



EXPERIMENT - 5

Software Testing and Quality Assurance

Abstract

Create a test plan document for Library Management System.

Syeda Reeha Quasar
14114802719

EXPERIMENT – 5

Aim:

Create a test plan document for Library Management System.

Theory:

A Test Plan is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. A test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager.

How to write a test plan:

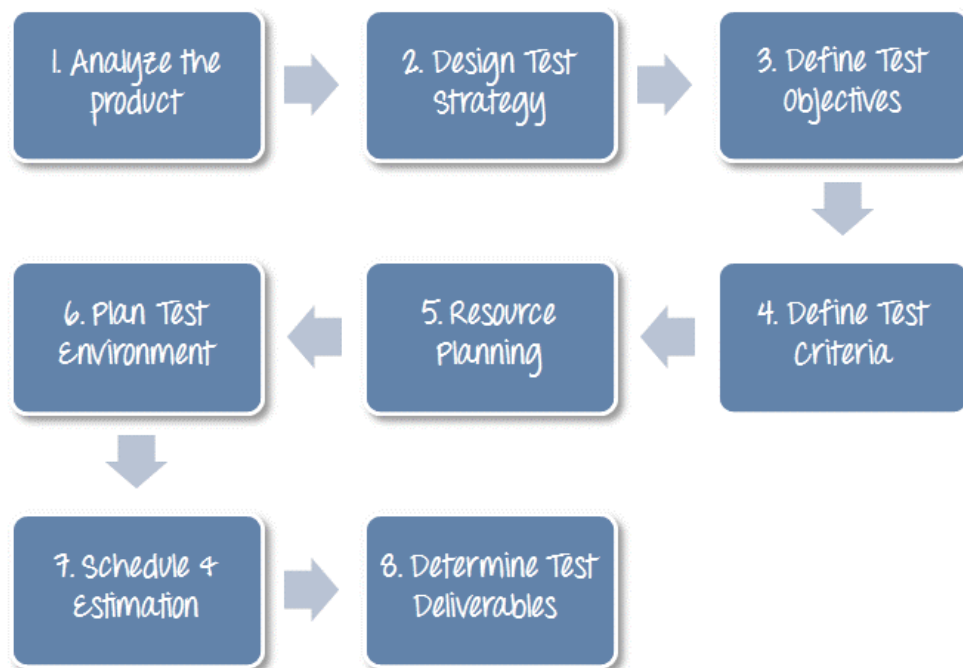


Table of Contents :

- 1) Introduction
- 2) Purpose
- 3) Scope
- 4) Testing Strategies

- 4.1. Unit testing
- 4.2 System & Integration testing
- 4.3. Performance testing & Stress testing
- 4.4. Acceptance Testing

- 5) features to be listed
- 6) Hardware requirements
- 7) Environment requirements
- 8) Test schedule
- 9) Risk & Mitigation
- 10) Tools

1. Introduction

Every college/ school has their library both for teachers and students to use. These traditional system to manage them is either keeping track of them in a register or keeping track of a similar entry in computer. Its very time consuming. Online library management system helps in solving this issue.

2. Purpose

The library management system is an online application for testing a librarian in managing book library in a university. This test plan document support the following objective :

- Identify existing project information of software that should be tested.
- List the recommended tools requirements.
- Recommend and describe the testing strategies to be employed.
- List the deliverable elements of the test activities.

3. Scope

The system that is to developed provides the related information on students and system administration

- Creating a system administrator who will be the sole user managing the system on the backend.
- System admin can add/ delete/view/ edit the books
- Admin can add/delete/view/edit the books issued
- Admin can search for the books issued.

4. Testing Strategies

The aim of the system testing process is to determine all the defects in the project.

4.1 Unit Testing :

In order to test a single module, we need to provide a complete environment and besides the module we could require

- The procedure belonging to other modules.
- Non local data structures that module accesses.
- A procedure to call the functions of the module on for test.

Unit testing was done on each of every modules that it describes under the module description :

1) Test for admin module:

- Testing admin for login
- Student account registration

2) Test for student login module :

- Test for student login interface
- Test for account creation

3) Test for teacher login module:

- Test for teacher login interface

4.2 System of integration testing :

The primary objective is to test the module interfaces

- UI user interface module, w/c is visible to end user

- DBMS is the database management system w/c has all data
- VAL is the validation module
- CNT : these contents are displayed in reports.

4.3 Performance of stress testing

Stress testing involving beyond normal operation capacity

4.4 User Acceptance Testing

There are different types of acceptance testing . The most common among them is the user acceptance (UA)

Test Schedule

S. No	Task	Days	Start Time	End time	Responsibility
1.	Understanding and analysis	5	2 July	7 July	Team
2.	Generating test cases	10	7 July	17 July	Member 1
3.	Test case documentation	40	17 July	17 July	Member 2
4.	Verify env. step	1	17 Aug	17 Aug	Member 3
5.	Unit testing	10	18 Aug	28 Aug	Member 4
6.	IVT testing	15	7 Sept	22 Sept	Member 5
7.	Final testing	15	21 Sept	24 Sept	End user 1
8.	Eval. test criteria	2	22 Sept	24 Sept	Member 1
9.	Summary report	1	25 Sept	25 Sept	Team

5) Features to be tested

- GUI testing
- Database testing
- Basic operations add/delete/etc
- Advance operations
- BIU

6) Hardware requirements / Software requirements

- Windows -OS
- Python language
- MYSQL database
- Visual studio code - IDE

7) Environment requirement

- Mainframe- Specify both the necessary and designed properties of test acquirement
- Work Stations - computers provided in the libraries to be used by admins and students.

8) Risk and mitigation

Keep battery back up and avoid electricity issues

9) Tools

- Selenium
- 2 pp

Solution:

What are the requirements of the library management system?

- Users can register and log in.

- The user can search for the added books and check in or out.
- The user can pay the fine or extend the duration of the borrowed period.
- The user can change the password and other profile details.
- The user can add books.
- The user can place the holds and modify existing holds.
- The user can manage the inventory of the books.

These are some of the common features expected from the library management system. So you have to write a test case for library management system to check for. In addition to these test scenarios, you have GUI-based software to check for bugs, usability, and functionality.

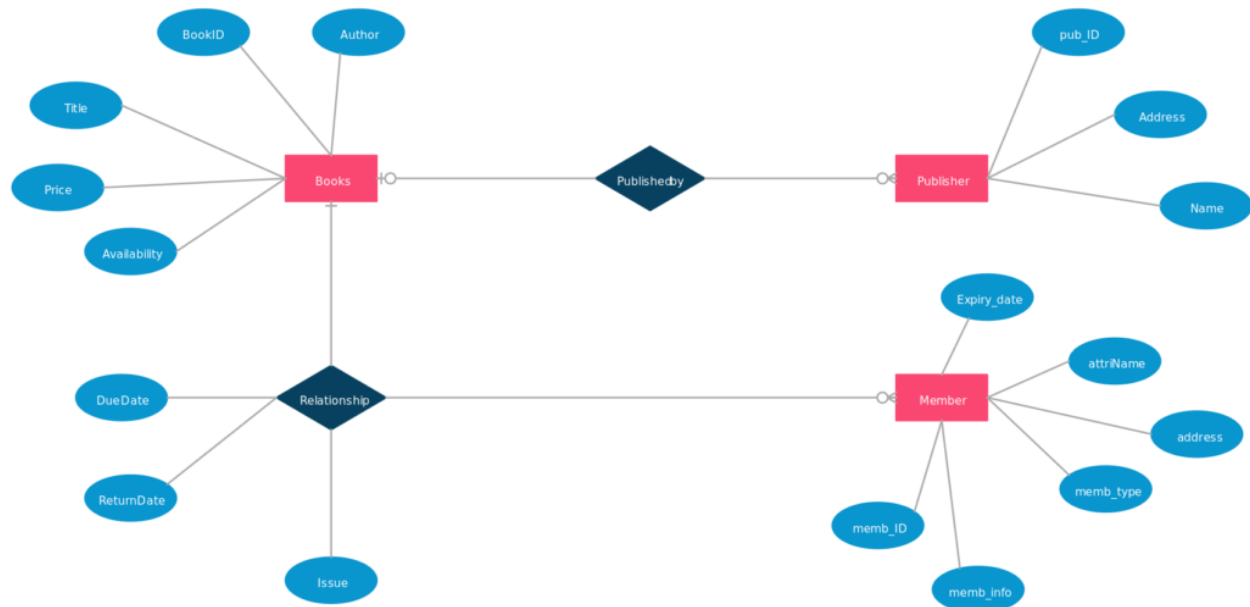
- Test Scenario For Hospital Management System
- Test Case For HR Payroll System
- Test Case For Computer Keyboard
- Test Case For Mouse
- Test Case For New Folder
- Test Case For Notepad

What is Library Management System?

The purpose of a library management system is to operate a library with efficiency and reduce costs. The system is entirely automated and streamlines all the tasks involved in the operations of the library. The activities of book purchasing, cataloging, indexing, circulation recording, and stock checking are done by the Library Management System software. Such software eliminates the need for repetitive manual work and minimizes the chances of errors.

The library management system software helps in reducing operational costs. Managing a library manually is labor intensive and an immense amount of paperwork is involved. An automated system reduces the need for manpower and stationery. This leads to lower operational costs.

The system saves time for both the user and the librarian. With just a click the user can search for the books available in the library. The librarian can answer queries with ease regarding the availability of books. Adding, removing, or editing the database is a simple process. Adding new members or canceling existing memberships can be done with ease.



Decision Table:

Result:

Test plan for Library management system was created and verified.

Viva Questions

Q1. Why we use decision tables?

The techniques of equivalence partitioning and boundary value analysis are often applied to specific situations or inputs. However, if different combinations of inputs result in different actions being taken, this can be more difficult to show using equivalence partitioning and boundary value analysis, which tend to be more focused on the user interface. The other two specification-based techniques, decision tables and state transition testing are more focused on business logic or business rules. A decision table is a good way to deal with combinations of things (e.g. inputs). This technique is sometimes also referred to as a 'cause-effect' table. The reason for this is that

there is an associated logic diagramming technique called 'cause-effect graphing' which was sometimes used to help derive the decision table.

Q2 What are the disadvantages of Decision Table testing?

The main disadvantage is that when the number of inputs increases the table will become more complex.

Q3. Why Decision Table Testing is Important?

Decision Table Testing is Important because it helps to test different combinations of conditions and provides better test coverage for complex business logic. When testing the behavior of a large set of inputs where system behavior differs with each set of inputs, decision table testing provides good coverage and the representation is simple so it is easy to interpret and use.

In Software Engineering, boundary value and equivalent partition are other similar techniques used to ensure better coverage. They are used if the system shows the same behavior for a large set of inputs. However, in a system where for each set of input values the system behavior is different, boundary value and equivalent partitioning technique are not effective in ensuring good test coverage. In this case, decision table testing is a good option.

This technique can make sure of good coverage, and the representation is simple so that it is easy to interpret and use.

This table can be used as the reference for the requirement and for functionality development since it is easy to understand and cover all the combinations.

Q4. What are the advantages of Decision Table Testing?

1. When the system behavior is different for different inputs and not the same for a range of inputs, both equivalent partitioning, and boundary value analysis won't help, but a decision table can be used.
2. The representation is simple so that it can be easily interpreted and is used for development and business as well.
3. This table will help to make effective combinations and can ensure better coverage for testing
4. Any complex business conditions can be easily turned into decision tables
5. In a case we are going for 100% coverage typically when the input combinations are low, this technique can ensure the coverage.

Q5: Why is Decision Table also called a Cause-Effect table?

Decision table testing is a software testing technique used to test system behavior for different input combinations. This is a systematic approach where the different input combinations and their corresponding system behavior (Output) are captured in a tabular form. That is why it is also called as a **Cause-Effect** table where Cause and effects are captured for better test coverage.