**Test Strategy Document**

**Project Overview**

Spree Ecommerce is an open source ecommerce framework. Spree has been used by numerous companies from different domains (Fashion, beauty , Health etc..) to

build products like marketplace , ecommerce sites quickly by leveraging the underlying solution , thus enabling them to release their products to market faster.There are stability issues in the current ecommerce framework. So the plan is to identify the issues and bottlenecks and provide a solid ecommerce framework with additional enhanced features so that they enter to new markets/regions.

**Project Objective**

* Improve the quality of core capabilities provided by Spree E-commerce framework which includes building a defect free solution.
* Understand and Address the current customers pain points/areas in-turn attain the customer satisfaction and improve the sales by retaining the existing customer base.
* Support in delivering new functionalities in a faster manner without impacting the core/existing framework functionalities.
* Enhance the solutions to meet the local needs of the markets.
* Provide a fully integrated omni-channel capability to the framework to enrich the customer experience.

**Test Objective**

Provide quality and stability to the Framework as a whole so that the functionality meets the business requirements.

1. Analyse all the existing capabilities of the current framework.
2. Analyse the area of the newly introduced bugs affecting the customers pain points and prioritize it based on criticality.
3. Analyse the New functionality proposed by Business and assess the impact on the existing functionalities. Develop a solution free of defects through functional and regression testing strategy.
4. Understand the key regions or markets they are focussing and its localisation needs.
5. Devise a testing plan to test the Omni-channel capability.
6. Understand the non functional aspects and devise a plan to address the need.
7. Track/Examine the existing security measures and suggest if any Vulnerability assessment is required.
8. Examine the existing Test Automation framework an assess if any gaps are there.
9. Understand the compliance needs and adhere to the framework standards.

**Test Approach**

Following are the types of testing that will be performed for this project.

1. Unit Testing

Unit testing and Unit Integration testing will be performed by development team before deploying the Code in Quality Test Env.

1. Smoke/Sanity Testing

To Ensure that the build is stable enough to proceed further with detailed testing.

1. Functional Testing

To verify and ensure that the framework is as per the specifications intended for it as well as the functional requirements mentioned in developmental documentation.

1. Integration/API Testing

Verify and validate that all the interface requirements work correctly. The tests will be conducted from end to end perspective.

1. Regression Testing

Regression suite will be executed at the end of the last code drop in UAT environment to ensure release stability.

1. Performance/NFT Testing

Performance testing, a non-functional testing technique will be performed to determine the system parameters in terms of responsiveness and stability under various workloads.

1. Acceptance Testing

Verifies the end to end flow of the system is as per the requirement and all the features and functionalities work as expected.

1. Production Validation Testing

Smoke testing will be performed on production environment after code is deployed.

1. Business Validation/Beta Testing

Business users will perform the high level testing in real environment before releasing to the market for the actual end users.

1. Browser/Device Compatibility Testing

Validates whether the application runs on all versions of different browsers and different devices

1. Security Testing

To check whether the framework is secure from internal or external threats.

1. Automation Testing

Run the automated tests in a continuous fashion as part of the CI/CD pipeline and share the quick feedback. The test results should be consistent and reliable.

**Testing Strategy**

**Unit Testing**

Objective and Scope

Unit testing and Unit Integration testing will be performed by development

team before deploying the Code in QA Env.

Entry Criteria

* Coding is completed for stories.

Actions Required

* Development Team to execute unit test cases and upload against stories in JIRA
* QA to review unit test results and report in cases of gap identified.
* QA team to perform high level testing in development environment for few initial sprints where unit test results are not documented.

Exit Criteria

* All unit test cases are executed and passed ,results are uploaded in JIRA stories.

**Smoke Testing**

Objective and Scope

Testing the critical business functions after every code drop to ensure that build is good enough for functional /Integration testing.

Entry Criteria

Code drop to test environment is completed ,communication via release note/JIRA update is done.

Exit Criteria

All smoke test cases have passed.

**Functional & Integration /API Testing**

Scope

This will include testing all requirements mentioned within the JIRA stories

Objective

Testing the business functions,business rules and business validations with respect to the application.

New functionality integrated with existing one needs to be tested.

Identify defects that might arise while testing.

Report these failure to the development team so that the identified defects are fixed.

Help determine the extent to which the application is ready for release.

Provide input to the defect trend analysis effort.

Exit Criteria

All functionalities work as expected

**Regression Testing**

Scope

Tests covering end to end functionalities will be executed at the end of the functional and integration testing of all user stories.

The first category of the regression test will include high priority tests for commonly expected functionality.

The second category of tests will include any functional defects that are found and fixed.

Objective

The objective of this test cycle are to ensure that the new software or defect fixes does not cause problems with existing software.

Entry Criteria

Functional and Integration Testing is completed

All priority 1 & 2 functional and integration defects are fixed and closed.

Exit Criteria

Publish regression test results to stakeholders.

Regression Test Suite executed in QC/JIRA.

**Acceptance Testing**

Entry Criteria

Business Requirement must be fully developed and available.

Unit,Integration,Functional and Regression Testing should be completed

Without any showstopper,high,medium defects.

Exit Criteria

No critical defects should be open.

Business process and functionality work satisfactorily.

**Performance Testing(Non-Functional Testing)**

Entry Criteria

Non functional requirements are documented in JIRA and approved.

Benchmarks for performance testing i.e decision will made on User Load with which SDFC will be hit.

Exit Criteria

Publish performance test results to stakeholders

**Production Validation Testing**

Entry Criteria

The code should be deployed to production environment and technical validation by development team should have been completed.

QA Action Required

QA will perform smoke test in production environment to ensure the stories are deployed correctly

Exit Criteria

Publish test results so that Business Validation testing can begin.

**Business Validation Testing**

Entry Criteria

Production Validation testing should have been completed by QA team and all identified defects are fixed.

QA Action Required

Business users will perform required testing to ensure user stories which are deployed to production environment meets acceptance criteria.

Exit Criteria

Test Results are published all identified defects are fixed.

**Browser Compatibility Testing**

Entry Criteria

Functional and Integration Testing is completed

All priority 1 & 2 functional and integration defects are fixed and closed.

Exit Criteria

Test results should show that it is compatible with all browsers and devices.

**Security Testing**

Entry Criteria

Perform the security testing regular interval and when a new functionality is introduced with the functional testing.

Exit Criteria

Find out the hitches/loopholes in system and help developers in fixing the problems.

**Automation Testing**

Entry Criteria

Select the set of test/regression test cases for automation to reduce the manual/regression testing effort.

Exit Criteria

All the possible manual and regression test cases converts to automation scripts.