

<Library Management System>

Software Requirement Specification

**Project Code: <LibMan>**

**Document Code: <LibManDOC >– v<1.0>**

**<Ha Noi, 08/03/2013>**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| 06/03/2013 | The whole document | A | First sketch | 1.0 |
| 07/03/2013 | Specific Requirement | A | Add use-cases detail description | 1.0 |
| 08/03/2013 | Introduction  Overall Description  Supporting Information | A | Add brief description | 1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table of Contents

[1 Introduction 4](#_Toc350531898)

[3.1.1 Purpose 4](#_Toc350531899)

[3.1.2 Scope of testing 4](#_Toc350531900)

[3.1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc350531901)

[3.1.4 References 4](#_Toc350531902)

[3.1.5 Overview 4](#_Toc350531903)

[2 Overall Description 5](#_Toc350531904)

[3 user requirements 6](#_Toc350531905)

[4 Specific Requirements 7](#_Toc350531906)

[3.1.1 System Features: 7](#_Toc350531907)

[4.1.1.1 Server 7](#_Toc350531908)

[4.1.1.2 Client side 20](#_Toc350531909)

[3.1.2 Software System Attributes 24](#_Toc350531910)

# Introduction

## Purpose

* Identify existing project information and the software components that should be tested.

## Scope of testing

* This project has 3 stages of Testing: Unit Test, Integration Test and System Test. Developers perform the Unit Test to check own created functions. Testers perform the Integration Test and System Test to test common sense and functions.
* Unit test: Developers must perform Unit test for all functionalities of application to find defects.
* Integration test: Test all functions of an application as correct as the system requirements after integrating functions. Every function will be tested.
* System test: For the System Test, the focus is put on:
* Functional, test all functions of an application as correct as the customer requirements.
* Data Volume, perform base on big test data to find performance defects.

## Definitions, Acronyms, and Abbreviations

|  |  |  |
| --- | --- | --- |
| ID | Acronym | Definition |
| 1 | DB | Database |
| 2 | PC | Personal Computer |
| 3 | HTTP | Hyper Text Transfer Protocol |

## References

* 4-UP Mobile Travel Guide SRS
* SRSExample-webapp.doc

## Overview

We intend to develop the application with the following main part:

1. System maintenance: this part contains most common tool for maintaining library system.
2. Domain Business: this part contains all module involving business activities such as lending items, return items, etc.
3. Client side: this part contains essential library features to borrowers such as view card information, browse title, etc.

# Overall Description

* 1. User characteristic:
* The Borrower is expected to be Internet literate and be able to use a search engine.
* The Librarian is expected to be Internet literate and to be able to use email with attachments.
* The Borrower and Librarian are expected to be Windows literate and to be able to use button, pull-down menus, and similar tools.

# user requirements

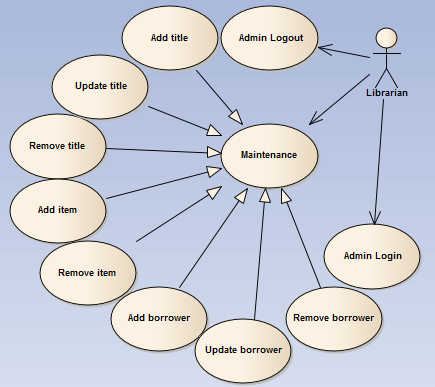
* This is a system whose functionality is assisting librarian in managing library.
* Library provides service of lending items to borrowers
  + Borrowers, books, magazines are pre-registered in the system.
* Library buys new books, magazine.
  + Popular books, magazine is bought in bulk.
  + Destroy old or broken down books, magazines.
* Librarian is the staff of the library communicating with borrowers and the system will assist his/her works.
* System should provide services of creating, updating and removing title, item, borrower and lending item in the system.
* System should be able to operate on Window 7 and have friendly user graphical interface.
* Easy to add new features to the system in future.

# Specific Requirements

## System Features:

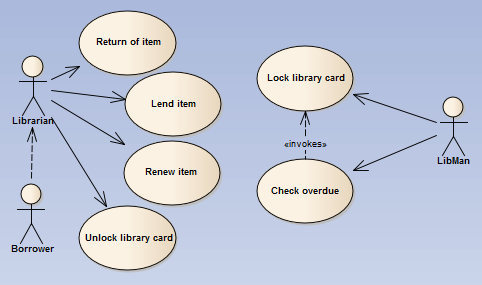
### Server

##### System maintenance



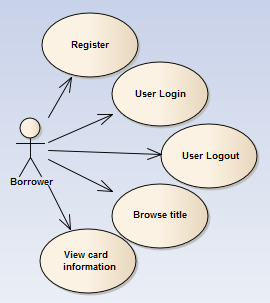
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - Specification | | | |
| Use Case ID: | MTN001 | | |
| Use case name: | Maintenance | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 06/03/2013 | **Date last updated:** | 06/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | This is an abstract use-case with the purpose of separating the business from the work of maintaining the system. | | |
| Pre-conditions: | N/A | | |
| Post-conditions: | N/A | | |
| Priority: | N/A | | |
| Activity step description: | N/A | | |
| Exceptions: | N/A | | |
| Special Requirements: | N/A | | |
| Notes and issues: | Because it’s an abstract use-case there is no need to implement it. | | |
| Use Case ID: | MTN002 | | |
| Use case name: | Admin Login | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 06/03/2013 | **Date last updated:** | 06/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | Librarians provide their access credentials in order to use the LibMan system. | | |
| Pre-conditions: | * To be pre-registered in the LibMan system * To have local network access through a computer. | | |
| Post-conditions: | A connection to database is established. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Login] screen for administrators (Librarians). 2. Librarian provides access information (user name and password) 3. The librarian information is then passed to the SystemKeeper module. 4. The SystemKeeper module queries the LibMan database. 5. The SystemKeeper accepts or rejects the login request. 6. If the login request is accepted librarian will be redirected to the Search screen. 7. If the login request is rejected an error message will show up and librarian can provide his or her access credential again. | | |
| Exceptions: | * In step 2, after 5 attempts of unsuccessful login, the account will be locked for 5 minutes. The only way to unlock it is to wait for 5 minutes. * In step 4, if LibMan cannot establish a connection to the database due to some reasons (ex: network failure) an error message will show up to tell the librarian about the problem. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | The content of the error messages will be decided by developers. | | |
| Use Case ID: | MTN003 | | |
| Use case name: | Admin Logout | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 06/03/2013 | **Date last updated:** | 06/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | When librarians finish their work they will logout of the system. | | |
| Pre-conditions: | * + To be logged in the system.   + To have local network access through a computer. | | |
| Post-conditions: | The connection to the database is terminated. | | |
| Priority: | High. | | |
| Activity step description: | 1. Librarian clicks [log out] button. 2. A logout request is sent to SystemKeeper module. 3. SystemKeeper module terminates the connection to the database. 4. Librarian is redirected to [Login] screen for administrators. | | |
| Exceptions: | N/A | | |
| Special Requirements: | * After 5.30PM of all working day (except for Saturday and Sunday) all administrators are automatically logged out. * If the logged in computer suffers from lost network connection it will automatically log out. * If the system is crashed or is forced to close unexpectedly all logged in users will be logged out. | | |
| Notes and issues: | N/A | | |
| Use Case ID: | MTN004 | | |
| Use case name: | Add title | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | A new book title is shipped to the library and the librarian adds the title together with related information (including code, ISBN, title, author, publisher, pages, CD, no of copies, available, call number) of the book to the database in order to well-manage it. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer | | |
| Post-conditions: | A new record containing data of the book is added to the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Librarian clicks the [New title] button on the tool bar and the [New title] window will show up. 2. Librarian provides all related information about the books then clicks [Save] button for each, a record corresponding to the book will be added to the [Book records] box. 3. After finish providing information, librarian clicks [Insert] button. 4. This information is passed to DBKeeper module. 5. DBKeeper module queries the database. 6. DBKeeper module accepts or rejects [Add title] request. 7. If the [Add title] request is accepted the [status] label will change from yellow “Standing by” to green “Successful” and the librarian is redirected to Search screen. 8. If the [Add title] request is rejected the [status] label will change from yellow “Standing by” to red “Failed” then an error message will show up to inform the librarian and he or she can reenter the information. | | |
| Exceptions: | * + In step 5, if the computer which sends the request suffers from lost network connection an error message will show up to inform the librarian. * In step 2, if librarian provides wrong data an error message will show up to inform his or her. * In step 2, if librarian provides insufficient data a message will appear at the top of the window and all required fields will be mark red. | | |
| Special Requirements: | Information of the book should satisfy the following conditions:   * **ISBN**: a string of 13 characters consist of 2 digits, a dash (-), a digit, a dash, 2 digits, a dash, 6 digits, a dash and a digit in the corresponding order. (Ex: 978-0-13-705346-9). Cannot be null. * **Title**: a string from 1 to 256 characters in length, can contain any kind of characters. Cannot be null. * **Author**: a string from 1 to 256 characters in length, can contain any kind of characters. * **Publisher**: a string from 1 to 256 characters in length, can contain any kind of characters. * **Pages**: a long integer in the range of [1 … 2­32]. * **CD**: an integer in the range of [0 … 1000]. Cannot be null. * **No of copies**: a long integer in the range of [1 … 232]. Cannot be null. * **Available**: a long integer in the range of [1 … 232]. Cannot be null. * **Call number**: a string of 11 characters consist of 3 digits, a dot, 2 digits, an underscore, an upper-letter and 3 digits. (Ex: 005.13\_C345). | | |
| Notes and issues: | The fields which don’t say “cannot be null” can be leave empty by default. | | |
| Use Case ID: | MTN005 | | |
| Use case name: | Update title | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | Modify an existed book title and related information (including code, ISBN, title, author, publisher, pages, CD, no of copies, available, call number) in the database. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | The record which librarian wants to modify is updated. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Search] screen. 2. Librarian searches for book title. 3. Librarian double-clicks on the title he or she wants to modify. 4. [Title update] screen shows up. 5. Librarian modifies the information then clicks [Update] button. 6. This data is passed to DBKeeper module. 7. DBKeeper queries the database. 8. DBKeeper module accepts or rejects the request. 9. If the [Update] request is accepted the [status] label will change from yellow “Standing by” to green “Successful” and the librarian is redirected to [Search] screen. 10. If the [Update] request is rejected the [status] label will change from yellow “Standing by” to red “Failed” then an error message will show up to inform the librarian and he or she can reenter the information. | | |
| Exceptions: | * In step 7, if the computer which sends the request suffers from lost network connection an error message will show up to inform the librarian. * In step 5, if the librarian provides wrong data an error message will show up to inform his or her. * In step 5, if librarian provides insufficient data a message will appear at the top of the window and all required fields will be mark red. | | |
| Special Requirements: | Same as use-case ID [MNT004](#MNT004_SpecialRequirements). | | |
| Notes and issues: | Same as use-case ID [MNT004](#MNT004_Note). | | |
| Use Case ID: | MTN006 | | |
| Use case name: | Remove title | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | Remove an outdated title from the system. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | The record corresponding to the title which librarian wants to delete is removed from the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Search] screen. 2. Librarian searches for book title. 3. Librarian double-clicks on the title he or she wants to delete. 4. The [Delete title] screen shows up. 5. Librarian clicks on [Delete] button, a warning message will appear to get him/her confirm his/her decision. 6. A [Delete] message is sent to the DBKeeper. 7. The DBKeeper queries the database. 8. Librarian is redirected to the [Search] screen. | | |
| Exceptions: | * In step 2, if no matched title is found an error message will appear to inform the librarian. * In step 7, if the computer suffers from lost network connection an error message will appear to inform the librarian. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | MTN007 | | |
| Use case name: | Add item | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | New copies of a title are bought so the librarians must add it to the database in order to well manage it. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | A new record containing data of the copies (including code) is added to the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Librarian clicks the [New item] button on the tool bar and the [New item] window will show up. 2. Librarian provides title to which the item is added then clicks [Search] button. 3. Librarian provides the code of the book then clicks [Save] button for each, a record corresponding to the book will be added to the [Item records] box. 4. After finish providing information, librarian clicks [Insert] button. 5. This information is passed to DBKeeper module. 6. DBKeeper module queries the database. 7. DBKeeper module accepts or rejects [Add item] request. 8. If the [Add item] request is accepted the [status] label will change from yellow “Standing by” to green “Successful” and the librarian is redirected to Search screen. 9. If the [Add item] request is rejected the [status] label will change from yellow “Standing by” to red “Failed” then an error message will show up to inform the librarian and he or she can reenter the information. | | |
| Exceptions: | * In step 5, if the computer which sends the request suffers from lost network connection an error message will show up to inform the librarian. * In step 2, if the title does not exist in the database an error message will appear to inform the librarian and he/she cannot enter the code of the item. * In step 3, if librarian provides wrong data an error message will show up to inform his or her. * In step 3, if librarian provides insufficient data a message will appear at the top of the window and all insufficient required fields will be mark red. | | |
| Special Requirements: | Information of item should satisfy the following conditions:   * **Code**: a string consist of 2 upper-case letters plus a forward slash (/) then 4 upper-case letters followed by 6 digits. (Ex: GT/TTHL00109). Cannot be null. | | |
| Notes and issues: | The fields which don’t say “cannot be null” can be leave empty by default. | | |
| Use Case ID: | MTN008 | | |
| Use case name: | Remove item | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarian | | |
| Goal/Description: | Remove an outdated or broken down item from the system. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | The record corresponding to the item which librarian wants to delete is removed from the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Search] screen. 2. Librarian searches for the item code. 3. Librarian double-clicks on the item record he or she wants to delete. 4. The [Delete item] screen shows up. 5. Librarian clicks on [Delete] button, a warning message will appear to get him/her confirm his/her decision. 6. A [Delete] message is sent to the DBKeeper. 7. The DBKeeper queries the database. 8. Librarian is redirected to the Search screen. | | |
| Exceptions: | * In step 2, if no matched item code is found an error message will appear to inform the librarian. * In step 7, if the computer suffers from lost network connection an error message will appear to inform the librarian. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | MTN009 | | |
| Use case name: | Add borrower | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | A new borrower is registered to the library and the librarian adds the information of the borrower to the database in order to well-manage it. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | A new record containing data of the book is added to the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Pending register] screen. 2. Librarian clicks on [Add] or [Add all] button. 3. If librarian clicks on [Add] button the chosen record will be passed to the DBKeeper module. 4. If librarian clicks on [Add all] button all record will be passed to the DBKeeper module. 5. DBKeeper module queries the database. 6. DBKeeper module accepts or rejects [Add borrower] request. 7. If the [Add borrower] request is accepted the [status] label will change from yellow “Standing by” to green “Successful”, the record (or all records) is removed from the table and the librarian is redirected to [Pending register] screen. 8. If the [Add borrower] request is rejected the [status] label will change from yellow “Standing by” to red “Failed” then an error message will show up to inform the librarian and he or she can recheck the records. | | |
| Exceptions: | * In step 5, if the computer which sends the request suffers from lost network connection an error message will show up to inform the librarian. * In step 2, if librarian clicks on [Reject] button the record will be removed from the table without being added to the database. * In step 2, if librarian clicks on [Cancel] button he/she will be redirected to [Search] screen. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | MTN010 | | |
| Use case name: | Update borrower | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | Modify an existed borrower information (including name, card number, valid from, valid until, reader group, level, faculty, office address, Email, living address, mobile phone, lending) in the database. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | The record which librarian wants to modify is updated. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Search] screen. 2. Librarian searches for the borrower. 3. Librarian double-clicks on the borrower record he or she wants to modify. 4. [Update Borrower Information] screen shows up. 5. Librarian modifies the information then clicks [Update] button. 6. This data is passed to DBKeeper module. 7. DBKeeper queries the database. 8. DBKeeper module accepts or rejects the request. 9. If the [Update] request is accepted the [status] label will change from yellow “Standing by” to green “Successful” and the librarian is redirected to Search screen. 10. If the [Update] request is rejected the [status] label will change from yellow “Standing by” to red “Failed” then an error message will show up to inform the librarian and he or she can reenter the information. | | |
| Exceptions: | * In step 7, if the computer which sends the request suffers from lost network connection an error message will show up to inform the librarian. * In step 5, if librarian provides wrong data an error message will show up to inform his or her. * In step 5, if librarian provides insufficient data a message will appear at the top of the window and all required fields will be mark red. | | |
| Special Requirements: | Same as use-case ID [MNT009](#MNT009_Exception). | | |
| Notes and issues: | Same as use-case ID [MNT009](#MNT009_SpecialRequirements). | | |
| Use Case ID: | MTN011 | | |
| Use case name: | Remove borrower | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarians | | |
| Goal/Description: | Remove borrower from the system. | | |
| Pre-conditions: | * To be logged in the system. * To have local network access through a computer. | | |
| Post-conditions: | The record corresponding to the title which librarian wants to delete is removed from the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Search] screen. 2. Librarian searches for the borrower ID. 3. Librarian double-clicks on the borrower record he or she wants to delete. 4. The [Delete Borrower] screen shows up. 5. Librarian clicks on [Delete] button, a warning message will appear to get him/her confirm his/her decision. 6. A [Delete] message is sent to the DBKeeper. 7. The DBKeeper queries the database. 8. Librarian is redirected to the Search screen. | | |
| Exceptions: | * In step 2, if no matched borrower ID is found an error message will appear to inform the librarian. * In step 7, if the computer suffers from lost network connection an error message will appear to inform the librarian. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |

##### Domain business



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - Specification | | | |
| Use Case ID: | DMB001 | | |
| Use case name: | Return of item | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarian | | |
| Goal/Description: | Borrowers return the borrowed items to library. | | |
| Pre-conditions: | * + To be logged in the system.   + To have local network access through a computer. | | |
| Post-conditions: | The lending records corresponding to the returned items is removed from the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Borrower brings items which he/she wants to return to librarian. 2. Librarian switches to [Return item] screen. 3. Librarian enters the item’s code sequently. 4. Librarian clicks on [Return item] button. 5. All information is passed to the DBKeeper module. 6. The DBKeeper module queries the database. 7. If the lending record of an item is successfully removed from the database the background of the text field corresponding to that item will goes green. 8. If the lending record of an item cannot be removed from the database the background of the text field corresponding to that item will goes red and an error message will appear to inform the librarian. 9. Whether or not the return of the items is successful librarian will stay at the [Return item] screen. | | |
| Exceptions: | * + In step 6, if the computer which sends the request suffers from lost network connection an error message will appear to inform librarian.   + In step 4, if librarian clicks on [Cancel] button he/she will be redirected to the [Search] screen.   + In step 3, if librarian enters nothing on any text field then LibMan ignores those text fields. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | The content of error messages will be decided by developers. | | |
| Use Case ID: | DMB002 | | |
| Use case name: | Lend item | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarian | | |
| Goal/Description: | Borrowers want to lend an item from the library. | | |
| Pre-conditions: | * + To be logged in the system.   + To have network access through a computer. | | |
| Post-conditions: | Lending records corresponding to the borrowed items are added to the database. | | |
| Priority: | High | | |
| Activity step description: | 1. Librarian retrieves library card ID. 2. Librarian switches to [Search] screen and search for card ID. 3. Librarian clicks on [Add lent item] button. 4. [Lend item] screen shows up. 5. Librarian enters the code of the items he/she wants to lend to borrower sequently in the text fields. 6. Librarian clicks on [Lend] button. 7. All information is passed to the DBKeeper module. 8. DBKeeper module queries database. 9. If the lending record of an item is successfully inserted to the database the background of the text field corresponding to that item will goes green. 10. If the lending record of an item cannot be inserted to the database the background of the text field corresponding to that item will goes red, an error message will appear to inform the librarian. 11. Whether or not the lending of the items is successful librarian will stay at the [Lend item] screen. | | |
| Exceptions: | * + In step 2, if the card ID cannot be found in the database an error message will appear to inform librarian.   + In step 8, if the computer sending the request suffers from lost network connection an error message will appear to inform librarian.   + In step 6, if librarian clicks on [Cancel] button he/she will be redirected to the [Search] screen.   + In step 5, if librarian enters nothing in any text field LibMan will ignore those text fields. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | The content of error messages will be decided by developers. | | |
| Use Case ID: | DMB003 | | |
| Use case name: | Renew item | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarian | | |
| Goal/Description: | Borrower wants to keep a borrowed item for another 2 weeks. | | |
| Pre-conditions: | * + To be logged in the system.   + To have local network connection through a computer. | | |
| Post-conditions: | The item’s due date is increased by 2 weeks. | | |
| Priority: | High | | |
| Activity step description: | 1. Librarian retrieves library card ID. 2. Librarian switches to [Search] screen and search for the card ID. 3. Librarian click on [Renew] button corresponding to the item he/she wants to renew. 4. A [Renew] message is sent to DBKeeper module. 5. DBKeeper module queries the database. 6. DBKeeper accepts or rejects the request. 7. If DBKeeper accepts the request an information message will appears to tell the librarian the date at which the item must be returned to the library and librarian will stay at the [Search] screen. 8. If DBKeeper rejects the request an error message will appear to inform the librarian and he/she will stay at the [Search] screen. | | |
| Exceptions: | * + In step 2, if the card ID cannot be found in the database an error message will appear to inform librarian.   + In step 5, if the computer sending the request suffers from lost network connection an error message will appear to inform librarian. | | |
| Special Requirements: | * Each item can only be renewed 4 times. | | |
| Notes and issues: | N/A | | |
| Use Case ID: | DMB004 | | |
| Use case name: | Unlock library card | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Librarian | | |
| Goal/Description: | Continue to provide library services to a borrower. | | |
| Pre-conditions: | * To be logged in the system * To have local network access through a computer. | | |
| Post-conditions: | Library card status is changed from locked to available. | | |
| Priority: | High | | |
| Activity step description: | 1. Librarian retrieves library card ID. 2. Librarian switches to [Search] screen and search for the card ID. 3. Librarian click on [Unlock] button corresponding to the item he/she wants to renew. 4. A [Renew] message is sent to DBKeeper module. 5. DBKeeper module queries the database. 6. DBKeeper accepts or rejects the request. 7. If DBKeeper accepts the request the [status] label will change from red [locked] to green [available] and librarian will stay at the [Search] screen. 8. If DBKeeper rejects the request the [status] label will remain red [locked], an error message will appear to inform librarian and he/she will stay at the [Search] screen. | | |
| Exceptions: | * + In step 2, if the card ID cannot be found in the database an error message will appear to inform librarian.   + In step 5, if the computer sending the request suffers from lost network connection an error message will appear to inform librarian. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | DMB005 | | |
| Use case name: | Lock library card | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | LibMan | | |
| Goal/Description: | Stop providing any library services to a borrower. | | |
| Pre-conditions: | * + Sever machine is turned on. | | |
| Post-conditions: | Library card statuses are changed from available to locked in the database. | | |
| Priority: | High | | |
| Activity step description: | 1. LibMan retrieves library card ID list. 2. LibMan sends the [lockCard (IDList)] message to the DBKeeper. 3. DBKeeper queries the database. 4. DBKeeper accepts or rejects the request. 5. If DBKeeper accepts the request a warning message will be send to the client side to inform the corresponding borrower. 6. If DBKeeper rejects the request the cause will be written into the system log file and librarian will be informed on the next login. | | |
| Exceptions: | * + In step 5, if the computer sending the request suffers from lost network connection an error message will be written into system log file and librarian will be informed on the next login. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | DMB006 | | |
| Use case name: | Check overdue | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | LibMan | | |
| Goal/Description: | Check whether all lent item are overdue. | | |
| Pre-conditions: | * + Server machine is turned on.   + The current time is between 00:00 and 01:00 AM. | | |
| Post-conditions: | All library cards corresponding to overdue items will be locked in the system. | | |
| Priority: | High | | |
| Activity step description: | 1. At 00:00 AM the Examiner module sends [getOverdueList()] message to the DBKeeper module. 2. DBKeeper module queries the database. 3. DBKeeper returns the list of the overdue library card IDs which have overdue items to the Examiner module. 4. The Examiner module passes the list to the use-case DMB005. | | |
| Exceptions: | * + In step 2, if the computer sending the request suffers from lost network connection an error message will be written in the system log file and librarian will be informed on the next login. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |

### Client side



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - Specification | | | |
| Use Case ID: | CLT001 | | |
| Use case name: | Register | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Borrowers | | |
| Goal/Description: | A new borrower must register their information (including name, card number, valid from, valid until, reader group, level, faculty, office address, Email, living address, mobile phone, lending) to the library to start using library service. | | |
| Pre-conditions: | * + To have internet connection through a computer. | | |
| Post-conditions: | A register request is sent to the server side. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Register] screen. 2. Borrower provides required information and clicks [Register] button. 3. An object containing the register information is created. 4. This object is sent to the server side through the Internet. 5. Borrower is redirected to [Login] screen for borrowers. | | |
| Exceptions: | * + In step 4, if the computer which sends the object suffers from lost internet connection an error message will show up to inform the borrower. * In step 2, if librarian provides wrong data an error message will show up to inform his or her. * In step 2, if librarian provides insufficient data a message will appear at the top of the window and all required fields will be mark red. | | |
| Special Requirements: | Information of the borrower should satisfy the following conditions:   * **Name**: a string from 1 to 256 characters in length, can only contain the following characters: a … z, A … Z, [space]. Cannot be null. * **Card number**: A string of 7 characters in length consist of 2 upper-case letter followed by 4 digits (Ex: SE02769). Cannot be null. * **Valid from**: A Date variable contains the day from which the card is valid. * **Valid until**: A Date variable contains the day to which the card is invalid. * **Reader group**: One element from the following list (cannot be null):   + - FU HN student     - FU HCM student     - FU DN student     - FU HN faculty     - FU HCM faculty     - FU DN faculty     - Other * **Level**: One element from the following list (cannot be null):   + - Primary school     - Secondary school     - Junior high school     - University/college     - Doctor     - Professor     - Master     - Other * **Faculty**: One element from the following list (cannot be null):   + - Researcher     - IT     - IT Professor     - Translator     - Examiner     - Electronic engineer     - English lecturer     - Japanese lecturer     - Student * **Office address**: A string from 1 to 256 characters, can contain any kind of characters. * **Email**: A string from 1 to 244 characters in length, can contain any kind of characters, followed by “@fpt.edu.vn”. (Ex: hieucdSE02769@fpt.edu.vn). Cannot be null. * **Living address**: A string from 1 to 256 characters in length, can contain any kind of characters. * **Mobile phone**: A string consists of 10 digits. Cannot be null. * **Lending**: An object indicates whether or not the borrower is borrowing items and how many items did he/she borrow. | | |
| Notes and issues: | The fields which don’t say “cannot be null” can be leave empty by default. | | |
| Use Case ID: | CLT002 | | |
| Use case name: | User login | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Borrower | | |
| Goal/Description: | Borrower provides his or her access credentials in order to use the LibMan system. | | |
| Pre-conditions: | * To be pre-registered in the LibMan system * To have internet access through a computer. | | |
| Post-conditions: | A connection to server is established. | | |
| Priority: | High | | |
| Activity step description: | 1. Retrieve [Login] screen for borrower. 2. Borrower provides access information (user name and password) 3. The borrower information is then passed to the SystemKeeper module. 4. The SystemKeeper module queries the LibMan database. 5. The SystemKeeper accepts or rejects the login request. 6. If the login request is accepted borrower will be redirected to the [Search] screen. 7. If the login request is rejected an error message will show up and borrower can provide his or her access credential again. | | |
| Exceptions: | * In step 2, after 5 attempts of unsuccessful login, the account will be locked for 5 minutes. The only way to unlock it is to wait for 5 minutes. * In step 4, if LibMan cannot establish a connection to the database due to some reasons (ex: network failure) an error message will show up to tell the borrower about the problem. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | The content of the error messages will be decided by developers. | | |
| Use Case ID: | CLT003 | | |
| Use case name: | User logout | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Borrowers | | |
| Goal/Description: | When borrowers finish their work they will logout of the system. | | |
| Pre-conditions: | * + To be logged in the system.   + To have internet access through a computer. | | |
| Post-conditions: | The connection to the server is terminated. | | |
| Priority: | High. | | |
| Activity step description: | 1. Borrower clicks [log out] button. 2. A logout request is sent to SystemKeeper module. 3. SystemKeeper module terminates the connection to the database. 4. Borrower is redirected to [Login] screen for borrowers. | | |
| Exceptions: | N/A | | |
| Special Requirements: | * If the logged in computer suffers from lost internet connection it will automatically log out. * If the system is crashed or is forced to close unexpectedly the logged in user will be logged out. | | |
| Notes and issues: | N/A | | |
| Use Case ID: | CLT004 | | |
| Use case name: | Browse title | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Borrowers | | |
| Goal/Description: | Borrower wants the system to list all titles related to a specific keyword. | | |
| Pre-conditions: | * To be logged in the system. * To have internet access through a computer. | | |
| Post-conditions: | All titles related to the keyword will appear on screen. | | |
| Priority: | High | | |
| Activity step description: | 1. Borrower switches to [Search] screen. 2. Borrower provides a keyword and clicks [Go] button. 3. An object containing data is created. 4. This object is sent to server side through the Internet. 5. Server side will sent an object containing all titles’ information related to the keyword in response to client request. 6. Client side computer analyzed the received object and show all information on the screen. | | |
| Exceptions: | * + In step 4, if the computer which sends the object suffers from lost internet connection an error message will show up to inform the borrower. * In step 2, if librarian provides nothing an information message will appear to ask him/her provide a keyword. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |
| Use Case ID: | CLT005 | | |
| Use case name: | View card information | **Version:** | 1.0 |
| Created by: | HieuCD | **Last updated by:** | HieuCD |
| Date created: | 07/03/2013 | **Date last updated:** | 07/03/2013 |
| Role: | Borrowers | | |
| Goal/Description: | Borrower wants to view his/her card information. | | |
| Pre-conditions: | * To be logged in the system. * To have internet connection through a computer. | | |
| Post-conditions: | The card information of the current logged in borrower will appear on the screen. | | |
| Priority: | Low | | |
| Activity step description: | 1. When borrower logins successfully the card information will automatically be sent to the client side via internet. 2. Borrower switches to [Info] screen. | | |
| Exceptions: | * + In step 1, if the computer which sends the request suffers from lost internet connection an error message will show up to inform the borrower. | | |
| Special Requirements: | N/A | | |
| Notes and issues: | N/A | | |

## Software System Attributes

#### Usability

* It takes normal user 3 hours to fully operate the system with the error rate of 3% per hours.
* It takes powered user 1 hour to fully operate the system with the error rate of less than 1% per hour.

#### Reliability

* System can be available 24/7 and can support a large number of users.
* When system failed it would took 30 minutes to completely restart and fully operate again.
* All float number will be rounded down to 3 digits after the point.
* The maximum defect rate is about 20 – 25 bugs per 1000 lines of code.
* The software development phase will follow the Project Coding Convention which is designed to help the team create explicit source code.
* Besides, the development will follow the Object-Oriented Paradigm. This will lead to easily maintenance and upgrading of the application in the future.

#### Performance

The system screens which process and store data on the server can response in 3 second. When the client synchronize or update data which the server, information service, the performance will depend on the network coverage’s condition. The system will be optimize to reduce bandwidths and has the timeout value of 120 seconds.

#### Interfaces

###### User Interfaces

A graphical user interface will be available in all workflow scenarios to allow easy access to all features of the system. The navigation options in every screen will be similar in order to help user can reduce the time learning and using system. All error occurring and exception handling will be catch and display for user with friendly messages.

###### Hardware Interfaces

The prototype software will be implemented on PC or Laptop without additional hardware components:

* + - Internet connection
    - Local TCP/IP network
    - Wi-Fi: Wi-Fi 802.11 b/g/n[[1]](#footnote-1)

###### Software Interfaces

Computer: Windows XP or later with Java Virtual Machine.

###### Communications Interfaces

The client side uses HTTP1.0 (or over) connection to communicate with server over the Internet.

1. IEEE 802.11 http://en.wikipedia.org/wiki/IEEE\_802.11 [↑](#footnote-ref-1)