Test Report: Asana Application Of Automation

Test Environment:

- Programming Language: Python

- Testing Framework: unittest

- Web Automation Library: Selenium

- Additional Libraries: BeautifulSoup (for HTML parsing)

- Web Browser: Google Chrome

- Operating System: [Specify the OS]

Executive Summary:

This document provides test documentation for a Python script that automates testing of Asana's key features using Selenium WebDriver. The script includes test scenarios for login, creating tasks, checking tasks, creating projects, and deleting tasks. The tests are organized into a test suite using the unittest framework.

Analyzing Asana's Key Features:

The testing script focuses on the following key features of Asana:

1. **Login**: Testing user login functionality.

2. **Create Task**: Testing the ability to create a new task.

3. **Check Task**: Testing the ability to search for and verify the presence of a task.

4. **Create Project**: Testing the creation of a new project.

5. **Delete Task**: Testing the ability to delete a task.

Identifying Critical Test Scenarios:

The following critical test scenarios have been identified:

Login

- Successful login with valid credentials.

- Failed login with invalid credentials.

Create Task

- Creating a task with a valid name.

- Creating a task with a null name.

Check Task

- Checking the presence of an existing task.

- Checking the absence of a non-existing task.

Create Project

- Creating a project with a valid name.

- Creating a project with a null name.

Delete Task

- Deleting an existing task.

Test Objectives:

1. Verify the login functionality.

2. Test task creation functionality.

3. Test task presence/absence checking functionality.

4. Test project creation functionality.

5. Test task deletion functionality.

Test Execution:

The testing script is implemented as a set of test methods within a Python class. The script uses the Selenium WebDriver to interact with the Asana web application. The test methods and their respective scenarios are as follows:

test_1_successful_login

- Scenario 1: Successful login with valid credentials.

- Scenario 2: Failed login with invalid credentials.

test_2_create_task

- Scenario 1: Creating a task with a valid name.

- Scenario 2: Creating a task with a null name.

test_3_check_task

- Scenario 1: Checking the presence of an existing task.

- Scenario 2: Checking the absence of a non-existing task.

test_4_create_project

- Scenario 1: Creating a project with a valid name.

- Scenario 2: Creating a project with a null name.

test_5_delete_task

- Scenario 1: Deleting an existing task.

The test cases are executed in the specified order.

Test Results:

Test results will be recorded in a log file named "test_report.log." The results will include a timestamp, test case name, test case status (Passed or Failed), and an optional error message in case of failure.

Test Analysis:

After executing the test suite, review the "test_report.log" file for the results of each test case. Analyze the results to identify any failures or issues. If a test case fails, the log will contain information about the failure, including error messages.

Note: Please ensure that you have the necessary Selenium and BeautifulSoup libraries installed, and a Chrome WebDriver executable configured for this script to run successfully.

Test Data:

- Test login credentials: "advkn" and "roo34" (invalid), "ayansameer60@gmail.com" and "root1234" (valid).

- Task name for creating tasks: "Coding" (valid), None (null).

- Task name for checking tasks: "Coding" (valid), "ZCACW" (non-existing).

- Project name for creating projects: "TogoRide" (valid), None (null).

Additional Instructions:

1. Make sure you have the necessary libraries installed and Chrome WebDriver configured.

2. Update the test data with valid credentials and names for testing as needed.

3. Execute the script in the specified order.

4. Review the "test_report.log" file for test results and potential issues.

5. Analyze the test results and any failure messages for further investigation if needed.