

**AN  
INTERNSHIP REPORT  
ON  
LIBRARY MANAGEMENT SYSTEM  
PROJECT  
BY  
KAMAL ACHARYA  
(Tribhuvan University)**

**Date: 2020/05/25**

# **1. INTRODUCTION**

## **1.1) Purpose**

The purpose of Library Management Pro system is to provide a medium for the public libraries to computerize their entire functioning and would contribute as a first step in digitalizing their libraries!

As till now in India, public libraries (medium and small scale) had their entire records on paper which again required maintenance and the problem of finding relevant information proved to be a herculean task. Even if the required information was found it was usually at cost of much valuable time.

Library Management Pro would not only simplify this process but also speedup the entire functioning of the library. It will also contribute towards increasing the efficiency of the library as a whole, right from the data entry to maintaining the historical records.

One major purpose is to provide user or the library members the opportunity to not only search through the books but to reserve books that are issued by other users and also request newer books.

## **1.2) Scope**

Library Management Pro (Limp) aims at providing a complete solution for all the library requirements of Medium and small scale libraries. Presently we have few public libraries in the country with even few of them computerized but the number is sure to grow as the focus shifts more on education. It would be a great help to these libraries if there is a software available that would help them with their day to day tasks.

Limp would not only help the librarian in the mundane tasks of the library but will be providing the scope to add newer user groups like data entry assistants etc. and also newer user groups depending on their subscriptions.

The feature of recording and retrieving user and book history details is going to throw some very interesting results regarding the books most popular among the users and also their favourite genre according to age group etc.

### **1.3) Problem in Existing System**

The existing system that is being used in majority of public libraries is completely manual in nature. Information about all the books and members/users is maintained separately in data entry registers. The entries made in each and every register are having a serial number corresponding to the register name and year. Also there are separate registers for purchase and sale of books and for different user plans. This often leads to redundant information too.

Though some libraries do have computerised systems that provide basic features such as adding books, user etc and also that of issuing the books, but what they lack is the user involvement and interactivity.

#### **Limitations of existing system:**

- Majority of libraries are dependent on paper-work which turns out to be very inefficient, and data backup is very difficult and tedious
- Users or members in such systems either become dependent on the librarian when they enquire about the books existing in the library or find themselves helpless when they try to search the library for their book of interest unknown of even its availability!!
- Present day systems involve the users very little in the entire process, and also do not consider the user wants for newer books etc. This non-involvement of end users, also mean that either they have to manually ask the librarian whether a particular book is already issued by some other user or they have to browse through the library.
- They are also unable to gather the information about user interests and books that are more popular than others.

## **1.4) Statement of Problem**

The current problem can be solved by automating the manual process followed in managing libraries with the help of our software. This would not only make the job of storing the data very easy and fast for the librarian but it would also make the process of retrieving data (regarding books, users etc.) in the future very convenient. It would also give users the facility to search the entire books of the library. He can also reserve books that he want to get issued (not currently available in the library).

## **2. SYSTEM REQUIREMENT ANALYSIS**

### **2.1. Information Gathering**

The informations have been gathered by conducting the surveys. Questionnaires and interviews with the librarians and users, are used to collect this information. Actual observation of work activities involved collection of forms and documents involved in the functioning of a library. Then we have studied the requirements to identify the features that our system should have and input and output methods.

#### **Determination of system requirements -**

- *Why such a system is required?*

Such a system is required to solve the problems of library management through a software solution, in an efficient manner, as well as to save the time. Also to result a systematic data storage system for the library.

- *If you have any previous version of this system then what are the problems in that?*

The previous versions of this system didn't have the flexibility to create newer user groups. Also they didn't involved the user dynamically.

- *What are your requirements?*

The major requirement is to maintain the information easily while saving time and effort. All the basic functions of the library will be done through the software. It should have user freindly interface and support fast retrieval of searched data. The proposed system must be faster so that the administrator does not has to wait long to retrieve any information. The system Reliability must be ensured so that there is no threat of data loss. It should be Scalable so that in near future the system can be expanded and customized to the growing needs of the library.

- *How to proceed with the problem solving?*

By analysing the requirements gathered.

## **2.2. Technology Specifications**

### **Hardware Requirements**

- CPU equivalent or above Intel Celeron 800 MHz
- RAM 256 MB or higher.
- Hard Disk – Minimum 4GB

### **Software Requirements**

- .Net framework 2.0 or higher.
- Any Windows OS (XP or later)
- Microsoft SQL 2005 as the data base.
- Microsoft Word for Documentation.

### **Implementation Language**

- Visual Basic.Net 2008

## **2.3. System Feasibility**

### **2.3.1. Technical Feasibility**

The technical feasibility of the system counts for the technical acceptance of the system. It refers to the ability of the process to take advantage of the current state of the technology in pursuing further improvement. The technical capability of the personnel as well as the capability of the available technology should be considered.

In technical feasibility the following issues are taken into consideration:

- *Whether the required technology is available or not?*

The work for the project can be done with the current equipment and existing software technology that the organisation possesses. .Net is used as a main technology which is easy to use.

- *Whether the required resources are available?*

The system does not have any rigid hard-ware and software requirements and there is availability of the people who can perform the software engineering activities required for the development of the system.

Hence, the system is technically feasible.

### **2.3.2. Behavioral Feasibility**

Behavioral feasibility is the measure that how effective the user uses the system. The behavioral efficiency is one of the major factors of feasibility analysis. The new or the proposed system should be easy to operate, convenient in maintenance and effective in its working. Thus behavioral feasibility is very important factor to be considered for effective working of system. Behavioral feasibility is dependent on human resources available for the project and involves projecting whether the system will operate and be used when installed. The system is behaviorally feasible if it fulfills the following:

1. The proposed system is easy to operate.
2. Existing members are not affected in anyway.
3. Retrieval of information is easy, accurate and fast.

### **2.3.2. Temporal Feasibility**

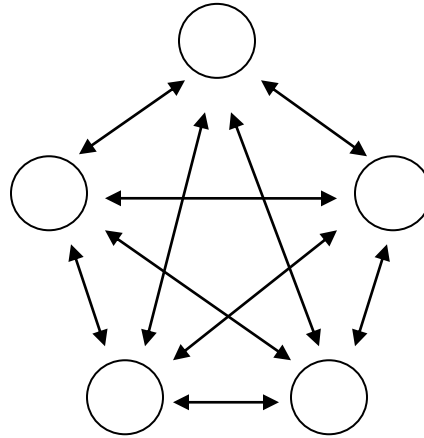
Temporal feasibility means whether the project is completed in the given time or not.

One of the most difficult aspects of project management is the formulation of estimates of the time required to develop a system. Estimates and time scheduling is an approximation of the hours, days, or months of efforts needed to produce the desired system. It depends on factors like programmer's ability, program complexity etc.

## **2.4) Team structure**

Our team structure is Democratic decentralized (DD) structure. This software engineering team has no permanent leader. Rather, "task coordinators are appointed for short durations and then replaced by others who may coordinate different tasks." Decisions on problems and approach are made by group consensus. Communication among team members is horizontal.

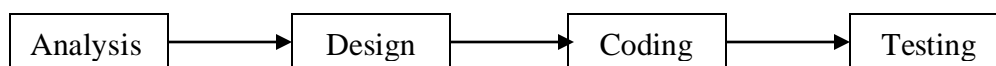
It has got an egoless approach in which each and every member is involved throughout the development of the software and also in the decision making process. The structure results in many communication paths between people.



*Fig 1: Communication Paths in an Egoless Team Structure*

## 2.5 Process Model Used

A software process model is a development strategy that incorporates the development process, methods and tools used to design software. It is chosen based on the nature of the software and the methods and tools used in development. The Software model used in the Project is **Linear Sequential Model**. It is also called the "Classic Life Cycle" or "the "Waterfall Model" that suggests a systematic and sequential approach to software development that begins at the system level and progresses through analysis, design, coding, testing and support. The waterfall model derives its name due to the cascading effect from one phase. In this model each phase has a well defined starting and ending point, with identifiable deliveries to the next phase.



*Fig. 2 - The Linear Sequential Model*



### **3. ANALYSIS**

#### **3.1 Methodology Used**

The project plan is to carry the design and implementation of the project in a completely step-by-step manner. The entire project is divided in the following phases.

##### **Phase 1 – Study and Analysis Phase**

In this phase requirements were analyzed and a detailed study regarding the functionalities of the libraries was performed. In this phase we interacted with the librarians and users so as to collect information regarding the project. A study of the drawbacks of the existing system is required and proposing the system as a solution to them. Also a deep study of the concept of databases and other material is required to design and develop the system.

##### **Phase 2 - Design Phase**

In this phase the database design of the system are made. After the analysis of the system the scenario of the library is to be defined. The database design is being carried in the following steps-

- Identification of entities and their relationships from the scenario.
- Designing the conceptual model.
- Designing the logical model of the system and normalizing the relations.
- Proposing the physical design of the system.
- Working on the interface design.
- Defining hardware and software requirements.

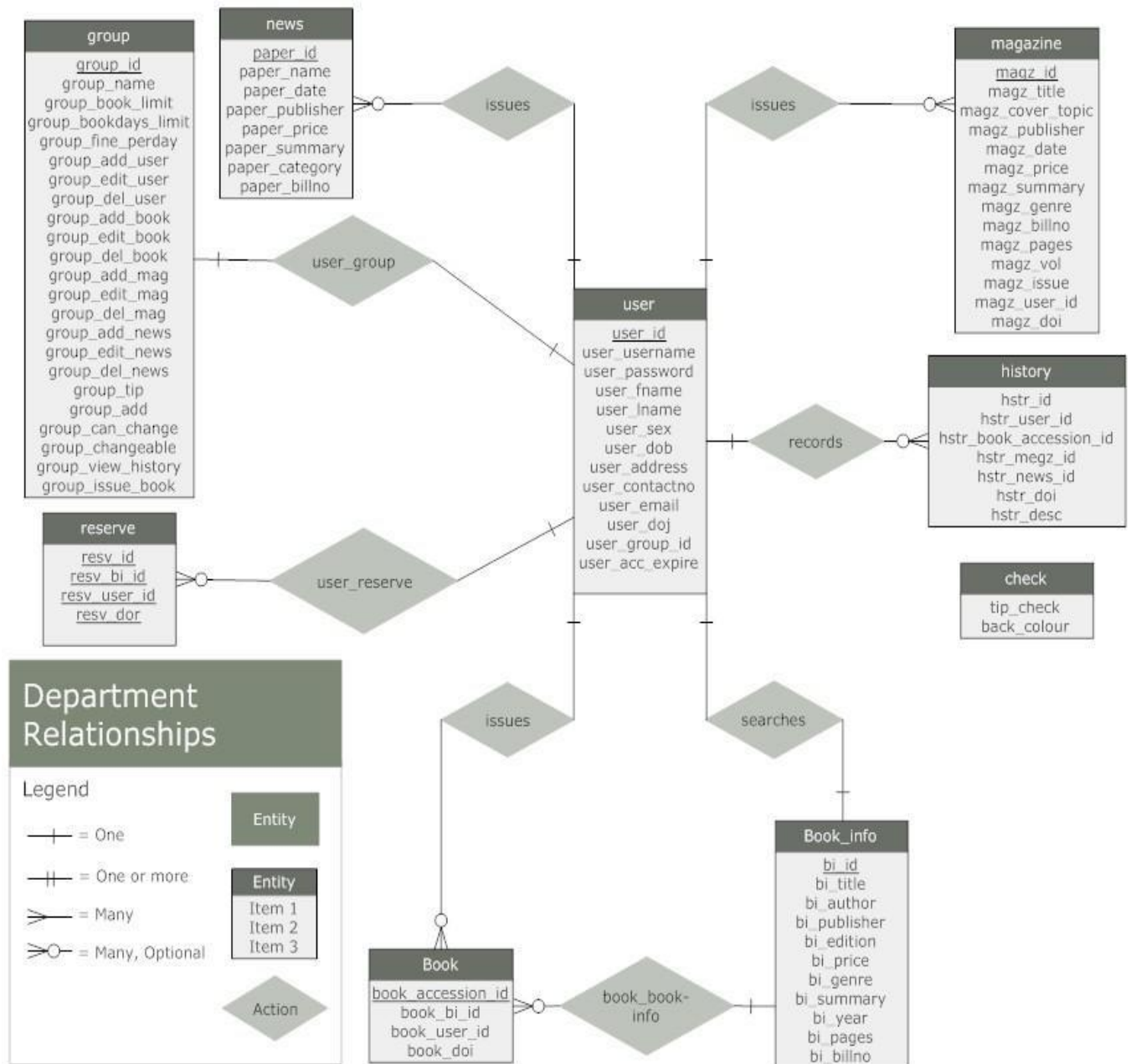
##### **Phase 3 - Coding Phase**

In this phase the design of the system is to be implemented through actual code. Code allows the developers to make the computer behave in a required manner and thus satisfying the needs of the end user.

#### **Phase 4 – Testing and Implementation**

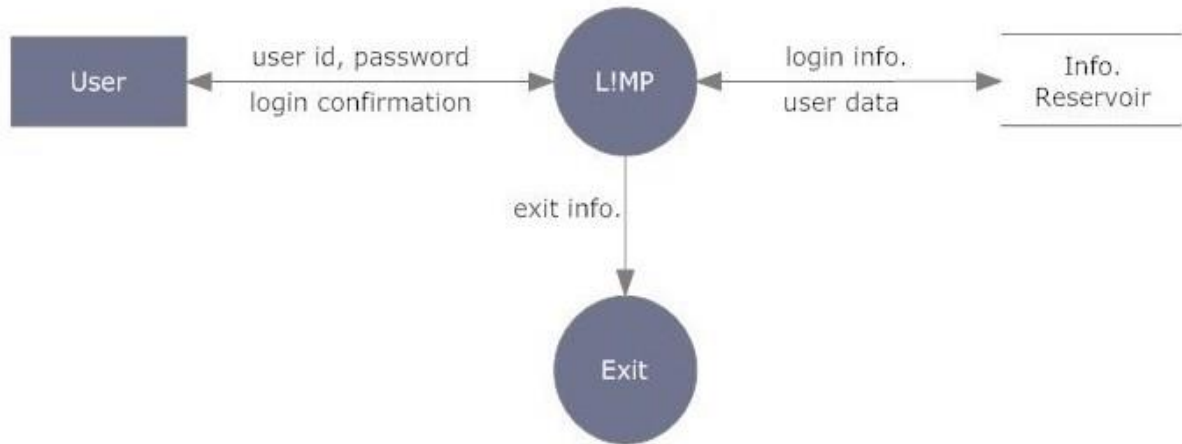
This phase will involve testing it with various test cases and data sets and implementation of the system. Testing is the stage where all possibilities of software failure are discovered and bugs are removed. After successful testing of the software implementation is done. Implementation implies the deployment of the software to the client location.

### 3.2) ER Model

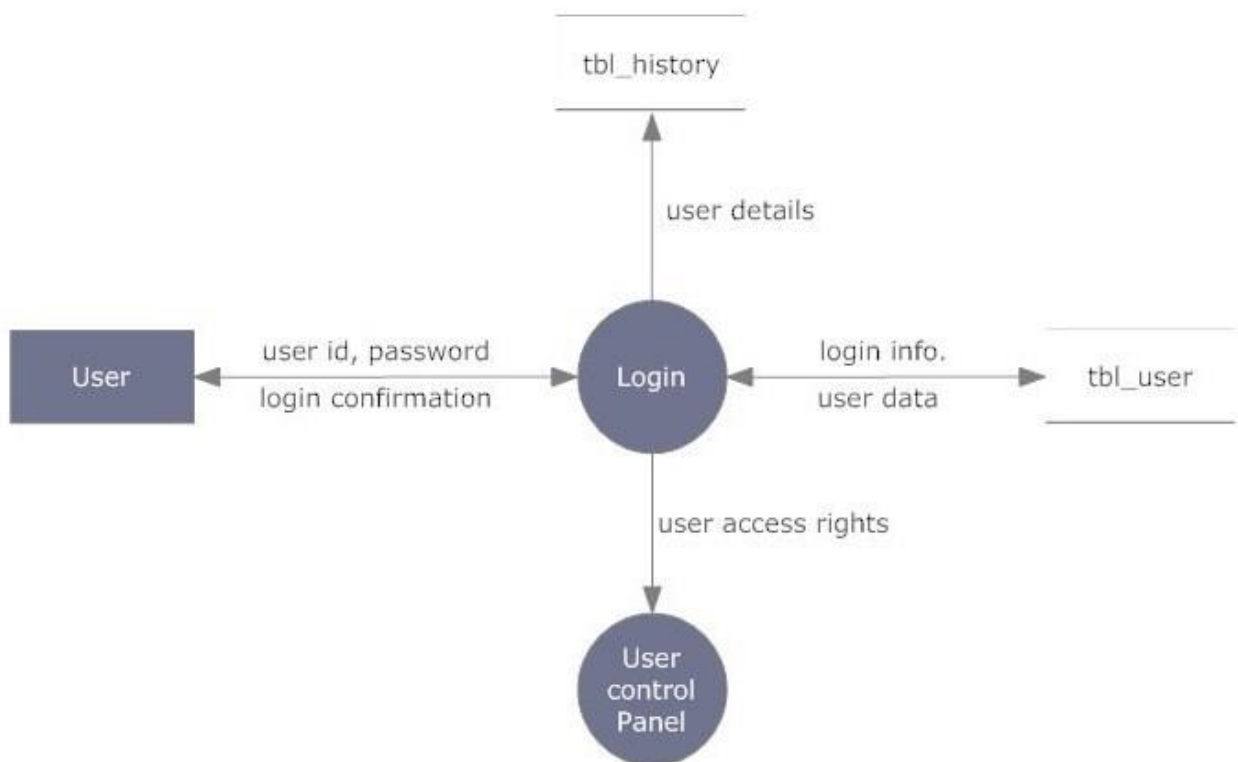


**Fig 3**

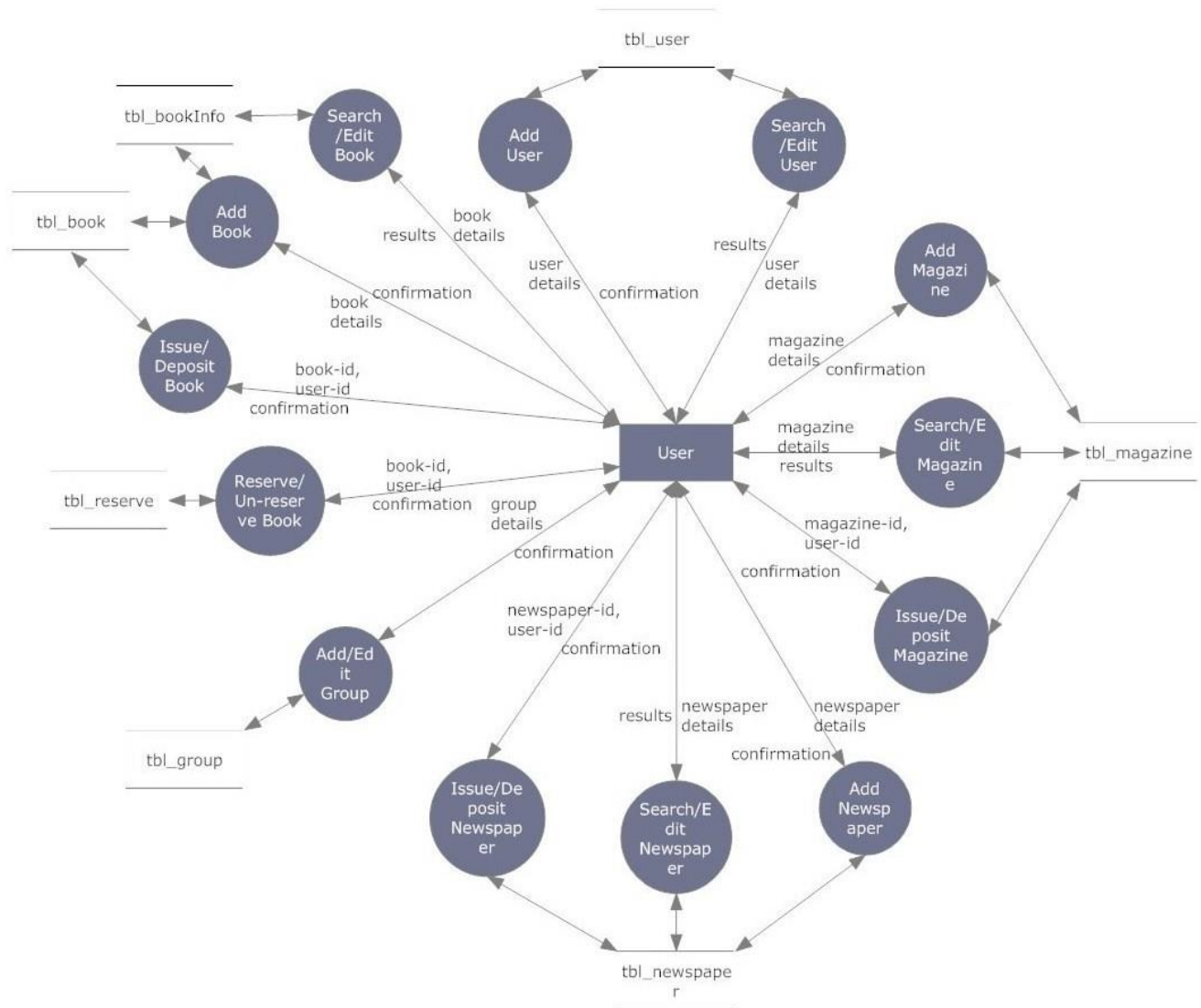
### 3.3) DFD- Data Flow Diagram



*Fig 4: Level 0 DFD*

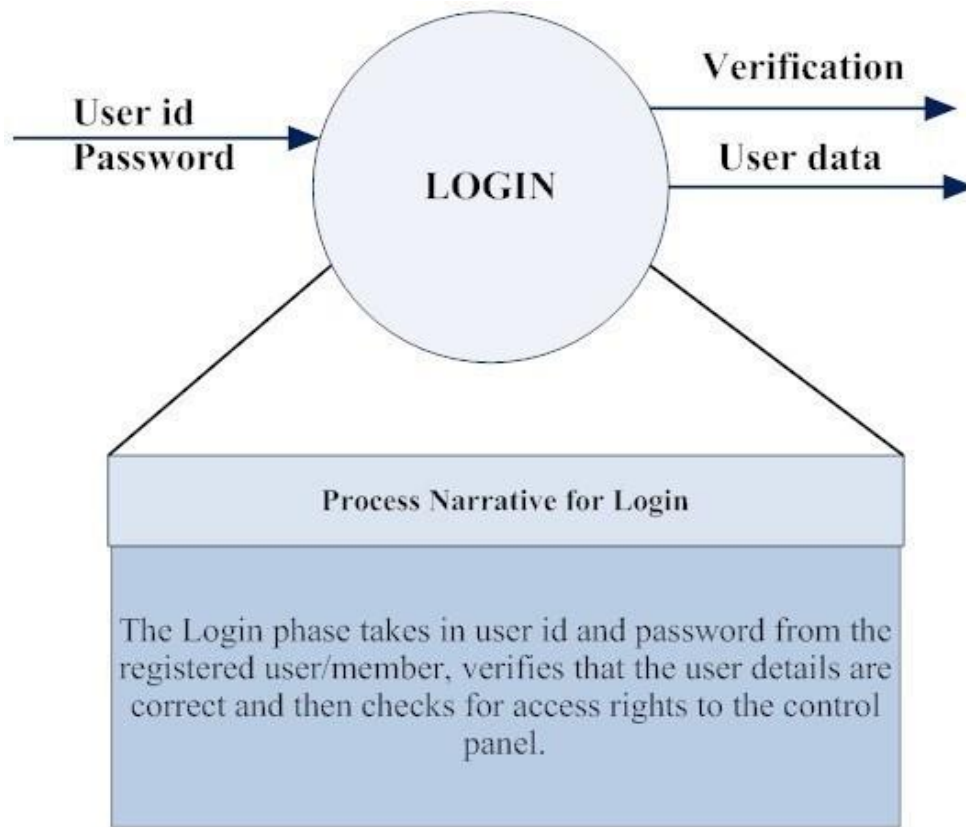


*Fig 5: Level 1 DFD*

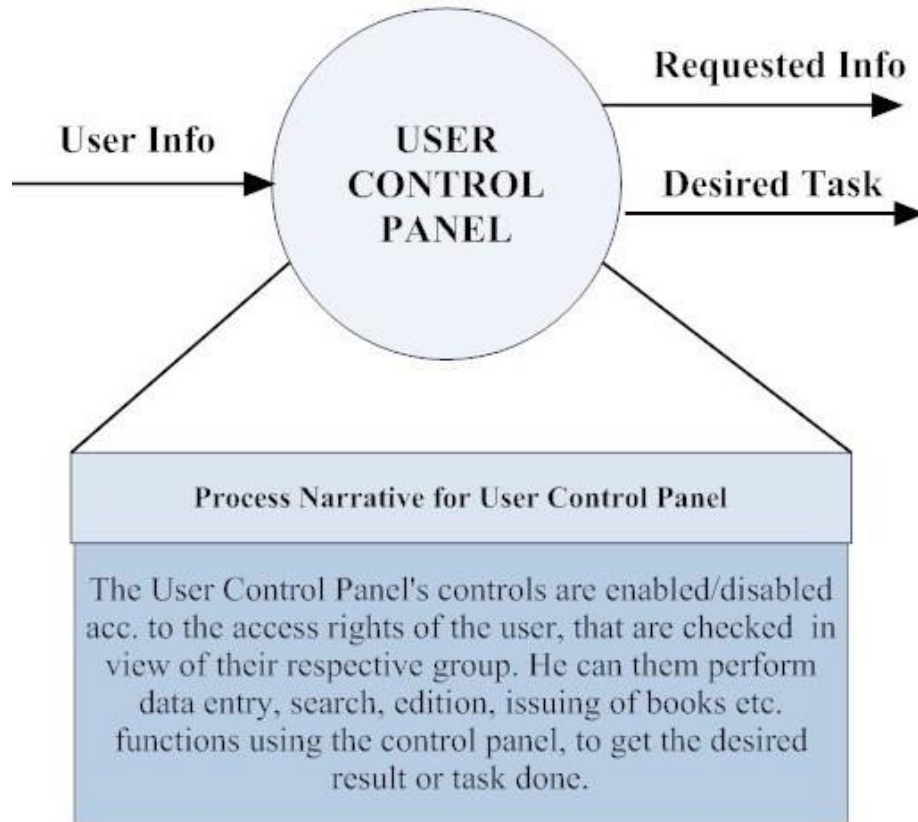


**Fig 6: Level 2 DFD**

### 3.4) Process Specification

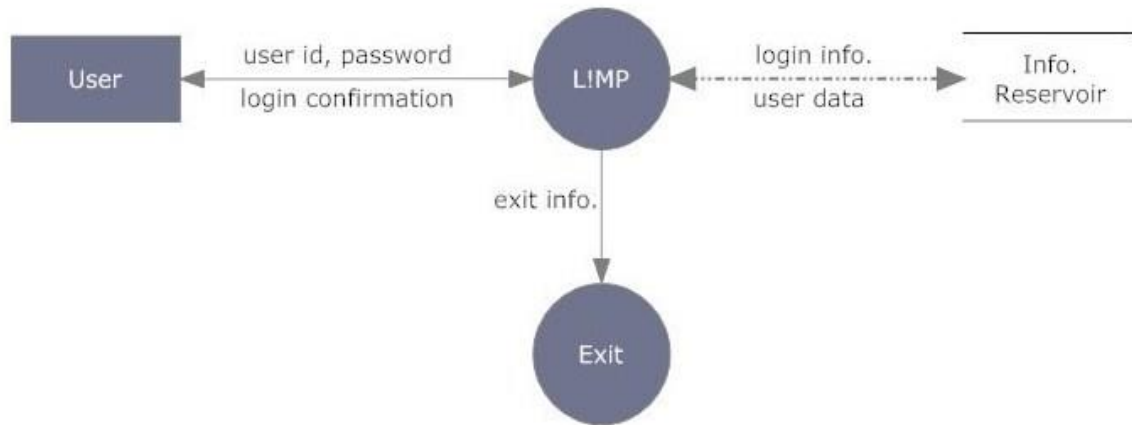


***Fig 7: Process Specification for Login***

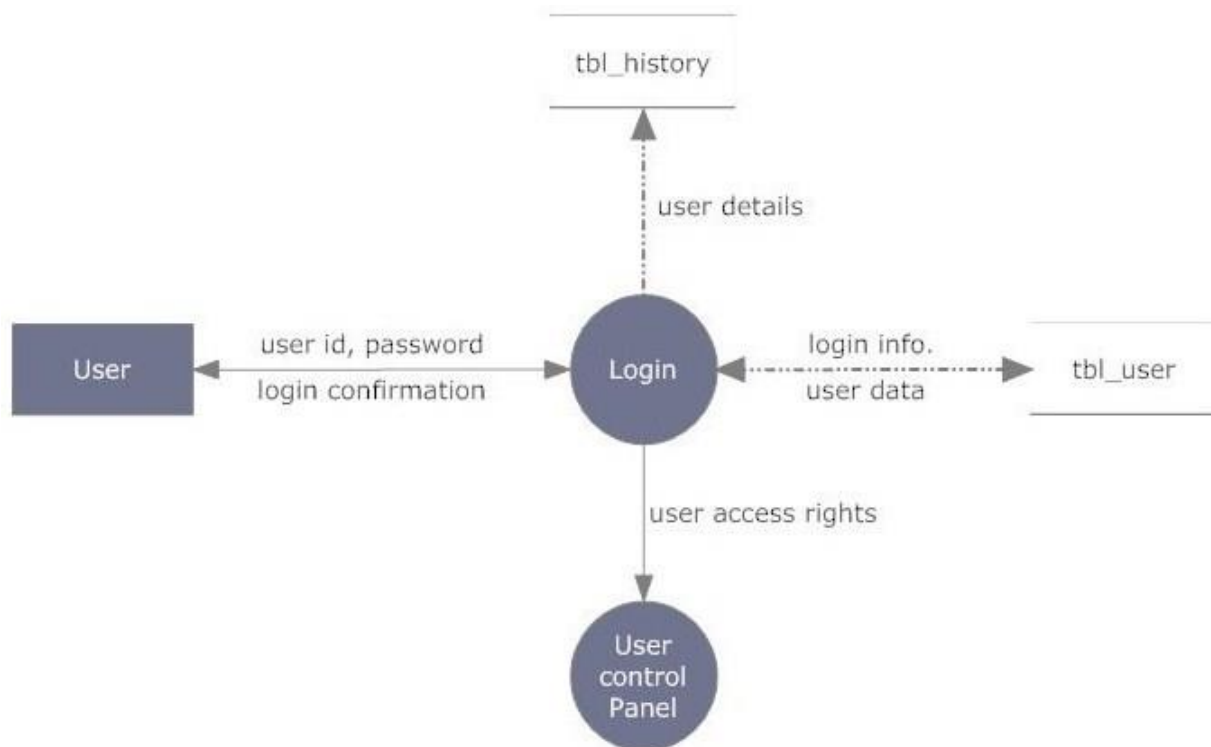


***Fig 8: Process Specification for User Control Panel***

### **3.5 CFD- Control Flow Diagram**

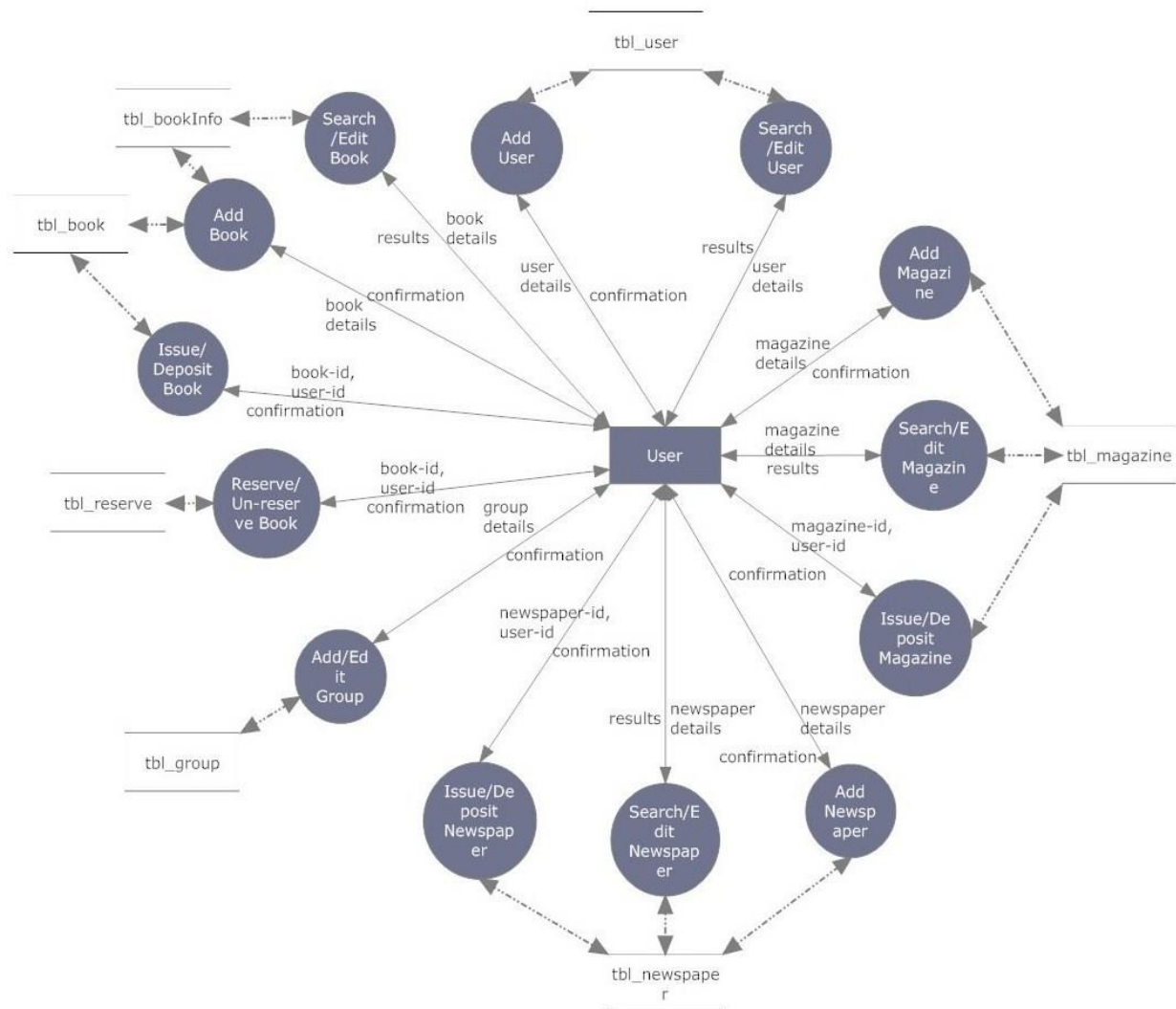


***Fig 9: Level 0 CFD***



***Fig 10: Level 1 CFD***



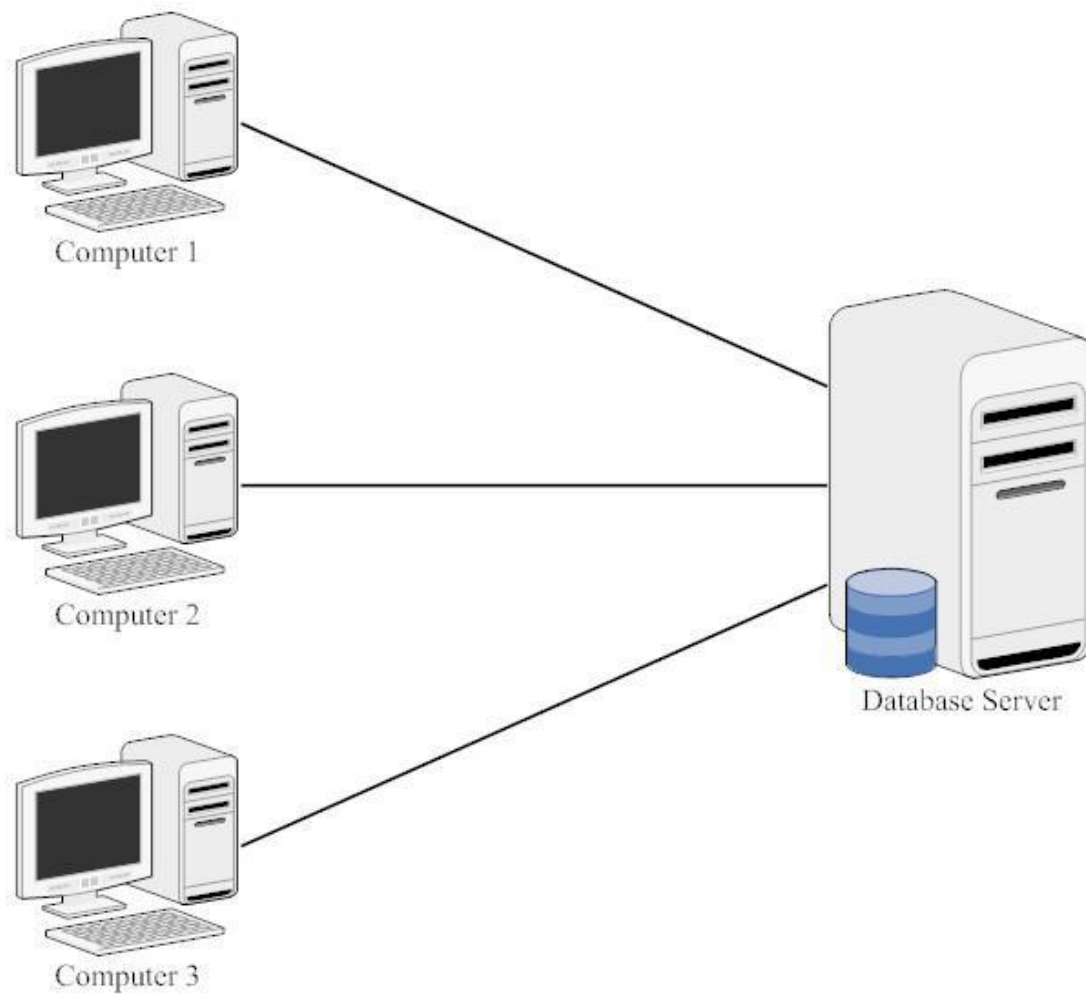


**Fig 11: Level 2 CFD**

## 4. DESIGN

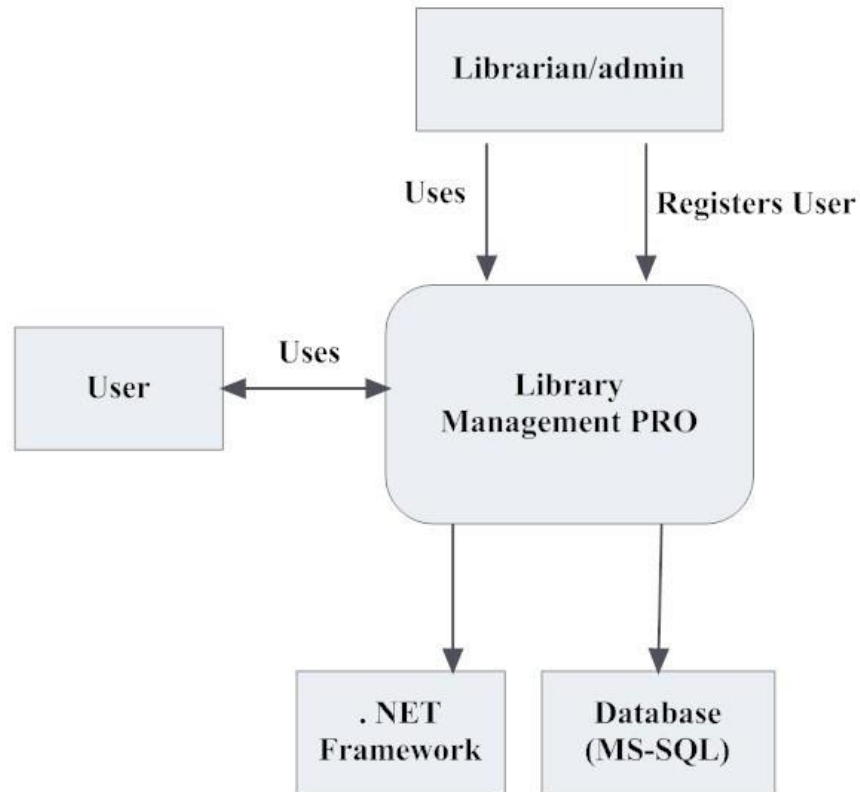
### 4.1 Architectural Design

#### 4.1.1) System Architecture Diagram



*Fig 12: Two-Tier Architecture*

#### 4.1.2) Architecture Context Diagram



*Fig 13: Architectural Context Diagram*

#### 4.1.3) Description Of Architectural Design

Two-tier architecture is where a client talks directly to a server, with no intervening server. It is simple to build. In our project this architecture is implemented because the client that is the user sends a request to the server that the server fulfill by simply referring to the database and retrieving the information asked for.

The most important limitation of the two-tier architecture is that it is not scalable, because each client requires its own database session that is 2-tier applications become complex and hard to support as the number of users increases in size.

## 4.2 Database Design

### 4.2.1) Data Dictionary

**Table name: user**

Column Name	Data Type	Length
user_id	int	50
user_username	varchar	50
user_password	varchar	50
user_fname	varchar	50
user_lname	varchar	50
user_sex	varchar	1
user_dob	datetime	
user_address	varchar	200
user_contactno	varchar	15
user_email	varchar	30
user_doj	datetime	
user_group_id	int	50
user_acc_expire	datetime	

*Table – 1*

**Table name: book\_info**

Column Name	Data Type	Length
bi_id	int	50
bi_title	varchar	50

bi_author	varchar	50
bi_publisher	varchar	50
bi_edition	varchar	50
bi_price	int	
bi_genre	varchar	100
bi_summary	varchar	MAX
bi_year	Number	int
bi_pages	int	50
bi_billno	int	50

***Table – 2***

**Table name: book**

Column Name	Data Type	Length
book_accession_id	int	50
book_bi_id	int	50
book_user_id	int	50
book_doi	datetime	

***Table – 3***

**Table name: group**

<b>Column Name</b>	<b>Data Type</b>	<b>Length</b>
group_id	int	Long Integer
group_name	varchar	50
group_book_limit	int	50
group_bookdays_limit	int	50
group_add_user	varchar	1
group_edit_user	varchar	1
group_del_user	varchar	1
group_add_book	varchar	1
group_edit_book	varchar	1
group_del_book	varchar	1
group_add_mag	varchar	1
group_edit_mag	varchar	1
group_del_mag	varchar	1
group_add_news	varchar	1
group_edit_news	varchar	1
group_del_news	varchar	1
group_tip	varchar	1
group_add	varchar	1
group_can_change	varchar	1

group_changeable	varchar	1
group_view_history	varchar	1
group_issue_book	varchar	1

**Table – 4**

**Table name: history**

Column Name	Data Type	Length
hstr_id	int	50
hstr_user_id	int	50
hstr_book_accession_id	int	50
hstr_magz_id	int	50
Hstr_news_id	int	50
hstr_doi	datetime	
hstr_desc	varchar	50

**Table 5**

**Table name: magazine**

Column Name	Data Type	Length
magz_id	int	50
magz_tilte	varchar	50
magz_cover_topic	varchar	50
magz_publisher	varchar	50
magz_date	datetime	

magz_price	int	
magz_summary	text	
magz_genre	varchar	100
magz_billno	int	
magz_pages	int	
magz_vol	int	
magz_issue	int	
magz_user_doi	int	
magz_doi	datetime	

***Table 6***

**Table name: newspaper**

<b>Column Name</b>	<b>Data Type</b>	<b>Length</b>
paper_id	int	
paper_name	varchar	50
paper_date	datetime	
paper_publisher	varchar	50
paper_price	int	
paper_summary	varchar	50
paper_category	varchar	50
paper_billno	int	
paper_user_id	int	
paper_doi	datetime	



***Table 7***

**Table name: reserve**

Column Name	Data Type	Length
resv_id	int	50
resv_bi_id	int	50
resv_user_id	int	50
res_doi	datetime	

***Table – 8***

### **4.2.2 Normalization**

#### *First Normal Form (1NF)*

First normal form (1NF) sets the very basic rules for an organized database:

- Eliminate duplicative columns from the same table.
- Create separate tables for each group of related data and identify each row with a unique column or set of columns (the primary key).

#### *Second Normal Form (2NF)*

Second normal form (2NF) further addresses the concept of removing duplicative data:

- Meet all the requirements of the first normal form.
- Remove subsets of data that apply to multiple rows of a table and place them in separate tables.
- Create relationships between these new tables and their predecessors through the use of foreign keys.

### *Third Normal Form (3NF)*

Third normal form (3NF) goes one large step further:

- Meet all the requirements of the second normal form.
- Remove columns that are not dependent upon the primary key.

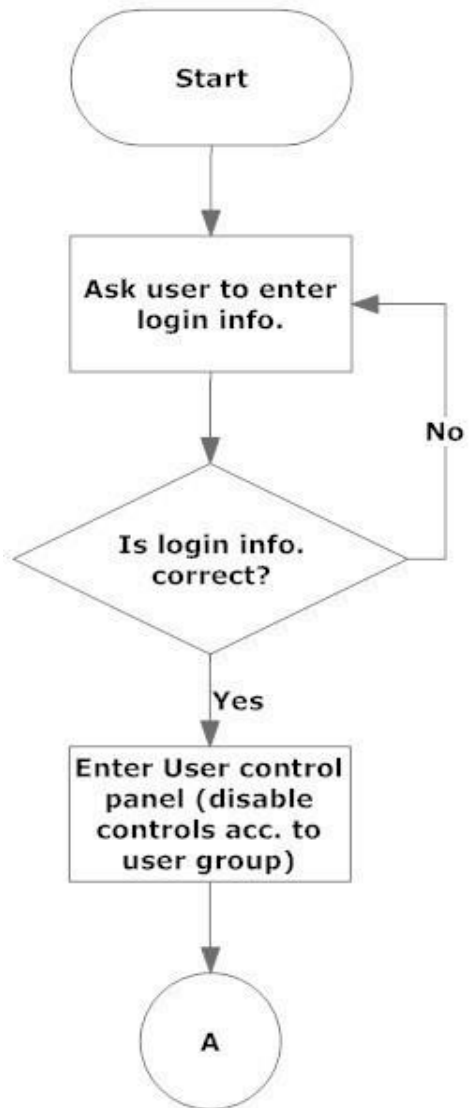
### *Fourth Normal Form (4NF)*

Finally, fourth normal form (4NF) has one additional requirement:

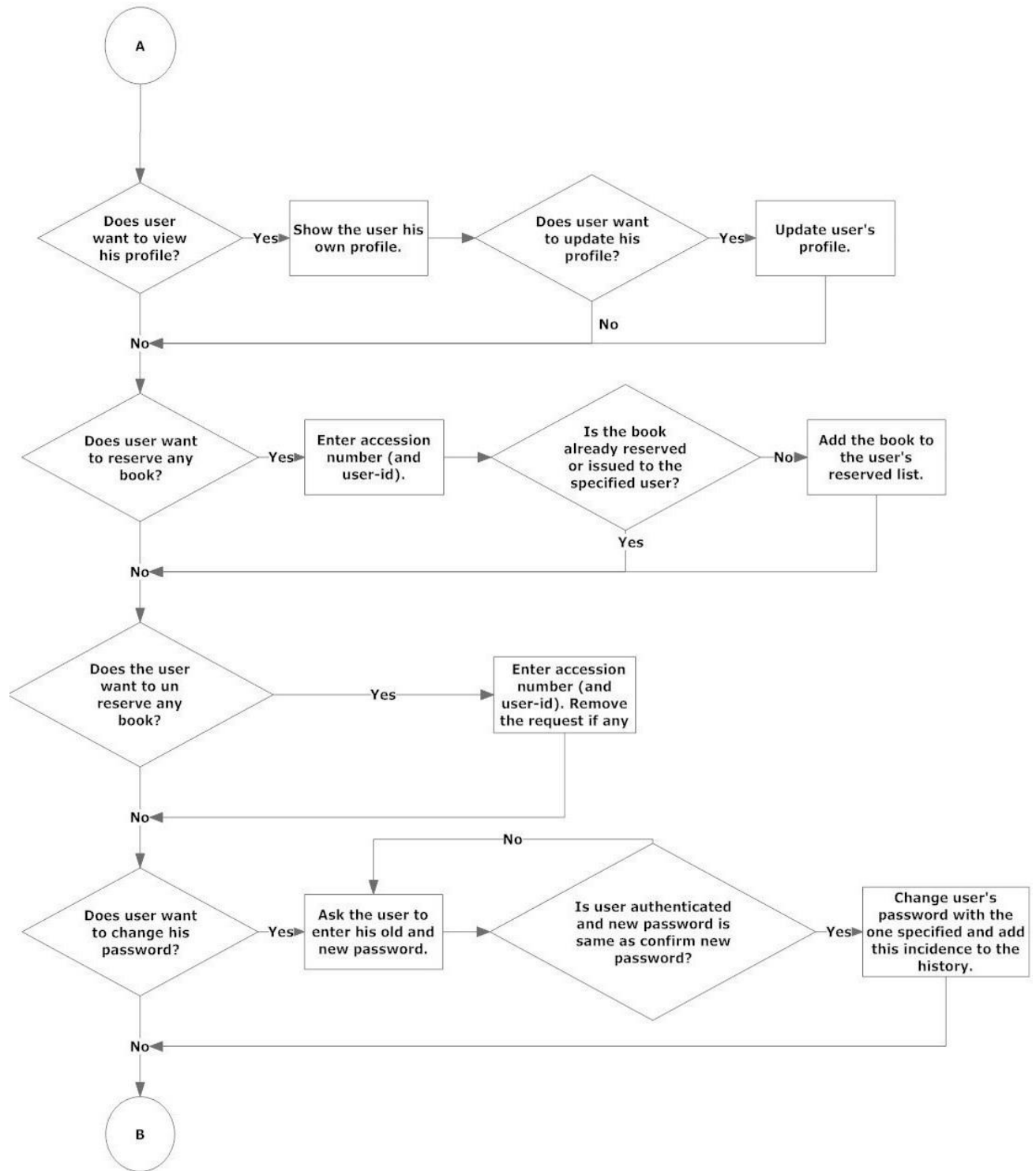
- Meet all the requirements of the third normal form.
- A relation is in 4NF if it has no multi-valued dependencies.

## 4.3 Component Design

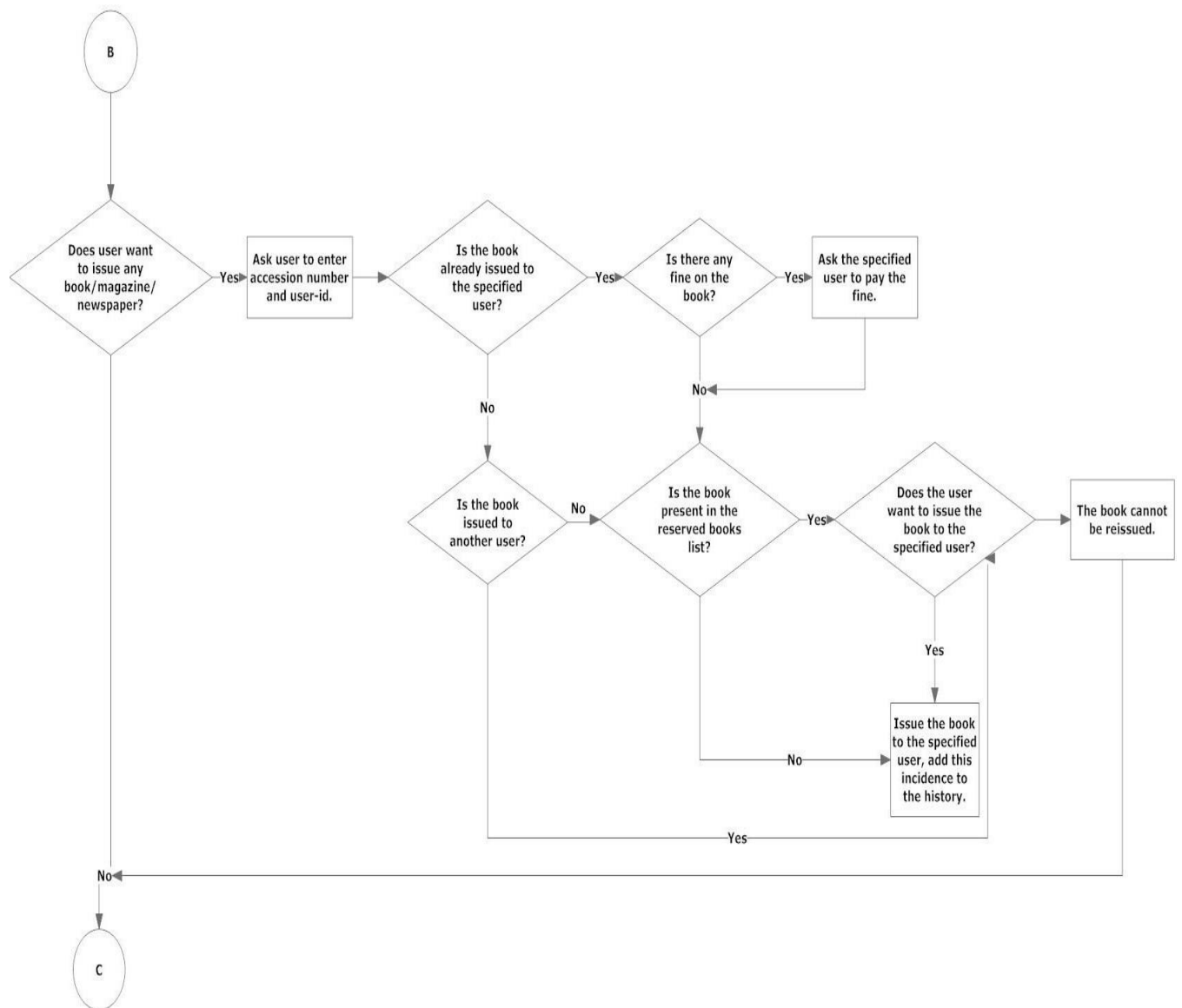
### 4.3.1 Flow Chart



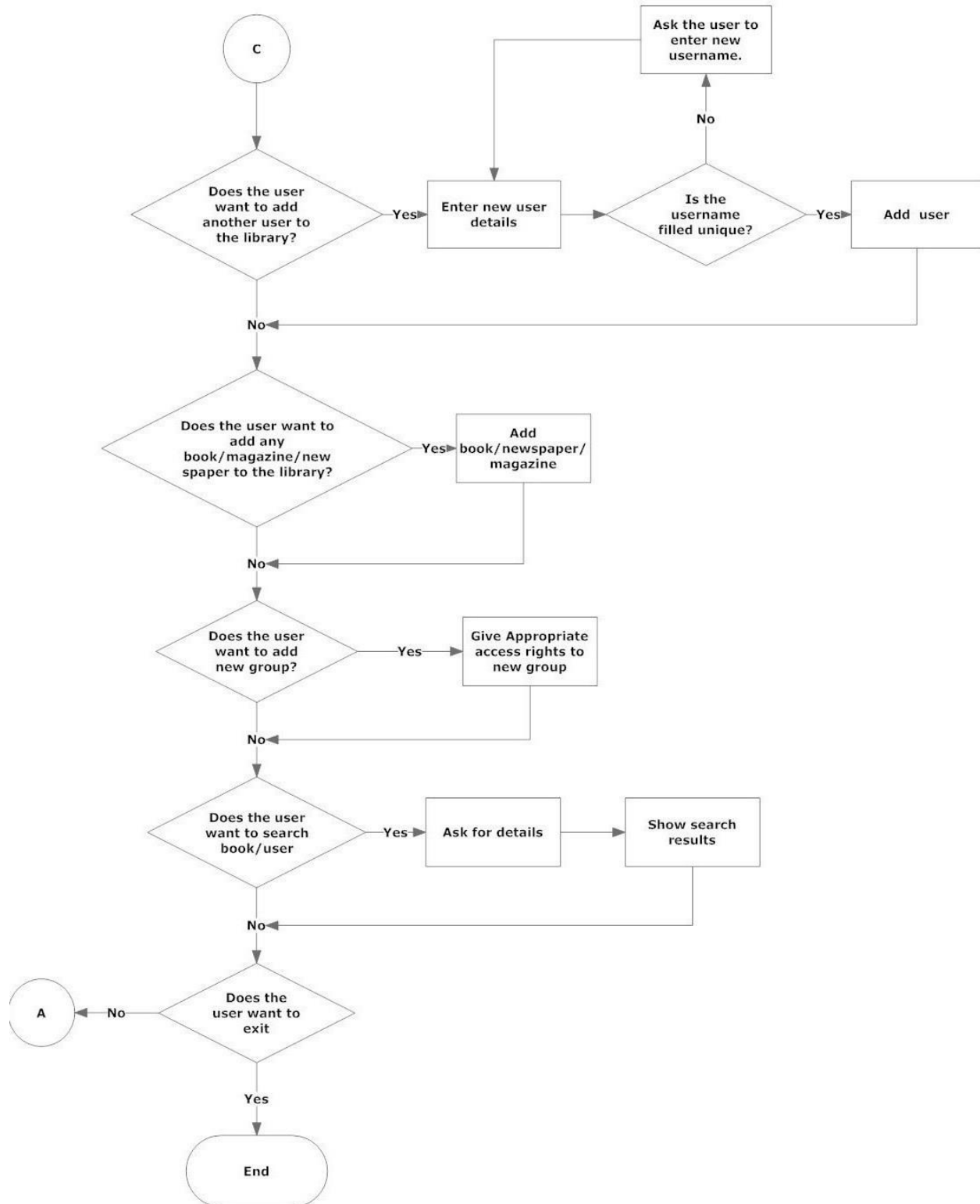
*Fig 14(a)*



**Fig 14(b)**



**Fig 16(c)**



**Fig 14(d)**

## 4.4 Interface Design

### 4.4.1 Screenshots



*Login Screen*



### *Home Screen*



Welcome to Library Management Pro

General Issue/Deposit Make Data Entry Search/Edit History Group Help

**My Profile**

Welcome adi !

First Name:

Last Name:

Address:

Contact no.:

E-mail id:

**Book I Have**

Book_Title	Book_Author	book_doi	ReturnDate	Publisher	Editor
Mastering C++	Robert Lafore			SAMS	Fifth
Mastering VB.net	GD Singh	12/12/2009	3/12/2010	Singh and sons	first
versace	gd	12/12/2006	3/12/2007	gd and sons	first

**Fines Due**

Book_Title	Book_Author	book_doi	ReturnDate	FineDue	Book_Pub
versace	gd	12/12/2006	3/12/2007	796	gd and sons

5/16/2009 11:18:13 PM

## *User Profile*

**Book Search**

Book Title:  Book Publisher:

Book Author:  Genre:

Summary:

☐ Price From  To  Bill No.

☐ Year From  To

☒ Find for all ☐ For for any

	bi_title	bi_author	bi_publisher	bi_edition	bi_price	bi_genre
▶	Mastering C++	Robert Lafore	SAMS	Fifth	695	Computer, Programr

### ***Book Search***

**Book History**

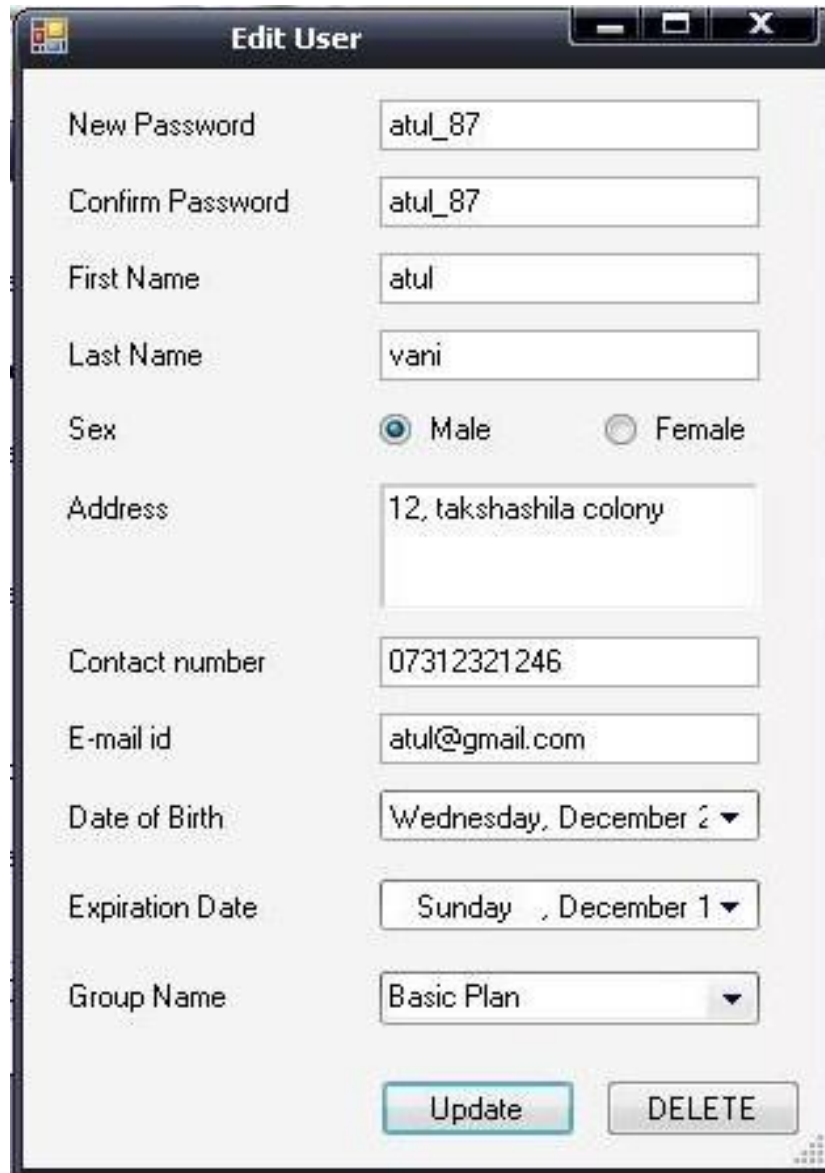
Accession No.

☐ Include Date Time From  To

Activity Over Last 24 Hours Total number of books issued yet = 1

	Username	Accession_No	Title	Date_of_Activity	Description
▶	adi	1	Mastering C++	5/12/2009 9:51 ...	Book Deposited
	adi	1	Mastering C++	5/12/2009 7:07 ...	Book Deposited
	adi	1	Mastering C++	5/12/2009 7:06 ...	Book Deposited
	adi	1	Mastering C++	5/12/2009 7:06 ...	Book Relssued
*					

### ***Book History***

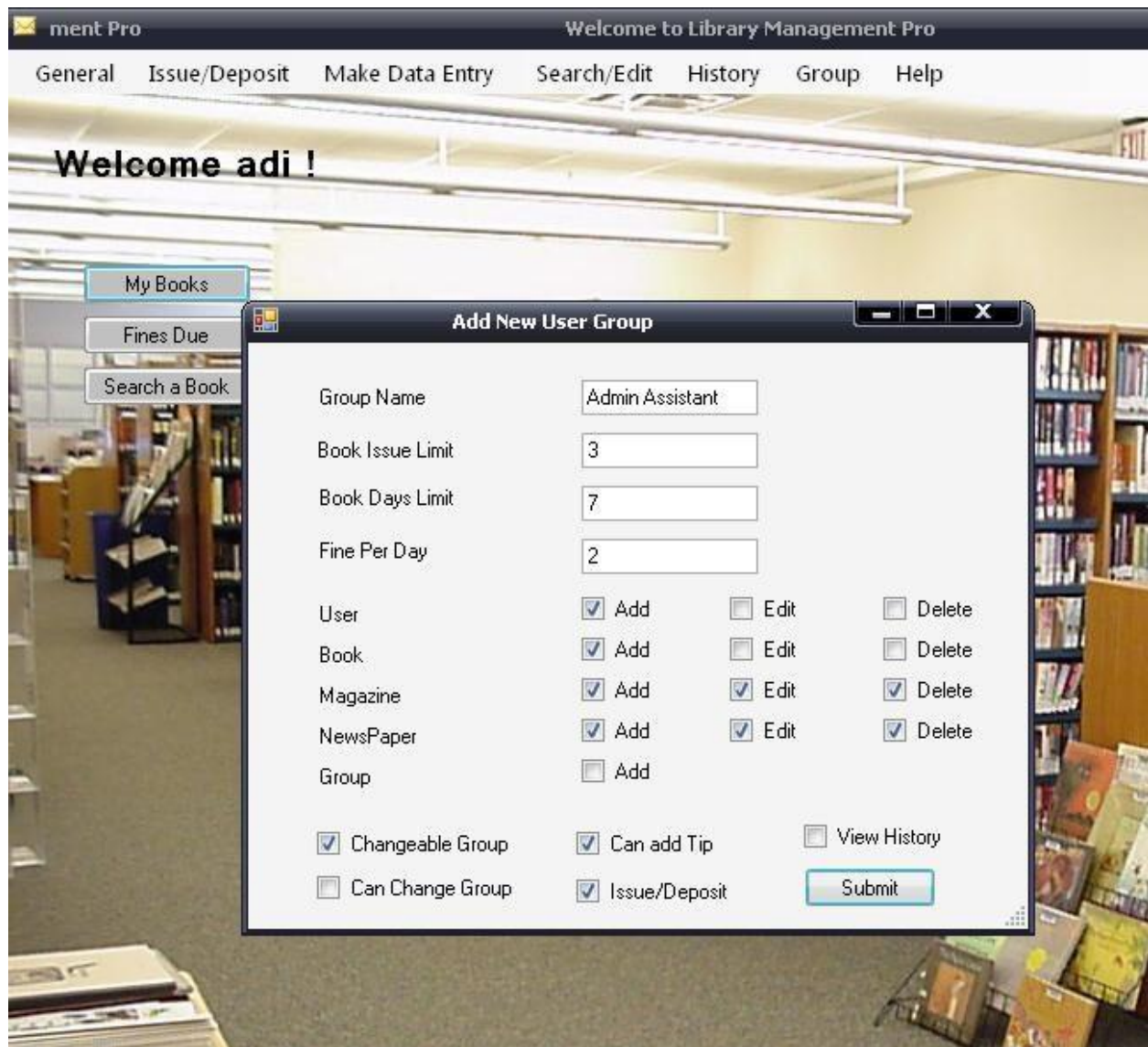


The screenshot shows a window titled "Edit User" with a standard Windows-style title bar (minimize, maximize, close buttons). The window contains a form with the following fields and values:

Field	Value
New Password	atul_87
Confirm Password	atul_87
First Name	atul
Last Name	vani
Sex	<input checked="" type="radio"/> Male <input type="radio"/> Female
Address	12, takshashila colony
Contact number	07312321246
E-mail id	atul@gmail.com
Date of Birth	Wednesday, December 2 ▼
Expiration Date	Sunday, December 1 ▼
Group Name	Basic Plan ▼

At the bottom of the form are two buttons: "Update" (highlighted with a blue border) and "DELETE".

***Edit User***



### *Adding New User Group*

## **5.) IMPLEMENTATION**

### **5.1) Language/ technology used for the implementation**

.Net technology is being used in this project

The language used for implementation is Visual Basic.Net 2008

The database used is MS-SQL 2005.

### **5.2) Features of language/ technology used for the project**

#### **5.2.1) Visual Basic .NET**

**Visual Basic .NET (VB.NET)** is an object-oriented computer language that can be viewed as an evolution of Microsoft's Visual Basic (VB) implemented on the Microsoft .NET framework it's an XML Web Services platform which allows us to build rich .NET applications, which allows users to interact with the Internet using wide range of smart devices (tablet devices, pocket PC's, web phones etc), which allows to build and integrate Web Services and which comes with many rich set of tools like Visual Studio to fully develop and build those applications.

##### **5.2.1.1) What is .NET Built On?**

.NET is built on the Windows Server System to take major advantage of the OS and which comes with a host of different servers which allows for building, deploying, managing and maintaining Web-based solutions. The Windows Server System is designed with performance as priority and it provides scalability, reliability, and manageability for the global, Web-enabled enterprise

### **5.2.1.2) .NET Framework**

.NET is a "Software Platform". It is a language-neutral environment for developing rich .NET experiences and building applications that can easily and securely operate within it. When developed applications are deployed, those applications will target .NET and will execute wherever .NET is implemented instead of targeting a particular Hardware/OS combination. The components that make up the .NET platform are collectively called the .NET Framework. The .NET Framework is a managed, type-safe environment for developing and executing applications. The .NET Framework manages all aspects of program execution, like, allocation of memory for the storage of data and instructions, granting and denying permissions to the application, managing execution of the application and reallocation of memory for resources that are not needed.

The .NET Framework is designed for cross-language compatibility. Cross-language compatibility means, an application written in Visual Basic .NET may reference a DLL file written in C# (C-Sharp). A Visual Basic .NET class might be derived from a C# class or vice versa.

The .NET Framework consists of two main components:

- Common Language Runtime (CLR)
- Class Libraries

### **5.2.1.3) Common Language Runtime (CLR)**

The CLR is described as the "execution engine" of .NET. It provides the environment within which the programs run. It's this CLR that manages the execution of programs and provides core services, such as code compilation, memory allocation, thread management, and garbage collection. Through the Common Type System (CTS), it enforces strict type safety, and it ensures that the code is executed in a safe environment by enforcing code access security. The software version of .NET is actually the CLR version.

#### **5.2.1.4) Working of the CLR**

When the .NET program is compiled, the output of the compiler is not an executable file but a file that contains a special type of code called the Microsoft Intermediate Language (MSIL), which is a low-level set of instructions understood by the common language run time. This MSIL defines a set of portable instructions that are independent of any specific CPU. It's the job of the CLR to translate this Intermediate code into a executable code when the program is executed making the program to run in any environment for which the CLR is implemented. And that's how the .NET Framework achieves Portability. This MSIL is turned into executable code using a **JIT (Just In Time)** compiler. The process goes like this, when .NET programs are executed, the CLR activates the JIT compiler. The JIT compiler converts MSIL into native code on a demand basis as each part of the program is needed. Thus the program executes as a native code even though it is compiled into MSIL making the program to run as fast as it would if it is compiled to native code but achieves the portability benefits of MSIL.

#### **5.2.1.5) Class Libraries**

Class library is the second major entity of the .NET Framework which is designed to integrate with the common language runtime. This library gives the program access to runtime environment. The class library consists of lots of prewritten code that all the applications created in VB .NET and Visual Studio .NET will use. The code for all the elements like forms, controls and the rest in VB .NET applications actually comes from the class library.

### 5.2.2) MS- SQL

**Microsoft SQL Server** is a relational model database server produced by Microsoft. It includes native support for managing XML data, in addition to relational data. For this purpose, it defined an xml data type that could be used either as a data type in database columns or as literals in queries. It includes following features:

- **Fast Recovery:**

A new faster recovery option improves availability of SQL Server databases. Administrators can reconnect to a recovering database after the transaction log has been rolled forward.

- **SQL Server Management Studio:**

SQL Server 2005 includes SQL Server Management Studio, a new integrated suite of management tools with the functionality to develop, deploy, and troubleshoot SQL Server databases, as well as enhancements to previous functionality.

- **Visual Studio Integration:**

Tight integration with Microsoft Visual Studio and the .NET Framework streamlines development and debugging of data-driven applications. Developers can build database objects, such as stored procedures, using any .NET language and can seamlessly debug across .NET and Transact-SQL (TSQL) languages.

- **Data Mining:**

Microsoft SQL Server 2005 Analysis Services (SSAS) provides tools for data mining with which you can identify rules and patterns in your data, so that you can determine why things happen and predict what will happen in the future – giving you powerful insight that will help your company make better business decisions.



## 6. TESTING

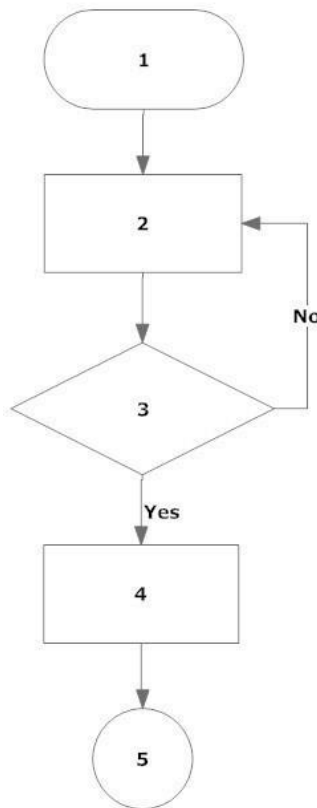
Our project has been tested using Black box and White box testing on all interfaces and loops.

### 6.1) White Box Testing

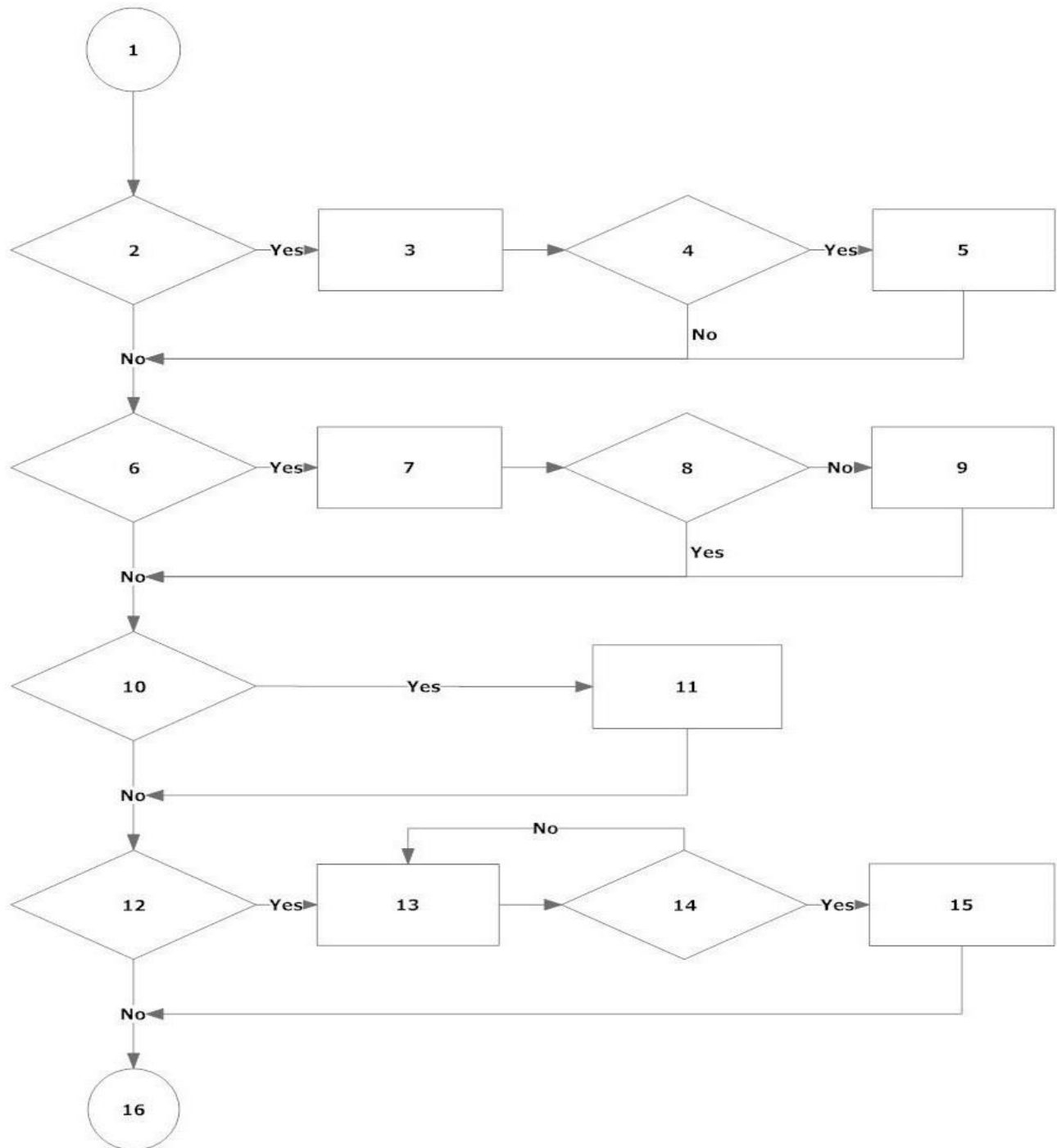
White – box testing, sometimes called glass-box testing, is a test case design philosophy that uses the control structure described as part of component-level design to derive test cases.

Using White box testing methods it can be ensured that:

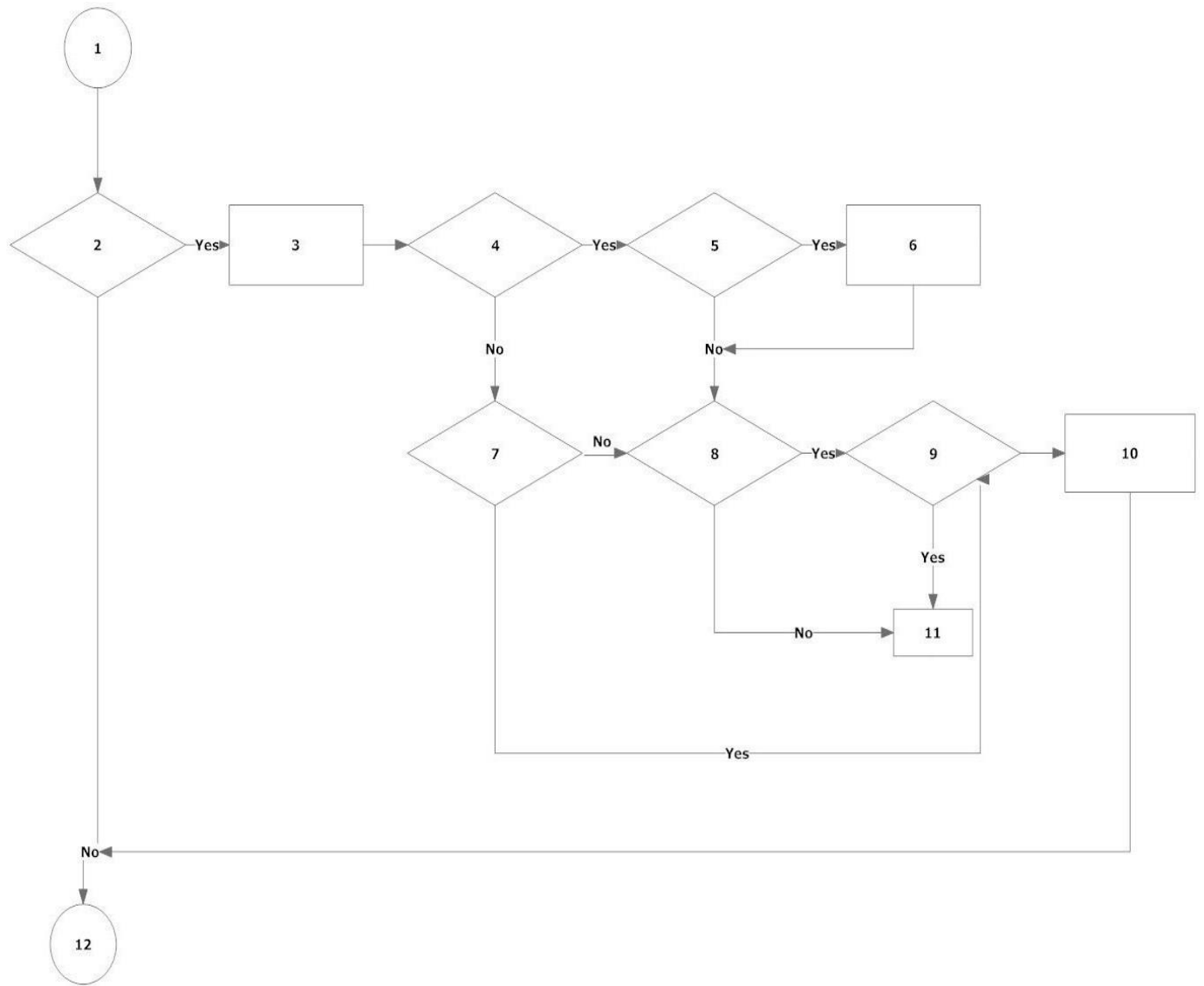
- All independent paths within a module have been exercised atleast once
- Exercise all logical decisions.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structures to ensure their validity.



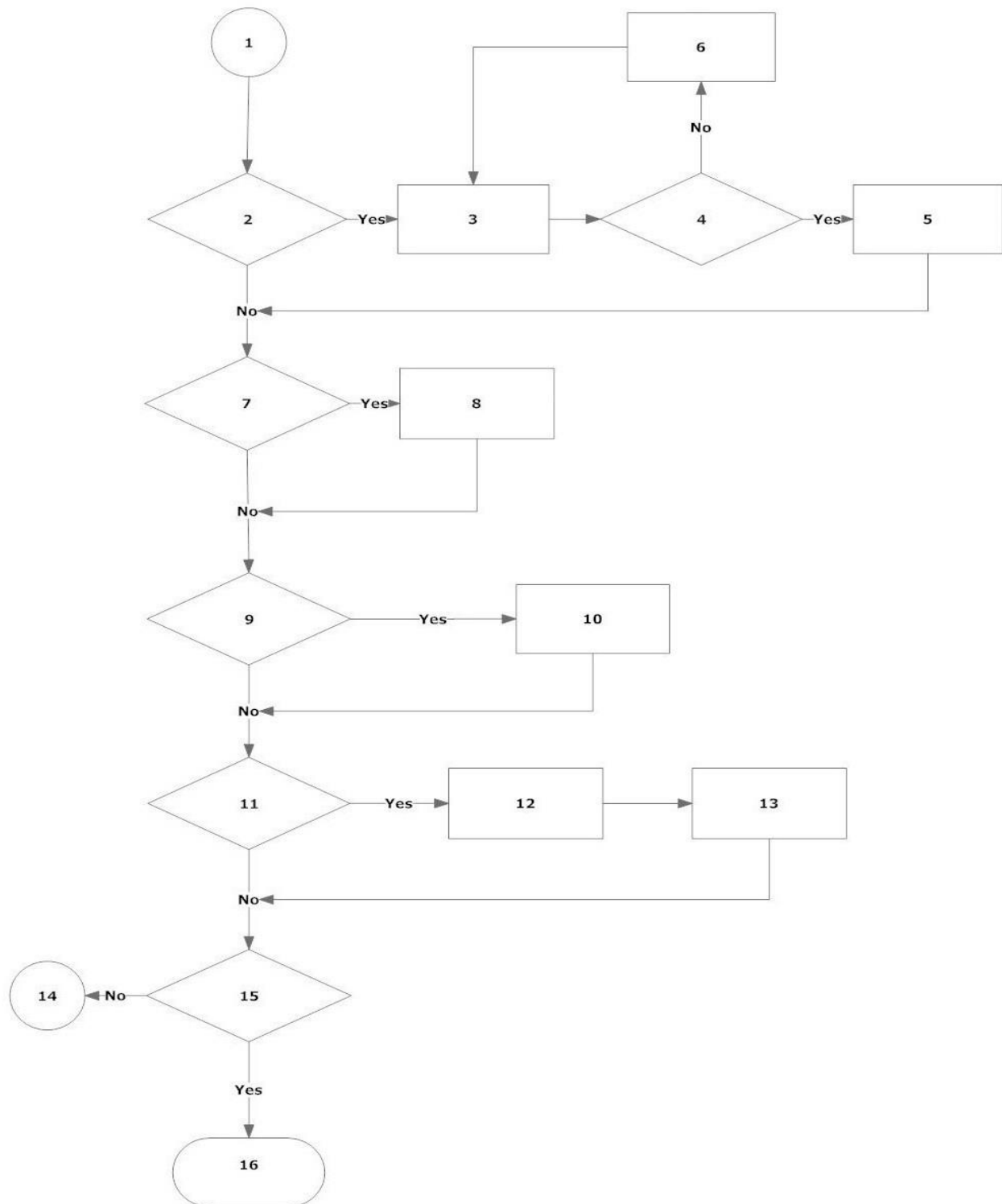
*Fig 15(a): Control Flow Structure*



*Fig 15(b)*



**Fig 15(c)**



*Fig 15(d)*

## 6.2) Cyclomatic complexity

We have calculated cyclomatic complexity for our project. Cyclomatic complexity is a software metric that provides a quantitative measure of the logical complexity of a program. The value computed for Cyclomatic complexity defines the number of independent paths in the basis set of a program and provides us with an upper bound for the number of tests that must be conducted to ensure that all statements have executed at least once. Cyclomatic complexity has a foundation in graph theory and it can be computed as the number of regions corresponds to the cyclomatic complexity.

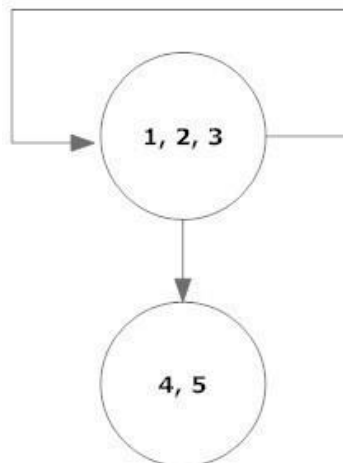
Cyclomatic complexity can be computed in the following ways:

- By counting the number of regions.
- Cyclomatic Complexity  $V(G)$  ,is given by :  $V(G) = E - N + 2$  ,  
Where  $E$  = number of edges,  $N$  = number of nodes.

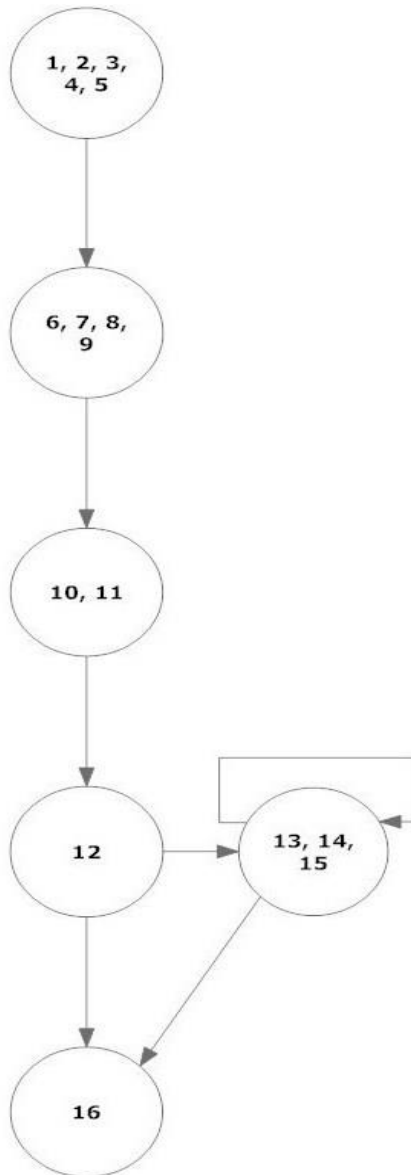
Calculating Cyclomatic Complexity by using the following methods, we get:

- The flow graph has 7 regions.
- $V(G) = 23 \text{ edges} - 18 \text{ nodes} + 2 = 7$ .

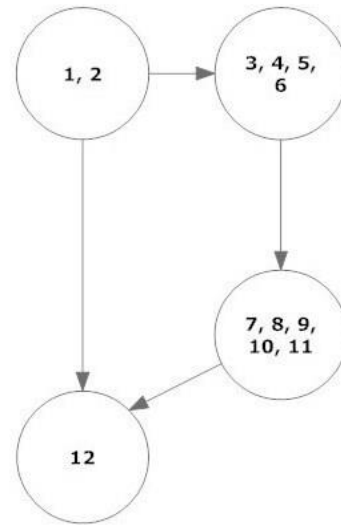
Since our flow graph has 6 regions so our total cyclomatic complexity is 7.



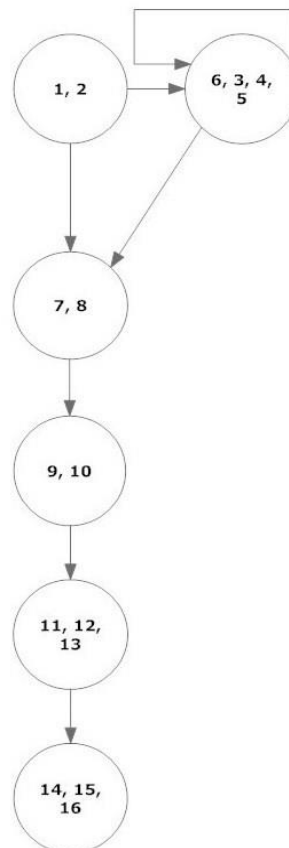
***Fig 16(a)- Flow Graph***



**Fig 16(b)**



**Fig 16(c)**

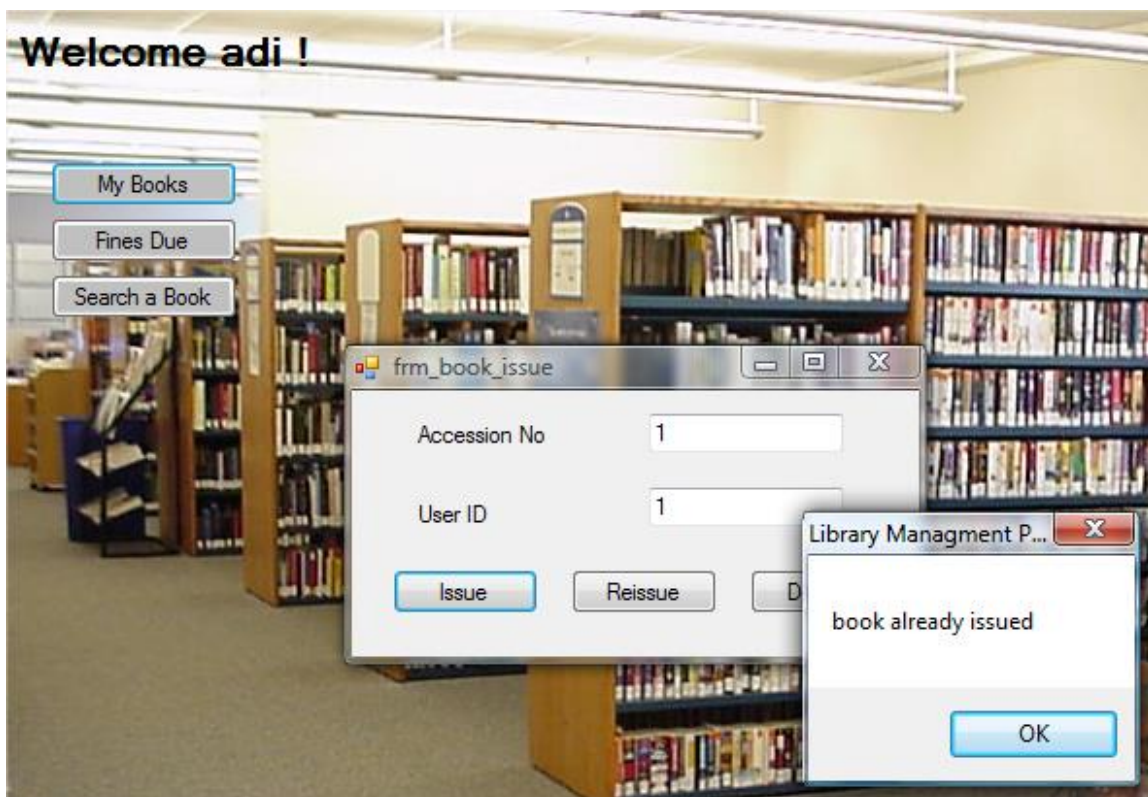
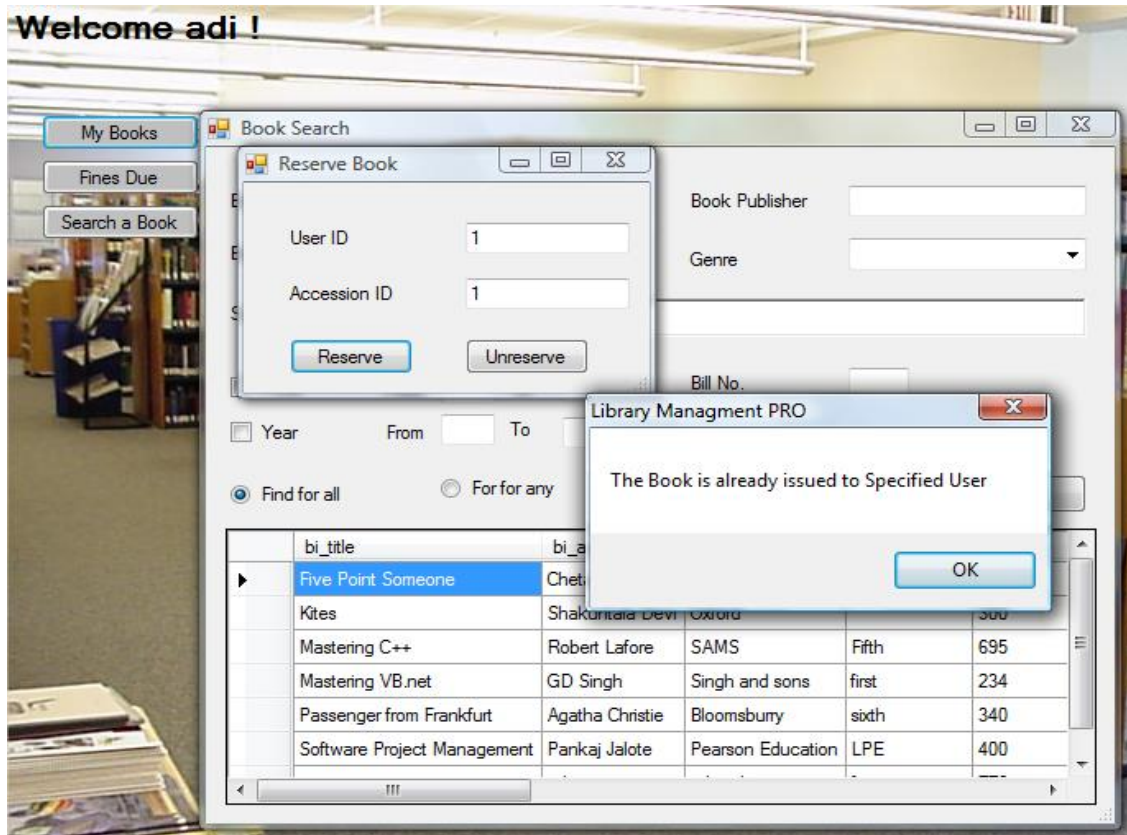


**Fig 16(d)**

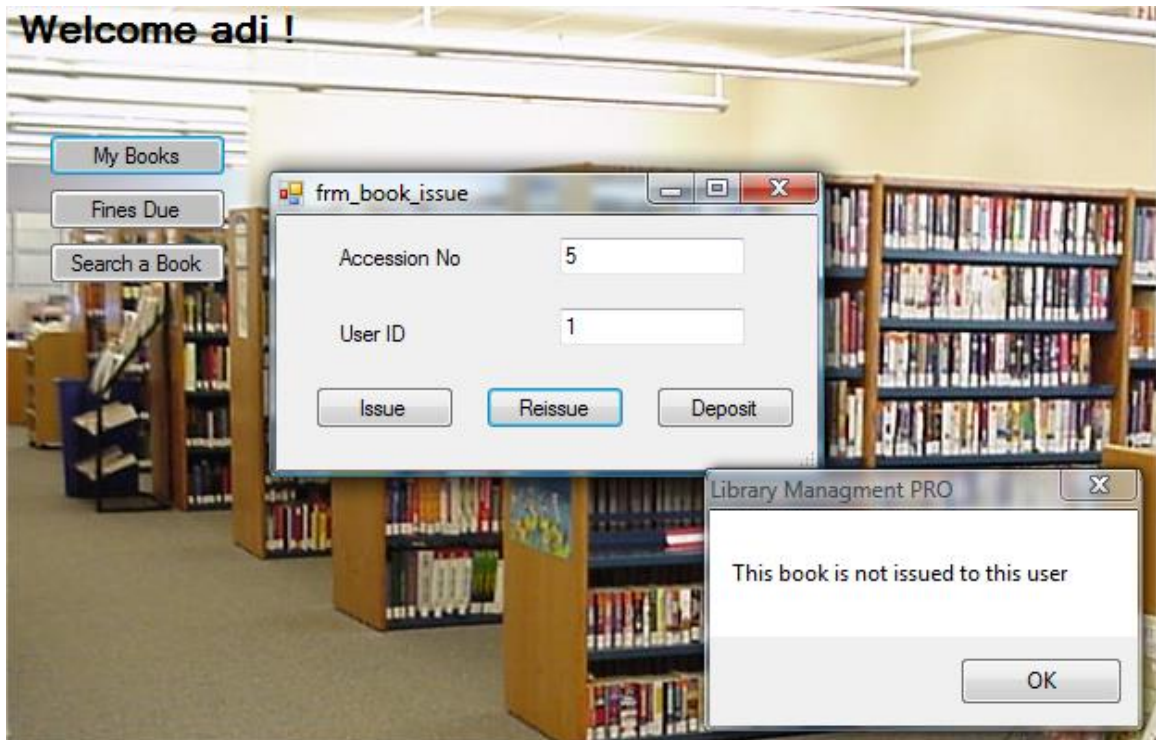
### 6.3) Black Box Testing

Black Box Testing is testing without knowledge of the internal workings of the item being tested. For example, when black box testing is applied to software engineering, the tester would only know the "legal" inputs and what the expected outputs should be, but not how the program actually arrives at those outputs. It is because of this that black box testing can be considered testing with respect to the specifications, no other knowledge of the program is necessary.









## **7. FUTURE SCOPE AND LIMITATIONS.**

Library Management Pro is in itself a complete system, though it has a few limitations but it has a lot of future scope and features that could be added to make it more widely acceptable.

One limitation is that our software is limited to small and medium scaled libraries. Also apart from Books, Magazine and Newspaper no new category can be added in the system (or in turn be issued) like CDs etc.

One of the major future scope is making our system online. Connecting libraries to a common data centre will provide globalization to the libraries, and then the user will be able to search books all over the city and nearby areas.

Reviews, rating, comparing of books and libraries can also be incorporated. This would help the user to browse through popular books and make his selection based on the books rating. Also data obtained from this can be used to discover topics, genre and books that the readers are interested in reading!

Usage of advanced [BOT]s for retrieval of information about new titles available throughout the world.

## **8.) CONCLUSION**

Library Management Pro has been created keeping in mind the needs of Small and Medium scale libraries. Its an efficient software that includes all the basic functionalities like making data entries for new books, newspapers and magazines, registering a new user, editing and deleting records that are required for smooth functioning of a library. Additionally the user login and book history are also stored and can be accessed by the administrator.

It also facilitates the librarian to create new user groups and edit their access levels and functions (like that of the assistants). Apart from this the general users are also given the rights to not only keep track of the books that they have issued and fines due but they too can search for the books/magazines/newspapers that interest them.

## 9. REFERENCE

- Acharya, Kamal. "STUDENT INFORMATION MANAGEMENT SYSTEM." *Authorea Preprints* (2023).
- Acharya, Kamal. "Library Management System." Available at SSRN4807104 (2019).
- ACHARYA, KAMAL, et al. "LIBRARY MANAGEMENT SYSTEM." (2019).
- Acharya, Kamal. "Online bus reservation system project report." *Authorea Preprints* (2024).
- Acharya, Kamal. "Online bus reservation system project report." (2024).
- Acharya, Kamal. "Online Bus Reservation System." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal. "Student Information Management System Project." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal. "ATTENDANCE MANAGEMENT SYSTEM." *International Research Journal of Modernization in Engineering Technology and Science* (2023): n. pag.
- Acharya, Kamal. "College Information Management System." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal, *Attendance Management System Project* (April 28, 2024). Available at SSRN: <https://ssrn.com/abstract=4810251> or <http://dx.doi.org/10.2139/ssrn.4810251>
- Acharya, Kamal, *Online Food Order System* (May 2, 2024). Available at SSRN: <https://ssrn.com/abstract=4814732> or <http://dx.doi.org/10.2139/ssrn.4814732>
- Acharya, Kamal, *University management system project*. (May 1, 2024). Available at SSRN: <https://ssrn.com/abstract=4814103> or <http://dx.doi.org/10.2139/ssrn.4814103>
- Acharya, Kamal, *Online banking management system*. (May 1, 2024). Available at SSRN: <https://ssrn.com/abstract=4813597> or <http://dx.doi.org/10.2139/ssrn.4813597>
- Acharya, Kamal, *Online Job Portal Management System* (May 5, 2024). Available at SSRN: <https://ssrn.com/abstract=4817534> or <http://dx.doi.org/10.2139/ssrn.4817534>
- Acharya, Kamal, *Employee leave management system*. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819626> or <http://dx.doi.org/10.2139/ssrn.4819626>

Acharya, Kamal, *Online electricity billing project report*. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819630> or <http://dx.doi.org/10.2139/ssrn.4819630>

Acharya, Kamal, *POLICY MANAGEMENT SYSTEM PROJECT REPORT*. (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4831694> or <http://dx.doi.org/10.2139/ssrn.4831694>

Acharya, Kamal, *Online job placement system project report*. (January 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4831638> or <http://dx.doi.org/10.2139/ssrn.4831638>

Acharya, Kamal, *Software testing for project report*. (May 16, 2023). Available at SSRN: <https://ssrn.com/abstract=4831028> or <http://dx.doi.org/10.2139/ssrn.4831028>

Acharya, Kamal, *ONLINE CRIME REPORTING SYSTEM PROJECT*. (August 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4831015> or <http://dx.doi.org/10.2139/ssrn.4831015>

Acharya, Kamal, *Burber ordering system project report*. (October 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4832704> or <http://dx.doi.org/10.2139/ssrn.4832704>

Acharya, Kamal, *Teachers Record Management System Project Report* (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4833821> or <http://dx.doi.org/10.2139/ssrn.4833821>

Acharya, Kamal, *Dairy Management System Project Report* (December 20, 2020). Available at SSRN: <https://ssrn.com/abstract=4835231> or <http://dx.doi.org/10.2139/ssrn.4835231>

Acharya, Kamal, *Electrical Shop Management System Project* (December 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835238> or <http://dx.doi.org/10.2139/ssrn.4835238>

Acharya, Kamal, *Online book store management system project report*. (February 10, 2020). Available at SSRN: <https://ssrn.com/abstract=4835277> or <http://dx.doi.org/10.2139/ssrn.4835277>

Acharya, Kamal, *Paint shop management system project report*. (January 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835441> or <http://dx.doi.org/10.2139/ssrn.4835441>

Acharya, Kamal, *Supermarket billing system project report*. (August 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4835474> or <http://dx.doi.org/10.2139/ssrn.4835474>

Acharya, Kamal, *Online taxi booking system project report*. (March 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4837729> or <http://dx.doi.org/10.2139/ssrn.4837729>

*Acharya, Kamal, Online car servicing system project report. (March 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4837832> or <http://dx.doi.org/10.2139/ssrn.4837832>*

*Acharya, Kamal, School management system project report. (July 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4837837> or <http://dx.doi.org/10.2139/ssrn.4837837>*

*Acharya, Kamal, Furniture Showroom Management System Project Report (March 21, 2021). Available*

*at SSRN: <https://ssrn.com/abstract=4839422> or <http://dx.doi.org/10.2139/ssrn.4839422>*

*Acharya, Kamal, Online Vehicle Rental System Project Report (March 21, 2019). Available at SSRN: <https://ssrn.com/abstract=4839429> or <http://dx.doi.org/10.2139/ssrn.4839429>*