Chapter 1

Abstract

The project titled Library Management System is about developing Library management software for monitoring and controlling the transactions in a library. It mainly focuses on basic operations in a library like adding new members, updating new books, and updating new information, searching books and members and facility to borrow and return books. This project of "LIBRARY MANAGEMENT" of gives us the complete information about the library. We can enter the record of new books and retrieve the details of books available in the library. We can issue the books to the students and maintain their records and can also check how many books are issued and stock available in the library. In this project we can maintain the late fine of students who returns the issued books after the due date.

Library Management System is a software application to maintain the records related to Book Purchase, Stock Maintenance, Book Search, Catalogue, Book issue, Book returns, Fine Collection, and all necessary requirements for the Library to manage day to day operations. The main objective of the application is to automate the existing system of manually the records of the Book Issue, Book Return from the student, Stock Maintenance, Catalogue and Book Search to be computerised. So that the Book Issue, Return Searching will be faster. The library management system is designed and developed for a receipt and issuance of books in the library analogue with the student's details.

Chapter 2

Literature Survey

2.1 Initial Investigation

In initial investigation study phase we had undergone through various steps which are described as under:

1. Identify the origin of the information at different level.

2. Identify the expectation of user from computerized system.

3. Analyze the drawback of existing system (manual) system.

St. Peters is a post-graduate college, which offers PG courses in Arts, Science and Management. It has a collection of about 20,000 books in its library. The director of the PG centre wants to automatic library transactions, which have been operated manually so far.

The current procedure is as soon as a book arrives, the details are entered into the stock registers and a manual book ID is attached to identify the books. Separate stock registers are maintained for different streams.

The books issued to students are entered into an issue register, along with the book ID and the date of return. Separate registers are maintained for arts, science and management books. “Details of books that are issued are entered in the issue register manually. This is time consuming and sometimes students have to stand in long queues to get books issued.

The Library Management System (LMS) is the core of any library. It supports all the ‘back-end’ operations of a library: acquiring stock, cataloguing stock, loaning stock and reporting on these functions for effective service management. The LMS is also vital for front-end operations as it enables users to find what items or information a library has and to borrow/access as appropriate their required items.

2.2 Problem Definition

“**Library Management System**” is a software application to maintain the records related to Book Purchase, Stock Maintenance, Book Search, Catalogue, Book issue, Book returns, Fine Collection, and all necessary requirements for the Library to manage day to day operations. The main objective of the application is to automate the existing system of manually the records of the Book Issue, Book Return from the student, Stock Maintenance, Catalogue and Book Search to be computerised. So that the Book Issue, Return Searching will be faster. The library management system is designed and developed for a receipt and issuance of books in the library analogue with the student's details.

2.3 **Evaluation of Existing System**

- In the existing system manual record takes more amount of time to access the books.
- It also need more amount of manpower to manage the things like arranging different types of books.
- It needs more time to keep record of books manually.
- Updating information about books, other study materials and customers take more time and are difficult to maintain the record.
- Tracing a book is very difficult.
- Information about issue/return of the books is not properly maintained.

2.4 Software Selection

Visual Basic.Net

Visual Studio.NET is an environment for developing Windows and Web applications. VisualBasic.NET is just one of the languages you can use to program your applications. Visual Studio .NET was designed to host any language, and many companies are working on languages that will be integrated in Visual Studio .NET. Some people will develop Windows applications in Visual Studio .NET with COBOL, or FORTRAN. So Visual Studio .NET is the environment that provides all the necessary tools for developing applications. The language is only one aspect of a Windows application. The visual interface of the application isn't tied to a specific language, and the same tools you'll use to develop your application's interface will also be used by all programmers, regardless of the language they'll use to code the application. The tools you'll use to access databases are also independent of the language. Visual Studio provides tools that allow you to connect to a database inspect its objects, retrieve the information you're interested in, and even store it in objects that can be accessed from within any language. There are many visual tools in the IDE, like the Menu Designer. This tool allows you to visually design menus and to set their names and basic properties (such as checking, enabling, or disabling certain options). Designing a menu doesn't involve any code, and it's carried out with point-and-click operations. Of course, you will have to insert some code behind the commands of your menus, and (again) you can use any language to program them.

To simplify the process of application development, Visual Studio .NET provides an environment that's common to all languages, which is known as integrated development environment (IDE). The purpose of the IDE is to enable the developer to do as much as possible with visual tools, before writing code. The IDE provides tools for designing, executing, and debugging your applications. It's your second desktop, and you'll be spending most of your productive hours in this environment. At Start-up control, is where you define what you want Visual Studio .NET to do when it starts.

The choices are the following:

- Show Start Page every time you start Visual Studio .NET, this page will appear.
- Load Last Loaded Solution Once you start working on a real project (a project that will take you from a few days to a few months to complete), select this option so that the project will be loaded automatically every time you start Visual Studio .NET.
- Show Open Project Dialog Box every time you start Visual Studio .NET, the Open Project dialog box will appear, where you can select a project to open.
- Show New Project Dialog Box every time you start Visual Studio .NET, the New Project dialog box will appear, where you can specify the name of a new project—a setting to avoid.
- Show Empty Environment This option instructs Visual Studio .NET to start a new empty solution, and you're responsible for adding new or existing projects to the solution and new or existing items to a project.

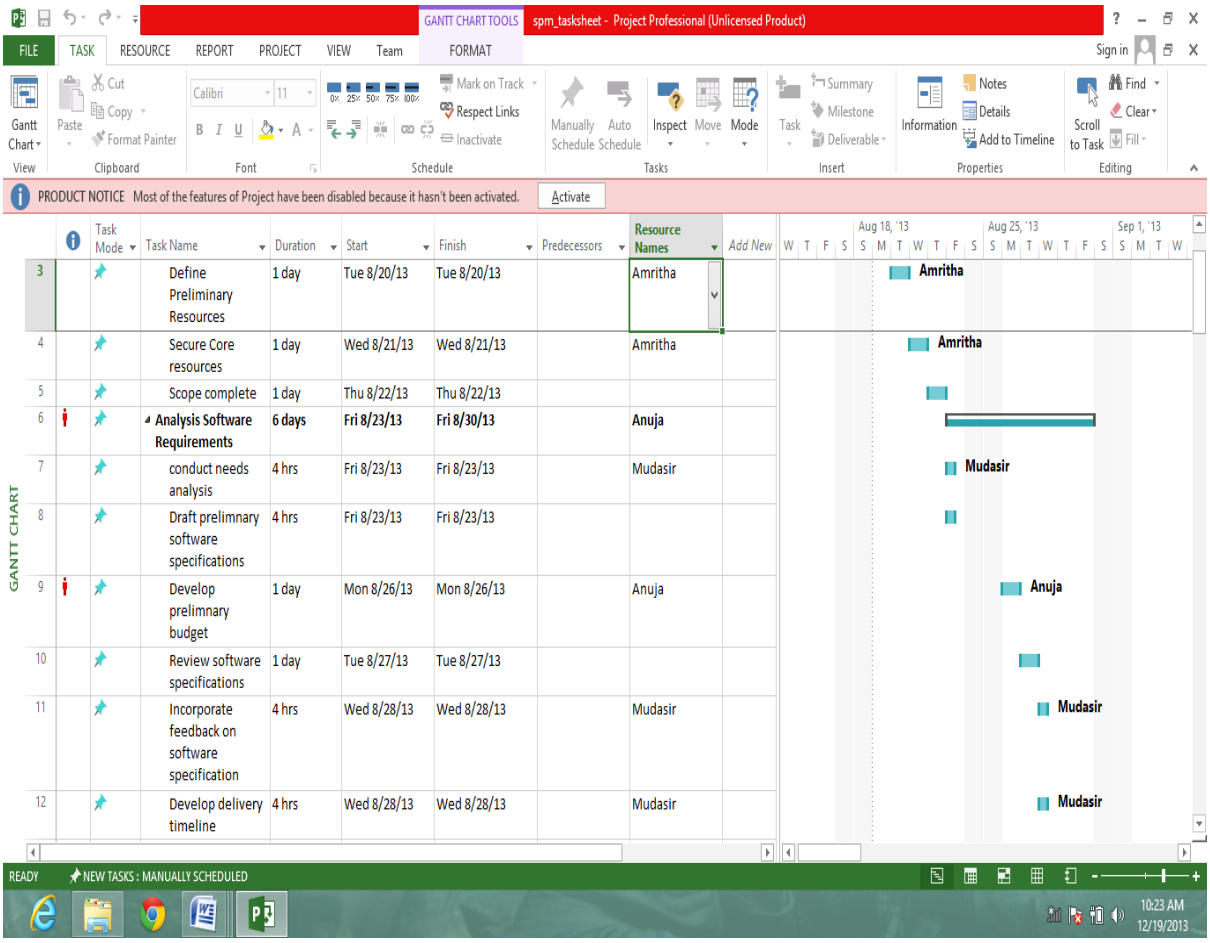
SQL SERVER 2005

- SQL Server 2005 can be used to store information for personal use, for departmental use, for mid-size company use, or for enterprise use. SQL Server 2005 has editions to meet the needs in each of those scenarios:
- **Enterprise:** Provides a relational database to meet the exacting needs of the largest enterprises and busiest online databases. The Enterprise Edition includes high-end business intelligence support and clustering.
- **Workgroup:** Meets the needs of small- to medium-sized businesses that don't require the features of Standard Edition.
- SQL Server 2005 is a client-server database. Typically, the SQL Server 2005 database engine is installed on a server machine to which you connect anything from a few

machines to many hundreds or thousands of client machines. Client-server architecture can handle large amounts of data better than a desktop database such as Microsoft Access. The SQL Server instance provides security, availability, and reliability features that are absent from databases such as Access. Client-server architecture also can reduce network traffic.

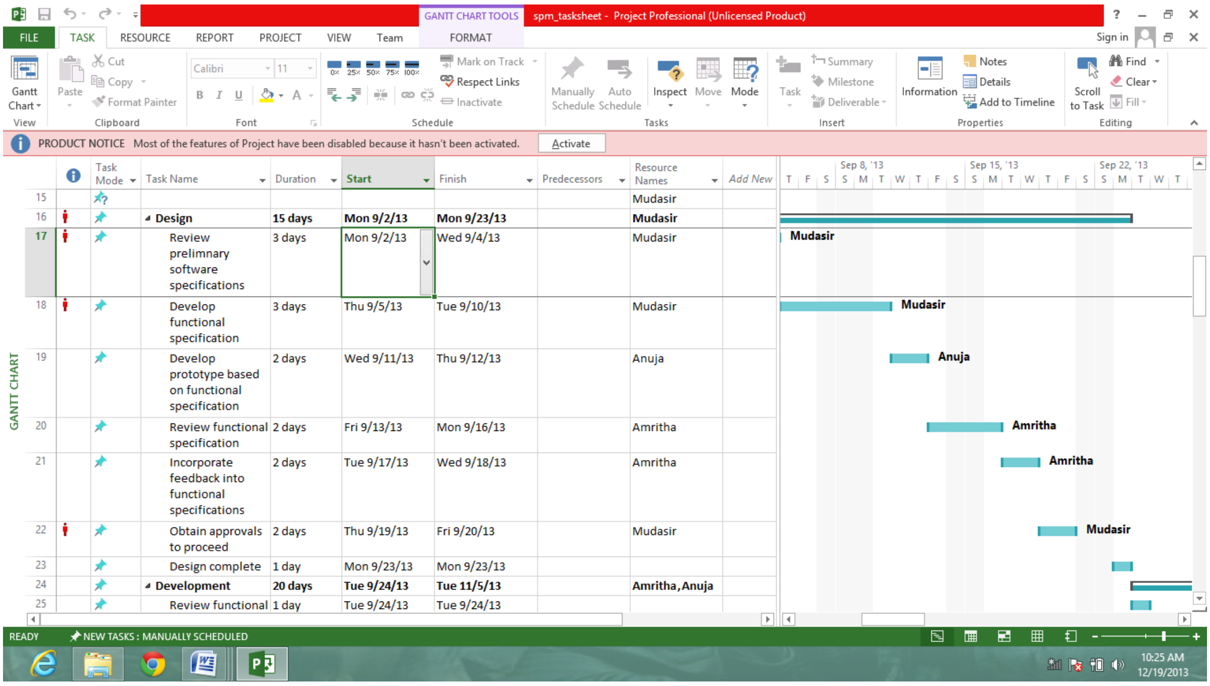
- **Secured Database:** If the data on which your business depends is stored in SQL Server, you need to keep the wrong people from accessing the data or, worse, changing or deleting the data. SQL Server 2005 implements Microsoft's recent emphasis on security. Unlike its predecessor (SQL Server 2000), SQL Server 2005 is much more secure by default.

2.5 Project Plan Using MSP-2013



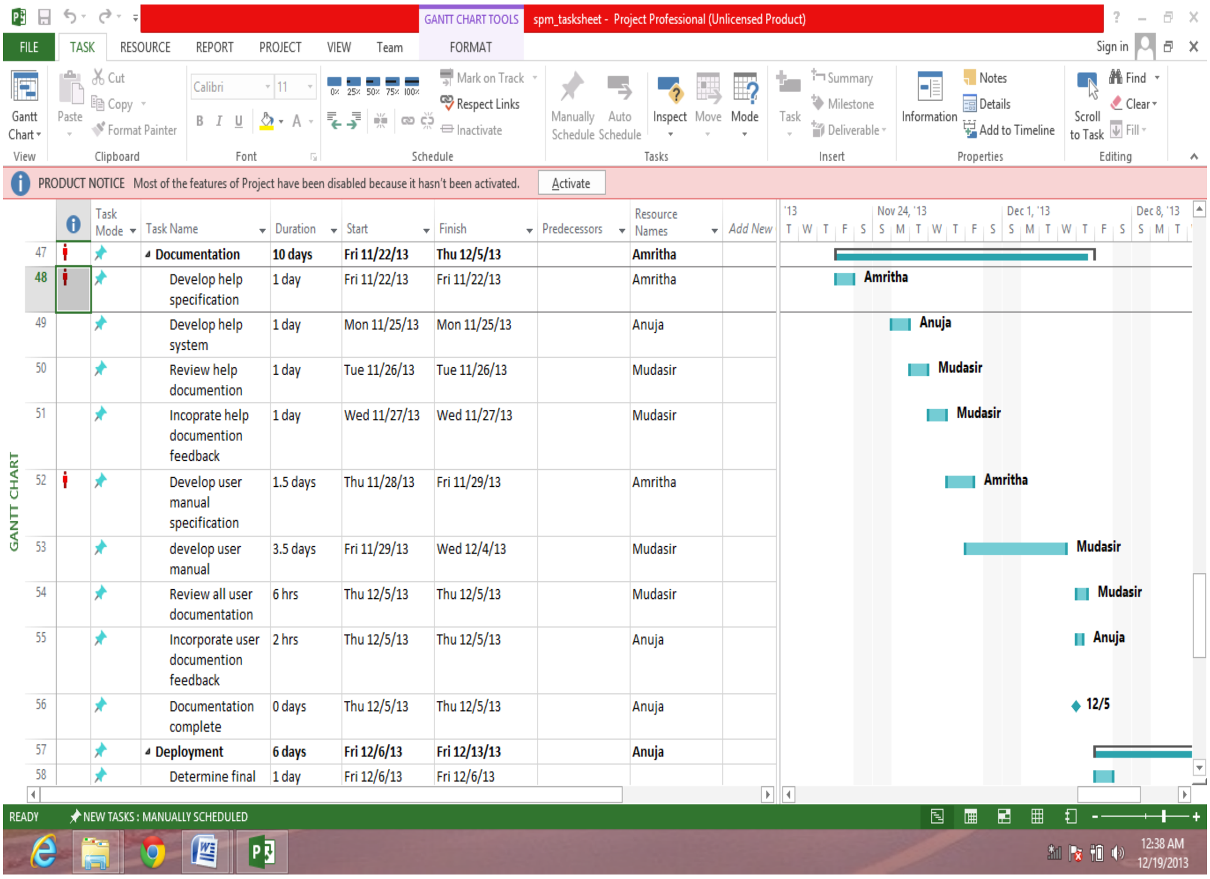
PRODUCT NOTICE Most of the features of Project have been disabled because it hasn't been activated. Activate

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names	Add New
3		Define Preliminary Resources	1 day	Tue 8/20/13	Tue 8/20/13		Amritha	
4		Secure Core resources	1 day	Wed 8/21/13	Wed 8/21/13		Amritha	
5		Scope complete	1 day	Thu 8/22/13	Thu 8/22/13			
6		**Analysis Software Requirements**	**6 days**	**Fri 8/23/13**	**Fri 8/30/13**		**Anuja**	
7		conduct needs analysis	4 hrs	Fri 8/23/13	Fri 8/23/13		Mudasir	
8		Draft prelimnary software specifications	4 hrs	Fri 8/23/13	Fri 8/23/13			
9		Develop prelimnary budget	1 day	Mon 8/26/13	Mon 8/26/13		Anuja	
10		Review software specifications	1 day	Tue 8/27/13	Tue 8/27/13			
11		Incorporate feedback on software specification	4 hrs	Wed 8/28/13	Wed 8/28/13		Mudasir	
12		Develop delivery timeline	4 hrs	Wed 8/28/13	Wed 8/28/13		Mudasir	



PRODUCT NOTICE Most of the features of Project have been disabled because it hasn't been activated. Activate

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names	Add New
15							Mudasir	
16		Design	15 days	Mon 9/2/13	Mon 9/23/13		Mudasir	
17		Review prelimnary software specifications	3 days	Mon 9/2/13	Wed 9/4/13		Mudasir	
18		Develop functional specification	3 days	Thu 9/5/13	Tue 9/10/13		Mudasir	
19		Develop prototype based on functional specification	2 days	Wed 9/11/13	Thu 9/12/13		Anuja	
20		Review functional specification	2 days	Fri 9/13/13	Mon 9/16/13		Amritha	
21		Incorporate feedback into functional specifications	2 days	Tue 9/17/13	Wed 9/18/13		Amritha	
22		Obtain approvals to proceed	2 days	Thu 9/19/13	Fri 9/20/13		Mudasir	
23		Design complete	1 day	Mon 9/23/13	Mon 9/23/13			
24		Development	20 days	Tue 9/24/13	Tue 11/5/13		Amritha,Anuja	
25		Review functional	1 day	Tue 9/24/13	Tue 9/24/13			



PRODUCT NOTICE Most of the features of Project have been disabled because it hasn't been activated. Activate

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names	Add New
47		Documentation	10 days	Fri 11/22/13	Thu 12/5/13		Amritha	
48		Develop help specification	1 day	Fri 11/22/13	Fri 11/22/13		Amritha	
49		Develop help system	1 day	Mon 11/25/13	Mon 11/25/13		Anuja	
50		Review help documention	1 day	Tue 11/26/13	Tue 11/26/13		Mudasir	
51		Incoprate help documention feedback	1 day	Wed 11/27/13	Wed 11/27/13		Mudasir	
52		Develop user manual specification	1.5 days	Thu 11/28/13	Fri 11/29/13		Amritha	
53		develop user manual	3.5 days	Fri 11/29/13	Wed 12/4/13		Mudasir	
54		Review all user documentation	6 hrs	Thu 12/5/13	Thu 12/5/13		Mudasir	
55		Incorporate user documention feedback	2 hrs	Thu 12/5/13	Thu 12/5/13		Anuja	
56		Documentation complete	0 days	Thu 12/5/13	Thu 12/5/13		Anuja	
57		Deployment	6 days	Fri 12/6/13	Fri 12/13/13		Anuja	
58		Determine final	1 day	Fri 12/6/13	Fri 12/6/13			

Chapter 3

Hardware and Software Requirements

The Hardware and software required for development of this project are as follows:

Hardware Requirements

Processor	**:**	Intel Pentium IV or more
Ram	**:**	512 MB or more
Cache	**:**	512 KB
Hard disk	**:**	16 GB hard disk recommended for primary partition.

Software Requirements

Operating system	**:**	Windows XP or later
Front End	**:**	VB.Net implemented in Visual Studio 2008
Back End	**:**	My SQL

- **Front-end tool**
  - The front end design tool, which is used to design the client part of the client server architecture. Clients are otherwise known as front-end
- **Back-end tool:**
  - The back-end design tool is a tool, which is used to design the server part of the client-server architecture. Servers are otherwise known as back-end. They store all data and fulfil the requests made by the client programs or front-end.

Chapter 4

Software Requirement Specification

Software requirement specification is the beginning point of the development activity. Software requirement is one such area, to which a little importance was attached in the early days of Software development as the emphasis was in coding and design. The tacit assumption was that developers understood the problem clearly when it was explained to them. As system grew more complex, it becomes evident that the goals of the entire system couldn't be easily comprehended.

Hence the needs for a rigorous requirement analysis phase arrive. Now for larger software system the requirement analysis is the most difficult and iterative activity.

The software project is initiated by the clients need. These needs are in mind of various people in the client organization. The requirement analyst has to identify the requirement after discussing with the people and understanding their needs. In situations where the software is to automate a currently manual process, most of the needs can be understood by observing the current practice.

The SRS is a means of translating the ideas in the minds of the client into a formal document. Thus, the output of the phase is a set of formally specified requirements, which hopefully are complete and consistent, while the input has none of these properties. Clearly the process of specifying requirements cannot be totally formal. Any formal translation process producing a formula output must have a precise and unambiguous input.

4.1 Software Scope

The *Library Management System* that is to be developed provides the students and employees of the library with books information and many other facilities. The scope of the Library Management System is as follows:

- The system allows the Librarian to create the books catalogue, add/delete books and maintain the books catalogue.
- The system updates the system as and when the student borrows or returns a book.
- The book catalogue is automated and the decision of offering the book based on the category of the book is automatically decided.
- The system automatically intimates about the late submission of books and thereby provides a mechanism to adding a late fine according to the delay caused in returning or renewing the books.

- The system also allows reserving a book for a student who wishes to subscribe it.
- Along with the book details the system will also have the supplier and invoice details.
- It provides "better and efficient" service to members.
- Reduces the workload of employee.
- Faster retrieval of information about the desired book.
- Provide facility for proper monitoring reduces paper work and provide data security.
- All details will be available on a click.

The scope that is described in this document is used in the future phases of the software development cycle. The features described here meet the needs of all the users. The success criteria for the system are based in the level up to which the features described in this document are implemented in the system.

4.2 Product Perspective

The Library Management System is a package to be used by the college library to improve the efficiency of Librarians and Users (students and professors). The Library Management System to be developed benefits greatly the students, professors and the Librarian. The system provides books catalogue and information to the librarian and helps them decide on the books to issue from the library. The Librarian can keep the books catalogue updated all the time so that the members (students and the professors) get the updated information all the time (i.e. about the availability of particular books and their stocks). It also helps librarian to know about which books are due to be renewed or returned so that late fines can be included wherever applicable. The system automatically includes a late fine on delayed renewal and return of books.

4.3 User Classes and Characteristics

The various Classes involved in the system are:

Class: Books, Librarian, User, Publisher, Reference Book, General Book, Book Bank, Student, Faculty.

Here, in this system there could be two types of users: Student and Faculty. Both use to share many of the properties and methods. So, we defined a new class that is user and from it both student and faculty class inherits properties and methods. Hence, User is basically an abstract class whose object directly can't be created.

Similarly, Reference Book, General Book and Book Bank all three classes share many of its attributes so, we generalized all the three classes with super class Books and from it all the other three classes inherits methods and properties. Unlike User class, Books class is not an abstract class. We can create as well as instantiate its objects directly.

4.4 Features of Proposed System

Library is regarded as the brain of any institute; many institutes understand the importance of the library to the growth of the institute and their esteem users (students). LMS support the general requirement of the library like acquisition, cataloguing, circulation.

Library project system that offers many flexible and convenient features, allowing librarians and library users to maximize time and efficiency. Library System gives the all detailed information about students, staff and books. It will track on the how many books available in library and books issued to the students. It shows popular book among the students. It will provide book lost in library. It keeps the record of the suppliers and book binders. It generates MIS reports for management. Our software is customizable for any library requirement.

Features of library management system:

- Only basic knowledge of computers is required for operation of Library Management System. As it has user-friendly application interface.
- Library Management System is Customizable and User Configurable.
- An inbuilt Settings module makes Library Management System flexibility to cater to diverse organizational needs.
- It is build on .NET technology - one of the most latest and upcoming Technologies in the field of Information Technology, which makes you a forerunner in the world of Information technology.
- Library Management System brings information to the user's desktop through integration across all modules.
- Library Management System has pre-defined reports. These are used for normal reporting as well as Administration & Staff development purpose. Additionally, Library Management System can be easily customized for their own customized reports.
- Staff as well as student record is maintained.
- Newspapers attendance is maintained.
- Automatic fine fees calculation.
- Keeps record of supplier’s and binders.
- Customized Report designing.
- Configurable as per user’s requirements.
- It provides "better and efficient" service to members.
- Reduce the workload of employee.
- Faster retrieval of information about the desired book.
- Provide facility for proper monitoring reduces paper work and provide data security.
- All details will be available on a click.

4.5 Non-Functional Requirement & Constrains

A Non-functional Requirement is usually some form of constraints or restriction that must be considered when designing the solution. It tends to identify “user” constrains and “system” constrains.

- Non-Functional Requirements have the following characteristics:
- Uses simple language
- Not ambiguous
- Contains only one point
- Specific to one type of user
- Is qualified
- Describes what and not how

Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore it is required to take the database backup.

Security Requirements

We are going to develop a secured database for the library .There are different categories of users namely teaching staff, administrator, library staff ,students etc., Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, delete, append etc., all other users other than library staff only have the rights to retrieve the information about database.

Software Quality Attributes

The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database

Hardware Constraints

The system requires a database in order to store persistent data. The database should have backup capabilities.

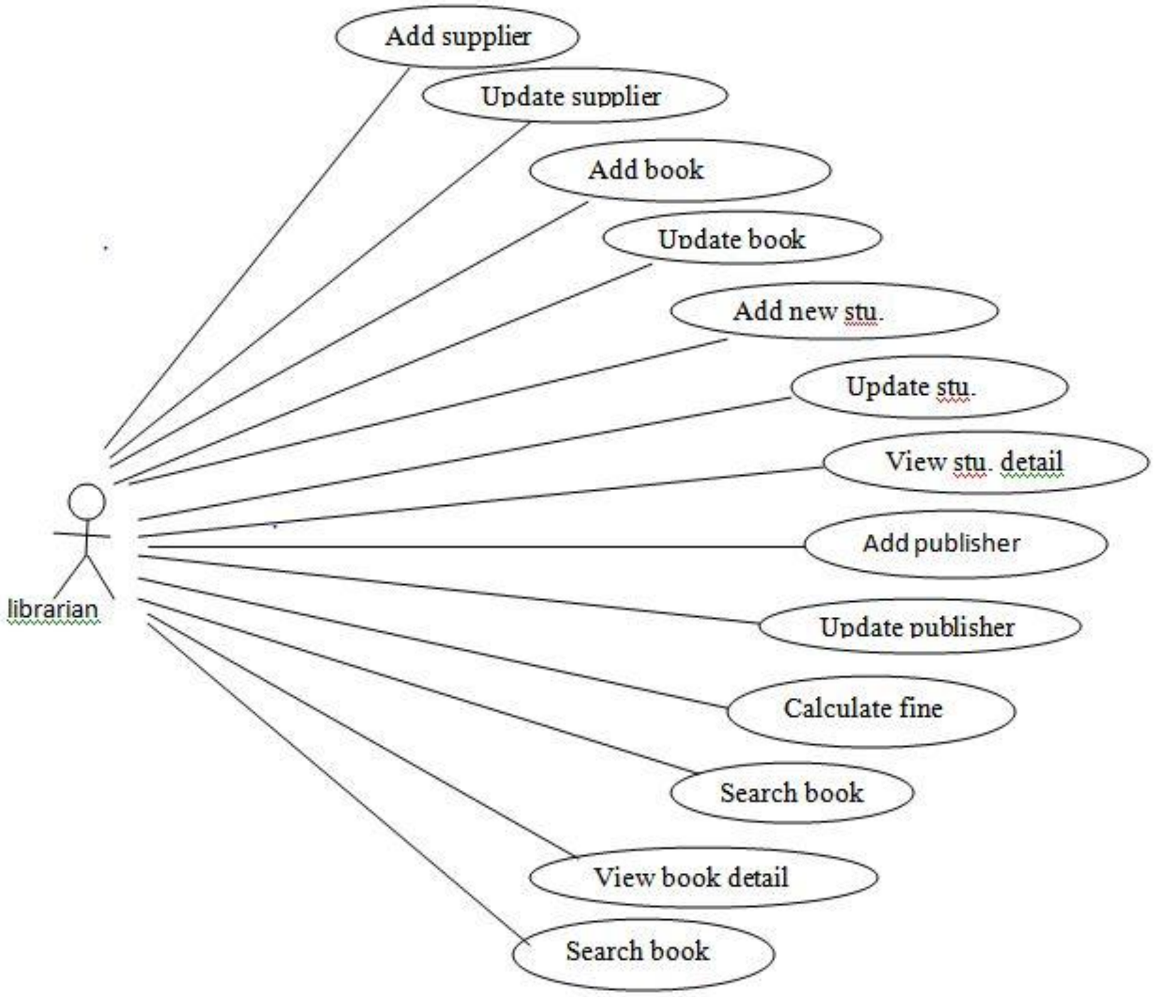
Chapter 5

System Definition

5.1 System Architecture

System architecture involves system and software design. System design partitions requirements to hardware and software systems. Software design represents software functions as a precursor to implementable executable programs. Following figure describes the typical phase of software design consist of Architectural, High-level and Low- level partition:

5.1.1 Architectural Representation



5.1.2 Architectural Description

After the requirements have been determined, the necessary specifications for the hardware, software, people and data resources and the information products that will satisfy the functional requirements of the proposed system can be determined. The design will serve as a blueprint for the system and helps detect problems before these errors or problems are built into the final system.

Library management system consists of the following modules:

Student Module:

When a new student arrives into the library in order to get a book issued, he/she is supposed to register himself/herself. Students will be given a registration or membership form which they are supposed to fill and submit to the librarian. The librarian then adds the students into the database. As the student enters different semesters of his academic year, the library database is updated with the relevant information. If the student fails to return or renew a book before a specified date, then he/she is charged with a late fine. Fines are also imposed in case the books or library cards are lost by the student.

Book/Magazines module:

Books/Magazines are maintained in the library database based on the categories(eg. Arts, Commerce, Science etc.,) Books are purchased from suppliers and they belong to different publishers, the details of which are maintained in the database. The librarian can add, edit, update books/magazines in the database, or search for the availability or get the stock details of a particular book/magazine. Some books come under the category of ‘reference books’ those books are issued to the students and the librarian keeps a check of those books too in the library.

Publisher/supplier module:

Every book is of a particular publisher and is bought from different suppliers. The details of the publishers and suppliers are also maintained in the library database. The librarian can add, delete or update a particular publisher or supplier.

5.2 Use Case Analysis

A use case analysis is the most common technique used to identify the requirements of a system and the information used to both define processes used and classes(which are a collection of actors and processes) which will be used both in the use case diagram and overall use case in the development or redesign of a software system or program. The use case analysis is the foundation upon which the system is built.

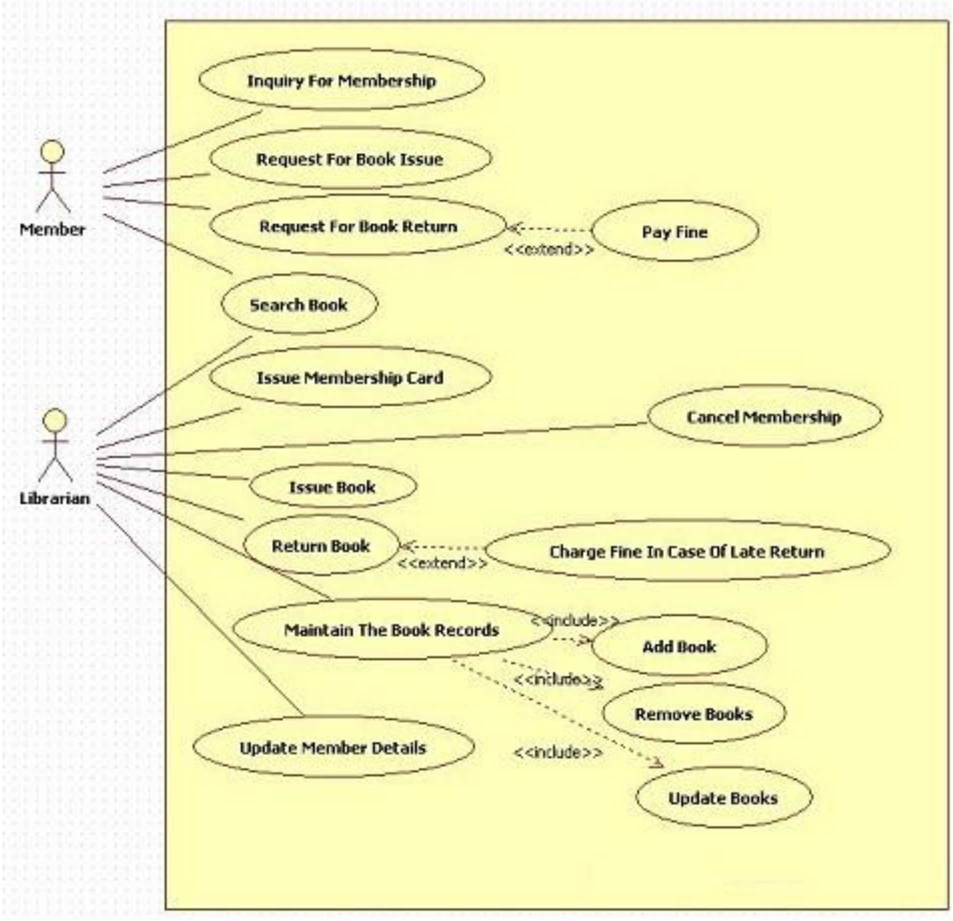
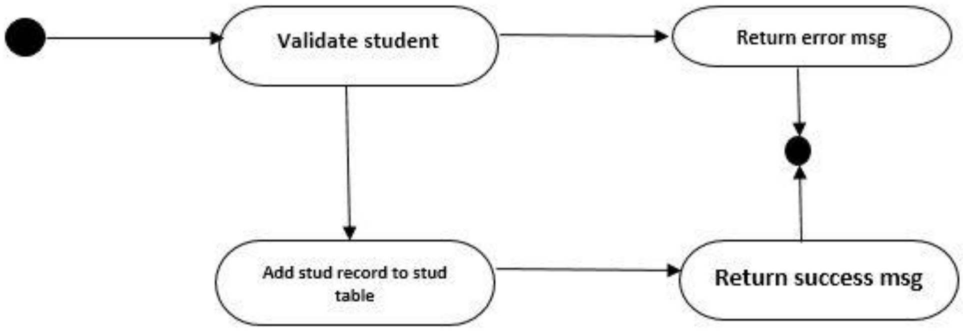


Fig: Use Case Diagram

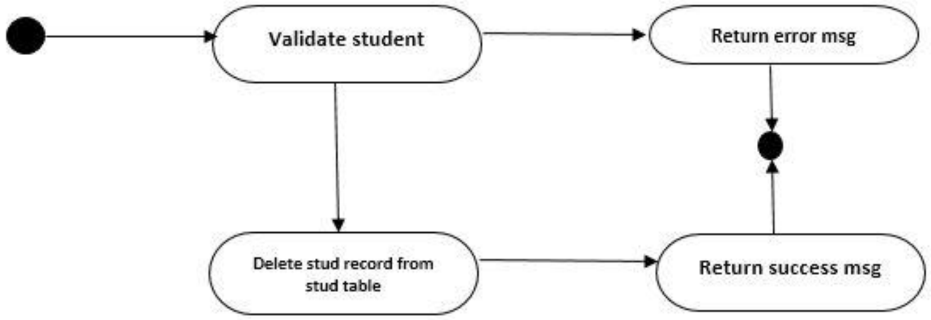
5.3 Activity Diagrams

<u>Student class</u>

Add a new Student

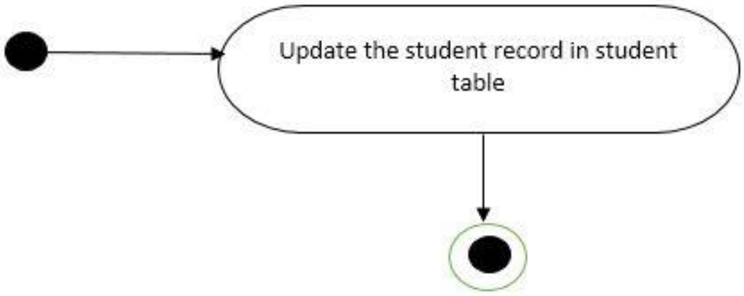


Delete Student

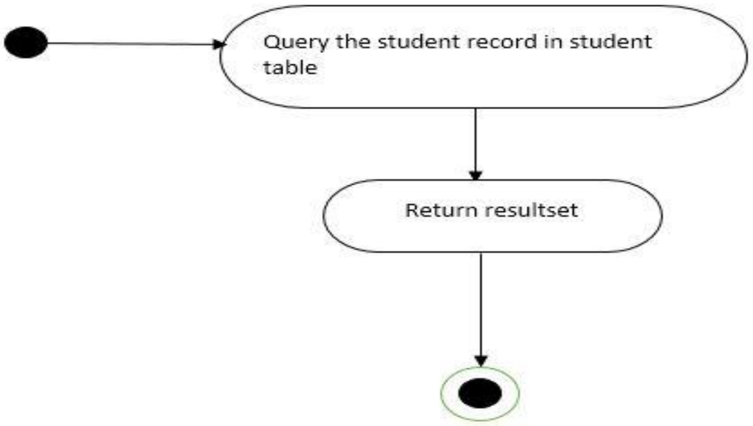


Update student

Update student:

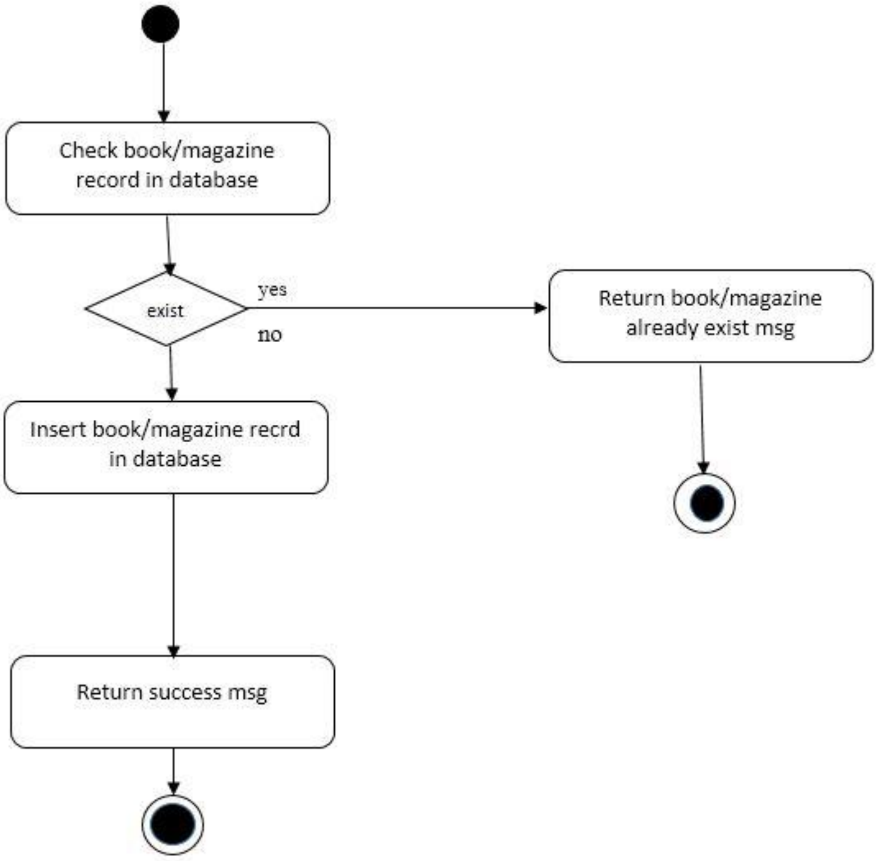


Search student

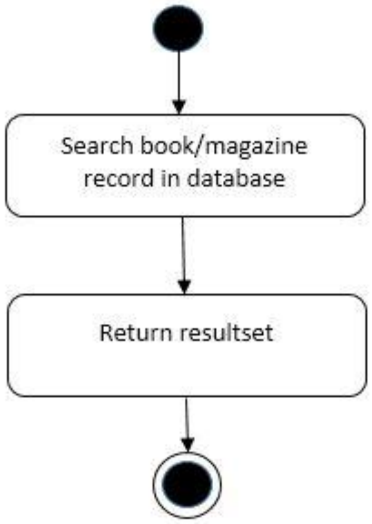


Book class

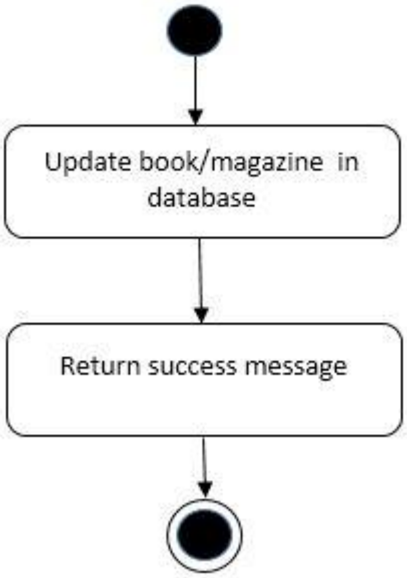
Add Book/Add Magazine in the Database


Check book/magazine record in database
exist
yes
no
Return book/magazine already exist msg
Insert book/magazine recrd in database
Return success msg

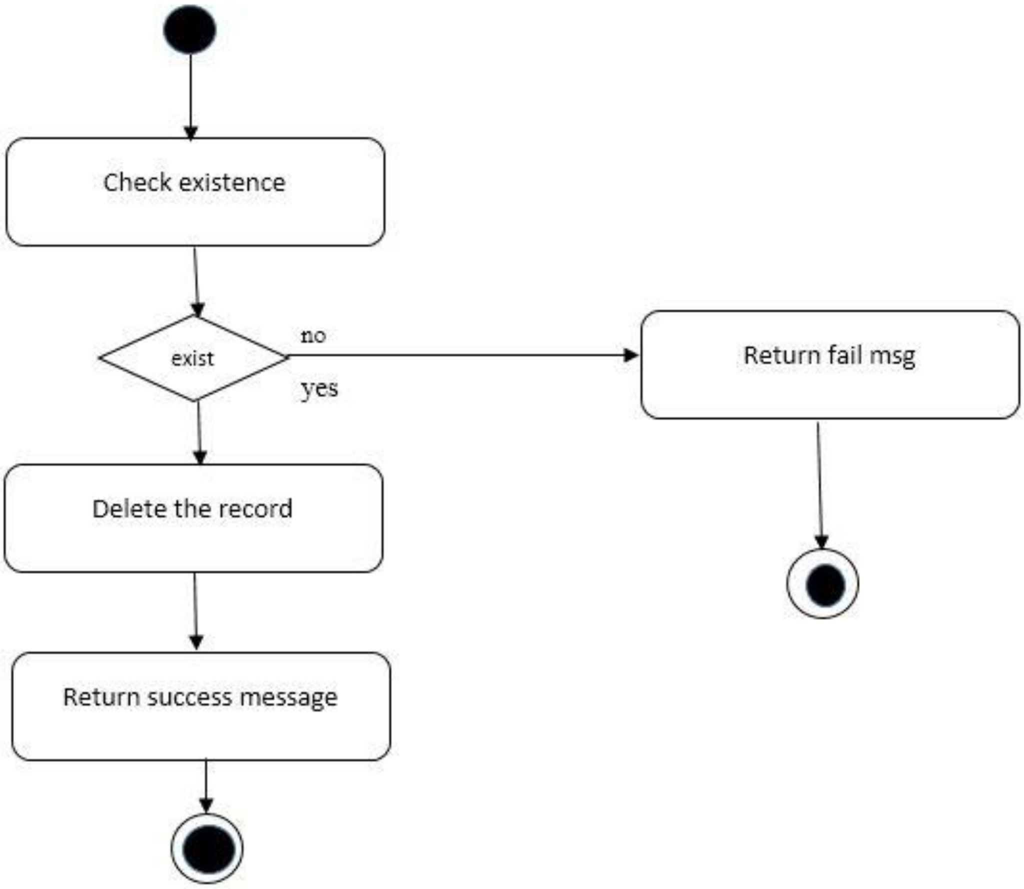
Search book


Search book/magazine record in database
Return resultset

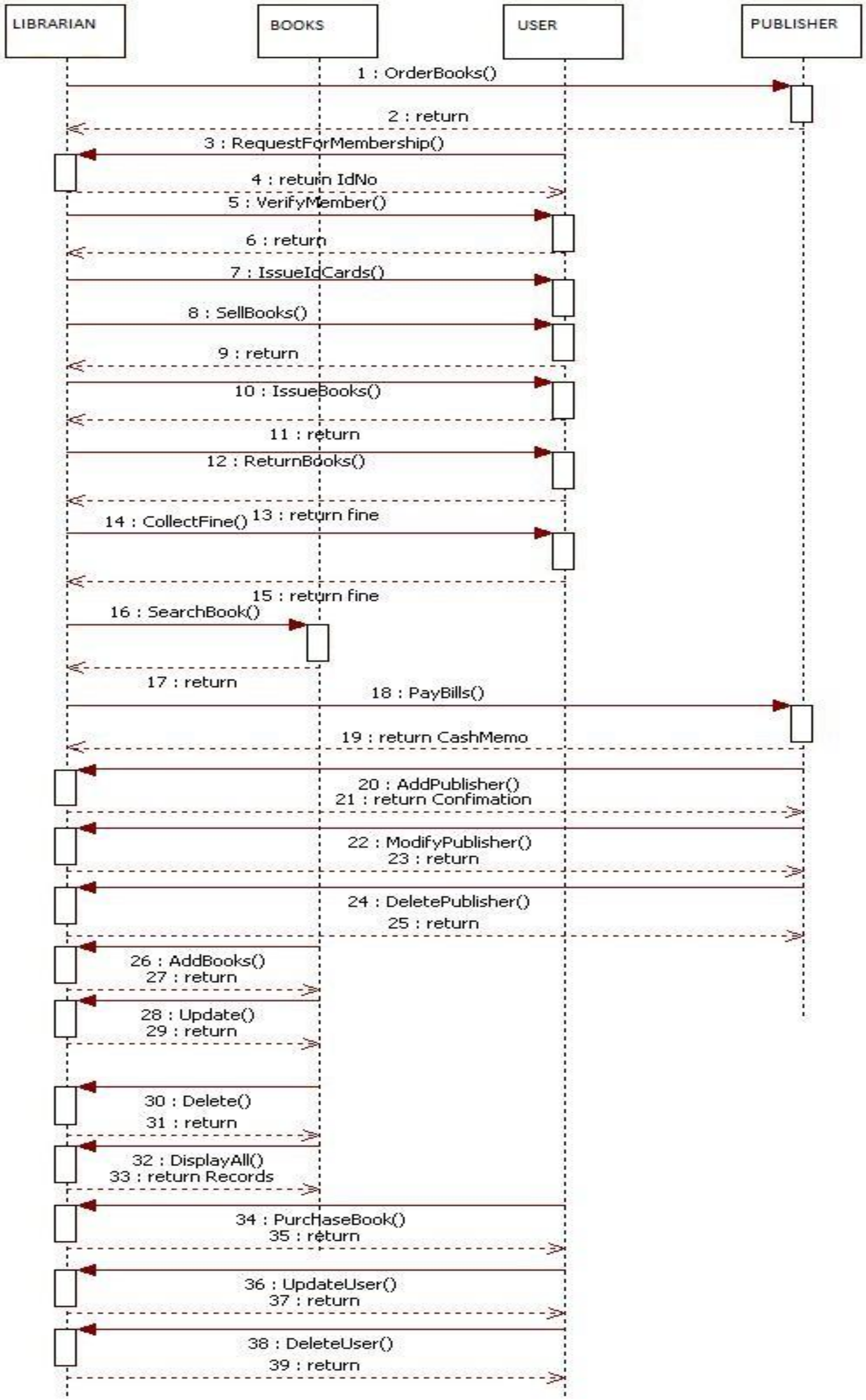
Update book


Update book/magazine in database
Return success message

Delete a book/magazine in the database



5.4 Sequence Diagram

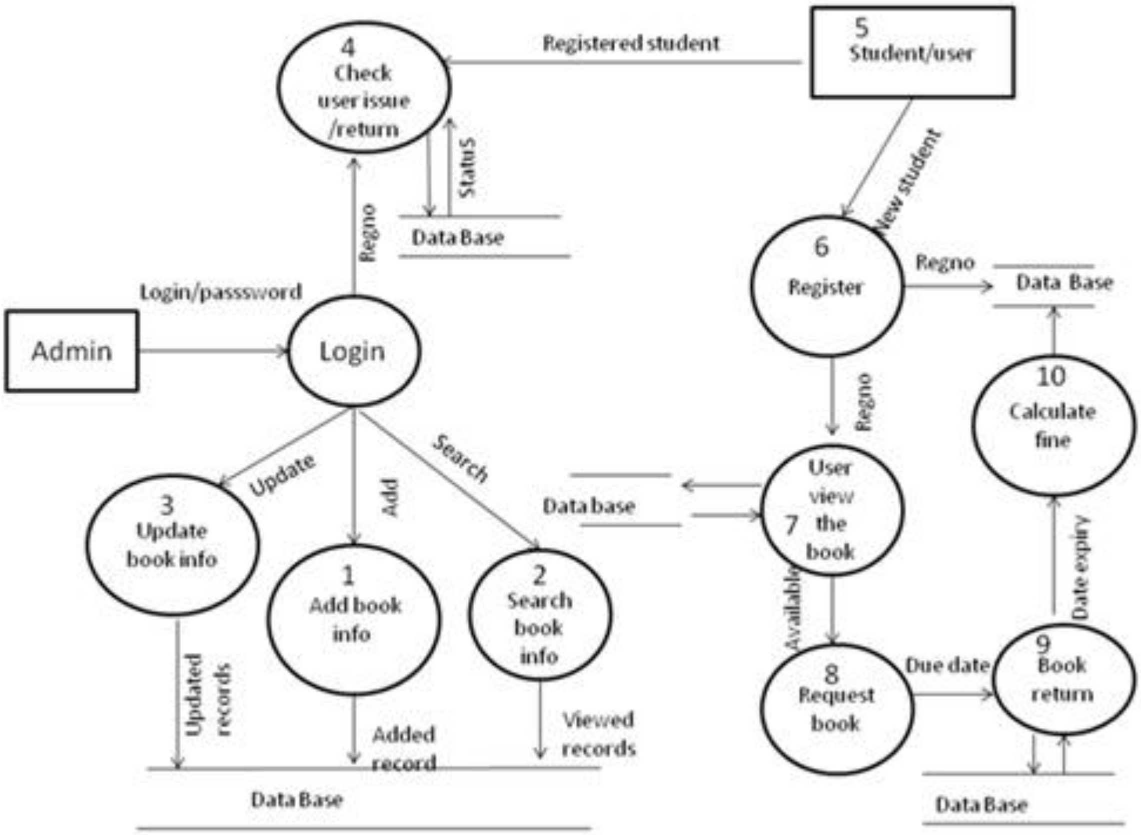


5.5 Data Flow Diagrams

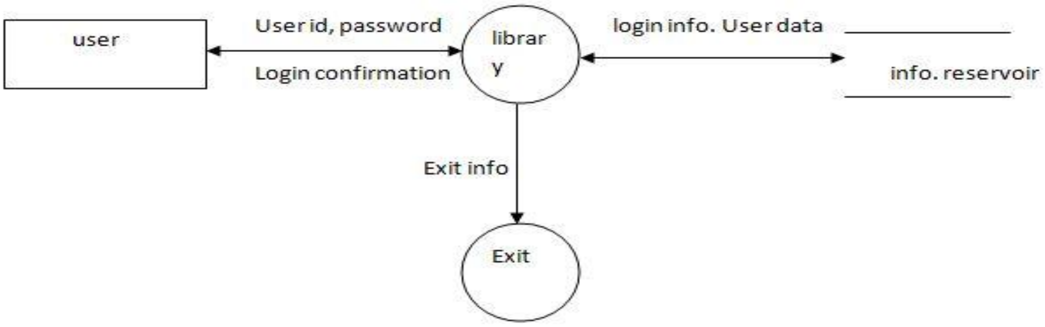
In DFD there are four symbols:

- **SQUARE** defines the originator or the destination of the system data.
- An **ARROW** identifies the data flows in motion. It’s a pipeline thru which information flows.
- A **CIRCLE** or a **BUBBLE** represents the process that transforms incoming data flow(s) into outgoing data flow(s).

Note: DFD describes what does flow (logical) rather than how they are processed, so it does not depend on hardware, software, data structures, or file organization. The key question that we are trying to answer is: what major transformations must occur for input to be correctly transformed into output?

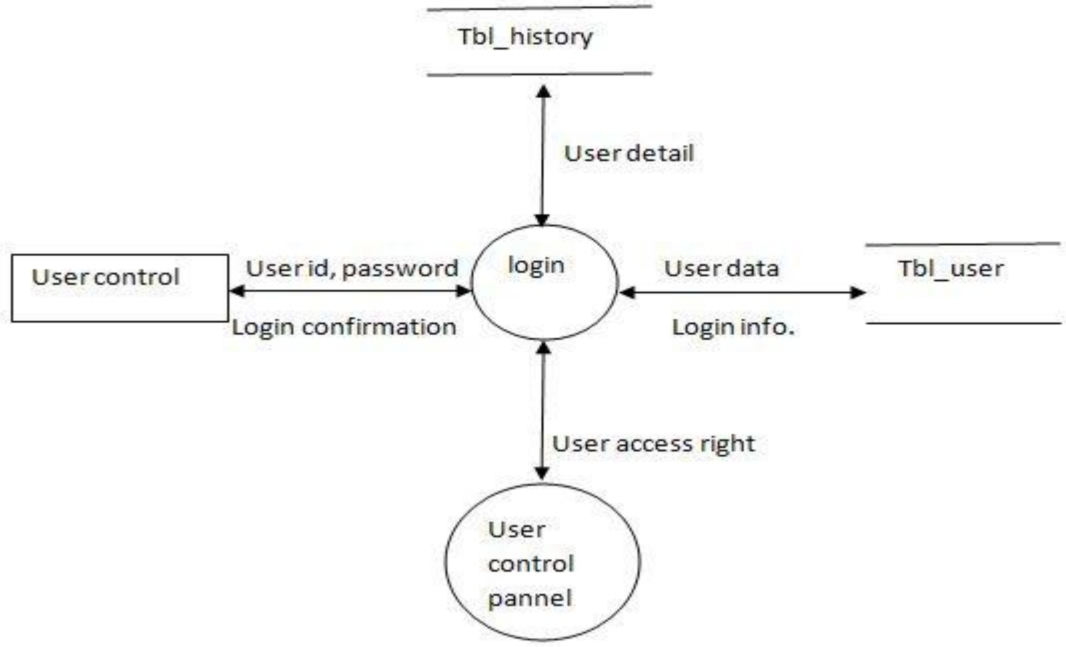


Level O DFD

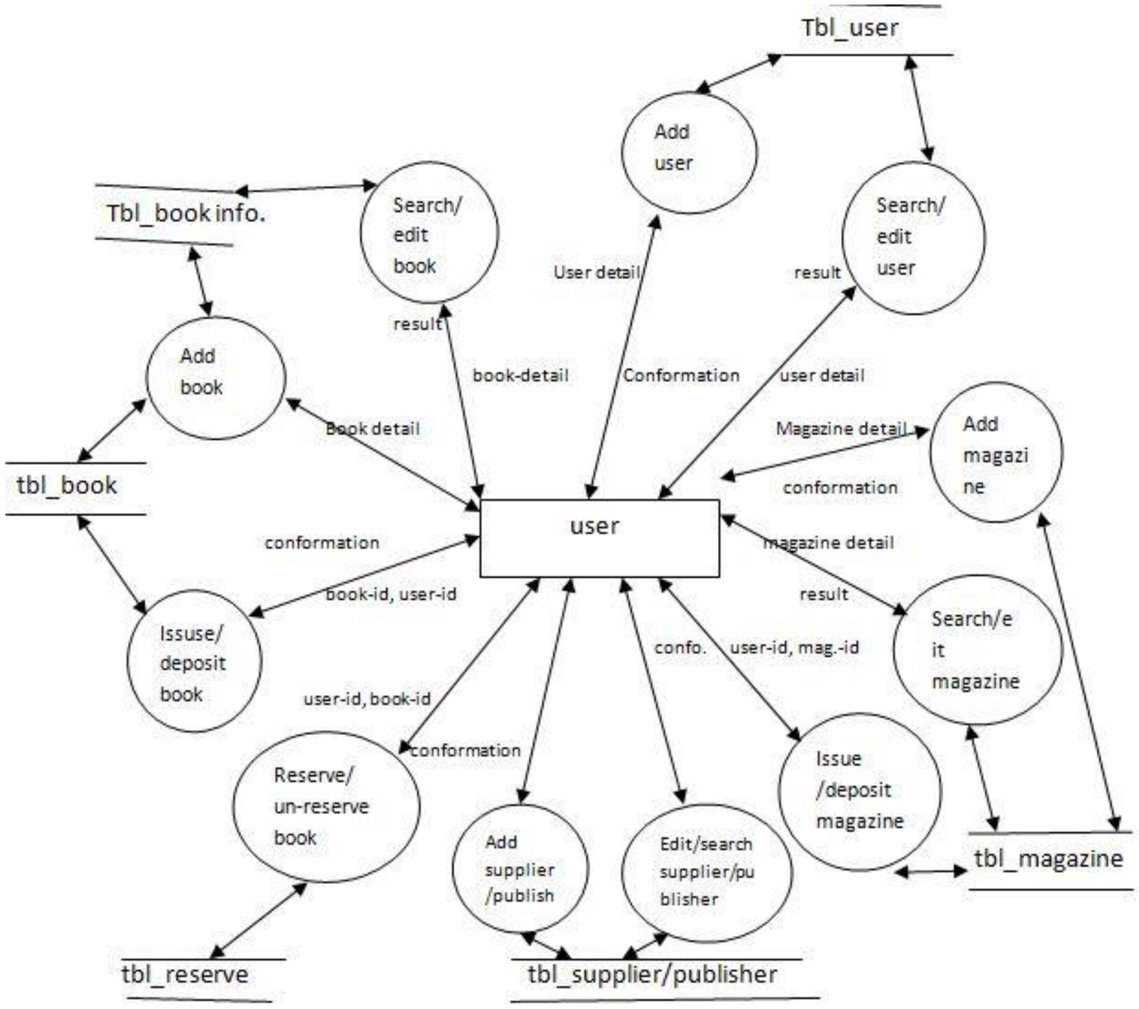


DFD 0

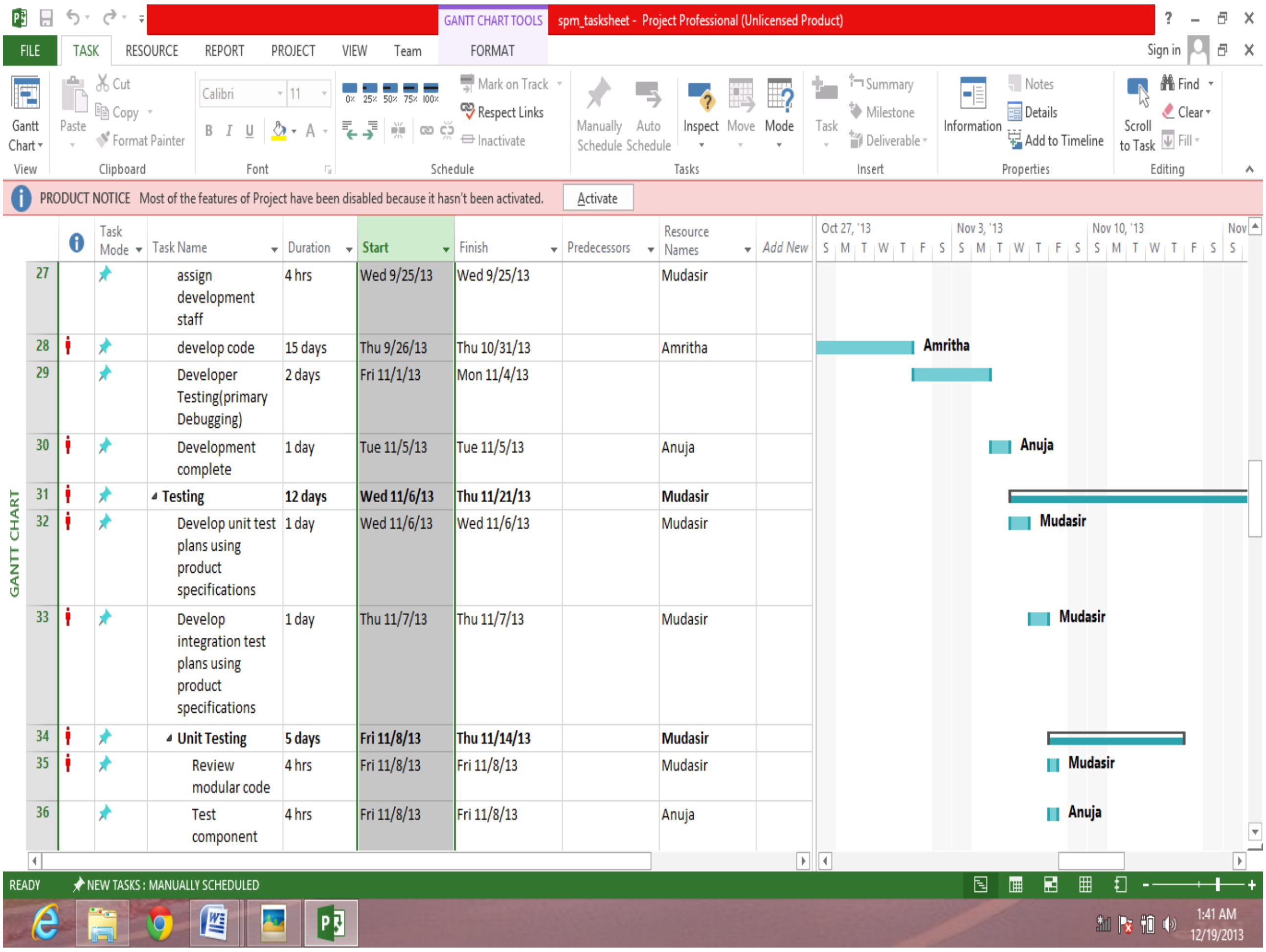
Level 1 DFD



Level 2 DFD



5.6 Design Tracking Using MSP 2013



spm_tasksheet - Project Professional (Unlicensed Product)

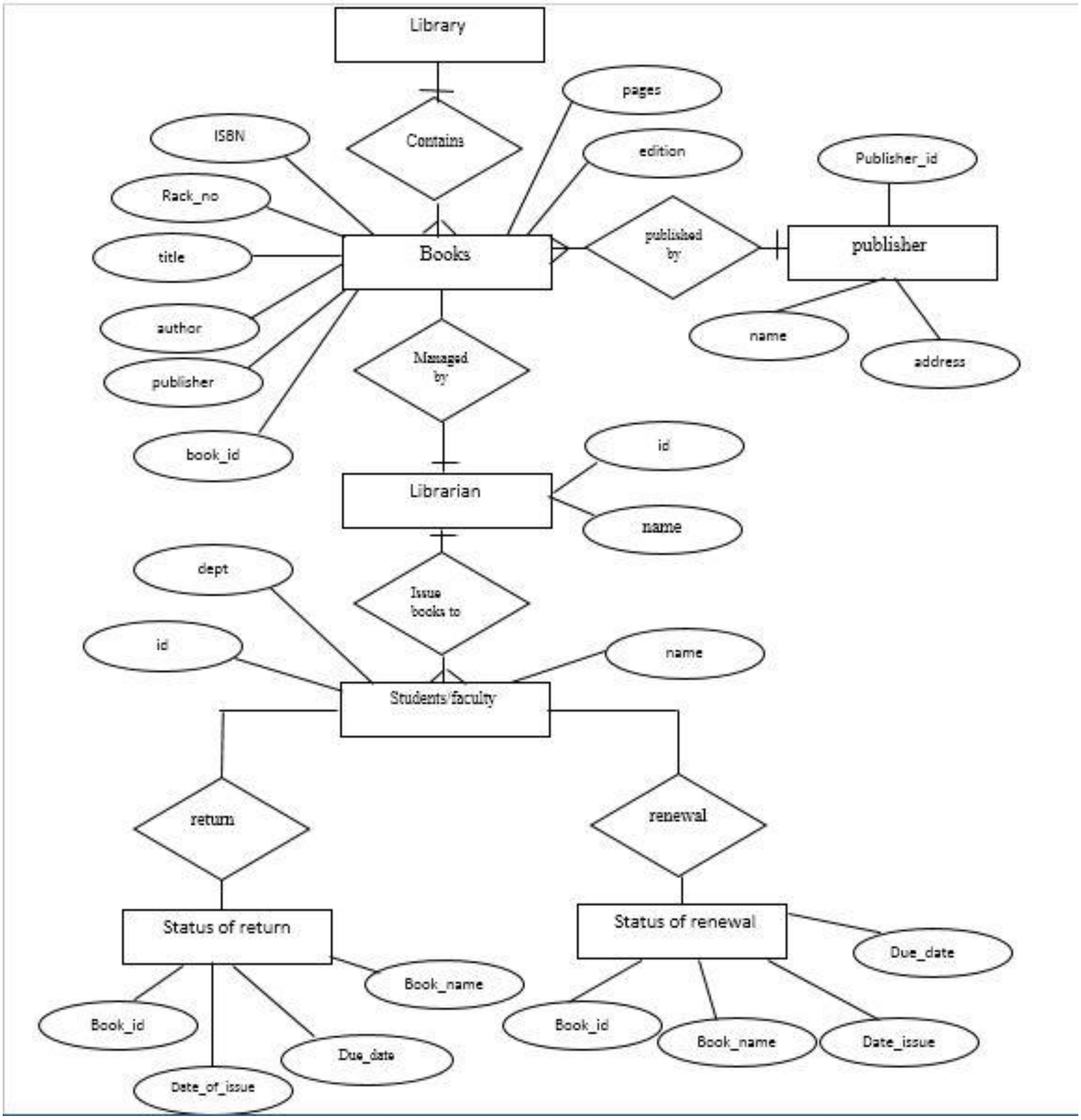
PRODUCT NOTICE Most of the features of Project have been disabled because it hasn't been activated.

		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names	Add New
27			assign development staff	4 hrs	Wed 9/25/13	Wed 9/25/13		Mudasir	
28			develop code	15 days	Thu 9/26/13	Thu 10/31/13		Amritha	
29			Developer Testing(primary Debugging)	2 days	Fri 11/1/13	Mon 11/4/13			
30			Development complete	1 day	Tue 11/5/13	Tue 11/5/13		Anuja	
31			**Testing**	**12 days**	**Wed 11/6/13**	**Thu 11/21/13**		**Mudasir**	
32			Develop unit test plans using product specifications	1 day	Wed 11/6/13	Wed 11/6/13		Mudasir	
33			Develop integration test plans using product specifications	1 day	Thu 11/7/13	Thu 11/7/13		Mudasir	
34			**Unit Testing**	**5 days**	**Fri 11/8/13**	**Thu 11/14/13**		**Mudasir**	
35			Review modular code	4 hrs	Fri 11/8/13	Fri 11/8/13		Mudasir	
36			Test component	4 hrs	Fri 11/8/13	Fri 11/8/13		Anuja	

Chapter 6

Detailed Design

6.1 Entity Relationship Diagram



6.2 Database Design

Table name: tbllogin

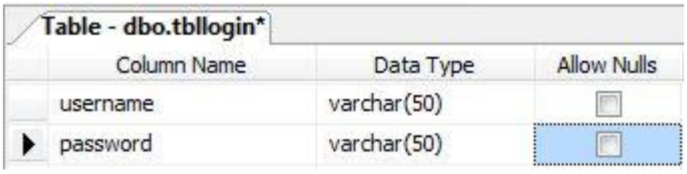


Table - dbo.tbllogin*

	Column Name	Data Type	Allow Nulls
	username	varchar(50)	□
▶	password	varchar(50)	□

Table name: tbl_bookdetails

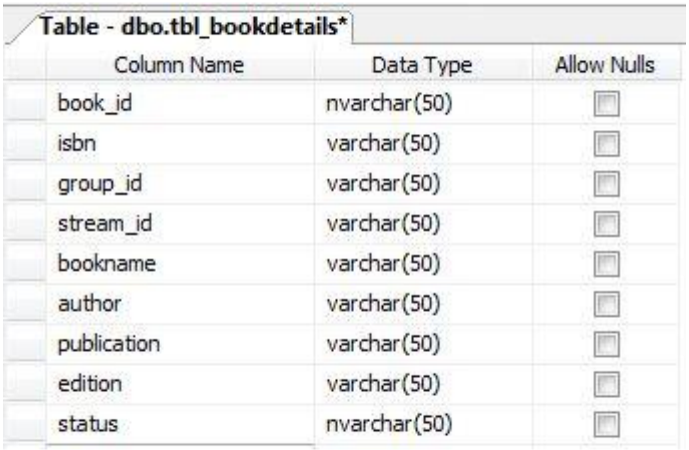


Table - dbo.tbl_bookdetails*

Column Name	Data Type	Allow Nulls
book_id	nvarchar(50)	□
isbn	varchar(50)	□
group_id	varchar(50)	□
stream_id	varchar(50)	□
bookname	varchar(50)	□
author	varchar(50)	□
publication	varchar(50)	□
edition	varchar(50)	□
status	nvarchar(50)	□

Table name: tbl_issue

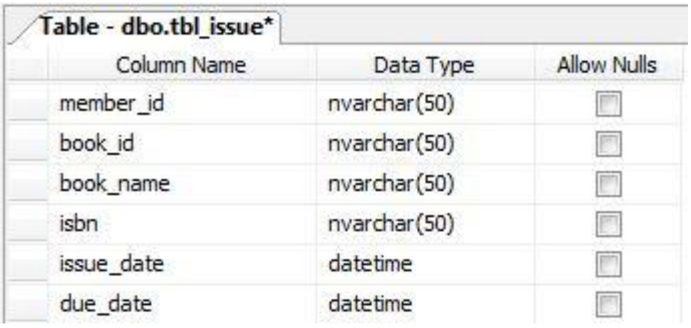


Table - dbo.tbl_issue*

Column Name	Data Type	Allow Nulls
member_id	nvarchar(50)	□
book_id	nvarchar(50)	□
book_name	nvarchar(50)	□
isbn	nvarchar(50)	□
issue_date	datetime	□
due_date	datetime	□

Table name: tbl_renewalbook

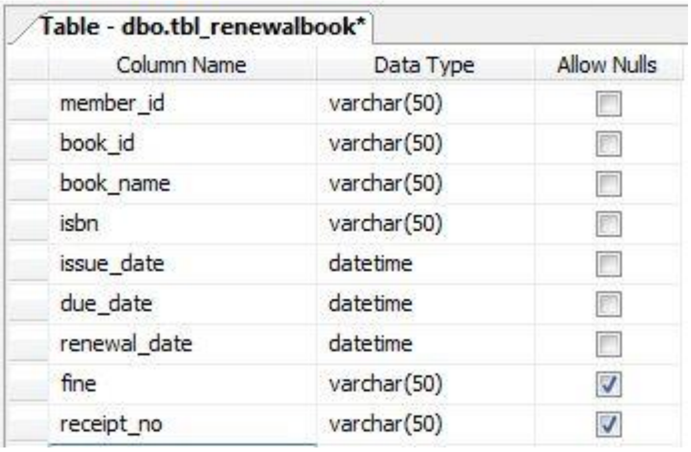


Table - dbo.tbl_renewalbook*

Column Name	Data Type	Allow Nulls
member_id	varchar(50)	☐
book_id	varchar(50)	☐
book_name	varchar(50)	☐
isbn	varchar(50)	☐
issue_date	datetime	☐
due_date	datetime	☐
renewal_date	datetime	☐
fine	varchar(50)	☑
receipt_no	varchar(50)	☑

Table name: tbl_stream

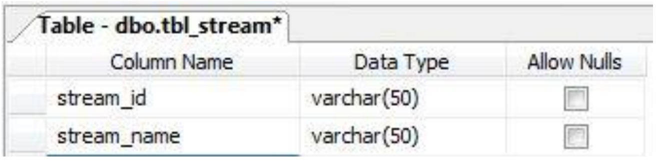


Table - dbo.tbl_stream*

Column Name	Data Type	Allow Nulls
stream_id	varchar(50)	☐
stream_name	varchar(50)	☐

Table name: tbl_member

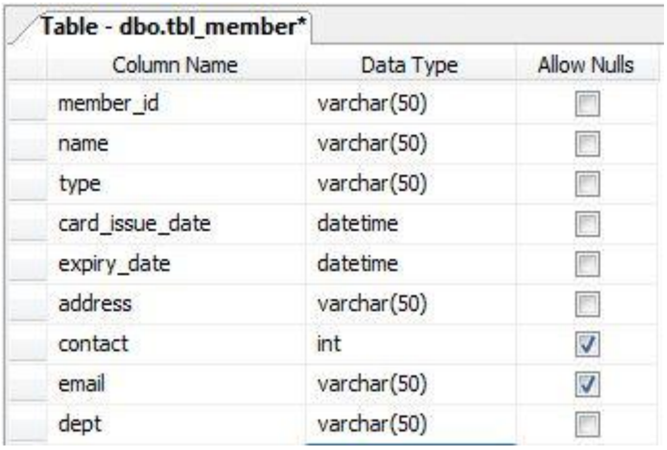


Table - dbo.tbl_member*

Column Name	Data Type	Allow Nulls
member_id	varchar(50)	☐
name	varchar(50)	☐
type	varchar(50)	☐
card_issue_date	datetime	☐
expiry_date	datetime	☐
address	varchar(50)	☐
contact	int	☑
email	varchar(50)	☑
dept	varchar(50)	☐

Table name: tbl_returnbook

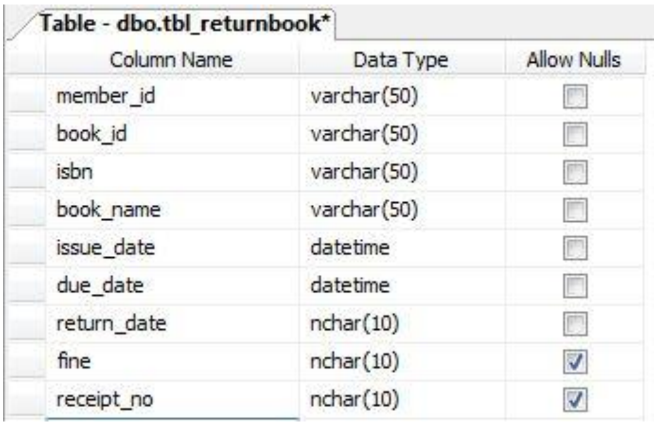


Table - dbo.tbl_returnbook*

Column Name	Data Type	Allow Nulls
member_id	varchar(50)	□
book_id	varchar(50)	□
isbn	varchar(50)	□
book_name	varchar(50)	□
issue_date	datetime	□
due_date	datetime	□
return_date	nchar(10)	□
fine	nchar(10)	☑
receipt_no	nchar(10)	☑

Table name: tbl_stockdetails

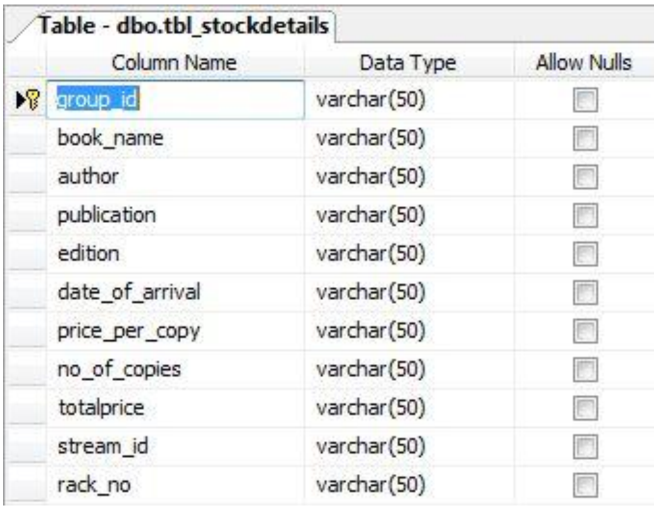
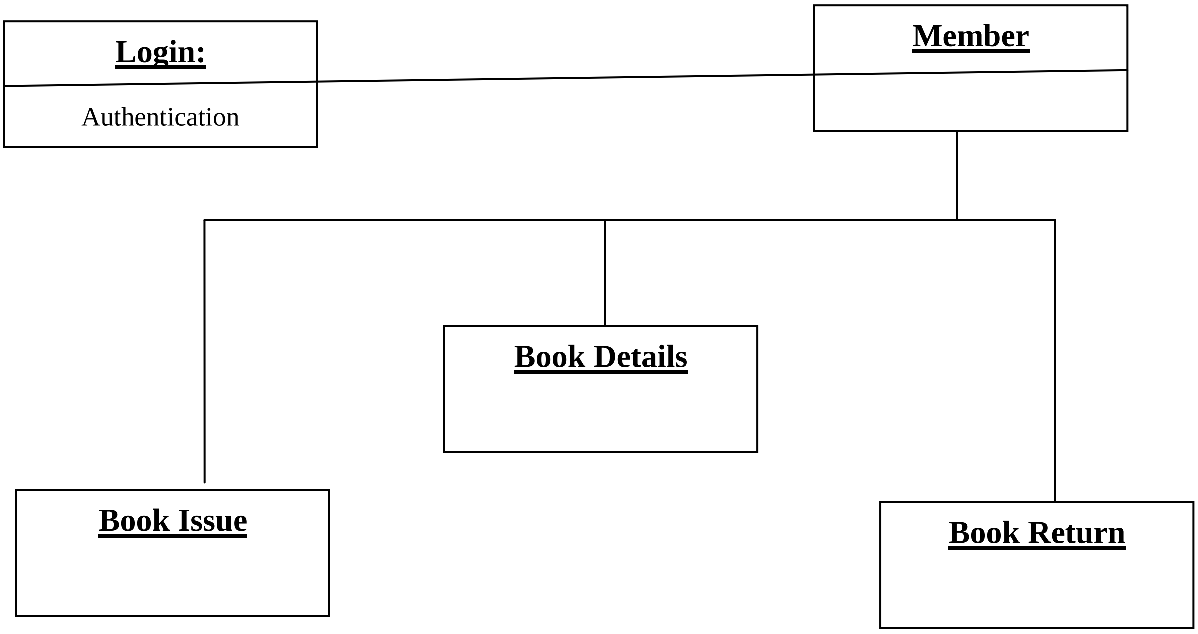


Table - dbo.tbl_stockdetails

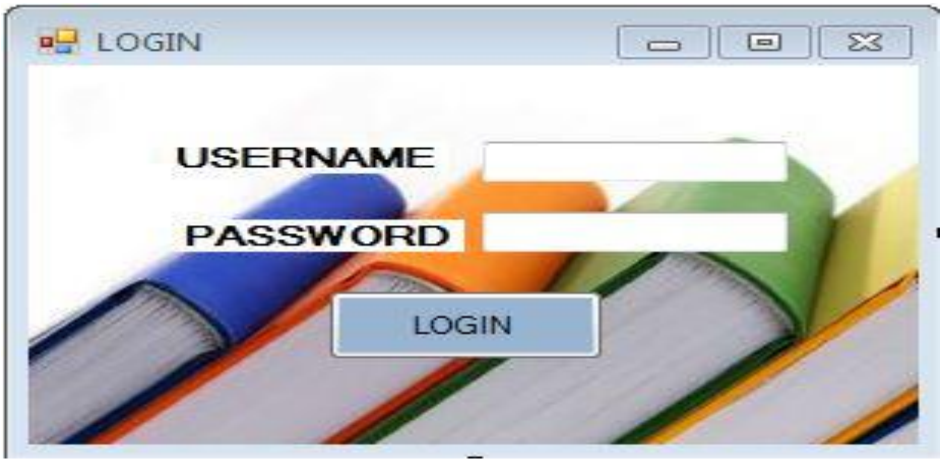
Column Name	Data Type	Allow Nulls
group_id	varchar(50)	□
book_name	varchar(50)	□
author	varchar(50)	□
publication	varchar(50)	□
edition	varchar(50)	□
date_of_arrival	varchar(50)	□
price_per_copy	varchar(50)	□
no_of_copies	varchar(50)	□
totalprice	varchar(50)	□
stream_id	varchar(50)	□
rack_no	varchar(50)	□

6.3 Object Design

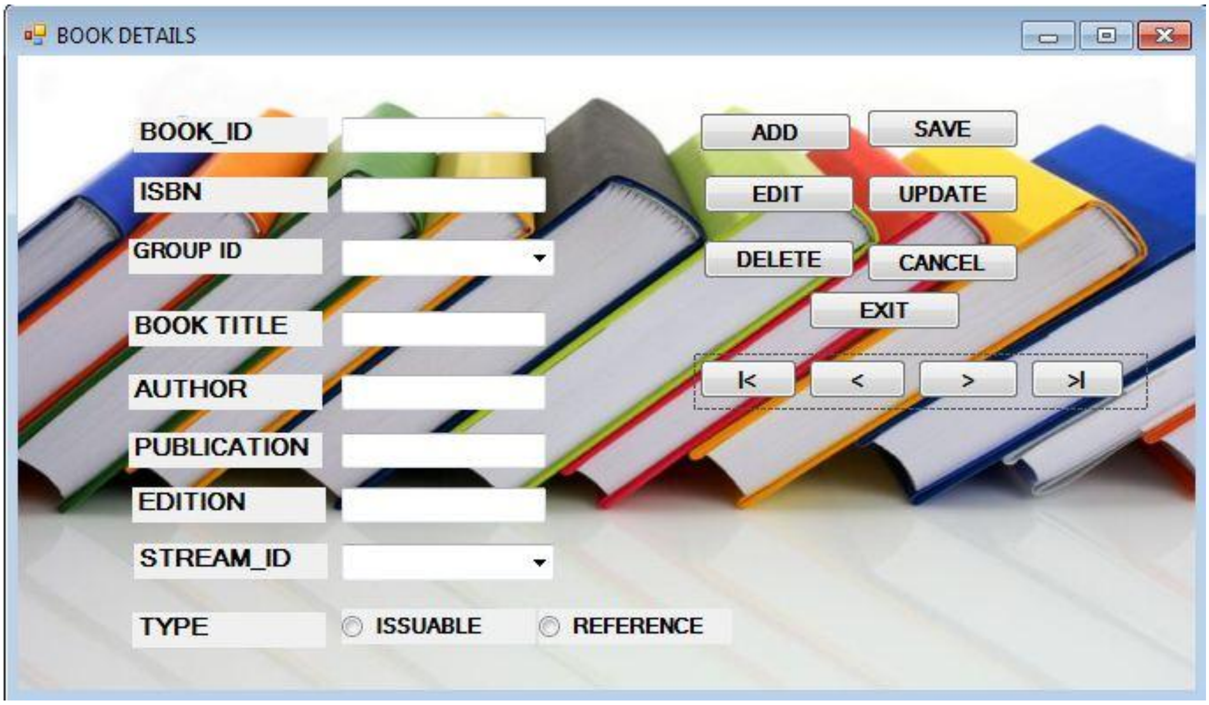


6.4 Screen Design

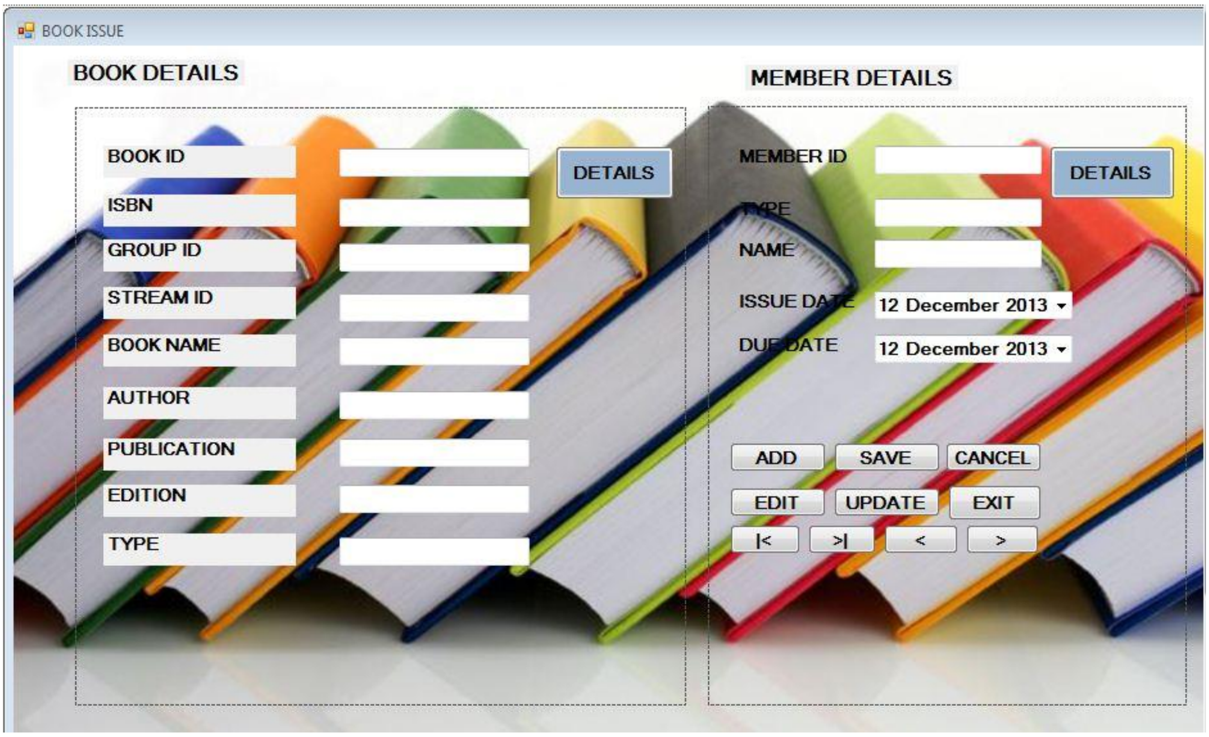
6.4.1. Main Page



6.4.2 Book Details



6.4.3 Book Issue



BOOK ISSUE

BOOK DETAILS

BOOK ID

ISBN

GROUP ID

STREAM ID

BOOK NAME

AUTHOR

PUBLICATION

EDITION

TYPE

MEMBER DETAILS

MEMBER ID

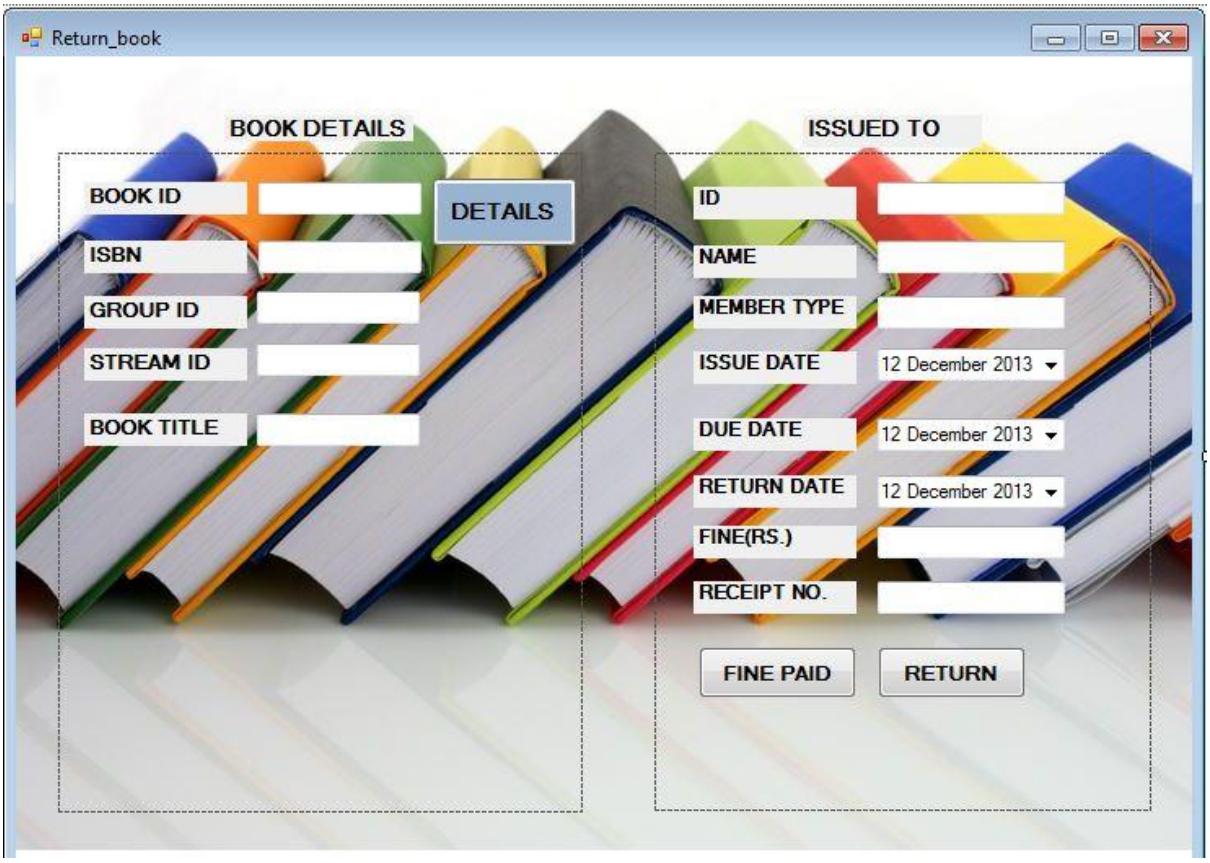
TYPE

NAME

ISSUE DATE: 12 December 2013

DUE DATE: 12 December 2013

6.4.4 Book Return



Return_book

BOOK DETAILS

BOOK ID

ISBN

GROUP ID

STREAM ID

BOOK TITLE

ISSUED TO

ID

NAME

MEMBER TYPE

ISSUE DATE: 12 December 2013

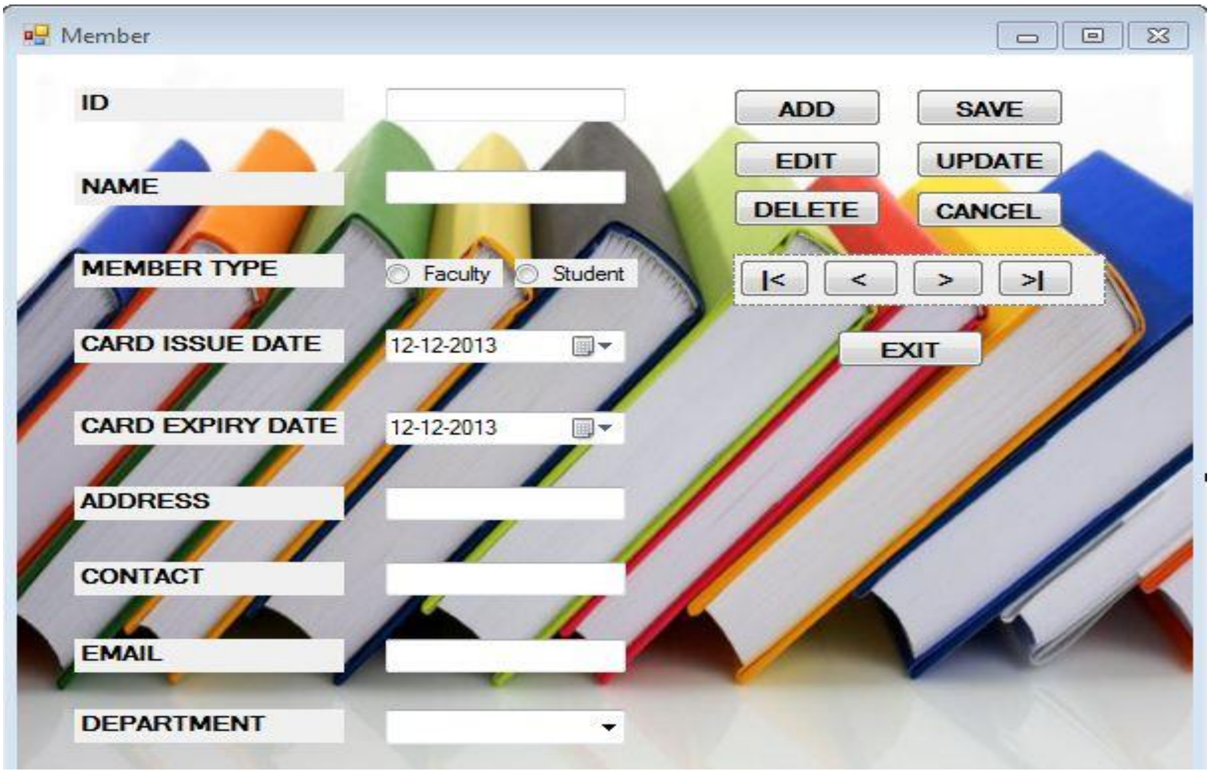
DUE DATE: 12 December 2013

RETURN DATE: 12 December 2013

FINE(RS.)

RECEIPT NO.

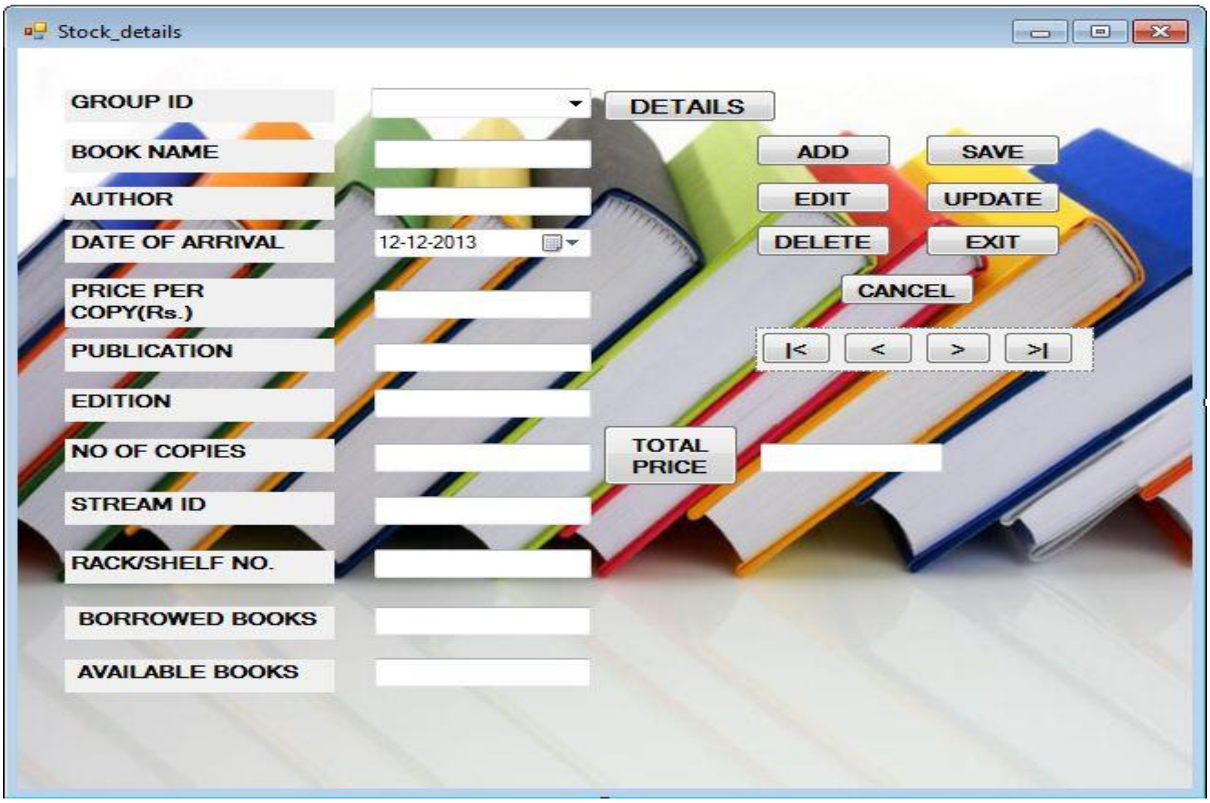
6.4.5 Member



CARD ISSUE DATE 12-12-2013

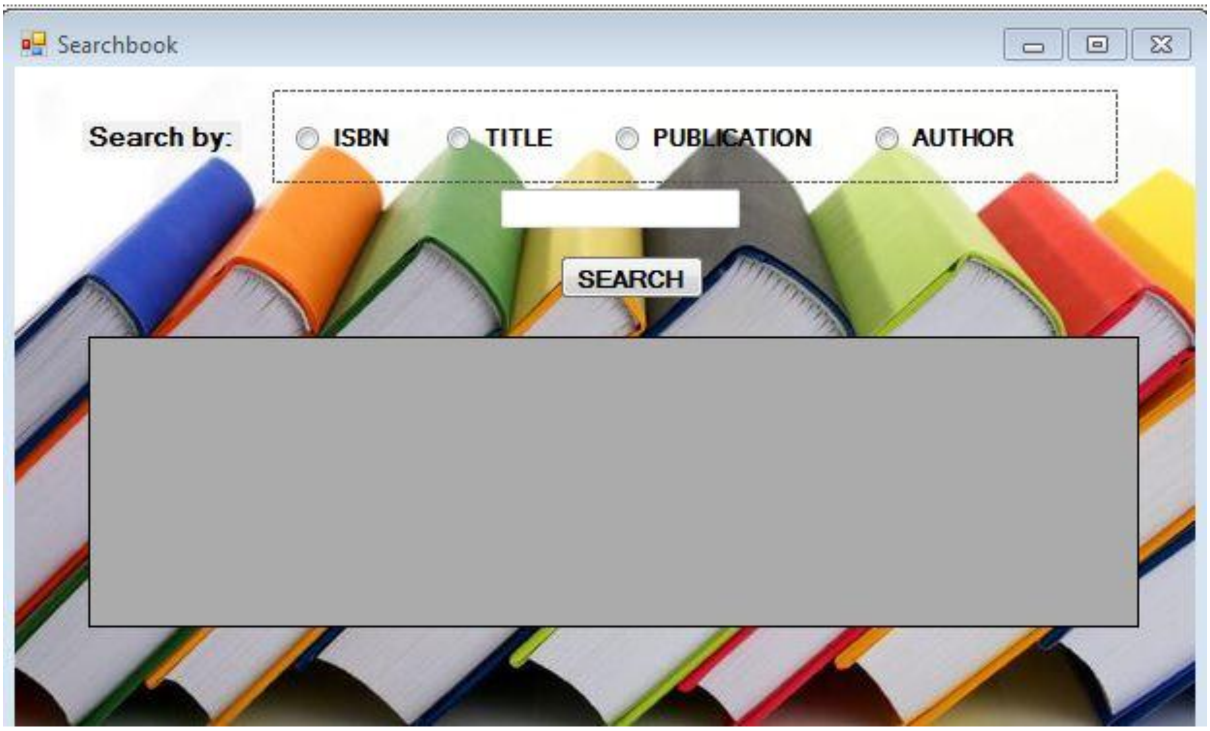
CARD EXPIRY DATE 12-12-2013

6.4.6 Stock Details

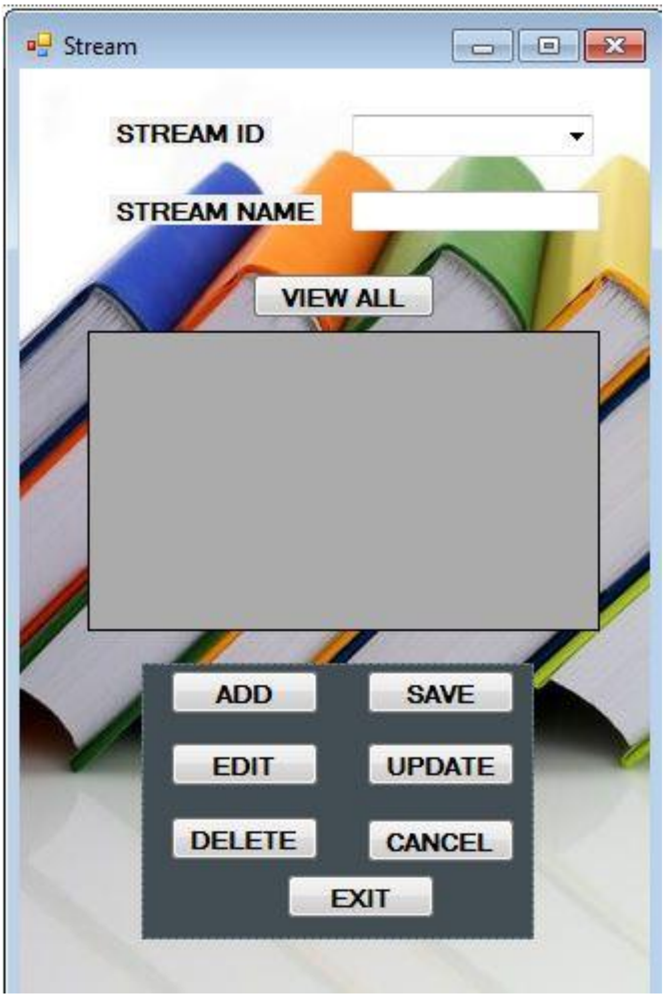


DATE OF ARRIVAL 12-12-2013

6.4.7 Search Book



6.4.8 Stream


Stream
STREAM ID
STREAM NAME
VIEW ALL
ADD
SAVE
EDIT
UPDATE
DELETE
CANCEL
EXIT

Chapter 7

Implementation

System implementation is the stage when the user has thoroughly tested the system and approves all the features provided by the system. The various tests are performed and the system is approved only after all the requirements are met and the user is satisfied. The new system may be totally new; replacing an existing manual or automated system, or it may a major modification to an existing system. In the either case, proper implementation is essential to provide a reliable system to meet organization requirements. Successful implementation may not guarantee improvement in the organization using the new system (that is a design question), but improper will prevent it. Implementation is the process of having systems personal check out and put new equipment to use, train users, install the new application and construct any files of data needed to use it. This phase is less creative than system design. Depending on the size of the organization that will be involve in using the application and the risk involved in its use, system developer may choose to test the operation in only one area of the firm with only one or two persons. Sometimes, they will run both old and new system in parallel way to compare the results. In steel other situations, system developers stop using the old systems one day and start using the new one the next. The implementation of the web based or LAN base network project has some extra steps at the time of implementation. The modules in this system are:

- Login
- Book Renew
- Book Return
- Member

LOGIN:

```
Imports System.Data.SqlClient
PublicClass login
PrivateSub btnlogin_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnlogin.Click
Dim dbcon AsNew SqlConnection("Data Source=AMRITHA-PC;Initial
Catalog=library_manage;Integrated Security=True")
        dbcon.Open()
Dim dbcmd AsNew SqlCommand
        dbcmd.Connection = dbcon
        dbcmd.CommandText = "Select Count(*) From tbllogin where 
Username='"& TextBox1.Text &"' AND Password='"& TextBox2.Text &"'"
If dbcmd.ExecuteScalar = 1 Then
            MsgBox("Login successful")
Else
            MsgBox("Incorrect username/password! Login failed")
            TextBox1.Clear()
            TextBox2.Clear()
EndIf
        dbcmd.Dispose()
        dbcon.Dispose()
EndSub
```

Book Issue:

```
Imports System.Data.SqlClient

PublicClass Issue_details
Dim str1 AsString
Dim dated As DateTime
Dim id AsInteger
Dim dr As SqlDataReader
Dim issdate As DateTime
Dim duedate As DateTime
Dim flag123 AsBoolean

Dim ds AsNew DataSet
Dim ds1 AsNew DataSet
Dim ds2 AsNew DataSet
Dim ds3 AsNew DataSet
Dim ds4 AsNew DataSet
Dim ds5 AsNew DataSet

Dim da AsNew SqlDataAdapter
Dim da1 AsNew SqlDataAdapter
Dim da2 AsNew SqlDataAdapter
Dim da3 AsNew SqlDataAdapter
Dim da4 AsNew SqlDataAdapter
Dim da5 AsNew SqlDataAdapter

Dim i AsInteger
Dim chk AsBoolean
Dim strbg AsString
Dim count AsString

PublicSub discontrol()
Me.txtbookid.ReadOnly = True
Me.txtisbn.ReadOnly = True
Me.txtgid.ReadOnly = True
Me.txtbname.ReadOnly = True
Me.txtsid.ReadOnly = True
Me.txtpub.ReadOnly = True
Me.txtauthor.ReadOnly = True
Me.txtedition.ReadOnly = True
Me.txtype.ReadOnly = True
Me.txtmemid.ReadOnly = True
Me.txttype.ReadOnly = True
Me.txtmemname.ReadOnly = True
```

Book Renew

```
Imports System.Data.SqlClient
PublicClass bookrenewal
Dim dr As SqlDataReader
Dim dr1 As SqlDataReader
Dim dr2 As SqlDataReader
Dim ds AsNew DataSet
Dim ds4 AsNew DataSet
Dim da AsNew SqlDataAdapter
Dim da4 AsNew SqlDataAdapter
Dim due As DateTime
Dim returndate As DateTime
Dim nextduedate As DateTime
Dim diff As TimeSpan
Dim days AsInteger
Dim finerate AsInteger
Dim renewno AsInteger
Dim mid AsString
Dim bid AsString
PublicSub checkbookid() ' if book id is left blank
IfMe.txtbookid.Text = ""Then
            MessageBox.Show("Enter the Book ID to get details", "Enter",
MessageBoxButtons.OK, MessageBoxIcon.Error)
EndIf
EndSub
PrivateSub btndetails_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btndetails.Click
Call checkbookid()
Call conecDB()
Call initCMD()
        ds4.Clear()
        da4 = New SqlDataAdapter("select * from tbl_issue where book_id='"&
txtbookid.Text &"'", connDB)
        da4.Fill(ds4, "tbl_issue")
' If (ds4.Tables(0).Rows.Count.Equals("")) Then
If (ds4.Tables(0).Rows.Count.Equals(0)) Then
            MessageBox.Show("Invalid Book ID", "Enter",
MessageBoxButtons.OK, MessageBoxIcon.Error)
Else
```

Book Return

```
Imports System.Data.SqlClient
PublicClass Return_book
Dim dr As SqlDataReader
Dim dr1 As SqlDataReader
Dim dr2 As SqlDataReader
Dim ds AsNew DataSet
Dim ds4 AsNew DataSet
Dim da AsNew SqlDataAdapter
Dim due As DateTime
Dim returndate As DateTime
Dim da4 AsNew SqlDataAdapter
Dim diff As TimeSpan
Dim days AsInteger
Dim finerate AsInteger
Dim mid AsString
Dim bid AsString
```

```
PublicSub checkbookid() ' if book id is left blank
IfMe.txtbookid.Text = ""Then
            MessageBox.Show("Enter the  book id to get details")
EndIf
EndSub
PrivateSub btndetails_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btndetails.Click
Call checkbookid()
Call conecDB()
Call initCMD()
        ds4.Clear()
        da4 = New SqlDataAdapter("select * from tbl_issue where book_id='"&
txtbookid.Text &"'", connDB)
        da4.Fill(ds4, "tbl_issue")
' If (ds4.Tables(0).Rows.Count.Equals("")) Then
If (ds4.Tables(0).Rows.Count.Equals(0)) Then
            MessageBox.Show("Invalid Book ID")
Else
Call conecDB()
Call initCMD()
            SQL = "Select member_id,book_name,isbn,issue_date,due_date from
tbl_issue where book_id='"& txtbookid.Text &"'"
Call execComDB(SQL)
            dr = comDB.ExecuteReader
            dr.Read()
```

Member

```
Imports System.Data.SqlClient
PublicClass Member
Dim ds AsNew DataSet
Dim da AsNew SqlDataAdapter
Dim issuedate As DateTime
Dim i AsInteger
Dim str1 AsString
Dim dated As DateTime
Dim datedissue As DateTime
Dim datedexpiry As DateTime
PrivateSub btnadd_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnadd.Click
Call disnavigate()
        btnadd.Enabled = False
        btndel.Enabled = False

        dtissue.Value = Now()
        btnedit.Enabled = False
        btnupdate.Enabled = False
Me.btncancel.Enabled = True
Me.btnsave.Enabled = True
Call conecDB()
Call initCMD()

If (ds.Tables(0).Rows.Count) > 0 Then
            da = New SqlDataAdapter("select * from tbl_member", connDB)

            da.Fill(ds, "tbl_member")
            i = ds.Tables(0).Rows.Count - 1 'go to last record

            cleartxtbox()
            enaControl()
```

Chapter 8

Testing & Results

8.1 Need For Testing

The need for testing is to find the errors/problems in the project to make it bug free.

Test for valid Input

In module, if it is cleary defined that what input it should accept or reject. So very often this is the starting point of testing.

All valid input must be accepted by the program. To find any anomaly in this, the valid inputs testing necessary.

Test for boundary value

This is very important,in a module, different type of variables and looping statement will be used, to check that they violate the boundary value criteria, this test is necessary.

Test for Exception

There was always some exception, to which we cannot help, but can catch them and publish a message to the user so that could understand the input is giving is wrong.

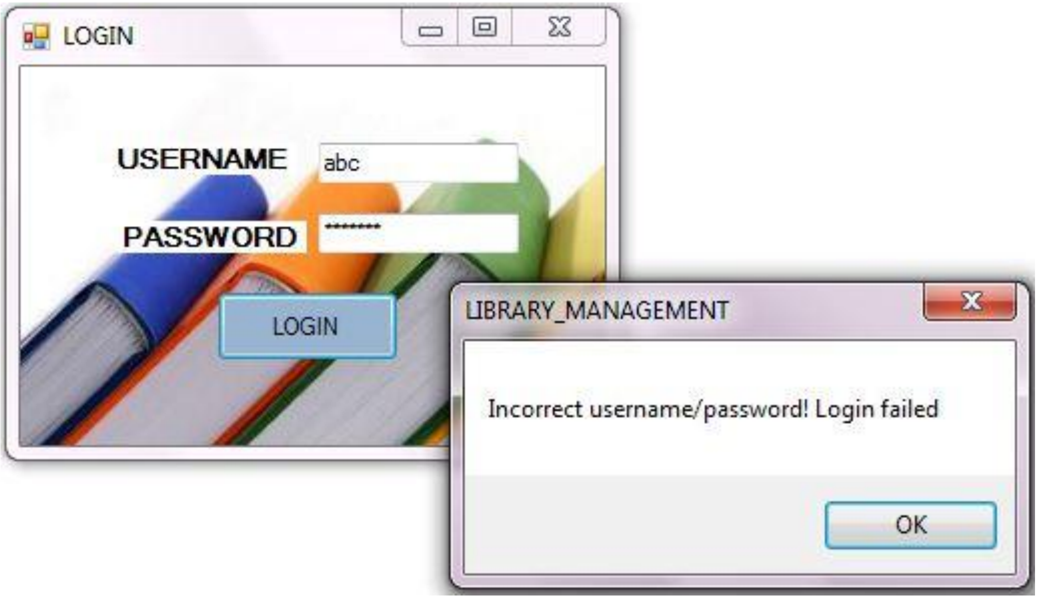
Test for Design Error

There is always a space for an experience tester to give apporiate suggestion on the design methodology adopted to develop the project.

8.2 Test Cases

Module Name	Test Case Paths	Test Cases	Output
Login	2 paths	a)correct input b)invalid input	a) Go to main form b)Incorrect username & password
Member	2 paths	a)correct input(entry of all fields) b) invalid/wrong input(blank fields)	a) Successful save/update b)Error message
Return/Renew Book	2 paths	a)correct input(book id) b)invalid input(book id)	a)Display the book details b)Error message(invalid book ID)
Return/Renew Book	2paths	a)renewal/return of book after due date b)return/renewal of book before due date	a)pay fine b)successful renewal/return of book
Book	2 paths	a)valid input(not a reference book) b) invalid(issue a reference book)	a) Issue the book b)error message(reference book cannot be issued)

Login



LOGIN

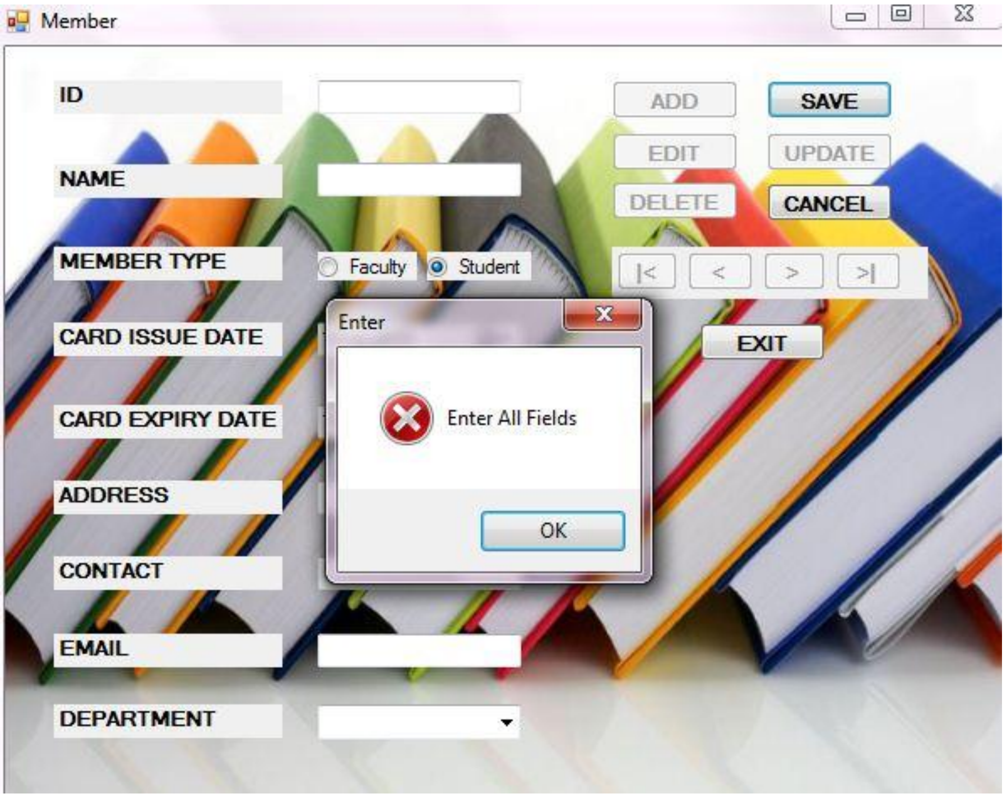
USERNAME abc

PASSWORD ********

LIBRARY_MANAGEMENT

Incorrect username/password! Login failed

Member



Member

ID

NAME

MEMBER TYPE ○ Faculty ● Student

CARD ISSUE DATE

CARD EXPIRY DATE

ADDRESS

CONTACT

EMAIL

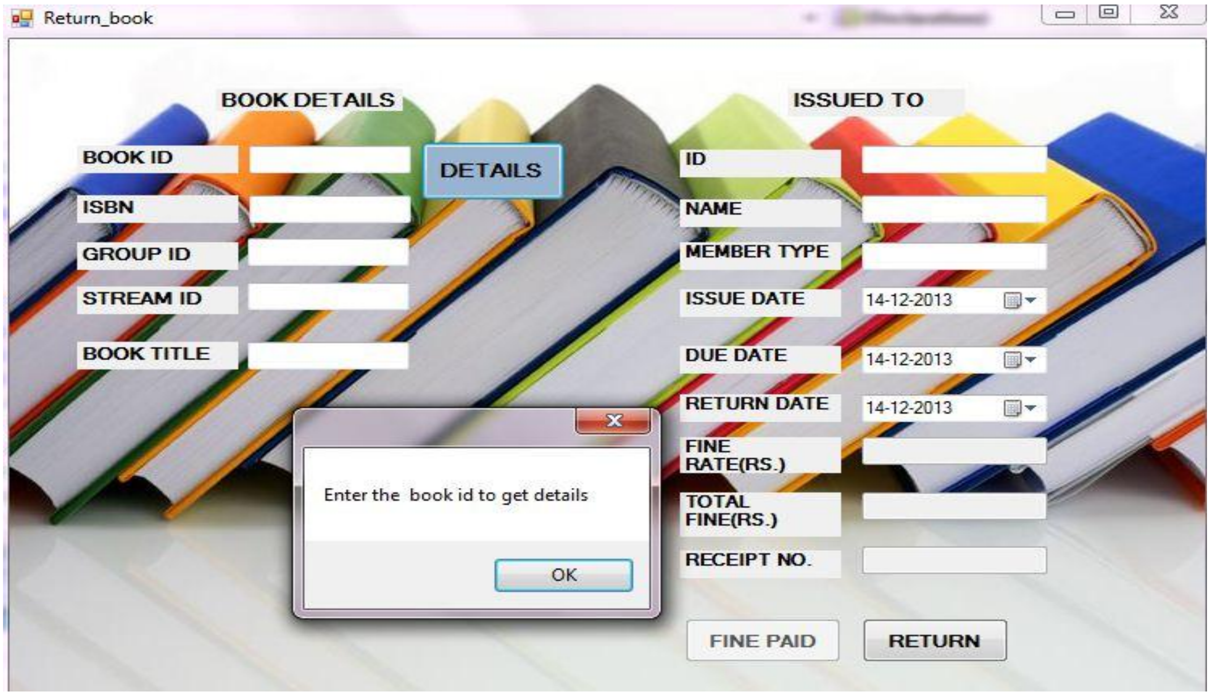
DEPARTMENT

Enter

Enter All Fields

(if any field is left blank, error message is displayed)

Return Book



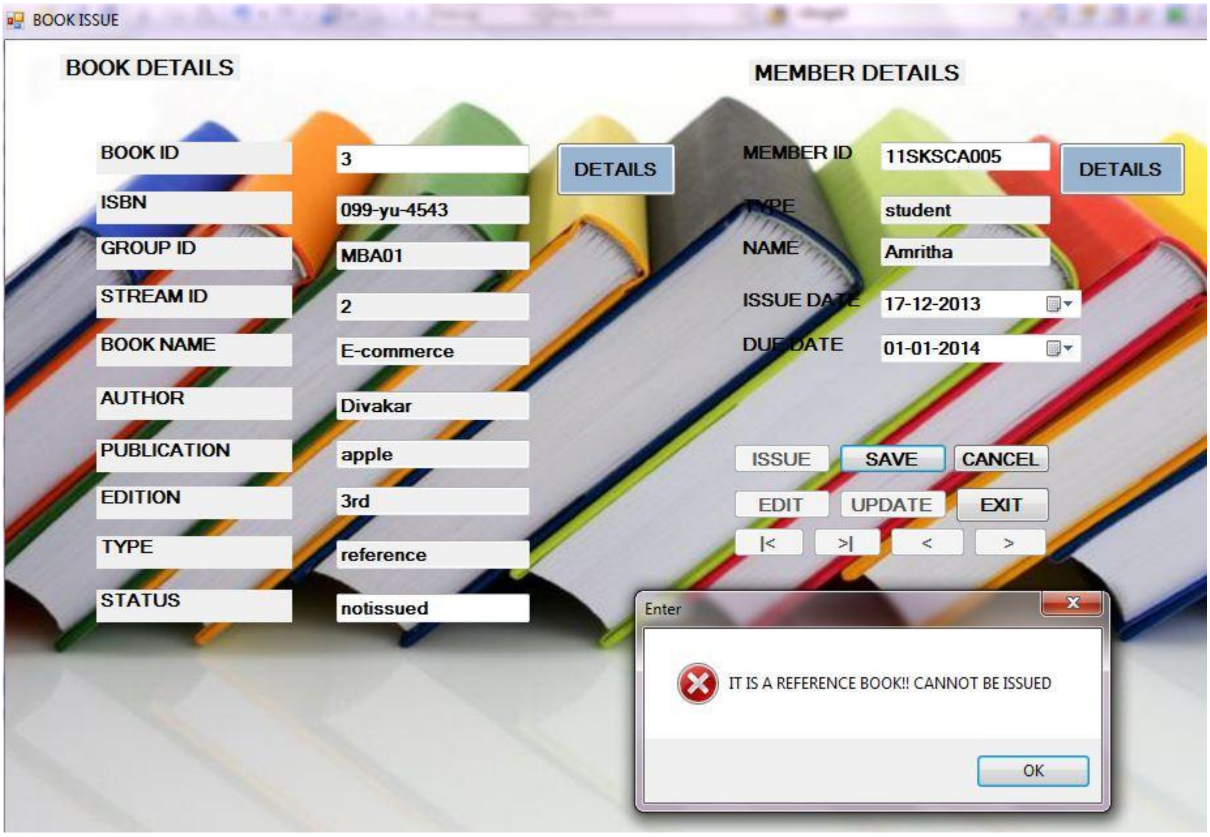
ISSUE DATE: 14-12-2013

DUE DATE: 14-12-2013

RETURN DATE: 14-12-2013

Enter the book id to get details

Book Details



BOOK ID: 3

ISBN: 099-yu-4543

GROUP ID: MBA01

STREAM ID: 2

BOOK NAME: E-commerce

AUTHOR: Divakar

PUBLICATION: apple

EDITION: 3rd

TYPE: reference

STATUS: notissued

MEMBER ID: 11SKSCA005

TYPE: student

NAME: Amritha

ISSUE DATE: 17-12-2013

DUE DATE: 01-01-2014

IT IS A REFERENCE BOOK!! CANNOT BE ISSUED

Chapter 9

Conclusion

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient GUI based component. This application is working properly and meeting to all user requirements. This component can be easily plugged in many other systems. The proposed project can be applied to any educational organization where the actual library of that particular organization can be link with the digital library, so as to provide book both offline and online. It provides an environment similar to a library where particular related books can be easily found, read, and easily downloadable and readable. Provides a huge range of subjective books specials for students, either technical or non-technical. Also provides social, economical, journals, magazines, etc. For specified group of peoples which can be studied.

Chapter 10

Future Enhancement

This application can be easily implemented under various situations. We can add new features as and when we require. Reusability is possible as and when required in this application. There is flexibility in all modules. This software is extendable in many ways. The following principles enhance extensibility like hide data structure, avoid traversing multiple links or methods, avoid case statements on object type and distinguish public and private operations.

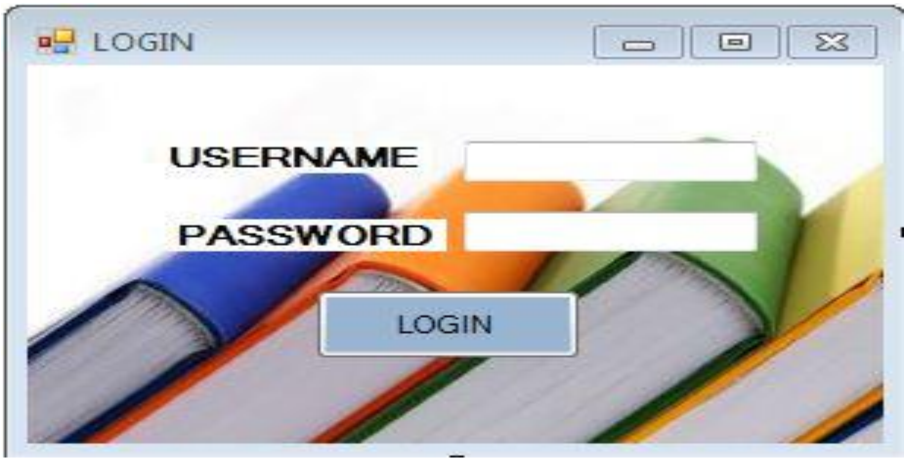
Reusability is possible as and when require in this application. We can update it to next version. Reusable software reduces design, coding and testing cost by amortizing effort over several designs. Reducing the amount of code also simplifies understanding, which increases the likelihood that the code is correct. We follow up both types of reusability: Sharing of newly written code within a project and reuse of previously written code on new projects.

Chapter 11

User Manuel

To operate the software user needs to follow the guidelines for effective use of the application.

How to Sign in



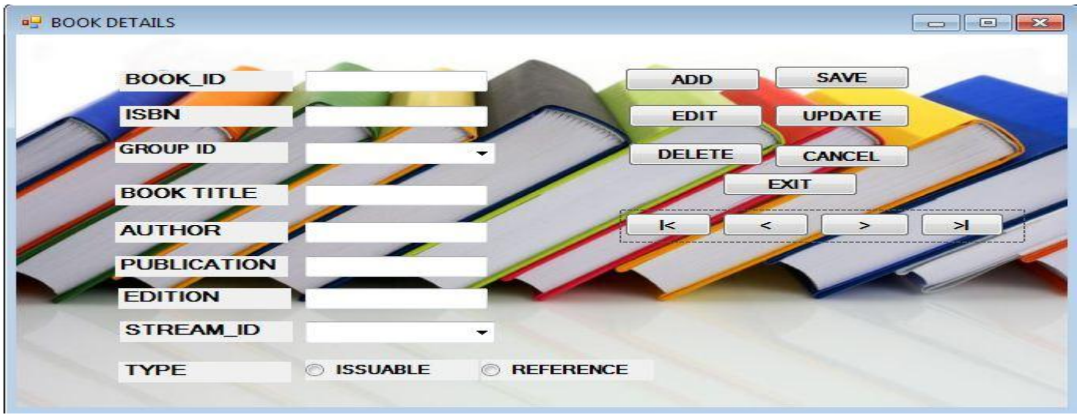
Click on the select in button.

-Enter user name.

-Enter password.

-Click on Login button.

Book Details

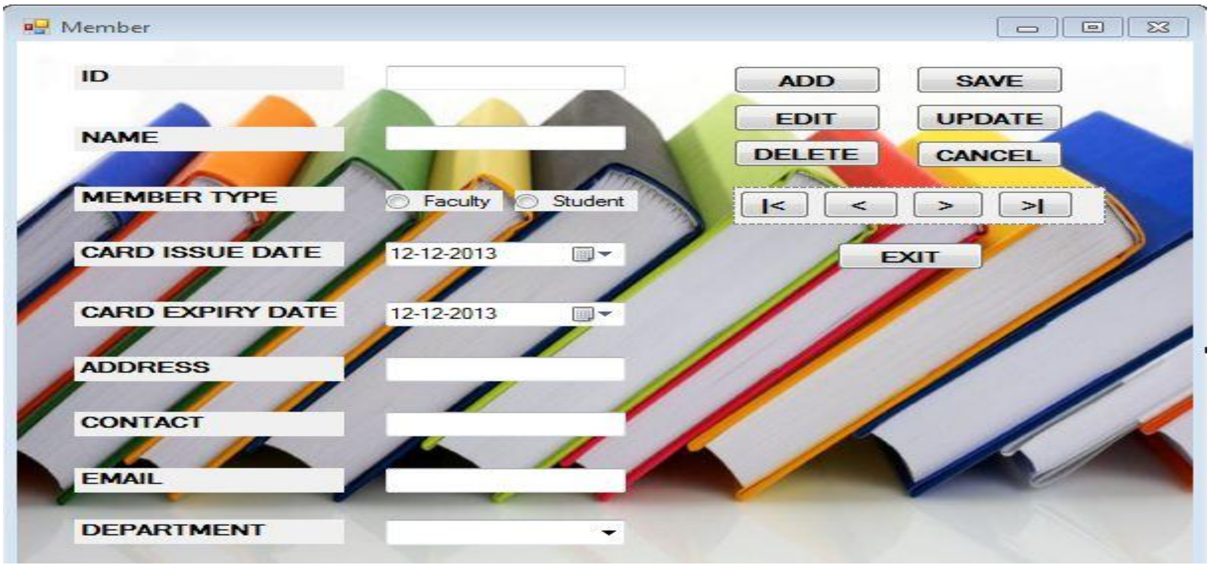


-Enter all required details.

-Select type

-Add, Save, Edit, Update.

Member



Member

ID

NAME

MEMBER TYPE ○ Faculty ○ Student

CARD ISSUE DATE 12-12-2013

CARD EXPIRY DATE 12-12-2013

ADDRESS

CONTACT

EMAIL

DEPARTMENT

-Enter all required details.

-Select Member type

-Add, Save, Edit, Update.