SE302 - Software Testing and Maintenance

Homework 02

Homework Assignment

Task Description - Test Cases for Web Testing

The purpose of this assignment is to practice writing test cases for a website. For the provided website, analyze its key features, and write 10 different test cases that cover various aspects of its functionality. This will help you learn to think critically about testing and how to ensure a website works as expected in different scenarios. The website that should be used for the homework is: **https://sweetshop.netlify.app/**

Instructions:

1. Identify key features that should be tested:
   - Analyze the website and identify at least 3 key features you will focus your test cases on. Some examples of features to test could include:
     - Login/Signup forms
     - Search functionality
     - Navigation menus
     - Product or content displays
     - Contact forms
     - User profile management
   - Write your test cases based on these features.
2. Write 10 test cases:
   - Write 10 test cases in total. Your test cases should cover a variety of scenarios to ensure comprehensive testing:
     - **Positive Test Cases:** Check that the system works as expected when the input or action is correct.
     - **Negative Test Cases:** Test how the system behaves with invalid or incorrect inputs.
     - **Boundary Test Cases:** Test the limits, such as the maximum or minimum values allowed.
     - **Usability Test Cases:** Check if the site is easy to use and intuitive.
     - **Security Test Cases:** Test the website's security, such as password encryption and protection against vulnerabilities.
   - Each test case should include:

- **Test Case ID:** A unique identifier (e.g., TC_01, TC_02).
- **Test Case Title:** A brief title for the test case.
- **Description:** A brief explanation of what the test case is meant to validate.
- **Test Steps:** A step-by-step guide of what actions to perform during the test.
- **Expected Results:** What should happen when you perform the test steps.
- **Pass/Fail Criteria:** Define how you will determine whether the test passes or fails (e.g., if the login succeeds, the test passes).

Example Test Case

Test Case ID	TC_01
Test Case Title	Verify User Login
Description	Check whether a registered user can successfully log into the system using valid credentials.
Test Steps	• Navigate to the login page • Enter a valid username and password • Click the *Login* button
Expected Result	User is successfully authenticated and redirected to the home page.
Pass / Fail Criteria	• **Pass:** User is logged in and redirected correctly • **Fail:** Login fails or user is not redirected

Additional Explanations:

1. Using the 10 manually written test cases from Step 2, choose any 5 and implement them using Playwright (TypeScript or JavaScript is preferred).

   - **Requirements:**
     - Each test should match one of the manually written test cases.
     - Use Playwright's recommended test structure:
       - test.describe()

    - test()
    - expect()
  - At least one test must interact with:
    - A form (input fields + submit)
    - A button or navigation element
  - At least one test must include a negative scenario (e.g., wrong input)
  - At least one test should include a locator assertion or URL assertion

Deliverables

i. A document/report that includes:

- Test cases following the structure mentioned.
- Link to your Github repository. The repository should include implementation of your tests in Playwright following POM structure. Refer to the link shared for projects in the presentation about Playwright (similar to your projects).

Submission Format: PDF document titled: `SE302_Homework02_<StudentID>.pdf`

Important Notes: Ensure your answers are detailed and can be easily understood by each stakeholder. **Plagiarism** will not be tolerated. Your assignment should be original work! **Good luck with your assignment!**