# Test Plan for Vistock POS System

## Table of Content

1. **Introduction**
2. **Web Page Overview**
3. **Purpose**
4. **Scope of Testing**
5. **Formation**
6. **Passing Over**
7. **Testing Environment**
8. **Plan of Action**
   * Action-01: Test Case and Test Scenario
   * Action-02: Test Process After Getting an Application
9. **Test Process**
   * Smoke Testing or Sanity Testing
   * Functionality Testing
   * Usability Testing
   * Compatibility Testing
   * Performance Testing
   * Security Testing
   * Link Testing
   * Accessibility Testing
   * Regression Testing
   * Application Programming Interface (API) Testing
   * Automation Testing
10. **Test Diagram**
11. **Validation**
12. **User Acceptance Testing (UAT)**
13. **Defect Reporting Procedure**
14. **Schedule Table-001**
15. **Testing Tools**
16. **Defect or Bug Fixing**
17. **Recommendations**
18. **Acknowledgement**

## 1. Introduction

This document outlines the test plan for the **Vistock POS System**, a comprehensive point-of-sale solution used for inventory management, sales transactions, employee management, and other business operations. The test plan ensures that all features of the POS system perform as expected, offering a smooth, secure, and efficient experience for the users.

## 2. Web Page Overview

**Vistock POS** is a feature-rich point-of-sale system that supports a variety of functionalities, including product management, sales, purchases, employee tracking, reports generation, and much more. It provides easy-to-use interfaces for both cashiers and managers and supports critical business activities such as inventory management, purchasing, sales returns, and invoicing.

## 3. Purpose

The purpose of this test plan is to ensure that the **Vistock POS System** operates as intended in all required environments. This includes ensuring its functionalities, security measures, and compatibility with different hardware and software setups. The primary goal is to deliver a system that is secure, reliable, bug-free, and user-friendly across various platforms and devices.

## 4. Scope of Testing

The scope of testing includes the following functionalities:

* **Product Management** (Create, Edit, Delete products)
* **Sales Transactions** (Invoices, Returns, Payments)
* **Inventory Management** (Stock Updates, Transfers, Adjustments)
* **Employee Management** (Attendance, Payroll, Permissions)
* **Purchase Management** (Purchase Orders, Receipts, Suppliers)
* **Reports** (Sales Reports, Purchase Reports, Product Reports)
* **System Security** (Role-based access, data security, login/logout)
* **User Interface** (Usability, navigation, responsiveness)
* **API Integrations** (Communication with third-party services)
* **Mobile Compatibility** (Android, iOS, and mobile web interfaces)

## 5. Formation

The test plan will involve the following personnel:

* **Test Case Developed By**: [Your Name or QA Team Name]
* **Test Case Reviewed By**: [Reviewer's Name or Senior QA Lead]
* **Test Execution Team**: [Names of the testers or QA team]

## 6. Passing Over

Upon successful completion of the testing phases, the test cases and reports will be handed over to the development team for defect fixing. Post-fix, further rounds of testing will be conducted.

## 7. Testing Environment

* **Platforms**:
  + **Windows OS** (Chrome, Firefox, Microsoft Edge)
  + **Mac OS** (Safari)
  + **Android OS** (Chrome)
  + **iOS** (Safari)
* **Databases**: MySQL, PostgreSQL
* **Web Servers**: Apache, Nginx
* **Testing Tools**: Selenium, JMeter, Postman, JIRA

## 8. Plan of Action

### ****Action-01: Test Case and Test Scenario****

We will use various testing design techniques for creating test cases:

* **Equivalence Class Partitioning (ECP)**
* **Boundary Value Testing (BVT)**
* **Decision Table Testing**
* **Error Guessing**

### ****Action-02: Test Process After Getting an Application****

Once the application build is received, the testing process begins with smoke testing. If successful, further testing such as functional and performance testing will proceed. All defects identified during the testing will be reported and tracked.

## 9. Test Process

### ****Smoke Testing or Sanity Testing****

Ensure that basic functionalities like login, product management, and payment processing are working.

### ****Functionality Testing****

Test the core functionality of the POS system, including product creation, sales transactions, inventory management, and reporting.

### ****Usability Testing****

Test the user-friendliness of the interface, ensuring that it is intuitive and easy to navigate.

### ****Compatibility Testing****

Ensure the system functions across multiple browsers, devices, and operating systems.

### ****Performance Testing****

Check how the system performs under load, particularly during peak hours or high transaction volumes.

### ****Security Testing****

Test for vulnerabilities, ensuring that sensitive data like customer information and payment details are securely handled.

### ****Link Testing****

Verify that all links within the application (e.g., footer, menu) are working correctly.

### ****Accessibility Testing****

Ensure that the system meets accessibility standards for users with disabilities.

### ****Regression Testing****

Test the entire system after fixes and updates to ensure that no previously working functionality is broken.

### ****API Testing****

Test integrations with external systems, ensuring that APIs return the correct data and function as expected.

### ****Automation Testing****

Automate repetitive test cases, especially for regression and performance tests.

## 10. Test Diagram

A mind map will be created to visualize the workflows and modules of the **Vistock POS System**, mapping out all the critical components for testing.

## 11. Validation

**Validation** will confirm whether the system meets the user requirements and stakeholder expectations. This will involve reviews and approval from stakeholders to ensure that the system is in line with their needs.

## 12. User Acceptance Testing (UAT)

Once validation testing is complete, **UAT** will be conducted by real users or stakeholders to ensure that the system is fit for production use and aligns with the business objectives.

## 13. Defect Reporting Procedure

Defects identified during testing will be logged in **JIRA**, detailing:

* Steps to reproduce
* Expected vs. actual results
* Screenshots, if applicable
* Priority and severity

## 14. Schedule Table-001

| **Task Name** | **Date From-To** | **Days** |
| --- | --- | --- |
| Test Plan Creation | TBD | TBD |
| Test Case Writing | TBD | TBD |
| Bug Reporting | TBD | TBD |
| UAT | TBD | TBD |

## 15. Testing Tools

* **Bug Tracking**: JIRA
* **Performance Testing**: JMeter
* **API Testing**: Postman
* **Automation**: Selenium
* **Documentation**: Word, Excel
* **Screen Capture**: Lightshot

## 16. Defect or Bug Fixing

Defects will be fixed promptly by the development team and re-tested for validation. Defect resolution will follow the defect severity and priority guidelines set by the QA team.

## 17. Recommendations

Post-deployment recommendations include continuous monitoring of system performance, implementing security audits, and addressing user feedback to ensure that the Vistock POS system remains reliable, secure, and user-friendly.

## 18. Acknowledgement

This test plan is created with input from the QA team, development team, and business stakeholders. Collaboration between these teams ensures that the **Vistock POS System** is delivered with high quality and meets the business and user needs.