**DESIGN AND IMPLEMTENTATION OF WEBBASED MOVIE**

**TICKET RESERVATION SYSTEM**

**A PROJECT REPORT SUBMITTED BY**

**ENEH CHIDERA VALERIE**

**U14/NAS/CSC/052**

**A PROJECT REPORT IN THE DEPARTMENT OF COMPUTER SCIENCE**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF A DEGREE OF BACHELOR DEGREE**

**IN**

**COMPUTER SCIENCE**

**GODFREY OKOYE UNIVERSITY,**

**ENUGU STATE.**

**JULY, 2018**

**CERTIFICATION PAGE**

I Eneh Chidera Valerie an undergraduate of the department of Computer Science/Mathematics, Godfrey Okoye University with the registration number U14/NAS/CSC/052 do hereby affirm that the work embodied in this research: Design and Implementation of Online Movie Ticket Reservation System is original and has not been submitted in part or full in any other diploma or degree of this university.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ENEH CHIDERA VALERIE**

**U14/NAS/CSC/052**

**APPROVAL**

This project has been approved to have satisfied the requirements for the award of Bachelor of Science Degree in the department of Computer Science /Mathematics, Godfrey Okoye University, Ugwuomu-Nike, Enugu State.

**SUPERVISOR DATE**

**Dr (Mrs) Monica Agu**

**Dr (Mrs) Monica Agu DATE**

**Head of Department**

**DEDICATION**

This research work is dedicated to Engr. & Dr (Mrs.) .M.C.C Eneh for their support in achieving this stage of my academic pursuit.

**ACKNOLEDGEMENT**

I am grateful to God for the gift good health, strength and ability to complete this project. I wish to acknowledge the efforts of my indefatigable supervisor; Mrs. Monica Agu for her supervision throughout the course of this work.

I want to thank the head of department and all the lecturers of Computer Science whose teaching techniques made an impact in me academic wise.

My profound gratitude goes to all my friends, course mates and well-wishers who contributed in one way or the other to the successful completion of this work

**Abstract**

Movie Ticket Booking System is an online based application that can be accessed through the internet via a mobile device like phone, tablet and laptop etc by anyone at any time. This application is responsive and will help reserve tickets. Users are required to login to the system to book for ticket and make payment via e-payment system options. Watching movies with family and friends in theatres is one of the best medium of entertainment after having a hectic schedule. But all this excitement vanishes after standing in hours in long queues to get tickets booked hence this automated system. The system provides complete information regarding currently running movies on all the screens with details of show timings, available seats. Ticket reservations are done using e-payment system and can be cancelled if need be. This online tickets reservation system is one of the best opportunities for those who cannot afford enough time to get their tickets reserved standing in long queues. People can book tickets online at any time of day or night at their pace.

The application was developed using (SSAD) methodology using the following technologies PHP as the front end and MYSQL at the back end JavaScript for form validations

table of contents

Title Page i

Certification page ii

Approval page iii

Dedication iv

Acknowledgement v

Abstract vi

List of figures ix

List of tables x

**CHAPTER 1: Introduction**

1.0 Background of study 1

1.1 Statement of the problem 2

1.2 Objectives 2

1.3 Significance of the project 2-3

**CHAPTER 2: Literature Review**

2.0 Introduction 4

2.1 Theoretical Background 5

2.2 Review of Related Literature 5-6

2.3 Summary 7

**CHAPTER 3: System Analysis and Design**

3.0 Introduction 8

3.1 Description of the Exiting System 8

3.2 Analysis of the Proposed System 9

3.3 Design of Proposed System 11

3.4 Database Design 11-13

**CHAPTER 4: System Implementation**

4.0 Introduction 14

4.1 Choice of Development Environment 14

4.2 Implementation Architecture 15

4.2.1 Admin Module 16

4.2.2 Transaction Module 17

4.2.3 Maintenance Module 18

4.2.4 Report Module 19

4.2.5 System Flowchart 19-22

4.3 Software Testing 23-25

4.4 Documentation 26

4.4.1 User Manual 26-27

4.4.2 Source Code (Attached in Appendix A)

**CHAPTER 5: Summary and Conclusion**

5.0 Summary 28

5.1 Conclusion 28

5.2 Recommendations 29

**References 30-32**

**Appendices 33-45**

**LIST OF FIGURES**

Fig3.3.1 Use case diagram ………………………………………………. 10

Fig4.3. Admin flowchart…………………………………………………. 20

Fig 4.3.2 Transaction flowchart……………………………………………. 21

Fig 4.3.3 Maintenance flowchart…………………………………………… 22

Fig 4.4.1 System Testing Process…………………………………………… 23

Fig.4.4.2 Screenshot of login Interface……………………………………… 24

Fig 4.4.3 Screenshot of Movie Display……………………………………… 25

Fif4.4.4 Screenshot of Movie Timings……………………………………… 25

**LIST OF TABLES**

Table3.4.1 Credit card information……………………………………. 11

Table3.4.2 User Information……………………………………………12

Table3.4.3 Movie table…………………………………………………12

Table3.4.4 Booking table……………………………………………… 13

Table3.4.5 login Interface………………………………………………13

**CHAPTER 1**

**INTRODUCTION**

**1.0 Background of Study**

Online cinema booking system is basically made for providing the movie ticket anywhere and anytime and to get information about the movies at convenient. The user can easily be able to know about the movies released and then make a choice.

It is a web-based system. The customers can buy ticket online and cancel the seat at a suitable time (2 days before the show to 1hour before the show). To enhance the refund function, all the customers have to register to become a Member before buying the ticket. Staff can use the system to insert and delete data (example film description, time table) which will update the webpage. Also staff can check the statistic information from the system. The system is to provide an alternate and convenient way for a customer to buy movie tickets. It is an automatic system. After the data has been fed into the database, the staff does not need to do anything with the order once it is received through the system.

The name “movie” originates from the fact that photographic film (also called film stock) has historically been the medium for recording and displaying motion pictures [1]. Many other terms exist for an individual motion picture, including picture, picture show, moving picture, photoplay and flick. Today, it is believed that the application of computer technology in any activity would go a long way in making that activity much easier.

This statement remains a theory until proven otherwise by the implementation of online movie ticket reservation system. It also believed that the benefit and advantages of using computer out weight that of the manual methods beyond doubt[2] by the researcher through the implementation of the new system.

**1.1 Statement of the Problem**

* Inability to check and track customers that sneak into the cinema hall with fake seats number printout
* Inability to book seat/Ticket by customers at their convenient
* Inability of the customer to create its own account (member registration) and login to book for ticket
* Lack of secure e-payment option for customer

**1.2 Aim and objective of the Project:**

The objective of this project is to develop an Online Movie Ticket Reservation system with the following objectives:

* In order to establish an efficient way to promote the film on the internet
* To provide an alternate and convenient way for a customers to buy movie tickets and make payment at their pace
* Minimize the number of staff at the ticket box
* Customers can buy ticket online and cancel the seat at a suitable time (2 days before the show to 1hour before the show).

**1.3 Significance of the Project**

The significant of the study are:

* To provide an easy and user friendly movie ticket system that is mobile responsive
* To obtain statistic information from the booking record
* Provide a system where Users can check film data by clicking on a certain film on main page(e.g. The cinema which will show this films).
* To provide a secure e-payment system with payment options for customer.

**CHAPTER 2**

**LITERATURE REVIEW**

2.0 **Introduction**

The system is developed using PHP, MYSQL, JavaScript and Codeigniter for making Content Management System (CMS). The theoretical background covered the various technologies use in this system and the related work on this project.

**2.1 Theoretical Background**

The Technologies used in this work are listed below

1. Hypertext pre-processor (PHP)
2. JavaScript
3. MYSQL database
4. Cascade Style Sheet

**PHP:** PHP is a scripting language, and a tool for making dynamic and interactive Web pages**.** It can be embedded into the HTML (hypertext mark-up language).

The interface of the forms was develop using HTML (HTML Form) and control by PHP scripting language, the script collect the data from the form and move it to the MYSQL database where is been store and also displaying and retrieving of the data from the database when required

The above HTML code was used to build the form and CSS is use to add style to the form

**JAVASCRIPT:**

JavaScript is a client-side script, used to control a web page at the client side once it has downloaded. It is used to make webpages interactive. It supports event-driven, functional, and imperative programming style.

**MYSQL:**

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". The MySQL Database powers the most demanding Web, E-commerce and Online Transaction Processing (OLTP) applications. It is a fully integrated transaction-safe, ACID compliant database with full commit, rollback, crash recovery and row level locking capabilities. MySQL delivers the ease of use, scalability, and performance that has made MySQL the world's most popular open source database.

MySQL is the database construct that enables PHP and Apache to work together to access and display data in a readable format to a browser. It is a Structured Query Language server designed for heavy loads and processing of complex queries. As a relational database system, MySQL allows many different tables to be joined together for maximum efficiency and speed.

**2.2 Review of Relevant Literature**

In 1979 Redifon Computers was a part of the UK Rediffusion group of companies and in March 1980 they went on to launch Redifon's Office Revolution, which allowed consumers, agents, distributors, suppliers and service companies to be connected on-line to the corporate systems and allow business transactions to be completed electronically in real-time[3].In 2009 they designed, installed, maintained and supported many online movie booking systems, using videotex technology. These systems which also provided voice response and handprint processing pre-date the Internet and the World Wide Web, the [IBM PC](http://en.wikipedia.org/wiki/IBM_Personal_Computer), and [Microsoft](http://en.wikipedia.org/wiki/Microsoft) OS, and were installed mainly in the UK by large corporations. Hillenbrand [8] cautions against a computer and software purchase for small operations, suggesting that a dedicated telephone answering machine be utilized. Reid (1983) observed that computers were replacing the traditional paper and pencil systems, allowing several telephones to simultaneous access the same seating plan, thus avoiding a patron queue and improving on customer services, compared to a manual system. [10] proposes that theatres should match the conveniences provided by the numerous other service providers such as toll free numbers, credit card processing, internet access to products and services, mail order, all of which today’s consumers have come to expect. In addition to knowing about the performance, the modern day box office treasurer is expected to be able to give patrons advice on restaurants and parking, as well as operate it until well after normal working hours, so as to provide convenient opening hours to the public[11].

Tickets may be paid for in advance, and either mailed to the patron, or kept at

the box office to be collected when the patron arrives at the theatre for the

performance [12]. The box office can make use of voice mail to reserve

tickets for subsequent purchase, but should sell any unclaimed tickets fifteen

minutes prior to the beginning of the performance. At least 10% of tickets

reserved for a performance are not collected [13], so reserving them while there is demand at the door would be unsound economics.

**2.3 Summary**

This online movie booking system was achieve by codeigniter , PHP and MYSQL this system have a live chat on the footer for customer interaction and a secure payment options for easy of ticket and seat reservation payment

**CHAPTER 3**

**SYSTEM ANALYSIS AND DESIGN**

**3.0 Introduction**

Methodology involves a process whereby the existing or current system is studied to identify the information requirements. It is used to refer to a specific series of steps or procedures which governs the analysis and design of a particular project. It also includes the techniques and methods which are used to collect and analyse information. To achieve all these stated above, Methodology (SSADM) is a systems approach to the analysis and design of information systems. SSADM method involves the application of a sequence of analysis, an internationally accepted software engineering model was use which is Structured System Analysis and Design Methodology (SSADM). Structured System Analysis and design tasks concerned with analysis of the current system, logical data design, logical process design etc.

* + 1. **Description of the Existing System**

An increasing number of customers standing in long queues outside the theaters which, theater owners "Customer Delight" faced the challenge of providing. Customers strict order and without delay and inconvenience of standing in long queues to pay for their tickets wanted an easy way. Their loyalty program "theater" to the administration, as well as partners to help maximize their presence at the multiplex other promotional and subscription services provides a mechanism is needed. Cinemas phone book was launched, but un-sold seats to customers, leading to frequent "no shows" were, because it was not feasible to prove the profitability of the business affected.

**3.2 Analysis of the Proposed System**

The proposed system is more reliable, entertaining and easier than the present system. The solution targets those users who do not have spare time to stand in queue for booking tickets. This proposes an easy way of ordering and paying for the tickets without any delays and inconvenience. It enables users book tickets without physically present at the theater booking center.

Customers who book tickets on their mobile phones will receive an instant message (m-ticket). By splashing the M- ticket at the counter of the multiplex, the client can receive physical tickets. No longer issue tickets to staff at the multiplex complex manual ticket availability and tracking system is needed. Ticket information can be accessed at any time for verification, which is safely stored in a database.

**3.3 Design of the Proposed System**

**Use case Diagram** Use Case diagram for Online Movie Ticket Booking System is shown below. The various participants of the same are detailed below:-  
**Actors**: - Registered User, Visitor, and Admin  
The corresponding use cases for these actors are:

* **Registered User**: Choose Movie, Select Time And Venue, Select Seat, Select Ticket Type, Child, Adult, Make Payment By Credit/Debit Card, Confirm Transaction, Cancel Ticket, Get Refund
* **Visitor** : Get Registered, Check Movie Data
* **Admin** : Login, Insert Movie Records, Update Movie Records, Delete Movie Records

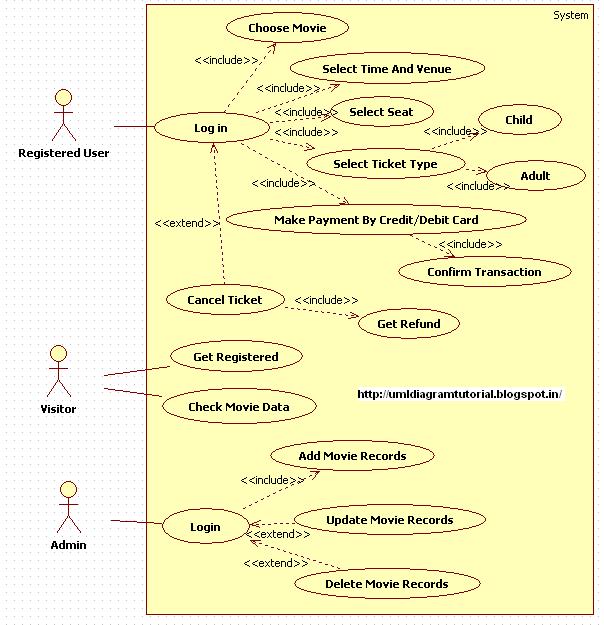


Fig3.3.1 showing the use case diagram of an online booking system.

**3.4 Database Design**

In this section, the basic structure of the tables composing the database for the project are shown along with information about primary and foreign keys.

**Credit Card information**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/NO** | **ATTRIBUTES** | **TYPE/WIDTH** | **DESCRIPTION** |
| 1 | Number of credit card | Number (14) | Users credit card number |
| 2 | Ticket | Number (12) |  |
| 3 | Name | Varchar (20) | Registered Users full name |
| 4 | User id | Varchar(20) | Customers login user id |
| 5 | Exp date | Varchar (20) | Expiration date of the credit card |
| 6 | Movie | Varchar (20) | Name of the movie to be booked |

table3.4.1 This table is the credit card table the information of the customer credit card are store in this table , the customer provide the credit card details during payment.

**User Information:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/NO** | **ATTRIBUTES** | **DATATYPE/ WIDTH** | **DESCRIPTION** |
| 1 | First Name | Varchar (20) | First name of the user |
| 2 | Last Name | Varchar (20) | Last name of the user |
| 3 | User id | Varchar (20) | User id |
| 4 | Password | Varchar (10) | Security password |
| 5 | Contact | Numeric (11) | Phone number of user |
| 6 | Hint Question | Varchar (10) | Hint question to be asked |
| 7 | Email id | Varchar (20) | Email of the user |

Table3.4.2 **This table stores the details of the customer in the database table**

**Booking Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/NO** | **ATTRIBUTES** | **DATA TYPE/WIDTH** | **DESCRIPTION** |
| 1 | Movie Id | Varchar (20) | Movie id to be booked |
| 2 | Name of movie | Varchar (20) | Name of movie |
| 3 | Show Time | Varchar(20) | Timings of the movie |
| 4 | Screen id | Varchar (20) | Screen id of where the movie is to be shown |

Table 3.4.3 This is the movie booking table it store the details of the movie booked

**Movie Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/NO** | **ATTRIBUTES** | **DATA TYPE/WIDTH** | **DESCRIPTION** |
| 1 | Movie id | Varchar (10) | Primary key |
| 2 | Name of movie | Varchar (20) | Name of the movie |
| 3 | Details | Varchar (20) | Details of the movie |

Table3.4.4 This is the movie booking table it store the details of the movie to be shown

**Login Interface**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/NO** | **ATTRIBUTES** | **DATA TYPE/WIDTH** | **DECSRIPTION** | |
| 1 | Username | Varchar (20) | User id | |
| **2** | Password | Varchar (20) | | Unique id |

**T**able 3.4.5

**CHAPTER 4**

**SYSTEM IMPLEMENTATION**

**4.0 Introduction**

System implementation is a collection of inter-dependent physical devices together with their programming which provides the functionality and performance for which the system was designed. It covers all the activities necessary to set the system that has been analysed and designed to be fully functional to the users.

**4.1 Choice of Development Environment**

The development tools are the necessary requirement tools used during the design to enable us achieve the system design. The listed packages were used because of their features, accessibility and also because it is more effectivity.

1. MySQL database application

2. PHP scripting Language

3. HTML language and JavaScript

4. Notepad plus

5. CodeIgniter

**MySQL Database Server:** MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python/JSP". The MySQL Database powers the most demanding Web, E-commerce and Online Transaction Processing (OLTP) applications. It is a fully integrated transaction-safe, ACID compliant database with full commit, rollback, crash recovery and row level locking capabilities. MySQL delivers the ease of use, scalability, and performance that has made MySQL the world's most popular open source database.

**PHP:** PHP is a server-side scripting language that allows your Web site to be truly dynamic. PHP stands for PHP: Hypertext Preprocessor Its flexibility and relatively small learning curve (especially for programmers who have a background in C, Java, or Perl) makes it one of the most popular scripting languages around. PHP’s popularity continues to increase as businesses, and individuals everywhere embrace it as an alternative to Microsoft’s ASP language and realize that PHP’s benefits most certainly outweigh the costs.

**HTML Language:**

Hypertext Mark-up Language (HTML), the standard text-formatting language for documents on the interconnected computing network known as the World Wide Web. HTML documents are text files that contain two parts: content that is meant to be rendered on a computer screen; and mark-upor tags, encoded information that directs the text format on the screen and is generally hidden from the user.

**JavaScript** is a programming language that adds interactivity to your website for example: games, responses when buttons are pressed or data entered in forms, dynamic styling, animation etc.

**4.2 Implementation Architecture**

The new system is designed to be put into efficient use here, we will look into the various technical aspects that influenced the successful implementation of this system and determine the effective operation of the system. System implementation follows the approval of the system proposal and its objectives, thus it is to arrive at a satisfactory, implemented, completed, and function evaluated automated system.

**4.2.1 Admin Module**

The admin module makes the admin user to login with admin username and

password. The admin is the main administrator and controller of the whole system. He also maintains the user registration and the logins. The admin needs to manage the show timings.

Admin can add a movie, update the information regarding an old movie and even delete a movie from the theatre. The admin can even add a theatre and delete a theatre.

The admin makes the user available with search by movie and by theatre. The admin also maintains the list of movies.

The first page in my module is the main page of the project which contain a menu

consisting of options like home, now showing, cinema, coming soon, contact, advertise with us , and ticket info. This page also contains the login option for the users to login into their account and also to register an account with us.

The next page is the user login page asking them to enter their username and

password. If the password and the username exists in our database the user enters into his

main page. A registration page is also available for the new users to register.

In cinemas page the user can select the movie by using search by theatre. In now

showing page the user can search by movie.

Admin has a separate admin page where the admin can login using a username and

password. In admin the user can add a theatre, delete a theatre, add a movie and also delete a movie. In add theatre the admin can add a theatre by entering the name , type of theatre, capacity, number of screens, theatre manager, theatre address of a theatre. In delete theatre the admin can directly enter the theatre name and delete that particular theatre. In add movie the admin can add a movie by entering the name, rating, review, genre, director, producer and music director of a movie. In delete movie the admin can directly enter the movie name and delete the movie.

**4.2.3 Transaction Module**

The transaction module takes care about the issue of the tickets it is used to enter the details like name, cinema name, no of tickets, timings and dates, types of class, phone no and total price for ticket.

Next we enter submit and enter into payment page. In this payment page we enter details like CVV no and card number. During the payment after entering details we enter pay button then it run and display transaction is successful. This will be a safe payment procedure.

During the tickets booking the user can also add some refreshments to the payments. In this cancellation they can also cancel the tickets after ticket booking.

check out is used to verify whether details entered for ticket booking is correct or not if not entered correctly we can rectify the errors in this check out.

Points to customer is used to add points to customer for frequently used theatres .if the customers frequently enter to theatres they can gain more points.

The Validation is automatic checks make ensure that any data entered into system is sensible. This validation does not make sure that data has been entered correctly. It only ensures that data is sensible. In this module we provide favourites when we click the favourites next to theatre name on any show times page to mark it as favourites and then come back here to see show time at your favourite’s theatres.

**4.2.4 Maintenance Module**

In this maintenance module we have creation of users in which we enter details like name, account no, seat no's, amount. In this module we show the list of movies and maintaining schedule timings for each movie. Points to be added to the for frequently visiting customers, points for the available customers will be given some offers so that they will be attracted towards these cinemas. The Timings, upcoming movies should be updated time to time so that customers will be

given correct information about the movies running at present and the show timings. The Publicity material should be maintained such as the contact details and frequently updated. In this module we make the user available to contact us regarding any problem and also to advertise with us.

The Validation is an automatic check make ensure that any data entered into system is sensible. This validation does not make sure that data has been entered correctly. It only ensures that data is sensible.

At the end in this module we provide report for different customers. Here we provide different advertising options for marketing.

**4.2.5 Reports Module**

In the reports module we will maintain reports regarding many things like list of nil sale items, list of daily sales, list of frequently used theatre, list of frequently visited customers and others. This helps us to analyze the profits and also improve our infrastructure and services if necessary. In this module the user can also print his ticket. This will be shown at the theatre for getting the ticket at time of show.

**4.3 Flowchart**

A flowchart type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem.

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams

**4.3.1 Admin Flowchart**

Admin

USER

NO

NO

If login is correct

if correct

YES

YES

Select movie

Admin page

If movie is available

NO

Yes

Book movie

**Fig4.3.1**

**4.3.2 Transaction Module**

Home page

Issue ticket

is movie availabe

NO

NO

YES

want to proceed ?

YES

**Fig 4.3.2**

4.3.3 **Maintenance Module**

**Fig 4.3.3**

YES

NO

is login correct ?

update movie

Delete Movie

Add movie

Manage user account

Admin page

login

4.4 Software Testing

Testing is the last stage in the software development and it presents an interesting anomaly for the software engineer where he attempts to build software from an abstract concept to a tangible product. During testing, the engineer creates series of test cases to discard preconceived notions of the “correctness” of software just developed and overcome a conflict of interest that occurs when errors are uncovered. As a secondary benefit, testing demonstrates that the software functions appear to be working according to specification, that behavioral and performance requirements appear to have been met. In addition, data collected as testing is conducted provide a good indication of software reliability and quality as a whole.

Testing the software follows a certain process as shown below:



**Fig 4.4.1 System Testing Process**

**Unit Test**

Each unit of the new system was tested (test run) individually alongside with the old system in other to identify areas of further enhancement and development.

**System Test**

The entire system was as well tested (test run) in general alongside with the old system in other to identify areas of further enhancement and development.

Screenshot of the login interface.



Fig.4.4.2 Here the user login system for movie booking

Screenshot of Movie Display



Fig 4.4.3 The upcoming page contains a list of movie image with data list by clicking the image users can view all the information related to that particular movie.

Screenshot displaying timings and ticket availability



Fig 4.4.4 Here user can book ticket and also give information about

their credit card like card no, card type exp. Date etc.

**4.5 Documentation**

In order for the proposed system to be used on any computer system it takes the following:

i. Boot the system.

ii. install wampserver on the system

iii. Click on the wampserver icon on desktop.

iv. Click on the wampserver tray icon on the task bar

v. Select www root directory and processed to upload the files

vi. Upload your web files into the directory

vii. Open any browser on your system (Microsoft internet Explorer, Mozilla Firefox,Netscape Navigator, Opera, Flock, Safari etc.)

viix. Type <http://127.0.0.1/movie_booking_system> on the address bar and press the return key or enter key.

vix.Then the website will come up

**4.6 User Manual**

The steps to use the proposed system are as follows:

i. On the address bar of any browser type <http://127.0.0.1/movie_booking_system> ii. After view the movie catalog you can book for seat reservation and adding to the cart at checkout you will be prompted to supply username and password this verifies that you are a registered user if not the customer will be prompt to register

iii. If the username and password supplied are correct as that of a user you are prompted with the user dashboard page with the list of available task /action which you can do or based on choice.

iv. The username and password are in three formats as an administrator, as a Staff and as well as a user/customer.

V. As a Staff you are to type http://127.0.0.1/vehicle\_show\_room/ staff.php on the address bar.

vi. As a staff you are prompted with the staff page where the staff can manager products.

vii. As an Administrator you type <http://127.0.0.1/vehicle_show_room>/admin.php on the address bar.

viii. As an administrator you are prompted with the administrator page that is the back end of the system.

**CHAPTER 5**

**SUMMARY, CONCLUSIONS AND RECOMMENDATIONS.**

**5.1 Summary**

Cinema being the modern society social, economic and cultural influence is a serious incident, the most popular out-of-Home is one of the cultural activities.. Through this project a multiplexes offer a comprehensive solution for ticket booking. Theater management system, easy to understand, easy to use and fast for customers that provides point-and-click simplicity of the service is an online ticket selling software. The intuitive visual interface refund, exchange, and for both users and administrators quick and easy reporting, day-to-day aspects of selling makes. Theater management saved in a database of all the back-end functionalities movie details, ticket rates, and show time, customer information and sales history, such as, etc theater administrator intelligent counter reports daily, weekly, monthly reports and film-like details of the report Reports that manages controls

**5.2 Conclusions**

Nowadays, the traditional book cinema tickets dying methods. This technology has dominated human life, where is the new era. Software and technical equipment, exceptions are reduced and even eliminated. Moreover, for every part of their lives easy, fast and secure way to make the choice. This project is a cinema ticket booking system is designed to meet the requirements. It has been developed in PHP and database keeping in mind the specifications of the system has been created in My SQL server. Our project: ticket system with the cinema; Cinema companies can meet customer comfort. Cinema manager, employee, and customer relations to complete the process of ticket satisfied a good communication. With this platform developed, to avoid wasting time, reduce misunderstandings, easy data flow, customer happiness, and are expected to provide less difficult task..

**5.3 Recommendations**

It is obvious that this theatre booking system is far from being fully functional.

One of the fundamental steps towards this goal is to have all the database tables

linked by the appropriate relationship, which in turn requires that all tables be

normalised. A substantial understanding of database table design principles,

namely *normal forms* and *functional dependency* would be required, and would

most certainly result in the re-structuring of the tables. Setting up a functioning Access web page would also require some effort. Once the database can be modified from a *page*, this can simply be posted on the theatre’s web site, allowing online users to book their seats for the chosen performance.

**REFERENCES**

[1] *Perspectives on Microsoft Office 2017*,2nd edn, Thomson Learning, Inc. USA.

[2] Carmack Celentano, S & Marshall, K 2015, *Theatre Management*, Players

Press, Inc., USA.

[3] Cavanaugh, J 2014, *Organization and Management of the Non-Professional*

*Theatre*, Richards Rosen Press, Inc., New York.

[4] Cohen, GP 2014, *The Community Theatre Handbook*, Heinemann, Portsmouth,

NH, USA.

[5] Collins, T 2014, *How Theatre Managers Manage*, The Scarecrow Press, Inc,

Lanham, Maryland, USA.

[6] Gifford, D & Spector, A 2015, ‘The TWA Reservation System’,

*Communications of the ACM*, vol. 7, no. 7, p. 654.

[7] Grippo, C 2013, *The Stage Producer’s Business and Legal Guide*, Allworth

Press, New York.

[8] Hillenbrand, M 2012, *Produce Your Play Without a Producer*, Smith and

Kraus, Inc, Hanover, NH, USA.

[9] Kroenke, DM 2016, *Database Processing Fundamentals, Design and*

*Implementation*, Pearson Education, Inc, New Jersey.

[10] Langley, S 2017, *Theatre Management in America Principle and Practice*,

Drama Book Publishers, New York.

[11] Mannino, MV 2011, *Database Application Development and Design*, McGraw-Hill Higher Education, New York.

.

[12] Powell, G 2016, *Beginning Database Design*, Wiley Publishing, Inc.,

Indianapolis, Indiana.

[13] Pick, J 2013, *The Theatre Industry*, Comedia Publishing Group, UK.

Raymond, C 2013, *Essential Theatre: The Successful Management of Theatres*

*and Venues with Present the Performing Arts*, Arts Council of England,

London.

[14] Reid, F 2015, *Theatre administration*, A&C Black (Publishers) Ltd, London.

Schneider, RE & Ford, MJ 2015, *The Well-Run Theatre*, Drama Book

Publishers, New York.

[15] Sweeting, E 2013, *Theatre Administration*, Sir Isaac Pitman and Sons Ltd,

London.

[16] Elmasri and Navathe, “Fundamentals of Database Systems”, 3/e, Addison - Wesley, 2017.

[17] http://blogs.nasscom.in/rail-budget-2013-what-does-it-mean-for-the-indian-it-industry.

[18] A Silberschaltz, H.F. Korth, and S sudarshan, “Database System Concepts”, 3/e, Tata Mcgraw Hill,2016.

[19] Thomas M. Connolly, Carolyn E. Begg, “Database Systems & Practical Approach to Design Implementation and Management”, 4/e, Addison – Wesley, 2015

**Appendix A**

Admin module

<?php

session\_start();

?>

<?php

$con = mysql\_connect('localhost', 'root', '');

if (!$con)

{

die('Could not connect: ' . mysql\_error());

}

mysql\_select\_db("movie\_booking", $con);

?>

<?

$user = $\_POST['user'];

?>

<?

$sql = "select \* from users\_tbl where username='$user'";

$result= mysql\_query($sql);

$num = mysql\_numrows($result);

if( $num==0)

{

echo "No such user exist";

}

else

{

$sql = "Update users\_tbl set userlevel='9' where username='$user'";

$result = mysql\_query($sql);

if($result)

{

echo "Selected user was made admin<br>";

}

}

?>

<p>Back To <a href="admin.php">Admin Centre</a>

</p>

<p><a href="logout.php">Logout</a></p>

**User Login Page**

<?php session\_start();

session\_destroy(); ?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Book Ur Show</title>

<style type="text/css">

html, body {height:100%; margin:0; padding:0;}

#page-background {position:fixed; top:0; left:0; width:100%; height:100%;}

#content {position:relative; z-index:1; padding:10px;}

#

</style>

</head>

<body>

<div id="page-background"><img src="images/main%20baclground.jpg" width="100%" height="100%" alt="Smile"></div>

<center>

<div class="container" style="width:800px" id="content">

<div class="header"><img src="images/logo.png" width="177" height="61" longdesc="main.php" /> <!-- end .header --></div>

<center>

<div class="content" style="background-image:url(); height:427px; color: #FFF;vertical-align:middle">

<table width="300" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="">

<tr>

<form name="form1" method="post" action="checklogin.php">

<td>

<table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="">

<tr>

<td colspan="3"><strong>Member Login </strong></td>

</tr>

<tr>

<td width="78">Username</td>

<td width="6">:</td>

<td width="294"><input name="myusername" type="text" id="myusername"></td>

</tr>

<tr>

<td>Password</td>

<td>:</td>

<td><input name="mypassword" type="password" id="mypassword"></td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td><input type="submit" name="Submit" value="Login"></td>

</tr>

</table>

</td>

</form>

</tr>

</table>

<table width="300" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="">

<tr>

<form name="form2" method="post" action="signup.php">

<td>

<table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="">

<tr>

<td colspan="3"><p>&nbsp;</p>

<p><strong> Sign Up Here</strong></p></td>

</tr>

<tr>

<td width="78">Username</td>

<td width="6">:</td>

<td width="294"><input name="myusername" type="text" id="myusername"></td>

</tr>

<tr>

<td>Enter Password</td>

<td>:</td>

<td><input name="mypassword" type="password" id="mypassword"></td>

</tr>

<tr>

<td>Confirm Password</td>

<td>:</td>

<td><input name="mypassword2" type="password" id="mypassword2"></td>

</tr>

<tr>

<td>Email id</td>

<td>:</td>

<td><input name="myemail" type="text" id="myemail"></td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td><input type="submit" name="Submit" value="Sign Up"></td>

</tr>

</table>

</td>

</form>

</tr>

</table>

</div>

</center>

</center>

</body>

</html>

**Movie schedule**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<?

session\_start();

$city= $\_POST['city'];

$movie= $\_POST['movie'];

$date = $\_POST['date'];

$city = stripslashes($city);

$movie = stripslashes($movie);

$date = stripslashes($date);

session\_register("city");

session\_register("movie");

session\_register("date");

?>

<?php

//$q=$\_GET["q"];

$con = mysql\_connect('localhost', 'root', '');

if (!$con)

{

die('Could not connect: ' . mysql\_error());

}

mysql\_select\_db("movie\_booking", $con);

?>

<title>Book Ur Show</title>

<style type="text/css">

a:link {

color:#ffffff;

text-decoration: underline;

}

a:visited {

color: #ffffff;

text-decoration: underline;

}

html, body {height:100%; margin:0; padding:0;}

#page-background {position:fixed; top:0; left:0; width:100%; height:100%;}

#content {position:relative; z-index:1; padding:10px;}

</style>

</head>

<body>

<div id="page-background"><img src="images/main%20baclground.jpg" width="100%" height="100%" alt="Smile"></div>

<center>

<div class="container" style="width:800px" id="content">

<div class="header"><img src="images/logo.png" width="177" height="61" longdesc="main.php" /> <!-- end .header --></div>

<center>

<div class="content" style="background-image:url(); height:427px; color: #FFF;">

<p align="right"><?php $username = $\_SESSION['myusername'];

$sql= "select \* from users\_tbl where username='$username' and userlevel='9'";

$result = mysql\_query($sql);

if($row = mysql\_fetch\_array($result))

{

echo "[<a href=\"admin.php\">Admin Center</a>]";

}

?> [<a href="first.php">Main Page</a>] [<a href="logout.php">Logout</a>]</p><p align="left"><?php

$username = $\_SESSION['myusername'];

echo "Welcome $username";

?></p>

<form name="form1" action="book.php" method="post">

<table width="200" border="0">

<tr>

<td>City</td>

<td><input name="city" type="text" id="city" readonly="true" style="background-color:#000; color:#FFF" value="<? $sql="Select \* from city where city\_id='$city'";$sqlresult=mysql\_query($sql);$row = mysql\_fetch\_array($sqlresult);$cityname=$row['city\_name'];echo $cityname;?>" /></td>

</tr>

<tr>

<td>Movie</td>

<td><input name="movie" type="text" id="movie" readonly="true" style="background-color:#000; color:#FFF" value="<? $sql="Select \* from movie where movie\_id='$movie'" ;$sqlresult=mysql\_query($sql);$row = mysql\_fetch\_array($sqlresult);$moviename=$row['movie\_name'];echo $moviename;?>" /> </td>

</tr>

<tr>

<td>Date</td>

<td><input name="date" type="text" id="date" readonly="true" style="background-color:#000; color:#FFF" value="<? $sql="Select \* from movie where date='$date' and movie\_id='$movie' and city\_id='$city'";$sqlresult=mysql\_query($sql);$row = mysql\_fetch\_array($sqlresult);$date2=$row['date'];echo $date2;?>" /></td>

</tr>

</table>

<?php

echo "<br><br>";

$sql = "Select theatre\_id,showtiming from movie where movie\_id='$movie' and city\_id='$city' and date='$date'";

$result = mysql\_query($sql);

echo "<table>";

echo "<tr>

<td width=\"100px\">Theatre</td>

<td width=\"100px\">Show Timing</td>

<td width=\"100px\">Book Ticket</td></b>

</tr>";

while($row = mysql\_fetch\_array($result))

{

echo "<form name=\"form1\" action=\"book.php\" method=\"post\">";

$sql2 = "Select theatre\_name from theatre where theatre\_id=".$row['theatre\_id']."";

$result2 = mysql\_query($sql2);

$row2 = mysql\_fetch\_array($result2);

$tname = $row2['theatre\_name'];

$stime = $row['showtiming'];

echo "<tr>

<td><input name=\"tname\" type=\"text\" id=\"tname\" readonly=\"true\" style=\"background-color:#000; color:#FFF\" value='$tname'/></td>

<td><input name=\"stime\" type=\"text\" id=\"stime\" readonly=\"true\" style=\"background-color:#000; color:#FFF\" value='$stime'/></td>

<td align=\"center\"><input name=\"book\" type=\"submit\" value=\"Book\" /></td>

</tr>";

echo "</form>";

}

echo "</table>";

?>

</form>

</div>

</center>

</body>

</html>

**Logout Code**

<? Php session\_start();

if(!$\_SESSION['myusername']){

header("location:main.php");

}

?>

<html>

<body>

<?php

$username = $\_SESSION['myusername'];

echo "Welcome $username";?><a href="logout.php">Logout</a></body>

</html>