ENSEK Automation Candidate Test Application

Test Plan Document

Version	Date	Written By	Reviewed By	Comments
1.0	18-Sep-24	Bhavani Gandhi		

Table of Contents

1. Objective

This testing effort aims to assess the functionality, performance, usability, and reliability of the ENSEK Automation Candidate Test application hosted at the given URL. This plan outlines the different aspects of the application that will be tested to ensure it meets the functional and non-functional requirements, is free of defects, and works as expected under various scenarios.

NOTE – This is a test plan template which I have worked on at a high-level as a test artifact for Exercise – 1. Please note that this plan requires a lot of modifications to reflect the actual testing needs.

2. Scope of Testing

The scope includes functional, non-functional, and usability testing of the web application. The key modules and features that will be tested include:

- Authentication: Testing login functionality.
- Main Application Functions: Core application functionalities like data input, processing, and output.
- API interactions: Testing API responses, status codes, and data integrity.
- UI/UX: Evaluating the overall user experience and layout design.
- Error Handling and Validation: Ensuring proper error messages and validation checks.
- Performance Testing: Ensuring the application performs well under normal and high-load conditions.

3. Testing Types and Strategy

3.1 Functional Testing

This will focus on validating the features and functions of the application to ensure that it works according to the specified requirements.

User Authentication:

- Verify the login functionality, including valid and invalid credentials.
- Ensure proper redirects on successful login or error messages for failures.

Data Input Forms:

- Check all forms for field validation (e.g., required fields, character limits, correct data formats).
- Submit valid and invalid data to verify proper error handling.

Core Features:

- Identify and test the core functionality of the application such as Buy Energy, Sell Some Energy, Contact

UI Elements:

- Test all buttons, links, drop-downs, browser minimise and maximise, and interactive elements to ensure they behave as expected.

3.2 Non-Functional Testing

This will focus on areas like performance, security, and scalability.

Performance Testing:

- Test the page load times and responsiveness under different network conditions.
- Simulate a large number of users or heavy data load to see how the application handles traffic.

Security Testing:

- Test for any potential security vulnerabilities such as weak password policies or potential SQL injection points.
- Ensure that user data is handled securely and that proper session management is implemented.

Cross-Browser and Device Compatibility:

- Verify that the application works across different browsers (e.g., Chrome, Firefox, Edge, Safari).
- Test responsiveness across various screen sizes and devices (mobile, tablet, desktop).

3.3 Usability Testing

This focuses on the user interface and user experience.

Ease of Navigation:

- Ensure the user can navigate through the application without confusion.
- Verify that important elements (such as buttons, links, etc.) are easy to find and use.

Clarity of Information:

- Validate that the language used is clear, concise, and understandable to users.

Layout and Design Consistency:

- Ensure that the layout remains consistent across the application and that elements are properly aligned and readable.

4. Test Plan

A detailed test plan will guide the testing process, including test cases, expected outcomes, and timeframes for execution.

4.1 Test Cases

Here is a breakdown of the test cases that will be created for each module:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Outcome
TC001	User Registration	1. Open the registration page 2. Enter email 3. Enter password 4. Enter confirm password 5. Click Register button	Email - ***** Password - **** Confirm Password - ****	The user registered successfully
TC002	User Login (Valid Credentials)	1. Open the login page 2. Enter valid credentials 3. Click Login button	Email - ***** Password - ****	The user is logged in successfully
TC003	User Registration without entering field values	1. Open the registration page 2. Click Register button		The user should see “Email field” is required and the password field is required messages
TC004	User Registration without entering confirm password field value	1. Open the registration page 2. Enter Email and password 3. Click Register button	Email - ***** Password - ****	The user should see “The password and confirmation password do not match” message.
TC005	User Login without entering field values	1. Open the sign-in page 2. Click Login button		The user should see “Email field” is required and the password field is required messages

TC006	User Login without entering password field value	1. Open the sign-in page 2. Click Login button	Email - *****	The user should see the password field is required message
TC007	Home Page validation	1. Open the application by entering the URL in the browser	URL - https://enseka utomationcand idatetest.azure websites.net/	The user should see Home, About, Contact, Register and Login menu items

Additional test cases will be added for specific functionalities as they are discovered during exploratory testing.

NOTE – Test cases will be maintained in test case management tools such as AIO, Azure DevOps, Confluence or JIRA stories. And, those details will be linked to the test plan.

5. Testing Tools

- Manual Testing: Testing will be performed manually for all functionalities.
- Automation Testing Tools: If automation is required, tools such as Postman, Playwright, Selenium WebDriver (for web-based automated functional testing) may be used.
- Load Testing Tools: For performance testing, tools like JMeter or LoadRunner could be utilized to simulate multiple users accessing the system simultaneously.
- Browser Developer Tools: For inspecting element behavior, network responses, and debugging.

6. Defect Management

Defects will be logged and tracked using a bug-tracking system (e.g., Jira, Azure DevOps, AIO, or a similar tool). Each defect will be assigned a severity and priority based on its impact on the application.

7. Test Environment

- Browsers: Latest versions of Chrome, Firefox, Edge, and Safari.
- Devices: Desktop, mobile, and tablet.
- OS: Windows, macOS, iOS, Android.
- Network: Tests will be performed under different network conditions (e.g., Wi-Fi, 4G, 3G) to measure application responsiveness.

8. Reporting

+ Daily Test Reports: Summarize daily test activities, results, and any critical issues found.
+ Defect Reports: Detailed reports on defects, severity, and steps to reproduce.
+ Final Test Report: Comprehensive report on overall testing activities, coverage, pass/fail rate, and outstanding issues.

9. Assumptions & Risks

+ It is assumed that the system requirements and testable functionalities are well-defined.
+ Risks include potential delays if test data is insufficient or unclear.

10. Conclusion

The outlined testing approach aims to ensure that the ENSEK Automation Candidate Test application is robust, user-friendly, and meets quality standards. The focus will be on validating the core features, ensuring the security and performance of the system, and providing a positive user experience.