**LIBRARY MANAGEMENT SYSTEM**

**DOCUMENT**

**BSSE 4th Semester ‘A’**

**Prepared By,**

**Mohsin Ali**

**Muhammad Amir**

**Muhammad Taneer Khan**

**Junaid Ali**

**Asad Ur Rehman**

**Date Created: 24-10-2018**

**DEPARTMENT OF COMPUTER SCIENCE - UBIT**

Abstract

With the advancement of technology, library need to be ready for change and bring the technology to the library in order to retain their customer’s interest in visiting the library. Thus in this situation, library’s system should be improved as well to facilitate librarian in doing their work effectively and efficiently. Library Management System (LMS) in the library environment assists librarian in doing their work. Integration with all departments within the library shows how LMS is important in library environment.

This document describes the project development of Library Management System that was developed to manage the transaction of books, manage the students data and books record more efficiently. It can improve management of books in library.

Table of Contents

1. Introduction 3

1.1 Purpose Of This Document 3

1.2 How To Use This Document 3

1.3 Scope Of The Product 4

1.4 Overview Of The Requirements Document 4

2. General Description 4

2.1 Product Perspective 4

2.2 Product Functions 5

2.3 User Characteristics 5

2.4 General Constraints 5

2.5 Assumptions and dependencies 5

2.6 Methodologies & Techniques 6

2.7 Evaluation Methods 6

2.8 Results of Project 6

2.9 Important Names in Fields 6

2.10 Related Books 6

2.11 Available Resources 7

2.12 Tools of Project 7

3. Specific Requirements 12

3.1 Functional Requirements 12

3.2 Non-Functional Requirements 13

3.3 Interface Requirements 14

3.4 Licensing Requirements 19

4. Conclusion 19

5. References 19

**LIBRARY MANAGEMENT SYSTEM**

1. Introduction

1.1 Purpose of This Document

This is the Software Requirements Specification (SRS) for the LMS (Library Management System). The purpose of this document is to convey Information about the application's requirements, both functional and nonfunctional, to the reader. This document provides:

* A description of the environment in which the application is expected to operate.
* A definition of the application's capabilities.
* A specification of the application's functional and nonfunctional requirements.

The document is intended to serve several groups of audiences:

* First, it is anticipated that the SRS will be used by the **application designers.** Designers will use the information recorded here as thebasis for creating the application's design.
* Second, the **client** for the project, the library manager in our case, is expected to review this document. The SRS will serve to establish a basis for agreement between the client and development team about the functionality to be provided by the application.
* Third, the **application maintainers** will review the document to clarity their understanding of what the application does.

1.2 How to Use This Document

**Types of Reader**

* Stakeholders
* Developers
* Designers
* Testers

**Technical Background Required**

For software requirement analysis document there is no need of technical background.

1.3 Scope of the Product

The purpose of this software development project is to create a new application called: LMS SYSTEM. The Library Management System will be PC-based application, allowing library staff to search for books and library staff members to manage the book inventory and user database. The application will provide the following capabilities:

* Library staff will be able to manage library user accounts including remove, change, and add.
* Library staff will be able to manage the book inventory database including remove, change, and add.
* The application will provide search function of books based Book name or category.

The project's client has determined that this application will provide the following benefits:

* Provide additional flexibility and convenience to the library staff.
* Provide better reliability and security of the library information.
* Provide a more productive environment for the library staff member.
* Reduce the cost of the library operations.
  1. Overview of the Requirements Document

The rest of the SRS examines the specifications of the Library Management System in detail. Section 2 of the SRS presents the general factors that affect the LMS and its requirements, such as user characteristics and general constraints. Section 3 outlines the detailed, specific functional requirement, non-functional requirement and other related requirements.

2. General Description

2.1 Product Perspective

Library Management System is a Windows based application. LMS is used by library staff which is responsible to register a student and perform operations i.e. issue/search books, return books and many more. By LMS System the data of library is stored in database which can be manipulated by library manager and easy to access.

2.2 Product Functions

Functionality of this is

**LIBRARIAN:**

* A librarian can issue a book to the student.
* Can view the List of books available in each category.
* Can take the book returned from students.
* Add books and their information of the books to the database.
* Edit the information of the existing books.
* Can access all the accounts of the students.

2.3 User Characteristics

The two types of user for the LMS System are:

* **Library Manager:**
* Good understanding to library operation
* Responsible for library operation as a whole.
* Responsible for library staff managing
* **Librarian:**
* Good understanding to library operation
* Responsible for library operation.

2.4 General Constraints

LMS System can potentially have more users. So, it is unrealistic to provide training for everyone. Therefore, the system should be designed for easy to use and appropriate error messages for invalid user inputs.

2.5 Assumptions and Dependencies

The Product need following third party products:

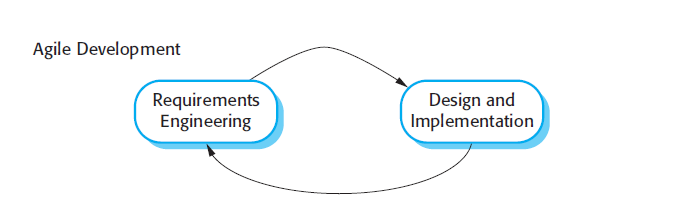
* Microsoft Visual Studio to develop product.
* Microsoft Access to store the database.

Users have basic understanding to PC and Windows and internet.

There is a method to convert all book records and library user records from the existing system into the LMS System.

2.6 Methodologies & Techniques:

For Library Management System we used Agile Development methodology and other techniques.



2.7 Evaluation Methods:

Mixed Methods is used for evaluation of library management system, tutorial based approach and other sources.

2.8 Results of Project:

Library Management gives Add Books, Delete Books, Books Data, Issue Books, Student Registration, and Student Data.

2.9 Important names in field:

* Library Management System Project the major field is Database in which all data of books and students can be stored and can be manipulated it.
* Library Management System Project Database can be kept information about books to be issued and retained in shelf.
* In LMS registration to be modify by person who handle all data.

2.10 Related Books:

* [Lean Library Management: Eleven Strategies for Reducing Costs and Improving Services](https://www.goodreads.com/book/show/10199631-lean-library-management).
* Library Management, Volume 1.

2.11 Resources Available:

* **Libero** [**https://libero.com.au/**](https://libero.com.au/)**:**

LIBERO is the perfect solution to your eResource, learning, and discovery requirements. Customisation options, ongoing product development and LIBERO WebServices ensure the library continues to effectively support the curriculum and the long-term strategy of your institution.

* **ABCD** [**http://abcd.netcat.be/files/**](http://abcd.netcat.be/files/)**:**

ABCD is web-based integrated library management software comprising the main basic library functions. The main characteristics of ABCD are the coverage of the main library functions, its web centrality and its development and maintenance under the methodology of Free and Open Source Software.

* **LibreSource**[**http://dev.libresource.org**](http://dev.libresource.org/)**:**

LibreSource is a versatile collaborative platform. Open Source, modular and highly customizable, LibreSource is adapted to the collaborative software development (forge), groupware, community leading, e-archiving and Web publishing.

* 1. Tools of Project:
     1. **ANALYSIS TOOLS:**

Software analysis includes all activities, which help the transformation of requirement specification into implementation. Requirement specifications specify all functional and non-functional expectations from the software. These requirement specifications come in the shape of human readable and understandable documents.

**OUR PROJECT:**

We used Microsoft Visio in our project for generating Activity Diagram and Use Case Diagram.

* **MICROSOFT VISIO:**

Microsoft Visio is a diagramming and vector graphics application and is part of the Microsoft Office family.

**OTHER TOOLS:**

* **DATA FLOW DIAGRAM:**

Data flow diagram is graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The Data Flow Diagram does not mention anything about how data flows through the system.

* **STRUCTURE CHARTS:**

Structure chart is a chart derived from Data Flow Diagram. It represents the system in more detail than Data Flow Diagram. It breaks down the entire system into lowest functional modules, describes functions and sub-functions of each module of the system to a greater detail than Data Flow Diagram.

Lucid Chart  and Visual Paradigm  are two examples of tools that are used for analysis.

* + 1. **DESIGN TOOLS:**

Software design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation.

**OUR PROJECT:**

We used Microsoft Visio is our project to generate Class Diagram and Sequence Diagram.

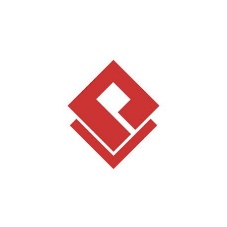
* **MICROSOFT VISIO:**

Microsoft Visio is a diagramming and vector graphics application and is part of the Microsoft Office family.

**OTHER TOOLS:**

* **MAGIC DRAW:**

Magic Draw is a visual UML, SysML, BPMN, and UPDM modeling tool with team collaboration support. Designed for business analysts, software analysts, programmers, and QA engineers, this dynamic and versatile development tool facilitates analysis and design of object oriented systems and databases.

* **VISUAL PARADIGM:**

Visual Paradigm is a UML CASE Tool supporting UML 2, SysML and Business Process Modeling Notation from the Object Management Group. In addition to modeling support, it provides report generation and code engineering capabilities including code generation.

* **PAPYRUS:**

Papyrus can either be used as a standalone tool or as an Eclipse plugin. It provides support for [Domain Specific Languages](https://en.wikipedia.org/wiki/Domain-specific_language) and [SysML](https://en.wikipedia.org/wiki/SysML" \o "SysML). Papyrus is designed to be easily extensible as it is based on the principle of [UML Profiles](https://en.wikipedia.org/wiki/Profile_(UML)).

* + 1. **PROJECT MANAGEMENT TOOLS:**

Project management includes developing a project plan, which involves defining and confirming the project goals and objectives, how they will be achieved, identifying tasks and quantifying the resources needed, and determining budgets and timelines for completion.

**OUR PROJECT:**

We used Microsoft Project for creating Gantt chart, Pert Chart and Timeline in our Project Management.

* **MICROSOFT PROJECT:**

Microsoft Project is a [project management software](https://en.wikipedia.org/wiki/Project_management_software) product, developed and sold by [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It is designed to assist a [project manager](https://en.wikipedia.org/wiki/Project_manager) in developing a [plan](https://en.wikipedia.org/wiki/Plan), assigning [resources](https://en.wikipedia.org/wiki/Resource_(project_management)) to tasks, tracking progress, managing the [budget](https://en.wikipedia.org/wiki/Budget), Gantt charts and analyzing workloads

**OTHER TOOLS:**

* **REDMINE:**

Redmine is a free and open source, web-based project management and issue tracking tool. It allows users to manage multiple projects and associated subprojects. It features per project wikis and forums, time tracking, and flexible, role-based access control.

* **ZOHO OFFICE SUITE:**

Zoho Office Suite is a web-based online office suite containing word processing, spreadsheets, presentations, databases, note-taking, wikis, web conferencing, customer relationship management, project management, invoicing, and other applications.

* **EASY PROJECTS:**

Easy Projects provides functionality common to software designed for [Project Management](https://en.wikipedia.org/wiki/Project_Management) and [Collaboration](https://en.wikipedia.org/wiki/Collaboration) including configurable [Dashboards](https://en.wikipedia.org/wiki/Dashboard), Integrated Reporting, [Message boards](https://en.wikipedia.org/wiki/Message_board), issue and request tracking, statistics, email notifications, and [Gantt charts](https://en.wikipedia.org/wiki/Gantt_chart).

* + 1. **DATABASE MANAGEMENT TOOLS:**

A database management system (DBMS) is system software for creating and managing [databases](https://searchsqlserver.techtarget.com/definition/database). The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage [data](https://searchdatamanagement.techtarget.com/definition/data).

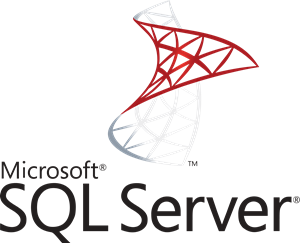
**OUR PROJECT:**

We used Microsoft Access in our project for managing data.

* **MICROSOFT ACCESS:**

Microsoft Access is a database management system from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools.

**OTHER TOOLS:**

* **MICROSOFT SQL SERVER:**

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network.

* **ORACLE DATABASE:**

Oracle Database is a multi-model database management system produced and marketed by Oracle Corporation. It is a database commonly used for running online transaction processing, data warehousing and mixed database workloads.

* + 1. **DOCUMENTATION TOOLS:**

Software documentation is written text or illustration that accompanies [computer software](https://en.wikipedia.org/wiki/Computer_software) or is embedded in the source code. It either explains how it operates or how to use it or may mean different things to people in different roles.

[Documentation](https://en.wikipedia.org/wiki/Documentation) is an important part of [software engineering](https://en.wikipedia.org/wiki/Software_engineering). Types of documentation include:

* [Requirements](https://en.wikipedia.org/wiki/Requirement): Statements that identify attributes, capabilities, characteristics, or qualities of a system. This is the foundation for what will be or has been implemented.
* Design: Overview of software. Includes relations to an environment and construction principles to be used in design of software components.
* Technical: Documentation of code, algorithms, interfaces, and [APIs](https://en.wikipedia.org/wiki/API_documentation).
* [End user](https://en.wikipedia.org/wiki/End_user): Manuals for the end-user, system administrators and support staff.
* Marketing: How to market the product and analysis of the market demand.

**OUR PROJECT:**

We used Microsoft Word for documentation of our project.

* **MICROSOFT WORD:**

Microsoft Word is a [word processor](https://en.wikipedia.org/wiki/Word_processor) developed by [Microsoft](https://en.wikipedia.org/wiki/Microsoft).

**OTHER TOOLS:**



* **MARKDOWNPAD:**

**MarkdownPad** is a full-featured [Markdown](http://markdownpad.com/#whatIsMarkdown) editor for Windows.

* **DROPBOX PAPER:**

Dropbox Paper, or simply Paper, is a [collaborative](https://en.wikipedia.org/wiki/Collaborative) document-editing service developed by [Dropbox](https://en.wikipedia.org/wiki/Dropbox_(service)).

* **READ THE DOCS:**

Read the Docs simplifies software documentation by automating building, versioning, and hosting of your docs for you.

3. Specific Requirements

3.1 Functional Requirements:

**3.1.1 Issuing books:**

* The Library management system can issue books to students.
* All the information of the student is added in the system who is borrowing book(s).
* Date of issue and return can be added in the system, so that books can be kept on track.

**3.1.2 Searching books:**

* Books can be searched from the database and then issued to students.
* Category and name of the book can be searched and the system will notify the row in which it is kept.

**3.1.3 Adding books:**

* New books can be added into the database of the system.
* All the information such as name, quantity, row, shelf etc. are entered.

**3.1.4 Removing books:**

* Unwanted or discontinued books can be removed from the database.

**3.1.5 Retrieving Books Data:**

* Data about the book can be retrieve by simply entering the Name and category of the book.

**3.1.6 Student Data:**

* Data of the student can also be retrieved.
  1. Non-Functional requirements:

**3.2.1 Usability:**

The system should be usable for the operator. The User interface is simple and adaptable for the end user.

**3.2.2 Extensibility:**

The system is extensible. New custom features can be added for a particular system environment.

**3.2.3 Reliability:**

The system can be relied on performance. Minimal bugs and performance lags should be experienced by the user.

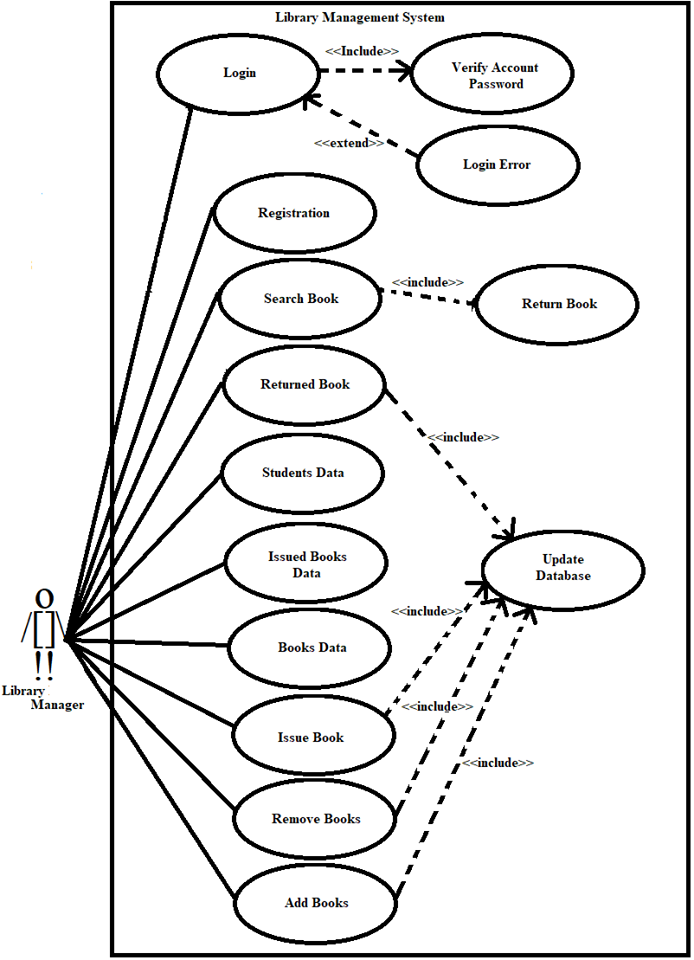
**3.2.4 Privacy and Security:**

The system should be secure. No private information of students should be leaked from the system.

3.3 Interface Requirements

3.3.1 Overview of LMS System Interaction

**USE CASE:**



**CLASS DIAGRAM:**



**SEQUENCE DIAGRAM:**



**ACTIVITY DIAGRAM:**



3.3.2 Database Structure

**Table 3-1: Librarian Login Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Null Value** | **Description** |
| User Name | Long Text | Not Null |  |
| Password | Long Text | Not Null |  |

**Table 3-2: Books Data Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Null Value** | **Description** |
| Book Name | Long Text |  |  |
| Category | Long Text |  |  |
| Shelf No | Short Text |  |  |
| Row No | Short Text |  |  |
| Quantity | Long Text |  |  |
| Available Quantity | Long Text |  |  |

**Table 3-3: Student Accounts Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Null Value** | **Description** |
| Student Name | Long Text |  |  |
| Account No | Long Text | Not Null | Primary Key |
| Department | Long Text |  |  |
| Semester / Year | Long Text |  |  |
| Contact No | Long Text |  |  |

**Table 3-4: Issue Book Data Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Null Value** | **Description** |
| Student Name | Long Text |  |  |
| Account No | Long Text | Not Null | Primary Key |
| Department | Long Text |  |  |
| Semester / Year | Long Text |  |  |
| Contact No | Long Text |  |  |
| Book Name | Long Text |  |  |
| Category | Long Text |  |  |
| Shelf No | Short Text |  |  |
| Row No | Short Text |  |  |
| Date Of Issue | Long Text |  |  |
| Date Of Return | Long Text |  |  |

3.4 Licensing Requirements

Library Management System will be available under an End User License Agreement (EULA).

4. Conclusion

Library Management System (LMS) is the most important aspects of the system where it can facilitate the work of librarian and serve the customers well. Librarians needs to enhance their skills in facing new technology to face new transition in the library environment. Library Management System (LMS) provides some positive impacts towards library’s environment and this integrated system should be expanded and enhanced for future use in order to compete with other application system.

5. References

[1] Anthony Cassels, *Information Centre Manager Fera*, Room 11G08, Sand Hutton, York, YO41 1LZ.

[2] [http://repository.sustech.edu/bitstream/handle/123456789/7384/chapter%203.pdf? sequence=6](http://repository.sustech.edu/bitstream/handle/123456789/7384/chapter%203.pdf?%20sequence=6)

[3] IEE STD 830-1998 Standard Recommended Practice For Software Requirement Specification.

[4] H.C. Tao. Software engineering, Beijing: economic science press, pp. 292-311, 2004.

[5] Rachel S. Smith, Senior Interface Designer, CSU Center for Distributed Learning.