Jenkins 2.401.3

TEST PLAN

Abstract

This test plan describes the testing approach and overall framework that will drive the testing of the Jenkins 2.401.1 open-source automation server.

Version 1.0
07/03/2023

Alla Anderson

Test Plan Contents:

1. INTRODUCTION

Customers want a perfect application, which passes the full cycle of manual and automation testing. Given the specificity of the app it is important to have the same quality of the app as per requirements.

2. SCOPE

The document mainly targets the GUI testing, positive and negative testing, and validating data in report output as per Requirements Specifications provided by Client.

3. QUALITY OBJECTIVES

A primary objective of testing is to: assure that the system meets the full requirements, including quality requirements (functional and non-functional requirements) and fit metrics for each quality requirement and satisfies the use case scenarios and maintain the quality of the product.

The secondary objectives of testing will be to identify and expose all issues and associated risks, communicate all known issues to the project team, and ensure that all issues are addressed in an appropriate matter before release.

4. TEST APPROACH

Analytical test approach was used, in accordance to requirements-based strategy, where an analysis of the requirements' specification form is the basis for planning, estimating and designing tests. Test cases will be created during exploratory testing. All test types are determined in Test Strategy.

The project is using an agile approach, with weekly iterations. At the end of each week the requirements identified for that iteration will be delivered to the team and will be tested.

5. ENTRY AND EXIT CRITERIA

Entry Criteria:
- Proper test data should be available.
- All the necessary documentation, design, and requirements information should be available that will allow testers to operate the system and judge the correct behavior.
- All the standard software tools including the testing tools must have been successfully installed and functioning properly.
- All test hardware platforms must have been successfully installed, configured, and functioning properly.

Exit Criteria:
- No high priority or severe bugs are left outstanding.
- A certain level of requirements coverage has been achieved.
- All high-risk areas have been fully tested, with only minor residual risks left outstanding.
- Cost – when the budget has been spent.
- The schedule has been achieved.

6. SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

Suspension criteria:
- Software/Hardware problems.
- Significant change in requirements suggested by client.
- The build contains many serious defects which seriously or limit testing progress.
- Assigned resources are not available when needed by test team.

Resumption criteria:
- Resumption will only occur when the problem that caused the suspension has been resolved.

7. TEST STRATEGY

QA role in test process:
- Understanding Requirements.
- Requirement specifications will be sent by client.

Preparing Test Cases:
- QA will be preparing test cases based on the exploratory testing. This will cover all scenarios for requirements.

Preparing Test Matrix:
- QA will be preparing test matrix which maps test cases to respective requirement. This will ensure the coverage for requirements.

Reviewing test cases and matrix:
- Review for test cases and test matrix will be conducted by QA Lead.
- Any comments or suggestions on test cases and test coverage will be provided by reviewer
- Suggestions or improvements will be updated by preparer and sent to QA Lead for approval.
- Updates and improvements will be reviewed and approved by reviewer.

Creating Test Data:
- Test data will be created by respective QA based on scenarios and Test cases.

Executing Test Cases:
- Test cases will be executed by respective QA based on designed scenarios, test cases and Test data.
- Test result (Actual Result, Pass/Fail) will be updated in test case document Defect

Logging and Reporting:
- QA will be logging the defect/bugs in Excel spreadsheet and JIRA, found during execution of test cases.

Retesting and Regression Testing:
- Retesting for fixed bugs will be done by respective QA once issue is resolved by respective developer and bug/defect status will be updated accordingly. In certain cases, regression testing will be done if required.

Deployment/Delivery:
- Once all bugs/defect reported after complete testing is fixed and no other bugs are found, report will be deployed to the client, along with sample output by email to respective lead and Report group

Bug life cycle:
- All the issues found while testing will be logged into JIRA.

Testing types:
- Exploratory testing – includes a type of software testing where Test cases are not created in advance but QA checks system "hands-on". QA may note down ideas about what to test before test execution
- Black box testing – also called behavioral testing or Partition testing. This kind of testing focuses on the functional requirements of the software. It enables one to derive sets of input conditions that that will fully exercise all functional requirements for a program.
- GUI Testing – GUI testing will include testing of the UI part of report. It covers users Report format, look and feel, error messages, spelling mistakes, GUI guideline violations.
- Positive testing – includes the type of testing that can be performed on the system by providing the valid data as input. It checks whether an application behaves as expected with positive inputs.
- Negative testing – also known as failure testing or error path testing, is a method of testing an application or system that ensures that the plot of the application is according to the requirements and can handle the unwanted input and user behavior. Invalid data is inserted to compare the output against the given input.
- ADHOC testing – includes an informal testing type with an aim to break the system.

Bug Severity and Priority Definition:

Bug Severity and Priority fields are important for categorizing bugs and prioritizing if and when the bugs will be fixed. QA Tester will assign a severity level to all bugs. The QA Test Lead will be responsible to see that a correct severity level is assigned to each bug.

8. RESOURCES AND ENVIRONMENT NEEDS

Testing Tools:
- Process Tool
- Test case creation Microsoft Word, Microsoft Excel, JIRA
- Test case tracking JIRA, Confluence
- Test case execution Manual
- Test case execution Automation

- Test case management Microsoft Excel, JIRA, Confluence, Trello
- Test reporting JIRA
- Check list creating Microsoft Excel, JIRA

Test Environment x Support level 1 (browsers):
1. <u>Device</u>*:* Mac Book Air (M1,2020). *OS:* macOS Monterey Version: 12.4
   <u>Browser</u>*:* Google Chrome. *Version* 103.0.5060.53 (Official Build) (arm64)
2. <u>Device</u>*:* Mac Book Pro (M1,2020). *OS:* macOS Big Sur : 11.04
   <u>Browser</u>*:* Safari. *Version 15.5 (17613.2.7.1.8)*
3. <u>Device</u>*:* Mac Book Air (M1,2020). *OS:* macOS Monterey Version: 12.4
   <u>Browser</u>*:* Firefox. *Version 103.0.1 (64-bit)*
4. <u>Device</u>: iPhone 11 Pro. *Software Version*: 15.6
   <u>Browser</u>: Safari Version 15.6