AtlantBH Task

TEST PLAN

Project Name: Contact List Application

Test Types:
- Manual UI tests
- API tests
- Automated smoke tests

Objective

The objective of this test plan is to verify the functionality of the Contact List application by performing UI and API testing. The test plan covers real user flows, including positive and negative scenarios.

Scope

The testing will focus on the Contact List application available at https://thinking-tester-contact-list.herokuapp.com/.

Test Environment

Browser: Google Chrome
Testing Framework: Postman for API testing, Selenium with C# and NUnit for UI testing
Programming Language: C#
Automation Tools: Postman, Selenium WebDriver, ChromeDriver

Test Schedule

Start Date: 19.01.2024.
End Date: 22.01.2024.

Test Resources
QA Test Engineer: Almedina Hadžan

UI TEST CASES

Positive tests:

1. Test Case 1: Validate Sign Up flow

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Click on the Sign Up button	Add User page is displayed
3. Populate the following fields: - First Name (mandatory field) - Last Name (mandatory field) - Email - Password	The following fields are populated: - First Name (e.g. John) - Last Name (e.g. Doe) - Email - Password
4. Click on the Submit button	The user is successfully created

2. Test Case 2: Validate User can Login into the application

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Fill in the following fields: - Email - Password	The following fields are populated: - Email - Password
3. Click on the Submit button	The user is successfully logged in

3. Test Case 3: Validate that the User can Add a New Contact

Steps	Expected result
1. Click on the Add a New Contact	Add Contact page is displayed
2. Populate the following fields: - First Name (mandatory field) - Last Name (mandatory field) - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code	The following fields are populated: - First Name (e.g. John) - Last Name (e.g. Doe) - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code

Steps	Expected result
- Country	- Country
3. Click on the Submit button	The user is redirected to the Contact List page
4. Validate the new contact	The new contact is successfully added

4. Test Case 4: Validate the User can Delete Contact

Steps	Expected result
1. Click on a User from the Contact list	Contact Details page is displayed
2. Click on Delete Contact button	The following pop-up message is displayed: *Are you sure you want to delete this contact?*
3. On the pop-up message click OK	Pop-up message is closed and the User is redirected to the Contact List
4. Validate the table	The User is no longer displayed in the table

5. Test case 5: Validate the User can Edit Contact

Steps	Expected result
1. Click on a User from the Contact list	Contact Details page is displayed
2. Click on Edit Contact button	The following pop-up message is displayed: *Are you sure you want to delete this contact?*
3. Click on the Submit button	The Contact is successfully edited
4. Click on the Return to Contact List button	The User is redirected to the Contact List page
5. Validate the table	The User is edited

6. Test Case 6: Validate the User can Logout from the Contact List page

Steps	Expected result
Precondition: The user is already logged in	
1. Click on the Logout button	The user is logged out and redirected to the Contact List App page

7. Test Case 7: Validate the User can Logout from the Add Contact page

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to Add Contact page	The Add Contact page is displayed
2. Click on the Logout button	The user is logged out and redirected to the Contact List App page

8. Test Case 8: Validate the User can Logout from the Contact details

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to any Contact details page	The Contact details page is displayed
2. Click on the Logout button	The user is logged out and redirected to the Contact List App page

9. Test Case 9: Validate the User can Logout from the Edit Contact page

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to any Contact details page	The Contact details page is displayed
2. Click on the Edit Contact button	The Edit Contact details page is displayed
3. Click on the Logout button	The user is logged out and redirected to the Contact List App page

10. Test Case 10: Cancel Adding a New Contact

Steps	Expected result
1. Navigate to the Add Contact page	
2. Click on the Cancel button	User should be redirected to the Contact List page without adding a new contact

Negative test cases

1. Test Case 11: Validate Sign up flow with invalid email

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Click on the Sign Up button	Add User page is displayed
3. Populate the First Name (mandatory field) correctly	The First Name is entered successfully
4. Populate the Last Name (mandatory field) correctly	The Last Name is entered successfully
5. Populate Email in wrong format (e.g. jdoe@com)	The Email is entered in wrong format
6. Populate the Password field correctly	The Password is populated
7. Click on the Submit button	The following error message is displayed:" User validation failed: email: Email is invalid"

2. Test Case 12: Validate Sign up flow with an invalid password

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Click on the Sign Up button	Add User page is displayed
3. Populate the First Name (mandatory field) correctly	The First Name is entered successfully
4. Populate the Last Name (mandatory field) correctly	The Last Name is entered successfully
5. Populate Email in the correct format (e.g. jdoe@fake.com)	The Email is entered in the correct format
6. Leave the Password field empty	The Password is empty
7. Click on the Submit button	The following error message is displayed: " User validation failed: password: Path `password` is required."

3. Test Case 13: Validate Sign Up flow with an existing user

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Click on the Sign Up button	Add User page is displayed
3. Populate the following fields the same as for the existing user: - First Name (mandatory field) - Last Name (mandatory field) - Email - Password	The following fields are populated: - First Name (e.g. John) - Last Name (e.g. Doe) - Email - Password
4. Click on the Submit button	The following error message is displayed: „Email address is already in use"

4. Test Case 14: Validate Login flow with invalid email

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Fill in the email in the wrong format (e.g. jdoe@com)	The Email is entered in the wrong format
3. Fill in the password correctly	The Email is entered in correctly
4. Click on the Submit button	The following error message is displayed: „Incorrect username or password"

5. Test Case 15: Validate Login flow with invalid password

Steps	Expected result
1. Open https://thinking-tester-contact-list.herokuapp.com/	Contact List App page is displayed
2. Fill in the Email correctly	The Email is entered in the correct format
3. Fill in the Password wrong	The Password is entered in the wrong format
4. Click on the Submit button	The following error message is displayed: „Incorrect username or password"

6. Test Case 16: Add Contact with Missing Information

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to the Add Contact page	Add Contact page is displayed with the following fields: - First Name - Last Name - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code - Country
2. Without populating any field click on the Submit button	The following error is displayed: „Contact validation failed: firstName: Path `firstName` is required., lastName: Path `lastName` is required. "

7. Test Case 17: Adding a contact with an incorrect Email

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to the Add Contact page	Add Contact page is displayed with the following fields: - First Name - Last Name - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code - Country
2. All fields populate correctly and the Email field in the wrong format (e.g. jdoe@com)	The following error message is displayed:" User validation failed: email: Email is invalid"

8. Test Case 18: Adding a contact with an incorrect birth date format

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to the Add Contact page	Add Contact page is displayed with the following fields: - First Name - Last Name - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code - Country
2. All fields populate correctly and for the Birth Date field use the wrong format (e.g. 19871013)	The following error message is displayed: „Contact validation failed: birthdate: Birthdate is invalid"

9. Test Case 19: Adding a contact with an incorrect Postal Code

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to the Add Contact page	Add Contact page is displayed with the following fields: - First Name - Last Name - Date of Birth - Email - Phone - Street Address 1 - Street Address 2 - City - State of Province - Postal Code - Country
2. All fields populate correctly and for the Postal Code field use the wrong format (e.g. 1)	The following error message is displayed: „Contact validation failed: postalCode: Postal code is invalid"

10. Test Case 20: Validate if the User can access the Contact List even if is logged out

Steps	Expected result
Precondition: The user is already logged in	
1. Navigate to the Add Contact page	Add Contact page is displayed
2. Copy the URL	The URL is successfully copied
3. Click on the Log Out button	The user is logged out
4. Open a new tab or incognito window and paste the URL	The homepage is displayed and the user is not logged

API tests in the Postman collection

BUGS:

1. Title: The app is accessible without being logged in

Browser: Google Chrome
Device: MacBook Pro MacOS: 14.1.2
Application: https://thinking-tester-contact-list.herokuapp.com/

Steps to reproduce:
1. Login to https://thinking-tester-contact-list.herokuapp.com/
2. Navigate to any page e.g. Add Contact page
3. Copy URL e.g. https://thinking-tester-contact-list.herokuapp.com/contactList
4. Click on the Log Out button
5. Open a new tab or Incognito tab
6. Paste the URL from the step 3.
7. Validate the page

Expected result:
Step 7.: The Contact List App page (homepage) is displayed and the user is not logged in

Actual result:
Step 7.: The Contact List page is displayed as if the user is logged in even though they are not