Medical Lab Automation System

Software Test Plan

By

Naresh R (13CS30024)
Barnopriyo Barua (13CS30009)

Software Engineering Assignment

TABLE OF CONTENTS

INTRODUCTION

The Software Test Plan (STP) is designed to prescribe the scope, approach, resources, and schedule of all testing activities. The plan must identify the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and schedule required to complete testing, and the risks associated with the plan. This Software Testing Plan provides a complete description of testing of all the functions and specifications of the Medical Lab Automation System (MLAS) developed for automating various activities required to be performed in a Medical Laboratory.

OBJECTIVES

The software test plan is aimed at verifying the functionality and correct working of every aspect and part of the software. The software should not cease to work correctly when it has been exposed to stress and all corner cases should be handled effectively. Testing software is an important part of the development life cycle of a software. It is an expensive activity. Hence, appropriate testing methods are necessary for ensuring the reliability of a software. According to the ANSI/IEEE 1059 standard, the definition of testing is the process of analyzing a software item, to detect the differences between existing and required conditions i.e. defects/errors/bugs and to evaluate the features of the software item.

TESTING STRATEGY

Testing is the process of analyzing a software item to detect the differences between existing and required conditions and to evaluate the features of the software item.

Specific test plan components include:

- Purpose for this level of test
- Management and technical approach
- Pass / Fail criteria
- Hardware/ software requirements
- Reported bugs

SCOPE

Testing will be performed at several points in the life cycle as the product is constructed. Testing is a very 'dependent' activity. As a result,

test planning is a continuing activity performed throughout the system development life cycle. Test plans must be developed for each level of product testing.

PURPOSE

The purpose of testing is to verify and validate a software and to find the defects present in the software. The purpose of finding those problems is to get them fixed.

- Verification:

  It is the checking or the testing of software for consistency and conformance by evaluating the results against pre-specified requirements.

- Validation:

  It looks at the correctness of the system, i.e. the process of checking the fact that what has been specified is what the user wanted.

- Defect:

  It is a variance between the expected and actual result. The defect's ultimate source may be traced to a fault introduced in the specification, design, or development (coding) phases.

TESTS

- Features to be tested:

  All the use cases are thoroughly tested by white box testing and black box testing. The databases are checked after every use case and if they do not match with the expected outcome, they are reported to be failed.

- Features not to be tested:
  - JDBC MySQL connector.

UNIT TESTING

Each unit has been tested for consistency by giving input manually through the GUI and the corresponding screenshots have been included in the Test Suite Results document.

INTEGRATION TESTING

All of the above units has been integrated together and tested for consistency. The dataflow from one unit to another has been verified.

INTERFACE TESTING

The software has been tested on machines with different configurations (for example computers with different operating systems like Windows or Linux) and the functionality of the interface has been verified.

SECURITY TESTING

Password validation system has been implemented for the management.

PERFORMANCE TESTING

Each of the use cases are tested with respect to their response times and are implemented in such a way that the response time is as small as possible. To achieve this number of SQL queries that are made while carrying out a particular use case is at most three. Furthermore only two queries, at most, suffice at all places except for the Bill generation and Test Report generation use cases. In this use case Bills, Patients and Tests are queried upon. The use case of generation of Test Reports also involves 3 queries on the databases, namely, Reports, Bills and Patients. These are the only use cases where three SQL queries are made and hence the use case with the maximum delay in response time.

USE CASE TESTING

The following use cases were checked to cover all Corner Cases as far as possible:

- If the management tries to introduce a test which already exists in the database a message box is displayed appropriately.
- If passwords and usernames do not match appropriate messages are displayed.
- If there are no notifications by the employees, a blank notification section is displayed for the management.

- If the patient comes to collect the test report before the promised date a message box is displayed appropriately.
- If a new stock item is added which already exists in the medical lab inventory a message box is displayed appropriately.
- If the value to be deducted is more than the existing value of the stock item, an appropriate message box is displayed and the bill generation fails.

ALPHA TESTING

The software was distributed among a group of 10 people in our Department and they were made to test its functionality.

PASS/FAIL CRITERIA

SUSPENSION CRITERIA

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

- The Software crashes.
- It results in wrong output.

APPROVAL CRITERIA

The test is considered accepted if any of the following is encountered:

- The correct results are obtained
- The software does not take much time to achieve the aforementioned criteria.

ENVIRONMENTAL REQUIREMENTS

HARDWARE

- Working network connection
- MySQL should be installed on an appropriate server so that it can handle simultaneous request from various users.

SOFTWARE

- Operating System: Windows/Linux
- Java should be installed (preferably JDK 7)
- JDBC libraries should be included
- MySQL should be installed

BUGS REPORTED

- As the Patient Database involves only name and gender, so there is no provision for addition of Patients with same name and same gender as there will be no unique way to identify each of the similar Patients. But this could be improved by adding more information about the Patient or using some unique id for the Patient.
- Tests with same name are not allowed to be added in the Medical Laboratory. It is logical enough not to have tests having same name but having different Test charges and normal Values.