**Task 2**

**Creation of Test Plan document for the Sauce Labs demo application**

**1. Introduction**

**1.1 Purpose**

The purpose of this Software Test Plan (STP) is to outline the approach, scope, resources, and schedule for testing the Sauce Labs demo application software system. This plan ensures that the product meets the specified requirements and is free of defects before release.

**1.2 Scope**

This test plan covers all aspects of testing for the Sauce Labs demo project, including functional testing, performance testing, security testing, and user acceptance testing (UAT). It specifies the test objectives, testing strategy, test environment, and resource requirements.

**1.3 Test Objectives**

* Verify core functionalities: product browsing, cart management, checkout process, and payment gateway integration.
* Ensure compatibility across devices, browsers, and operating systems.
* Validate security measures like data encryption and user authentication.
* Test performance under various loads.

**2. Test Items**

This section lists the software modules, components, or features that will be tested. It includes:

* **Module A**: User registration and login functionality
* **Module B**: Product search, filtering, add to cart, remove from cart and Wishlist functionalities
* **Module C**: Payment gateway integrations (Credit/debit cards, UPI, wallets), Order confirmation

**3. Test Approach**

**3.1 Test Types**

The following types of testing will be performed:

* **Functional Testing**: Verifying that the software performs the functions as intended.
* **UI Testing**: Ensure the interface is user-friendly and visually appealing
* **Regression Testing**: Ensuring new code changes do not negatively impact existing functionality.
* **Performance Testing**: Assessing system performance under various conditions like website speed, load time and scalability.
* **Security Testing**: Verifying that security features like authentication, authorization work as expected.
* **Compatibility Testing**: Verifying compatibility with different browsers, operating systems, and devices.

**3.2 Test Levels**

* **Unit Testing**: Individual components or units of code are tested.
* **Integration Testing**: Testing the interaction between integrated modules.
* **System Testing**: End-to-end testing of the complete software system.
* **User Acceptance Testing (UAT)**: Testing performed by the end-users to validate the software meets their needs.

**4. Test Deliverables**

* **Test Plan Document**: This document.
* **Test Cases**: Detailed test cases based on requirements and specifications.
* **Test Scripts**: Automated scripts for regression and performance testing.
* **Test Execution Reports**: Reports on the results of test executions.
* **Defect Reports**: Documentation of defects found during testing.
* **Test Summary Report**: Final report summarizing test activities and outcomes.

**5. Testing Schedule**

This section provides a detailed timeline of the testing activities. Include the following:

* **Test Preparation Phase**: (Oct 2, 2025 – Oct 5, 2025)
* **Test Execution Phase**: (Oct 6, 2025 – Oct 10, 2025)
* **Test Closure Phase**: (Oct 11, 2025 – Oct 12, 2025)

**6. Test Environment**

**6.1 Hardware Requirements**

* CPU, RAM, Disk Space, Network Configuration.

**6.2 Software Requirements**

* Operating System versions.
* Databases and Web Servers used.
* Test Automation Tools (e.g., Selenium, JUnit, TestNG).

**6.3 Test Environment Setup**

* Instructions for setting up the test environment, including configurations and network details.

**7. Test Resources**

**7.1 Personnel**

* **Test Manager**: Responsible for managing testing activities and ensures timelines are met.
* **Test Engineers**: Responsible for executing test cases, logging bugs and retesting.
* **Automation Engineers**: Responsible for creating and maintaining automated test scripts and generating reports of the executed test scripts.

**7.2 Tools**

* Test Management Tool: JIRA, ALM
* Test Automation Tools: Selenium
* Performance Testing Tools: JMeter, LoadRunner
* Version Control Systems: Git

**8. Risks and Mitigation**

**8.1 Identified Risks**

* **Delayed Release**: Impacting testing timelines.
* **Environment Setup Issues**: Delay in the setup of the test environment.
* **Incomplete Requirements**: Potential for incomplete or unclear requirements leading to incomplete tests.

**8.2 Mitigation Strategies**

* Develop contingency plans for environment issues.
* Regularly review requirements with stakeholders to ensure completeness.
* Perform early testing to identify delays early.

**9. Test Strategy**

**9.1 Test Execution**

Tests will be executed manually for functional and usability testing, regression and performance testing. Automated tests will be integrated into the CI/CD pipeline.

**9.2 Test Reporting**

Defects will be reported in the defect management tool (e.g., JIRA), and test execution results will be shared via the test management tool.

**9.3 Entry and Exit Criteria**

**Entry Criteria**:

* + Test environment is ready.
  + Test cases are reviewed and approved.
  + Development is complete.

**Exit Criteria**:

* + All critical defects are resolved
  + Test execution is 100% complete.
  + QA is signed off