

Darshan University

A Project Report on

**“Project Title”**

Under the subject

**Software Engineering (2305CS202)**

MCA, Semester – II

Department of Computer Application

|  |  |
| --- | --- |
| Submitted By | |
| Student Name: | Enrollment No.: |
| Academic Year  (2024-2025) | |
| Internal Guide  Prof. Devangi L. Kotak  Darshan University | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

|  |  |
| --- | --- |
|  | **Department of Computer Application**  **Darshan University** |

**DECLARATION**

We hereby declare that the SRS, submitted along with the **Software Engineering** **(2305CS202)** for entitled **“Project Title”** submitted in partial fulfilment for the Semester-2 of **Master of Computer Application (MCA)** in **Department of Computer Application (DoCA)** Departmentto Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of (**Guide name)** and that no part of any of report has been directly copied from any students’ reports, without providing due reference.

(Student name)

Student’s Signature

Date: \_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | **Department of Computer Application**  **Darshan University** |

**CERTIFICATE**

This is to certify that the SRS on **“Project Title” has** been satisfactorily prepared by **Student name** (**Enrollment**) under my guidance in the fulfillment of the course **Software Engineering (2305CS202)** work during the academic year 2024-2025.

|  |  |  |
| --- | --- | --- |
| Internal Guide  Prof. Devangi L. Kotak  Darshan University |  | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

**Acknowledgement**

I wish to express my sincere gratitude to my project guide Prof. **Guide name** and all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co- ordination in completing this work.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You

**Student Name**

**ABSTRACT**

Library management system is a system which aims in developing a computerized system to maintain all the daily work of library. This system will act as a tool to transfer traditional library into digital library. This project has many features which are generally not available in general library system like facility of user login, books search on one click etc. It also has a facility of admin login through which the admin can monitor the whole system.  This system will be designed with the basic features such as librarian can add/view/update/delete books and students' details in it. It has also a facility where student after logging in their accounts can see list of books issued and its issue date and return date and also they can request the librarian to add new books by filling the book request form. The librarian after logging into his account i.e. admin account can generate various reports such as student report , issue report, teacher report and book report.

Main purpose of this system is to reduce human efforts as much as possible.

# List of Figures

|  |  |  |
| --- | --- | --- |
| **Fig. No.** | **Name** | **Pg.no.** |
| Fig. 2.1.1 | Use case diagram for library management system | 3 |
| Fig. 2.2.1 | Activity diagram for Book Issue | 4 |
| Fig. 2.2.2 | Swimlane diagram for Book Issue | 4 |
| Fig. 2.3.1 | Sequence diagram for Book Issue | 5 |
| Fig. 2.4.1 | State diagram of Book | 6 |
| Fig. 2.4.2 | State diagram for Librarian | 6 |
| Fig. 2.5.1 | Class diagram for Library management system | 7 |
| Fig. 2.6.1 | Context diagram for Library management system | 8 |
| Fig. 2.6.2 | DFD level-1 for Library management system | 8 |
| Fig. 2.6.3 | DFD level-2 for Issue book | 9 |
| Fig. 3.1.1 | Screen-1: Registration Form | 10 |
| Fig. 3.1.2 | Screen-2: Login Form | 11 |
| Fig. 3.1.3 | Screen-3: Add borrower book | 12 |

# List of Tables

|  |  |  |
| --- | --- | --- |
| **Table no.** | **Name** | **Pg.no.** |
| Table 3.1-1 | Screen element of Registration form | 10 |
| Table 3.1-2 | Screen element of Login form | 11 |
| Table 3.1-3 | Screen element of Add borrower book | 12 |
| Table 4.1-1 | Table: Book | 13 |
| Table 4.1-2 | Table: Borrower | 13 |
| Table 4.1-3 | Table: Staff | 13 |
| Table 4.1-4 | Table: Student | 13 |

**TABLE OF CONTENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Chapter No** | **Chapter Title** | | **Page No** |
|  | Acknowledgement | | iii |
|  | Abstract | | iv |
|  | List of Figure | | v |
|  | List of Table | | vi |
|  | Table of Contents | | vii |
| **Chapter 1** | **INTRODUCTION** | | 1 |
|  | 1.1 | Product perspective | 1 |
|  | 1.2 | Product features | 1 |
|  | 1.3 | Functional requirements | 1 |
|  | 1.4 | Non-functional requirements | 2 |
| **Chapter 2** | **DESIGN AND IMPLEMENTATION CONSTRAINTS** | | 3 |
|  | 2.1 | Use case diagram | 3 |
|  | 2.2 | Activity diagram and Swimlane diagram | 4 |
|  | 2.3 | Sequence diagram | 5 |
|  | 2.4 | State diagram | 6 |
|  | 2.5 | Class diagram | 7 |
|  | 2.6 | Data flow diagram | 8 |
|  |  | 2.6.1 Context diagram (level-0) | 8 |
|  |  | 2.6.2 DFD Level-1 | 8 |
|  |  | 2.6.3 DFD Level-1 | 9 |
| **Chapter 3** | **EXTERNAL INTERFACE REQUIREMENT (SCREENS)** | | 10 |
|  | 3.1 | Screen-1: Registration Form | 10 |
|  |  | Screen-2: Login | 11 |
|  |  | Screen-3: Add borrower book | 12 |
| **Chapter 4** | **DATABASE DESIGN** | | 13 |
|  | 4.1 | List of tables | 13 |
|  | **REFERENCES** | | 14 |

# Introduction

## Product perspective

This project is basically updating the manual library system into an internet-based application so that the users can know the details of their account, availability of books etc. It is a multi-user version and can take care of all the fundamental functions of a Library like Cataloguing, Circulation, Accessioning and Housekeeping. It can satisfactorily cater to all the basic functions of a small library.

## Product features

### There are three different users who will be using this product:

* Librarian who will be acting as the administrator.
* Member who will be accessing the library.
* Guest who will request for membership.

### The features that are required for the Librarian are:

* Control the movement of books and other material and avoid losing the same.
* Search if you have a specific book in your collection based on the title, author etc.
* Print the spine labels for the book.
* Find what a specific person has borrowed from you. It offers the following modules Cataloguing, Circulation, and Queries.
* 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 check the report of the issued Books.
* Can access all the accounts of the students.

### The features that are required for the Member are:

* Can view the different categories of books available in the library.
* Can view the List of books available in each category.
* Can own an account in the library.
* Can view the books issued to him.
* Can put a request for a new book.

## Functional Requirement

### **Librarian**

* **Add Article:** New entries must be entered in database
* **Update Article:** Any changes in articles should be updated in case of update3.
* **Delete Article:** Wrong entry must be removed from system
* **Membership management:** To maintain a detailed database of the members. The system records the name, ID and password of each user. The system helps in ascertaining the track record of the member.
* **Bar-coding:** To give specific identification to each book. All books, old and new, are bar-coded on the basis of title, author, topic and date of publishing.
* **Barcode scanning:** To read the barcode easily using RFID sensors. The database is automatically updated when books are scanned while issuing or returning.
* **Search function:** to enable both the librarian and the members to search the catalog of books in the library. The search functions can be filtered to the need of each user.
* **Check out Article:** To issue any article must be checked out
* **Check In article:** After receiving any article system will renter article by Checking
* **Reserve Article:** This use case is used to reserve any book with the name of librarian, it can be pledged.
* **Set user Permission:** From this user case Librarian can give permission categorically, also enabling/disabling of user permission can be set through this use case

### **Member**

* **Authentication:** User must be authenticated before accessing system
* **Search Article:** User can search any article
* **Request Article:** After successful searching member mark this book as requested article
* **Check Account:** This use case is used to check account details

## Non-Functional Requirement

### **Usability:**

* The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

### **Accuracy:**

* The data stored about the books and the fines calculated should be correct, consistent, and reliable.

### **Availability:**

* The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

### **Maintainability:**

* The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

# Design and Implementation Constraints

## Use case diagram

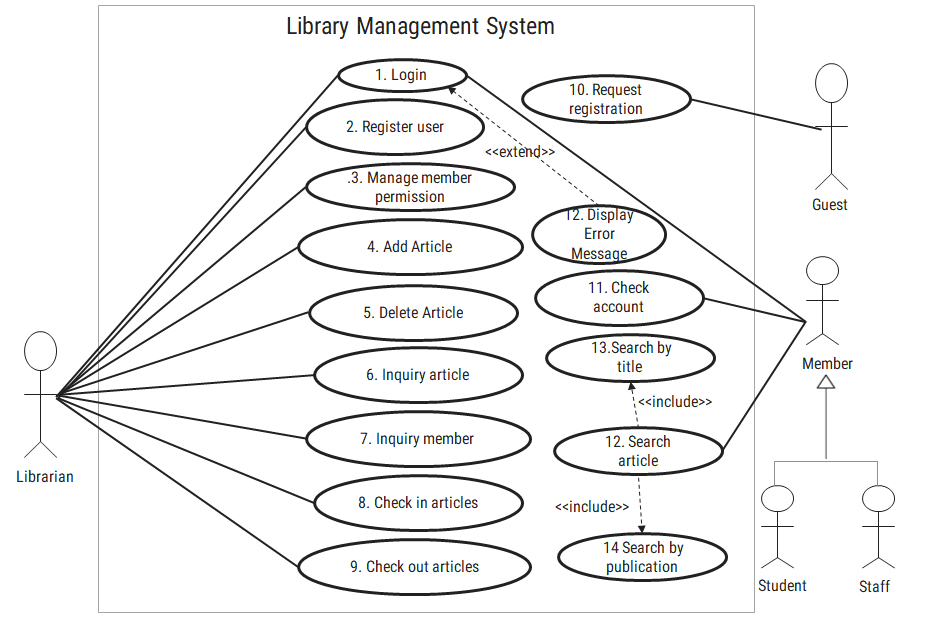


Figure 2.1‑1 Use case diagram for library management system

## Activity diagram and Swimlane diagram

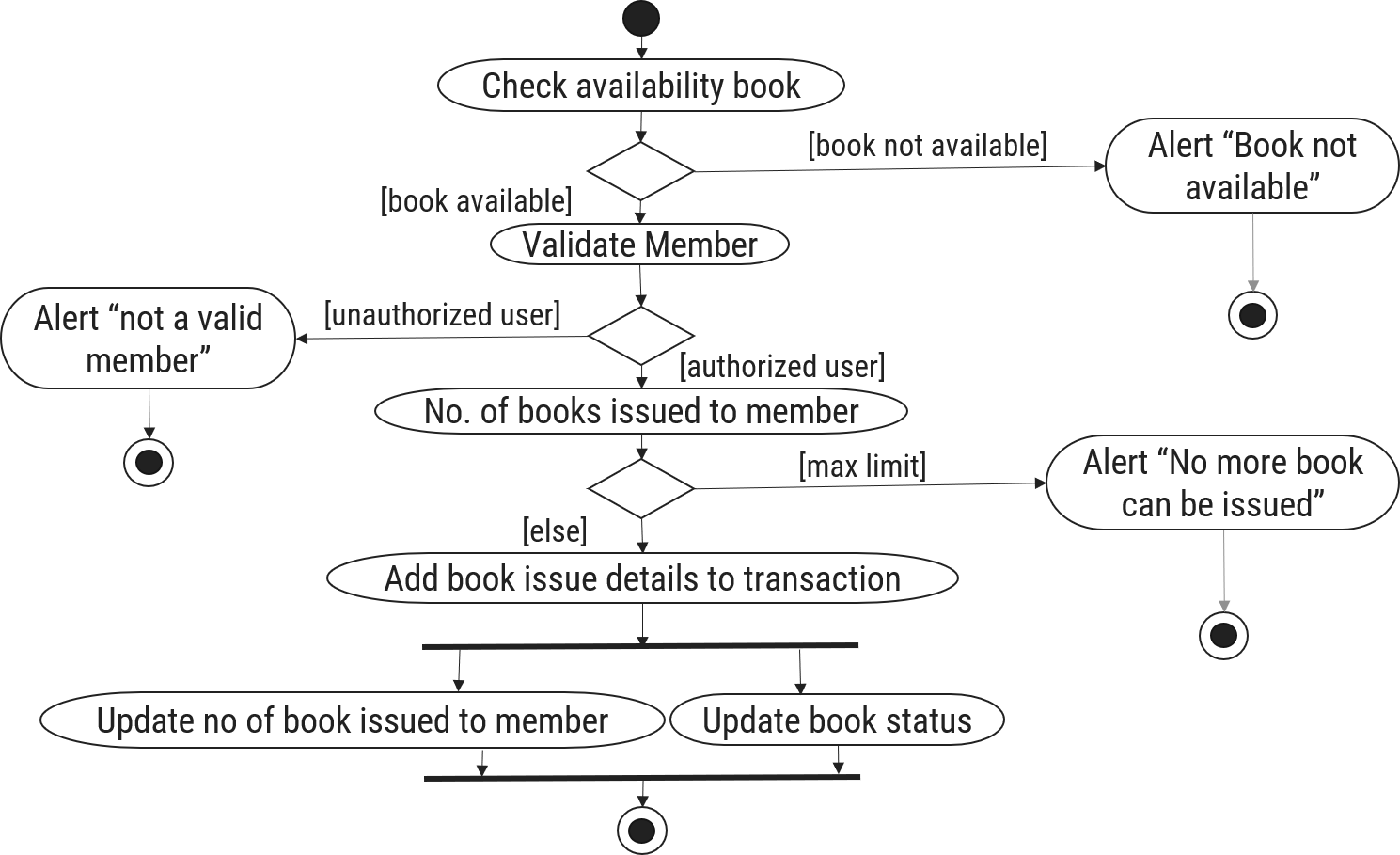


Figure 2.2‑1 Activity diagram for Book Issue

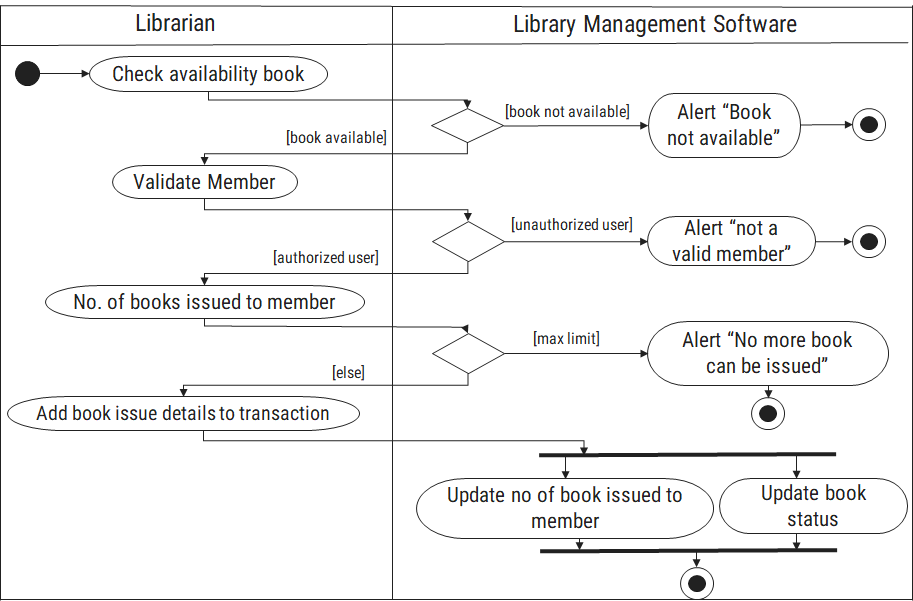


Figure 2.2‑2 Swimlane diagram for Book Issue

## Sequence diagram



Figure 2.3‑1 Sequence diagram for Book Issue

## State diagram



Figure 2.4‑1 State diagram of Book



Figure 2.4‑2 State diagram for Librarian

## Class diagram



Figure 2.5‑1 Class diagram for Library management system

## Data flow diagram

### Context diagram (level-0)



Figure 2.6‑1 Context diagram for Library management system

### DFD Level-1



Figure 2.6‑2 DFD level-1 for Library management system

### DFD Level-2



Figure 2.6‑3 DFD level-2 for Issue book

# External interface requirement (Screens)

## Screen-1: Registration Form



Figure 3.1‑1 Screen-1: Registration Form

**Purpose:** This form will allow the target end-users to register in the system. To register , the following information will be encoded in the system.

Table 3.1‑1 Screen element of Registration form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Username | Textbox | M | 1 | Username field should be editable and accept the Username. |
| 2 | Email | Textbox | M | 1 | Email field should be editable and accept the email with proper format. |
| 3 | Password | Password | M | 1 | Password field should be editable and accept the password and display as star or dot. |
| 4 | Retype password | Password | M | 1 | Retype Password field should be editable and accept the password and display as star or dot. |
| 5 | I agree to the terms | Checkbox | M | 1 | Checkbox for accepting terms and condition. |
| 6 | I already have a membership | Link | ------ | ------ | Link for navigate to login page for registered user. |
| 7 | Register | Button | ------ | ------ | Register is a button for store the entered data into database. |

## Screen-2: Login Form



Figure 3.1‑2 Screen-2: Login Form

**Purpose:** This form will be used by the system’s users to access records and features of the system. The users will input the correct combination of their username and password to be able to login to the system.

Table 3.1‑2 Screen element of Login form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Username | Textbox | M | 1 | Username field should be editable and accept the Username. |
| 2 | Password | Password | M | 1 | Password field should be editable and accept the password and display as star or dot. |
| 3 | Remember Me | Checkbox | M | 1 | Saving login credentials through remember me checkbox |
| 4 | I forgot my password | Link | ------ | ------ | Link for navigate to Forgot password page for allows users to recover password. |
| 5 | Register a new membership | Link | ------ | ------ | Link for navigate to membership registration. |
| 6 | Sign in | Button | ------ | ------ | Login button navigates to another page even if valid login credentials. |

## Screen-3: Add borrower book



Figure 3.1‑3 Screen-3: Add borrower book

**Purpose:** This module will allow the system administrator to add, edit, update or delete borrowers of book. The admin can add borrower information and manage it.

Table 3.1-3 Screen element of Add borrower book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Borrower ID | Textbox | M | 1 | Borrower ID field should be editable and fill autogenerated Borrower ID. |
| 2 | Student ID | Textbox | M | 1 | Student ID field should be editable and accept Student barcode id. |
| 3 | First Name | Textbox | M | 1 | First Name field should be editable and accept only string as a borrower first name. |
| 4 | Middle Name | Textbox | O | 1 | Middle Name field should be editable and accept only string as a borrower middle name. |
| 5 | Last Name | Textbox | M | 1 | Last Name field should be editable and accept only string as a borrower last name. |
| 6 | Contact | Textbox | M | 1 | Contact field should be editable and accept only 10-digit mobile number. |
| 7 | Grade Id | Textbox | O | 1 | Auto fetch detail from the database if proper student id is entered. |
| 8 | Submit | Button | ----- | ----- | Submit is a button for store the entered data into database. |
| 9 | Cancel | Button | ----- | ----- | Cancel is a button for close add borrower dialog. |

# Database design

## List of Tables

* Book
* Borrower
* Student
* Staff

Table 4.1‑1 Table: Book

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BookID | int | NN | PK (Auto Increment) |  |
| BookName | varchar(100) | NN |  |  |
| ISBN | varchar(100) | AN |  |  |
| Publication\_year | int | AN |  |  |
| Language | varchar(50) | AN |  |  |

Table 4.1‑2 Table: Borrower

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value Description |
| BorrowerID | int | NN | PK (Auto Increment) |  |
| BookID | varchar(100) | NN | FK | Reference of Book Table |
| BorrowedFromDate | DateTime | AN |  |  |
| BorrowedToDate | DateTime | AN |  |  |
| ActualReturnDate | DateTime | AN |  |  |
| IssuedBy | int | NN | FK | Reference of Student and Staff Table |

Table 4.1‑3 Table: Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StaffID | int | NN | PK (Auto Increment) |  |
| StaffName | varchar(100) | AN |  |  |
| IsAdmin | Boolean | AN |  |  |
| Designation | varchar(100) | AN |  |  |

Table 4.1‑4 Table: Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StudentID | int | NN | PK (Auto Increment) |  |
| StudentName | varchar(100) | AN |  |  |
| Gender | varchar(100) | AN |  |  |
| DOB | DateTime | AN |  |  |
| Department | varchar(100) | AN |  |  |
| Contact | number(10,0) | AN |  |  |

# References

* http://www.w3schools.com/html/html\_intro.asp
* https://www.w3schools.com/php/default.asp
* https://www.javatpoint.com/uml