Test Plan

Document Version Number: 3.0

Project Team:
Syed Abrarul huq [Lead]
dhomesh Beesetty [QA]
Shrinath Ramarao Pawar [QA]
Sneha [QA]

Project Sponsor:
Winnovate Capstone

I. Introduction

This Test Plan details the testing approach, scope, resources, schedule, and activities for the testing of the Swiggy web application, considering the newly provided test cases. The project will cover Sanity Testing, End-to-End (E2E) Functional Testing, Regression Testing, and Negative Testing.

II. Test Plan

- **Validate** the functionality of the Swiggy application according to the specified test cases.
- **Identify and log** defects, ensuring they are resolved to maintain application quality.
- **Verify** that updates and changes do not adversely affect existing features.
- Below is the template for test design

Test ID	Description	Expected Results	Actual Results

Where:

- Test ID is a unique identifier for the test case. The unique identifier should relate back to the particular requirement the test case is verifying. For example, if your naming scheme for requirements is numbers, test cases for requirement 3 could have test IDs TC3.
- Description should clearly document the steps that need to be done in order to run the test case. Write the description specifically, such that any team member can run the test case, even if the author of the test case is not present.
- Expected results is a statement of what should happen when the test case is run.
- Actual results are an indication of whether the test case is currently passing or failing when it is run. The actual results could be recorded simply as "Pass" or "Fail." However, it is also helpful to describe what happened in cases where a test case fails.

Ultimately, your customer should agree to the test case. When test cases are written so specifically, often requirements understanding is enhanced.

III. Testing Deliverables

3.1 Test Plan Document:

Test plan document is a formal document that outlines the strategy, scope, objectives, resources, schedule and activities required to effectively test a product, system or application

3.2 Test Cases Design:

Based on the requirement shared by the project team, testcases are designed on High level and low level and is documented

3.3 Requirements Traceability Matrix (RTM):
Document that maps and traces user requirement with test cases.

3.4 Test Execution Reports:
All the testcase evidence and the status of execution is reported in this document.

3.5 Exit Report:
This report consists of all the details of number of testcases passed, defects created during execution, outstanding testcase count.

3.6 Final Project Closure Documentation:
Post closure of execution the final report on the project is extracted and shared across to be stakeholders.

IV. Environmental Requirements

- As part of this demo project we will be testing in the live portal. However, we have the understanding that for live projects we will have testing environments which will be confirmed by the project team, in which the testing will be performed.
- Test data readiness for all testcases
- Testing tools such as the application (Swiggy here), automation tool : Selenium setup is done

V. Staffing

5.1 Team Composition:

- **Lead Member:** Syed
- **Team Members:** Dhomesh, Shrinath, Sneha

5.2 Roles & Responsibilities:

- **Syed (Lead):** Project oversight, test planning, and coordination with team members and Designing and Execution.
- **Sneha:** Responsible for designing and executing E2E Functional.
- **Dhomesh:** Manages Defect Logging, Triage and Executing Regression Testcases.
- **Shrinath:** Focus on Sanity and Negative Test Cases.

Note: The total estimated hours are distributed among the 4 team members, ensuring balanced workload and efficient utilization.

VI. Schedule

PHASE	START DATE	END DATE	DURATION (DAYS)	TASKS
Requirement	07/29/2024	08/07/2024	7	- Gather and analyze requirements

Gathering				- Identify and document test cases
Test Planning	08/08/2024	08/16/2024	7	- Develop Test Plan - High-level and Low-level Test Case Design - Create RTM - Test Schedule creation - Test Environment setup
Test Case Design	parallel with Test Planning	08/23/2024	5	- Design 19 Test Cases (E2E Functional, Sanity, Regression, Negative)
Test Execution	08/23/2024	09/05/2024	10	- Execute Sanity, E2E Functional, Regression, and Negative Testing - Manage defects
Defect Management	8/23/2024	09/05/2024	Considered along with Execution	- Log and triage defects - Retest and close defects
Project Closure	09/05/2024	09/08/2024	3	- Prepare Exit Report - Attach all evidence to test cases and defect logs - Final project documentation and formal closure

VII. Risks and Contingencies

Risk	Impact	Mitigation Strategy
Non-stable Environment	High	- Ensure environment setup activities are completed before execution. - Buffer time for environment stabilization.
Data unavailability	High	- Request data in advance and create backups. Validate data before execution.
Delay in Defect Triage	High	- Identify key Points of Contact (POCs) for all involved development teams. Prepare escalation paths before starting.
Resource Unavailability	Medium	- Crosstrain team members and ensure backups are ready.
Scope Creep	High	- Clearly document and define the project scope. Obtain formal signoffs from stakeholders.

VIII. Communication Plan

8.1 Daily Stand-ups: Updates on progress and any blockers.
8.2 Weekly Status Reports: Provide comprehensive progress reports to the stakeholders.
8.3 Project Closure Meeting: Formal communication to the project team regarding testing completion and closure.

IX. Conclusion

This revised Test Plan outlines a systematic approach to testing the Swiggy application using updated test cases. With clear goals, well-defined phases, and robust risk management, the project is set for successful execution and on-time completion.