



Sri Eshwar
College of Engineering
An Autonomous Institution
Affiliated to Anna University, Chennai



ONLINE MOVIE TICKET BOOKING SYSTEM

PYTHON PROJECT REPORT

Submitted by

DINESH KUMAR I.

[2303722813421013]

BACHELOR OF ENGINEERING

in

COMPUTER AND COMMUNICATION ENGINEERING

SRI ESHWAR COLLEGE OF ENGINEERING

(AN AUTONOMOUS INSTITUTION)

COIMBATORE – 641 202

APRIL - MAY 2024

BONAFIDE CERTIFICATE

Certified that this project report “**ONLINE MOVIE TICKET BOOKING SYSTEM**” is the bonafide work of

DINESH KUMAR I. **[2303722813421013]**

who carried out the project work under my supervision

.....

SIGNATURE

Dr. C. GANESH , M.E.,Ph.D

SUPERVISOR

Assistant Professor,

Dept. of Computer &

Communication Engineering,

Sri Eshwar College of Engineering,

Coimbatore-641202.

TABLE OF CONTENTS

| S.NO. | TITLE | PAGE |
|--------------|--------------------------------------|-------------|
| 1. | Abstract | 01 |
| 2. | Introduction | 01 |
| 3. | Hardware and Software Specifications | 01 |
| 4. | Architecture | 02 |
| 5. | Project Code | 03 |
| 6. | Result-Snap Shots | 16 |
| 7. | Result and Discussion | 20 |

Online Movie Ticket Booking System

1. Abstract:

In this project we book ticket using “Online Movie Ticket Reservation System”. We enter into Web page by logging with User Name and Password. Then we select the Movie and later in which Theatre movie is running.

Later choose Show Timings and enter no of tickets you want .Finally it displays the details of the procedure and print the form to show at respective ticket counter to get ticket.

2. Introduction:

The main purpose of our online ticket booking system is to provide an alternate and convenient way for a customer to buy cinema 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.

3. Hardware and Software specifications:

3.1 Software Requirements:

Operating System : Windows

Client side Technology : Python

Database : MongoDB

3.2 Hardware Requirements:

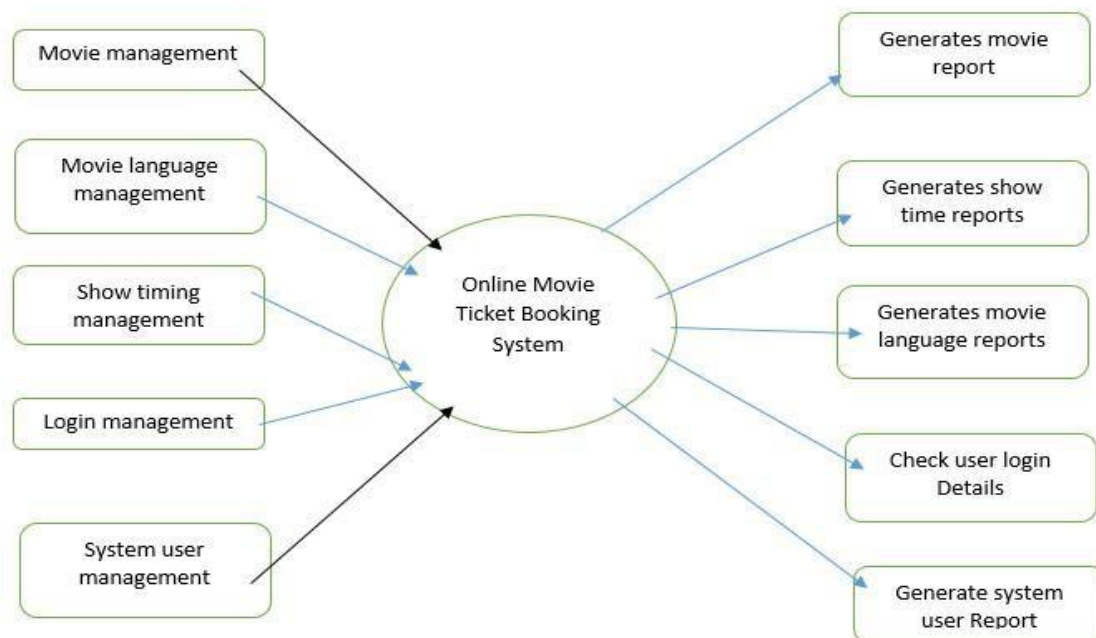
Intel Pentium

Processor speed-1.2 GHz or above

1 GB RAM minimum

1024 GB Hard Disk Space

4. Architecture – DFD:



4.1 Modules:

- User login module
- Selection of Movie module
- Selection of show time module
- Selecting Theaters module
- Number of Tickets booking module
- Display booked Tickets with Amount

5. Code for the Project

5.1 index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ticket Booking System</title>

  <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
</head>
<body>
  <div class="container">
    <h1>Ticket Booking System</h1>
    {% if not session.get('logged_in') %}
      <div class="login-container">
        
        <h2 class="login-title">Login</h2>
        <form action="/login" method="post" class="login-form">
          <input type="text" id="username" name="username" placeholder="Username" required>
          <input type="password" id="password" name="password" placeholder="Password" required>
          <button type="submit">Login</button>
        </form>
      </div>
    {% else %}
      <div class="movie-container">
        <h2>Available Movies</h2>
        <div class="movie-images">
          
          
          
          
          
        </div>
        <h2>Your Booked Tickets</h2>
        <ul>
          {% for ticket in booked_tickets %}
            <li>{{ ticket }}</li>
          {% endfor %}
        </ul>
        <a href="/logout">Logout</a>
      </div>
    {% endif %}
  </div>
</body>
</html>
```

5.2 movies.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Available Movies</title>
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@6.0.0-beta3/css/all.min.css">
</head>
<body>
  <div class="container">
    <h1>Available Movies</h1>
    <div class="movie-container">
      <div class="holder">
        <div class="hol">
          
          <a href="/book1"><button class="btn">Book Now</button></a>
        </div>
        <div class="context"><p>Rating:9.7/10.0</p></div>
        <div class="star-rating">
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
        </div>
      </div>
      <div class="holder">
        <div class="hol">
          
          <a href="/book2"><button class="btn">Book Now</button></a>
        </div>
        <div class="context"><p>Rating:9.5/10.0</p></div>
        <div class="star-rating">
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
        </div>
      </div>
      <div class="holder">
        <div class="hol">
          
          <a href="/book3"><button class="btn">Book Now</button></a>
        </div>
        <div class="context"><p>Rating:9.8/10.0</p></div>
        <div class="star-rating">
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
        </div>
      </div>
      <div class="holder">
        <div class="hol">
          
          <a href="/book4"><button class="btn">Book Now</button></a>
        </div>
        <div class="context"><p>Rating:9.6/10.0</p></div>
        <div class="star-rating">
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
          <i class="fas fa-star filled"></i>
        </div>
      </div>
    </div>
    <a href="/logout" class="log">Logout</a>
  </div>
</body>
</html>
```

5.3 book1.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Seat Booking</title>
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style2.css') }}">
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style1.css') }}">
</head>
<body>
  <div class="container">
    <div class="name"><h1>PT SIR - Tamil</h1></div>
    <div class="content-wrapper">
      <div class="img1">
        
      </div>
      <div class="cont">
        <div class="abt">
          <h3 class="des">About The Movie</h3>
        </div>
        <div class="des">
          <p class="abt">PT Sir is a Tamil movie starring Hiphop Tamizha, Kashmira Pardeshi, Anikha Surendran, and Munishkanth in prominent roles. It is written and directed by Karthik Venugopalan.
          </p>
          <button class="btn">2D</button>
          <button class="btn">3D</button>
          <button class="btn">Romance</button>
        </div>
      </div>
    </div>
  </div>
  <div class="center">
    <div class="tickets">
      <div class="ticket-selector">
        <div class="seats">
          <div class="status">
            <div class="item">Available</div>
            <div class="item">Booked</div>
            <div class="item">Selected</div>
          </div>
          <div class="all-seats">
            <!-- Seats will be dynamically loaded here -->
          </div>
        </div>
        <div class="timings">
          <div class="dates">
            <input type="radio" name="date" id="d1" value="11" checked />
            <label for="d1" class="dates-item">
              <div class="day">Sun</div>
              <div class="date">11</div>
            </label>
            <input type="radio" name="date" id="d2" value="12" />
            <label for="d2" class="dates-item">
              <div class="day">Mon</div>
              <div class="date">12</div>
            </label>
            <input type="radio" name="date" id="d3" value="13" />
            <label for="d3" class="dates-item">
              <div class="day">Tue</div>
              <div class="date">13</div>
            </label>
            <input type="radio" name="date" id="d4" value="14" />
            <label for="d4" class="dates-item">
              <div class="day">Wed</div>
              <div class="date">14</div>
            </label>
          </div>
          <div class="times">
            <input type="radio" name="time" id="t1" value="11:00" checked />
            <label for="t1" class="time">11:00</label>
            <input type="radio" name="time" id="t2" value="02:00" />
            <label for="t2" class="time">02:00</label>
            <input type="radio" name="time" id="t3" value="06:00" />
            <label for="t3" class="time">06:00</label>
            <input type="radio" name="time" id="t4" value="10:00" />
            <label for="t4" class="time">10:00</label>
          </div>
        </div>
        <div class="price">
          <div class="total">
            <span><span class="count">0</span> Tickets </span>
            <div class="amount">0</div>
          </div>
          <button class="btn" type="button" onclick="bookTickets()">Book</button>
        </div>
      </div>
    </div>
  </div>
</body>
</html>

<script>
document.addEventListener('DOMContentLoaded', function() {
  let seatsContainer = document.querySelector('.all-seats');
  let dateRadios = document.querySelectorAll("input[name='date']");
  let timeRadios = document.querySelectorAll("input[name='time']");

  dateRadios.forEach(radio => radio.addEventListener('change', fetchSeats));
  timeRadios.forEach(radio => radio.addEventListener('change', fetchSeats));

  fetchSeats();

  function fetchSeats() {
    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch(`/seats?date=${selectedDate}&time=${selectedTime}`)
      .then(response => response.json())
      .then(data => {
        seatsContainer.innerHTML = ''; // Clear previous seats
        for (let i = 1; i <= 99; i++) {
          let seatStatus = data[i];
          let booked = seatStatus === 'booked' ? 'booked' : '';
          seatsContainer.insertAdjacentHTML(

```



```

        'beforeend',
        `<input type="checkbox" name="tickets" id="s${i}" ${booked ? 'disabled' : ''}/><label for="s${i}" class="seat ${booked}"></label>`
    );
    }
    attachSeatListeners();
  });
}

function attachSeatListeners() {
  let tickets = seatsContainer.querySelectorAll("input");
  tickets.forEach((ticket) => {
    ticket.addEventListener("change", () => {
      let amount = document.querySelector(".amount").innerHTML;
      let count = document.querySelector(".count").innerHTML;
      amount = Number(amount);
      count = Number(count);

      if (ticket.checked) {
        count += 1;
        amount += 200;
      } else {
        count -= 1;
        amount -= 200;
      }
      document.querySelector(".amount").innerHTML = amount;
      document.querySelector(".count").innerHTML = count;
    });
  });
}

function bookTickets() {
  let selectedSeats = [];
  let tickets = document.querySelectorAll(".all-seats input:checked");
  tickets.forEach(ticket => {
    selectedSeats.push(parseInt(ticket.id.slice(1)));
  });

  let selectedDate = document.querySelector("input[name='date']:checked").value;
  let selectedTime = document.querySelector("input[name='time']:checked").value;

  fetch('/book_ticket', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
    body: JSON.stringify({ selected_seats: selectedSeats, date: selectedDate, time: selectedTime })
  })
  .then(response => response.json())
  .then(data => {
    if (data.status === 'success') {
      alert('Seats booked successfully!');
      document.querySelector(".amount").innerHTML = '0';
      document.querySelector(".count").innerHTML = '0';
      fetchSeats(); // Refresh seat availability
    } else {
      alert('Failed to book seats. Please try again.');
```

5.4 book2.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Seat Booking</title>
  <link rel="stylesheet" href="{ url_for('static', filename='css/style2.css') }">
  <link rel="stylesheet" href="{ url_for('static', filename='css/style1.css') }">
</head>
<body>
  <div class="container">
    <div class="name"><h1>STAR - Tamil</h1></div>
    <div class="content-wrapper">
      <div class="img1">
        
      </div>
      <div class="cont">
        <div class="abt">
          <h3 class="des">About The Movie</h3>
        </div>
        <div class="des">
          <p class="abt">
            The story unfolds as a beautiful ode to the bond between Mr. Pandian, a photographer, and his son, an actor. The father and son find solace, fulfillment, and redemption in their symbiotic relationship.
          </p>
          <button class="btn">2D</button>
          <button class="btn">Tamil</button>
          <button class="btn">Family</button>
        </div>
      </div>
    </div>
  </div>
  <br>
  <div class="center">
    <div class="tickets">
      <div class="ticket-selector">
        <div class="seats">
          <div class="status">
            <div class="item">Available</div>
            <div class="item">Booked</div>
            <div class="item">Selected</div>
          </div>
          <div class="all-seats">
            <!-- Seats will be dynamically loaded here -->
          </div>
        </div>
        <div class="timings">
          <div class="dates">
            <input type="radio" name="date" id="d1" value="11" checked />
            <label for="d1" class="dates-item">
              <div class="day">Sun</div>
              <div class="date">11</div>
            </label>
            <input type="radio" name="date" id="d2" value="12" />
            <label for="d2" class="dates-item">
              <div class="day">Mon</div>
              <div class="date">12</div>
            </label>
            <input type="radio" name="date" id="d3" value="13" />
            <label for="d3" class="dates-item">
              <div class="day">Tue</div>
              <div class="date">13</div>
            </label>
            <input type="radio" name="date" id="d4" value="14" />
            <label for="d4" class="dates-item">
              <div class="day">Wed</div>
              <div class="date">14</div>
            </label>
          </div>
          <div class="times">
            <input type="radio" name="time" id="t1" value="11:00" checked />
            <label for="t1" class="time">11:00</label>
            <input type="radio" name="time" id="t2" value="02:00" />
            <label for="t2" class="time">02:00</label>
            <input type="radio" name="time" id="t3" value="06:00" />
            <label for="t3" class="time">06:00</label>
            <input type="radio" name="time" id="t4" value="10:00" />
            <label for="t4" class="time">10:00</label>
          </div>
        </div>
        <div class="price">
          <div class="total">
            <span><span class="count">0</span> Tickets </span>
            <div class="amount">0</div>
          </div>
          <button class="btn" type="button" onclick="bookTickets()">Book</button>
        </div>
      </div>
    </div>
  </div>
</body>
</html>

<script>
document.addEventListener('DOMContentLoaded', function() {
  let seatsContainer = document.querySelector('.all-seats');
  let dateRadios = document.querySelectorAll('input[name="date"]');
  let timeRadios = document.querySelectorAll('input[name="time"]');

  dateRadios.forEach(r => r.addEventListener('change', fetchSeats));
  timeRadios.forEach(r => r.addEventListener('change', fetchSeats));

  fetchSeats();

  function fetchSeats() {
    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch(`/seats?date=${selectedDate}&time=${selectedTime}`)
      .then(response => response.json())
      .then(data => {
        seatsContainer.innerHTML = ''; // Clear previous seats
        for (let i = 1; i <= 99; i++) {
          let seatStatus = data[i];
          let booked = seatStatus === 'booked' ? 'booked' : '';
          seatsContainer.insertAdjacentHTML(

```

```

        'beforeend',
        `<input type="checkbox" name="tickets" id="s${i}" ${booked ? 'disabled' : ''}/><label for="s${i}" class="seat ${booked}"></label>`
    );
    }
    attachSeatListeners();
    });
}

function attachSeatListeners() {
    let tickets = seatsContainer.querySelectorAll("input");
    tickets.forEach((ticket) => {
        ticket.addEventListener("change", () => {
            let amount = document.querySelector(".amount").innerHTML;
            let count = document.querySelector(".count").innerHTML;
            amount = Number(amount);
            count = Number(count);

            if (ticket.checked) {
                count += 1;
                amount += 200;
            } else {
                count -= 1;
                amount -= 200;
            }

            document.querySelector(".amount").innerHTML = amount;
            document.querySelector(".count").innerHTML = count;
        });
    });
}

function bookTickets() {
    let selectedSeats = [];
    let tickets = document.querySelectorAll(".all-seats input:checked");
    tickets.forEach(ticket => {
        selectedSeats.push(parseInt(ticket.id.slice(1)));
    });

    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch('/book_ticket', {
        method: 'POST',
        headers: {
            'Content-Type': 'application/json'
        },
        body: JSON.stringify({ selected_seats: selectedSeats, date: selectedDate, time: selectedTime })
    })
    .then(response => response.json())
    .then(data => {
        if (data.status === 'success') {
            alert('Seats booked successfully!');
            document.querySelector(".amount").innerHTML = '0';
            document.querySelector(".count").innerHTML = '0';
            fetchSeats(); // Refresh seat availability
        } else {
            alert('Failed to book seats. Please try again.');
```

5.5 book3.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Seat Booking</title>
  <link rel="stylesheet" href="{ url_for('static', filename='css/style2.css') }">
  <link rel="stylesheet" href="{ url_for('static', filename='css/style1.css') }">
</head>
<body>
  <div class="container">
    <div class="name"><h1>G.O.A.T - Tamil</h1></div>
    <div class="content-wrapper">
      <div class="img1">
        
      </div>
      <div class="cont">
        <div class="abt">
          <h3 class="des">About The Movie</h3>
        </div>
        <div class="des">
          <p class="abt">
            The Greatest of All Time (also known and marketed as The GOAT) is an upcoming Indian Tamil-language science fiction action film directed by Venkat Prabhu and produced by AGS Entertainment.
          </p>
          <button class="btn">2D</button>
          <button class="btn">Tamil</button>
          <button class="btn">Action</button>
        </div>
      </div>
    </div>
  </div>
  <br>
  <div class="center">
    <div class="tickets">
      <div class="ticket-selector">
        <div class="seats">
          <div class="status">
            <div class="item">Available</div>
            <div class="item">Booked</div>
            <div class="item">Selected</div>
          </div>
          <div class="all-seats">
            <!-- Seats will be dynamically loaded here -->
          </div>
        </div>
        <div class="timings">
          <div class="dates">
            <input type="radio" name="date" id="d1" value="11" checked />
            <label for="d1" class="dates-item">
              <div class="day">Sun</div>
              <div class="date">11</div>
            </label>
            <input type="radio" name="date" id="d2" value="12" />
            <label for="d2" class="dates-item">
              <div class="day">Mon</div>
              <div class="date">12</div>
            </label>
            <input type="radio" name="date" id="d3" value="13" />
            <label for="d3" class="dates-item">
              <div class="day">Tue</div>
              <div class="date">13</div>
            </label>
            <input type="radio" name="date" id="d4" value="14" />
            <label for="d4" class="dates-item">
              <div class="day">Wed</div>
              <div class="date">14</div>
            </label>
          </div>
          <div class="times">
            <input type="radio" name="time" id="t1" value="11:00" checked />
            <label for="t1" class="time">11:00</label>
            <input type="radio" name="time" id="t2" value="02:00" />
            <label for="t2" class="time">02:00</label>
            <input type="radio" name="time" id="t3" value="06:00" />
            <label for="t3" class="time">06:00</label>
            <input type="radio" name="time" id="t4" value="10:00" />
            <label for="t4" class="time">10:00</label>
          </div>
        </div>
        <div class="price">
          <div class="total">
            <span><span class="count">0</span> Tickets </span>
            <div class="amount">0</div>
          </div>
          <button class="btn" type="button" onClick="bookTickets()">Book</button>
        </div>
      </div>
    </div>
  </div>
</body>
<script>
document.addEventListener('DOMContentLoaded', function() {
  let seatsContainer = document.querySelector('.all-seats');
  let dateRadios = document.querySelectorAll("input[name='date']");
  let timeRadios = document.querySelectorAll("input[name='time']");

  dateRadios.forEach(radio => radio.addEventListener('change', fetchSeats));
  timeRadios.forEach(radio => radio.addEventListener('change', fetchSeats));

  fetchSeats();

  function fetchSeats() {
    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch(`/seats?date=${selectedDate}&time=${selectedTime}`)
      .then(response => response.json())
      .then(data => {
        seatsContainer.innerHTML = ''; // Clear previous seats
        for (let i = 1; i <= 90; i++) {
          let seatStatus = data[i];
          let booked = seatStatus === 'booked' ? 'booked' : '';
          seatsContainer.insertAdjacentHTML(
```

```

        'beforeend',
        `<input type="checkbox" name="tickets" id="s${i}" ${booked ? 'disabled' : ''}/><label for="s${i}" class="seat ${booked}"></label>`
    );
}
attachSeatListeners();
});
}

function attachSeatListeners() {
    let tickets = seatsContainer.querySelectorAll("input");
    tickets.forEach((ticket) => {
        ticket.addEventListener("change", () => {
            let amount = document.querySelector(".amount").innerHTML;
            let count = document.querySelector(".count").innerHTML;
            amount = Number(amount);
            count = Number(count);

            if (ticket.checked) {
                count += 1;
                amount += 200;
            } else {
                count -= 1;
                amount -= 200;
            }
            document.querySelector(".amount").innerHTML = amount;
            document.querySelector(".count").innerHTML = count;
        });
    });
}

function bookTickets() {
    let selectedSeats = [];
    let tickets = document.querySelectorAll(".all-seats input:checked");
    tickets.forEach(ticket => {
        selectedSeats.push(parseInt(ticket.id.slice(1)));
    });

    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch('/book_ticket', {
        method: 'POST',
        headers: {
            'Content-Type': 'application/json'
        },
        body: JSON.stringify({ selected_seats: selectedSeats, date: selectedDate, time: selectedTime })
    })
    .then(response => response.json())
    .then(data => {
        if (data.status === 'success') {
            alert('Seats booked successfully!');
            document.querySelector(".amount").innerHTML = '0';
            document.querySelector(".count").innerHTML = '0';
            fetchSeats(); // Refresh seat availability
        } else {
            alert('Failed to book seats. Please try again.');
```

5.6 book4.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Seat Booking</title>
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style2.css') }}">
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style1.css') }}">
</head>
<body>
  <div class="container">
    <div class="name"><h1>Ponniyin Selvan - Tamil</h1></div>
    <div class="content-wrapper">
      <div class="img1">
        
      </div>
      <div class="cont">
        <div class="abt">
          <h3 class="des">About The Movie</h3>
        </div>
        <div class="des">
          <p class="abt">
            Vandiyathevan, a brave and brilliant young man sets out to cross the Chola land to deliver a message from the Crown Prince Adithya Karikalan.
          </p>
          <button class="btn">2D</button>
          <button class="btn">Tamil</button>
          <button class="btn">Historical</button>
        </div>
        <br>
      </div>
    </div>
  </div>
  <br>
  <div class="center">
    <div class="tickets">
      <div class="ticket-selector">
        <div class="seats">
          <div class="status">
            <div class="item">Available</div>
            <div class="item">Booked</div>
            <div class="item">Selected</div>
          </div>
          <div class="all-seats">
            <!-- Seats will be dynamically loaded here -->
          </div>
        </div>
        <div class="timings">
          <div class="dates">
            <input type="radio" name="date" id="d1" value="11" checked />
            <label for="d1" class="dates-item">
              <div class="day">Sun</div>
              <div class="date">11</div>
            </label>
            <input type="radio" name="date" id="d2" value="12" />
            <label for="d2" class="dates-item">
              <div class="day">Mon</div>
              <div class="date">12</div>
            </label>
            <input type="radio" name="date" id="d3" value="13" />
            <label for="d3" class="dates-item">
              <div class="day">Tue</div>
              <div class="date">13</div>
            </label>
            <input type="radio" name="date" id="d4" value="14" />
            <label for="d4" class="dates-item">
              <div class="day">Wed</div>
              <div class="date">14</div>
            </label>
          </div>
          <div class="times">
            <input type="radio" name="time" id="t1" value="11:00" checked />
            <label for="t1" class="time">11:00</label>
            <input type="radio" name="time" id="t2" value="02:00" />
            <label for="t2" class="time">02:00</label>
            <input type="radio" name="time" id="t3" value="06:00" />
            <label for="t3" class="time">06:00</label>
            <input type="radio" name="time" id="t4" value="10:00" />
            <label for="t4" class="time">10:00</label>
          </div>
        </div>
        <div class="price">
          <div class="total">
            <span><span class="count">0</span> Tickets </span>
            <div class="amount">0</div>
          </div>
          <button class="btn" type="button" onclick="bookTickets()">Book</button>
        </div>
      </div>
    </div>
  </div>
</body>
<script>
  document.addEventListener('DOMContentLoaded', function() {
    let seatsContainer = document.querySelector('.all-seats');
    let dateRadios = document.querySelectorAll('input[name="date"]');
    let timeRadios = document.querySelectorAll('input[name="time"]');

    dateRadios.forEach(radio => radio.addEventListener('change', fetchSeats));
    timeRadios.forEach(radio => radio.addEventListener('change', fetchSeats));

    fetchSeats();

    function fetchSeats() {
      let selectedDate = document.querySelector("input[name='date']:checked").value;
      let selectedTime = document.querySelector("input[name='time']:checked").value;

      fetch(`/seats?date=${selectedDate}&time=${selectedTime}`)
        .then(response => response.json())
        .then(data => {
          seatsContainer.innerHTML = ''; // Clear previous seats
          for (let i = 1; i <= 99; i++) {
            let seatStatus = data[i];
            let booked = seatStatus === 'booked' ? 'booked' : '';
            seatsContainer.insertAdjacentHTML(
```

```

        'beforeend',
        `<input type="checkbox" name="tickets" id="s${i}" ${booked ? 'disabled' : ''}/><label for="s${i}" class="seat ${booked}"></label>`
    );
    }
    attachSeatListeners();
});
}

function attachSeatListeners() {
    let tickets = seatsContainer.querySelectorAll("input");
    tickets.forEach((ticket) => {
        ticket.addEventListener("change", () => {
            let amount = document.querySelector(".amount").innerHTML;
            let count = document.querySelector(".count").innerHTML;
            amount = Number(amount);
            count = Number(count);

            if (ticket.checked) {
                count += 1;
                amount += 200;
            } else {
                count -= 1;
                amount -= 200;
            }
            document.querySelector(".amount").innerHTML = amount;
            document.querySelector(".count").innerHTML = count;
        });
    });
}

function bookTickets() {
    let selectedSeats = [];
    let tickets = document.querySelectorAll(".all-seats input:checked");
    tickets.forEach(ticket => {
        selectedSeats.push(parseInt(ticket.id.slice(1)));
    });

    let selectedDate = document.querySelector("input[name='date']:checked").value;
    let selectedTime = document.querySelector("input[name='time']:checked").value;

    fetch('/book_ticket', {
        method: 'POST',
        headers: {
            'Content-Type': 'application/json'
        },
        body: JSON.stringify({ selected_seats: selectedSeats, date: selectedDate, time: selectedTime })
    })
    .then(response => response.json())
    .then(data => {
        if (data.status === 'success') {
            alert('Seats booked successfully!');
            document.querySelector(".amount").innerHTML = '0';
            document.querySelector(".count").innerHTML = '0';
            fetchSeats(); // Refresh seat availability
        } else {
            alert('Failed to book seats. Please try again.');
```

5.7 transaction.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Movie Booking - Transaction</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <header>
    <h1>Movie Booking System</h1>
  </header>
  <main>
    <section id="movie-selection">
      <h2>Select Movie</h2>
      <select id="movie">
        <option value="1">Movie 1</option>
        <option value="2">Movie 2</option>
        <option value="3">Movie 3</option>
        <!-- Add more movies as needed -->
      </select>
    </section>
    <section id="seat-selection">
      <h2>Select Seats</h2>
      <div id="seats">
        <!-- Generate seat layout here -->
        <!-- Example layout -->
        <div class="row">
          <span class="seat">A1</span>
          <span class="seat">A2</span>
          <span class="seat">A3</span>
          <span class="seat">A4</span>
        </div>
        <div class="row">
          <span class="seat">B1</span>
          <span class="seat">B2</span>
          <span class="seat">B3</span>
          <span class="seat">B4</span>
        </div>
        <!-- Add more rows as needed -->
      </div>
    </section>
    <section id="payment">
      <h2>Payment Information</h2>
      <form id="payment-form" action="receipt.html" method="GET">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required>
        <label for="card-number">Card Number:</label>
        <input type="text" id="card-number" name="card-number" required>
        <label for="expiry-date">Expiry Date:</label>
        <input type="text" id="expiry-date" name="expiry-date" required>
        <label for="cvv">CVV:</label>
        <input type="text" id="cvv" name="cvv" required>
        <input type="submit" value="Book Now">
      </form>
    </section>
  </main>
  <footer>
    <p>&copy; 2024 Movie Booking System</p>
  </footer>
</body>
</html>
```


5.8 receipt.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Movie Booking - Receipt</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <header>
    <h1>Movie Booking System</h1>
  </header>
  <main>
    <section id="receipt">
      <h2>Receipt</h2>
      <p>Thank you for your purchase, <span id="receipt-name"></span>!</p>
      <p>Movie: <span id="receipt-movie"></span></p>
      <p>Seats: <span id="receipt-seats"></span></p>
      <p>Total Amount: <span id="receipt-amount"></span></p>
    </section>
  </main>
  <footer>
    <p>&copy; 2024 Movie Booking System</p>
  </footer>
  <script>
    // Extract query parameters and display them in the receipt
    const urlParams = new URLSearchParams(window.location.search);
    document.getElementById('receipt-name').innerText = urlParams.get('name');
    document.getElementById('receipt-movie').innerText = document.querySelector("#movie option[value='${urlParams.get('movie')}']").innerText;
    document.getElementById('receipt-seats').innerText = urlParams.get('seats'); // This assumes seats are passed as a query parameter
    document.getElementById('receipt-amount').innerText = '$20'; // Replace with dynamic calculation if needed
  </script>
</body>
</html>
```

5.9 main.py

```
from flask import Flask, render_template, request, jsonify, redirect, url_for, session, flash
import urllib.parse
from pymongo import MongoClient
import certifi

app = Flask(__name__)
app.secret_key = 'your_secret_key'
username = "Dinesh"
password = "Dkimp2005"
database_name = "cluster0"
encoded_username = urllib.parse.quote_plus(username)
encoded_password = urllib.parse.quote_plus(password)
connection_string = f"mongodb+srv://{encoded_username}:{encoded_password}@cluster0.mongodb.net/{database_name}?retryWrites=true&w-majority"
client = MongoClient(f"mongodb+srv://{Dinesh:Dkimp2005@cluster0.hlwx20a.mongodb.net/?retryWrites=true&w-majority&appName=Cluster0}")
# Select database and collection
db = client[database_name]
users_collection = db["users"]
seats_collection = db["seats"]
@app.route('/')
def index():
    if 'logged_in' in session:
        return render_template('index.html')
    return render_template('index.html')
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']

        user = users_collection.find_one({"username": username})
        if user and user['password'] == password:
            session['logged_in'] = True
            return redirect(url_for('movie_page'))
        else:
            error_message = "Incorrect username or password. Please try again."
            return render_template('index.html', error_message=error_message)
    return render_template('index.html')
@app.route('/logout')
def logout():
    session.pop('logged_in', None)
    return redirect(url_for('index'))
@app.route('/book_ticket', methods=['POST'])
def book_ticket():
    if 'logged_in' in session:
        selected_seats = request.json.get('selected_seats', [])
        date = request.json.get('date')
        time = request.json.get('time')
        key = f"{date}-{time}"

        seats = seats_collection.find_one({"key": key})
        if not seats:
            seats = {"key": key, "seats": {str(i): 'available' for i in range(1, 100)}}
            seats_collection.insert_one(seats)

        response = {'status': 'success', 'booked': [], 'already_booked': []}
        for seat in selected_seats:
            seat = str(seat)
            if seats['seats'].get(seat) == 'available':
                seats['seats'][seat] = 'booked'
```

```

        response['booked'].append(seat)
    else:
        response['already_booked'].append(seat)
    seats_collection.update_one({"key": key}, {"$set": {"seats": seats['seats']}})
    return jsonify(response)
    return redirect(url_for('index'))
@app.route('/movies')
def movie_page():
    if 'logged_in' in session:
        return render_template('movies.html')
    return redirect(url_for('index'))
@app.route('/book1')
def seat_booking1():
    if 'logged_in' in session:
        return render_template('book1.html')
    return redirect(url_for('index'))
@app.route('/book2')
def seat_booking2():
    if 'logged_in' in session:
        return render_template('book2.html')
    return redirect(url_for('index'))
@app.route('/book3')
def seat_booking3():
    if 'logged_in' in session:
        return render_template('book3.html')
    return redirect(url_for('index'))
@app.route('/book4')
def seat_booking4():
    if 'logged_in' in session:
        return render_template('book4.html')
    return redirect(url_for('index'))
@app.route('/seats', methods=['GET'])
def get_seats():
    if 'logged_in' in session:
        date = request.args.get('date')
        time = request.args.get('time')
        key = f"{date}-{time}"
        seats = seats_collection.find_one({"key": key})
        if not seats:
            seats = {str(i): 'available' for i in range(1, 100)}
        else:
            seats = seats['seats']
        return jsonify(seats)
    return redirect(url_for('index'))
@app.route('/initialize_seats', methods=['POST'])
def initialize_seats():
    # Initialize seat availability for a specific date and time
    date = request.form['date']
    time = request.form['time']
    key = f"{date}-{time}"
    if seats_collection.find_one({"key": key}):
        flash('Seats already initialized for this date and time.')
    else:
        seats_collection.insert_one({"key": key, "seats": {str(i): 'available' for i in range(1, 100)}})
        flash('Seats initialized successfully.')
    return redirect(url_for('index'))
if __name__ == '__main__':
    app.run(debug=True)

```

6. Result - Snap Shots

1. User login module



A screenshot of a user login module. It features a light gray background with a central white rectangular box containing the login form. The form has two input fields: one for 'USERNAME' and one for 'PASSWORD'. Below these fields are two buttons: 'LOGIN' and 'NEW REGISTER'.

| | |
|-----------------|--------------------------|
| USERNAME | <input type="text"/> |
| PASSWORD | <input type="password"/> |
| LOGIN | NEW REGISTER |

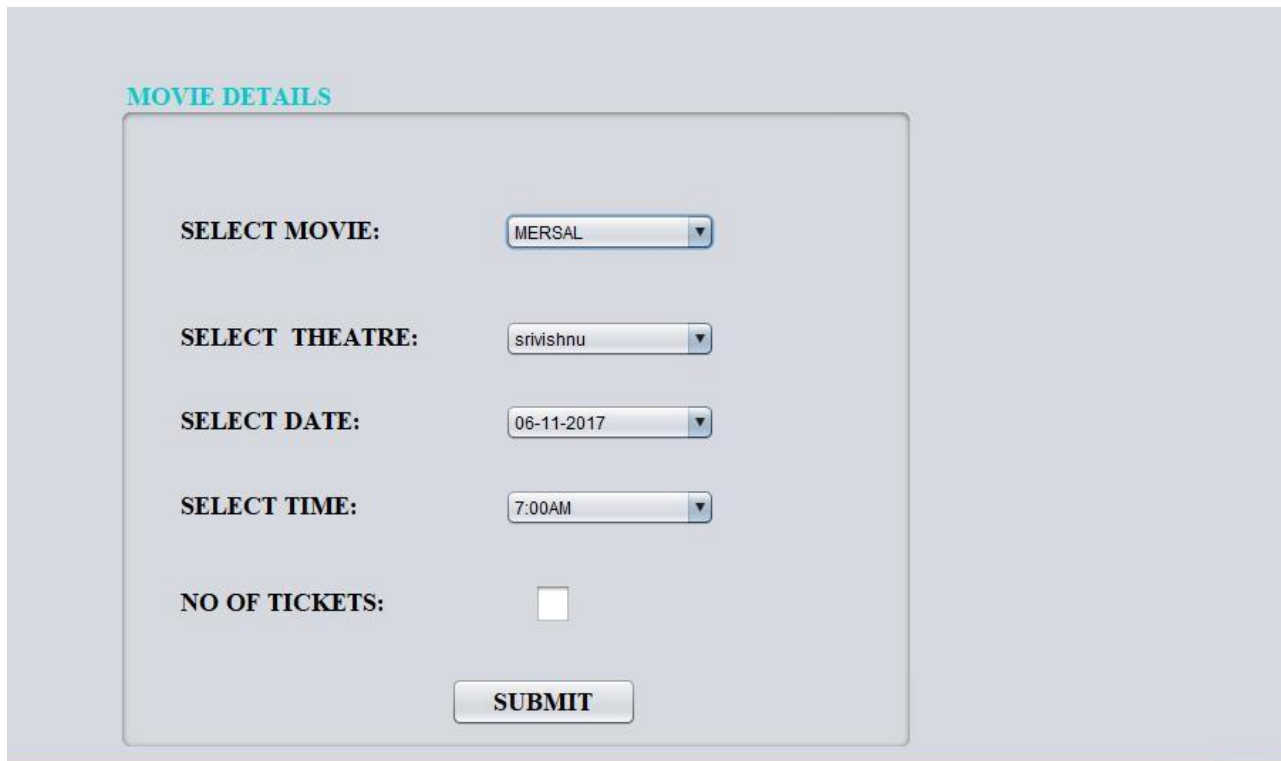
2. New registration module:



A screenshot of a web application window titled "NEW REGISTER". The window has a light blue header bar with standard window controls (minimize, maximize, close). The main content area is white and contains the following fields and a button:

- NAME**: A text input field with a cursor inside.
- EMAIL ID**: A text input field.
- PHONE NUMBER**: A text input field.
- PASSWORD**: A text input field.
- SIGN UP**: A button with a light blue gradient and a dark blue border.

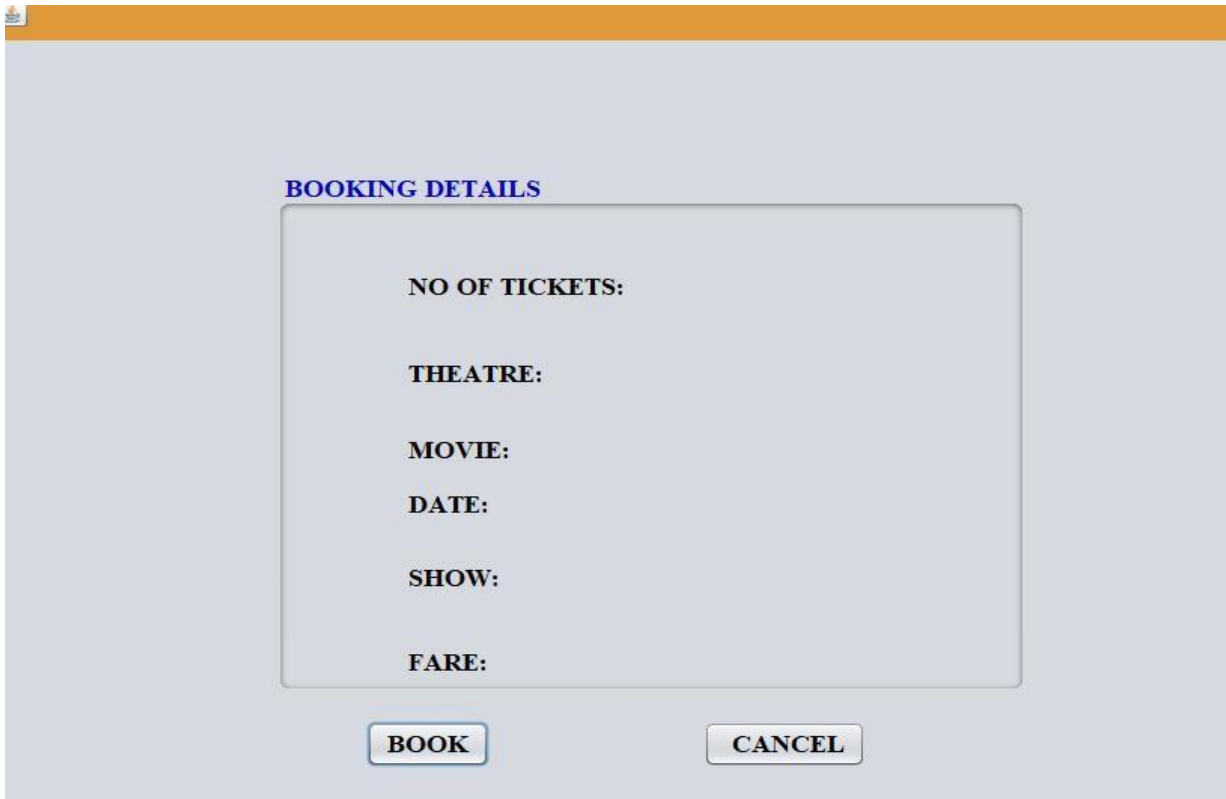
3. Selection of Movie time, date, theater and ticket details:



A screenshot of a web application window titled "MOVIE DETAILS". The window has a light blue header bar with standard window controls. The main content area is white and contains the following fields and a button:

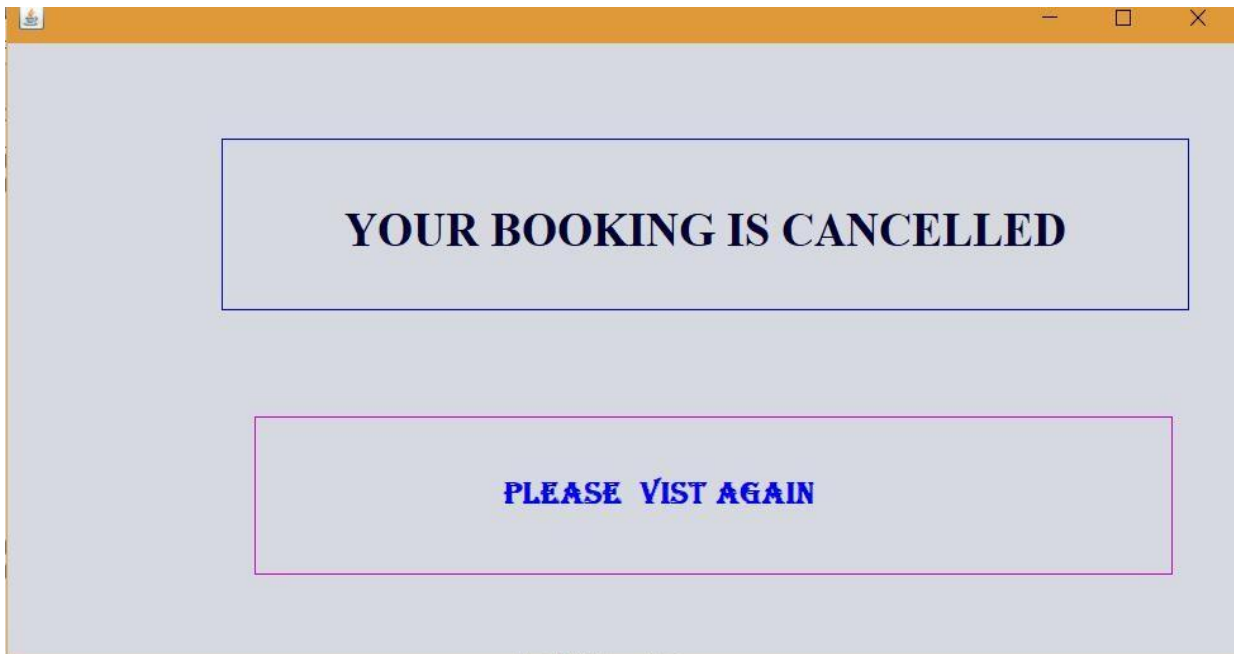
- SELECT MOVIE:** A dropdown menu with "MERSAL" selected.
- SELECT THEATRE:** A dropdown menu with "srivishnu" selected.
- SELECT DATE:** A dropdown menu with "06-11-2017" selected.
- SELECT TIME:** A dropdown menu with "7:00AM" selected.
- NO OF TICKETS:** A text input field with a cursor inside.
- SUBMIT**: A button with a light blue gradient and a dark blue border.

4. Book ticket:



A screenshot of a web application window with an orange title bar. The main content area is light gray. At the top, the text **BOOKING DETAILS** is displayed in blue. Below it is a white rectangular form with a gray border. Inside the form, the following labels are listed vertically: **NO OF TICKETS:**, **THEATRE:**, **MOVIE:**, **DATE:**, **SHOW:**, and **FARE:**. Below the form, there are two buttons: **BOOK** (with a blue border) and **CANCEL** (with a gray border).

5. Cancel ticket:



A screenshot of a web application window with an orange title bar. The main content area is light gray. In the center, there is a white rectangular box with a blue border containing the text **YOUR BOOKING IS CANCELLED**. Below this box, there is another white rectangular box with a purple border containing the text **PLEASE VISIT AGAIN**.

6. Card selection:

CONFIRM DETAILS

| | |
|-------------------------|---------------------------|
| NO OF TICKETS: 2 | THEATER: srivishnu |
| MOVIE: MERSAL | DATE: 06-11-2017 |
| FARE: 200 | TIME: 7:00AM |

PAYMENT TYPE

☐ Credit card

☒ Debit card

☐ NET BANKING

☐ PAYTM WALLET

7. Card details:

CONFORM BOOKING

| | |
|-------------------------|---------------------------|
| NO OF TICKETS: 2 | THEATER: srivishnu |
| MOVIE: MERSAL | DATE: 06-11-2017 |
| FARE: 200 | TIME: 7:00AM |

CARD DETAILS

card number

47041524

card holder name

bnses

CVV *** **exp** 01 2018

8. Generating Tickets :

RECEIPT

| | |
|----------------|------------|
| NO OF TICKETS: | 2 |
| THEATRE: | srivishnu |
| MOVIE: | 06-11-2017 |
| DATE: | 7:00AM |
| SHOW: | MERSAL |
| FARE: | 200 |

THANKS FOR BOOKING

7. Result and discussion:

So, First we login to the system with username and password that we already data has been fed into the database. Then it takes into another page where we select movie and then after we selecting movie another page comes then we select show timings. After selecting show timings it takes into another page there we select number of tickets we want. After selecting number of tickets it takes into another page there we print ticket with cost.