M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur - 639 113, TAMILNADU.

A Project Report

on

TEMPLE DHARSHAN BOOKING APP

Submitted in partial fulfillment of requirements for the award of the course

of

EGB1221-DATABASE MANAGEMENT SYSTEMS

Under the guidance of

Mr.G.Subramaniam, M.E.,/Mrs.P.Sasirekha

Assistant Professor / EEE

Submitted By

LIBNI DHIHONA J M **(927623BEE047)**

LITHIKA P **(927623BEE049)**

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING
(Autonomous)

KARUR – 639 113

MAY 2025

M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur - 639 113, TAMILNADU.

M. KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous Institution affiliated to Anna University, Chennai)

KARUR – 639 113

BONAFIDE CERTIFICATE

This is to certify that this project report on **"TEMPLE DHARSHAN BOOKING APP"** is the bonafide work of **LIBNI DHIHONA J M (927623BEE047) , LITHIKA P (927623BEE049)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

Signature

Mr.G.Subramaniam, M.E.,

SUPERVISOR,

Department of Electrical and Electronics Engineering,

M.Kumarasamy College of Engineering,

Thalavapalayam, Karur -639 113.

Signature

Dr.J.Uma, M.E., Ph.D.,

HEAD OF THE DEPARTMENT,

Department of Electrical and Electronics Engineering,

M.Kumarasamy College of Engineering,

Thalavapalayam, Karur -639 113.

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

VISION OF THE INSTITUTION

To emerge as a leader among the top institutions in the field of technical education

MISSION OF THE INSTITUTION

- Produce smart technocrats with empirical knowledge who can surmount the global challenges
- Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students
- Maintain mutually beneficial partnerships with our alumni, industry, and Professional associations

DEPARTMENT VISION, MISSION, PEO, PO AND PSO

VISION

To produce smart and dynamic professionals with profound theoretical and practical knowledge comparable with the best in the field.

MISSION

M1: Produce hi-tech professionals in the field of Electrical and Electronics Engineering by inculcating core knowledge

M2: Produce highly competent professionals with thrust on research.

M3: Provide personalized training to the students for enriching their skills.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

- **PEO1:** Graduates will have flourishing career in the core areas of Electrical Engineering and allied disciplines.
- **PEO2:** Graduates will pursue higher studies and succeed in academic/research careers.
- **PEO3:** Graduates will be a successful entrepreneur in creating jobs related to Electrical and Electronics Engineering /allied disciplines.
- **PEO4:** Graduates will practice ethics and have habit of continuous learning for their success in the chosen career.

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Apply the basic concepts of mathematics and science to analyse and design circuits, controls, Electrical machines and drives to solve complex problems.
- **PSO2:** Apply relevant models, resources and emerging tools and techniques to provide solutions to power and energy related issues & challenges.
- **PSO3:** Design, Develop and implement methods and concepts to facilitate solutions for electrical and electronics engineering related real world problems.

ABSTRACT

The Temple Darshan Booking App is a mobile-based application designed to streamline and enhance the experience of devotees visiting temples by enabling online booking of darshan slots. In the wake of increasing pilgrim footfall and the need for effective crowd management—especially during festivals and special events—this app aims to offer a seamless, contactless, and user-friendly solution for devotees. Key features include slot booking, e-pass generation, queue management, real-time updates on temple timings, crowd density notifications, and integration with digital payment systems for donations and special puja services. The app supports multi-language functionality, location-based services to find nearby temples, and personalized alerts for special events. By digitizing the traditional process, the app not only improves convenience and safety for visitors but also assists temple authorities in managing resources and operations efficiently.

ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	COs MAPPED	POs MAPPED	PSOs MAPPED
The Multi-Temple Booking System enables easy pooja bookings at temples like Murugan and Mariyamman. It offers OTP-based login, dynamic pricing, QR code entry, and seat tracking. Admins can manage bookings and view data through a dashboard, ensuring smooth and inclusive temple operations.	CO1 CO2 CO3 CO4 CO5	PO1 PO2 PO3 PO4 PO5 PO6 PO7 PO8 PO9 PO10 PO11 PO12	PSO1 PSO2

SUPERVISOR **HEAD OF THE DEPARTMENT**

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE

The objective of the Multi-Temple Booking System is to provide a user-friendly, secure, and efficient platform for managing temple bookings and facilitating seamless interactions between devotees and temple authorities. The system enables users to register, log in, and book pooja slots at multiple temples, select from predefined pooja types (Regular or Special) with associated pricing, and choose time slots based on availability. It ensures transparency by dynamically calculating booking costs based on the number of members, their ages. The system incorporates email verification through OTP and QR code scanning to enhance payment security and provides an admin dashboard for temple administrators to manage bookings, reset seat counts, and view temple specific data. Additionally, it supports inclusivity by offering assistance options for differently abled users and ensures accessibility through intuitive navigation and responsive design. Overall, the system aims to streamline the booking process while maintaining accountability, security, and convenience for both users and administrators.

1.2 OVERVIEW

The Multi-Temple Booking System is a web-based application designed to streamline the process of booking poojas and managing temple visits for users. It provides an where users can register, log in, and select from multiple temples, such as the Murugan Temple and Mariyamman Temple, each offering predefined pooja types like "Regular" and "Special" with associated pricing and time slots.The system ensures secure transactions through email verification using OTP and QR code scanning, while also allowing users to calculate booking costs dynamically based on the number of members, their ages, and any additional charges for special poojas. Features like seat availability tracking, age-based pricing (₹100 for ages 10–60), and options for differently-abled users enhance accessibility and usability. Administrators can manage bookings, reset seat counts, and view temple-specific data through a dedicated admin dashboard. With its user-friendly design, robust validation mechanisms, and focus on transparency.

1.3 DATABASE MANAGEMENT SYSTEMS' CONCEPTS

The Temple Dharsana Booking System uses a relational database to store and manage information efficiently. The database is designed using normalization principles to reduce redundancy and ensure data integrity. It consists of multiple interrelated tables that support user activities, temple schedules, pooja categories, and booking records.

Key Tables and Their Descriptions:

1. Users Table :

Stores user account information.

Fields: user_id (PK), name, email, password, phone, address, role (user/admin)

2. Temples Table :

Stores details of each temple.

Fields: temple_id (PK), temple_name, location, description, status

3. Pooja_Categories Table :

Lists all types of pooja services offered.

Fields: pooja_id (PK), temple_id (FK), pooja_name, duration, price

4. Time_Slots Table :

Stores available time slots for pooja/dharisana.

Fields: slot_id (PK), temple_id (FK), date, time, capacity, available_seats

5. Bookings Table:

Records each user's booking.

Fields: booking_id (PK), user_id (FK), temple_id (FK), pooja_id (FK), slot_id (FK), booking_date, status

6. Admins Table :

Stores admin user information for managing backend activities.

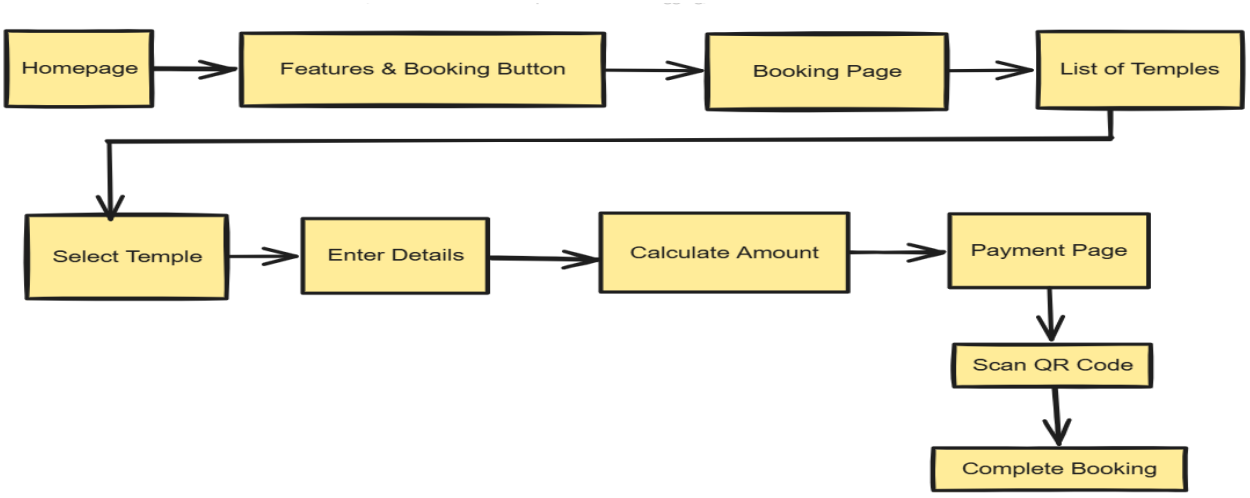
CHAPTER 2

PROJECT METHODOLOGY

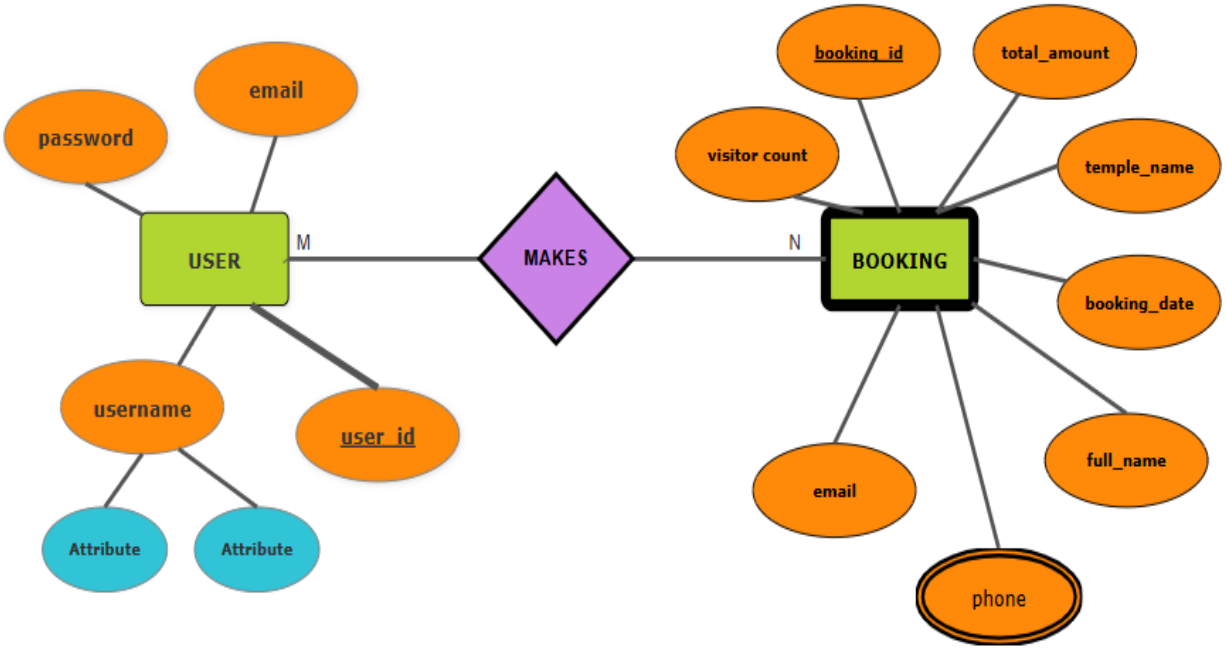
2.1 PROPOSED WORK

The Multi-Temple Booking System is a comprehensive web-based application designed to facilitate seamless booking of poojas and temple visits while ensuring user convenience and administrative efficiency. The system allows users to register, log in, and select from multiple temples, such as the Murugan Temple and Mariyamman Temple, each offering predefined pooja types like "Regular" and "Special" with associated pricing and time slots. Users can book slots by selecting a date, time, and number of members, with dynamic age-based pricing applied for individuals aged 10–60. Additional features include email verification through OTP and QR code scanning to ensure secure transactions, seat availability tracking, and calendar-based suggestions for optimal booking times. The system also supports differently-abled users by providing options for assistance and ensures transparency by displaying total costs before payment. Administrators can manage bookings, reset available seats, and view temple-specific data through a dedicated admin dashboard.

2.2 ARCHITECTURE



2.3 E-R DIAGRAM



CHAPTER 3

MODULE DESCRIPTION

3.1 Module 1: User Authentication Module

This module handles user registration, login, and session management. It ensures secure access to the system by validating user credentials and maintaining user-specific data throughout the session.

Features :

• Registration of new users with unique usernames and passwords.

• Login functionality to authenticate users.

• Error handling for invalid credentials or duplicate usernames.

• Session tracking using the current User variable to personalize the user experience.

3.2 Module 2: Booking Module

This module allows users to select a temple, view available seats, and book slots for poojas. It includes dynamic features like seat availability tracking and time slot selection based on the chosen pooja type.

Features :

• Display of temples with images and descriptions (e.g., Murugan Temple, Mariyamman Temple).

• Dynamic updates of available seats (TEMPLE_SEATS) based on bookings.

• Selection of pooja types (Regular/Special) and corresponding time slots.

• Validation of booking date and time to prevent past bookings.

3.2.1 Module 3: Donation Module

The Donation Module is an essential component of the Temple Darshan Booking App, designed to facilitate transparent, secure, and user-friendly contributions from devotees. It provides multiple options for devotees to donate to temples, support specific causes, or contribute to daily or festival rituals.

Features:

• General donation (Annadhanam, temple maintenance, etc.)

• Festival/event-based donations.

• Notifications and reminders for upcoming recurring donations.

3.3 Module 4 Cancellation Module

This module allows users to view their past bookings, cancel existing bookings, and initiate refunds. It enhances user convenience and flexibility in managing their schedules.

Features :

• Display of all bookings made by the logged-in user.

• Option to cancel bookings with refund initiation.

• Real-time updates to seat availability upon cancellation.

• Logout functionality to end the user session securely.

CHAPTER 4

RESULTS AND DISCUSSION

Login and Register:

Fig : 4.1

The image shows a login/register page for a Multi-Temple Booking web app. It features a top menu with options like Login/Register, Select Temple, Book Now, and View Bookings. The center form includes username and password fields with Register and Login buttons. The background is a blurred temple corridor, giving a traditional and spiritual feel.

Temple Booking

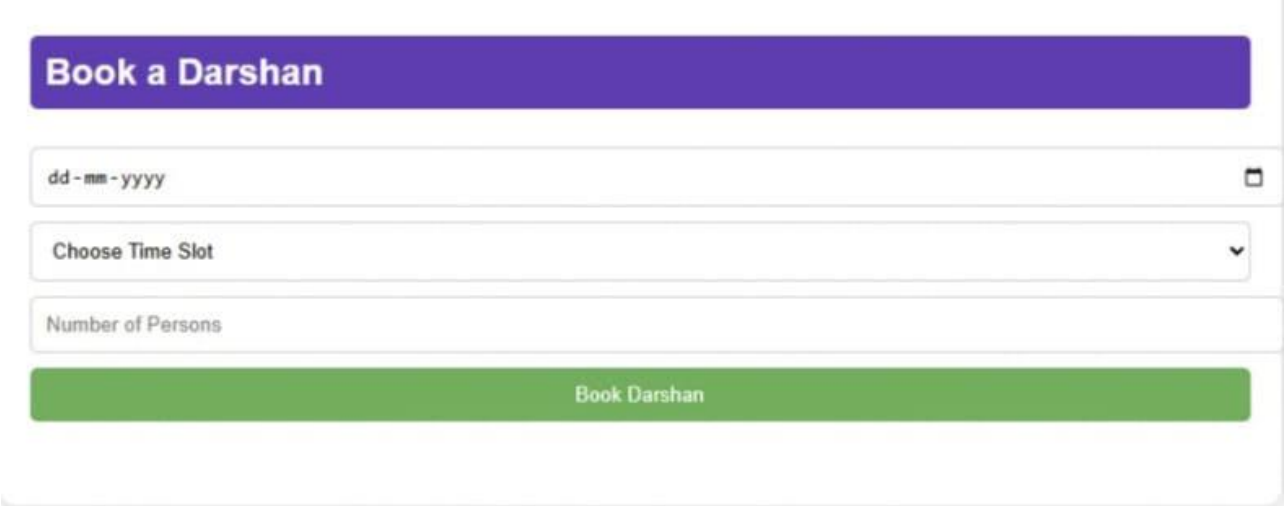


Fig : 4.2

A booking form for Temple with details like date (14-May-2025), time (04:00 PM), 1 member aged 22, pooja type (Kappu Vizha), and disability status (No). It also includes a Calculate Amount button with total ₹0 displayed.

Donation

Fig : 4.3

The Donation feature in the Temple Darshan Booking App offers devotees a convenient and secure platform to contribute towards the welfare and development of temples and associated religious activities. Through this feature, users can make voluntary contributions to support various causes such as temple maintenance, daily rituals, annadhanam (free food distribution), festival arrangements, and charitable activities carried out by the temple trust.

Booking Cancellation

Fig : 4.4

The Booking Cancellation feature allows users to cancel their darshan or service bookings easily through the app. Upon cancellation, users receive a confirmation along with refund details (if applicable), ensuring a smooth and transparent process.

View Bookings

Fig : 4.5

The image shows a "Your Bookings" screen from the Multi-Temple Booking System. It lists a confirmed booking for the Murugan Temple on May 14, 2025, at 16:00, with details such as no special assistance required and an amount paid of ₹100. Users can cancel the booking or log out using the provided buttons.

CHAPTER 5

CONCLUSION

The Multi-Temple Booking System is a comprehensive web-based application designed to streamline the process of temple bookings for users while providing administrators with tools to manage resources efficiently. The system integrates modern web development practices, including dynamic HTML/CSS for user interfaces, JavaScript for interactivity, and simulated backend data structures for managing users, bookings, and seat availability. Key features such as email verification, OTP-based authentication, and QR code scanning ensure secure transactions, while the inclusion of age-based pricing, seat tracking, and admin dashboards enhances usability and accountability.

The project demonstrates the practical application of database management concepts, such as data modeling, integrity, and querying, to create a robust and scalable solution. By addressing real-world challenges like preventing overbooking, ensuring inclusivity for differently abled users, and providing transparent payment processes, the system offers a seamless experience for both devotees and temple authorities. Future enhancements could include integrating a real database (e.g., MySQL or MongoDB), adding more temples and pooja types, and implementing advanced features like calendar-based booking suggestions or AI-driven assistance.

REFERENCES:

[1].https://www.researchgate.net/publication/380729587_TEMPLE_DHARSHAn_BOOKING_SYSTEM

[2].https://ieeexplore.ieee.org/document/8531455

APPENDIX

(Coding)

```
<!DOCTYPE html>
<html>
<head><title>Register</title></head>
<body>
    <h2>Register</h2>
    <form action="register_process.php" method="POST">
        <input type="text" name="name" placeholder="Full Name" required><br>
        <input type="email" name="email" placeholder="Email" required><br>
        <input type="password" name="password" placeholder="Password" required><br>
        <button type="submit">Register</button>
    </form>
</body>
</html>
<?php
include 'config.php';

$name = $_POST['name'];
$email = $_POST['email'];
$password = password_hash($_POST['password'], PASSWORD_BCRYPT);
$sql = "INSERT INTO users (name, email, password) VALUES (?, ?, ?)";
$stmt = $conn->prepare($sql);
$stmt->bind_param("sss", $name, $email, $password);
if ($stmt->execute()) {
```

```
 echo "Registration successful. <a href='index.php'>Login here</a>";
} else {
    echo "Error: " . $stmt->error;}
?>
<?php
session_start();
include 'config.php';

$email = $_POST['email'];
$password = $_POST['password'];

$sql = "SELECT * FROM users WHERE email = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $email);
$stmt->execute();
$result = $stmt->get_result();

if ($result->num_rows === 1) {
    $user = $result->fetch_assoc();
    if (password_verify($password, $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['name'] = $user['name'];
        header("Location: dashboard.php");
    } else {
        echo "Wrong password.";}
```

```
} else {
    echo "No user found.";
}
?>
<?php
session_start();
include 'config.php';

$email = $_POST['email'];
$password = $_POST['password'];
$sql = "SELECT * FROM users WHERE email = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $email);
$stmt->execute();
$result = $stmt->get_result();

if ($result->num_rows === 1) {
    $user = $result->fetch_assoc();
    if (password_verify($password, $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['name'] = $user['name'];
        header("Location: dashboard.php");
    } else {
        echo "Wrong password.";
    }
} else {
    echo "No user found.";
```

```
}
?>
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header("Location: index.php");
    exit();}
?>
<!DOCTYPE html>
<html>
<head><title>Donate</title></head>
<body>
    <h2>Make a Donation</h2>
    <form action="donate_process.php" method="POST">
        <input type="text" name="donor_name" placeholder="Your Name" required><br>
        <input type="number" name="amount" placeholder="Amount (INR)" required><br>
        <select name="category" required>
            <option value="">Select Category</option>
            <option value="Annadhanam">Annadhanam</option>
            <option value="Renovation">Renovation</option>
        </select><br>
        <textarea name="message" placeholder="Message (optional)"></textarea><br>
        <button type="submit">Donate</button>
    </form>
```

```
    <p><a href="dashboard.php">Back to Dashboard</a></p>
</body>
</html>
<?php
session_start();
include 'config.php';
$user_id = $_SESSION['user_id'];
$name = $_POST['donor_name'];
$amount = $_POST['amount'];
$category = $_POST['category'];
$message = $_POST['message'];

$sql = "INSERT INTO donations (user_id, donor_name, amount, category, message) VALUES (?, ?, ?, ?, ?)";
$stmt = $conn->prepare($sql);
$stmt->bind_param("isiss", $user_id, $name, $amount, $category, $message);

if ($stmt->execute()) {
    echo "Thank you for your donation! <a href='dashboard.php'>Back</a>";
} else {
    echo "Error: " . $stmt->error;}
?>
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header("Location: index.php"); // redirect to login
    exit();
```

```
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Darshan Booking</title>
</head>
<body>
    <h2>Book Your Darshan</h2>
    <form action="booking_process.php" method="POST">
        <label>Date of Darshan:</label><br>
        <input type="date" name="darshan_date" required><br><br>

        <label>Time Slot:</label><br>
        <select name="time_slot" required>
            <option value="">Select Time Slot</option>
            <option value="6 AM - 8 AM">6 AM - 8 AM</option>
            <option value="9 AM - 11 AM">9 AM - 11 AM</option>
            <option value="4 PM - 6 PM">4 PM - 6 PM</option>
            <option value="7 PM - 9 PM">7 PM - 9 PM</option>
        </select><br><br>

        <label>Number of Persons:</label><br>
        <input type="number" name="num_persons" min="1" required><br><br>

        <button type="submit">Book Now</button>
    </form>
```

```
    <p><a href="dashboard.php">Back to Dashboard</a></p>
</body>
</html>
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header("Location: index.php");
    exit();
}
include 'config.php';

$user_id = $_SESSION['user_id'];
$date = $_POST['darshan_date'];
$slot = $_POST['time_slot'];
$persons = $_POST['num_persons'];

$sql = "INSERT INTO bookings (user_id, darshan_date, time_slot, num_persons)
VALUES (?, ?, ?, ?)";
$stmt = $conn->prepare($sql);
$stmt->bind_param("issi", $user_id, $date, $slot, $persons);

if ($stmt->execute()) {
    echo "Booking successful! <a href='dashboard.php'>Go to Dashboard</a>";
} else {
    echo "Booking failed: " . $stmt->error;}
?>
<?php
```

```
$host = 'localhost';
$db = 'temple_db';
$user = 'root';
$pass = '';

$conn = new mysqli($host, $user, $pass, $db);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header("Location: index.php");
    exit();
}
?>
<!DOCTYPE html>
<html>
<head><title>Dashboard</title></head>
<body>
    <h2>Welcome, <?php echo $_SESSION['name']; ?>!</h2>
    <ul>
        <li><a href="donation.php">Make a Donation</a></li>
        <li><a href="logout.php">Logout</a></li>
    </ul>
</body>
```

```
</html>
<?php
session_start();
include 'config.php';
?>

<!DOCTYPE html>
<html>
<head>
    <title>Admin View - Temple Darshan</title>
    <style>
        body {
            font-family: sans-serif;
            margin: 0;
            padding: 0;
            background-image: url('murugan.jpg');
            background-size: cover;
            background-attachment: fixed;
            background-repeat: no-repeat;
            background-position: center;
            color: #fff;
            animation: fadeInBody 2s ease-in;
        }

        @keyframes fadeInBody {
            from { opacity: 0; }
            to { opacity: 1; }
```

```
  }
  overlay {
    background: rgba(0, 0, 0, 0.75);
    padding: 40px;
    min-height: 100vh;
}

table {
    border-collapse: collapse;
    width: 100%;
    margin-bottom: 40px;
    background: rgba(255, 255, 255, 0.95);
    color: #000;
    animation: fadeInSection 1.5s ease-in;
    border-radius: 8px;
    overflow: hidden;
}

th, td {
    border: 1px solid #ccc;
    padding: 10px;
    text-align: left;
}

th {
    background: #d32f2f;
    color: white;
```

```
  }
  h1, h2 {
    text-align: center;
    color: gold;
    text-shadow: 1px 1px 2px #000;
    animation: fadeInText 1s ease-in;
}

@keyframes fadeInSection {
    from { transform: translateY(30px); opacity: 0; }
    to { transform: translateY(0); opacity: 1; }
}

@keyframes fadeInText {
    from { opacity: 0; transform: scale(0.9); }
    to { opacity: 1; transform: scale(1); }
}

a {
    color: #ffcc00;
    text-decoration: none;
    display: block;
    text-align: center;
    margin-top: 30px;
    font-weight: bold;
}
```

```
        a:hover {
            text-decoration: underline;
        }
    </style>
</head>
<body>
<div class="overlay">
    <h1> Admin Dashboard - Temple Darshan</h1>

    <!-- Users Table -->
    <h2> Registered Users</h2>
    <table>
        <tr><th>ID</th><th>Name</th><th>Email</th><th>Registered On</th></tr>
        <?php
        $res = $conn->query("SELECT * FROM users");
        while ($row = $res->fetch_assoc()) {
            echo "<tr>
                    <td>{$row['id']}</td>
                    <td>{$row['name']}</td>
                    <td>{$row['email']}</td>
                    <td>{$row['created_at']}</td>
                  </tr>";
        }
        ?>
    </table>

    <!-- Bookings Table -->
```

```
<h2>🛕 Darshan Bookings</h2>
  <table>
    <tr><th>ID</th><th>User ID</th><th>Date</th><th>Time Slot</th><th>Persons</th><th>Booked On</th></tr>
    <?php
    $res = $conn->query("SELECT * FROM bookings");
    while ($row = $res->fetch_assoc()) {
      echo "<tr>
            <td>{$row['id']}</td>
            <td>{$row['user_id']}</td>
            <td>{$row['darshan_date']}</td>
            <td>{$row['time_slot']}</td>
            <td>{$row['num_persons']}</td>
            <td>{$row['booked_at']}</td>
          </tr>";
    }
    ?>
  </table>

  <!-- Donations Table -->
  <h2>💰 Donations</h2>
  <table>
    <tr><th>ID</th><th>User ID</th><th>Name</th><th>Amount</th><th>Type</th><th>Message</th><th>Donated On</th></tr>
    <?php
```

```
$res = $conn->query("SELECT * FROM donations");
    while ($row = $res->fetch_assoc()) {
        echo "<tr>
            <td>{$row['id']}</td>
            <td>{$row['user_id']}</td>
            <td>{$row['donor_name']}</td>
            <td>{$row['amount']}</td>
            <td>{$row['category']}</td>
            <td>{$row['message']}</td>
            <td>{$row['donated_at']}</td>
          </tr>";
    }
    ?>
  </table>

  <a href="index.php">← Back to Home</a>
</div>
</body>
</html>
<?php
session_start();
session_destroy();
header("Location: index.php");
exit();
?>
```