SAUCEDEMO LIMITED

Saucedemo.com

TEST STRATEGY

Document Version:	v0.1	Date:	07 Sep 2024

Document Change History

Version	Change	Author	Date
0.1	Initial Draft	Shanmuga Priya Alagarsamy	07 Sep 2024

Approvals & Distribution

Approvers:			
Name	**Position**	**Department**	**Approved Date**
	IT Project Manager	IT	
	Business Project Manager	IT	
	Development Manager	IT Development	
	Test Manager	IT QA	
Distribution:			
	Solution Architect	IT	Distribution only
	Test Analyst	IT QA	Distribution only
	Tester	IT QA	Distribution only
	Development Lead	IT Development	Distribution only

Table of Contents

1 Introduction
1.1 Document Purpose
1.2 Related Documents
2 Strategy Overview
2.1 Scope
2.2 Test Objectives
3 Test Approach
3.1 Test Phases
3.2 Unit Testing
3.3 System Testing
3.4 System Integration Testing
3.5 Performance Testing
3.6 User Acceptance Testing
4 Test Automation
4.1 Progressive Automation
4.2 Regression Automation
5 Risks
6 Test Environments & Data
6.1 Test Environments
6.2 Test Tools
6.3 Test Data
6.4 Data Constraints
7 Testing Management
7.1 Test Timeline
7.2 Resource Requirements
7.3 Test-Related Meetings
7.4 Defect Management Process
8 QA Deliverables
9 Signatures/Responses

1 INTRODUCTION

1.1 Document Purpose

The purpose of this document is to define the Test Strategy for project **saucedemo.com**. It covers the activities applicable for each test phase of the project.

The document outlines the following points at a high level:

- project/test timelines, actions and deliverables
- required test phases for the project and the roles, responsibilities and resources for each phase
- test scope, exclusions dependencies and assumptions
- test environments to be used for each test phase
- identified risks, issues and mitigation actions

If changes to this strategy are required they will need to be agreed to and signed off by those identified as signatories on this document. Any changes will need to be change tracked within the document.

1.2 Related Documents

This document draws upon the following related project documents.

Ref	Document	Version	Signoff / Status
1.	Business Requirements	1.0	Approved
2.	Solution Concept	1.0	Approved
3.	High level Design	1.0	Review
4.	Detailed Design	1.0	Review

2 STRATEGY OVERVIEW

2.1 Scope

The testing scope includes the following features:

- Login and Logout
- Product Search
- Filtering
- Product listing page
- Product details page
- Add to Cart and Remove from Cart
- Checkout
- Order Confirmation

2.2 Test Objectives

The following outlines the high-level objectives of testing:

- Verify that all features function as expected according to the specified requirements.
- Ensure the website performs optimally under various load conditions and meets all the non functional requirements.
- Test for vulnerabilities to protect user data and prevent unauthorized access.
- Build progressive test automation suit to ensure early defect detection and shift-left in SDLC

3 Test Approach

3.1 Test Phases

The test phases required to be executed for this project, as per the Quality Assurance Test Methodology, are:

- Unit Test
- System Test
- System Integration Test
- Performance Test
- User Acceptance Test

3.2 Unit Testing

Owner of Test Phase	Development Manager
Teams executing tests	Development Team

Unit Test will be conducted by the development team who will determine the test scripts and the results should be shared with the QA team.

3.2.1 Entry Criteria

The entry criteria for this test phase are:

Ref	Item	Owner
1.	Business Requirements – signed off	Project Manager
2.	Detailed Solution Design – signed off	Project Manager
3.	Functional Specifications – signed off	Project Manager
4.	Unit Test Plan document – signed off	Development Lead
5.	DEV environment established with relevant data	Development Lead
6.	Code modification completed and migrated to the DEV environment	Development Lead

3.2.2 Management and Execution

This table defines the tasks required to achieve a successful completion of this phase of the test.

Ref	Task	Owner
1.	Coordinate and manage progression of Unit Test phase	Development Lead
2.	Engage test team for knowledge transfer	Development Lead
3.	Manage Change Control including code migration, documentation, and implementation plan/s	Development Lead
4.	Execute unit test scripts	Developers
5.	Raise defects	Developers
6.	Manage escalation and resolution of defects	Development Lead
7.	Sign-off Unit Test Phase as completed	Development Lead

3.2.3 Exit Criteria

The exit criteria for this test phase are:

Ref	Item	Owner
1.	Unit testing of all high and medium risk areas completed	Development Lead
2.	Unit Test Summary report, including list of tests and outcomes	Development Lead
3.	No outstanding major issues (Sev 1 or 2)	Development Lead

3.3 System Testing

The purpose of System Testing is to evaluate the different modules of the website like login, product search, filtering & cart ensuring it meets the specified requirements and functions as intended.

Owner of Test Phase	Test Manager
Teams executing tests	IT Quality Assurance

3.3.1 Entry Criteria

The entry criteria for this test phase are:

Ref	Item	Owner
1.	Unit Test Exit Criteria – completed	Development Lead
2.	System Test Strategy / Plan – signed off	Test Lead
3.	System Test scripts are documented	Test Lead
4.	SYSTEM TEST environment established	Development Lead
5.	Code migrated to the SYSTEM TEST environment	Development Lead
6.	SYSTEST TEST data set up	Test Lead

3.3.2 Management and Execution

This table defines the tasks required to achieve a successful completion of this phase of the test.

Ref	Task	Owner
1.	Coordinate and manage progress of System Test phase	Test Lead
2.	Execute system test scripts and raise defects	Quality Assurance team
3.	Manage escalation and resolution of defects	Test Lead
4.	Sign-off System Test Phase as completed	Test Lead

3.3.3 Exit Criteria

The exit criteria for this test phase are:

Ref	Item	Owner
1.	System test of all high and medium risk areas completed	Test Lead
2.	No outstanding major issues (Sev 1 or 2)	Test Lead
3.	System Test Completion report	Test Lead

3.4 System Integration Testing

The purpose of System Integration Testing is

- To verify that different components of the application interact correctly with each other.
- To verify Integration with payment systems
- To verify product detail API

Owner of Test Phase	Test Manager
Teams executing tests	IT Quality Assurance

3.4.1 Entry Criteria

The entry criteria for this test phase are:

Ref	Item	Owner
7.	System Test Exit Criteria – completed	Development Lead
8.	Scope for SIT signed off by system owners	Test Lead
9.	System Integration Test scripts are documented	Test Lead
10.	System integration environment established	Development Lead
11.	Code migrated to the SIT environment	Development Lead
12.	SIT data set up	Test Lead

3.4.2 Management and Execution

This table defines the tasks required to achieve a successful completion of this phase of the test.

Ref	Task	Owner
5.	Coordinate and manage progress of SIT Test phase	Test Lead
6.	Liaise with QA Offshore Team	OnsiteCoordinator
7.	Execute SIT test scripts and raise defects	Quality Assurance team
8.	Manage escalation and resolution of defects	Test Lead
9.	Sign-off System Test Phase as completed	Test Lead

3.4.3 Exit Criteria

The exit criteria for this test phase are:

Ref	Item	Owner
4.	System test of all high and medium risk areas completed	Test Lead
5.	No outstanding major issues (Sev 1 or 2)	Test Lead
6.	System Test Completion report	Test Lead

3.5 Performance Testing

Owner of Test Phase	Test Manager
Teams executing tests	IT Quality Assurance

The purpose of this phase is to ensure the solution meets the performance requirements and SLAs.

The Performance Test will be executed in parallel with the User Acceptance Test, on the <environmentName> environment which is comparable to the production environment, and will be populated with PROD live data.

Refer to the Performance Test Plan for this project for details of the inclusions, exclusions, scenarios, transactions and iterations planned.

3.5.1 Entry Criteria

The entry criteria for this test phase are:

Ref	Item	Owner
1.	System Test completed to 85%	Test Lead
2.	Performance Test Plan– signed off	Test Manager
3.	Performance test environment established	Development Lead
4.	Code migrated to the performance test environment	Development Lead
5.	Data created in the performance test environment	Performance Tester

3.5.2 Management and Execution

This table defines the tasks required to achieve a successful completion of this phase of the test.

Ref	Task	Owner
1.	Coordinate and manage progress of Performance Test phase	Test Manager
2.	Execute Performance Tests, report results and issues	Performance Tester
3.	Investigate issues, performance tuning	Development Manager
4.	Manage escalation and resolution of defects	Test Manager
5.	Sign-off Performance Test Phase as completed	Test Manager

3.5.3 Exit Criteria

The exit criteria for this test phase are:

Ref	Item	Owner
1.	Performance test completed with results acceptable to business	Test Manager
2.	No outstanding major issues (Sev 1 or 2)	Test Manager
3.	Performance Test Report – signed off	Test Manager

3.6 User Acceptance Testing

Owner of Test Phase	Business Manager
Teams executing tests	Business Team

Business team will be engaged to perform User Acceptance Testing for the business requirements.

3.6.1 Entry Criteria

The entry criteria for this test phase are:

Ref	Item	Owner

1.	System Test Exit Criteria – completed	Test Lead
2.	UAT Test Plan – signed off	Test Lead
3.	UAT scripts are documented	Test Lead
4.	UAT environment established	Development Lead
5.	Code migrated to the UAT environment	Development Lead
6.	UAT data set up	Test Lead

3.6.2 Management and Execution

This table defines the tasks required to achieve a successful completion of this phase of the test.

Ref	Task	Owner
6.	Coordinate and manage progress of UAT phase	UAT Lead
7.	Execute test scripts & raise defects	Business Manager
8.	Manage escalation and resolution of defects	UAT Lead
9.	Sign-off UAT Phase as completed	UAT Lead

3.6.3 Exit Criteria

The exit criteria for this test phase are:

Ref	Item	Owner
1.	UAT of all high and medium risk areas completed	UAT Lead
2.	No outstanding major issues (Sev 1 or 2)	UAT Lead
3.	UAT Completion report	UAT Lead

4 Test Automation

Selenium Test Automation framework will be used for both progressive and regressive automation. Selenium framework is chosen considering below reasons:

1. **Cross-browser compatibility:** Selenium supports a wide range of web browsers, including Chrome, Firefox, Edge, Safari, and Internet Explorer. This makes it ideal for testing web applications across different platforms and devices.
2. **Open-source and free:** Selenium is an open-source project, which means it's free to use and distribute. This lowers the barrier to entry for organizations and individuals who want to automate their web testing.
3. **Powerful API and ecosystem:** Selenium provides a rich API and ecosystem of tools and libraries, making it easy to create and maintain automated test scripts. Popular languages like Java, Python, C#, and JavaScript can be used with Selenium.
4. **Community support:** Selenium has a large and active community of developers and users, which means there are plenty of resources, tutorials, and forums available to help you get started and troubleshoot issues.
5. **Flexibility:** Selenium is highly flexible and can be used for a variety of testing scenarios, including functional testing, regression testing, and performance testing.
6. **Integration with other tools:** Selenium can be easily integrated with other testing tools and frameworks, such as TestNG, JUnit, and Jenkins, to create comprehensive testing pipelines.
7. **Continuous integration and continuous delivery (CI/CD):** Selenium is well-suited for integrating into CI/CD pipelines, allowing automated testing to be run as part of the development process.
8. **Scalability:** Selenium can be scaled to handle large-scale testing projects, making it suitable for organizations with complex web applications.

9. **Platform independence:** Selenium can be used on various operating systems, including Windows, macOS, and Linux.
10. **Regular updates and improvements:** The Selenium project is actively maintained, with regular updates and improvements being released to address bugs, add new features, and improve performance.

4.1 Progressive Automation

Identify Suitable Test Cases: Select test cases that are well-suited for automation, such as those that are repetitive, time-consuming, or error-prone.

Develop Automation Scripts: Create automated test scripts for the selected test cases, ensuring they are maintainable and reusable.

Execute and Validate: Run the automated test scripts and validate the results against expected outcomes.

Monitor and Maintain: Continuously monitor the performance of automated tests and make necessary adjustments to ensure their effectiveness.

4.2 Regression Automation

Test Case Identification: Identify critical test cases that need to be automated for regression testing.

Test Script Development: Create automated test scripts using the chosen framework, covering the identified test cases.

Test Data Management: Develop strategies for managing test data effectively.

Test Environment Setup: Configure the testing environment with the necessary tools, drivers, and dependencies.

Test Execution and Reporting: Execute the regression test suite and generate detailed reports.

Maintenance: Regularly update and maintain the regression test suite to keep it aligned with changes to the website.

5 Risks

Risks considered major, likely to impact testing, and specific to this project are described below along with mitigation actions.

Ref	Description	Impact	Mitigation Strategy	Owner
1.				
2.				
3.				

6 Test Environments & Data

6.1 Test Environments

The following environments have been identified as suitable for each test phase:

Test Phase / Purpose	SystemA	SystemB	SystemC	SystemD
Unit Test				
System Test				
System Integration Test				
Performance Test				
User Acceptance Test				

6.2 Test Tools

Tool	Purpose	Account / Connectivity
Test Management Tool(Eg: qTest, HP ALM or Jira)	Plan and execute tests, manage defects, produce reports and metrics	Project: Saucedemo.com
Selenium	Test Automation Tool	

6.3 Test Data

Data will be generated by QA team as follows:

Deliverable	Data Set	Test Data Generation Method

6.4 Data Constraints

The following table highlights any data anticipated to adversely impact the progression of system and/or acceptance test:

Ref	Element	Impact
1.		
2.		

7 Testing Management

7.1 Test Timeline

The project plan document includes the dates for test activities and is being maintained under version control in the following location:

<linkToProjectPlan>

7.1.1 Test Execution Timeline

The following table shows the proposed dates for execution of each test phase:

Task Name	Duration	Start Date	Finish Date
Unit Test	**5 days**		
System Test	**16 days**		
Sanity Test	1 day		
System Test – Cycle 1	5 days		
System Test – Cycle 2	5 days		
Regression Test	5 days		
Performance Test	**5 days**		
UAT	**8 days**		

7.1.2 Dependant Provisions

The above timeline is based on agreement to the following provisions:

Ref	Provision	Owner
1.	Development teams provide a maximum of 24 hour (business hours) turnaround time for resolution of every Sev 1 and 2 defect	
2.	**Code to be installed** in the test environment to enable commencement of script execution on the Start Date, as defined above	
3.		
4.		
5.		

7.2 Resource Requirements

Based on the documents to hand and project meetings at the time of writing this document, the following high level resource estimates have been identified:

Role	Tasks	UNIT	SYSTEM	UAT	PROD
Test Manager	Estimate test effort Review Test strategy & Test Plan Defect Management Manage progress of test phase Prepare and submit SLA metrics Manage cost of testing Prepare Test completion report Bring improvements in the team	0.25	0.25	0.25	0.25
Senior Test Analyst	Produce Test Strategy Produce Test Plan Manage entry and exit criteria Manage creation of test script content and	1	1.5	1.5	1

	execution System test data setup technical support of application for test Review test cases				
Quality Assurance Testers	Script tests Execute tests Create defect reports Retest defects		5		
Development Team	Test data setup (DEV) Unit test script execution (DEV) Resolution of defects (SYSTEST) Technical support for performance tuning (PROD)	team	team		team
Infrastructure	Monitoring performance Resolution of performance issues				TBA
Business Testers	Perform UAT				

7.3 Test-Related Meetings

The following test-related meetings are planned for this project, based on the nature and size of the project. The test team will chair the meetings, with the exception of the PWG which is chaired by the project manager.

Meeting Name	Purpose	Attendees	Frequency
Project Working Group (PWG)	Review project progress. Updates from functional leads.	Project manager; leads from biz, development, infra, db, test, support.	Weekly
Test Strategy Workshop	Get input from stakeholders and project team to facilitate approval of the test strategy.	Project manager; leads from biz, development, infra, db, test, support.	Once
Test Scenario Review	Agree test coverage with business and development teams by reviewing test scenarios.	Project manager; leads from biz, development, infra, db, test, support.	Once per release
Test Phase Kickoff	Confirm entry criteria for the test phase are met. Review plan for testing. Discuss concerns.	Test team; Project manager; Development leads; business manager (UAT)	Once per test phase
Defect Triage	Facilitate effective resolution of defects by clarifying issues, reviewing priorities. Facilitate effective project team work by discussing and planning how teams can work effectively together.	Project manager; leads from biz, development, infra, db, test, support.	Daily during execution
Test Phase Completion	(May be combined with Kick Off meeting for the next test release) Review the test phase exit criteria with the test status. Plan remedial actions as needed.	Project manager; leads from biz, development, infra, db, test, support.	Once per test phase
Test Completion	Record the lessons learnt during the project testing.		Once

7.4 Defect Management Process

Defect reports will be created in the test management tool to track issues detected during testing through to resolution. The issues are not necessarily system faults and may be resolved by clarification. Issues typically include problems in requirements, specifications, code, configuration, packaging, data and tests.

Defect workflow can be tailored to meet project requirements.

Defect's Priority and Severity will be maintained for each defect.

Defect Priority states the order in which a defect should be fixed. Higher the priority the sooner the

defect should be resolved.

1. **Priority 1(High) -** The defect must be resolved as soon as possible as it affects the system severely and cannot be used until it is fixed. It is blocking most of the test cases.
2. **Priority 2 (Medium) -** During the normal course of the development activities, defects should be resolved. It can wait until the next drop of the release is created. Not blocking many test cases.
3. **Priority 3 (Low) -** The defect is an irritant but fix can be done once the more serious defect has been fixed. Not blocking or Blocking very few test cases

Defect Severity will be identified based on the below guidelines.

Severity	Description
Severity 1 (Critical)	Entire system or key business process is unusable or does not meet the needs of the business, many users affected and no work-around is available; or, Corruption or loss of data occurs that is not immediately recoverable and prevents the business from continuing.
Severity 2 (High)	Part of the system or key business process is unusable or does not meet the needs of the business, few users affected but a work-around is available; or, Corruption or loss of data occurs that is immediately recoverable and allows the business to continue.
Severity 3 (Medium)	A non-critical incident, affecting a single user, occurs which affects the ability to provide the best service but there is a workaround.
Severity 4 (Low)	Cosmetic errors, documentation anomalies, requests for information or advice required.

8 QA Deliverables

Below artifacts will be produced by the QA team throughout the project.

1. **Test Plan:** A detailed document outlining the testing activities, resources, and schedule.
2. **Test Cases:** Detailed descriptions of test scenarios, expected results, and actual results.
3. **Test Scripts:** Automated test scripts.
4. **Test Data:** Data used to execute test cases.
5. **Test Reports:** Summary of test activities, defects found, and overall test coverage.
6. **Defect Tracking Reports:** Documentation of defects discovered, their severity, and resolution status.
7. **Regression Test Suite:** A collection of automated test cases designed to verify the stability of the website after changes.

9 SIGNATURES / RESPONSES

End of Document