

**DESIGN AND DEVELOPMENT OF
MOVIETIC**

A

MAJOR PROJECT-I REPORT

Submitted in partial fulfillment of the requirements

for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

By

GROUP NO. 05

Daksh Raj Singh	0536CS201012
Dikshant Kumar Bhatiya	0536CS201014
Deepak Kumar	0187CS201048
Naina Soni	0187CS201103
Shreya Singh	0187CS201161

Under the guidance of

Prof. Bhavna Soni
(Assistant Professor)



December - 2023

Department of COMPUTER SCIENCE & ENGINEERING

Sagar Institute of Science & Technology (SISTec)

Bhopal (M.P.)

Approved by AICTE, New Delhi & Govt. of M.P.

Affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.)

Sagar Institute of Science Technology (SISTec), Bhopal
Department of COMPUTER SCIENCE ENGINEERING
Bhopal (M.P.)



December - 2023

CERTIFICATE

I hereby certify that the work which is being presented in the B.Tech Major Project-I Report entitled **MOVIETIC** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering** and submitted to the Department of Computer Science & Engineering, Sagar Institute of Science & Technology (SISTec), Bhopal (M.P.) is an authentic record of my own work carried out during the period from July-2023 to December-2023 under the supervision of **Prof. Bhavna Soni (Project guide)**. The content presented in this project has not been submitted by me for the award of any other degree elsewhere.

Signature

Daksh Raj Singh	0536CS201012
Dikshant Kumar Bhatiya	0536CS201014
Deepak Kumar	0187CS201048
Naina Soni	0187CS201103
Shreya Singh	0187CS201161

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Prof. Bhavna Soni
Project Guide

Dr. Rahul Dubey
HOD

Dr. Dinesh Kumar Rajoriya
Principal

ABSTRACT

To define, design, and implement a Database to aid in comprehensively managing all aspects of a theatre. This system deals with a single branch of a theatre. It keeps track of the various halls, currently showing movies, show timings, booked tickets, and price listings in the theatre as well as the associated attributes. It is possible for the management of the theatre to use the database to fully understand all parts of the system without use of additional software. The system allows a cashier to add tickets for a given show, and for a manager to add movies and shows to the database. The system automatically checks the validity of the new entries and disallows incorrect data from being entered. This is done automatically without user intervention

ACKNOWLEDGEMENT

We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. We extend our sincere and heartfelt thanks to our esteemed guide,

Prof. Bhavna Soni , for providing us with the right guidance and advice at the crucial junctures and for showing us the right way. We also take this opportunity to express a deep sense of gratitude to **Dr. Rahul dubey (HOD)** and **Prof. Deepti Jain (Project Coordinator)** . We also wish to express our gratitude to **Dr. Swati Saxena (Vice Principal)** and **Dr Dinesh Kumar Rajoriya (Principal)**.

First of all, I would like to thanks all those people who helped me directly or indirectly to complete my project whenever I found my self in problems. Our all faculties encourages me and due to their kindness and helpful nature and help I got very much confidence to complete this project.

I am deeply inherited who devoted his precious time in giving me the information about the various aspect and gave support and guidance at every point of time. I am really thankful to their kind and supportive nature. His inspiring nature has always made my work easy.

It's a very difficult task to build a project from scratch and take it up to the expectation of our teachers, I would like to thank my teachers my friends who helped us in this crucial time and encouraged us to build a project up to the mark so that we can present it to our teachers. We would like to thank our friends and family for the support and encouragement they have given us during the course of our work.

LIST OF FIGURES

FIG. NO.	TITLE	PAGE NO.
6.1	MovieTic Use Case Diagram.	10
6.2	MovieTic ER Diagram.	11
7.1	Login page.	12
7.2	Manager page.	12
7.2.1	Manager page (View Booking)	13
7.2.2	Manager page (Insert a Movie)	14
7.2.3	Manager page (Schedule a show)	15
7.2.4	Manager page (Alter Prices)	15
7.3	Cashier page.	16
7.3.1	Cashier page (Pick a Date).	16

LIST OF ABBREVIATIONS

ACRONYM	FULL FORM
HTML	Hyper Text Mark-up Language
CSS	Cascading Style Sheet
VSCODE	Visual Studio Code
AJAX	Asynchronous JavaScript And XML
XML	Extensible Markup Language
SQL	Structured Query Language
HTTP	HyperText Transfer Protocol
PIP	Python Package Manager
Distro	Distribution

TABLE OF CONTENTS

TITLE	PAGE NO.
Abstract	iii
Acknowledgement	iv
List of figures	v
List Of Abbreviations	vi
Chapter 1 Introduction	1
1.1 Introduction	1
1.1.1 Purpose of the project	1
1.1.2 Scope	1
1.1.3 Existing System	1
1.1.4 Proposed System	1
Chapter 2 Software And Hardware Requirements	2
2.1 Recommended Operating System	2
2.2 Software Requirements	2
2.3 Hardware Requirements	2
Chapter 3 Problem Description	3
Chapter 4 Literature Survey	4
Chapter 5 Software Requirements Specification	6
5.1 Functional Requirement	6
5.1.1 Manager	6
5.1.2 Cashier	6
5.2 Non-Functional Requirements	7
5.2.1 Performance And Scalability	7
5.2.2 Portability And Compatibility	7
5.2.3 Reliability,Maintainability And Availability	8
5.2.4 Security	8
5.2.5 Usability	9
Chapter 6 Software Design	10
6.1 Use Case Diagram	10
6.2 ER Diagram	11
Chapter 7 Output Screens	12

Chapter 8	Conclusion And Future Work	17
	References	18
	Project Summary	19

Chapter 1

Introduction

CHAPTER 1

INTRODUCTION

1.1 Introduction

This project is a Theatre Management System. It is a website made in HTML connected to a MySQL Database using Python with JavaScript and CSS enhancements. The database has tables dealing with all aspects of a theatre: halls, movies available, current showing, price lists, and booked tickets. These are accessed by the webpage to show relevant details for a user. Thus, by use of this project, the core aspects of a theatre are easily maintained and manipulated as needed by the manager.

1.1.1 Purpose of the project

This project is aimed at small theatres. It is a tool to be used to simplify the process of adding, storing, viewing and manipulating the movies available at the theatre, the various showings of the movie, the tickets booked for the shows and the pricing of the tickets. It is a robust and comprehensive system that is minimalistic and simple for use by cashiers and managers who have no prior training.

1.1.2 Scope

The scope of this project is at the ticket booking kiosk of the cashier and at the manager's desk. It is only concerned with active use of resources of the theatre and does not have long term storage of information. It only deals with movies, movie showings and booked tickets, and does not handle other aspects, such as online booking, concessions, etc.

1.1.3 Existing System

The currently used system in theatre management is highly primitive and/or inefficient and/or slow. It is, in some cases, worked on in ledgers and with a manual booking system. Even in cases where it is on computers instead of a manual system, it is not new or often upgraded/maintained as required to make it work properly. Thus, our system is brought in to replace it.

1.1.4 Proposed System

The proposed system is a fully functional system that takes care of all aspects related to movies on-site. It allows cashiers to book tickets and the manager to add movies and shows. The system is designed so as to be used by anyone with a basic understanding of english and no other training at all.

Chapter 2

Software & Hardware Requirements

CHAPTER 2

SOFTWARE AND HARDWARE REQUIREMENT

2.1 RECOMMENDED OPERATING SYSTEM

- **Windows:** 7 or newer
- **MAC:** OS X v10.7 or higher
- **Linux:** Ubuntu

2.2 SOFTWARE REQUIREMENTS

People often ask what browser they should use. There is no single answer for this. Use whichever browser works best on your computer. However, we recommend downloading Firefox and/or Chrome in addition to having Internet Explorer or Safari.

- [Firefox](#)
- [Chrome](#)

2.3 HARDWARE REQUIREMENTS

We strongly recommend a computer fewer than 5 years old.

- Processor: Minimum 1 GHz; Recommended 2GHz or more
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
- Hard Drive: Minimum 32 GB; Recommended 64 GB or more
- Memory (RAM): Minimum 1 GB; Recommended 4 GB or above

Chapter 3

Problem Description

CHAPTER 3

PROBLEM DESCRIPTION

Small theatres generally have old or primitive management systems which they do not replace or upgrade as it is perceived to be not worth the cost compared to revenues. However, old systems can be very slow and inefficient and as such, slow down the process. Furthermore, they are difficult to maintain and the cost of doing so can build up over time and eventually exceed costs of not replacing the system. This Project is designed for use by such cinemas, who can use this highly minimalistic design to replace current systems at low cost and with no additional training requirements. The code behind the project is designed to put maximum functionality within it so as reduce cost and difficulty of maintenance and future upgrades.

Chapter 4

Literature Survey

CHAPTER 4

LITERATURE SURVEY

Theatre Management and Ticket Booking Systems (TMTBS) have evolved significantly in recent years due to advancements in technology and changes in consumer expectations. This literature survey provides an overview of key research and developments in the field of TMTBS, highlighting trends, challenges, and innovative solutions.

1. Ticket Booking Trends:

Online Ticket Booking and Mobile Apps: Research has shown a notable shift from traditional box office sales to online ticket booking platforms and mobile apps. Customers prefer the convenience of booking tickets from their smartphones and computers (Guttentag, 2015).

Dynamic Pricing: Studies suggest that dynamic pricing strategies, which adjust ticket prices based on demand and availability, have gained prominence in the industry. These strategies can optimize revenue for theatres (Chen et al., 2019).

2. User Experience and Customer Engagement:

Personalization: Researchers emphasize the importance of personalization in TMTBS. Personalized recommendations, targeted promotions, and tailored communication enhance the customer experience (Vural et al., 2018).

Customer Feedback Analysis: Literature highlights the significance of analyzing customer feedback and reviews to improve services and shows. Natural language processing techniques are employed for sentiment analysis (He et al., 2017).

3. Show Scheduling and Management:

Optimization Algorithms: Operations research and optimization algorithms are used to efficiently schedule shows, allocate resources, and minimize conflicts in multi-screen and multi-event venues (Agrawal et al., 2016).

4. Security and Privacy:

Data Security: Ensuring data security in TMTBS is critical. Research explores encryption methods, access controls, and compliance with data protection regulations (Shetty et al., 2019).

5. Integration and Scalability:

Integration with Third-party Systems: Researchers emphasize the importance of integrating TMTBS with external systems, such as payment gateways, marketing tools, and analytics platforms, to enhance functionality (Kim et al., 2020).

Scalability: Scalability is a key concern in TMTBS. Studies investigate scalable architecture design and cloud-based solutions to handle peak traffic loads (Ghani et al., 2017).

6. Emerging Technologies:

Blockchain: Some studies explore the potential of blockchain technology for secure ticketing, reducing fraud, and enhancing transparency in the ticketing process (Hassan et al., 2019).

Artificial Intelligence: AI and machine learning are employed for predictive analytics in ticket sales and customer behavior analysis (Fan et al., 2018).

7. COVID-19 Impact:

The COVID-19 pandemic significantly impacted the theatre industry. Research discusses how TMTBS adapted to the pandemic, including the implementation of social distancing measures, contactless ticketing, and virtual shows (Ku et al., 2021).

The literature survey reveals a dynamic and evolving landscape in Theatre Management and Ticket Booking Systems. Technology-driven innovations, customer-centric approaches, and adaptability to changing circumstances, such as the pandemic, are reshaping the theatre industry. Future research in this field will likely continue to explore emerging technologies and novel strategies to enhance the theatre experience while ensuring efficiency and security in ticketing and management processes.

Chapter 5

Software Requirement

Specification

CHAPTER 5

SOFTWARE REQUIREMENTS SPECIFICATION

5.1 FUNCTIONAL REQUIREMENTS:

5.1.1 Manager :

1. The ability to insert new Movies. Movie name, language, length, showing start date and end date are inputted. In case of incorrect or invalid input, entry is nullified. If correctly entered, then a movie ID is randomly generated and the data is passed to MySQL via JavaScript, Flask and Python.
2. Similarly, Shows can be entered by entering movie, hall, time and date. If correctly entered, a show ID is randomly generated and ask data is put into database. Then, the price ID is fetched and stored after. Shows cannot be added if an invalid type is chosen or if the hall in which it is to be shown is already occupied.
3. The manager can view tickets that have been booked by the cashier, such as tickets numbers and seat numbers for a given show that have been booked.
4. The manager can alter prices. The list of prices based on day and your is brought up. Then, a price is then selected and altered.
5. The cashier had one functionality. The cashier books tickets for the customer. The cashier selects a movie, and based on that, a showing is selected for a time and date. Then a seating chart is brought up and a seat is chosen. Then a unique ticket ID is generated and the data is stored in the database.
6. All data is stored on a common database. All systems in the theatre (cashier's terminals and manager's computer) access the same database for their bookings. Multiple branches of the same theatre may share the same database, but different theatres use different database.

5.1.2 Cashier :

1. The cashier had one functionality. The cashier books tickets for the customer. The cashier selects a movie, and based on that, a showing is selected for a time and date. Then a seating chart is brought up and a seat is chosen. Then a unique ticket ID is generated and the data is stored in the database.
2. All data is stored on a common database. All systems in the theatre (cashier's terminals and manager's computer) access the same database for their bookings. Multiple branches of the same theatre may share the same database, but different theatres use different database.

5.2 NON-FUNCTIONAL REQUIREMENTS:

5.2.1 PERFORMANCE AND SCALABILITY.

- How fast does the system return results? How much will this performance change with higher workloads?

Performance Deals with the measure of the system's response time under different load conditions.

Example of performance requirements:

- The landing page supporting 5,000 users per hour must provide 6 second or less response time in a Chrome desktop browser, including the rendering of text and images and over an LTE connection.

Scalability assesses the highest workloads under which the system will still meet the performance requirements.

Example of scalability requirements:

- The system must be scalable enough to support 1,000,000 visits at the same time while maintaining optimal performance

5.2.2 PORTABILITY AND COMPATIBILITY.

- Which hardware, operating systems, and browsers, along with their versions does the software run on? Does it conflict with other applications and processes within these environments?

Portability determines how a system or its element can be launched within one environment or another

Example of portability requirements:

- A program running on Windows 10 must be able to run on Windows 11 without any change in its behavior and performance

Compatibility, as an additional aspect of portability, defines how a system can coexist with another system in the same environment.

Example of compatibility requirements:

- The iOS application must support iPhone devices running on OS versions,3.63.3,3.4

5.2.3 RELIABILITY, MAINTAINABILITY AND AVAILABILITY.

- How often does the system experience critical failures? How much time does it take to fix the issue when it arises? And how is user availability time compared to downtime

Reliability specifies how likely the system or its element would run without a failure for a given period of time under predefined conditions.

Example of reliability requirements:

- The system must perform without failure in 95 percent of use cases during a month.

Maintainability defines the time required for a solution or its component to be fixed, changed to increase performance or other qualities, or adapted to a changing environment.

Example of maintainability requirements:

- The mean time to restore the system (MTTRS) following a system failure must not be greater than 10 minutes. MTTRS includes all corrective maintenance time and delay time

Availability describes how likely the system is accessible to a user at a given point in time.

Example of availability requirements:

- The web dashboard must be available to US users 99.98 percent of the time every month during business hours EST.

5.2.4 SECURITY.

- How well are the system and its data protected against attacks?

Security is a non-functional requirement assuring all data inside the system or its part will be protected against malware attacks or unauthorized access

Example of security requirement:

- The user password saved in hash form in database that protect from attacker

5.2.5 USABILITY.

- How easy is it for a customer to use the system?

Usability indicates how effectively and easy user can learn and use a system

Example of Usability requirement:

- User easily navigate by simplicity and easy user interface of our website

Chapter 6

Software Design

CHAPTER 6

SOFTWARE DESIGN

6.1 USE CASE DIAGRAM

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

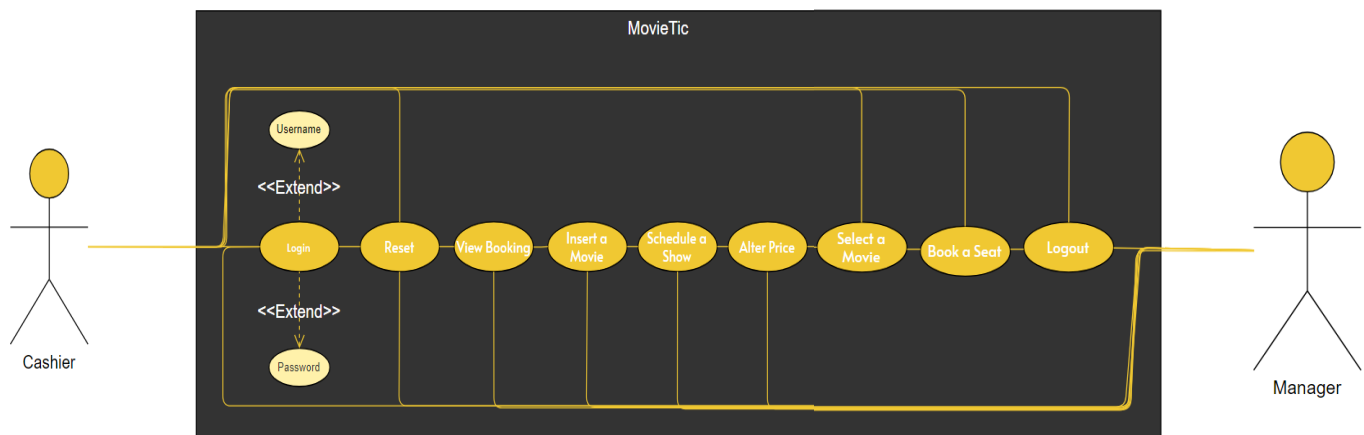


Figure 6.1 : MovieTic Use Case Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

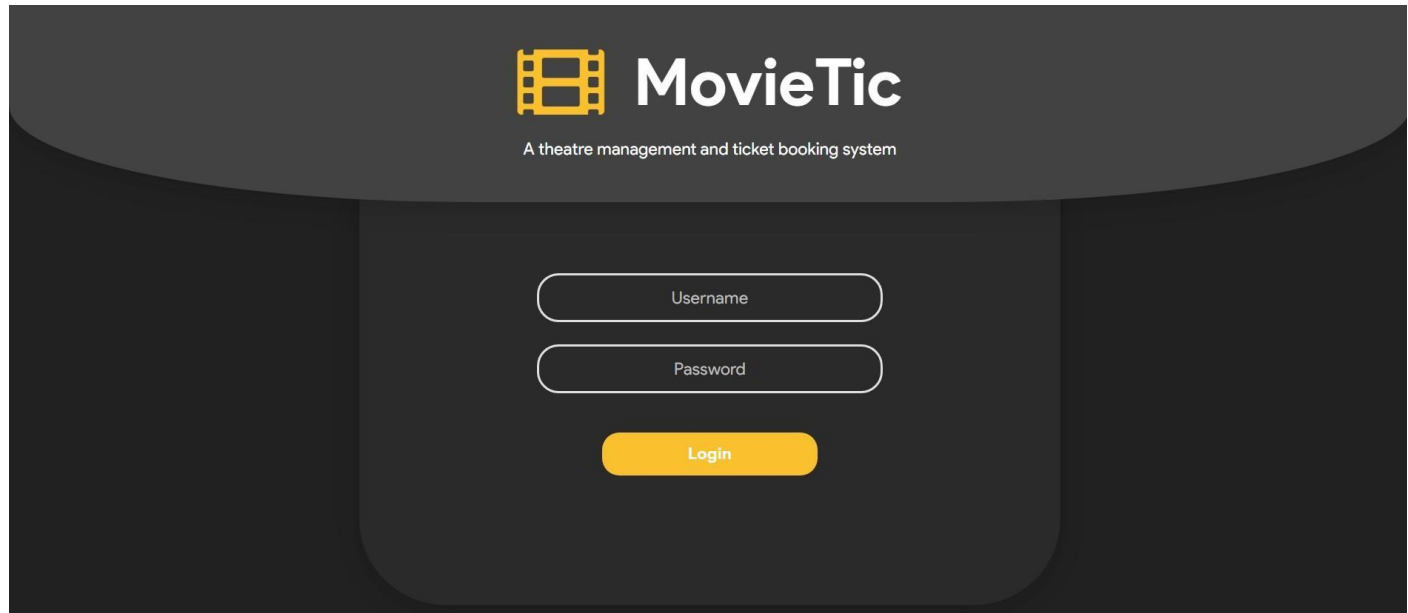


Chapter 7

Output Screens

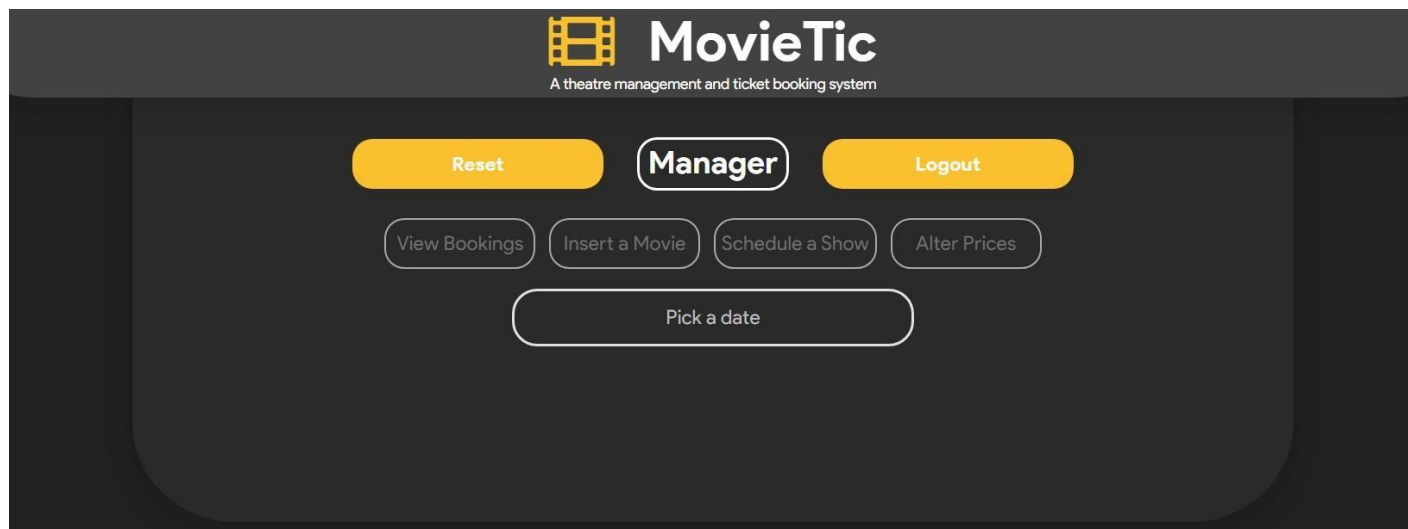
CHAPTER 7

OUTPUT SCREEN



The login page features a dark grey header with the MovieTic logo (a yellow film strip icon) and the text "MovieTic" in white. Below the logo, the tagline "A theatre management and ticket booking system" is displayed in a smaller white font. The main content area is a dark grey rectangle with rounded corners, containing three white input fields: "Username", "Password", and a yellow "Login" button.

Figure 7.1 : Login page



The manager page features a dark grey header with the MovieTic logo (a yellow film strip icon) and the text "MovieTic" in white. Below the logo, the tagline "A theatre management and ticket booking system" is displayed in a smaller white font. The main content area is a dark grey rectangle with rounded corners, containing a row of three yellow buttons: "Reset", "Manager" (highlighted with a white border), and "Logout". Below these buttons are four white input fields: "View Bookings", "Insert a Movie", "Schedule a Show", and "Alter Prices". At the bottom of the main content area is a large white input field labeled "Pick a date".

Figure 7.2 : Manager page

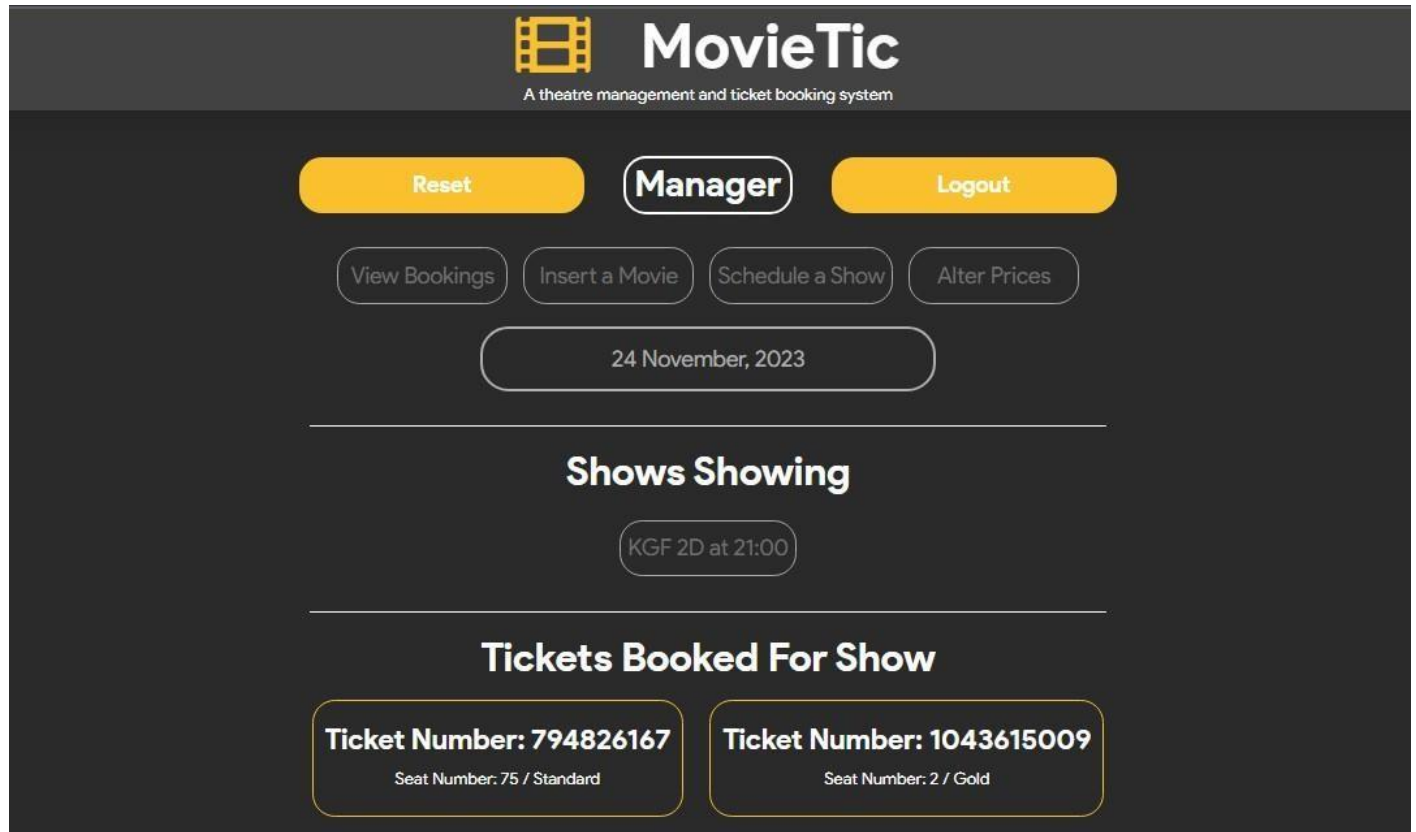


Figure 7.2.1 : Manager page (View Booking)

The screenshot displays the 'MovieTic' manager interface. At the top, there is a header with a filmstrip icon, the title 'MovieTic', and the subtitle 'A theatre management and ticket booking system'. Below the header, there are three main buttons: 'Reset', 'Manager' (which is highlighted), and 'Logout'. Under the 'Manager' section, there are four sub-buttons: 'View Bookings', 'Insert a Movie', 'Schedule a Show' (which is the active button), and 'Alter Prices'. The 'Schedule a Show' section contains two input fields: '24 November, 2023' and '11:00 AM', followed by a 'Submit' button. Below this, there is a section titled 'Movies Available For Showing' which displays a card for the movie 'KGF' with details '2D / 120 mins / Hindi'. The next section is 'Select Movie Type For Show', which has a '2D' button. The final section is 'Halls Available On Given Date And Time', which shows three buttons for 'Hall ID: 1', 'Hall ID: 2', and 'Hall ID: 3'. At the bottom, a message states 'Show Successfully Scheduled' with the 'Show ID: 1609085814'.

MovieTic
A theatre management and ticket booking system

Reset **Manager** **Logout**

View Bookings **Insert a Movie** **Schedule a Show** **Alter Prices**

24 November, 2023

11:00 AM

Submit

Movies Available For Showing

KGF
2D / 120 mins / Hindi

Select Movie Type For Show

2D

Halls Available On Given Date And Time

Hall ID: 1 Hall ID: 2 Hall ID: 3

Show Successfully Scheduled
Show ID: 1609085814

Figure 7.2.2 : Manager page (Schedule a Show)

MovieTic
A theatre management and ticket booking system

Reset **Manager** **Logout**

View Bookings Insert a Movie Schedule a Show Alter Prices

Movie Details

Name

Length

Language

Types Available (2D, 3D, 4DX) separated by space

Premiere Date

Last Date in theatres

Submit

Figure 7.2.3 Manager page (Insert a Movie)

MovieTic
A theatre management and ticket booking system

Reset **Manager** **Logout**

View Bookings Insert a Movie Schedule a Show Alter Prices

Select Price Listing

2D / Sunday Standard: ₹ 550 Gold: ₹ 825.0	3D / Sunday Standard: ₹ 335 Gold: ₹ 502.5	4DX / Sunday Standard: ₹ 495 Gold: ₹ 742.5	2D / Monday Standard: ₹ 210 Gold: ₹ 315.0	3D / Monday Standard: ₹ 295 Gold: ₹ 442.5	4DX / Monday Standard: ₹ 380 Gold: ₹ 570.0	2D / Tuesday Standard: ₹ 210 Gold: ₹ 315.0	3D / Tuesday Standard: ₹ 295 Gold: ₹ 442.5
4DX / Tuesday Standard: ₹ 380 Gold: ₹ 570.0	2D / Wednesday Standard: ₹ 210 Gold: ₹ 315.0	3D / Wednesday Standard: ₹ 295 Gold: ₹ 442.5	4DX / Wednesday Standard: ₹ 380 Gold: ₹ 570.0	2D / Thursday Standard: ₹ 210 Gold: ₹ 315.0	3D / Thursday Standard: ₹ 300 Gold: ₹ 450.0	4DX / Thursday Standard: ₹ 380 Gold: ₹ 570.0	
2D / Friday Standard: ₹ 320 Gold: ₹ 480.0	3D / Friday Standard: ₹ 335 Gold: ₹ 502.5	4DX / Friday Standard: ₹ 495 Gold: ₹ 742.5	2D / Saturday Standard: ₹ 320 Gold: ₹ 480.0	3D / Saturday Standard: ₹ 335 Gold: ₹ 502.5	4DX / Saturday Standard: ₹ 495 Gold: ₹ 742.5		

Figure 7.2.4 : Manager page (Alter Prices)

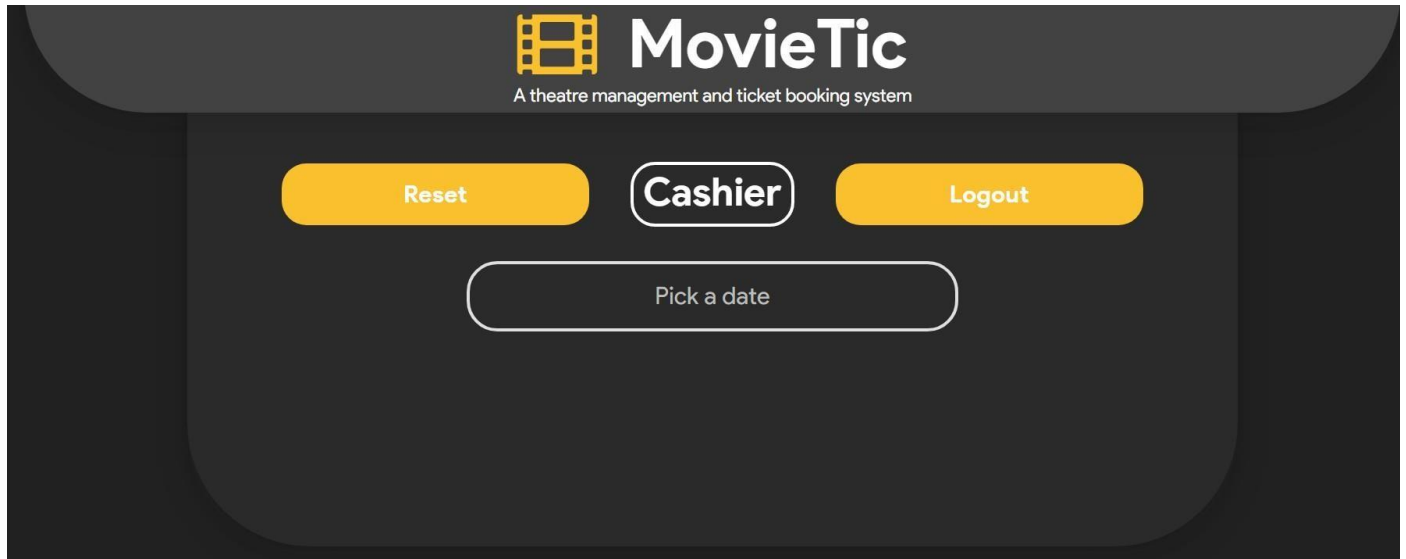


Figure 7.3 : Cashier page

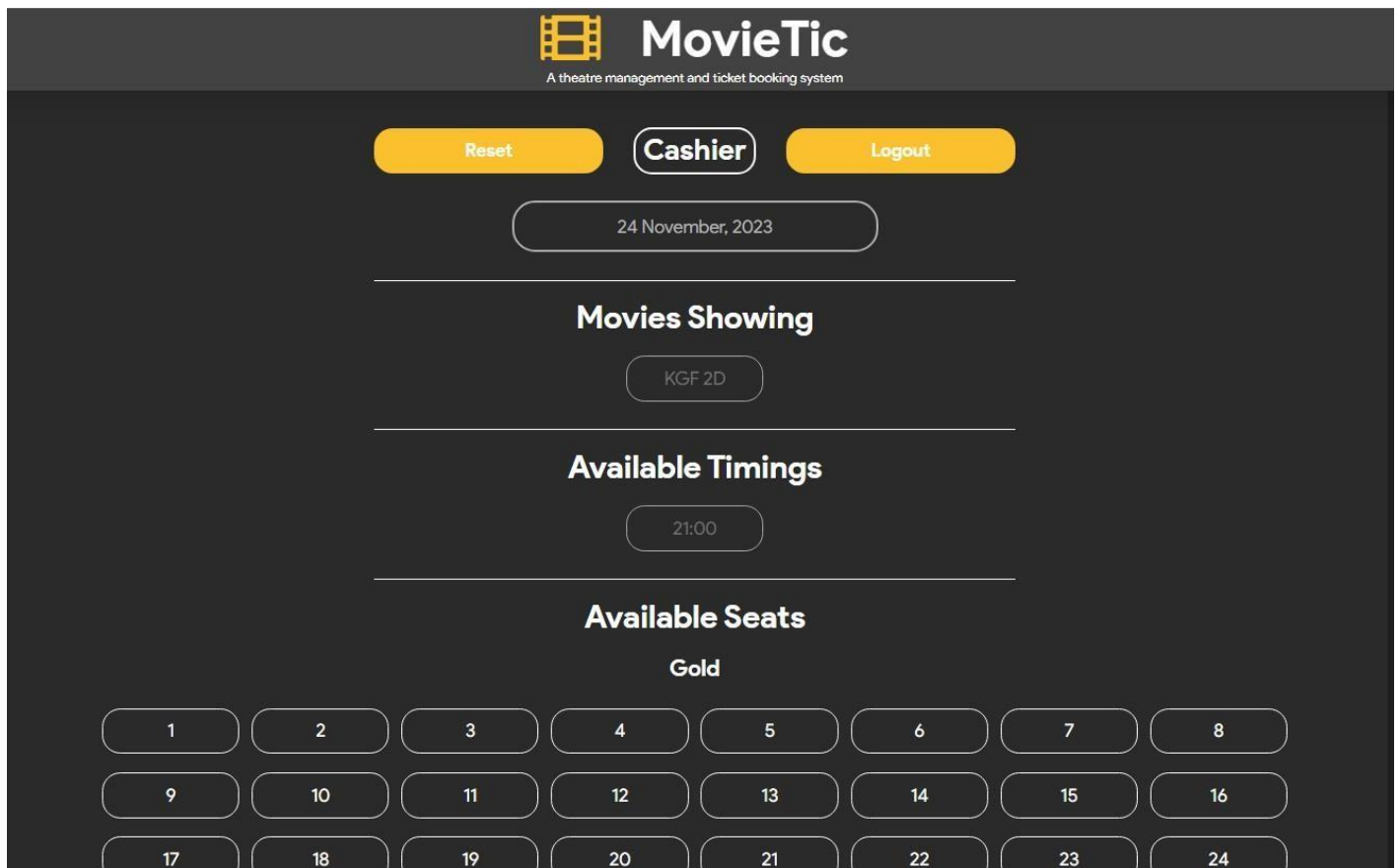


Figure 7.3.1 : Cashier page (Pick a date)

Chapter 8

Conclusion & Future Works

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 CONCLUSION

As per the Requirements of the project, here is implemented a System that enables Theatre staff to fully minimize all aspects of a theatre movie management to an efficient and fast computer system. It allows movies, and shows to be added to the system, and tickets booked for shows. As such, need for bulky ledgers drops sharply, and all relevant is stored and can be printed as needed. It automatically determines prices and deletes old shows, movies and tickets reducing the amount of data to traverse for manager. Thus, we have implemented a fully comprehensive and minimalistic efficient system for use by managers and cashiers without any additional training.

8.2 LIMITATIONS OF THE PROJECT

- This project only considers theatre functions relating to movies and showings. Other functions such as employees, concessions, cleaning schedules, etc. have not been considered in this release.
- Halls cannot be inserted or updated from the front end in any way.

8.3 FUTURE ENCHANEMENTS

The future enhancements that may be brought about in the project may pertain to the limitations of the project. This includes:

- Factoring in additional functionalities relating to the theatre not covered under movies:
 - Concession Stands: Stalls selling food and drinks for movie-goers. This includes items sold, pre-orders etc.
 - Cleaning schedules: Halls need to be cleaned between movies, and theatre needs to be cleaned before opening and after closing.
 - Employees: Keeping track of employees hired, salaries, bonuses, duties, remarks etc.
- Account for expansion: Allow a high-ranking executive (with a new login ID) to add new halls to a location, or add new locations, with manager able to add shows only for their location, etc.

References

- Google (www.google.com)
- ChatGpt (<https://chat.openai.com>)
- YouTube (<https://www.youtube.com>)
- <http://amsul.ca/pickadate.js/date/>
- [tps://teamtreehouse.com/community/nested-loops-in-flask-how-to-iterate-and-make-nested-lists](https://teamtreehouse.com/community/nested-loops-in-flask-how-to-iterate-and-make-nested-lists)
- <https://www.w3resource.com/mysql/date-and-time-functions/date-and-time-functions.php>
- <https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html>
- <https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html>
- <https://getbootstrap.com/docs/4.1/getting-started/introduction/>
- <https://w3resource.com/mysql/date-and-time-functions/mysql-dayname-function.php>
- <https://w3resource.com/mysql/date-and-time-functions/mysql-curdate-function.php>
- <https://w3resource.com/mysql/date-and-time-functions/mysql-datediff-function.php>

About Project

Title of the project	MoiveTic										
Semester	7										
Members	5										
Team Leader	Daksh Raj Singh										
Describe role of every member in the project	<table> <tr> <td>Daksh Raj Singh</td><td>Frontend and Graphics</td></tr> <tr> <td>Dikshant Kumar Bhatiya</td><td>Frontend and Backend</td></tr> <tr> <td>Deepak Kumar</td><td>Frontend ,Backend and Database</td></tr> <tr> <td>Naina Soni</td><td>Database</td></tr> <tr> <td>Shreya Singh</td><td>Frontend</td></tr> </table>	Daksh Raj Singh	Frontend and Graphics	Dikshant Kumar Bhatiya	Frontend and Backend	Deepak Kumar	Frontend ,Backend and Database	Naina Soni	Database	Shreya Singh	Frontend
Daksh Raj Singh	Frontend and Graphics										
Dikshant Kumar Bhatiya	Frontend and Backend										
Deepak Kumar	Frontend ,Backend and Database										
Naina Soni	Database										
Shreya Singh	Frontend										
What is the motivation for selecting this project?	Industry Demand										
Project Type (Desktop Application, Web Application, Mobile App, Web)	Web – Application										

Tools & Technologies:

Programming language used	HTML, CSS, JAVASCRIPT,PYTHON
Compiler used (with version)	VSCODE (1.73.1)
IDE used (with version)	VSCODE
Front End Technologies (with version, wherever Applicable)	HTML5, CSS3 , JAVASCRIPT ES6
Back End Technologies (with version, wherever applicable)	FLASK (2.2.2)
Database used (with version)	MYSQL (5.0.37)

Software Design Coding:

Is prototype of the software developed?	No
SDLC model followed (Waterfall, Agile, Spiral etc.)	Agile
Why above SDLC model is followed?	Everyone worked on different components
Justify that the SDLC model mentioned above is followed in the project.	Project was divided into different components and each component was made by different member.
Software Design approach followed (Functional or Object Oriented)	Object Oriented
In case Object Oriented approach is followed, which of the OOPS principles are covered in design?	Encapsulation, polymorphism
No. of Tiers (example 3-tier)	3-tier
Total no. of front end pages	0
Total no. of tables in database	6
Database is in which Normal Form?	First
Are the entries in database encrypted?	No
Front end validations applied (Yes / No)	Yes
Session management done (in case of web applications)	Yes
Is application browser compatible (in case of web applications)	Yes
Exception handling done (Yes / No)	Yes
Commenting done in code (Yes / No)	Yes
Naming convention followed (Yes / No)	Yes
What difficulties faced during deployment of project?	Difficulty in connecting database to backend, difficulty in applying exception handling
Total no. of Use-cases	1
Give titles of Use-cases	MovieTic Use Case Diagram

Project Requirements

MVC architecture followed (Yes / No)	Yes
If yes, write the name of MVC architecture followed (MVC-1,MVC-2)	MVC – 1
Design Pattern used (Yes / No)	Yes
If yes, write the name of Design Pattern used	MVC
Interface type (CLI / GUI)	GUI
No. of Actors	2
Name of Actors	Cashier,Manager
Total no. of Functional Requirements	8
List few important non Functional Requirements	Performance & Scalability,Portability & Compatibility,Reliability,Maintainability, Availability,Security,Usability

Testing

Which testing is performed ?(Manual or Automation)	Manual
Is Beta testing done for this project?	Yes

Write project narrative covering above mentioned points

We have developed a highly comprehensive and easy to use system for any small Theatre. It is easy to implement and requires no training to use. It provides options for managers and cashiers. It is error-proof and does large amount of work in the background. Thus, the system aids to simplify the processes used by cashier and manager as well as reduce operational costs, the primary concerns of all businesses.

Daksh Raj Singh **0536CS201012**

Signature

Dikshant Kumar Bhatiya **0536CS201014**

Deepak Kumar **0187CS201048**

Project guide

Naina Soni **0187CS201103**

(Prof. Bhavna Soni)

Shreya Singh **0187CS201161**