|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Government College of Engineering, Nagpur**        **Department of Computer Science & Engineering**    **3rd Year 5th Semester**  **Academic Year - 2020-21**  **Project - Movie Ticket Booking System**  **Submitted By: Query Coder’s**       |  |  |  | | --- | --- | --- | | **Name** | **Roll No.** | **Department** | | **Roshan Bonde (Leader)** | **54** | **CSE** | | **Aman Sharma** | **6** | **CSE** | | **Vishal Yadav** | **73** | **CSE** | | **Rohit Ramteke** | **52** | **CSE** | |

|  |  |
| --- | --- |
| **Government College of Engineering, Nagpur**  **Department of Computer Science & Engineering**       |  | | --- | | **DECLARATION** |   We hereby declare that this project based lab report entitled “**Movie ticket booking System**” has been prepared by us in Computer Science and Engineering during the academic  year 2020 - 21.  We also declare that this project based lab report is of our own effort and it has not  been submitted to any other university.  Date: 25 / 11 / 2020  Place: Nagpur |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ACKNOWLEDGEMENT      It is great pleasure for me to express my gratitude to our honourable HOD **Dr.Latesh Bhagat** for giving the opportunity and platform with facilities in accomplishing the project based laboratory report.I express sincere gratitude to our HOD for her administration towards academic growth and also providing us the efficient faculty and facilities to bring our ideas into reality.  Finally,it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.  **PROJECT ASSOCIATES**     |  |  |  | | --- | --- | --- | | **Name** | **Roll No.** | **Department** | | **Roshan Bonde (Leader)** | **54** | **CSE** | | **Aman Sharma** | **6** | **CSE** | | **Vishal Yadav** | **73** | **CSE** | | **Rohit Ramteke** | **52** | **CSE** | |

|  |
| --- |
| **TABLE OF CONTENTS**    Acknowledgment ………………………………………………………………. 4  Abstract ………………………………………..……………………………….. 6  Introduction ……………………………………………………………………. 7  Project Description ……………………………………………………...…....... 8  List of Entities & Attributes ……………………………………………………. 10  ER Diagram(Conceptual Model) ......................................................................... 12  Create & Insert SQL Queries …………………………………………………... 16  SQL Queries related to Report Generation …………………………………..... 19  Project Based Screenshot ……………………………………………………... 22  Conclusion ……………………………………………………………………... 24 |

|  |
| --- |
| **ABSTRACT**    The Movie Ticket Booking System facilitates the users to enquire about the  recent movies available movies,booking and cancellation of movie tickets according    to theatre type and class type,enquire about the status of the booked tickets, etc.  The aim of this project is to design a website that gives an easy platform for    people to get details of trending films and to get movie tickets in the    easiest possible way making it simple for all to buy tickets from anytime  and anywhere.  This project contains introduction to movie ticket booking system.It is    computerized way of reserving the seats of movie in advanced.This online movie ticket    booking system can make the process of booking movie tickets much easier than ever    before.Then this project contains entity relationship model diagram based on movie    ticket booking system an introduction to relational model also example of some sql  queries to retrieve data from the database of this system. |

|  |
| --- |
| **INTRODUCTION**  Database is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports processes requiring information. A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible. The DBMS manages three important Things: the data, the database engine that allows data to be accessed, locked or modified and the database schema, which defines the database’s logical structure. These three foundational elements help provide concurrency, security , data integrity and uniform administration procedures. The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data.  The main purpose of maintaining a database for Movie Ticket Booking System is to reduce the manual errors involved in the bookings and cancellation of tickets and make it convenient for the customers and the providers to maintain the data about their customers and also about the seats available at them. Due to automation many loopholes that exist in the manual maintenance of the records can be removed. The speed of obtaining and processing the data will be fast. For future expansion the proposed system can be web enabled so that clients can make various enquiries about movie ticket booking. Due to this, sometimes a lot of problems occur and they are facing many disputes with the customers. To solve the above problem,we design a database which includes customer details, availability of movie seats, etc. |

|  |
| --- |
| **PROJECT DESCRIPTION**  This project is about creating a webapp for movie ticket booking system. The movie ticket booking system allows the user to do enquiry and get information about the trending movies and get to know about availability of seats for that movie according to theatre type,class type etc. This website will provide users an easy way for booking and cancellation of movie tickets, enquire about the status of booked tickets, provide feedback about the system, etc. Also in this ticket booking webapp all types of payment methods are accepted. There is a feedback form also in this webapp through which users can express their views openly to us about our system and give their valuable suggestion about the system functioning.  Also there is an admin dashboard through which the system providers can keep track of functioning of the system. Through this admin dashboard system providers can see bookings of all movies and make new entries of movies according to new releases and edit data also delete booking in some cases. Also through this admin dashboard system providers keep track of the number of bookings and also maintains user feedback given by users of this system. Also through this admin dashboard system providers keep track of recent releases of movies and provide the same information to the users and also keep track of recent booking of the users.Also at the end of payment users get e-ticket containing details about the movie time, theatre type, class type etc. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **LIST OF ENTITIES & ATTRIBUTES**     |  |  | | --- | --- | | Entities | Attributes | | bookingtable | **bookingID**  movieID  bookingTheatre  bookingType  bookingDate  bookingTime  bookingFName  bookingLName  bookingPNumber  bookingEmail  amount  ORDERID  DATE-TIME | | feedbacktable | **msgID**  senderfName  senderlName  sendereMail  senderfeedback | | movietable | **movieID**  movieImg  movieTitle  movieGenre  movieDuration  movieRelDate  movieDirector  movieActors  mainhall  viphall  privatehall | |

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | users | **id**  username  name  password | |

|  |
| --- |
| **ER DIAGRAM(CONCEPTUAL MODEL)** |

|  |
| --- |
| CREATE & INSERT SQL QUERIES  Create Table Queries:   1. bookingtable   Query:  CREATE TABLE `bookingtable` (  `bookingID` int(11) NOT NULL,  `movieID` int(11) DEFAULT NULL,  `bookingTheatre` varchar(100) NOT NULL,  `bookingType` varchar(100) DEFAULT NULL,  `bookingDate` varchar(50) NOT NULL,  `bookingTime` varchar(50) NOT NULL,  `bookingFName` varchar(100) NOT NULL,  `bookingLName` varchar(100) DEFAULT NULL,  `bookingPNumber` varchar(12) NOT NULL,  `bookingEmail` varchar(255) NOT NULL,  `amount` varchar(255) NOT NULL,  `ORDERID` varchar(255) NOT NULL,  `DATE-TIME` datetime NOT NULL DEFAULT current\_timestamp()  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;   1. feedbacktable     Query:  CREATE TABLE `feedbacktable` (  `msgID` int(12) NOT NULL,  `senderfName` varchar(50) NOT NULL,  `senderlName` varchar(50) DEFAULT NULL,  `sendereMail` varchar(100) NOT NULL,  `senderfeedback` varchar(500) DEFAULT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |

|  |
| --- |
| 1. movietable     Query:  CREATE TABLE `movietable` (  `movieID` int(11) NOT NULL,  `movieImg` varchar(150) NOT NULL,  `movieTitle` varchar(100) NOT NULL,  `movieGenre` varchar(50) NOT NULL,  `movieDuration` int(11) NOT NULL,  `movieRelDate` date NOT NULL,  `movieDirector` varchar(50) NOT NULL,  `movieActors` varchar(150) NOT NULL,  `mainhall` int(11) NOT NULL,  `viphall` int(11) NOT NULL,  `privatehall` int(11) NOT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;   1. users   Query:  CREATE TABLE `users` (  `id` int(11) NOT NULL,  `username` varchar(80) NOT NULL,  `name` varchar(80) NOT NULL,  `password` varchar(80) NOT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=latin1; |

|  |
| --- |
| Insert Queries:     1. booking table   INSERT INTO `bookingtable` (`bookingID`, `movieID`, `bookingTheatre`,`bookingType`, `bookingDate`, `bookingTime`, `bookingFName`, `bookingLName`, `bookingPNumber`, `bookingEmail`,`amount`, `ORDERID`,`DATE-TIME`) VALUES (38, 1, 'private-hall', '7d', '13-3', '15-00', 'Rajat', 'Kulkarni', '7438344524', 'robin@gmail.com', '200.00', 'ORD74294887', '2020-11-25 18:07:24');   1. feedbacktable   INSERT INTO `feedbacktable` (`msgID`, `senderfName`, `senderlName`, `sendereMail`, `senderfeedback`) VALUES (1, 'Ahmed', 'Ali', 'Ahmed@mail.com', 'Hello first');   1. movietable   INSERT INTO `movietable` (`movieID`, `movieImg`, `movieTitle`, `movieGenre`, `movieDuration`, `movieRelDate`, `movieDirector`, `movieActors`, `mainhall`, `viphall`, `privatehall`) VALUES (1, 'img/movie-poster-1.jpg', 'Captain Marvel', ' Action, Adventure, Sci-Fi ', 220, '2018-10-18', 'Anna Boden, Ryan Fleck', 'Brie Larson, Samuel L. Jackson, Ben Mendelsohn', 0, 0, 0);   1. users   INSERT INTO `users` (`id`, `username`, `name`, `password`) VALUES (1, '123', 'Aman', '123'); |

|  |
| --- |
| **PROJECT BASED SCREENSHOT**  **Webapp Homepage**          **Webapp Ticket Booking Page**    **Webapp Payment Page**      **Webapp Reciept Page**      **Admin Dashboard**    **Admin Booking Details Page**    **Admin Movie Details Page**        **Admin Add Movie Page**      **Admin User Feedback Page** |

|  |
| --- |
| **TECHNOLOGY STACK**  **Front Technologies:**        **HTML CSS JAVASCRIPT**  **Database Technologies:**        **MySQL PHPMyAdmin**      **Backend Technologies:**      **PHP** |

|  |
| --- |
| **CONCLUSION**  In our project movie ticket booking system we have stored all information about the movies schedules and the users booking tickets and also status of availability of seats in particular theatre, movie, class type etc.This system will be helpful for the users want to book a movie tickets in an easiest available way also get information about the recent releases of the movies also get some small information about film. We have considered the most important requirements only for now,many more features and details can be added to our project in order to obtain even more user friendly applications.These applications are already in progress and in future they can be upgraded and may become part of amazing technology. |