|  |
| --- |
| Software requirement specification |
| Library system |
| Version 1.0 |
| Prepared by Class SE0715, Group 1  Nguyen Anh Tuan  Huynh Thi Nhung  ….  September 10, 2013 |

Contents

[Revision History 4](#_Toc366535016)

[1. Introduction 5](#_Toc366535017)

[1.1. Purpose 5](#_Toc366535018)

[1.2. Reading suggestion 5](#_Toc366535019)

[1.3. Project Scope 5](#_Toc366535020)

[2. Overall Description 6](#_Toc366535021)

[2.1. Product Perspective 6](#_Toc366535022)

[2.2. User Classes and Characteristics 6](#_Toc366535023)

[2.3. Operating Environment 6](#_Toc366535024)

[2.4. Design and Implementation Constraints 6](#_Toc366535025)

[2.5. User Documentation 7](#_Toc366535026)

[2.6. Assumptions and Dependencies 7](#_Toc366535027)

[3. System Feature 7](#_Toc366535028)

[3.1. Register Book 7](#_Toc366535029)

[3.1.1. Description and Priority 7](#_Toc366535030)

[3.1.2. Stimulus/Response Sequences 7](#_Toc366535031)

[3.1.3. Functional Requirements 8](#_Toc366535032)

[3.2. Register Copy 8](#_Toc366535033)

[3.2.1. Description and Priority 8](#_Toc366535034)

[3.2.2. Stimulus/Response Sequences 8](#_Toc366535035)

[3.2.3. Functional Requirements 9](#_Toc366535036)

[3.3. Register Member 9](#_Toc366535037)

[3.3.1. Description and Priority 9](#_Toc366535038)

[3.3.2. Stimulus/Response Sequences 9](#_Toc366535039)

[3.3.3. Functional Requirements 10](#_Toc366535040)

[3.4. Borrow book(s) 10](#_Toc366535041)

[3.4.1. Description and Priority 10](#_Toc366535042)

[3.4.2. Stimulus/Response Sequences 10](#_Toc366535043)

[3.4.3. Functional Requirements 11](#_Toc366535044)

[3.5. Return Book(s) 11](#_Toc366535045)

[3.5.1. Description and Priority 11](#_Toc366535046)

[3.5.2. Stimulus/Response Sequences 11](#_Toc366535047)

[3.5.3. Functional Requirements 12](#_Toc366535048)

[3.6. Reserve Book 12](#_Toc366535049)

[3.6.1. Description and Priority 12](#_Toc366535050)

[3.6.2. Stimulus/Response Sequences 12](#_Toc366535051)

[3.6.3. Functional Requirements 12](#_Toc366535052)

[4. External Interface Requirements 13](#_Toc366535053)

[4.1. User Interfaces 13](#_Toc366535054)

[4.2. Hardware Interfaces 13](#_Toc366535055)

[4.3. Software Interfaces 13](#_Toc366535056)

[4.4. Communications Interfaces 13](#_Toc366535057)

[5. Other Nonfunctional Requirements 13](#_Toc366535058)

[5.1. Performance Requirements 13](#_Toc366535059)

[5.2. Safety Requirements 13](#_Toc366535060)

[5.3. Security Requirements 14](#_Toc366535061)

[5.4. Software Quality Attributes 14](#_Toc366535062)

[Appendix A: Data Dictionary and Data Model 14](#_Toc366535063)

[Appendix B: Analysis Models 16](#_Toc366535064)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Authors | Modifications | Version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

This SRS describes the software functional and nonfunctional requirements for release 1.0 of the Library System. This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are high priority and committed for release 1.0.

## Reading suggestion

This requirement document contains general information about Library system, the main classes and use cases, functions, features and special technologies. It describes in detail all that Library system needs to work properly and with safety.

The document is divided into sections for better understanding.

* In section 2 an overall description of Library system is provided. First product perspective is presented with product features and main functions. Then follow user classes and characteristics, operating environments that Library system supports as well as design and implementation constraints. After all that user documentation is presented and will provide you with more details about each feature’s technology.
* In section 3 most important features are presented with detailed description, use cases and requirements.
* In section 4 user and communication interfaces are described.
* In section 5 requirements about safety and performance are presented.

## Project Scope

This project is the final project of the Course C#. The project will create the new Library system that allows the registered members to borrow, return and reserve books in Library room. It will be implemented by the Team X of 5 students.

# Overall Description

## Product Perspective

The Library system is the new system that allows registered member to borrow, return, and reserve books in the Library room. The use case diagram that describes external entities and system interfaces for the Library system 1.0 is shown in the Figure 1.

Library System

Librarian

Borrower

Figure 1. Use case diagram of Library system

## User Classes and Characteristics

Librarian: The person is working in the Library room. The librarian will be trained in using the system. The librarian will register book, borrower. The librarian will use the system to implement borrowing, returning, or reserving book(s) for the borrower.

Borrower: The person uses services of Library system, including borrowing, returning, and reserving book(s).

## Operating Environment

OE-1: The Library system will operate in the server located in Library room

## Design and Implementation Constraints

CO-1: The Programming language is C# using Framework .NET, Visual studio 2010

C0-2: Database is SQL Server 2008

C0-3: The system must be finished at …. Novermber 2013 with the Team of 5 students

## User Documentation

UD-1: User manual will guide briefly how to install and use the Library system

## Assumptions and Dependencies

AS-1: The Library system will be opened for the week days from 8h AM to 6h PM

# System Feature

## Register Book

### Description and Priority

The librarian may also register new books into the Library.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian requests to register book.
2. The system allows the user to enter some book information.
3. The librarian requests to filter.
4. The system displays the list of books that meet the input information.
5. The librarian requests to add a new book.
6. The system allows the user to enter the book information.
7. The librarian requests to add this book.
8. The system generates the book number and save this book.

#### Alternative Data Flows

##### 3.1.2.2.1. Alternative Data Flow 1 (edit book)

5. The librarian selects one book among the list and requests to edit this book.

6. The system displays the selected book and allows the user to edit.

7. The librarian requests to save this book.

8. The system saves this book.

##### 3.1.2.2.2. Alternative Data Flow 2 (delete book)

5. The librarian selects one book among the list and requests to delete this book.

6. The system deletes this book.

### Functional Requirements

REQ-1: Register.Book.Filter: The system shall filter books by the given information display the list of these books.

REQ-2: Register.Book.Add: The system shall generate unique book number for each book added.

REQ-3: Register.Book.Edit: The librarian can select any book in the list and can edit it.

REQ-3: Register.Book.Delete: The librarian can select any book in the list and can delete it.

## Register Copy

### Description and Priority

The librarian may also register a new copy for any existing book into the Library.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian request to register copy.
2. The system allows the user to enter some book information.
3. The librarian enters some book information and requests to filter.
4. The system displays the list of these books.
5. The librarian selects one among this list and requests to view all copies for the selected book.
6. The system displays the list of these copies.
7. The librarian requests to add a new copy.
8. The system allows the user to enter the copy information.
9. The librarian requests to add this copy.
10. The system generates copy number and save this copy.

#### Alternative Data Flows

##### 3.2.2.2.1. Alternative Data Flow 1 (edit copy)

7. The librarian selects one among this list and requests to edit

8. The system allows the librarian to edit.

9. The librarian requests to save this copy.

10. The system saves this copy.

##### 3.2.2.2.2. Alternative Data Flow 2 (delete copy)

7. The librarian selects one among the list and requests to delete.

9. The system deletes this copy.

### Functional Requirements

REQ-1:Register.Copy.List: The system shall list of copies for the given book

REQ-2:Register.Copy.Add: The system shall generate the copy number for each copy added

REQ-3: Register.Copy.Edit: The librarian can select any copy in the list and can edit it.

REQ-4: Register.Copy.Delete: The librarian can select any copy in the list and can delete it.

## Register Member

### Description and Priority

The librarian may also register a new member for any existing book into the Library.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian requests to register member.
2. The system allows the user to enter some member information.
3. The librarian requests to filter.
4. The system filters and displays the list of members.
5. The librarian requests to add new member.
6. The system allows the user to enter the member information.
7. The librarian requests to add this member.
8. The system generates the member code and adds this member.

#### Alternative Data Flows

##### 3.2.2.2.1. Alternative Data Flow 1 (edit member)

5. The librarian selects one among the list and request to edit.

6. The system allows the librarian to edit.

7. The librarian requests to save this member.

8. The system saves this member.

##### 3.2.2.2.1. Alternative Data Flow 2 (delete member)

5. The librarian selects one among this list and request to delete.

6. The system deletes this member.

### Functional Requirements

REQ-1: Register.Member.Filter: The system shall filter members by the given member information and displays this list

REQ-2: Register.Member.Add: The system shall generate the member code for any member added

REQ-3: Register.Member.Edit: The librarian can select any member in the list and can edit this selected member.

REQ-4: Register.Member.Delete: The librarian can select any member in the list and can delete this selected member.

## Borrow book(s)

### Description and Priority

The librarian may also record the book(s) borrowed by the borrower.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian requests to borrow the book.
2. The system allows the user to enter the member information.
3. The librarian enters the member information and requests to check.
4. The system checks this member is valid or not.
5. The librarian enters the copy number given by the borrower and requests to check.
6. The system checks if it is referenced or not. If it is referenced, the borrower can’t borrow it. The system checks the number of copies borrowed by this borrower. If it is equal or greater than 5, the borrower can’t borrow any more. The system checks the reservation list. If the book was reserved by the others or the borrower is not in the first of the reservation list, the borrower can’t borrow it.
7. The librarian requests to borrow this copy.
8. The system adds a borrowing record including borrowed date, due date (after 2 weeks from borrowed date) and updates status of this copy as “Borrowed”. The system also set the status of the borrower’s reservation record as “Available” if the book was reserved by this borrower.

### Functional Requirements

REQ-1: Borrow.Book.IsReferencedCopy: The system shall check if the book’s copy is referenced or not.

REQ-2: Borrow.Book.GetBorrowedCopies: The system shall get the list of copies borrowed by the borrower.

REQ-3: Borrow.Book.GetFirstReservation: The system shall get the first reservation for the given book number.

REQ-4: Borrow.Book.SetReservationStatus: The system shall set the status of the borrower’s reservation record as “Avaliable” if the book is reserved for this borrower.

REQ-5: Borrow.Book.Borrow: The system shall add a borrowing record including borrower code, copy number, borrowed date, and due date.

## Return Book(s)

### Description and Priority

The librarian may also record the book(s) returned by the borrower.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian requests to return the book.
2. The system allows the user to enter the member code and book number.
3. The librarian requests to display borrower’s information.
4. The system searches and displays the member information and the list of copies borrowed by this member.
5. The librarian selects some copies and requests to return.
6. The system calculate fine amount for each returned copy if the return date is greater the due date and displays the total fine amount.
7. The librarian confirms fine amount.
8. The system updates the borrowing record with returned date, fine amount. The system also updates the borrower’s reservation status as “Available” if it was reserved by the borrower.

### Functional Requirements

REQ-1: Return.Book.CalculateFine: The system shall calculate fine amount with 1 USD for each delayed day.

REQ-2: Return.Book.SetReservationStatus: The system shall set the status of the borrower’s reservation record as “Available” if the book was reserved by this borrower.

REQ-3: Return.Book.Return: The system shall update the borrowing record with returned date, fine amount.

## Reserve Book

### Description and Priority

The librarian may also keep the book as reserved for the borrower if no any available copy in the Library.

### Stimulus/Response Sequences

#### Basic Data Flows

1. The librarian requests to reserve the book.

2. The system allows the librarian to enter the member information and book number.

3. The librarian enters the member code, the book number and requests to check.

4. The system checks if the member code is valid and the book is not referenced. The system also checks if there is any available copy of this book in the shelf. The borrower can’t reserve this book if there is any available copy for this book. The system also checks if the borrower reserved any reservation. The borrower can’t reserve if there is any reservation.

5. The librarian requests to reserve this book.

6. The system adds a reservation record including borrower code, book number, date, and status as “Reserved”.

### Functional Requirements

REQ-1: Reserve.Book.IsReferencedCopy: The system shall check if the book’s copy is referenced or not

REQ-2: Reserve.Book.NumberAvaliableCopies: The system shall calculate the number of available copies for the given book.

REQ-2: Reserve.Book.Reserve: The system shall add a new reservation record including borrower code, book number, date, and status as “Reserved”.

# External Interface Requirements

## User Interfaces

UI -1: User interface includes only the main window and various message boxes. The main window consists of the main menu bar with functions Member, Book, Borrow, Return and Reserve. When a function is selected, the main window will be updated for the user to perform the actions.

The Library system as preferred uses MS Windows interfaces.

## Hardware Interfaces

No hardware interfaces have been identified.

## Software Interfaces

No software interfaces have been identified.

## Communications Interfaces

CI-1: Can send email to the members to remind them about their borrowed books that was overdue (for the future, not for version 1.0)

# Other Nonfunctional Requirements

## Performance Requirements

PE-1: The system shall accommodate 10 librarians during the peak usage time window of 8:00 am to 6:00 pm local time, with an estimated average session duration of 8 minutes.

PE-2: Responses to queries shall take no longer than 7 seconds to load onto the screen after the user submits the query.

PE-3: The system shall display confirmation messages to users within 4 seconds after the user submits information to the system.

## Safety Requirements

No safety requirements have been identified.

## Security Requirements

SE-1: The system also permits the librarians who are authorized

## Software Quality Attributes

Availability-1: The Library System shall be available to users on the Library room on the week days from 8h AM to 6h PM, no down time.

# Appendix A: Data Dictionary and Data Model

Book information = Book Number (generated by system)

+ Title

+ Publisher

+ Author

Copy Information = Copy Number (generated by system)

+ Book Number

+ Sequence Number (the sequence number of each copy for the same book)

+ Price

+ Status (A: Available; B: Borrowed; R: Referenced)

Borrower information = Borrower Code

+ Name

+ Sex (F/M)

+ Address

+ Telephone

+ Email

Circulated copy = Borrower code

+ Copy number

+ Borrowed date

+ Due Date

+ Returned Date

+ Fine Amount

Reservation = Borrower code

+ Book number

+ Date

+ Status (A: Available; R: reserved)

Book

Copy

Borrower

Reserve

Borrow/Return

1

0..1

0

0..5

**Figure 2. Data model for Library system (not completed)**

# Appendix B: Analysis Models

Available

Borrowed

Borrow

Return

Referenced

Start

Register

Figure 3. State-transition diagram for copy status.