Automation Project Documentation

Objective

The objective of this automation project is to test the happy scenario of a user logging in, adding items to his cart, and then checking out.

Tools Used

- **Selenium WebDriver**: For browser automation.
- **TestNG**: For managing and running the test cases.
- **Java**: The programming language used for writing the test scripts.

Project Structure

- **Login Page**
  - Responsible for user authentication.
  - Functions: enter username, enter password, click login button.
- **Products Page**
  - Displays a list of products.
  - Functions: sort products, add products to cart.
- **Cart Page**
  - Displays products added to the cart.
  - Functions: view cart, proceed to checkout.
- **Checkout Page**
  - Handles the checkout process.
  - Functions: fill in user details, continue checkout, finish purchase.

Test Data

- **URL**: [Swag Labs (saucedemo.com)](https://saucedemo.com)
- **Users List**:
  - Standard_user
  - Locked_out_user
  - Problem_user
  - Error_user
  - Performance_glitch_user
  - Visual_user
- **Password**: secret_sauce

Testing Process & Cycle

1. **Open the website**
   - Navigate to the website URL.
2. **Enter credentials**
   - Select random username from the usernames list.
   - Input username and password.
3. **Login**
   - Authenticate the user by clicking the login button.
   - Navigate to products page.
4. **Sort products by price**
   - Sort the products displayed on the Products page by price (low to high).
5. **Select lowest price item and add it to Cart**
   - Identify and add the product with the lowest price to the cart.
6. **Open cart**
   - Navigate to the Cart page to review added products.
7. **Proceed to checkout**
   - Click checkout button
   - Proceed to the Checkout page.
8. **Add required data in information screen**
   - Input necessary user details for checkout (first name – last name – ZIP code).
9. **Navigate to checkout overview**
   - Click Continue to navigate to checkout overview page.
10. **Complete the Order**
    - Click finish to Complete the purchase.

Notes on the Users List

- **Standard_user**
  - All cycle works fine; the user can login, sort products, add items to cart, and checkout successfully.
- **Locked_out_user**
  - Cannot sign in and displays an error message: "Sorry, This user has been locked out."
- **Problem_user**
  - Cannot apply filters and sorting functions on the Products page
  - Has a problem with filling the form before continue to checkout; the last name field can't accept value and when trying to write any text the first letter is assigned to the first name field.
- **Error_user**
  - Cannot apply filters and sorting functions on the Products page and returns an error
    "sorting is broken! This error has been reported to Backtrace".

- o Has a problem with filling the form before continue to checkout; the last name field can't accept value.
- **Performance_glitch_user**
  - o Take mush time to Login or navigate to Products page
- **Visual_user**
  - o Has a UI/UX issues
    - ▪ The side bar icon is not in it's expected style
    - ▪ The cart icon is not in it's expected place
    - ▪ "checkout" button is not in it's expected place

Conclusion

This documentation outlines the structure and flow of the automation project. The happy scenario involves a series of steps from logging in to checking out, and specific user behaviors are noted for testing various edge cases. By automating these tests, we ensure the stability and functionality of the user login and checkout process.