

Summer Training Report

On

Java SE

(Subject Code: BCA – 508P)

*Submitted in partial fulfilment of the
requirement for the award of the degree*

of

BACHELOR OF COMPUTER APPLICATION

Submitted by

PRAKHAR GUPTA

[202040040036]

Under the guidance of

Dr Santosh Kumar Dwivedi



**DEPARTMENT OF COMPUTER APPLICATION,
SHRI RAMSWAROOP MEMORIAL COLLEGE OF MANAGEMENT, LUCKNOW**

Affiliated to

LUCKNOW UNIVERSITY, LUCKNOW



AUGUST – 2022

CERTIFICATE



Analyze Infotech, Janpriya Complex Phase-II,
Tedhi Pulia, Aliganj, Lucknow, (U.P.)
Phone :-9453193268, 9839434821,
Web :-www.analyzeinfotech.in
E-mail :-admin@analyzeinfotech.in

Ref. No. ai/22/216

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Mr. Prakhar Gupta** S/O Mr. Ashish Gupta student of BCA from Shri Ramswaroop Memorial College of Management, Lucknow in recognition of successful completion of Summer Training on "**Java Software Development**" at Analyze InfoTech. The duration of his project was from 22 June 2022 to 25 August 2022.

He has done a good job during his engagement with the software Development & Testing Division of Company. He has completed his project during the Training tenure. His performance was good and satisfactory.

I would like to take this opportunity to express my appreciation to Mr. **Prakhar Gupta** for his work and wish him All the best for future endeavors.

Yours Sincerely



Date: **25 August 2022**

Pankaj Mishra

(Project Manager)

For Analyze InfoTech, Lucknow.

ABSTRACT

This project **multiplex ticket booking** is a process of supplying a user interface to the user to Book Multiple movies in one software. It helps user's to re about the film that is available in the Movie theatre, and it supplies an uncomplicated way to book a ticket and see the timing of the film and upcoming movies, this software offers a way to compare movies rate from multiple theatres and available seats are there. At the front end, we have used Java NetBeans, and at the back-end MySQL database. The project proceeds through a sequence of a well-designed jForm and new JFrame provided with validations to ensure consistency, reliability and most importantly correctness of information fed into the database.

The whole working will include the following steps:

1. The user/Admin can login into the software
2. User should be booking a ticket
3. Admin can add the theatre or add new movies
4. User can book a ticket through multiple time slots, and they have multiple option to choose which theatre they want to select.
5. The admin can manage the time slots and available movies they can add new movies And remove movies.
6. And they can create its own account and drop the account
7. They can also update its profile.

ACKNOWLEDGEMENT

Firstly, we would like to thank the people who lead us, shared their thoughts, ideas and gave innumerable suggestions during the development of our project. We would also like to take the opportunity to express to our sincere graduate to **Analyze InfoTech** who is our instructor for his immense support during the project works. We are also very thankful and want to show our immense respect to **Dr. Santosh Dwivedi** who is our Head of the department also our project Guide during the difficult situation of project development. He has helped us with resourceful helpful ideas and suggestions throughout our journey of our project. Last but not least we want to thank us institute **Shri Ramswaroop Memorial College of Management** for its valuable help. I am taking this opportunity to acknowledge their support and I wish that they keep supporting me like this in the future.

Prakhar Gupta

[202040040036]

DECLARATION

I undersigned solemnly declare that the project report is based on my own work conducted during our study under the supervision of Dr. Santosh Kumar Dwivedi. I assert the statements made and conclusions drawn are an outcome of my research work. I further certify that:

- I. The work contained in the report is original and has been done by me under the general supervision of my supervisor.
- II. The work has not been given to any other Institution for any other degree/diploma/certificate in this university or any other University of India or abroad.
- III. We have followed the guidelines provided by the university in authoring the report.
- IV. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and giving their details in the references.

Signature of the Student

[Prakhar Gupta]

TABLE OF CONTENTS

TOPIC PAGE:

1. COVER PAGE -----	I
2. CERTIFICATE -----	1
3. ABSTRACT -----	2
4. ACKNOWLEDGEMENT -----	3
5. DECLARATION -----	4
6. CHAPTER 1	
i. INTRODUCTION-----	7
• BACKGROUND-----	7
• OBJECTIVE -----	7-8
• PURPOSE AND SOCPE-----	8
7. CHAPTER 2	
i. SURVAYS OF TECHNOLOGY -----	9
• JAVA PROGRAMMING LANGUAGE-----	9
• MySQL-----	9
8. CHAPTER 3	
i. SOFTWARE ENGINEERING-----	10
• SOFTWARE DEVELOPMENT LIFE CYCLE-----	10
• THE STAGES OF SDLC-----	11-12
• WATERFALL MODEL-----	12-13
9. CHAPTER 4	
i. REQUIREMENT ANALYSIS-----	14
• PROBLEM DEFINITION-----	14
• PLANNING AND SCHEDULING-----	14-15
• REQUIREMENT SPECIFICATION-----	16
• PRELIMINARY PRODUCT-----	16-17
• CONCEPTUAL MODEL-----	17-18
• USECASE DIAGRAM-----	19-20
• ER-DIAGRAM-----	21-22
• DATA FLOW DIAGRAM-----	23-25

10. CHAPTER 5	
i. TESTING -----	26
• SYSTEM TESTING -----	26
• WHY TESTING IS DONE -----	26
• CAUSES OF ERRORS -----	26-27
• TESTING PRINCIPLES -----	27
• TESTING OBJECTIVES -----	27
• KINDS OF TESTING -----	28
• TESTING TECHNIQUE USED -----	29
11. CHAPTER 6	
i. SCREENSHOT OF DATA TABLE -----	30-31
ii. PROJECT SCREENSHOT -----	32-37
iii. USER LOGIN WINDOW -----	37-42
12. CHAPTER 7	
i. IMPLEMENTATIONS -----	43
• SYSTEM IMPLEMENTATION -----	43
• SYSTEM SPECIFICATION -----	43
13. FUTURE SCOPE OF PROJECT -----	44
14. CONCLUSION -----	45
15. REFERENCES -----	46

CHAPTER 1

1 INTRODUCTION

1.1 BACKGROUND:

In the fast-changing world, information technology and information management are going to play a key role. We are living in the computer age during past some year. The computer has gaining popularity. Computer revolution found its way into almost every aspect of human life and living. A computer is admirably suited to manage any information and hence is an information processor that is, it can receive data, perform some basic operations on that data and produces results according to a predetermined program.

This Software is used mainly for the multiplex cinema halls for booking the ticket and storing the information of customers. The multiplex ticket booking software is so designed as to ease the workload of multiplex cinema hall professionals. The main feature includes ticket booking, movie show details management and client management.

1.2 OBJECTIVE:

Today's world is computer world because most of work is doing with the help of computer. Dependency on computer is behind the few reasons. We cannot easily manage to store considerable number of data or information single handle. If we will be needing some information or data in urgency, then we cannot manage in manually these works are exceedingly difficult if we cannot use computer.

As this is generic software it can be used by a wide variety of multiplex cinema halls to automate the process of manually keeping the records related to the subject of keeping the movie details and customer data.

This software is basically updating the manual multiplex ticket booking to automated ticket booking. So that organization can manage their record in efficient and organize them.

- The main aim is to automate non computer environment
- Save workforce.
- It will speed the processing of data and transaction.
- It will supply best security features such as provisions of passwords.

- **System Objective:**

Today's world is computer world because most of work is doing with the help of computer. Dependency on computer is behind the few reasons. We cannot easily manage to store considerable number of data or information single handle. If we will be needing some information or data in urgency, then we cannot manage in manually these works are exceedingly difficult if we cannot use computer.

- **System Context:**

This section clearly depicts the environment and boundaries of the Multiplex ticket booking system and the entities with which it interacts. It helps us see how the system fits into the existing scheme of things. What the system will do by itself.

- **Functional Requirement:**

This Software must request Username and Password for access to data, after authentication will allow access to the system. The Software must allow input of products data from administrator and secured access.

- **Non-Functional Requirement:**

In this Software Input error will be returned in red with right message box. System should automatically update after every transaction.

1.3 PURPOSE AND SCOPE:

- **Purpose:**

The purpose of this document is to specify requirements and to give guidelines for the development of above said project. It gives guidelines on how to prepare the above said project. This document is intended to be a practical guide for people who developing this software.

- **Scope:**

As this is generic software it can be used by a wide variety of multiplex cinema halls to automate the process of manually supporting the records related to the subject of keeping the movie details and customer data.

CHAPTER 2

2 SURVEYS OF TECHNOLOGY:

2.1 JAVA PROGRAMMING LANGUAGE:

Java is general-purpose computer-programming language that is concurrent, class-based, Object-oriented, and specifically designed to have as few implementations' dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java Virtual machine (JVM) regardless of computer architecture. As of 201, Java is one of the Most popular programming languages in use, particularly for client-server sub applications, with report 9 million developers. Developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core part of Sun Microsystem's Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

2.2 MySQL:

MySQL is an open-source relational database management system (RDBMS). Its name is A combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the Abbreviation for Structured Query Language. The My SQL development project has made its source code available under the terms the GNU General Public License, as well as under a variety of proprietary agreements. MySQL Lab, no owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer more functionality.

CHAPTER 3

3. SOFTWARE ENGINEERING:

Software Engineering is an engineering branch related to the evolution of software product using Well-defined scientific principles, techniques, and procedures. The result of software engineering Is an effective and reliable software product.



3.1 SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC):

A software life cycle model (also termed process model) is a pictorial and diagrammatic representation of the software life cycle. A life cycle model stands for all the methods needed to make a software product transit through its life cycle stages. It also captures the structure in which these methods are to be undertaken.

3.2 THE STAGES OF SDLC ARE AS FOLLOWS:

STAGE1: PLANNING AND REQUIREMENT ANALYSIS:

Requirement Analysis is the most important and necessary stage in SDLC. The senior members of the team perform it with inputs from all the stakeholders and domain experts or SMEs in the industry. Planning for the quality assurance requirements and identifications of the risks associated with the projects is also done at this stage. Business analyst and Project organizer set up a meeting with the client to gather all the data like what the customer wants to build, who will be the end user, what is the aim of the product. Before creating a product, a core understanding or knowledge of the product is very necessary.

STAGE2: DEFINING REQUIREMENTS:

Once the requirement analysis is done, the next stage is to certainly stand for and document the software requirements and get them accepted from the project stakeholders. This is conducted through "SRS"- Software Requirement Specification document which holds all the product requirements to be constructed and developed during the project life cycle.

STAGE3: DESIGNING THE SOFTWARE:

The next phase is about to bring down all the knowledge of requirements, analysis, and design of the software project. This phase is the product of the last two, like inputs from the customer and requirement gathering.

STAGE4: DEVELOPING THE PROJECT:

In this phase of SDLC, the actual development begins, and the programming is built. The implementation of design begins concerning writing code. Developers must follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement the code.

STAGE5: TESTING:

After the code is generated, it is assessed against the requirements to make sure that the products are solving the needs addressed and gathered during the requirements stage.

STAGE6: DEPLOYMENT:

Once the software is certified, and no bugs or errors are told, then it is deployed. Then based on the assessment, the software may be released as it is or with suggested enhancement in the object segment.

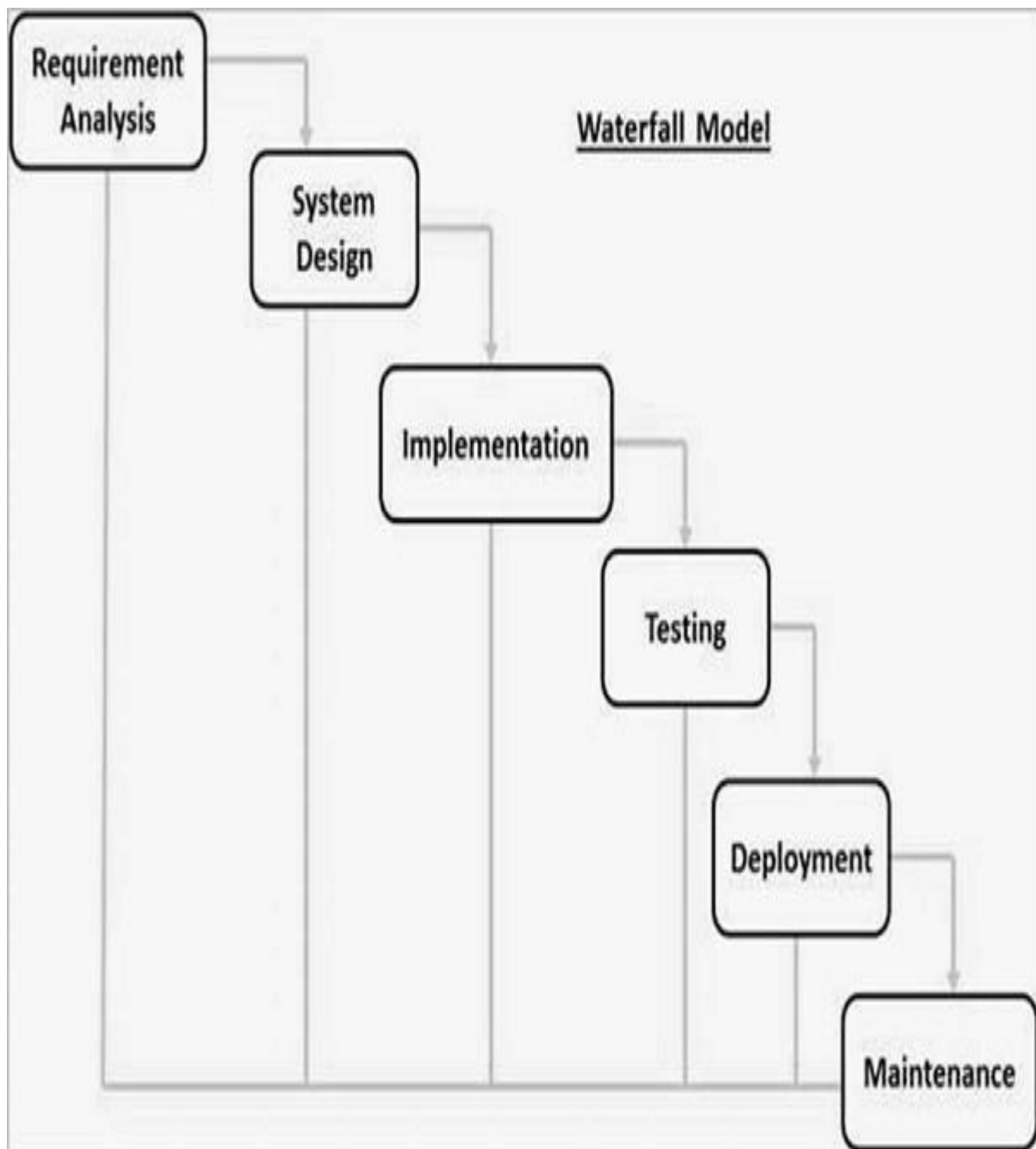
STAGE7: MAINTENANCE:

Once when the client starts using the developed systems, then the genuine issues come up and requirements to be solved from time to time.

3.3 WATERFALL MODEL:

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.



CHAPTER 4

4 REQUIREMENTS & ANALYSIS:

4.1 PROBLEM DEFINITION:

Cinema-going is one of the most popular out-of-home cultural activities, affecting a series of social, economic, and cultural phenomena in modern societies. Cinemas are an integral part of cities, and they contribute to the definition of a local geography and identity. They also constitute a significant social and cultural practice linked to a specific place, which acts as a common reference or landmark for many individuals. Through this project we present a comprehensive solution for ticket booking in multiplexes. Theatre management system, and ticket selling software that is easy to understand, easy to use and offers the simplicity of fast point-and-click service to the employee and admin. This powerful software program is specifically designed for theatre owners, to sell tickets. This intuitive visual interface makes day-to-day aspects of selling, exchanging, refunding, and reporting fast and easy for both the users and administrators. Theatre Management controls all back-end and front-end functionalities like, movies details, ticket rate and show time, customer information and sales history saved in database, etc. Theatre admin manages the report details like report details like counter wise report, daily, weekly, monthly report and movie report etc.

4.2 PLANNING AND SCHEDULING:

PERT CHART:

A project plan needs to be created to ensure the prompt complexity of the project. As part of project analysis, we break the project down to several stages and use a Gantt chart and PERT chart to describe specific tasks and status.

The Work Breakdown Structure of our proposed system “E-Commerce” is shown below:

<u>ID</u>	<u>Task Name</u>	<u>Duration</u>	<u>Start</u>	<u>Finish</u>
1	Project Initiation	3 days	17/07/22	19/07/22
2	Draft Project Plan	3 days	20/07/22	22/07/22
3	Analysis Phase	5 days	22/07/22	26/07/22
4	Plan User Interviews	2 days	26/07/22	27/07/22
5	Schedule users Interviews	1 days	28/07/22	28/07/22
6	Conducting users Interviews	1 days	28/07/22	28/07/22
7	System Design	10 days	29/07/22	07/8/22
8	Modules Design	3 days	06/8/22	08/08/22
9	Data Structure Design	3 days	07/08/22	09/08/22
10	User Interface Design	3 days	09/08/22	11/08/22
11	Coding Phase	13 days	08/08/22	20/08/22
12	Testing Phase	13 days	08/08/22	20/08/22
13	Integration Testing	5 days	20/08/22	24/08/22
14	System Level Testing	4 days	22/08/22	25/08/22
15	Implementation	2 days	25/08/22	26/06/22
16	Post-Implementation Review	3 days	26/08/22	28/08/22

4.3 REQUIREMENTS SPECIFICATION:

- Software Requirements:
 - 1 Java/JDK
 - 2 NetBeans
 - 3 MySQL
 - 4 SQL Workbench
- Hardware Requirements:
 - 1 Pentium IV Processor
 - 2 512 MB RAM
 - 3 40 GB HDD
 - 4 Color Monitor
 - 5 Keyboard, Mouse

4.4 PRELIMINARY PRODUCT DESCRIPTION:

MODULE DISCRIPTION:

This system will cover mainly two Modules, i.e.

1. User (EMPLOYEE).
2. Theatre (Admin).

❖ Theatre Module will cover FOUR Sub modules,

a. Adding a Show.

Theatre can add a show with timing and price of Ticket. This can be seen and booked by User.

b. Removing a Show.

Theatre can drop show if it was performed

c. Updating Show Details (Its Status).

Theatre can Update show timing and Price

d. Each Customer Detailing.

Theatre can see each user details, which is registered (booked a ticket before)

❖ **EMPLOYEE Module will cover FOUR Sub modules,**

1. REGISTER CUSTOMER
2. Booking a Ticket. (House full if there is no Ticket)
3. Booking Details. (Booking Status)
4. Updating CUSTOMER Profile.

4.5 CONCEPTUAL MODELS:

SYSTEM DEVELOPMENT LIFE CYCLE:

The System development life cycle (SDLC), or Software development processing systems engineering, information systems and software engineering, is a process of creating or altering information systems, and the models and method that people use to develop these systems. In software engineering, the SDLC concept underpins

many kinds of software development methodologies. These methodologies from the for planning and controlling the creation of an information system the software development process.

Broadly, following are the different activities to be considered while defining the system development life cycle for the said project:

- Problem Definition
- System Analysis
- Study of existing system
- Drawback of the existing system
- Proposed system
- System Requirement study
- Data flow analysis
- Feasibility study
- System design
- Input Design (Database & Forms)
- Updating
- Query /Report design
- Administration
- Testing

- Implementation
- Maintenance

1. SYSTEM ANALYSIS:

Systems analysis is the study of sets of interacting entities, including computer systems analysis. This field is closely related to requirements analysis or operations research. It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker) show a better course of action and make a better decision than he might otherwise have made.

System development can generally be thought of having two major components: systems analysis and systems design. In System Analysis more emphasis is given to understanding the details of an existing system or a proposed one and then deciding whether the proposed system is desirable or not and whether the existing system needs improvements. Thus, system analysis is the process of investigating a system, showing problems, and using the information to recommend improvement to the system.

1.2 SYSTEM DESIGN:

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. If the broader topic of product development "blends the perspective of marketing, design, and manufacturing into a single approach to Product development," then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user. Until the 1990s systems design had a crucial and respected role in the data processing industry. In the 1990s standardization of hardware and software resulted in the ability to build modular systems. The increasing importance of software running on generic platforms has enhanced the discipline of software engineering.

Object-oriented analysis and design methods are becoming the most widely used methods for computer systems design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modelling software systems and is increasingly used for high designing non-software systems and organizations.

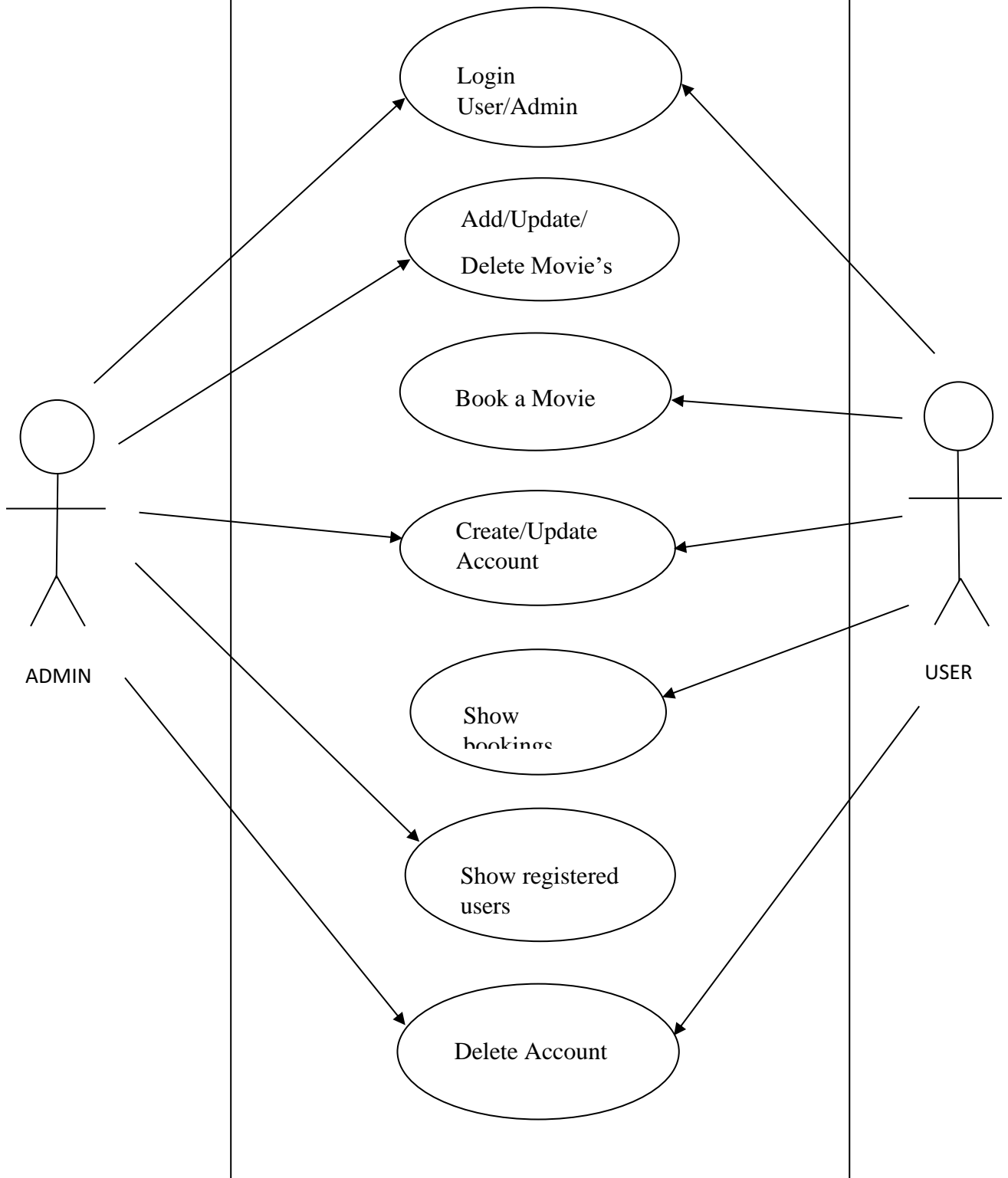
USECASE DIAGRAM:

In the unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use cases diagram can help your team discuss and represent:

- Scenarios in which your system or application interacts with people, Organizations, or external systems
- Goals that your system or application help those entities (known as actors) achieve.
- The scope of your system

UML is the modelling toolkit that you can use to build your diagrams. Use cases are represented with a labelled oval shape. Stick figures stands for actors in the process, and the actor's participation in the system is modelled with a line between the actor and use case. Depict the system boundary, draw a box around the use case itself.

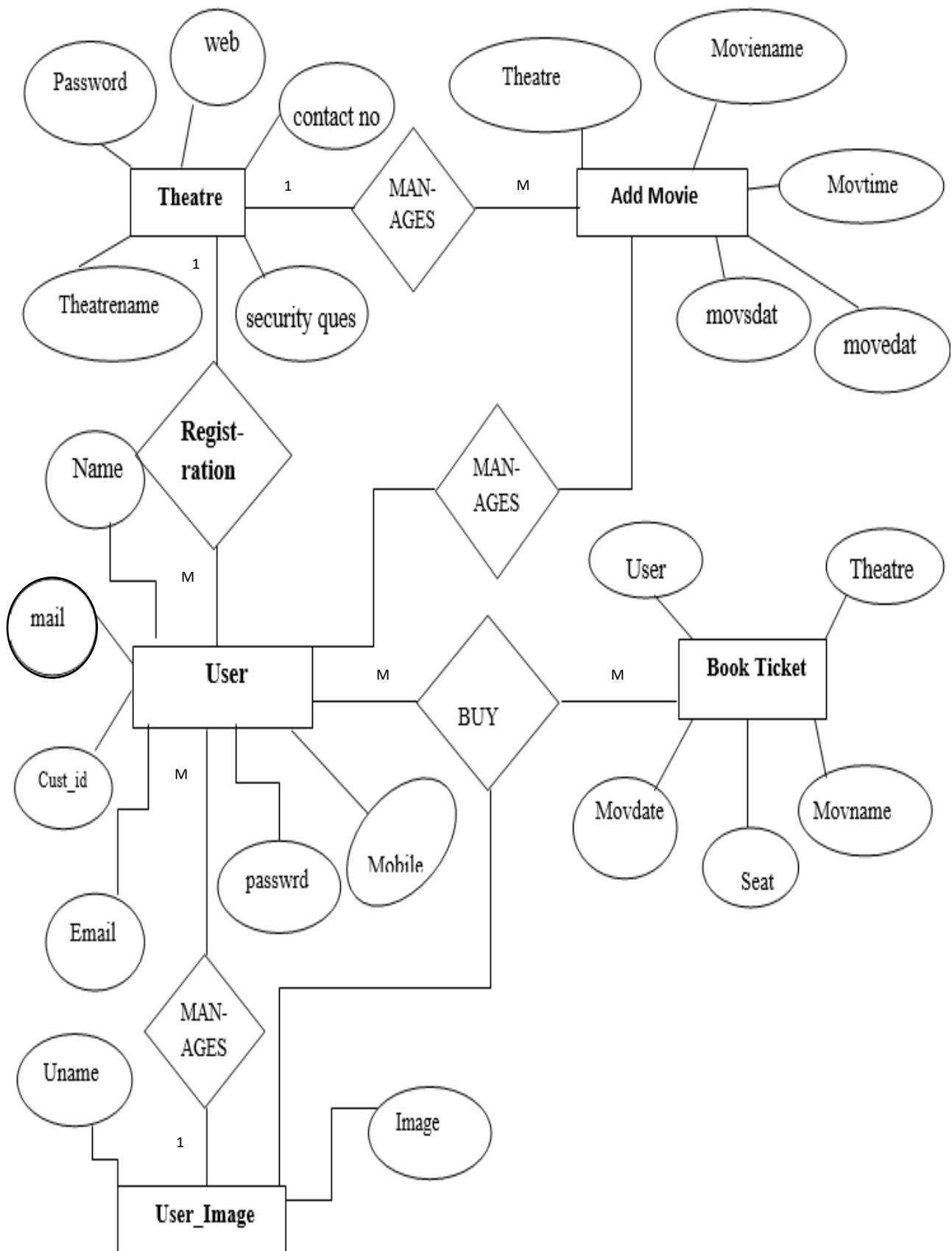
Use Case Diagram



ENTITY RELATION DIAGRAM:

The Entity Relation Model or Entity Relation Diagram (ERD) is a data model or diagram for high-level description of conceptual data model, and it supplies a graphical notation for being such data models in the form of entity relationship diagrams. Such models are typically used in the first stage of Management information system design; they are used for example, to describe information needs and/ or the type of information that is to be stored in the Database during the requirement analysis. The data modeling technique, however, can be used to describe any ontology (i.e., an overview and classification of used term and them relationships) for a certain universe of discourse (i.e., area of interest).

In the case of design, a Management Information System that is based on a database, the conceptual data model is, a later stage (usually called logical design), mapped to a logical data model such as, relational data model; this is turn in mapped to a physical model during physical design. Note that sometimes, both phases are referred a “physical design”. There are number of conventions for entity-relation diagrams (ERDs). The classical notation is described in the rest of this article, and mainly related to the conceptual modeling. There is a range of notation more typically employed in physical and logical database design.



ENTITY RELATIONSHIP DIAGRAM

DATA FLOW DIAGRAM:

The data flow diagram shows the flow of data within any system. It is important Tool for designing phase of software engineering. Larry Constantine first developed it. It stands for graphical view of flow of data. It's also known as BUBBLE CHART. The purpose of DFD is major transformation that will become in system design symbols used in DFD.

In the DFD, four symbols are used, and they are as follows:

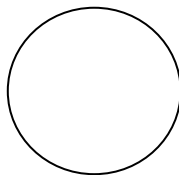
1. A square defines a source (originator) or destination of system data.



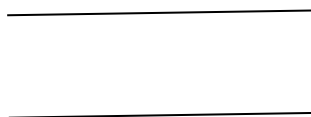
2. An arrow shows data flow-data in motion. It is a pipeline through which information flows.



3. A circle or a "bubble" (Some people use an oval bubble) stands for a process that transfers incoming data flows into outgoing data flow.

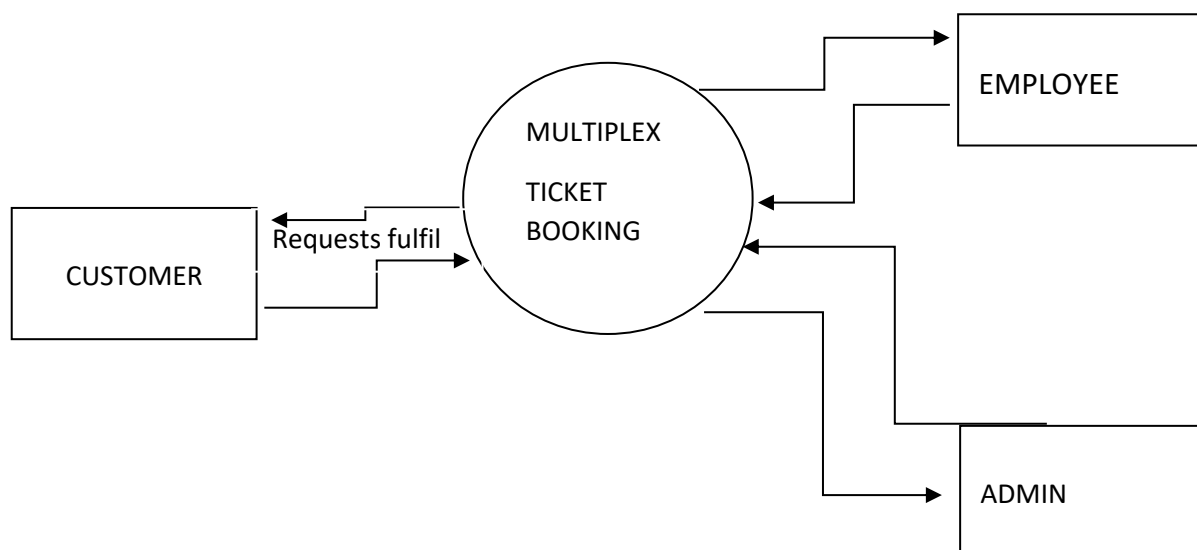


4. An open rectangle is a data store-data at rest, or a temporary Repository of data.



CONTEXT LEVEL DATA FLOW DIAGRAM:

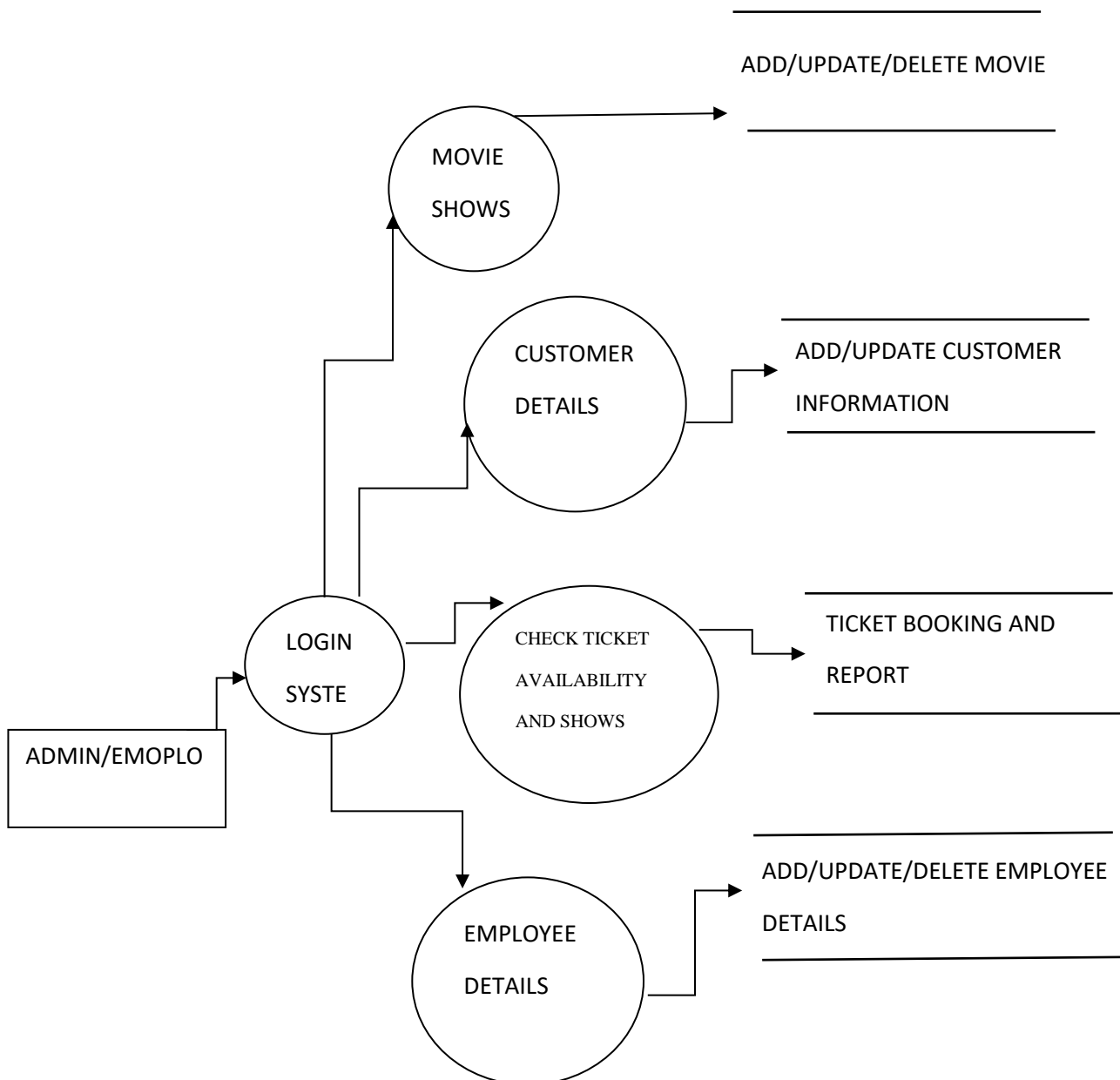
This level shows the overall context of the system and its operating environment And shows the entire system as just one process. Online bookstore is shown as one process in the context diagram, which is also known as zero level DFD, shown below. The context diagram plays significant role in understanding the system and deciding the boundaries. The main process can be broken into sub-processes and system can be studied with more detail; this is where 1st level DFD comes into play.



Zero Level Data Flow Diagram

First Level DFD:

This level (level 1) shows all processes at the first level of numbering, data stores, external entities and the data flows between them. The purpose of this level is to Show the major high-level processes of the system and their interrelation. A Process model will have one, and only one, level-1 diagram A level-1 diagram must be balanced with its parent context level diagram, I.e., there must be the same external entities and the same data flows, these can be broken down to more detail in the level 1.



ONE LEVEL DATA FLOW DIGRAM

CHAPTER 5

5. TESTING:

5.1 SYSTEM TESTING:

Black box testing method was used for system testing. The black box testing usually proves that software functions are operational; that the input is properly accepted, and the output is correctly produced, and that integrity of external information (databases) is supported.

5.2 WHY TESTING IS DONE:

- Testing is the process of running a system with the intention of finding errors.
- Testing enhances the integrity of a system by detecting deviations in design and errors in the system.
- Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system.
- Testing also add value to the product by confirming to the user requirements.

5.3 CAUSES OF ERRORS:

The most common causes of errors in a software system are:

- **Communication gap between the developer and the business decisionmaker:**

A communication gap between the developer and the business decision maker is normally due to subtle differences between them. The differences can be classified into five broad areas: Thought process, Background and Experience, Interest, Priorities, Language.

- **Time provided to a developer to complete the project:**

A common source of errors in projects comes from time constraints in delivering a product. Keep to the schedule, features can be cut. To keep the features, the schedule can be slipped. Not adjusting the feature set or schedule when problems are discovered can lead to rushed work and flawed systems.

- **Over Commitment by the developer:**

High enthusiasm can lead to over commitment by the developer. In these situations, developers are usually unable to adhere to deadlines or quality due to lack of resources or needed skills on the team.

- **Insufficient testing and quality control:**

Insufficient testing is also a major source of breakdown of e-commerce systems during operations, as testing must be done during all phases of development.

- **Inadequate requirements gathering:**

A fleeting time to market results in developers starting work on the Web site development without understanding the business and technical requirements. Also, developers may create client-side scripts using language that may not work on some client browsers.

- **Keeping pace with the fast-changing Technology:**

Innovative technologies are constantly introduced. There may not be adequate time to develop ability in the innovative technologies. This is a problem for two reasons. First, the technology may not be properly implemented. Second, the technology may not integrate well with the existing environment.

5.4 TESTING PRINCIPLES:

- Discover yet undiscovered errors.
- All tests should be traceable to customer's requirement.
- Tests should be planned long before the testing begins.
- Testing should begin "in the small" & progress towards "testing in the large".
- Exhaustive Testing is not possible.
- To be most effective training should be conducted by an Independent Third Party

5.5 TESTING OBJECTIVES:

- Testing is a process of executing a program with the intent of finding errors.
- A good test case is one that has a high probability of finding a yet undiscovered error.
- A successful test is one that uncovers a yet undiscovered error.

5.6 KINDS OF TESTING:

5.6.1 BLACK BOX TESTING:

Not based on any knowledge of internal designs or code. Tests are based on requirements and functionality.

5.6.2 WHITE BOX TESTING:

Based on the knowledge of the internal logic of an application's code. Tests are based on coverage of code statements, branches, paths, and statements.

5.6.3 UNIT TESTING:

The most 'micro' scale of testing; to evaluate functions and code modules. Typically done by the programmer and not by the testers, as it requires detailed knowledge of the internal program design and code. Not always easily done unless the application has a well-designed architecture with tight code; may require developing test driver modules or test harnesses.

5.6.4 INTEGRATION TESTING:

Testing of joint parts of an application to figure out if they function together correctly. The 'parts' can be code modules, individual applications, client, and server applications on a network, etc. This type of testing is especially relevant to client/ server and distributed systems.

5.6.5 FUNCTIONAL TESTING:

Black-box type testing geared to functional requirements of an application; testers should do this type of testing. This doesn't mean that the programmers shouldn't check that their code works before releasing it.

5.6.6 REGRESSION TESTING:

Re-testing after fixes or modifications of the software or its environment. It is difficult to figure out how much retesting is needed, especially near the end of the development cycles. Automated testing tools can be especially useful for this type of testing.

5.6.7 ACCEPTANCE TESTING:

Final testing based on the specifications of the end user or customer or based on use by end-users/ customers over some limited period.

5.6.8 USER ACCEPTANCE TESTING:

Figuring out if software is satisfactory to an end user customer

5.7 TESTING TECHNIQUE USED:

We will continuously evaluate our project to ensure that it is fully functional. To perform testing test cases are designed with the intent of finding the errors in the project and help in removing those errors. Testing begins at the module level and is conducted systematically. It is generally conducted by independent test groups or third party.

Testing is done in our project Multiplex Ticket Booking System with help of black box testing that exercises all the functional requirement of the project test cases are designed using these approaches by supplying set of input conditions to get the expected output.

CHAPTER 6


6.1 SCREENSHOT OF DATA TABLE:


THEATRE REGISTRATION:


Result Grid		Filter Rows:	Export:		Wrap Cell Cont	
	Field	Type	Null	Key	Default	Extra
▶	theatername	varchar(70)	YES		NULL	
	website	varchar(30)	YES		NULL	
	uname	varchar(25)	YES		NULL	
	cno	decimal(10,0)	YES		NULL	
	pass	varchar(20)	YES		NULL	
	sque	varchar(70)	YES		NULL	
	ans	varchar(25)	YES		NULL	

USER REGISTRATION:

Result Grid

 Filter Rows:


Export: 


Wrap Cell Content: 

	Field	Type	Null	Key	Default	Extra
▶	cid	int	NO	PRI	NULL	auto_increment
	Name	varchar(20)	YES		NULL	
	Mail	varchar(30)	YES		NULL	
	Username	varchar(25)	YES		NULL	
	Pass	varchar(25)	YES		NULL	
	Mob	decimal(10,0)	YES		NULL	
	Security	varchar(70)	YES		NULL	
	Answer	varchar(25)	YES		NULL	


ADD MOVIE:

Result Grid

 Filter Rows:



 Export:

Wrap Cell Content:



	Field	Type	Null	Key	Default	Extra
▶	theater	varchar(30)	YES		NULL	
	movname	varchar(60)	YES		NULL	
	movtim	varchar(15)	YES		NULL	
	movsdat	varchar(15)	YES		NULL	
	movedate	varchar(20)	YES		NULL	
	movdes	varchar(1500)	YES		NULL	
	movpri	varchar(10)	YES		NULL	
	movpos	longblob	YES		NULL	

BOOK TICKET:

Result Grid		Filter Rows: <input type="text"/>		Export: 		Wrap Cell Content: 	
	Field	Type	Null	Key	Default	Extra	
▶	User	varchar(30)	YES		NULL		
	theater	varchar(25)	YES		NULL		
	movnam	varchar(50)	YES		NULL		
	movtim	varchar(15)	YES		NULL		
	movdat	varchar(20)	YES		NULL		
	seat	varchar(10)	YES		NULL		
	tpric	varchar(10)	YES		NULL		

USER IMAGE:

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

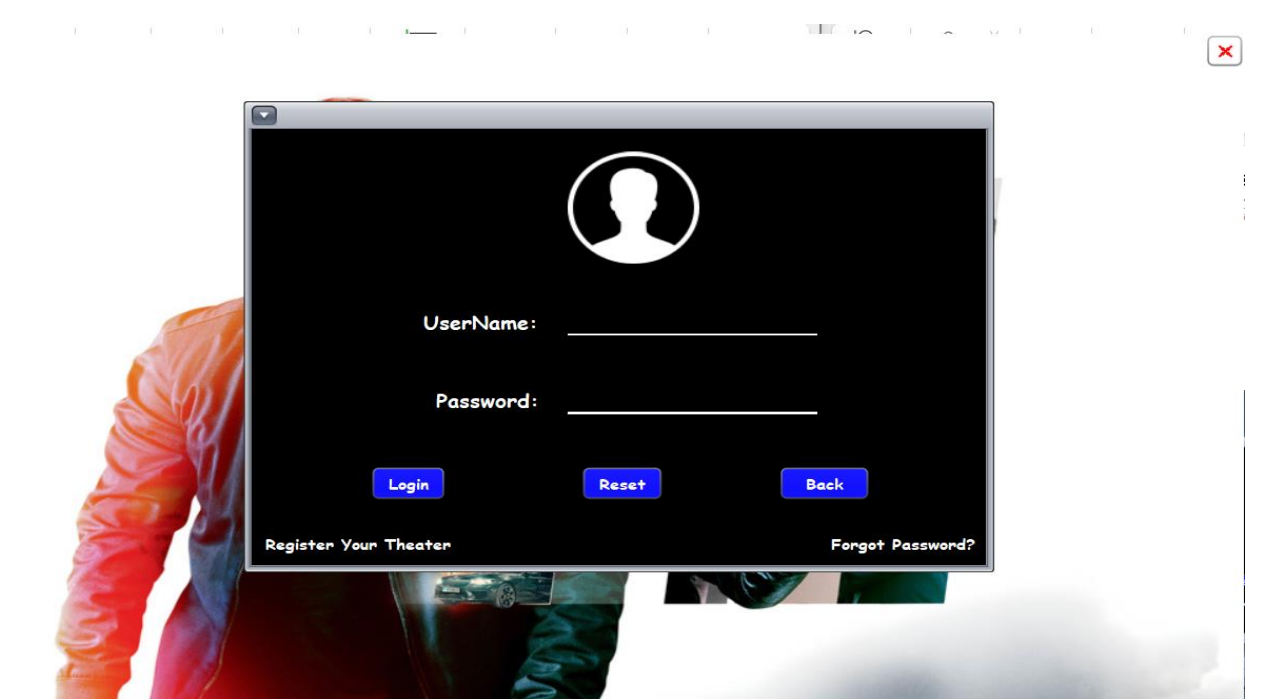
	Field	Type	Null	Key	Default	Extra
▶	uname	varchar(20)	YES		NULL	
	img	longblob	YES		NULL	

6.2 PROJECT SCREENSHOT:

MAIN WINDOW:



THEATRE LOGIN WINDOW:



REGISTER YOUR THEATRE:

Theater Name: Website:

UserName: Contact no.

Password: Confirm Password:

Security Question:

Answer:

* Write mall name/landmark with theater name if another branch exist.
* Write your Name and Username carefully .

FORGOT PASSWORD:

Security Question:

Answer:

New Password:

Confirm Password:

ADD SHOW:



PVR

Add a show

Delete a show

Update show details

User Details

Update Profile

Delete your account

Sign Out

Movie Name : Spider-Man: No Way Home

Movie Timming : 12:30 Movie Price : 390

Starting Date : 2022-08-30 Ending Date : 2022-09-17

Movie Poster : _1643979290388.webp

Movie Description : Spider-Man seeks the help of Doctor Strange to forget his exposed secret identity as Peter Parker. However, Strange's spell goes horribly wrong, leading to unwanted guests entering their universe.

ADD

DELETE A SHOW:



PVR

Add a show

Delete a show

Update show details

User Details

Update Profile

Delete your account

Sign Out

Movie Name : wqw **Search**

Movie Timming : 12:00

Starting Date : 2018-07-06 Ending Date : 2018-07-26

Delete

UPDATE SHOW DETAILS:



pvr

Add a show

Delete a show

Update show details

User Details

Update Profile

Delete your account

Sign Out

Movie Name : 3 Idiots

Search

Movie Timming : 10am

Movie Price : 180

Starting Date : 2009-12-25

Ending Date : 2010-01-14

Movie Poster : -lesson-for-bloggers1.jpg

Movie Description : Two friends are searching for their long lost companion. They revisit their college days and recall the memories of their friend who inspired them to think differently, even as the rest of the world called them "idiots".

Update

USER DETAILS:



pvr

Add a show

Delete a show

Update show details

User Details

Update Profile

Delete your account

Sign Out

Name	E-mail	Contact Number
Sunny	mesunny@gmail.com	77367263
ADAM	wo	83862637
Inshaal	sjdv	7868768
Akash	akash17250@gmail.com	7054419782

UPDATE PROFILE:



pvr

Add a show

Delete a show

Update show details

User Details

Update Profile

Delete your account

Sign Out

Name: Username:

Website: Contact no.:

Password:

Security Question:

Answer:

DELETE YOUR ACCOUNT:



pvr

Add a show

Delete a show

Update show details

User Details

Update Profile

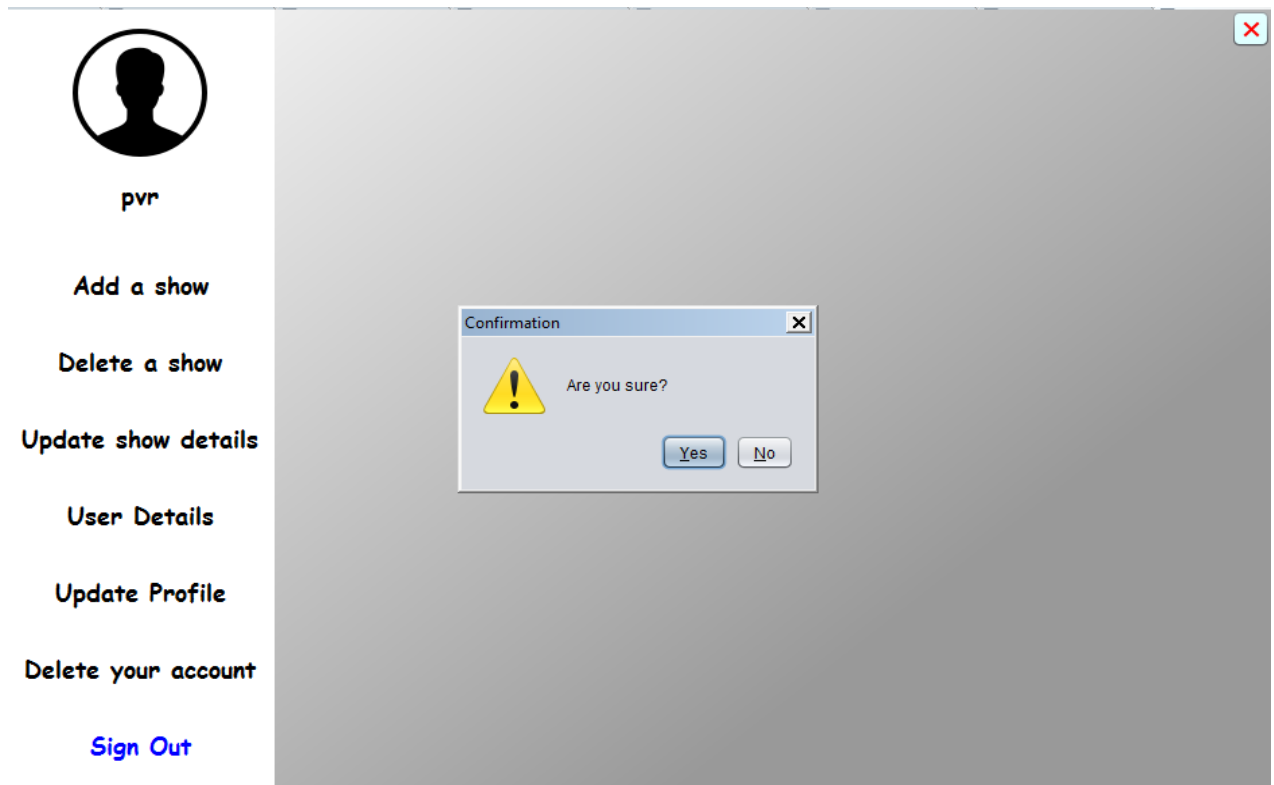
Delete your account

Sign Out

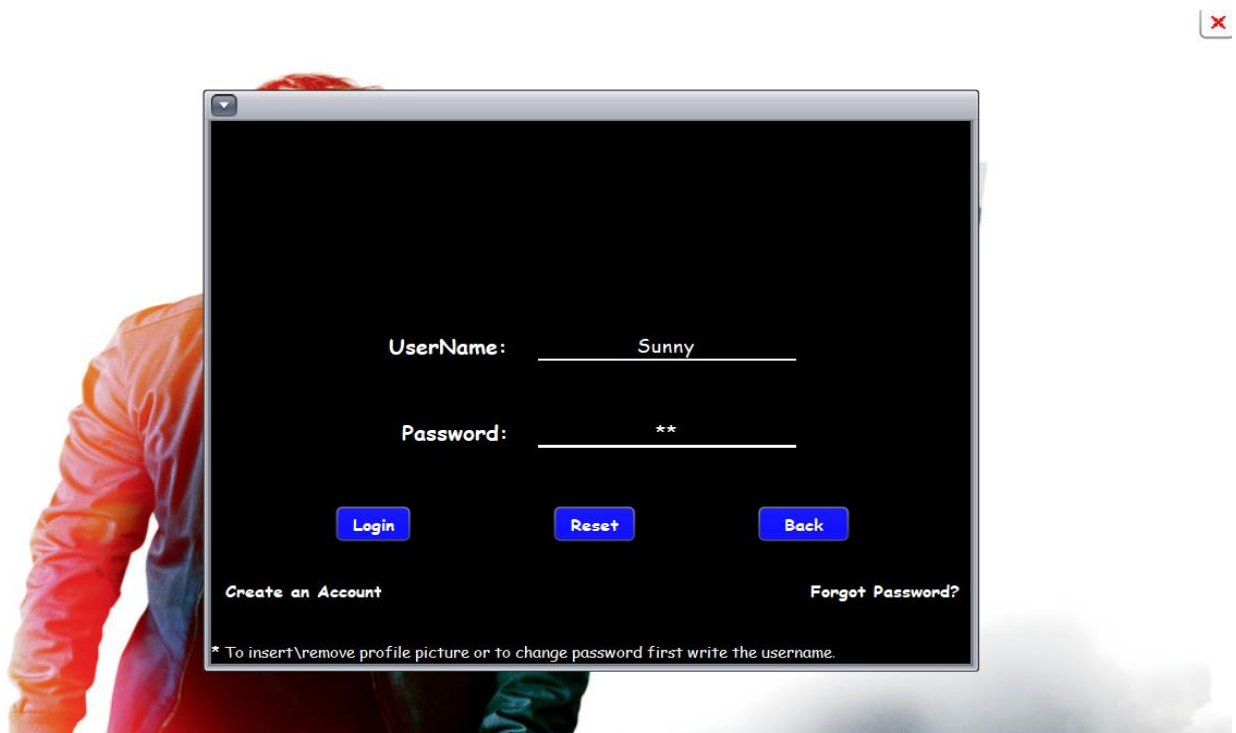
Warning

! You will no longer access this account

SIGN OUT:



6.3 USER LOGIN WINDOW:



CREATE AN ACCOUNT:

Name: E-mail:

UserName: Mobile:

Password: Confirm Password:

Security Question:

Answer:

* Your customer ID will be autogenerated .
* Write your Name and Username carefully .

FORGOT PASSWORD:

Security Question:

Answer:

New Password:

Confirm Password:

BOOK A TICKET:

Prakhar

Book a Ticket

Booking History

Update Profile

Theater Contact Details

Delete Your Account

Not Prakhar

Sign Out ?

Theater Name: PVR

Movie Name: Doctor Strange

Movie Timmings: 2:24

Movie Date: 2022-09-14

No. of Seats: 3

Total Price: 900

About this movie

Confirm Ticket

ABOUT THIS MOVIE:

Prakhar

Book a Ticket

Booking History

Update Profile

Theater Contact Details

Delete Your Account

Not Prakhar

Sign Out ?

Description :

In an accident, Stephen Strange, a famous neurosurgeon, loses the ability to use his hands. I

Available in : PVR,

Ok

BOOKING HISTORY:

Prakhar

[Book a Ticket](#)

[Booking History](#)

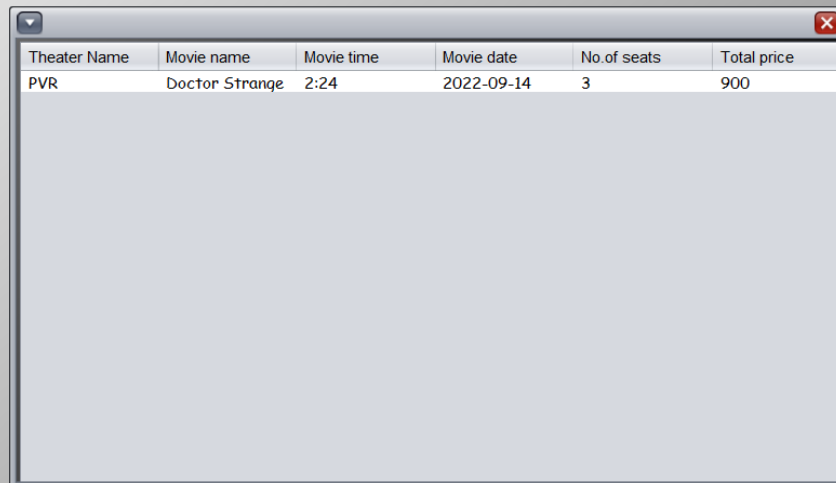
[Update Profile](#)

[Theater Contact Details](#)

[Delete Your Account](#)

[Not Prakhar](#)

[Sign Out ?](#)



Theater Name	Movie name	Movie time	Movie date	No.of seats	Total price
PVR	Doctor Strange	2:24	2022-09-14	3	900

UPDATE PROFILE:

Prakhar

[Book a Ticket](#)

[Booking History](#)

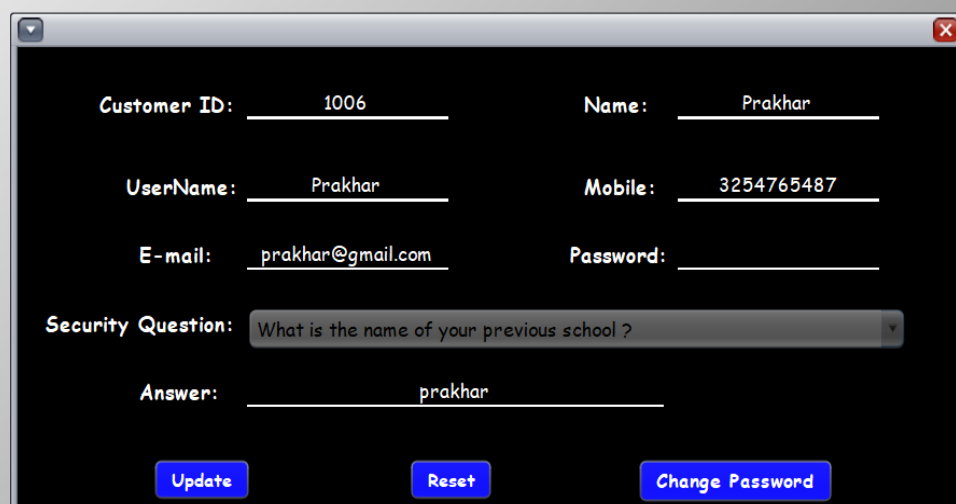
[Update Profile](#)

[Theater Contact Details](#)

[Delete Your Account](#)

[Not Prakhar](#)

[Sign Out ?](#)



Customer ID: Name:

UserName: Mobile:

E-mail: Password:

Security Question:

Answer:

THEATRE CONTACT DETAILS:

Prakhar

[Book a Ticket](#)

[Booking History](#)

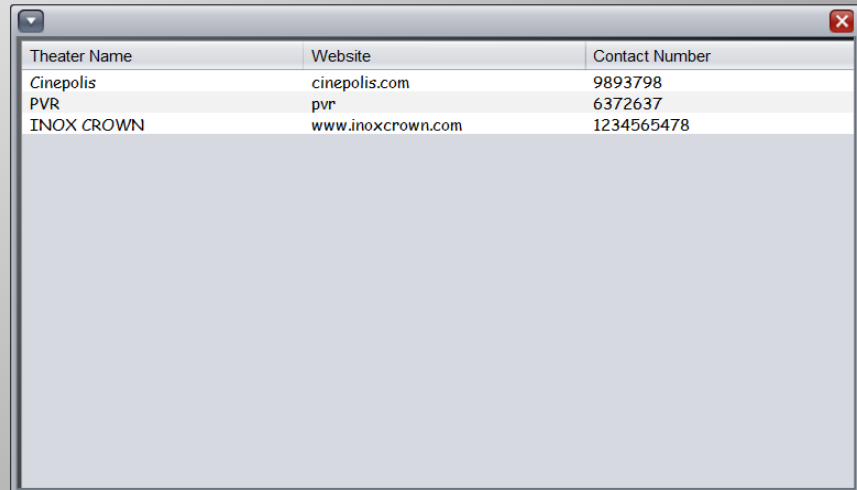
[Update Profile](#)

[Theater Contact Details](#)

[Delete Your Account](#)

[Not Prakhar](#)

[Sign Out ?](#)

A warning dialog box with a red exclamation mark icon and the text "Warning" in the title bar. The message inside says "You will no longer access this account". There are "OK" and "Cancel" buttons at the bottom.

Theater Name	Website	Contact Number
Cinepolis	cinepolis.com	9893798
PVR	pvr	6372637
INOX CROWN	www.inoxcrown.com	1234565478

DELETE YOUR ACCOUNT:



Akash

[Book a Ticket](#)

[Booking History](#)

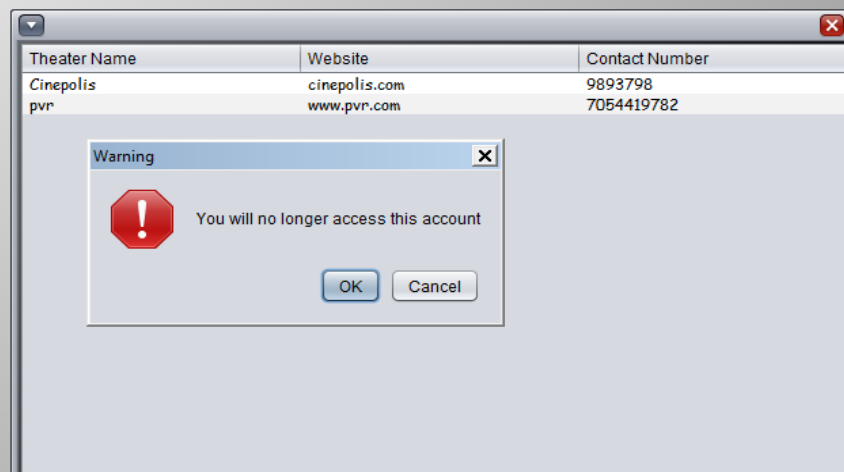
[Update Profile](#)

[Theater Contact Details](#)

[Delete Your Account](#)

[Not Akash](#)

[Sign Out ?](#)

A warning dialog box with a red exclamation mark icon and the text "Warning" in the title bar. The message inside says "You will no longer access this account". There are "OK" and "Cancel" buttons at the bottom.

Theater Name	Website	Contact Number
Cinepolis	cinepolis.com	9893798
pvr	www.pvr.com	7054419782

SIGN OUT:

Prakhar

Book a Ticket

Booking History

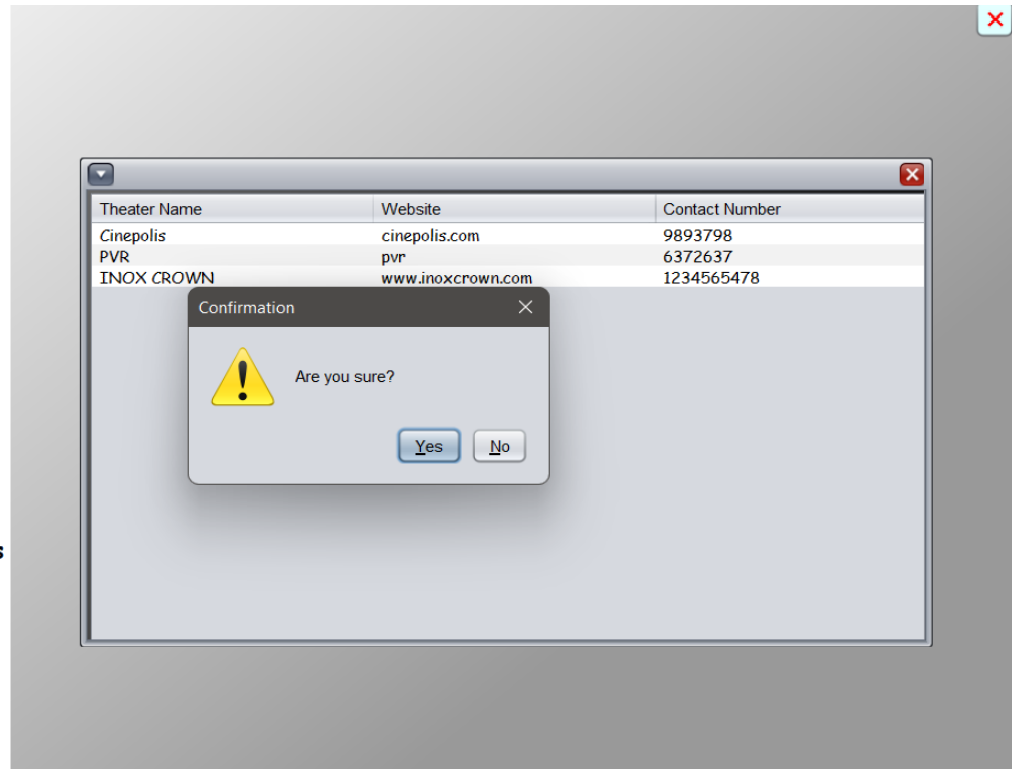
Update Profile

Theater Contact Details

Delete Your Account

Not Prakhar

Sign Out ?



CHAPTER 7

7 IMPLEMENTATIONS:

7.1 SYSTEM IMPLEMENTATION:

During the implementation stage the stage the system in physically created. Necessary programs are coded, debugged and document. A new hardware is Selected, ordered, and installed.

7.2 SYSTEM SECIFICATION:

Every computer system consists of three major elements.

1. The Hardware
2. Application software such as visual studio
3. Operating System

For successful operation of the package following must be kept in mind:

- To many packages should not be used, as very few systems may all those Packages Installed due to memory problem. Thus, the compatibility of the system development will get reduced.

FUTURE SCOPE OF PROJECT

The proposed system helps them in many ways. It helps them do billing very easily. Account maintenance also becomes easier. They can keep track of their all-movie Details And customer account details. The software is provided with all the master entries to enter any new movies or to add or change and remove the movie shows and timings.

As this is generic software it can be used by a wide variety of multiplex cinema halls to automate the process of manually keep the record related to the subject of maintaining the movie details and customer data. In future it can be change, so that it can be done online.

CONCLUSION

The software will be developed by implementing the concept of modularity which It turns reduces the complexity involved in keeping it. The administrator should Have a sound technical knowledge about keeping the software and further Enhancements will be undertaken by the developer. The application is portable which ensure its adaptability for use on different computer Terminals with different operating system and standards.

The factor guarantees the software's availability includes proper termination and correct Input details. Also, t resources used for the project development are Microsoft certified Which speaks of its high-quality standards.

Hence, we may conclude that the application system being developed helps a great deal In changing the computerized MULTIPLEX TICKET BOOKING SYSTEM.

REFERENCE

BOOK REFERENCE:

The following books were referred during the analysis and execution phase of the project:

By Rajib Mall - Fundamental of Software Engineering - PHI Learning Private Limited, 2014

By Herbert Scheldt - The Complete Reference Java – Mc Grew Hill, 2007

By Gradi Booch, Ivar Jacobson - UNIFIED MODELING LANGUAGE – Addison Wesley, 1998

by Saied Tabagboghi & Hugh E. Williams - Leaning MySQL – O'REILLY, November 2006

GitHub - Java AWT Reference

Software Engineering – <https://www.javatpoint.com/>

IMAGES:

-Google Search

-Bing Search

-Msn Search

DESIGNING AND IMPLEMENTATION PHASE:

1. Software engineering: a practitioner's approach by roger s pressman.
2. System analysis and design by Elias m. Ewad.
3. DBMS: Bipin C Desai
4. Introduction to dbms – shailendrasharma
5. Modern database management -drkumar
6. Fundamentals of database systems- ramezelmsari
7. An intro to dbms- christhopher j date
8. Database system concepts-abrahamsilbersch
9. Database management systems-raghuramkrishnan
10. Principles of database systems- jd Ullman
11. Foundation of databases- sergebabiteboul
12. Database systems-connolly