{projectName} Test Plan

Date	Version	Change Summary	Prepared By
01-01-2024	1.0	Initial draft	Matish Shrestha

Test Plan Identifier

{projectName} test plan will be identified by {projectInitialsOrAbbreviation} 1.0.

References

- Requirement Documents
- Project Plan
- Design Documents
- User Stories and Epics

Introduction

Purpose

The main purpose of this test plan is to help communicate the test approaches with the team. This document comprises testing scope, testing strategies, risk, assumption, and testing schedule.

Project Overview

Give a brief description of the project, problem statement, and solution.

In Scope

- Functionality Testing
    - Functionality testing is a testing where the system is tested against the functional requirements/specifications. Functionality testing will be carried out to determine whether different functionalities are working as specified.
- UI Testing
    - UI testing is the process of testing the visual elements of an application to validate whether they accurately meet the expected performance and functionality. UI testing will be done to verify the actual product with the specified design mockups.
- Usability Testing
    - Usability testing is a technique for measuring how easy and user-friendly a software application is. Testing the project keeping actual users in consideration for ease of use while using the system.
- API Testing
    - API testing is a type of software testing that involves testing application programming interfaces (APIs) directly and as part of integration testing to determine if they meet expectations for functionality, reliability, performance, and security. API testing will be done using Postman so that the API can be checked without the GUI being built. Early verification of the API will be beneficial if bugs are found early.

- Smoke Testing
    - Smoke Testing is a software testing technique performed post-software build to verify that the critical functionalities of the software are working fine. Smoke testing will be performed to ensure that the most important functions work properly.
- Sanity Testing
    - Sanity testing is a subset of regression testing. Sanity testing is performed to ensure that the code changes that are made are working as properly. High-level testing will be conducted to ensure the major functionalities are working as expected. Sanity testing will be conducted per the sanity checklist.

Out Scope

- Unit Testing
    - Unit testing is a software testing technique where individual units of code, typically functions or methods. This will be performed by the Developers to test their codes.
- Security Testing
    - Security testing is a type of software testing that uncovers vulnerabilities, threats, and risks in a software application and prevents malicious attacks from intruders. The DevOps team will be responsible for conducting security testing.

Test Approach

The Scrum methodology of the Agile model will be followed.

Following the Software Testing Life Cycle (STLC) activities will be performed while conducting the tests.

- Requirement Analysis
    - The requirements provided by the stakeholders will be analyzed and documented. A feasibility study of the requirements will also be carried out.

- Creating Test Plan
    - A test plan is created that serves as a blueprint for testing activities. Various types of testing that will be performed on the application are mentioned along with execution strategy.
- Creating Test Cases
    - A test case document is the set of activities that will be performed to verify whether the functionalities are working as expected. Test cases will be prepared based on the user stories and acceptance criteria. Test cases provide full coverage of different aspects of the module under test. Google Sheets will be used to document test cases.
- Test Environment Setup
    - Proper environments will be set up to execute the test cases. Test Environments include hardware and software requirements, different browsers on which application is to be tested, and different environments like Dev.
- Test Execution
    - After test environments are set up, test cases will be executed. Bugs will be reported if a module does not meet its requirements.
- Test Closure
    - Test Reports and other deliverables will be generated for stakeholders, confirming that all incidents have been reported and testing has successfully been performed on the application.

Testing Environment

The following roles are responsible for verifying the features in different environments:

Environment	**Tester**
Dev	Developers, QA, PM, TL
Test/QA	QA, PM, TL

Production	

Tools

The following tools are used during the testing process.

S.N.	Tools	Purpose
1	Postman	API Testing
2	Jmeter	Load Testing
3	Google Sheets	Preparing test cases, bug report
4	Google Docs	Requirement documents, Test Plan

Assumptions

The following assumptions are made to achieve our testing goal and maintain the quality of the product:

- Availability of requirements: Assume complete, accurate, and up-to-date requirements documents exist. Verify this with the product owner or document any discrepancies.
- Access to test environments: Assume necessary hardware, software, and network resources are available throughout the testing cycle. Confirm with IT and document any limitations.
- Team availability and expertise: Assume testers have the required skills and experience for the project. Assess individual skill sets and plan training if needed.

- Test case completeness and coverage: Assume test cases cover all critical functionalities and potential user scenarios. Review test cases thoroughly and address any gaps identified.
- Reliability of testing tools and automation: Assume testing tools and automation scripts function flawlessly. Test and debug these tools before relying on them heavily.
- Clarity of communication between stakeholders: Assume developers, designers, and stakeholders provide clear and timely information. Establish communication channels and protocols to avoid misunderstandings.
- Availability of defect reporting and tracking mechanisms: Assume bug reporting and tracking systems are readily available and functional. Confirm access and train testers on proper usage.

Risks and Mitigation

S.N.	Risks	Impact	Probability	Mitigation plan
1	Misunderstanding between stakeholders, development team, and QA regarding requirements.	High	High	Constant communication between stakeholders, QA, and development team.
2	Delay in testing due to change in requirement in the late stage of the sprint.	High	High	All requirements should be discussed with stakeholders and clear confusion if any.
3	Incorrect estimation of the story points will hamper development and testing.	High	High	Story points must be estimated including the time needed for the code review and testing.

4	Change in team members may lead to increased development time as new members need time to understand the tasks and process.	Low	Medium	Same team members throughout the development process. Onboarding for the new members for quick understanding of the project.
5	Delay in the environment setup.	High	High	Frequent updates and healthy communication between development teams and stakeholders.
6	Inexperienced member of the team.	Low	Low	Training sessions for the basics.

Test Deliverables

- Test Plan Document
    - A test plan is a document describing software testing scope and activities. It is the basis for formally testing any software/product in a project.
- Test Cases
    - Test Case describes the purpose of a specific test, identifies the required inputs and expected results, provides step-by-step procedures for executing the test, and outlines the pass/fail criteria for determining acceptance.
- Bug Report
    - Bug Report is proper documentation of testing results, all the bugs, and issues are documented properly. It contains the module name, issue description, steps to reproduce, and actual and expected results. Jira is used to report bugs found in the project.
- Sanity Checklist

    - The sanity checklist documents the sanity test which is performed to check that all the major functionalities of the project are working fine. It outlines the pass/fail criteria of each functionality.

- Release Notes
    - A release note will be delivered when an update is released. The release note outlines the corrections, changes, or enhancements made to the product.

Schedule

Task	**Schedule**
User Story and Acceptance Criteria	Before starting of sprint
Test Cases	In parallel with the development cycle on a sprint basis
Test Execution	After deployment of each new feature in Dev
Bug Report	During the sprint
Performance Test	During end of the product life cycle
Sanity Test	After Dev release

Clone Order Management Test Plan

Date	Version	Change Summary	Prepared By
07-21-2021	1.0	Initial draft	Simran Shrestha
		API Testing included in Functional	Simran Shrestha

Test Plan Identifier

The Clone Order Management (COM) test plan will be identified by COM 1.0.

References

- User Stories and Acceptance Criteria
- Google Drive Folder for COM

Introduction

Purpose

The main purpose of this test plan is to help communicate the test approaches with the team. This document comprises testing scope, testing strategies, assumptions, risk and mitigations, and testing schedule.

Project Overview

Clone Order Management (COM) provides a platform for Vendors and Clients to be able to communicate for customized orders. They will be able to specify their Quote with an adequate amount of details and communicate to the vendors. The vendor will be able to see the request and communicate updates regarding shipping to the client. The client will be able to track the shipping of the requests.

In Scope

- Functionality Testing
    - Functionality testing is testing where the system is tested against the functional requirement/specifications. Functionality testing will be carried out in order to determine whether different functionalities are working as specified. Functional testing is high-priority testing that should not be compromised. Various techniques like Boundary Value Analysis, Equivalent Partitioning, and Error Guessing Techniques will be used.
- Smoke Testing
    - Smoke Testing is a software testing technique performed post software build to verify that the critical functionalities of the software are working fine. Smoke testing will be performed to ensure that the most important functions work properly.
- API Testing
    - API testing is a type of software testing that involves testing application programming interfaces (APIs) directly and as part of integration testing to determine if they meet expectations for functionality, reliability, performance, and security. API testing will be done using Postman so that the API can be

checked without the GUI being built. Early verification of the API will be beneficial if bugs are found early.

- UI Testing
    - UI testing is the process of testing the visual elements of an application to validate whether they accurately meet the expected performance and functionality. UI testing will be done to verify the actual product with the specified design mockups. Mockups are prepared in Figma. The UI should be verified with the prepared Mockup for alignment, color, padding, pixels, font size, font color, and the overall designs.
- Usability Testing
    - Usability testing is a technique for measuring how easy and user-friendly a software application is. Testing is done keeping Scientists in consideration for ease of their usability while using the system. The system is tested to ensure that system is easy to use and easily understandable.
- Sanity Testing
    - Sanity testing is a subset of regression testing. Sanity testing is performed to ensure that the code changes that are made are working as properly. High-level testing will be conducted to ensure the major functionalities are working as expected. Sanity testing will be conducted per the sanity checklist.
- Regression testing
    - Regression Testing is done once a new feature is added to the system that affects the other related components. Related components are tested to make sure that the related component is working properly.

Out Scope

- Security Testing
    - Security testing is a type of software testing that uncovers vulnerabilities, threats, risks in a software application and prevents malicious attacks from intruders. The DevOps team will be responsible for conducting security testing.
- Load Testing and Stress Testing:

- Load Testing is performed to test the level of load that can be handled by the application. It is usually done to ensure that the increase in the number of users will not affect the functionality of the user. Since the Clone Order Management application is used by only a few individuals of a company, the issue of more load-causing issues in the system will be very rare.

Test Approach

Scrum methodology of the Agile model will be followed.

Following the Software Testing Life Cycle (STLC) activities will be performed while conducting the tests.

- Requirement Analysis
  - The requirements provided by the stakeholders will be analyzed and documented. A feasibility study of the requirements will also be carried out.
- Creating Test Plan
  - A test plan is created that serves as a blueprint for testing activities. Various types of testing that will be performed on the application are mentioned along with execution strategy.
- Creating Test Cases
  - Test case document is the set of activities that will be performed to verify whether the functionalities are working as expected. Test cases will be prepared based on the user stories and acceptance criteria. Test cases provide full coverage of different aspects of the module under test. Google Sheets will be used to document test cases.
- Test Environment Setup
  - Proper environments will be set up to execute the test cases. Test Environments include hardware and software requirements, different browsers on which application is to be tested, different environments like Dev.
- Test Execution
  - After test environments are set up, test cases will be executed. Bugs will be reported if a module does not meet its requirements.

- Test Closure
    - Test Reports and other deliverables will be generated for stakeholders, confirming that all incidents have been reported and testing has successfully been performed on the application.

Testing Environment

Following people are responsible to verify the items in different environments:

Environment	**Tester**
Dev	Developers, QA, PM, TL
QA	QA, PM, TL
UAT	QA, PM, TL, User

Tools

Following tools are used during the testing process.

S.N.	**Tools**	**Purpose**
1	Postman	API Testing
2	Google Sheets	Preparing test cases, bug report

3	Google Docs	Requirement documents, test plan

Assumptions

Following assumptions are made to achieve our testing goal and maintain the quality of the product:

- It is assumed that story points are allocated correctly for every task so that there is plenty of time for development and testing.
- Users with different roles are created.
- Domain knowledge is known.
- The requirements are clearly defined.

Risks and Mitigation

S.N.	**Risks**	**Impact**	**Probability**	**Mitigation plan**
1	Misunderstanding between stakeholders, development team, and QA regarding requirements.	High	High	Constant communication between stakeholders, QA, and development team.
2	Delay in testing due to change in requirement in the late stage of the sprint.	High	High	All requirements should be discussed with stakeholders and clear confusion if any.
3	Incorrect estimation of the story points will hamper development and testing.	High	High	Story points must be estimated including the time needed for the code review and

				testing.
4	Change in team members may lead to increased development time as new members need time to understand the tasks and process.	Low	Medium	Same team members throughout the development process. Onboarding for the new members for quick understanding of the project.
5	Delay in the environment setup.	High	High	Frequent updates and healthy communication between development teams and stakeholders.
6	Inexperienced member of the team.	Medium	Low	Training sessions for the basics.
7	Lack of proper resources that are used for the testing phase.	High	Low	Proper estimation of resources should be done beforehand and backup should be planned for emergencies
8	Dependency in other Module	High	Low	Proper Coordination with the other module and detailed pre-planning to sync up the process should be done.

Test Deliverables

- Test Plan Document

    - A test plan is a document describing software testing scope and activities. It is the basis for formally testing any software/product in a project. Test Plan gives the overview of the features to be tested and the strategies to be followed while testing so that the team follows a protocol consistently.

- Test Cases
    - Test Case describes the purpose of a specific test, identifies the required inputs and expected results, provides step-by-step procedures for executing the test and outlines the pass/fail criteria for determining acceptance. Test cases are maintained in google sheets and linked to Jira tickets to track the progress of the feature.

- Bug Report
    - Bug Report is proper documentation of testing results, all the bugs, issues are documented in a proper way. It contains module name, issue description, steps to reproduce, actual and expected results. Jira is used to report bugs found in the project.

- Sanity Checklist
    - Sanity checklist documents the sanity test which is performed to check that all the major functionalities of the project are working fine. It outlines the pass/fails criteria of each functionality.

- Sprint Report
    - Sprint Reports are designed to help you understand your team's performance across active & completed sprints. You can quickly understand the work that has been completed and the ones that were returned to the backlog with the Sprint Reports.

- UAT Plan
    - UAT Test Plan documents the happy path testing steps to guide the user through the process. Clear steps and screenshots of necessary illustrations are documented along with an overview of the feature that has to be tested.

- Release Notes
    - Release notes are documents that are distributed with software products or hardware products, sometimes when the

product is still in the development or test state. For products that have already been in use by clients, the release note is delivered to the customer when an update is released. Release note outlines the corrections, changes, or enhancements made to the product.

Schedule

Task	**Schedule**
User Story And Acceptance Criteria	Before starting of sprint
Test Cases	In parallel with the development cycle on a sprint basis
Test Execution	After deployment of each new feature in Dev
Bug Report	During the sprint
Sanity Test	After Dev release
Sprint Report	At the end of each sprint
UAT Test Plan	After deployment to the UAT environment