
Software Test Plan

For

<Nexchar Point of Sale System>

Version 1.0

Prepared by –>Humayara Chowdhury Rafa

Date -> 19th May, 2021

Table of Contents

1. Test Plan Identifier	3
2. References.....	3
3. Introduction	3
4. Test Item.....	3
5. Features to be Tested.....	3
6. Features not to be Tested	4
7. Approach	4
7.2 Test Overview:.....	4
7.3 Test Tool:	5
7.4 Meeting:.....	5
8. Item Pass/Fail Criteria	5
8.1 Component.....	5
8.2 Integration.....	5
8.3 System.....	6
9. Suspension Criteria and Resumption Requirements	6
9.1 Automated Unit Test Suite	6
9.2 Build Acceptance Test (BAT).....	6
9.3 Regression Testing.....	6
9.4 System Design Changes:	7
10. Test Deliverables	7
11. Remaining Test Tasks.....	7
12. Environmental Needs.....	8
13. Staffing and Training Needs	8
13.1 Staff Needs	8
13.2 Training Needs.....	8
14. Responsibilities.....	8
15. Schedule	9
16. Planning Risks and Contingencies	9
17. RISKS & MITIGATION:.....	10
18. Approvals.....	12
19. Glossary	12

1. Test Plan Identifier

Nexchar_POS_System_1.0

2. References

- i. Development and Test standards Process: IEEE 829-1998 Standard for Software Test Documentation.
- ii. <https://www.gcreddy.com/2017/02/ieee-software-test-plan-template.html>

3. Introduction

This document is an outline of our testing plan for the Mobile POS System and eCommerce portal that performs all of the day-to-day operations. This project aims to provide a public-facing Point of Sale App for in-person and online orders. This document will cover the various requirements that will be applied to the listed application's device, integration, and device testing. The white box, black box, and system-testing paradigms can be used to apply testing criteria. The assessment requirements, processes, and test cases of the overall design would be included in this paradigm, but not limited to them. We'll use the test documentation requirements outlined in IEEE Standard 829 for Software Test Documentation during the testing period.

4. Test Item

The major functionalities of the system are as follows:

- 24/7 services
- Touch screen menu selection
- Product/item availability list on display
- Multiple purchase issue in one transaction
- Product barcode scan available
- A glitch-free customer experience across eCommerce channels
- Earlier identification of defects
- Faster development and deployments
- Improved business outcomes
- Enhanced user experience and stable performance across the POS software system
- Cancellation of transactions any time during the transaction
- Credit/Debit card transaction.

5. Features to be Tested

- Touch Screen monitors and keyboard interface.
- Oracle database server support.
- Credit/Debit card transaction.
- Product/Item cancellation support before the final confirmation of the purchase.

- Purchased item cancellation support by the administrator
- Item information availability.
- Multiple product purchase support simultaneously.

6. Features not to be Tested

- Network Security
- Hardware related issues
- Any personal information.
- Low risk, has been used before successfully and is considered stable
- UI/UX Design

7. Approach

7.1 The following represents the overall flow of the testing process approach:

Determine which conditions will be put to the test. The existing Program Specification must be used to develop all test cases. Determine the test(s) that will be used to put each module through its paces. Examine the test data and test cases to ensure that the device has been properly tested and that the test data and test cases are sufficient to confirm proper operation. For each test, determine the expected outcomes.

The test case setup, test data, and expected results should all be recorded. Carry out the examinations. During the testing process, keep track of the test results, test cases, and test configuration. The Unit/System Test Report will be used to submit this information (STR). Before the unit is eligible for component integration/system testing, it must pass unit testing.

Unsuccessful testing necessitates the development of a Bug Report Form. The test case, the problem observed, the probable cause and the sequence of events that led to the problem are all described in this document. It will serve as a foundation for future technological research. Documents and results from the tests must be submitted. Any requirements that need to be reviewed, changed, or improved must be done right away.

7.2 Test Overview:

- Integration Testing: N/A
- System Testing: N/A
- Acceptance Testing: N/A
- Regression Testing: Automated Regression suite will be executed as part of Regression.

7.3 Test Tool:

- Google Spreadsheets/Microsoft Excel will be used to write test cases.
- Google Spreadsheets/ JIRA-Test Management Tool will be used to manage bug reporting.

7.4 Meeting:

He / She should schedule a team meeting anytime the Project Manager (PM) considers it necessary. Team leader meetings are very casual and rare, and the activity needed for one of the meeting purposes. Flow Matrix for reporting and information:

Action	Weekly	Monthly
Status meetings	Project Team Meeting, Project Manager & Department Director	Project Manager and Directors
Reports	Complete Tasks, Details on Schedules & Tasks for Next Week	Summary of Standard Reports

8. Item Pass/Fail Criteria

This section lays out the general pass/fail requirements for the assessments in this plan. They've supplemented the test design specification with pass/fail requirements.

8.1 Component

Component tests pass only if the signatures, constraints, and interfaces specified in the Object Design Specification for that component are met. Positive, negative, and stress tests, as well as boundary tests, are also included. If a test demonstrates that a product fails to fulfill the objectives of the object design specification, it will fail, and a fault/issue will be identified in the defect tracking system for triage.

8.2 Integration

The signatures, limitations, and interfaces dictated by both the object design specification and the system architecture specification are only satisfied when tests on integrated components pass. This includes positive, negative, and stress checks, as well as boundary conditions and explicit interface manipulation tests (such as the physical connection to the database server). If a test reveals that a product fails to fulfill the objectives of both the object design and system architecture specifications, it will fail and a defect/issue will be identified to the triage team for analysis.

8.3 System

The functional specifications, non-functional requirements, and use cases are used as the oracle in tests run against the system to decide if it passes or fails. If a test reveals that a product fails to meet the objectives of any of the functional specifications, non-functional requirements, or use cases, the test will fail, and a flaw or problem will be identified to the triage team for analysis.

9. Suspension Criteria and Resumption Requirements

This section specifies the criteria for suspending the testing on the test items associated with the plan. It also specifies the test activities that must be repeated when testing is resumed.

9.1 Automated Unit Test Suite

Unit tests will be written to test the interfaces of the components as they are created, and low-level unit tests will be written to test the methods of the underlying classes of the components. The build server will run the automated unit test suite on a per-build basis as a prerequisite to the BAT. When the unit-test suite reports failures, testing on that build will be halted before the failures have been assessed and resolved. Testing will continue once the automated unit test suite has been passed.

9.2 Build Acceptance Test (BAT)

As production deems a build ready for testing, a build acceptance test is performed on the build. The BAT will consist of a wide but shallow series of tests designed to assess the overall stability of the build and determine whether or not it is worth testing. If the BAT fails on a specific build, testing will be halted before another build is produced with any BAT failure issues resolved, which will be checked by running the BAT again. Testing will restart once the BAT is passed. For the various test phases, different build acceptance tests will be built and used. Component BATs will be tiny and specific to each component. The integration BATs can differ depending on the degree of integration testing conducted. The Device Test BAT will include a series of tests that will make use of all of the system's components.

9.3 Regression Testing

Significant bug fixes or code updates will be checked on a build-by-build basis to assess the impact on the framework. If the changes are considered to pose a significant risk, regression test sets of the required size will be generated and run. A system-wide regression test will also be performed on the release candidate build to ensure that incremental improvements to the system have not changed the effects of the tests run earlier in the test cycle.

9.4 System Design Changes:

All testing will be halted if any problems are reported that involve a device design adjustment. Following the revisions to the criteria, device architecture, and object design, the prototype specifications will be reviewed and updated to ensure that they are correctly aligned with the revised system modifications. Testing will continue after the modifications are completed. Both tests in the area affected by the shift must be rerun. To guarantee that the modifications did not have a detrimental effect on other aspects of the system, a 10% regression of other experiments must be run.

10. Test Deliverables

Test Deliverables are the test artifacts which are given to the stakeholders of a software project during the SDLC (Software Development Life Cycle). Every software application goes through different phases of SDLC and STLC.

- Test Strategy
- Test Plan Document
- Test Cases
- Test Data
- Test Defect Report
- Test Summary Report
- Test Status Report
- Error Corrective Actions
- Test Evaluation Report
- Execution Log Files
- Summary

11. Remaining Test Tasks

TASK	Assigned To	Status
Create Acceptance Test Plan	TM, PM, Client	N/A
Create System Test Plan	TM, PM, QA	N/A
Create Integration Test Plan	TM, PM, Client	N/A
Create Test Cases	TM, PM, QA	N/A
Defect Report	TM, PM, QA	N/A

12. Environmental Needs

- Hardware and software are needed to execute our system properly.
- The database is needed to execute the system.
- Access to the recovery process.
- Need maximum capacity to use the system at the same time.
- Need commonly used a browser to use the system.

13. Staffing and Training Needs

13.1 Staff Needs

During the first quarter of the project, the project/test manager will act as a full-time test engineer, with a part-time test engineer assisting with the project's evaluation and preparation. After the initial build is complete, validation and evaluation testing would include two full-time testers, one of whom should have been involved with the project from the beginning as an assisting tester.

13.2 Training Needs

The Test Manager and Project Manager would consult with the administration staff to learn about available courses and prerequisite course requirements, as well as to practice the inner workings of a course allocation flow. [Based on the earlier portion of this test plan text, further personnel and preparation facilities will be included later.]

14. Responsibilities

	TM	PM	DEV TEAM	TEST TEAM	CLIENT
Unit test Documentation & Execution		X		X	X
Unit Test of Project Planning & Responsibilities	X				X
System Design Reviews	X				
Detail Design Reviews	X				X
Test procedures and Rules Check			X		X
Test Plan Affordability		X	X	X	X
Screen & Prototype Reviews	X		X		

15.Schedule

Time	Test Preferences	Responsibilities
20/05/2021	Unit test of project planning	Test Team
28/05/2021	System design reviews	Test Manager
30/05/2021	Detail Design reviews	Test Manager
06/06/2021	Change control and regression testing	Test team and developers
12/06/2021	System testing documentation and execution	Test Team
21/06/2021	Change control and regression testing	Test Manager
27/06/2021	System testing documentation and execution	Test team and test manager
10/07/2021	Test case review	Test Manager

16.Planning Risks and Contingencies

Risk	Probability	Risk Type	Owner	Contingencies Approach
Unable to acquire the necessary number of skilled personnel as the components become ready to test.	35%	Personnel Schedule	Test Manager	Resources for components will be split between the existing resources. Schedule must be adjusted accordingly.
Unable to acquire some of the necessary hardware and software required for integration and system testing	25%	Equipment	Program Test Development manager	Utilize existing acquired hardware. Split test execution into morning and evening shifts such that testing can occur for multiple teams in the same day using the limited hardware.
Components are not delivered on time	20%	Schedule	Development Manager	Integration testing with those components must be delayed until the Component is delivered. Overall integration test approach may be modified to do an appropriate amount of bottom-up as well as top-down or sandwich integration.

				Schedule must be adjusted accordingly
Turnover	20%	Personnel	Test Manager	Testers will work in pairs on components. If a single member of the team decides to leave, a secondary testing with the knowledge of the component will still be able to train a new tester or finish the work. Schedule must be adjusted accordingly.

17.RISKS & MITIGATION:

Sl.No	Issues	Risk	Mitigation
1.	Network and Software Weaknesses	The most popular POS issues stem from an insecure network. Hackers can gain access to these vulnerable systems and steal sensitive data such as consumer credit card numbers and company account records.	If at all practicable, keep POS technology on a different, password-protected network. To - the risk of penetration, we can update passwords at least once every 90 days. It's also important to keep our applications up to date. Patches and updates for operating systems are constantly released by software developers. Being up to date requires getting access to the most up-to-date security procedures, which gives us the peace of mind we need.

2.	Device Faults	Malware programs loaded into the POS system's memory were responsible for the majority of the above attacks. Hackers will remotely inject malicious apps into POS networks and then steal data without the customer or the merchant ever noticing it.	Even though the network is safe, the computers must be as well. Passcode-protecting the laptops and tablets is an excellent first move. Employees should be taught to log out of the POS system once they leave it and to never exchange passwords or other sensitive details. Select technology devices with built-in security features. If one app is hacked, Apple's iPads, for example, would not compromise other users.
3.	Insufficient Troubleshooting Support	Since we'll be working with a variety of POS suppliers to run our system, we must have plenty of resources when POS issues occur. Our IT department has internal knowledge of our network, but problems with applications, hardware, and computers may need additional assistance. We can also experience user error, necessitating the presence of IT personnel to assist. If POS device suppliers don't have enough tech resources for IT and other workers, it can lead to a long resolution time and missed revenue.	Anyone who uses a POS device should get proper instruction. To streamline training and operations, look for stable, consolidated tech solutions that allow you to do more with fewer applications and tools. Set up safe remote-control solutions for the IT department to troubleshoot problems from afar. Build a framework for dealing with POS issues in collaboration with your POS collaborators.

4.	Phishing	Phishing is a relatively modern malware tactic in which hackers use email to trick workers into clicking on malicious links. Hackers can obtain access to your device and data by clicking a connection.	To avoid this, teach staff not to access any unusual emails or connections that are sent to them. Install a decent antivirus program that can detect and block this kind of malware automatically. To avoid this issue entirely, educate and track staff on how to use their POS devices so that they never check their email or browse the web on company-owned machines.
-----------	-----------------	--	--

18.Approvals

- Test manager then project manager proves Documentation and Execution.
- System Structure is approved by the Development Team Leader and Project Manager.
- To do the Unit testing, Test Manager Approval is needed.
- To do the Acceptance testing, Test Manager Approval is needed.
- Validation of data is approved by the Administration Personnel
- The Project Manager Test Manager then administration personnel approved the Final Project completion.

19. Glossary

POS	Point of Sales
PM	Project Manager
TEST SUITE	A collection of test cases that have the same test objective.
STR	System Test Report
BAT	BAT Build Acceptance Test
QA	QA Quality Assurance
STLC	Software Testing Life Cycle
SDLC	software development life cycle