his credentials |

#### 4.Endpoint testing:

1. Valid Login:

* Endpoint: POST /api/login
* Test Steps:

1. Send a POST request to the login endpoint with valid username and password.
2. Verify that the response status code is 200 (OK).

2.Invalid Login - Incorrect username and incorrect password:

* Endpoint: POST /api/login
* Test Steps:

1. Send a POST request to the login endpoint with an invalid username and password.
2. Verify that the response status code is 401 (Unauthorized).

3.Missing Username or Password:

* Endpoint: POST /api/login
* Test Steps:

1. Send a POST request to the login endpoint without providing a username or password.
2. Verify that the response status code is 400 (Bad Request).

#### 5.Defect Reporting:

##### Defect1:

Defect Title: Login Failure with Valid Credentials

Steps to Reproduce:

* Open the web browser and navigate to the application login page.
* Enter a valid username in the username field.
* Enter a valid password in the password field.
* Click the "Sign In" button.

Expected Result:

The user should be successfully logged in, and the application's home page should be accessible.

Actual Result:

The user is not logged in, and an error message is displayed indicating login failure.

Screenshots: Attach screenshots or video showing the error.

# 1.2.“If we can see the city of origin of selected employees in a list.” Functionality

### Acceptance Criteria:

Scenario1: City of Origin Display in employees list

Given I navigate to the employee list,

When I inspect each entry,

Then the column “City” is displayed,

And the value of city of origin should be displayed for each respective employee.

Scenario2: City of Origin for Selected Employee

Given I have selected a specific or multiple employee(s) from the list,

When I click on the button “view selected data”,

Then the city of origin for each selected employee(s) should be clearly visible.

Scenario3: Responsive Design

Given I am using the application,

When I resize the browser window to various dimensions,

Then the city of origin should remain visible without overlapping or layout issues.

### Test Plan and test strategy

#### 1. Objective:

Validate that the city of origin is visible for selected employees in the employee list.

#### 2. Resources:

* Test Environments: List of Environments example local environment, qs, dev instances ..
* Test Data: some list of employees has city data.

#### 3.Test Cases:

A. Display of city origin in selected employee List:

Precondition:

* Ensure the application is accessible.
* there are employees with diverse city of origin entries in the database.

| Step | Action | Expected Result |
| --- | --- | --- |
| 1. | Open the web browser and log in to the application. | Successfully logged in, and the application is accessible. |
| 2. | Navigate to the employee list. | The column “City” is displayed,.  The values of city origin, get it from db ,are visible for each employee in the list. |
| 3. | Select a specific employee from the list.  Click on the button “view selected data” | the selected employee is displayed without errors in the box with the format : Firstname is from City |
| 4. | Select multiple employees from the list.  Click on the button “view selected data” | Multiple employees are selected without issues with the format :  Firstname is from City |
| 5. | Resize the browser window to various dimensions. | The city of origin remains visible and does not overlap with other elements on different screen sizes. |

Postcondition:

* The city of origin is displayed for selected employees.
* The application remains in a stable state.

#### 4.Endpoint testing:

1.Get List of Employees with City of Origin:

* Endpoint: GET /api/employees
* Test Steps:

1. Send a GET request to retrieve the list of employees.
2. Verify that the response status code is 200 (OK).

2.Select Multiple Employees and Retrieve Details:

* Endpoint: GET /api/employees/:id
* Test Steps:

1. Send multiple GET requests to retrieve details of selected employees.
2. Verify that the response status code is 200 (OK) for each request.

#### 5.Defect Reporting:

##### Defect1:

Defect Title: Incorrect Column Title for City in Employee List

Steps to Reproduce:

* Open the application and navigate to the employee list page.
* Observe the column headers in the employee list.

Expected Result:

The column header for the "City" should be displayed as "City."

Actual Result:

The column header for the "City" is displayed as "Title."

Screenshots: Attach screenshots highlighting the incorrect column title in the employee list.

##### Defect2:

Defect Title: Displaying "null" for City of Employee without City

Steps to Reproduce:

* Open the web browser and navigate to the employee list.
* Select an employee who does not have a city listed.
* Click on the "View Selected Data" button.

Expected Result:

The application should handle cases where the selected employee does not have a city (show specific message), and the display should not show "null" for the city.

Actual Result:

The application displays "null" for the city of the selected employee who does not have a city.

Screenshots: Attach screenshots showing the incorrect display of "null" for the city.

##### Defect3:

Defect Title: Non-Responsive Behavior on Window Resizing.

Steps to Reproduce:

* Open the web browser and navigate to list employees.
* Resize the browser window to various dimensions.

Expected Result:

The application should adjust its layout and remain usable at different screen sizes.

Actual Result:

The application does not respond appropriately to window resizing and hide some elements.

Screenshots: Attach screenshots illustrating the non-responsive behavior at different window sizes.

# 3. Do you agree with the acceptance criteria? What are you missing? Put one example.

Yes, I agree with most of the acceptance criteria. The specified conditions cover essential aspects of the feature and provide a solid foundation for testing.

However, one area that appears to be missing in the acceptance criteria is the handling of error messages in case of invalid login attempts. It would be beneficial to explicitly state the expected error messages and behavior when incorrect credentials are provided.

# 4. Test automation implementation:

You can find the automatic test at: https://github.com/RahmaDahs/ui-test-assessment-solution.git