University of Maryland Global Campus

CMSC 495

Professor Kayed

Group 1

Gonghweh Koo

Scott Richardson

Cynthia Reynolds

Jonathan Dreksler

November 11, 2020

Test Plan

Library Management System

Table of Contents

[1. Test Plan 4](#_Toc55933149)

[2. Introduction 4](#_Toc55933150)

[2.1. Background 4](#_Toc55933151)

[2.2. Purpose/Objectives 4](#_Toc55933152)

[2.3. References 4](#_Toc55933153)

[3. Test Items 4](#_Toc55933154)

[3.1. GUI Test 4](#_Toc55933155)

[3.2. Basic Function Test 4](#_Toc55933156)

[3.3. Database Test 4](#_Toc55933157)

[4. Format of Standard Test Case 4](#_Toc55933158)

[4.1. Test Case Number 4](#_Toc55933159)

[4.2. Requirement Number 4](#_Toc55933160)

[4.3. Test Case Description 4](#_Toc55933161)

[4.4. Expected Results 4](#_Toc55933162)

[4.5. Actual Results 4](#_Toc55933163)

[4.6. Pass/Fail 5](#_Toc55933164)

[4.7. Comments 5](#_Toc55933165)

[5. Features To Be Tested 5](#_Toc55933166)

[5.1. GUI test 5](#_Toc55933167)

[5.2. Database test 5](#_Toc55933168)

[5.3. Basic function test 5](#_Toc55933169)

[5.4. Check connectivity of the system 5](#_Toc55933170)

[6. Features Not To Be Tested 5](#_Toc55933171)

[6.1. Maximum simultaneous online users 5](#_Toc55933172)

[6.2. Licensing requirements 5](#_Toc55933173)

[7. Approach 5](#_Toc55933174)

[7.1. Unit Testing (class testing 5](#_Toc55933175)

[7.2. Integrity Testing (thread-based testing) 5](#_Toc55933176)

[7.3. Validation Testing 5](#_Toc55933177)

[8. Item Pass Criteria 5](#_Toc55933178)

[8.1. GUI test 5](#_Toc55933179)

[8.2. Database test 5](#_Toc55933180)

[8.3. Basic function test 6](#_Toc55933181)

[9. Suspension Criteria and Resumption Requirements 7](#_Toc55933182)

[10. Test Deliverables 7](#_Toc55933183)

[11. Environmental Needs 8](#_Toc55933184)

[11.1. Hardware 8](#_Toc55933185)

[11.2. Software 8](#_Toc55933186)

[12. UAT Test Cases 8](#_Toc55933187)

[13. Responsibilities 8](#_Toc55933188)

[14. Schedule 8](#_Toc55933189)

[15. Risks and Contingencies 9](#_Toc55933190)

[16. Glossary 9](#_Toc55933191)

Document Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Team Member |
| 9-Nov | 1.0 | Initial document | Scott Richards, Cynthia Reynolds |
|  |  |  |  |
|  |  |  |  |

# ****Test Plan****

* 1. Test Identifier
     1. STS\_LMS\_01

# Introduction

## Background

The Library Management System is an online application for assisting library patrons in accessing the assets in the library. The system willd provide a basic set of features to add/update library patrons, request/reserve library assets, search for library assets, and manage check-in/checkout processes. The test team will test the system based on the requirement specification.

## Purpose/Objectives

The purpose of this test plan is to provide specific details on how testing processes will be performed for the LMS. Objectives of the test plan include:

* Ensuring that the functionality being delivered works as expected.
* Ensuring the functionality required for library patron scenarios has been delivered.
* Ensuring the delivered functionality works to specification.

## References

* Group 1 Plans and Specifications\_v1.0.
* Group 1 Project Plan – Library Management System\_v1.1.
* IEEE 829-2008 – IEEE Standard for Software and System Test Documentation.

# Test Items

## GUI Test

Testing the LMS’ graphical user interface confirming proper functionality based on the specifications.

## Basic Function Test

Performing basic function testing will prove the software is performing the defined functions as expected by the library patron.

## Database Test

Performing database testing will ensure data mapping and data integrity.

# Format of Standard Test Case

## Test Case Number

Unique test case identification number.

## Requirement Number

Refers back to an specific section within the requirements documentation.

## Test Case Description

A description of the test to be executed.

## Expected Results

The expected results of each test executed.

## Actual Results

The actual results of each test executed.

## Pass/Fail

Test result indication of whether the test was successful (pass) or unsuccessful (fail)

## Comments

The Tester will place any necessary notes in this section.

# Features To Be Tested

## GUI test

System should provide a GUI for library patrons to interface with the frontend library database

## Database test

###### 5.2.1 Basic operations: add/update/delete/query database

###### 5.2.2 Advanced operations

## Basic function test

5.3.1. Add a new library patron

5.3.2. Update library patron personal information

5.3.3. Borrow boo

5.3.4. Borrow DVD

5.3.5. Return book

5.3.6. Return DVD

5.3.7. Search for book

5.3.8. Search for DVD

5.3.9. Check-in book

5.3.10. Check-out book

5.3.11. Check-in DVD

5.3.12. Check late fees

## Check connectivity of the system

# Features Not To Be Tested

## Maximum simultaneous online users

## Licensing requirements

Reason: Since they are free software, they need not be tested.

# Approach

## Unit Testing (class testing

## Integrity Testing (thread-based testing)

## Validation Testing

# Item Pass Criteria

## GUI test

Pass Criteria:

## Database test

Pass criteria: Results of all basic and advanced operations are normal (refer to section 6.4)

## Basic function test

* + 1. Add a new library patron

Pass criteria: Each new library patron should have following attributes: Name, Address, Email, Phone number, and Library card ID

* + 1. Update library patron information

Pass criteria: The record would be selected using the library patrons’ library card ID

Updates can be made to name, address, email, phone number, but not the library card ID

The record can be deleted if there are no books or DVD’s issued by library patron.

The updated values would be reflected if the same library patron’s library card ID is called for.

* + 1. Request a book

Pass criteria: Each book shall have following attributes: Call Number, ISBN, Title, Author name, Publication year. The retrieved book information should contain the four attributes.

* + 1. Return a book

Pass criteria: The book item can be returned using the call number

The book can be deleted only if no user has issued it.

The updated values would be reflected if the same call number is called for

If book were deleted, it would not appear in further search queries.

* + 1. Search for book

Pass criteria: The product shall let the library patron query books’ detail information by their ISBN, Title, or Author.

The search results would produce a list of books, which match the search parameters with the following Details: ISBN number, Title, Author

The display would also provide the number of copies which is available for issue

The display shall provide a means to select one or more rows to a user-list

The search display will be restricted to 20 results per page and there would be a means to navigate from sets of search results.

The library patron can perform multiple searches before finally selecting a set of books for check-in or check-out. These should be stored across searches.

A book may have more than one copy. But every copy with the same ISBN number should have same detail information.

* + 1. Check-in book

Pass criteria: Librarian can check in a book using its call number

The check-in can be initiated from a previous search operation where library patron has selected a set of books.

The return date would automatically reflect the current system date.

Any late fees would be computed as difference between due date and return date at rate of 10 cents a day.

A book, which has been checked in once, should not be checked in again

* + 1. Check-out book

Pass criteria:

Librarians can check out a book using its call number

The checkout can be initiated from a previous search operation where library patron has selected a set of books.

The library card ID who is issuing the book would be entered

The issue date would automatically reflect the current system date.

The due date would automatically be stamped as 10 days from current date.

A book, which has been checked out once, should not be checked out again

A library patron who has books overdue should not be allowed to check out any books

The maximum number of books that can be issued to a customer would be 10. The system should not allow checkout of books beyond this limit.

* + 1. View book detail

Pass criteria:

This view would display details about a selected book from search operation

The details to be displayed are: Call number, IBN, Title, Author, Publication year, Issue status (In library or checked out), If book is checked out it would display, User ID & Name, Checkout date, Due date

Books checked in should not display user summary

Books checked out should display correct library patron details.

* + 1. View library patron detail

Pass criteria: Librarians can select a user record for detailed view

The detail view should show:

a. User name, library card ID, Address, Email & Phone number

b. The books issued by user with issue date, due date, call number, title

c. Late fees & Fines summary and total

The display should match existing library patron profile

The books checked out should have their statuses marked

The book search query should show the library card ID correctly.

# Suspension Criteria and Resumption Requirements

|  |  |
| --- | --- |
| **Suspend Testing** | **Resume Testing** |
| Hardware and/or software is not available | Hardware and/or software resources become available |
| The output is not the same as the expected result | Output meets the expected results |
| Any critical defect impacting the testing progress | Defect has been identified and resolved |

# Test Deliverables

* **Test Plan – defines what is required to be performed in UAT.**
* **Designs – Criteria for UAT Acceptance.**
* **Test Cases – The input values and expected output values from the tests.**
* **The test data – The data used for testing the LMS functionality.**
* Software Test Specifications (STS) – Document that includes the plan for testing the software; specifies test cases and procedures to ensure the software fulfills the requirements as defined in the requirements specification.
* **Screen Prototypes –** Draft version of the LMS.

# Environmental Needs

## Hardware

The LMS is being developed via the Java programming language. The system testing will be performed on a personal computer (PC) with at least 2 GB of RAM and Internet connectivity.

## Software

Testing of the LMS will be performed on a Windows 7 or later, 64-bit operating system.

# UAT Test Cases

**All Users Test Cases**

## Login to the application.

* + 1. Enter username.
    2. Enter password.
    3. Press login button.

## Search for an item.

* + 1. Enter the search parameters.
    2. Press search button.

## View item information.

* + 1. Select an item from search results.
    2. Click to view information.

**Librarian User Test Cases**

## Check-out item.

* + 1. Search for item.
    2. Select check-out.
    3. Enter users id.

## Process returned item.

* + 1. Search for item.
    2. Deselect check-out.

## Add Patron user.

* + 1. Enter user information.
    2. Click add user.

## Remove Patron user.

* + 1. Enter user id.
    2. Click remove user.

## Add item to library.

* + 1. Enter item information.
    2. Click add item.

## Remove item from library.

* + 1. Search for item.
    2. View item information.
    3. Click remove item.

**Patron User Test Cases**

## Reserve an item.

* + 1. If item is checked out, user can select to reserve item.
    2. Click to reserve and a reserve flag is placed on item.
    3. When item is returned the user receives an email notification.

## Update profile information.

* + 1. User opens profile.
    2. Click edit.
    3. Change information as necessary.
    4. Click save.

## View late fees.

* + 1. User opens profile.
    2. Check late fees owed.

## Receive late item notification.

* + 1. After item is past due, user receives email notification.

# Responsibilities

|  |  |
| --- | --- |
| Team Member | Responsibilities |
| Cynthia Reynolds | Decide on features to be tested/not tested  Create and complete test plan  Perform test execution, as needed |
| Gonghweh Koo | Perform test execution, as needed. |
| Scott Richardson | Create user test cases  Perform test execution, as needed |
| Jonathan Dreksler | Focus on evaluating the quality of a user's experience when interacting with the LMS. Create and complete the test plan. Collaborate with developers in order to create test cases and test data. Report results, and makes recommendations to improve effectiveness and overall satisfaction of the user. |

# Schedule

**A Work Breakdown Structure (WBS) for the entire LMS is located in the Group 1 Project Plan – Library Management Sysstem\_v1.1.doc.**

# Risks and Contingencies

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk ID** | **Description** | **Impact** | **Probability** |
| 1 | UAT not completed | Unable to fix possible defects | Low |
| 2 | Late delivery of software | The number of tests performed will be reduced | Low |

# Glossary

* GUI – Graphical User Interface
* LMS – Library management system
* PC – Personal Computer
* RD – Requirements Documentation
* SRS – Software Requirements Specification
* STS – Software Test Specification
* UAT – User Acceptance Testing
* WBS– Work Breakdown Structure