**Step1: Project setup**

1. New directory creation

**mkdir demo\_test\_project**

**cd demo\_test\_project**

**OR**

Clone repository

**git clone <repository-url>**

1. Initializing new git repo

**git init**

1. Virtual environment setup

**python -m venv venv**

**venv\Scripts\activate**

1. Package installation (behave, selenium, behave-x)

**pip install behave selenium behavex**

**Step2: Write Feature File**

Here we will create a ‘.feature’ file (eg. Login.feature) containing gherkins scenarios described in the user story.

I’ve created login.feature file in features folder

**Feature: Login functionality on the Sauce Demo website**

**Introduction:**

This feature describes the behavior of the login functionality on the Sauce Demo website. Users with different account types attempt to log in to the website to access its features.

**Scenario 1: Successful Login**

**Given:** User is on the Demo Login Page

The user navigates to the login page of the Sauce Demo website.

**When:** User fills the account information for account standard\_user

The user enters the account credentials for a standard user into the username and password fields on the login form.

**And:** User clicks the Login Button

The user clicks on the login button to submit the entered credentials.

**Then:** User is redirected to the Demo Main Page

Upon successful login, the user is redirected to the main page of the Sauce Demo website.

**And:** App Logo is displayed

The app logo is displayed on the main page, indicating that the user has successfully logged in.

**Scenario 2: Failed Login**

**Given:** User is on the Demo Login Page

The user navigates to the login page of the Sauce Demo website.

**When:** User fills the account information for account locked\_out\_user

The user enters the account credentials for a locked-out user into the username and password fields on the login form.

**And:** User clicks the Login Button

The user clicks on the login button to submit the entered credentials.

**Then:** Error Message is displayed

Upon failed login attempt, an error message containing the text "Sorry, this user has been banned." is displayed to the user.

**Scenario 3: Order a Product**

**Given:** User is on the inventory page

The user navigates to the inventory page of the Sauce Demo website.

**When:** User sorts products from low price to high price

The user selects the option to sort the products from low price to high price.

**And:** User adds the lowest priced product to the cart

The user adds the product with the lowest price to the shopping cart.

**And:** User clicks on cart

The user navigates to the shopping cart page by clicking on the cart icon.

**And:** User clicks on checkout

The user proceeds to the checkout process by clicking on the checkout button.

**And:** User enters personal information

The user enters personal information, including first name, last name, and zip code, on the checkout page.

**And:** User clicks Continue button

The user continues the checkout process by clicking on the Continue button.

**Then:** Total amount for the added item is $8.63

The total amount for the added item in the checkout overview page is verified to be $8.63.

**When:** User clicks Finish button

The user completes the checkout process by clicking on the Finish button.

**Then:** Thank You header is shown in Checkout Complete page

Upon successful completion of the checkout process, a "Thank You" header is displayed on the checkout complete page.

**Step3: Step definitions implementation**

Here we will implement step definitions for the steps mentioned in our feature file. These step definitions will contain the selenium automation code with behave.

We will also store screenshots of particular steps in screenshots folder.

I’ve created step\_definitions.py file inside /features/Steps folder

**Step Definitions: Automation Logic for Login Functionality on the Sauce Demo Website**

**Introduction:**

The step definitions in this file provide the automation logic for the login functionality on the Sauce Demo website. These step definitions are executed as part of Behave scenarios defined in the feature files.

**Scenario 1: Successful Login**

1. **Given** I am on the Demo Login Page
   * This step definition visits the login page of the Sauce Demo website.
2. **When** I fill the account information for account standard\_user into the Username field and the Password field
   * This step definition enters the credentials for a standard user into the username and password fields on the login form.
3. **And** I click the Login Button
   * This step definition clicks on the login button to submit the entered credentials.
4. **Then** I am redirected to the Demo Main Page
   * This step definition verifies that the user is redirected to the main page of the Sauce Demo website upon successful login.
5. **And** I verify the App Logo exists
   * This step definition verifies the presence of the app logo on the main page, confirming successful login.

**Scenario 2: Failed Login**

1. **Given** I am on the Sauce Demo Login Page
   * This step definition visits the login page of the Sauce Demo website.
2. **When** I fill the account information for account locked\_out\_user into the Username field and the Password field
   * This step definition enters the credentials for a locked-out user into the username and password fields on the login form.
3. **And** I click the Login Button
   * This step definition clicks on the login button to submit the entered credentials.
4. **Then** I verify the Error Message contains the text "Sorry, this user has been banned."
   * This step definition verifies that an error message containing the specified text is displayed upon failed login attempt.

**Scenario 3: Order a Product**

1. **Given** I am on the inventory page
   * This step definition visits the inventory page of the Sauce Demo website.
2. **When** user sorts products from low price to high price
   * This step definition selects the option to sort products from low price to high price.
3. **And** user adds lowest priced product
   * This step definition adds the lowest priced product to the shopping cart.
4. **And** user clicks on cart
   * This step definition navigates to the shopping cart page by clicking on the cart icon.
5. **And** user clicks on checkout
   * This step definition proceeds to the checkout process by clicking on the checkout button.
6. **And** user enters first name John
   * This step definition enters the first name "John" in the checkout form.
7. **And** user enters last name Doe
   * This step definition enters the last name "Doe" in the checkout form.
8. **And** user enters zip code 123
   * This step definition enters the zip code "123" in the checkout form.
9. **And** user clicks Continue button
   * This step definition continues the checkout process by clicking on the Continue button.
10. **Then** I verify in Checkout overview page if the total amount for the added item is $8.63
    * This step definition verifies that the total amount for the added item in the checkout overview page is $8.63.
11. **When** user clicks Finish button
    * This step definition completes the checkout process by clicking on the Finish button.
12. **Then** Thank You header is shown in Checkout Complete page
    * This step definition verifies that a "Thank You" header is displayed on the checkout complete page upon successful checkout.

**Step4: Write Automation Code**

Here we will write the automation code using selenium to interact with the website elements and perform actions described in the scenario.

I’ve created automation.py file inside main directory demo\_test\_project

**Automation Code: Implementation of Behave Step Definitions for Login Functionality on the Sauce Demo Website**

**Introduction:**

The automation code in this file consists of Behave step definitions that implement the logic for automating the login functionality on the Sauce Demo website. These step definitions are executed as part of Behave scenarios defined in feature files.

**Dependencies:**

* Python: 3.6 or higher
* Selenium WebDriver
* Behave framework

**Step Definitions:**

**Scenario 1: Successful Login**

1. **visit\_login\_page(context)**:
   * Navigates to the login page of the Sauce Demo website.
2. **fill\_standard\_user\_credentials(context)**:
   * Enters the credentials for a standard\_user into the username and password fields on the login form.
3. **click\_login\_button(context)**:
   * Clicks on the login button to submit the entered credentials.
4. **verify\_redirected\_to\_main\_page(context)**:
   * Verifies that the user is redirected to the main page of the Sauce Demo website upon successful login.
   * Also verifies the presence of the app logo on the main page.

**Scenario 2: Failed Login**

1. **visit\_login\_page(context)**:
   * Navigates to the login page of the Sauce Demo website.
2. **fill\_locked\_out\_user\_credentials(context)**:
   * Enters the credentials for a locked\_out\_user into the username and password fields on the login form.
3. **verify\_error\_message(context)**:
   * Verifies that an error message containing the specified text is displayed upon failed login attempt.

**Scenario 3: Order a Product**

1. **visit\_inventory\_page(context)**:
   * Logs in as a standard user.
   * Navigates to the inventory page of the Sauce Demo website.
2. **sort\_products\_low\_to\_high(context)**:
   * Selects the option to sort products from low price to high price.
3. **add\_lowest\_priced\_product(context)**:
   * Adds the lowest priced product to the shopping cart.
4. **click\_cart(context)**:
   * Navigates to the shopping cart page by clicking on the cart icon.
5. **click\_checkout(context)**:
   * Proceeds to the checkout process by clicking on the checkout button.
6. **enter\_personal\_info(context)**:
   * Enters personal information (first name, last name, zip code) on the checkout page.
7. **click\_continue(context)**:
   * Continues the checkout process by clicking on the Continue button.
8. **verify\_total\_amt(context)**:
   * Verifies that the total amount for the added item in the checkout overview page is $8.63.
9. **click\_finish(context)**:
   * Completes the checkout process by clicking on the Finish button.
10. **thankyou\_header(context)**:
    * Verifies that a "Thank You" header is displayed on the checkout complete page upon successful checkout.

**Step5: Create Batch File**

Here we will create a batch file (e.g runtest.bat) to run our test.

I’ve created batch file inside main directory too.

**@echo off**

**rem Activate the virtual environment (if you are using one)**

**call venv\Scripts\activate**

**rem Run Behave tests with specified options**

**behave -f pretty -o test-results**

**rem Deactivate the virtual environment (if activated)**

**deactivate**

**Step6: Generate BehaveX Report**

BehaveX generates reports automatically. By using **behave -f Pretty -o test-results** in the batch file, it will generate the BehaveX report in the test-results directory.

Report is getting generated inside our main directory too.

**Step6: Run the code**

**\*To run acceptance criteria behave test just double click the batch file\* OR \*In cmd after setting up virtual environment run the behave –f Pretty –o test-results command\***

**Steps to setup virtual environment through CMD–**

1. Go to directory
2. Open cmd in that directory
3. Activate virtual environment

python –m venv venv

venv\Scrips\activate

1. Run the code

behave –f Pretty –o test-results

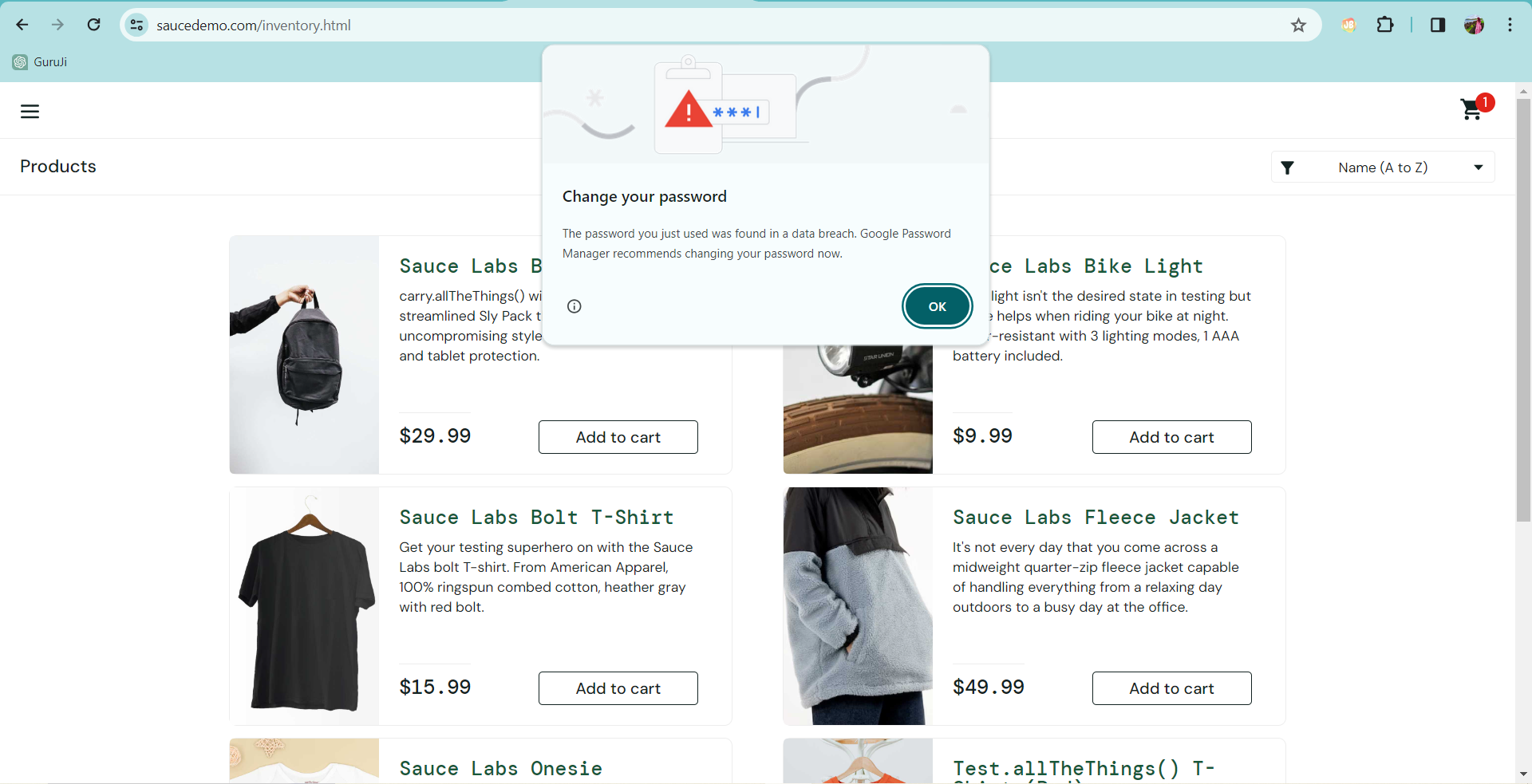
1. After successful test results deactivate the virtual environment

Deactivate

**Issues I encountered:**

**Scenario 1:**

1. After successful log in an alert was appearing in chrome not in edge

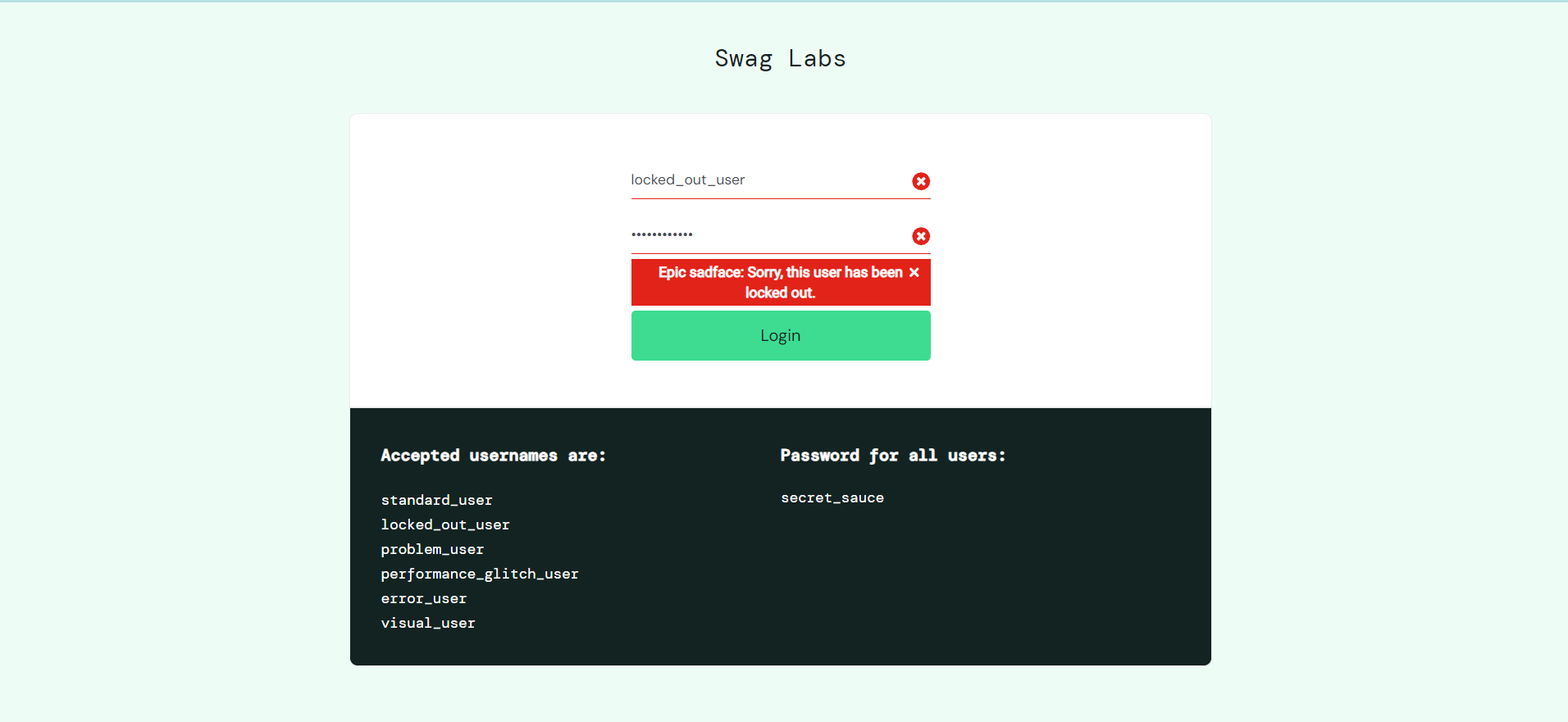


* So I’ve handled it in my code in try except block

**Scenario 2:**

1. In Scenario 2

The error message containing text is “Epic sadface: Sorry, this user has been locked out.” not “Sorry, this user has been banned.”



* I’ve handled it too.