

CHAPTER 1

INTRODUCTION

1.1 Introduction to DBMS

A Database Management system, or DBMS is software designed to assist in maintaining and utilizing large collection of data, and the need of such systems, as well as their use, is growing rapidly. The alternative to using a DBMS is to use ad hoc approaches that do not carry over from one application to another.

A Database is a collection of related data organized in a way that data can be easily accessed, managed and updated. Any piece of information can be a data. Database is actually a place where related piece of information is stored and various operations can be performed on it.

A DBMS is a software that allows creation, definition and manipulation of database. DBMS is actually a tool used to perform any kind of operation on data in database. It maintains data consistency in case of multiple users.

From the earliest days of computers, storing and manipulating data have been a major application focus. The first general-purpose DBMS was designed by Charles Bachman at General Electric in the early 1960s and was called the Integrated Data Store. It formed the basis for the network data model, which was standardized by the Conference on Data Systems Languages (CODASYL) and strongly influenced database systems through the 1960s. Bachman was the first recipient of ACM's Turing Award (the computer science equivalent of a Nobel Prize) for work in the database area; he received the award in 1973.

In the late 1960s, IBM developed the Information Management System (IMS) DBMS, used even today in many major installations. IMS formed the basis for an alternative data representation framework called the hierarchical data model. The SABRE system for making airline reservations was jointly developed by American Airlines and IBM around the same time, and it allowed several people to access the same data through. An interesting phenomenon is the emergence of several enterprise resource planning (ERP) and management resource planning (MRP) packages, which add a substantial layer of application-oriented features on top of a DBMS. Widely used packages include systems from Baan, Oracle, PeopleSoft, SAP, and Siebel.

1.2 Overview of Book Store Management

In the existing system, most of the records are maintained on paper. It becomes very inconvenient to modify the data. In the existing system, here is a possibility that the same data in different registers may have different values which means the entries of the same data do not match. This inconsistent state does not supply the concrete information which poses a problem in the case information related to particular search record. Our project is very useful. User is no longer required to check his register in search of records, as now it can be searched over the software by choosing some options. The user need not to type in most of the information. He/she is just required to enter the desired options. On the whole it liberates the user from keeping lengthy manual records. In a nutshell, it abates the work load of an organization.

In today's world, no one likes to perform calculations on calculator or manually when computer is there. Everyone wants his/her work to be done by computer automatically and displaying the result for further manipulations. A book shop uses a personal computer to maintain the inventory of books that are being sold at the shop. The list includes details such as author, book title, price, quantity, stock position, etc. Whenever a customer wants a book, the shopkeeper inputs the title and author of the book and the system replies whether it is in the list or not. If it is not, an appropriate message displayed. If book is in the list, then the system displays the book details and asks for number of copies. If the requested copies are available the, total cost of the books is displayed; otherwise, the message "Required copies not in stock" is displayed.

CHAPTER 2

SYSTEM REQUIREMENTS SEPECIFICATIONS

The System Requirements Specification (SRS) is a document focused on what the software needs to do and how it must perform. It lays the important groundwork so that every person involved with the project understands the most crucial details.

2.1 Functional Requirements

- **Inventory management:**

The ability to add, edit, and delete book records, track stock levels, and generate reports on inventory.

- **Sales management:**

The ability to process transactions, track sales, and generate reports on sales data.

- **Customer management:**

The ability to add, edit, and delete customer records, and track customer purchase history.

- **Financial management:**

The ability to track expenses and income, generate financial reports, and integrate with accounting software.

- **Reporting and analytics:**

The ability to generate various reports on sales, inventory, customers, employees, and financials.

- **Integrations:**

The ability to integrate with other systems such as accounting software, e-commerce platforms, and social media.

- **Security:**

The ability to secure the system and data, with features such as user authentication, role-based access, and encryption.

2.2 Non-Functional Requirements

- **Usability:**

The system should be easy to use and navigate for employees and customers.

- **Reliability:**

The system should be reliable and able to handle a large volume of transactions without errors.

- **Scalability:**

The system should be able to handle an increasing number of customers and transactions as the business grows.

- **Security:**

The system should protect sensitive customer and financial data from unauthorized access.

- **Performance:**

The system should be able to perform transactions and generate reports quickly.

- **Maintainability:**

The system should be easy to maintain and update as needed.

- **Compatibility:**

The system should be compatible with other software and systems used by the book store.

- **Accessibility:**

The system should be accessible from multiple locations and devices.

- **Compliance:**

The system should adhere to any relevant laws and regulations related to data privacy and financial transactions.

- **Backup and Recovery:**

The system should be able to backup and recovery data, transactions and reports in case of any failure.

2.3 Hardware and Software Requirement

2.3.1 Hardware Requirement

- Processor - Intel core I3
- Processor speed - 2.1Ghz
- Ram - 4 GB
- Hard Disk - 1 TB

2.3.2 Software Requirement

- Operating System - Windows 10
- Back End - Oracle 10g Express Edition
- Front End - Java Swings using NetBeans IDE 8.2

CHAPTER 3

SYSTEM DESIGN

It is an automated Book Store Management System. Through our software user can add members, add books, search members, search books, update information, edit information, borrow book in quick time. Our proposed system has the following advantages.

- It provides “better and efficient” services 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.
- It has more storage capacity and search facility and fast access to database. All the manual difficulties in managing the Book store have been rectified by implementing computerization.

3.1 ER DIAGRAM

An entity–relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

ER diagram should have mainly 3 components namely, entity, attribute, relationship. The following notations can be used for drawing an ER diagram.

- **Entity**

An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.

- **Attribute**

The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

- **Relationship**

A relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.

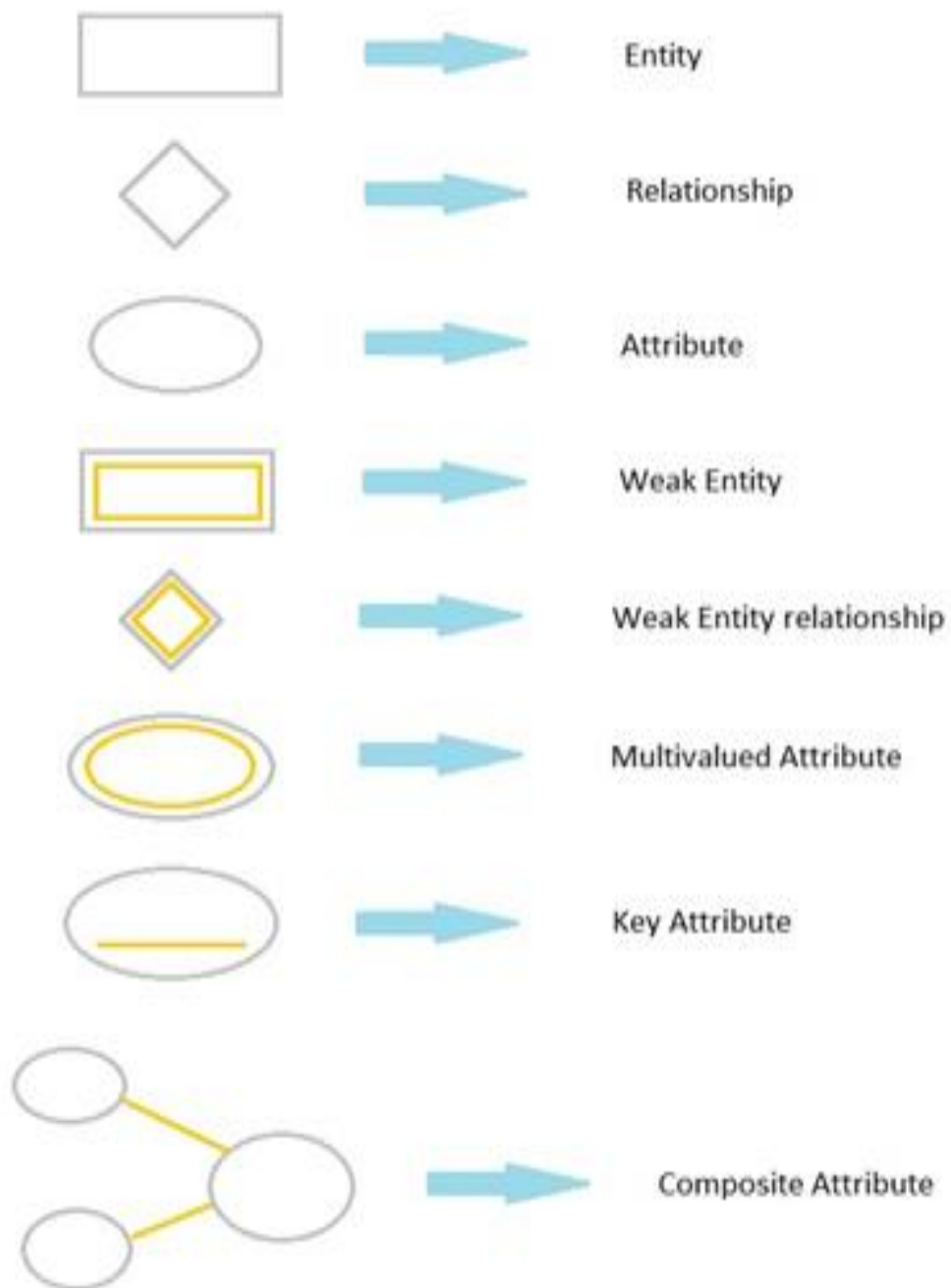


Figure 3.1: E-R Diagram Notations.

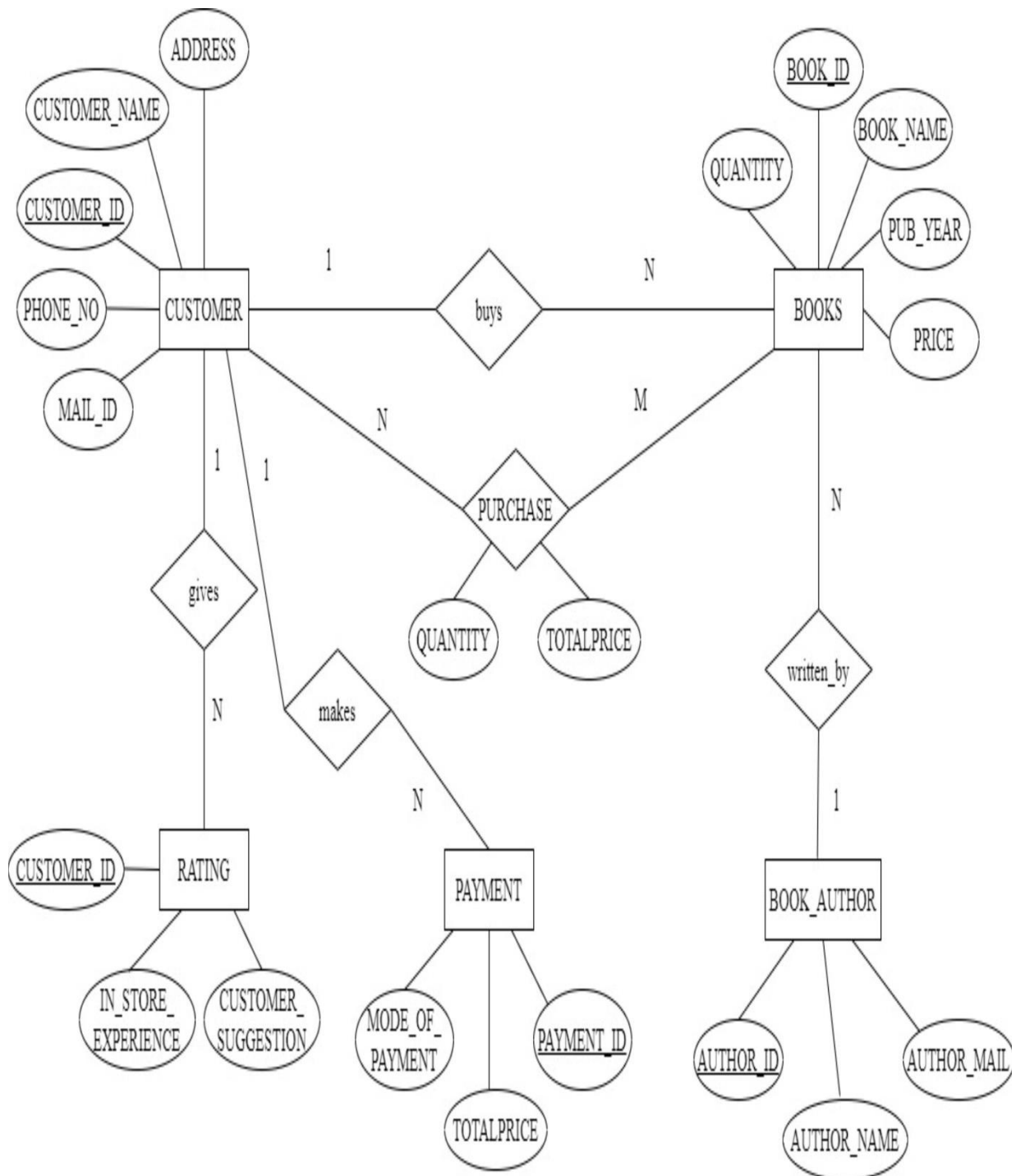


Figure 3.2: Shows the E-R diagram of Book Store Management.

3.2 Relational Schema

The relational schema diagram gives the relation of one entity with another as well as the information about the key constraints. The below figure is a sample relational schema diagram in which the attributes that are underlined are the primary key and the arrow line is used to represent the mapping.

In figure 3.3 shows the schema diagram for book store management system. As mentioned previously the book store management system consists of 6 entities. The name that are placed inside the rectangular boxes specify the attributes. The line with arrows is indicating the mapping between the relations.

BOOK_AUTHOR

<u>AUTHOR_ID</u>	AUTHOR_NAME	AUTHOR_MAIL
------------------	-------------	-------------

BOOKS

<u>BOOK_ID</u>	BOOK_NAME	AUTHOR_ID	PUB_YEAR	QUANTITY	PRICE
----------------	-----------	-----------	----------	----------	-------

CUSTOMER

<u>CUSTOMER_ID</u>	CUSTOMER_NAME	PHONE_NO	ADDRESS	MAIL_ID
--------------------	---------------	----------	---------	---------

PURCHASE

CUSTOMER_ID	BOOK_ID	QUANTITY	TOTALPRICE
-------------	---------	----------	------------

PAYMENT

<u>PAYMENT_ID</u>	CUSTOMER_ID	MODE_OF_PAYMENT	TOTALPRICE
-------------------	-------------	-----------------	------------

RATING

<u>CUSTOMER_ID</u>	IN_STORE_EXPERIENCE	CUSTOMER_SUGGESTION
--------------------	---------------------	---------------------

Figure 3.3: Shows the Schema diagram of Book Store Management

3.3 Attributes list

BOOK_AUTHOR: To store Book Author Information. Here Primary key is Author_id. Attributes of the table are Author_id, Author_name, Author_mail.

BOOKS: All Books in stock are available in this list. Primary key is Book_id. Attributes of table are Book_id, Book_name, Author_id, Pub_year, Quantity, Price.

CUSTOMER: Customer details will be stored in customer list. This will be used by sales list to sell items to customer in this list. Here primary key is Customer_id. Attribute of table are Customer_id, Customer_name, Phone_no, Address, Mail_id.

PURCHASE: All the purchase details of customer will be stored here. Attribute of the table are Customer_id, Book_id, Quantity, TotalPrice.

PAYMENT: In this table one can store Payment done by Customer sales. Primary key is Payment_id. Attributes are Payment_id, Customer_id, Mode_of_Payment, TotalPrice.

RATINGS: In this table one can store Ratings given by Customer. Attributes are Customer_id, In_Store_Experience, Customer_Suggestion.

CHAPTER 4

IMPLEMENTATION

4.1 TOOLS AND TECHNOLOGIES

4.1.1 Oracle 10g Express Edition

Oracle Application Express is installed in Oracle Database 10g Express Edition as the primary tool for managing the database and building Web-based applications. Oracle Application Express is a rapid web application development tool for the Oracle database. Using only a web browser and limited programming experience, you can develop professional applications that are both fast and secure. The deployed applications require only a browser and access to an Oracle database running Application Express.

4.1.2 Java Swings

Java Swing is a lightweight Java graphical user interface (GUI) widget toolkit that includes a rich set of widgets. It is part of the Java Foundation Classes (JFC) and includes several packages for developing rich desktop applications in Java. Swing includes built-in controls such as trees, image buttons, tabbed panes, sliders, toolbars, colour choosers, tables, and text areas to display HTTP or rich text format (RTF). Swing components are written entirely in Java and thus are platform-independent.

4.1.3 NetBeans IDE 8.2

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like Java Swings, PHP, C, C++, HTML5, and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

4.1.4 JDBC Jar

A JDBC driver (Java Database Connectivity driver) is a small piece of software that allows JDBC to connect to different databases.

Essentially, a JDBC driver makes it possible to do three things

- Establish a connection with a data source.
- Send queries and update statements to the data source.
- Process the results.

JDBC is an application program interface (API) specification for connecting programs written in Java to the data in popular databases. Each JDBC driver allows communication with a different database.

CHAPTER 5

RESULTS AND DISCUSSION

LOGIN:

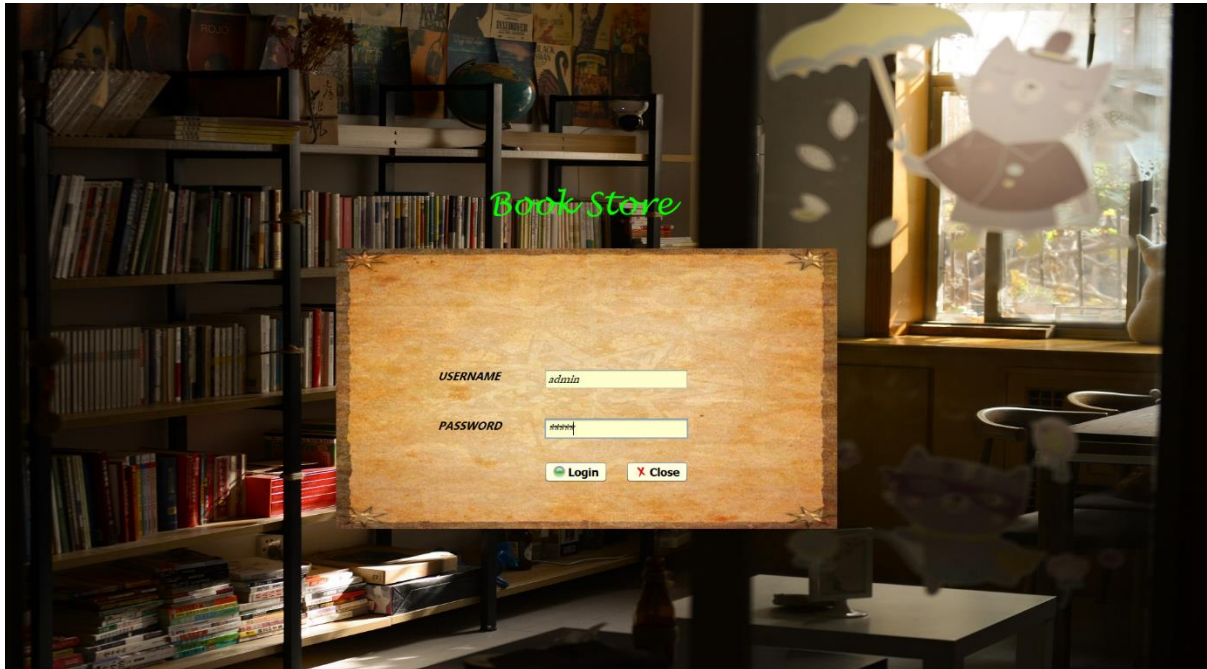


Figure 4.1: Shows the Snapshot of Login Page

HOME:

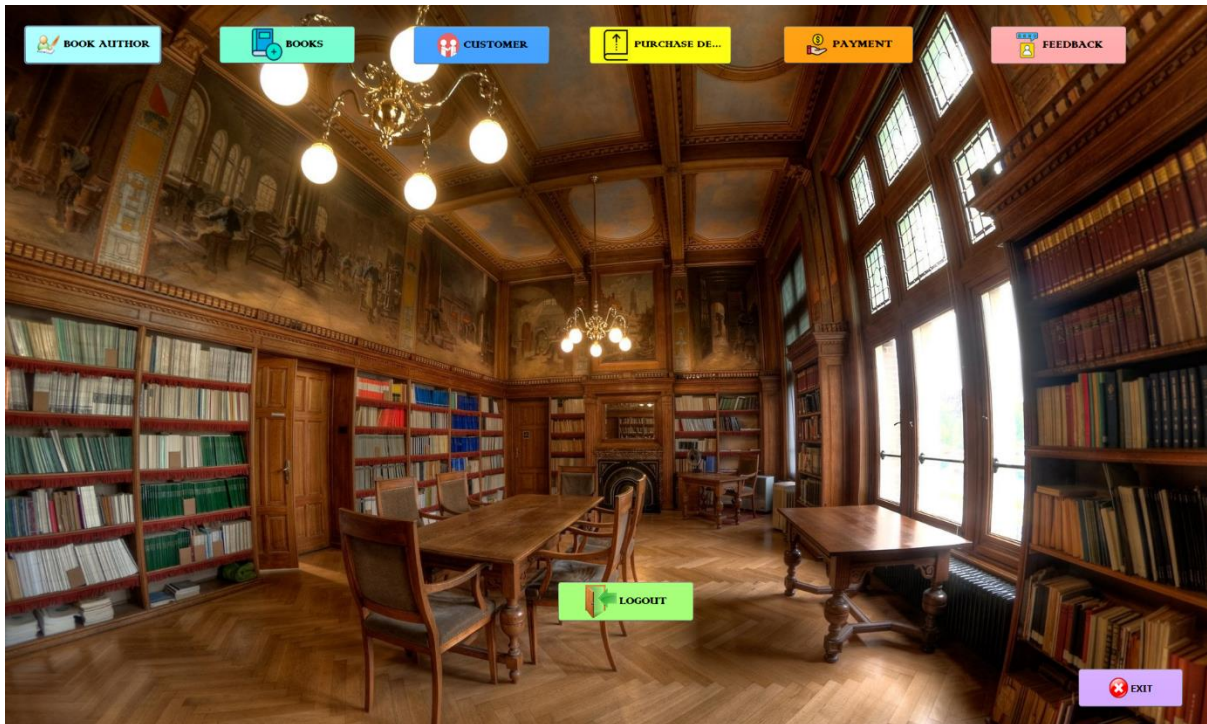


Figure 4.2: Shows the Snapshot of Home Page

BOOK_AUTHOR:

BOOK AUTHOR

BOOKS

CUSTOMER

PURCHASE DE...

PAYMENT

FEEDBACK

AUTHOR ID

AUTHOR NAME

AUTHOR MAIL

ADD

VIEW

RESET

DELETE

EXIT

LOGOUT

EXIT

Figure 4.3: Shows the Snapshot of Book Author Page**BOOKS:**

BOOK AUTHOR

BOOKS

CUSTOMER

PURCHASE DE...

PAYMENT

FEEDBACK

BOOKS

AUTHORS

Book ID

Price

Pub Year

Book Name

Author ID

Quantity

BOOK_ID

PRICE

PUB_YEAR

BOOK_NAME

AUTHOR_ID

QUANTITY

AUTHOR_ID

AUTHOR_NAME

ADD

VIEW

RESET

DELETE

EXIT

UPDATE BOOK QUANTITY

LOGOUT

EXIT

Figure 4.4: Shows the Snapshot of Books Page

CUSTOMER:

CUSTOMER

CID

CNAME

PHONE NO

ADDRESS

MAIL ID

CUSTOMER_ID	CUSTOMER_NAME	PHONE_NO	ADDRESS	MAIL_ID
-------------	---------------	----------	---------	---------

Figure 4.5: Shows the Snapshot of Customer Page**PURCHASE:**

PURCHASE

CUSTOMER ID

BOOK ID

QUANTITY

TOTAL PRICE

CUSTOMER_ID	BOOK_ID	QUANTITY	TOTAL PRICE
-------------	---------	----------	-------------

CUSTOMER

CUSTOMER_ID	CUSTOMER_NAME
-------------	---------------

Figure 4.6: Shows the Snapshot of Purchase Page

PAYMENT:

PAYMENT

PAYMENT ID	CUSTOMER_ID	PAYMENT_ID	MODE_OF_PAY...	TOTALPRICE
------------	-------------	------------	----------------	------------

PAYMENTMODE: CASH

TOTAL PRICE

CUSTOMER ID

FETCH PRICE PAY VIEW RESET DELETE EXIT

LOGOUT

EXIT

Figure 4.7: Shows the Snapshot of Payment Page**RATING:**

FEEDBACK

CUSTOMER ID	IN_STORE_EXPERIENCE	CUSTOMER_SUGGESTION
-------------	---------------------	---------------------

IN STORE EXPERIENCES: EXCELLENT

CUSTOMER SUGGESTION

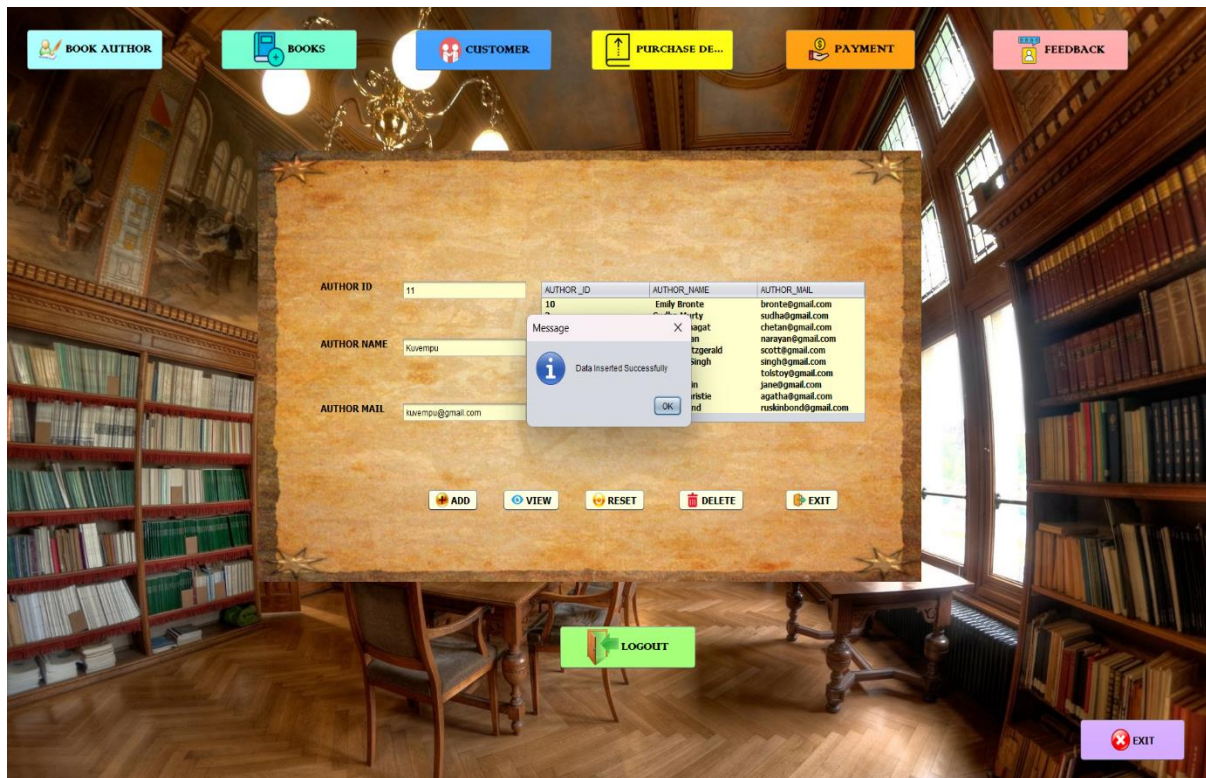
ADD VIEW RESET

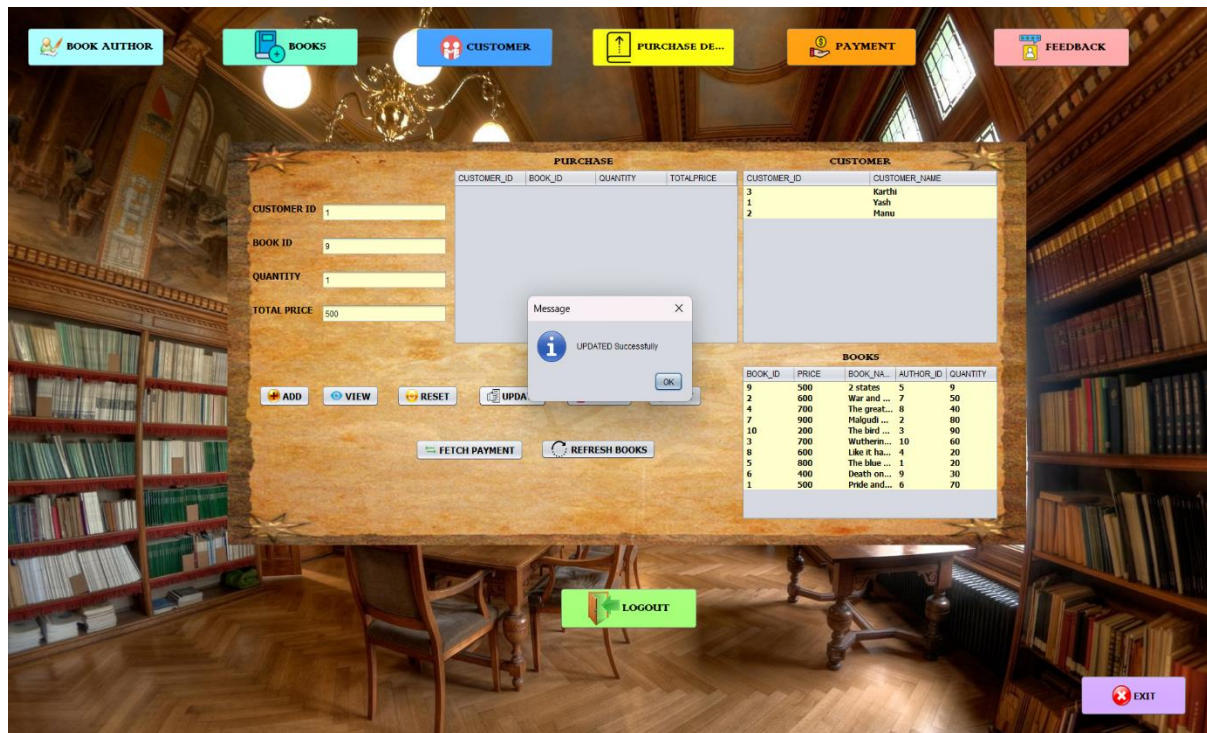
DELETE EXIT

LOGOUT

EXIT

Figure 4.8: Shows the Snapshot of Rating Page

INSERT:**Figure 4.9: Shows the Snapshot of Insertion Operation****DELETE:****Figure 4.10: Shows the Snapshot of Deletion Operation**

UPDATE:**Figure 4.11: Shows the Snapshot of Updation Operation**

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

In conclusion, managing a book store requires a combination of business acumen, strong organizational skills, and a passion for literature. Successful book store managers must be able to effectively market their products, manage their inventory, and provide excellent customer service. Additionally, they must stay up to date with industry trends and be able to adapt to changes in technology and consumer behaviour. By focusing on these key areas and continually seeking ways to improve, book store managers can build a thriving business that serves the community and promotes a love of reading.

6.2 FUTURE WORK

There are some limitations for the current system to which solutions can be provided as a future development:

1. The system is not configured for multi- users at this time. The concept of transaction can be used to achieve this.
2. The Software is not accessible to everyone. It can be deployed on a web server so that everybody who is connected to the Internet can use it.
3. Credit/Debit Card validation is not done. Third party proprietary device can be used for validation check.

REFERENCES

1. Ramez Elmasri and Shamkant B Navathe, “Database Systems Models, Languages, Design and Application Programming”, Pearson, 7th Edition, 2017.
2. Raghu Ramakrishnan and Gehrke, “Database management systems”, McGraw Hill, 3rd Edition, 2014.
3. Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
4. Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.