

# **Test Plan for Medicine Shop Automation Software**

- 1) Test Plan Identifier
- 2) References
- 3) Introduction
- 4) Test Items
- 5) Software Risk Issues
- 6) Features to be Tested
- 7) Features not to be Tested
- 8) Approach
- 9) Item Pass/Fail Criteria
- 10) Suspension Criteria and Resumption Requirements
- 11) Test Deliverables
- 12) Environmental Needs

**Abhishek Kumar Sah (19CS10004)**

**Saumyak Raj (19CS30040)**

**Hritaban Ghosh (19CS30053)**

# ***Test Plan and Test Suite for Medicine Shop Automation Software***

## **1 TEST PLAN IDENTIFIER**

**MSA-1.0**

### **Contact Information:**

Abhishek Kumar Sah [abhishekkumar2046@iitkgp.ac.in](mailto:abhishekkumar2046@iitkgp.ac.in)

Saumyak Raj 19CS30040 [rajsaumyak@iitkgp.ac.in](mailto:rajsaumyak@iitkgp.ac.in)

Hritaban Ghosh 19CS30053 [ghoshhritaban21@iitkgp.ac.in](mailto:ghoshhritaban21@iitkgp.ac.in)

## **2 REFERENCES**

Software Requirements Specification For Medicine Shop Automation Software- Code Cannibals

## **3 INTRODUCTION**

This is the Master Test Plan for the Medicine Shop Automation Software Project. This plan will address only those items and elements that are related to the Medicine Shop Automation Process, both directly and indirectly affected elements will be addressed. The primary focus of this plan is to ensure that the Medicine Shop Automation application provides all the information of the Medicine Shop's Inventory and its day-to-day transactions while allowing for improvements and increases in data acquisition and the level of details available (granularity). The project will have two levels of testing, Unit Testing, and GUI Testing.

The details are addressed in the approach section. The estimated timeline for this project is very aggressive (two (2) weeks), as such, any delays in the development process could have significant effects on the test plan.

## 4 TEST ITEMS

The following is a list, by version and release, of the items to be tested:

- A. Medicine package, Version **1.0**
- B. Medicine Inventory package, Version **1.0**
- C. Vendor package, Version **1.0**
- D. Customer Transaction package, Version **1.0**
- E. Vendor Transaction package, Version **1.0**
- F. Expired Medicines package, Version **1.0**
- G. Medicine Shop Automation Software software, the initial version to be Version **1.0**

A detailed listing of programs, databases, screens, and reports will be provided in the system and detailed design documents.

- H. B. Medicine Shop Automation Software System requirements document, MSA\_RQMT version **1.0**

## 5 SOFTWARE RISK ISSUES

There are several parts of the project that are not within the control of the Medicine Shop Software application but have direct impacts on the process and must be checked as well.

- A. Backup and Recovery of local databases must be carefully checked.
- B. The ability to restart the application in the middle of a process is a critical factor to application reliability.
- C. Database security and access must be defined and verified.

## 6 FEATURES TO BE TESTED

The following is a list of the areas to be focused on during testing of the application.

- A. Verify owner login
- B. Change login details
- C. Calculate threshold
- D. Generate items to be ordered
- E. Generate a list of expired items
- F. Print vendor wise payments
- G. Handling New Supplies
- H. Sell medicine process
- I. Add new vendor process
- J. Search for a medicine
- K. Calculate revenue and profit for a period

## 7 FEATURES NOT TO BE TESTED

The following is a list of the areas that will not be specifically addressed.

A. Various printing processes

Print vendor-wise payments, the print id of medicine, print a cheque favouring the vendor, and print cash receipt are not tested because they are features that are dependent on browser, os, and external hardware.

B. Database Manipulation and Data Retrieval

Testing of these features will be an indirect result of other testing efforts already present.

## 8 APPROACH

UNIT Testing will be done by the Code Cannibals Team and will be approved by Mentor and Professors. Proof of unit testing (test case list, sample output, data printouts, defect information) must be provided by the Code Cannibals Team to the Mentor and Professors before unit testing will be accepted and passed. All unit test information will also be provided.

### Measures and Metrics

The following information will be collected by the Code Cannibals team during the Unit testing process.

1. Defects by module and severity.
2. Defect Origin (Requirement, Design, Code)
3. Time spent on defect resolution by defect, for Critical & Major only.
4. Defects located at higher levels that should have been caught at lower levels of testing.

## 9 ITEM PASS/FAIL CRITERIA

The test process will be completed once all the functionalities of the software are verified and tested.

At the Unit Testing level, all the test cases must be completed with more than 90% of the test cases passed.

Code coverage of nearly 70-80% is a reasonable goal for system tests of most projects with most coverage metrics.

The test is considered to be suspended if any of the following is encountered:

1. The software crashes
2. The software produces incorrect output
3. The software takes more than expected time to produce the output.

The test is considered to be approved if the software produces the correct output as per client demand.

## 10 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

Testing is to be suspended in the following situations:

1. If the number of test cases failed is over 20%.
2. If the system crashes upon the majority of the occasions.

## **11 TEST DELIVERABLES**

1. Test plan document.
2. Test Suite which contains the Test cases.
3. Problem reports and corrective actions.

## **12 ENVIRONMENTAL NEEDS**

1. An additional hardware requirement is a printer. It is not a compulsory requirement.
2. The test data will be provided manually.
3. The recommended Operating System is Windows 10/Linux/MacOS. The recommended python version is 3.7 or more with required libraries (like SQLite and Django).