CHAPTER 1

INTRODUCTION OF THE PROJECT

1.1 Introduction of the Online Movie Ticket Booking System:

Welcome to newly designed website movie ticket booking is a faster, cleaner and a tad more personal website, specially designed to make your booking experience better. Log on, navigate and find out for yourselves and if time permits leave your valuable feedback. Customers may view the contents of any movie show at any time and may book any movie ticket as needed.

The program automatically calculates the subtotal and grand total. When a visitor decides to finally book the ticket, the order information including the buyer's name, address and billing instruction is stored in the database securely and payment has been made. The combo booking is also provided at the time of booking the ticket and there's a wonderful facility of delivering the combos at your seat when you are watching the movie.

You need to register a new user whenever you have first visited or site then for future it will be stored in our database permanently and you can book you movie ticket at any time you want with this username and password.

The proposed project "Online Movie Ticket Booking System" is the process whereby consumers directly buy movie tickets online from a multiplex website interactively in real-time without an intermediary service over the Internet. The Given System provides the detailed working of the Online Movie Ticket Booking Processing and what all happens whenever we or any one of us goes to book movie tickets online.

1.2 Abstract of the Project Online Movie Ticket Booking System :

This project is aimed at developing an online ticket reservation system for a Cinema Hall. The Ticket Reservation System is an Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. This application will automate the reservation of tickets. The website provides complete information regarding currently running movies on all the screens with details of show timings, available seats and fare charges of different classes. Seats can be reserved for different classes as well for same show and screen also. Ticket reservations are done using credit card and can be cancelled if needed.

Our 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. Our reservation system also provides option to cancel the tickets which are reserved previously.

In the current, the customer needs to visit theater for booking seats. Further, they don't have the data about the movie which is in the cinema lobby, its show times and distinctive rates of the ticket. The client will be most likely unable to get data about various film halls available in the city. In this way, in the event that he wishes to see a movie on a specific day he needs to first meander around the city to discover where it is being appeared at the particular time.

In order to overcome the existing problem, we are making this system as online and offline through telecommunication and that shows every information about any cinema hall, movie, snacks, vehicle parking and cancellation ticket can be done by online and as well as offline mode. This helps the customer to get better facilities at his own computer or laptop.

1.3 Software Requirement Specification

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

The proposed system has the following requirements:

- User-friendly interface: The platform should have an easy-to-use interface that allows users to book tickets, watch trailers, and share their review.
- Robust security features: The platform should be equipped with strong security measures to protect users' privacy and prevent any unauthorized access to their data. System need to maintain quantity record.
- User should be able to upload review.
- User should be able to book no. of tickets based on availability of seats.
- User should be able to watch trailers of current running and upcoming movies.

1.4 Modules of Online Movie Ticket Booking System :

- User Authentication Module: This module is responsible for managing user accounts, passwords, and login credentials. It should include features such as password encryption, password reset, and account verification.
- Movies Module: This module is responsible for processing and manipulating movies by categories and languages.
- Ticket Booking Module: This module is responsible for sbboking and retrieving tickets to users. It should include features such as booking, theater, and high availability.
- Contact Module: This module is responsible for contacting to the admin.
- About Module: This will allow user to display about the admin.
- Admin Module : In this module admin can add movies, remove movies, change timing, read reviews and delete reviews, etc.
- Login Module: Used for managing the login details
- Users Module : Used for managing the users of the WebApp.

CHAPTER 2

BRIEF OVERVIEW OF THE TECHNOLOGY

2.1 Objective of Project on Online Movie Ticket Booking System:

The project "Online Movie Ticket Booking System" is dedicated to the general requirements of multiplex theaters. The main objective of the project is to create an Online Movie Ticket Booking processing that allows customers to know about new movies, their schedules, cinema locations, class and ticket price etc. In the booking process when customer selects his city then cinemas of that city are filtered. In next step he/she selects his desired cinema where he/she wish to see movie, then selects movie and other details like show date, show time, class and no of tickets. Based on given parameters a graphical layout of seat status is visible to the customer. Now customer can select his desired seat location and number of seats. The Administrator will be able to see all booked and canceled tickets.

The main objectives of "Online Movie Ticket Booking System" project are as follows:

- Facility to store the information of new customer, different types of movie show timing, ticket rates of different types on show class etc.
- Interest to develop a good user friendly website with many online transactions using a database.
- Facility to generate different reports, which are helpful for the management in decision making.
- Facility to change user's password account.
- To increase my knowledge horizon in technologies like C#, SQL, CSS, HTML.
- To gain good experience in C# before joining in a full time job. Online Movie Ticket Booking System. To gain expertise using Data Grid, Data Set, Data Table, Data Adapter and Data Readers.

System should also provide accessories such as calculator, month viewer.

2.2 Functionalities provided by Online Movie Ticket Booking System are as follows:

- Provides simple Sign in page for user.
- User authentication: Users can create an account, log in, and manage their profile.
- Book movie tickets: Users can book tickets for movies.
- Upcoming movies: Users can view trailers of upcoming movies also.
- Privacy and security: The platform will provide robust privacy and security features to protect user data.
- User can manages the Profile password.

2.3 Identification of need:

The need for a online movie ticket movie booking system arises from the increasing demand for online ticket booking and the desire of users to connect with others through shared experiences. People have always been book to visual content, but with the rise of online booking system.

However, many existing online movie ticket booking apps or websites, such as Book my show, PVR, have become overcrowded and noisy, making it difficult for users to stand out and get noticed. Additionally, these platforms are not optimized for user friendly UI, and ticket are given on the website.

Following points should be well considered:

- User experience: The platform should provide a user-friendly and engaging experience that encourages users to stay on the platform and explore new content.
- Design: The platform should have an attractive and modern design that reflects the visual nature of the platform and enhances the user experience.
- Performance: The platform should be fast and reliable, with minimal downtime or technical issues.
- Privacy: The platform should provide users with control over their privacy settings and ensure that their personal information is kept confidential.
- Scalability: The platform should be designed to handle large volumes of users and data, with the ability to scale up as needed.
- Mobile optimization: The platform should be optimized for mobile devices, as many users will be accessing the platform from their smartphones.

CHAPTER 3

LITERATURE REVIEW

3.1 Feasibility Study:

Feasibility study is the process of determination of whether or not a project is worth doing. Feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report. The contents and recommendations of this feasibility study helped us as a sound basis for deciding how to precede the project. It helped in taking decisions such as which software to use, hardware combinations, etc.

A. Economical Feasibility

Economical feasibility has great importance as it can outweigh other feasibilities because costs affect organization decisions. The concept of Economic Feasibility deals with the fact that a system that can be developed and will be used on installation must be profitable for the Organization. The cost to conduct a full system investigation, the cost of hardware and software, the benefits in the form of reduced expenditure are all discussed during the economic feasibility.

During the economical feasibility test we maintained the balance between the Operational and Economical feasibilities, as the two were the conflicting. For example the solution that provides the best operational impact for the end-users may also be the most expensive and, therefore, the least economically feasible.

B. Technical Feasibility

Technical feasibility determines whether the work for the project can be done with the existing equipment, software technology and available personnel. Technical feasibility is concerned with specifying equipment and software that will satisfy the user requirement. This proposed project “Online Movie Ticket Booking System” is feasible on technical remarks also, as the proposed project is more beneficiary in terms of having a sound proof system with new technical components installed on the system. The proposed system can run on any machines supporting Windows and Internet services and works on the best software and hardware that had been used while designing the system so it would be feasible in all technical terms of feasibility.

C. Operational Feasibility

Our proposed project “Online Movie Ticket Booking System” is operationally feasible since there is no need for special training of staff member and whatever little instructing on this system is required can be done so quite easily and quickly as it is essentially This project is being developed keeping in mind the general people who one have very little knowledge of computer operation, but can easily access their required database and other related information. The redundancies can be decreased to a large extent as the system will be fully automated.

3.2 Preliminary Product Description:

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed judgment about the feasibility of the proposed project.

Analysts working on the preliminary investigation should accomplish the following objectives:

- Clarify and understand the project request
- Determine the size of the project.
- Assess costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.
- Report the findings to management, with recommendations outlining the acceptance or rejection of the proposal.

The Initial Cost

The initial cost of setting up the system will include the cost of hardware software (OS, add-on software, utilities) & labour (setup & maintenance). The same has to bear by the organization.

3.3 Tools/Platform, Hardware and Software Requirement specifications:

Software Requirements:

NAME OF COMPONENT	SPECIFICATION
Operating System	Windows 11
Launguage	PHP
Database	MySql Server
Browser	Any of Mozilla, Opera, Chrome etc

Hardware Requirements:

NAME OF COMPONENT	SPECIFICATION
Processor	Intel I3
RAM	8GB
Hard Disk	Minimum 30Gb

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 Implementation Methodology:

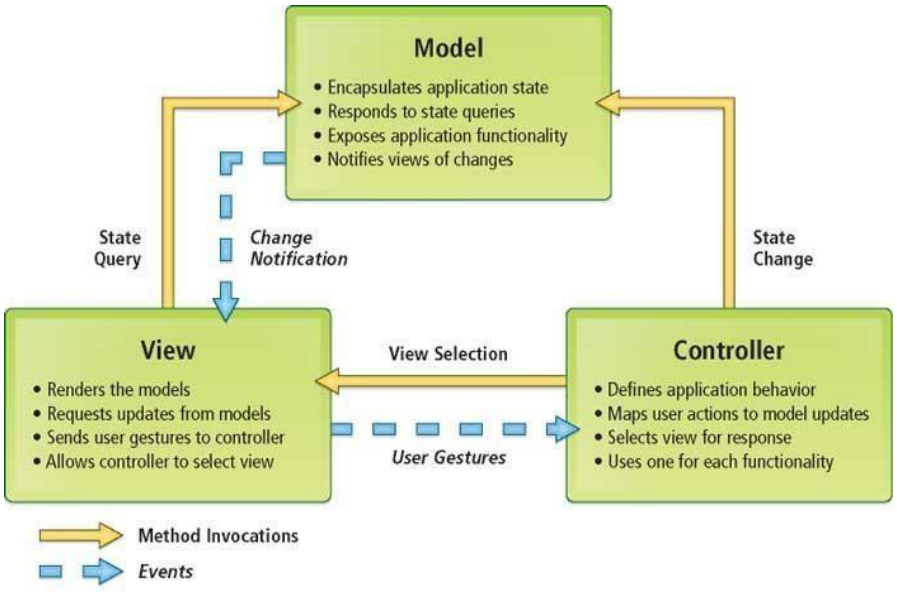
Model View Controller or MVC as it is popularly called, is a software design pattern for developing web applications. A Model View Controller pattern is made up of the following three parts:

- **Model** - The lowest level of the pattern which is responsible for maintaining data.
- **View** - This is responsible for displaying all or a portion of the data to the user.
- **Controller** - Software Code that controls the interactions between the Model andView.

MVC is popular as it isolates the application logic from the user interface layer and supports separation of concerns. Here the Controller receives all requests for the application and then works with the Model to prepare any data needed by the View. The View then uses the data prepared by the Controller to generate a final presentable response. The MVC abstraction can be graphically represented as follows.

MVC (Model View Controller Flow) Diagram

DATA FLOW DIAGRAMS



4.2 System Design of Online Movie Ticket Booking System

In this phase, a logical system is built which fulfils the given requirements. Design phase of software development deals with transforming the user's requirements into alogically working system. Normally, design is performed in the following in the following two steps:

1. **Primary Design Phase:**
   In this phase, the system is designed at block level. The blocks are created on the basis of analysis done in the problem identification phase. Different blocks are created for different functions emphasis is put on minimising the information flow between blocks. Thus, all activities which require more interaction are kept in one block.
2. **Secondary Design Phase:**
   In the secondary phase the detailed design of every block is performed.

The general tasks involved in the design process are the following:

1. Design various blocks for overall system processes.
2. Design smaller, compact and workable modules in each block.
3. Design various database structures.
4. Specify details of programs to achieve desired functionality.
5. Design the form of inputs, and outputs of the system.
6. Perform documentation of the design.
7. System reviews.

4.3 Cost estimation of the project:

Software cost comprises a small percentage of overall computer-based system cost. There are a number of factors, which are considered, that can affect the ultimate cost of the software such as - human, technical, Hardware and Software availability etc.

The main point that was considered during the cost estimation of **project** was its sizing. In spite of complete software sizing, function point and approximate lines of code were also used to "size" each element of the Software and their costing.

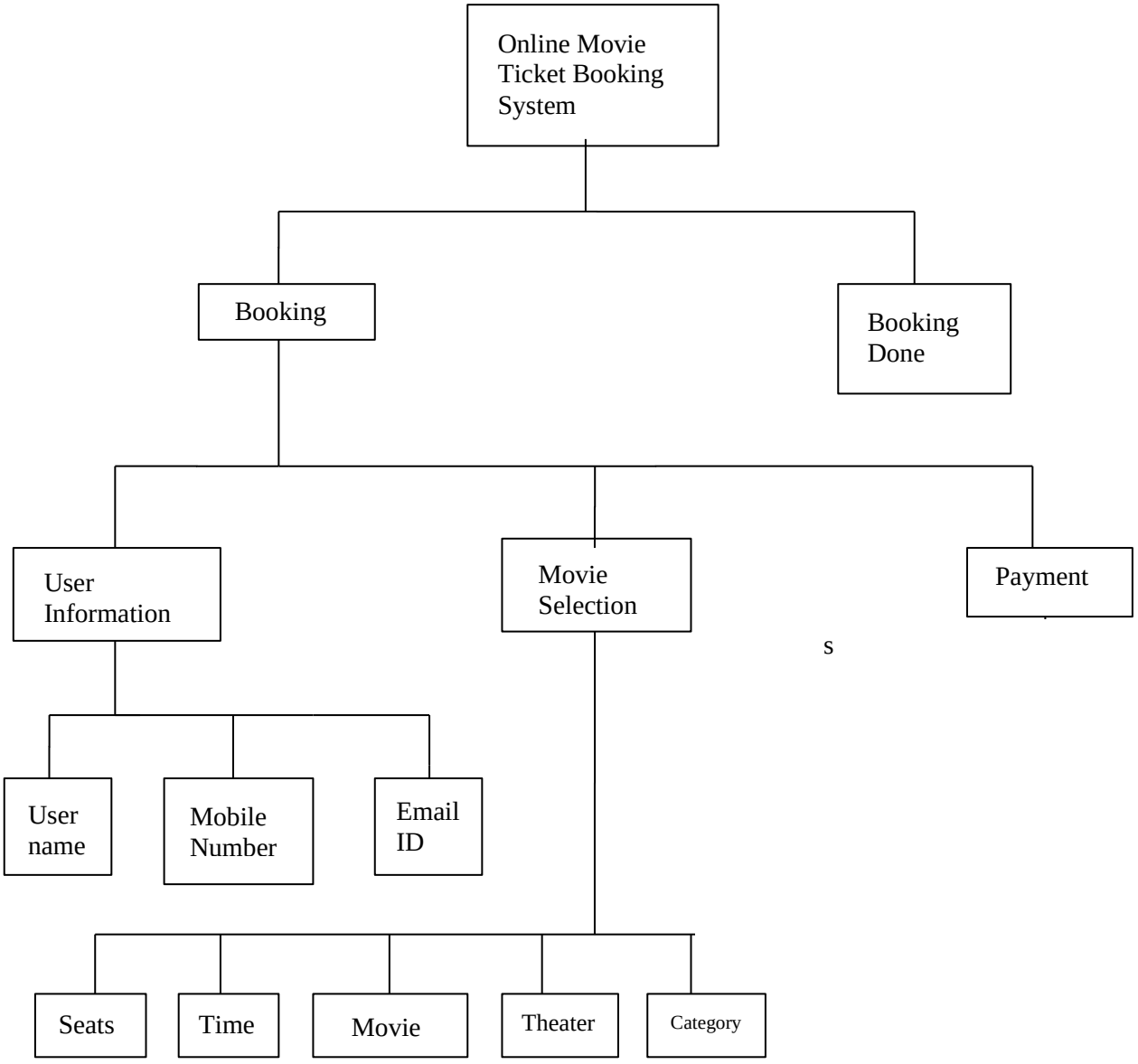
The cost estimation done by me for **Project** also depend upon the baseline metrics collected from past projects and these were used in conjunction with estimation variables to develop cost and effort projections.

We have basically estimated this project mainly on two bases -

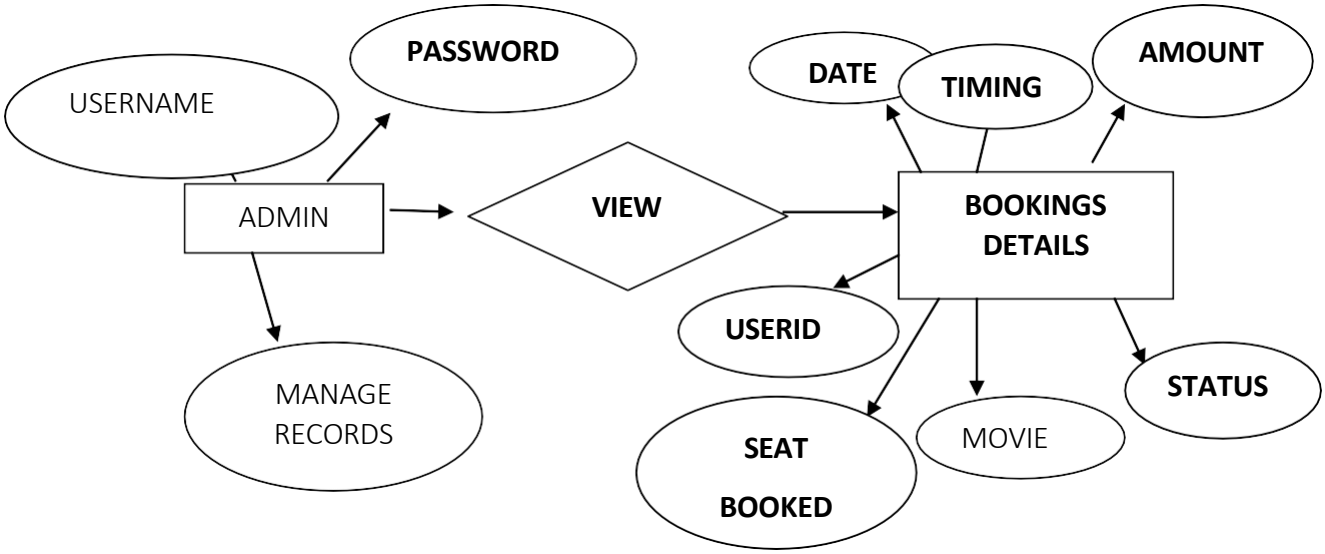
1) Effort Estimation - This refers to the total man-hours required for the development of the project. It even includes the time required for doing documentation and user manual.

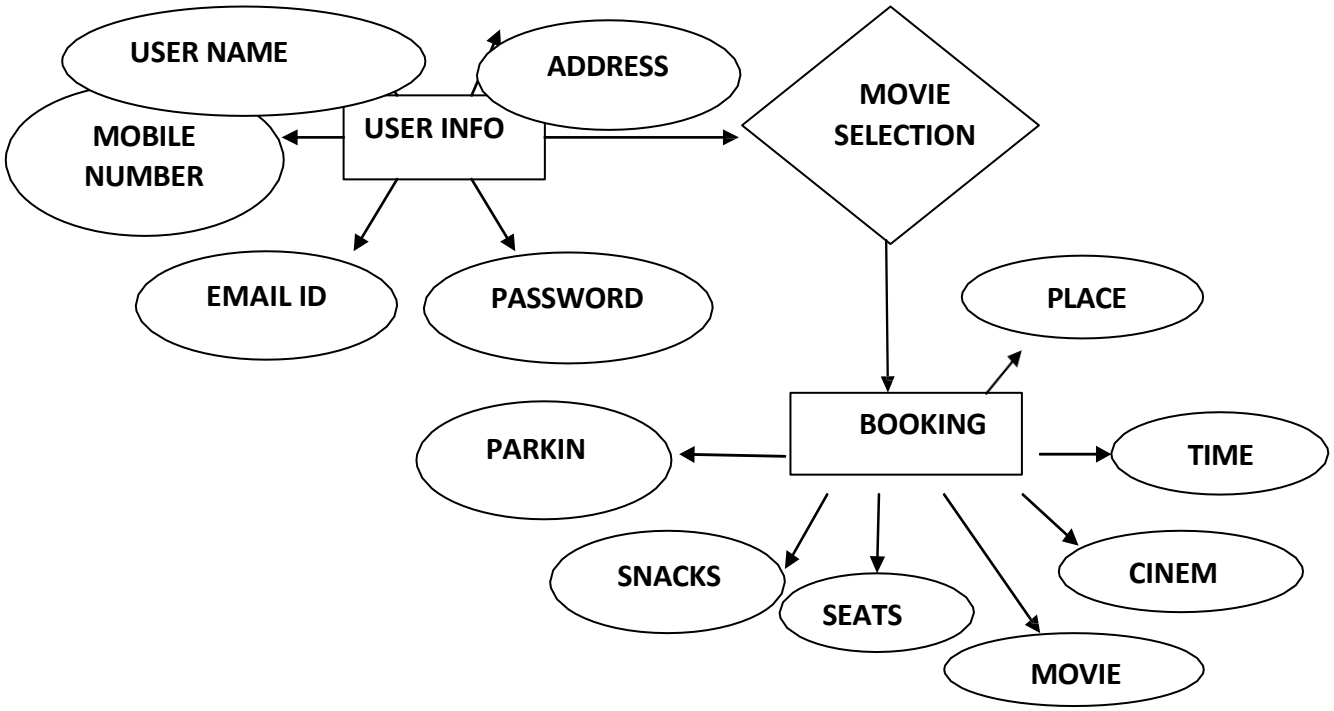
2) Hardware Required Estimation - This includes the cost of the PCs and the hardware cost required for development of this project.

4.4 HIERARCHIAL DIAGRAM:



4.5 ER DIAGRAMS





4.6 Test Cases

- **Test Case for Login / Signup Page**

Sr. No	Test ID	Objective	Description	Expected Result
1	TC001	To check textures whether it accept char and numeric both	Type both char And numeric In text users	Should Accept
2	TC002	Text user should maximum 10 characters	Type 10 char In text user	Give Error Message
3	TC003	Text password Should not display Asterisk	Type in the Text password	Only asterisk

4	TC004	To check the functionality of Login button	Type valid username and Password	Redirect to home page
5	TC005	To check the working of login button	Type either invalid username or invalid password	Error message

CHAPTER 5

SECURITY & SOFTWARE SPECIFICATION TESTING

5.1 Security Testing of the Project

Testing is vital for the success of any software. No system design is ever perfect. Testing is also carried in two phases. First phase is during the software engineering that is during the module creation. Second phase is after the completion of software. This is system testing which verifies that the whole set of programs hanged together.

White Box Testing:

In this technique, the close examination of the logical parts through the software are tested by cases that exercise species sets of conditions or loops. All logical parts of the software checked once. Errors that can be corrected using this technique are typographical errors, logical expressions which should be executed once may be getting executed more than once and errorresulting by using wrong controls and loops. When the box testing tests all the independent part within a module a logical decisions on their true and the false side are exercised , all loops and bounds within their operational bounds were exercised and internal data structure to ensuretheir validity were exercised once.

Black Box Testing:

This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements for a program. Black box testing tests the input, the output and the external data. It checks whether the input data is correct and whether we are getting the desired output.

Alpha Testing:

Acceptance testing is also sometimes called alpha testing. Be spoke systems are developed for a single customer. The alpha testing proceeds until the system developer and the customer agree that the provided system is an acceptable implementation of the system requirements.

Beta Testing:

On the other hand, when a system is to be marked as a software product, another process called beta testing is often conducted. During beta testing, a system is delivered among a number of potential users who agree to use it. The customers then report problems to the developers. This provides the product for real use and detects errors which may not have been anticipated by the system developers.

Unit Testing:

Each module is considered independently. it focuses on each unit of software as implementedin the source code. it is white box testing.

Integration Testing:

Integration testing aims at constructing the program structure while at the same constructing tests to uncover errors associated with interfacing the modules. Modules are integrated by usingthe top down approach.

Validation Testing:

Validation testing was performed to ensure that all the functional and performance requirements are met.

System Testing:

It is executing programs to check logical changes made in it with intention of finding errors. a system is tested for online response, volume of transaction, recovery from failure etc. System testing is done to ensure that the system satisfies all the user requirements.

5.2 Software Specification Testing:

Detailed Design of Implementation

This phase of the systems development life cycle refines hardware and software specifications, establishes programming plans, trains users and implements extensive testing procedures, to evaluate design and operating specifications and/or provide the basis for further modification.

Technical Design

This activity builds upon specifications produced during new system design, adding detailed technical specifications and documentation.

Test Specifications and Planning

This activity prepares detailed test specifications for individual modules and programs, job streams, subsystems, and for the system as a whole.

Programming and Testing

This activity encompasses actual development, writing, and testing of program units or modules.

User Training

This activity encompasses writing user procedure manuals, preparation of user trainingmaterials, conducting training programs, and testing procedures.

Acceptance Test

A final procedural review to demonstrate a system and secure user approval before a system becomes operational.

Installation Phase

In this phase the new computerized system is installed, the conversion to new procedures is fully implemented, and the potential of the new system is explored.

System Installation

The process of starting the actual use of a system and training user personnel in its operation.

Review Phase

This phase evaluates the successes and failures during a systems development project, and to measure the results of a new Computerized Transystem in terms of benefits and savings projected at the start of the project.

Development Recap

A review of a project immediately after completion to find successes and potential problems in future work.

Post - Implementation Review

A review, conducted after a new system has been in operation for some time, to evaluate actual system performance against original expectations and projections for cost-benefit improvements. Also identifies maintenance projects to enhance or improve the system.

5.3 The Steps In Software Testing

The steps involved during Unit testing are as follows:

a. Preparation of the test cases.
b. Preparation of the possible test data with all the validation checks.
c. Complete code review of the module.
d. Actual testing done manually.
e. Modifications done for the errors found during testing.
f. Prepared the test result scripts.

The unit testing done included the testing of the following items:

1. Functionality of the entire module/forms.
2. Validations for user input.
3. Checking of the Coding standards to be maintained during coding
4. Testing the module with all the possible test data.
5. Testing of the functionality involving all type of calculations etc.
6. Commenting standard in the source files.

After completing the Unit testing of all the modules, the whole system is integrated with all its dependencies in that module. While System Integration, We integrated the modules one by one and tested the system at each step. This helped in reduction of errors at the time of the system testing.

The steps involved during System testing are as follows:

- Integration of all the modules/forms in the system.
- Preparation of the test cases.
- Preparation of the possible test data with all the validation checks.
- Actual testing done manually.
- Recording of all the reproduced errors.
- Modifications done for the errors found during testing.
- Prepared the test result scripts after rectification of the errors.

The System Testing done included the testing of the following items:

1. Functionality of the entire system as a whole.
2. User Interface of the system.
3. Testing the dependent modules together with all the possible test data scripts.
4. Verification and Validation testing.
5. Testing the reports with all its functionality.

After the completion of system testing, the next following phase was the Acceptance Testing. Clients at their end did this and accepted the system with appreciation. Thus, we reached the final phase of the project delivery.

There are other six tests, which fall under special category. They are described below:

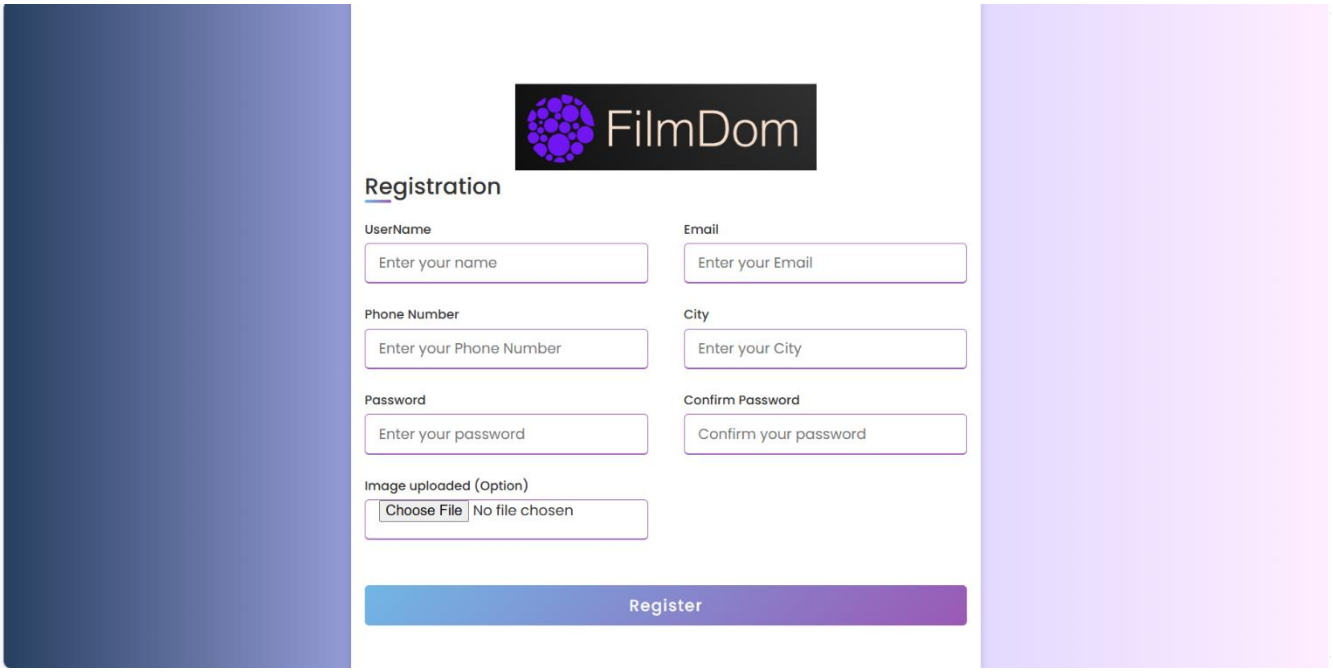
- Peak Load Test: It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time
- Storage Testing: It determines the capacity of the system to store transaction data on a disk or in other files.
- Performance Time Testing: it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.
- Recovery Testing: This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.
- Procedure Testing: It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.
- Human Factors Testing: It determines how users will use the system when processing dataor preparing reports.

CHAPTER 6

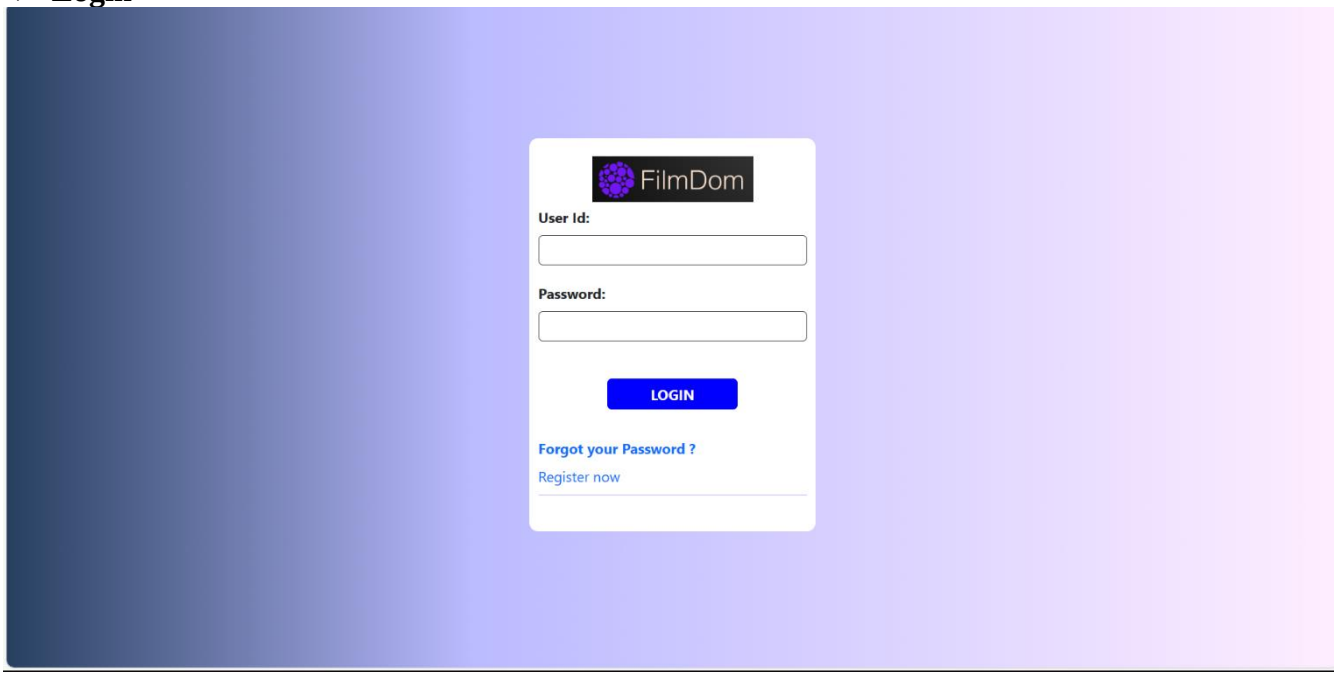
PROJECT SCREENS & CODE SAMPLE

6.1 Website Screens :

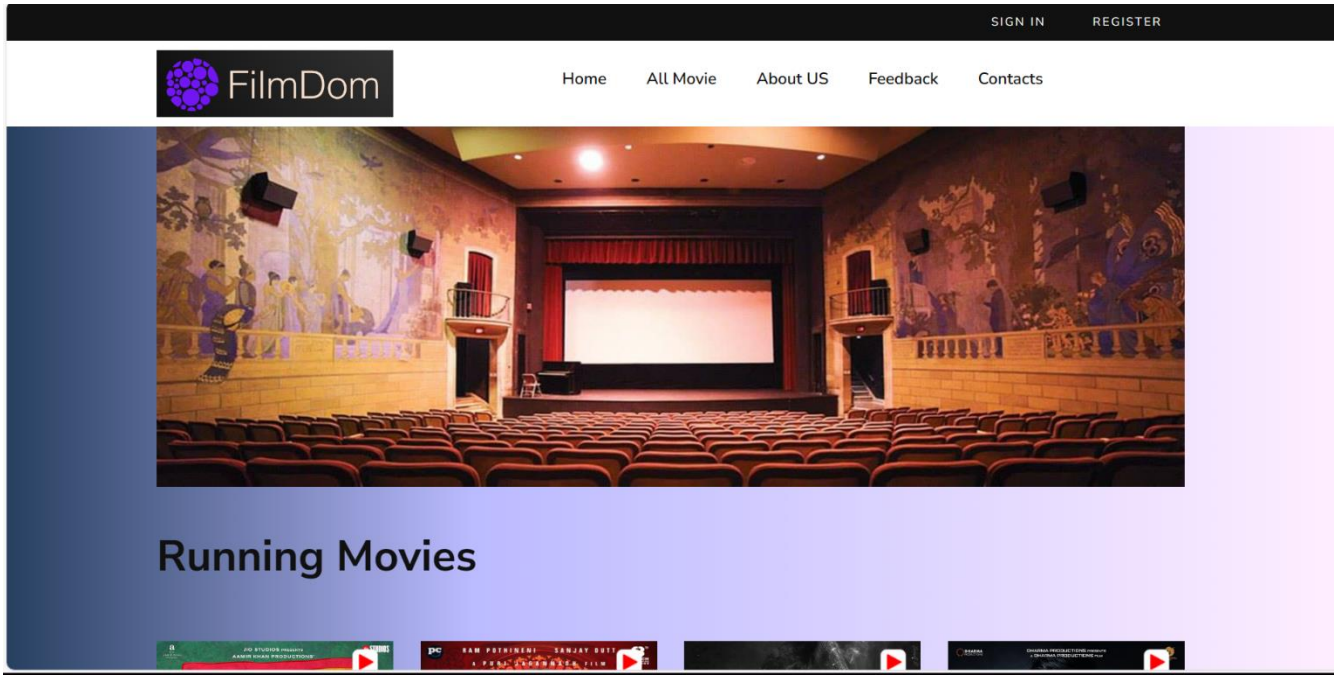
❖ **Register**

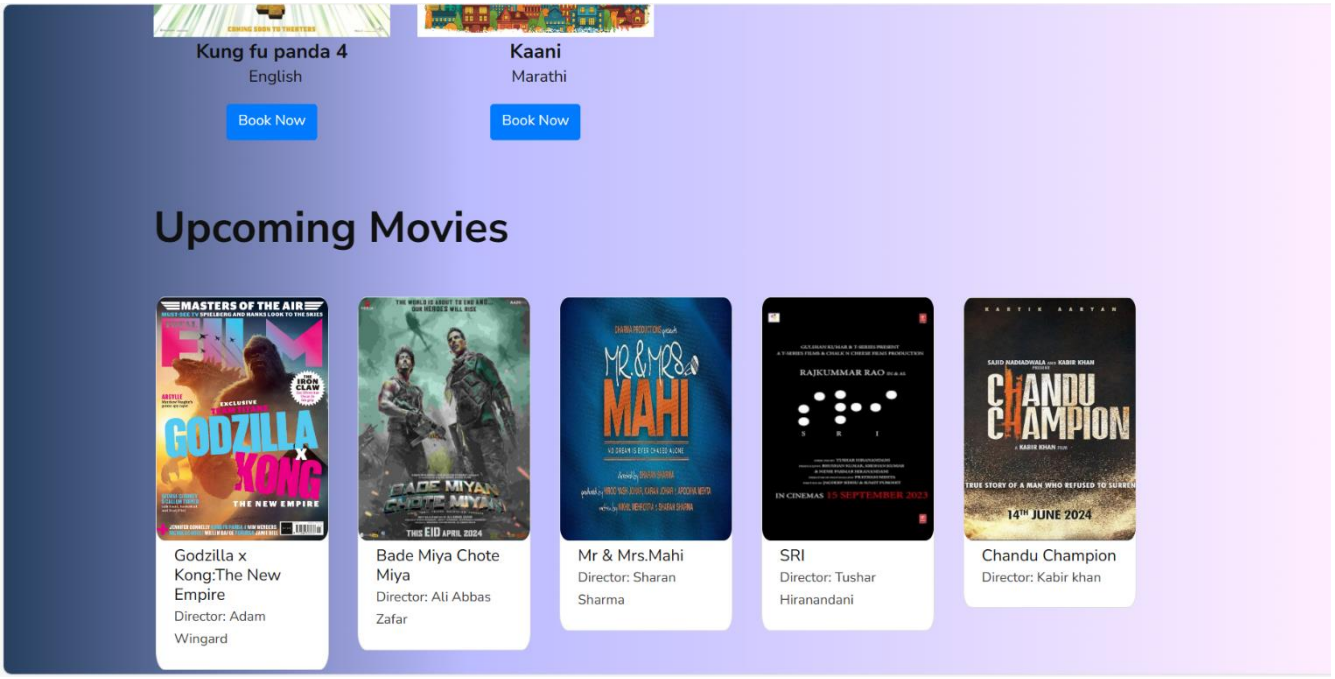


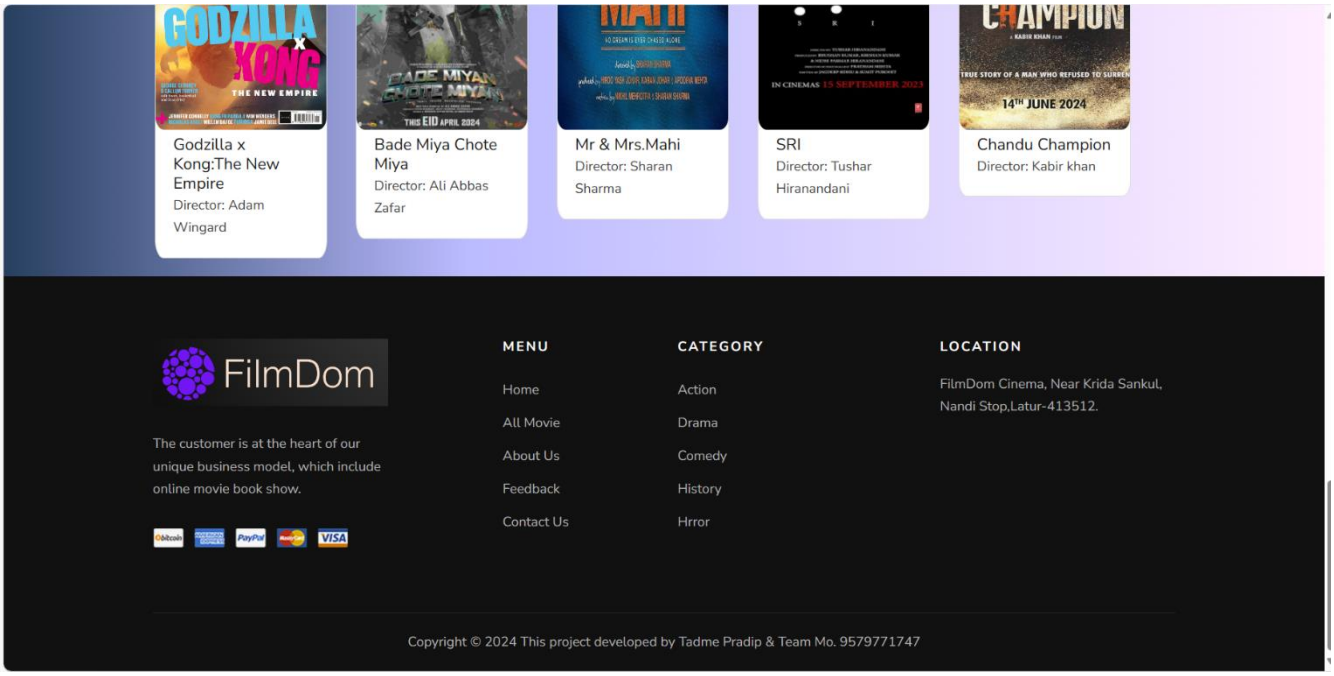
❖ **Login**



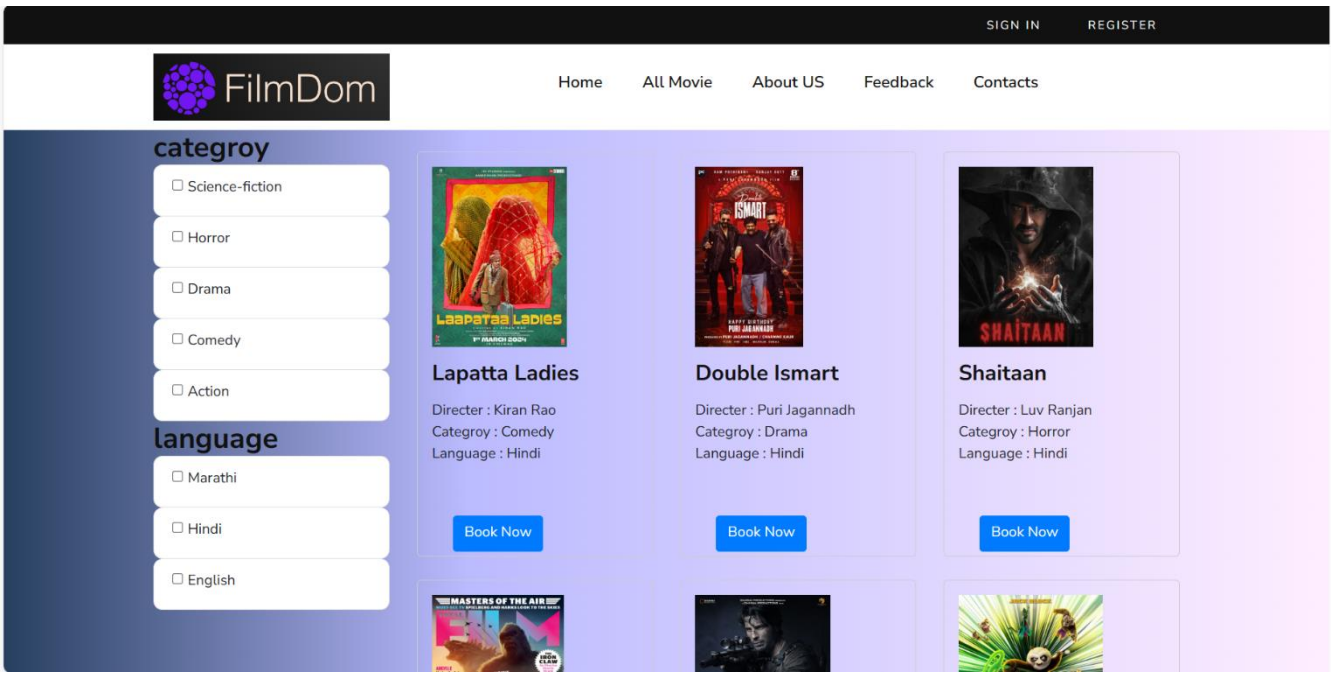
❖ Home


SIGN IN
REGISTER
FilmDom
Home
All Movie
About US
Feedback
Contacts
Running Movies

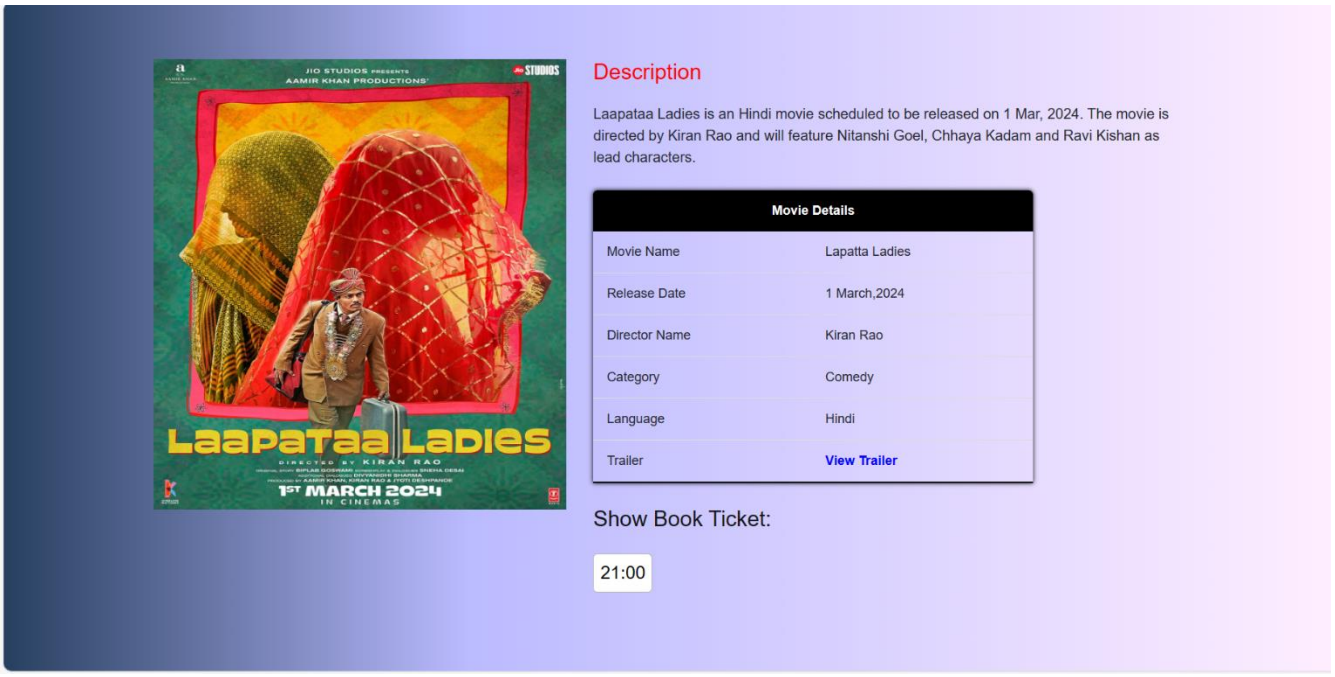

Kung fu panda 4
English
Book Now
Kaani
Marathi
Book Now
Upcoming Movies
Godzilla x Kong:The New Empire
Director: Adam Wingard
Bade Miya Chote Miya
Director: Ali Abbas Zafar
Mr & Mrs.Mahi
Director: Sharan Sharma
SRI
Director: Tushar Hiranandani
Chandu Champion
Director: Kabir khan

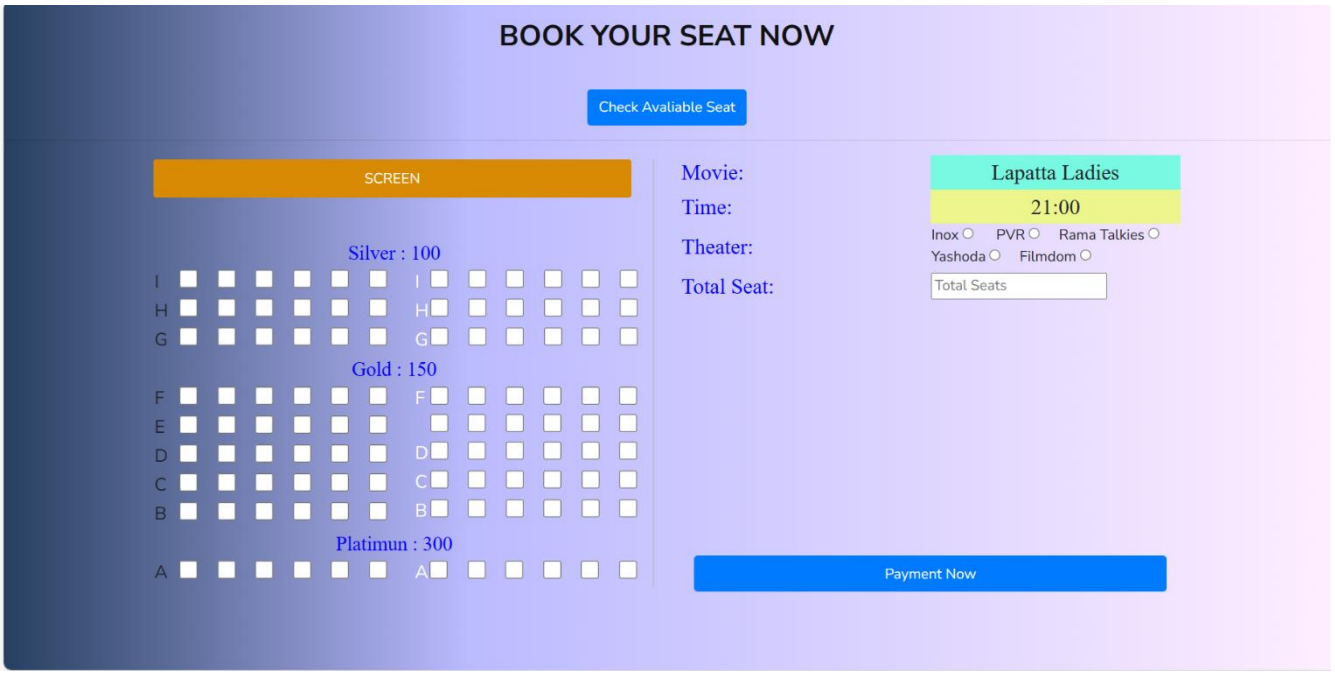

Godzilla x Kong:The New Empire
Director: Adam Wingard
Bade Miya Chote Miya
Director: Ali Abbas Zafar
Mr & Mrs.Mahi
Director: Sharan Sharma
SRI
Director: Tushar Hiranandani
Chandu Champion
Director: Kabir khan
FilmDom
The customer is at the heart of our unique business model, which include online movie book show.
MENU
Home
All Movie
About Us
Feedback
Contact Us
CATEGORY
Action
Drama
Comedy
History
Hrror
LOCATION
FilmDom Cinema, Near Krida Sankul, Nandi Stop,Latur-413512.
Copyright © 2024 This project developed by Tadme Pradip & Team Mo. 9579771747

❖ Categories

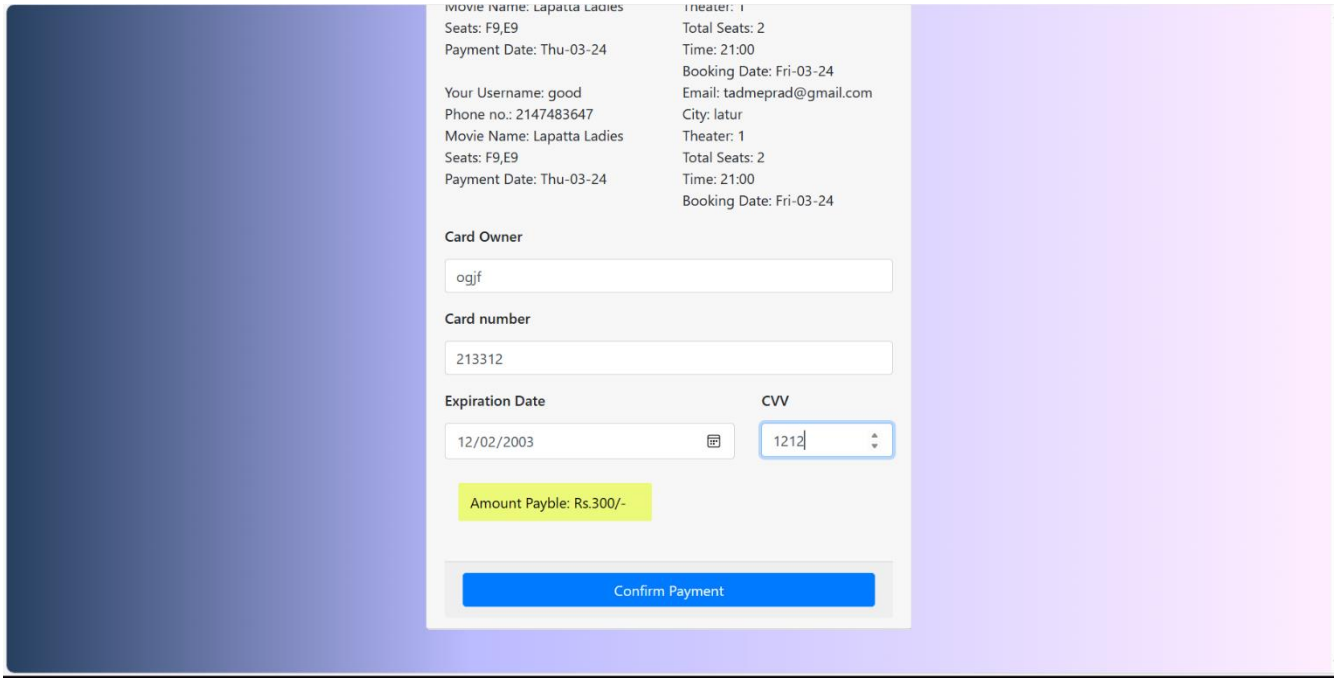

SIGN IN
REGISTER
FilmDom
Home
All Movie
About US
Feedback
Contacts
categroy
Science-fiction
Horror
Drama
Comedy
Action
language
Marathi
Hindi
English
Lapatta Ladies
Directer : Kiran Rao
Categroy : Comedy
Language : Hindi
Book Now
Double Ismart
Directer : Puri Jagannadh
Categroy : Drama
Language : Hindi
Book Now
Shaitaan
Directer : Luv Ranjan
Categroy : Horror
Language : Hindi
Book Now

❖ Ticket Booking

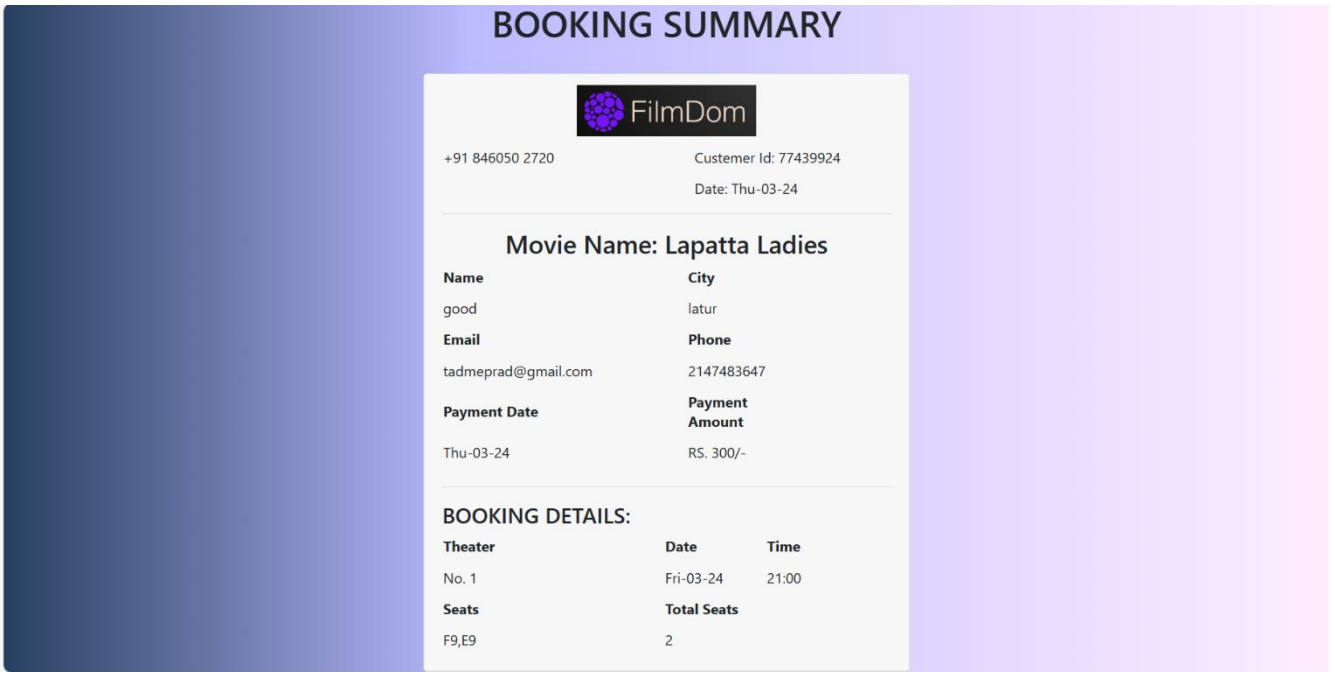




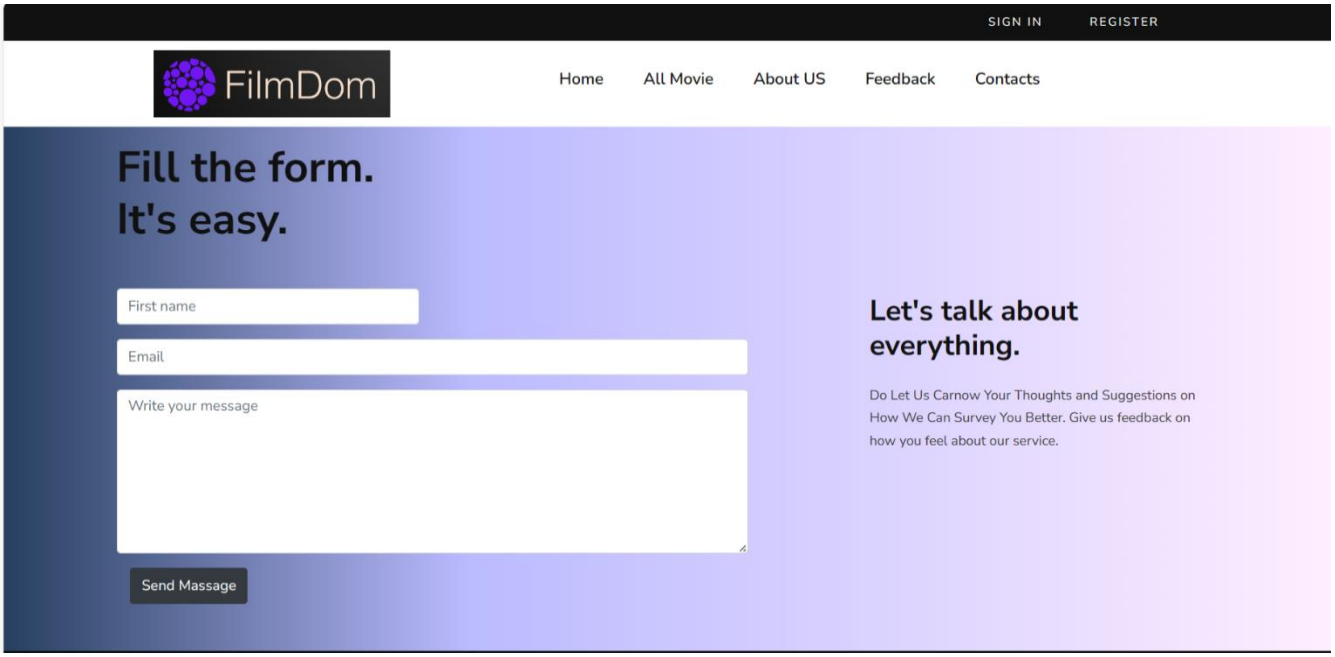
❖ Payment



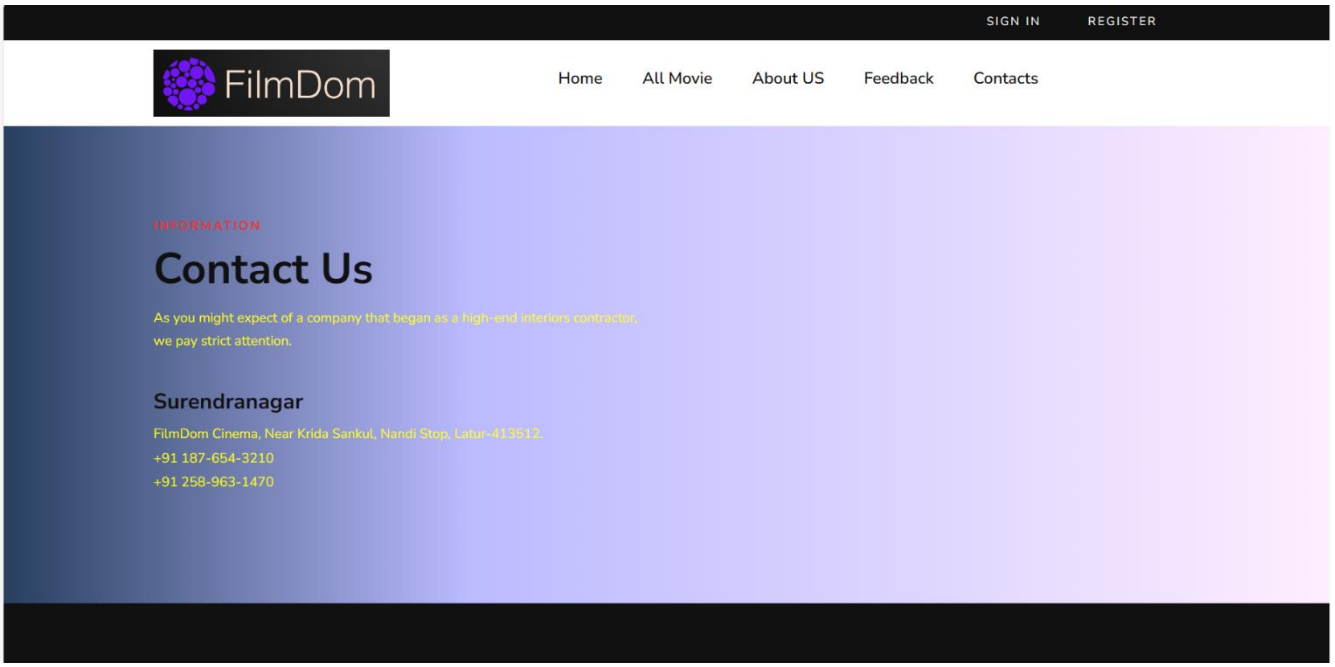
❖ Ticket



❖ Feedback



❖ Contact Us



6.2 CODE SAMPLES :

Directory Structure

Name	Date modified	Type	Size
admin	3/3/2024 11:57 AM	File folder	
css	3/3/2024 11:49 AM	File folder	
fonts	3/3/2024 11:49 AM	File folder	
image	3/3/2024 11:49 AM	File folder	
img	3/3/2024 11:49 AM	File folder	
js	3/3/2024 11:49 AM	File folder	
2	2/22/2024 11:31 PM	Python Source File	1 KB
2	2/22/2024 11:31 PM	Text Document	0 KB
about	3/5/2024 8:25 AM	PHP Source File	8 KB
allmovie	3/3/2024 3:40 PM	PHP Source File	6 KB
allmovie_fetch	2/3/2024 6:00 PM	PHP Source File	3 KB
contact	3/5/2024 8:23 AM	PHP Source File	4 KB
Database	2/3/2024 6:00 PM	PHP Source File	1 KB
database_connection	2/3/2024 6:00 PM	PHP Source File	1 KB
div	3/9/2024 6:59 PM	Opera Web Docu...	1 KB
feedback	3/5/2024 8:18 AM	PHP Source File	7 KB
feedback_action	2/3/2024 6:00 PM	PHP Source File	1 KB
footer	2/3/2024 7:44 PM	PHP Source File	4 KB
forget	2/3/2024 6:00 PM	PHP Source File	1 KB

- Index.php:

```
 <?php
session_start();
?>

<!DOCTYPE html>
<html>
<head>
   <meta charset="UTF-8">
   <meta name="description" content="Male_Fashion Template">
   <meta name="keywords" content="Male_Fashion, unica, creative, html">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>FilmDom</title>

   <!-- Google Font -->
   <link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans:wght@300;400;600;700;800;900&displa
y=swap"
   rel="stylesheet">
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css">
   <link href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-
free@5.15.3/css/fontawesome.min.css">

   <!-- Css Styles -->
   <link rel="stylesheet" href="css/bootstrap.min.css" type="text/css">
   <link rel="stylesheet" href="css/font-awesome.min.css" type="text/css">
   <link rel="stylesheet" href="css/elegant-icons.css" type="text/css">
   <link rel="stylesheet" href="css/magnific-popup.css" type="text/css">
   <link rel="stylesheet" href="css/nice-select.css" type="text/css">
   <link rel="stylesheet" href="css/slicknav.min.css" type="  text/css">
   <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body style="background: linear-gradient(90deg, #274060 0%,#bbbbff  35%, #ffeeff 100%);">
<?php
include("header.php");
?>
<div class="container">
  <img src=image/theatre_2.jpg alt="" class="image-resize" style="width: 100%; height: 400px;">
</div>
<br><br>
<div class="container">
   <h2>Running Movies</h2><br><br>
    <div class="row">
<?php
include_once 'Database.php';
$result = mysqli_query($conn,"SELECT * FROM add_movie");

if (mysqli_num_rows($result) > 0) {
 while($row = mysqli_fetch_array($result)) {
    if($row['action']== "running"){
   ?>
```

```
        <div class="col-lg-3 col-md-3 col-sm-6">
          <div class="running-movie">
           <img src=admin/image/<?php echo $row['image']; ?> alt="" class="image-resize2"
style="width: 100%;">
            <div class="top-right">
              <a data-toggle="modal" data-target="#tailer_modal<?php echo $row['id'];?>"><img
src="img/icon/play.png"></a></div>
              <h5><b><?php echo $row['movie_name'];?></b></h5>
              <h6><center><?php echo $row['language'];?></center></h6>
             <a href="movie_details.php?pass=<?php echo $row['id'];?>" class="btn btn-primary">Book
Now</a>
          </div>
         </div>

         <div class="modal fade" id="tailer_modal<?php echo $row['id'];?>" tabindex="-1"
role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
          <div class="modal-dialog" role="document">
            <div class="modal-content">
              <embed style="width: 820px; height: 450px;" src="<?php echo
$row['you_tube_link'];?>"></embed>
            </div>
          </div>
        </div>

  <?php
}
  }
}
?>
</div>
     </div>
<br><br>
<div class="container">
   <h2>Upcoming Movies</h2><br><br>
   <div class="row">
     <?php
include_once 'Database.php';
$result = mysqli_query($conn,"SELECT * FROM add_movie");

if (mysqli_num_rows($result) > 0) {
  while($row = mysqli_fetch_array($result)) {
     if($row['action']== "upcoming"){
   ?>       <div class="image-box">
        <div class="col-lg-2 col-md-3 col-sm-6">

          <div class="card" style="width: 12rem;">
            <img class="card-img-top image-resize4" src="admin/image/<?php echo $row['image']; ?> "
alt="Card image cap">

            <div class="card-body">
              <h5 class="card-title"><?php echo $row['movie_name'];?></h5>
              <p class="card-text">Director: <?php echo $row['directer'];?></p>
            </div>
          </div>
```

```
        </div> </div>
</div>
</div>
<?php
  include("footer.php");
  ?>
  <!-- Js Plugins -->
   <script src="js/jquery-3.3.1.min.js"></script>
   <script src="js/bootstrap.min.js"></script>
   <script src="js/jquery.nice-select.min.js"></script>
   <script src="js/jquery.nicescroll.min.js"></script>
   <script src="js/jquery.magnific-popup.min.js"></script>
   <script src="js/jquery.countdown.min.js"></script>
   <script src="js/jquery.slicknav.js"></script>
   <script src="js/mixitup.min.js"></script>
   <script src="js/owl.carousel.min.js"></script>
   <script src="js/main.js"></script>
</body>
</html>
```

- Login.php

```
<?php
include "Database.php";
session_start();
if($_POST['username'] == '' || $_POST['password'] == ''){
  foreach ($_POST as $key => $value) {
     echo "<li>Please Enter ".$key.".</li>";
  }
  exit();
}
$uname = mysqli_real_escape_string($conn,$_POST['username']);
$password = mysqli_real_escape_string($conn,$_POST['password']);

$sql_query = "SELECT count(*) as cntUser FROM user WHERE username='".$uname."' and
password='".$password."'";
$result = mysqli_query($conn,$sql_query);
$row = mysqli_fetch_array($result);
$count = $row['cntUser'];
if($row){
    if($count > 0){
       $_SESSION['uname'] = $uname;
       echo 1;
   }
}else{
   echo "<li>Invlid Username or password.</li>";
   exit();
}
     ?>
```

- Database.php

```
<?php
$servername='localhost';
$username='root';
$password='';
$dbname = "moviebook";
$conn=mysqli_connect($servername,$username,$password,"$dbname");
if(!$conn){
  die('Could not Connect My Sql:' .mysqli_error());
}
?>
```

- AllMovies.php

```
<?php
session_start();
//index.php

include('database_connection.php');

?>
<!DOCTYPE html>
<html lang="zxx">

<head>
    <meta charset="UTF-8">
    <meta name="description" content="Male_Fashion Template">
    <meta name="keywords" content="Male_Fashion, unica, creative, html">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>All movie page</title>

    <!-- Google Font -->
    <link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans:wght@300;400;600;700;800;900&displa
y=swap"
    rel="stylesheet">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css">
    <link href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-
free@5.15.3/css/fontawesome.min.css">


    <!-- Css Styles -->
    <link rel="stylesheet" href="css/bootstrap.min.css" type="text/css">
```

```
    <link rel="stylesheet" href="css/font-awesome.min.css" type="text/css">
    <link rel="stylesheet" href="css/elegant-icons.css" type="text/css">
    <link rel="stylesheet" href="css/magnific-popup.css" type="text/css">
    <link rel="stylesheet" href="css/nice-select.css" type="text/css">
    <link rel="stylesheet" href="css/owl.carousel.min.css" type="text/css">
    <link rel="stylesheet" href="css/slicknav.min.css" type="  text/css">
    <link rel="stylesheet" href="css/style.css" type="text/css">

</head>

<body style="background: linear-gradient(90deg, #274060 0%,#bbbbff  35%, #ffeeff 100%);">

    <?php
    include("header.php");
    ?>
    <!-- Page Content -->
    <div class="container">
       <div class="row">

          <div class="col-md-3">

                                    <div class="list-group">
                                        <h3>categroy</h3>
                <?php

                $query = "
                SELECT DISTINCT(categroy) FROM add_movie WHERE status = '1' ORDER BY
categroy DESC
                ";
                $statement = $connect->prepare($query);
                $statement->execute();
                $result = $statement->fetchAll();
                foreach($result as $row)
                {
                ?>
                <div class="list-group-item checkbox" style="border-radius:10px">
                   <label><input type="checkbox" class="common_selector categroy" value="<?php echo
$row['categroy']; ?>" > <?php echo $row['categroy']; ?></label>
                </div>
                <?php
                }

                ?>
             </div>

                                    <div class="list-group" >
                                        <h3> language</h3>
                                        <?php
                $query = "
                SELECT DISTINCT(language) FROM add_movie WHERE status = '1' ORDER BY
language DESC
```

```
                ";
                $statement = $connect->prepare($query);
                $statement->execute();
                $result = $statement->fetchAll();
                foreach($result as $row)
                {
                ?>
                <div class="list-group-item checkbox" style="border-radius:10px">
                   <label><input type="checkbox" class="common_selector language" value="<?php echo
$row['language']; ?>"  > <?php echo $row['language']; ?></label>
                </div>
                <?php
                }
                ?>
              </div>
            </div>

            <div class="col-md-9">
                    <br />
               <div class="row filter_data">

               </div>
            </div>
         </div>

      </div>
      <?php

      include("footer.php");
      ?>
<style>

</style>

      <!-- Js Plugins -->
      <script src="js/jquery-3.3.1.min.js"></script>
      <script src="js/bootstrap.min.js"></script>
      <script src="js/jquery.nice-select.min.js"></script>
      <script src="js/jquery.nicescroll.min.js"></script>
      <script src="js/jquery.magnific-popup.min.js"></script>
      <script src="js/jquery.countdown.min.js"></script>
      <script src="js/jquery.slicknav.js"></script>
      <script src="js/mixitup.min.js"></script>

      <script src="js/main.js"></script>
      <script src="js/jquery-1.10.2.min.js"></script>
      <script src="js/jquery-ui.js"></script>

<script>
$(document).ready(function(){
```

```
    filter_data();

    function filter_data()
    {
        $('.filter_data').html('<div id="loading" style="" ></div>');
        var action = 'fetch_data';
        var directer = get_filter('directer');
        var categroy = get_filter('categroy');
        var language = get_filter('language');
        $.ajax({
            url:"allmovie_fetch.php",
            method:"POST",
            data:{action:action, directer:directer, categroy:categroy, language:language},
            success:function(data){
                $('.filter_data').html(data);
            }
        });
    }

    function get_filter(class_name)
    {
        var filter = [];
        $('.'+class_name+':checked').each(function(){
            filter.push($(this).val());
        });
        return filter;
    }

    $('.common_selector').click(function(){
        filter_data();
    });

    $('#show_range').slider({
        range:true,
        min:1000,
        max:65000,
        values:[1000, 65000],
        step:500,
        stop:function(event, ui)
        {
            $('#show_show').html(ui.values[0] + ' - ' + ui.values[1]);
            filter_data();
        }
    });

});
</script>

</body>

</html>
```

- Register.php

```
<!DOCTYPE html>
<!-- Designined by CodingLab - youtube.com/codinglabyt -->
<html lang="en" dir="ltr">
  <head>
    <meta charset="UTF-8">
    <title> Responsive Registration Form | CodingLab </title>
    <link rel="stylesheet" href="css/register.css">
    <script src="js/jquery.min.js"></script>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
     <meta name="viewport" content="width=device-width, initial-scale=1.0">
   </head>
<body style="background: linear-gradient(90deg, #274060 0%,#bbbbff  35%, #ffeeff
100%);"><br><br><br>

  <div class="container" >
    <center><a href="./index.php"><img src="img/logo.png" alt="" style="margin-top: 80px; width:
50%;"></a></center>
    <div class="title">Registration</div>
    <div class="content">
      <form id="form" action="register.php" method="post" enctype="multipart/form-data"
onsubmit="return validate();">
        <div class="user-details">
          <div class="input-box">
            <span class="details">UserName</span>
            <input type="text" id="username" name="username" placeholder="Enter your name">
            <p id="nameerror"></p>
          </div>
          <div class="input-box">
            <span class="details">Email</span>
            <input type="text" id="email" name="email" placeholder="Enter your Email">
            <p id="emailerror"></p>
          </div>
          <div class="input-box">
            <span class="details">Phone Number</span>
            <input type="text" id="number" name="number" placeholder="Enter your Phone Number">
            <p id="numbererror"></p>
          </div>
          <div class="input-box">
            <span class="details">City</span>
            <input type="text" id="city" name="city" placeholder="Enter your City">
            <p id="cityerror"></p>
          </div>
```

```
        <div class="input-box">
          <span class="details">Password</span>
          <input type="password" id="password" name="password" placeholder="Enter your
password">
          <p id="passworderror"></p>
        </div>
        <div class="input-box">
          <span class="details">Confirm Password</span>
          <input type="password" id="cpassword" name="cpassword" placeholder="Confirm your
password">
          <p id="cpasswworderror"></p>
        </div>
        <div class="input-box">
          <span class="details">Image uploaded (Option)</span>
          <input type="file" id="image" name="image">
        </div>
       </div>
       <p id="error_para" ></p>
       <div id="err"></div>
       <div class="button">
         <input type="submit" value="Register" id="submit" name="submit">
       </div>
     </form>
   </div>
  </div>
<script type="text/javascript">
  function validate()
{
 var error="";
 var name = document.getElementById( "username" );
 var email = document.getElementById( "email" );
 var number = document.getElementById( "number" );
 var city = document.getElementById( "city" );
 var password = document.getElementById( "password" );
 var cpassword = document.getElementById( "cpassword" );

 if( name.value == "" )
 {
  error = " <font color='red'>!Requrie Name.</font> ";
  document.getElementById( "nameerror" ).innerHTML = error;
  return false;
 }
 if(name.value.length <= 2)
{
   error = " <font color='red'>!please not allow 2 and 20 chaecter</font> ";

  document.getElementById( "nameerror" ).innerHTML = error;
  return false;

}
if(!isNaN(name.value))
```

```
{
   error = " <font color='red'>!please only charecter allowed</font> ";

  document.getElementById( "nameerror" ).innerHTML = error;
  return false;

}
```

```
 else if( email.value == "")
 {
  error = " <font color='red'>!Requrie Email.</font> ";
  document.getElementById( "emailerror" ).innerHTML = error;
  return false;
 }
 else if( email.value.indexOf('@') <= 0 )
 {
  error = " <font color='red'>! ** @ invail position</font> ";
  document.getElementById( "emailerror" ).innerHTML = error;
  return false;
 }

 else if ((email.value.charAt(email.value.length-4)!='.') && (email.value.charAt(email.value.length-
3)!='.'))
 {
  error = " <font color='red'>! ** . invaild position</font> ";
  document.getElementById( "emailerror" ).innerHTML = error;
  return false;
 }
```

```
 else if( number.value == "")
 {
  error = " <font color='red'>!Requrie Name.</font> ";
  document.getElementById( "numbererror" ).innerHTML = error;
  return false;
 }
else if( number.value.length!=10)
 {
  error = " <font color='red'>! ** mobile number must 10 digit</font> ";
  document.getElementById( "numbererror" ).innerHTML = error;
  return false;
 }
```

```
else if(isNaN(number.value)){
  error = " <font color='red'>! ** mobile number must be not allow charecter</font> ";
  document.getElementById( "numbererror" ).innerHTML = error;
  return false;
}




 else if( city.value == "" )
 {
  error = " <font color='red'>!Requrie Name.</font> ";
  document.getElementById( "cityerror" ).innerHTML = error;
  return false;
 }

 else if( password.value == "" )
 {
  error = " <font color='red'>!Requrie Name.</font> ";
  document.getElementById( "passworderror" ).innerHTML = error;
  return false;
 }

  if(password.value.length <= 2)
{
   error = " <font color='red'>!not allow 2 and 10 chaecter</font> ";

  document.getElementById( "passworderror" ).innerHTML = error;
  return false;

}
  if(password.value.length >= 10)
{
   error = " <font color='red'>!not allow 2 and 10 chaecter</font> ";

  document.getElementById( "passworderror" ).innerHTML = error;
  return false;

}


else if( cpassword.value == "" )
 {
  error = " <font color='red'>!Requrie Name.</font> ";
  document.getElementById( "cpassworderror" ).innerHTML = error;
  return false;
 }

else if( cpassword.value != password.value)
 {
```

```
 error = " <font color='red'>!Conform Password Not Match.</font> ";
 document.getElementById( "cpassworderror" ).innerHTML = error;
 return false;
 }

 else
 {
 return true;
 }
}
</script>
</body>
</html>
```

- Admin_Index.php

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">
    <meta name="generator" content="Jekyll v3.8.5">
    <title>Dashboard Page</title>

<?php
session_start();
if (!isset($_SESSION['admin'])) {
  header("location:login.php");
}

include "./templates/top.php";

?>

<?php include "./templates/navbar.php"; ?>

<div class="container-fluid">
  <div class="row">

    <?php include "./templates/sidebar.php"; ?>

      <!-- <canvas class="my-4 w-100" id="myChart" width="900" height="380"></canvas> -->

      <h2>Total Admins</h2>
      <div class="table-responsive">
```

```
      <table class="table table-striped table-sm">
        <thead>
          <tr>
            <th>#</th>
            <th>Name</th>
            <th>Email</th>
            <th>Status</th>

          </tr>
        </thead>
        <?php
include_once 'Database.php';
$result = mysqli_query($conn,"SELECT * FROM admin");

if (mysqli_num_rows($result) > 0) {
  while($row = mysqli_fetch_array($result)) {
    ?>
    <tr><td><?php echo $row['id'];?></td>
        <td><?php echo $row['name'];?></td>
        <td><?php echo $row['email'];?></td>

        <td><?php echo $row['is_active'];?></td>

          </tr>
  <?php
  }
}
?>


      </table>
    </div>
   </main>
  </div>
</div>

<?php include "./templates/footer.php"; ?>

<script type="text/javascript" src="./js/admin.js"></script>
```

- Admin_login.php

```
<!DOCTYPE html>
<html>
<head>
  <title>Username and password validation in PHP using AJAX</title>
  <link href="css/bootstrap.min.css" rel="stylesheet">
  <script src="js/jquery-3.5.1.min.js"></script>
  <script type="text/javascript" src="ajaxValidation.js"></script>
  <style type="text/css">
    li{color: red;}
  </style>
</head>
<body>
  <div class="container col-md-5">
    <div class="mb-3">
      <label class="form-label">Name</label>
      <input type="email" class="form-control" id="userEmail">
    </div>
    <div class="mb-3">
      <label class="col-sm-2 col-form-label">Password</label>
      <input type="password" class="form-control" id="userPassword">
    </div>
    <p id="message"></p>
    <div class="mb-3 col-md-4">
      <button class="form-control btn btn-danger" id="checkValidation">Login</button>
    </div>
  </div>
</body>
</html>
```

- Admin_Logout.php

```
<?php

session_start();

if (isset($_SESSION["admin"])) {
        session_destroy();
        header("location:login.php");
}else{
        header("location:about.php");
}


?>
```

Chapter 7
Scope & Conclusion

7.1 Scope

Today the need of simplicity has driven application software programming to a new level. This project is a transaction related information storing project which will be used by the various multiplexes for online movie ticket booking through internet. Customers can view all currently running movies and book their tickets for any specific date and show also customer can pay online through credit card. This application has a user friendly interface so that the customer and administrator can easily and efficiently use the software and its features.

The main purpose of this project is to provide a reliable, secure, efficient and user friendly environment to the customers and management authorities. Also benefit to the customer with efficient and faster service.

The Project “Online Movie Ticket Booking System” as a wide scope as it is generalized software and can be easily used in any ticket booking process system with little or no change. The Changes in software can be easily accommodated. The addition and deletion of the modules in software can be easily adjusted. This project has a lot of scope for further enhancement too. This project can save money and efforts in managing the record, just a mouse click can make the task easy and faster.

7.2 Conclusion

The entire project has been developed and deployed as per the requirements stated by the user, it is found to be bug free as per the testing standards that is implemented. Any specification- untraced errors will be concentrated in the coming versions, which are planned to be developed in near future.

The system at present does not take care of the money payment methods, as the consolidated constructs need SSL standards and are critically to be initiated in the first face; the application of the credit card transactions is applied as a developmental phase in the coming days. The system needs more elaborative technicality for its inception and evolution.

7.3 References

1. Books

1. "Php: The Good Parts" by Douglas Crockford - This book is ideal for beginners and focuses on the most essential parts of the Php language.

2. "Eloquent Php: A Modern Introduction to Programming" by Marijn Haverbeke -This book is a great introduction to programming with Php, with an emphasis on practical examples and exercises.

3. "You Don't Know Php" by Kyle Simpson - This book series covers all aspects of JavaScript,from basic syntax to more advanced concepts like closures and prototypes.

4. "Php: The Definitive Guide" by David Flanagan - This book is a comprehensive reference guide to JavaScript, with in-depth coverage of the language's syntax and features.

2. Websites

1. FreeCodeCamp (https://www.freecodecamp.org/) - FreeCodeCamp offers a comprehensive curriculum for web development, including courses on php.

2. Codecademy (https://www.codecademy.com/) - Codecademy offers interactive courses on a variety of programming languages, including php.

3. W3Schools (https://www.w3schools.com/) - W3Schools is a popular online resource for web development tutorials, including free courses on Php.

4. Mozilla Developer Network (https://developer.mozilla.org/en-US/docs/Web/JavaScript) - Mozilla Developer Network offers extensive documentation on php, including tutorials, guides, and reference material.

5. React Official Documentation (https://reactjs.org/docs/getting-started.html) - The official React documentation provides a comprehensive guide to getting started with React, including tutorials and example code.