Test Document

Prepared by Li Zhenbang

Waseda University

2018.01.07

Table of Contents

1. Introduction

According to the development standard of software development, the system designer should write the specific development documents, recording the development of software development, and provides practical guidance for software developers.

Meanwhile, when software problems occur, the documents produced by software development can help project developers with actual reference.
The expected reference staff include users, testers, developers, project managers, and other quality managers. For the normal operation of the system, with timely detection of possible errors, the team plans to test each module, each module design multiple use cases.

1.1. Aim and requirement

We need a library management system which can afford many users to connect on it.

Software testing is a crucial step in the software designing, which checks if the overall software project is successful. In the testing stage, I have prepared 6 modules to test. Both the manual test (GUI test) and the automation test (API test) are used in different modules in this test. The database is tested with the professional Tellium and other 5 modules are tested within the black box test.

1.2. Background

The library management system is used for the management of the library.
The end user of the software is a library admin, citizen reader, student reader, or all other employees. These users all have basic levels of computer usage, at least one input method, and do not have to do much basic instructional work in use.
Through the use of this system software, it can help the leaders, administrators and personnel of enterprises to conveniently understand or evaluate the employee files, attendance and appraisals, etc., and realize the operation of inputting, modifying and querying the employee information so that the entire library management more scientific, orderly and standardized.

Software maintenance staff is mainly for routine maintenance of the database, such as data backup and recovery, as well as the system hardware and software operating environment maintenance. As a result, maintenance personnel are required to have a high level of education and technical expertise, be familiar with the Windows operating system and computer hardware configuration, acquire basic database knowledge, and recognize software development and visual programming.

1.2.1. Introduction

Follow the development standard of software development, write specific development documents, record the development of software development, and provide practical guidance for software developers. Meanwhile, when software problems occur, the documents produced by software development can give project developers actual reference.
Expected reference staff include users, testers, developers, project managers, and other quality managers. For the normal operation of the system, timely detection of possible errors, the team plans to test each module, each module design multiple use cases.

1.2.2. Testing backgrond

Project Name: LMS_Library management system
Project Developer: Li Zhenbang
User: Library management department
Operating environment: Windows XP or superior edition
Database: MySQL

1.3. Definition

1. Database: An organized collection of related data stored on a storage medium. Unit testing, also known as module testing, is the smallest unit of software design - program modules, the correctness of the test.
2. Integration testing is also called assembly testing or joint testing.
3. Security: The system has set different levels of user permissions, only the background database administrator user can set or modify the entire system, ordinary rights of the user can log in to simply add, modify, delete operation, non-Login users can only browse the search function.
4. LMS (Library Management System)
   Company Human Resource Management Information System
5. Black Box test: Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings.

2. Test Introduction

This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

2.1. Test Preparation

Test Hardware

Computer Type	MacBook Air
CPU Type	Intel(R) Core(TM) i5-6400 CPU @ 2.70GHz
Memory capacity(RAM)	8.00GB

Test Software

OS	Microsoft Windows 7 professional
Installation Environment	IIS 5.0
Database	Microsoft SQL Server 2000

2.2. Test items

Login/logout(GUI Black box test)
Database(API Automated test)
Basic data module(GUI Black box test)
New book module(GUI Black box test)
Borrow module(GUI Black box test)
System module(GUI Black box test)

2.3. Goals and Guidelines

Describe any goals, guidelines, principles, or priorities which dominate or embody the design of the system's software. Such goals might be:

- The KISS principle ("Keep it simple stupid!")
- Emphasis on speed versus memory use
- working, looking, or "feeling" like an existing product

For each such goal or guideline, unless it is implicitly obvious, describe the reason for its desirability. Feel free to state and describe each goal in its own subsubsection if you wish.

2.4. Test Methods

Firstly, we will identify elements and objects in the system programming. The detail element semantics and relationship are also considered by the tester. Select

each scenario set of scenarios related to a single function point. Walk through the activity of the scenario, assigning responsibilities to each abstraction sufficient to accomplish the desired behavior.

Measuring defects per thousand source lines of code (KSLOC) is the traditional approach, and is still generally applicable to OO system.

3. Part test

We will have 6 modules to test as following.

3.1. Login/Logout Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_01	**Test Designed by:** Li Zhenbang
Test Priority (Low/Medium/High): Med	**Test Designed date:** 2018.1.2
Module Name: login/logout module	**Test Executed by:** Li Zhenbang
Test Title: Verify login with valid username and password	**Test Execution date:** 2018.1.7
Description: Test LMS login and logout module	

Pre-conditions: User has valid username and password
Dependencies: Designed UI and inner logic

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Navigate to login page	User= lishinho	User should be able to login	User is navigated to	Pass	
2	Provide valid username	Password: 2345		dashboard with successful	Pass	
3	Provide valid password			login	Pass	
4	Click on Login button			click	Pass	

Post-conditions:
User is validated with database and successfully login to account. The account session details are logged in database.

3.2. Database Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_02	**Test Designed by:** Li Zhenbang
Test Priority (Low/Medium/High): Med	**Test Designed date:** 2018.1.2
Module Name: Database	**Test Executed by:** Li Zhenbang
Test Title: Test the consistency and inner logic of the DB	**Test Execution date:** 2018.1.7
Description: DB module	

Pre-conditions: Testers have DB schemes and ER diagram
Dependencies: Mysql database Tellurium Notepad++

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Test Reader scheme	Reader scheme in DB	User should know how to use Tellium	The system return the overcome	Pass	
2	Test Book scheme	Book scheme in DB			Pass	
3	Test BorrowQueue scheme	BorrowQueue in DB			Pass	
4	Test User scheme	User scheme in DB			Pass	
5	Test Admin scheme	Admin scheme in DB			Pass	

Post-conditions:
The data in the database can be used and the con sistency and affordance can be used in the available level.
T

3.3. Basic Data Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_03	**Test Designed by:** Li Zhenbang
Test Priority (Low/Medium/High): Med	**Test Designed date:** 2018.1.2
Module Name: Basic data module	**Test Executed by:** Li Zhenbang
Test Title: Basic data module test	**Test Execution date:** 2018.1.7
Description: Test LMS Basic data module	

Pre-conditions: User has valid username and password
Dependencies: Designed UI and inner logic

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Reader management	User= lishinho; pwd:2345	Can write data in the database and turn into the aimed sphere	User is navigated to dashboard with successful	Pass	
2	BookType management	Type :news			Pass	
3	Bookinfo Management	Book: 42 degree gray			Pass	
4	Exit	exit		click	Pass	

Post-conditions:
User is validated with database and successfully navigated. The account session details are logged in database.

3.4. New Book Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_04	**Test Designed by:** Li Zhenbang
Test Priority (Low/Medium/High): Med	**Test Designed date:** 2018.1.2
Module Name: new book module	**Test Executed by:** Li Zhenbang
Test Title: new book module	**Test Execution date:** 2018.1.7
Description:	

Pre-conditions: User has valid username and password
Dependencies: Designed UI and inner logic

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Order book	Order the book1	Can write data in the database and turn into the aimed sphere	User is navigated to dashboard with successful	Pass	
2	Check book	Check the book1			Pass	

Post-conditions:
User is validated with database and successfully login to account. The account session details are logged in database.

3.5. Borrow Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_05 | **Test Designed by:** Li Zhenbang

Test Priority (Low/Medium/High): Med | **Test Designed date:** 2018.1.2

Module Name: Borrow module | **Test Executed by:** Li Zhenbang

Test Title: Borrow module test | **Test Execution date:** 2018.1.7

Description:

Pre-conditions: User has valid username and password

Dependencies: Designed UI and inner logic

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Borrowing book	Borrow book1	Can write data in the database and turn into the aimed sphere	User is navigated to	Pass	
2	Return book	Return book1		dashboard with successful	Pass	
3	Search book	Search book1		login	Pass	

Post-conditions:

User is validated with database and successfully login to account. The account session details are logged in database.

3.6. System Module

Project Name: LMS_library management system

Test Case Table

Test Case ID: LMS_06 | **Test Designed by:** Li Zhenbang

Test Priority (Low/Medium/High): Med | **Test Designed date:** 2018.1.2

Module Name: System module | **Test Executed by:** Li Zhenbang

Test Title: System module test | **Test Execution date:** 2018.1.7

Description:

Pre-conditions: User has valid username and password

Dependencies: Designed UI and inner logic

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Change password	Change root pwd to 1234	Can write data in the database and turn into the aimed sphere	User is navigated to	Pass	
2	User management	Add user Koyanagi		dashboard with successful	Pass	

Post-conditions:

User is validated with database and successfully login to account. The account session details are logged in database.

4. Conclusion

The main test work is to test software and make the test environment.

I have spent 3 days to test the system and the level of this project test work is in medium, all the test work took three days, after reading books, the Internet search related information, the use of three computers for testing. Test results are good, all the test has passed.

5. Glossary

LMS – Library management system
HDD - Hard Disc Drive
RAM – Random Access Memory
IE – Microsoft Internet Explorer
Software testing – an investigation conducted to provide stakeholders with information about the quality of the software product or service under test

6. Bibliography

[1]Grady Booch, Object-Oriented-Analysis-and-Design-with-Applications-3rd-Edition
[2] Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley. 1995.
[3] Milind G. Limaye (2009). Software Testing. Tata McGraw-Hill Education. p. 216. ISBN 978-0-07-013990-9.