TEMPLE DARSHAN BOOKING WEBSITE 

# A PROJECT REPORT

***Submitted by***

# MONISHA C (2303811710622070)

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

**EGB1221- DATABASE MANAGEMENT SYSTEMS**

***In***

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

**SAMAYAPURAM – 621 112**

# JUNE - 2025

**K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)**

### SAMAYAPURAM – 621 112

**BONAFIDE CERTIFICATE**

Certified that this project report on **“TEMPLE DARSHAN WEBSITE”** is the bonafide work of **MONISHA C (2303811710622070)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

|  |  |
| --- | --- |
| **SIGNATURE**  Dr.S.SYEDAKBAR, M.E., Ph.D., **HEAD OF THE DEPARTMENT** ASSISTANT PROFESSOR  Department of ECE  K.Ramakrishnan College of Technology (Autonomous)  Samayapuram–621112. | **SIGNATURE** Mr.M.RAJA, M.E., (Ph.D)., **SUPERVISOR** ASSISTANT PROFESSOR  Department of ECE  K.Ramakrishnan College of Technology (Autonomous)  Samayapuram–621112. |

Submitted for the viva-voce examination held on …………….

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DECLARATION**

I declare that the project report on **“TEMPLE DARSHAN WEBSITE”** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of **BACHELOR OF ENGINEERING.** This project report is submitted on the partial fulfillment of the requirement of the award of the course **EGB1221 - DATABASE MANAGEMENT SYSTEMS.**

**Signature**

Monisha C

Place: Samayapuram Date:

**ACKNOWLEDGEMENT**

It is with great pride that I express our gratitude and indebtedness to our institution,

**“K. Ramakrishnan College of Technology (Autonomous)”**, for providing us with the opportunity to do this project.

I extend our sincere acknowledgment and appreciation to the esteemed and honorable Chairman, **Dr. K. RAMAKRISHNAN, B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director,

**Dr. S. KUPPUSAMY, MBA, Ph.D.**, for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.**, Principal, who gave the opportunity to frame the project to full satisfaction.

I thank **Dr. S. SYEDAKBAR, M.E., Ph.D.,** Head of the Department of **ELECTRONICS AND COMMUNICATION ENGINEERING**, for providing her encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide

**Mr. M. RAJA, M.E., (PhD)**, Department of **ELECTRONICS AND COMMUNICATION**

**ENGINEERING**, for his incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

### VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards

### MISSION OF THE INSTITUTION

* Be a center of excellence for technical education in emerging technologies by exceeding the needs of the industry and society.
* Be an institute with world class research facilities
* Be an institute nurturing talent and enhancing the competency of students to transform them as all-round personality respecting moral and ethical values

### VISION OF DEPARTMENT

To create innovative and socially responsible Electronics and Communication Engineers with design skills and research focus to meet Societal and Industrial needs.

### MISSION OF DEPARTMENT

**M1**: To provide high quality education and professional ethics to students through enhanced learning environment.

**M2:** To impart a creative environment towards centre of excellence in department with design skill and exposure for research.

**M3:** To nurture required employable skills of students to satisfy the industry and social needs with ethical and human values.

### PROGRAM EDUCATIONAL OBJECTIVES

**PEO1: Core Knowledge Development:** Graduates will have enhanced engineering skills in the field of electronics, communication and interdisciplinary areas to serve the society with global standards.

**PEO2: Professional development:** Graduates will apply the technical knowledge for continuous up gradation of their professional skills to become an inimitable employee, researcher or entrepreneur.

**PEO3: Analytical Thinking:** Graduates will have analytic and thinking skills to provide the innovative solutions for industry and societal requirements.

#### PROGRAM SPECIFIC OUTCOMES (PSOs)

**PSO1:** To analyze, design and develop solutions by applying foundational concepts of electronics and communication engineering.

**PSO2:** To apply design principles and best practices for developing quality products for scientific and business applications.

**PSO3:** To adapt to emerging information and communication technologies (ICT) to innovate ideas and solutions to existing/novel problems.

#### 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

1. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
2. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
3. **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.
4. **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.
5. **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.

# ABSTRACT

The Temple Darshan Booking System is a user-friendly web application designed to digitize and simplify the traditional process of temple darshan. It allows devotees to search for temples by location, view available options, and book darshan slots by selecting preferred dates and time slots. This not only enhances the convenience for users but also helps manage temple crowding by distributing visitor flow more evenly. The system includes essential modules such as user booking, temple management, and an admin interface to monitor bookings and add new temples. Built using HTML, CSS, PHP, and MySQL, this project serves as a practical example of applying database systems and web technologies in solving real-life management challenges.

The Temple Darshan Booking System is a web-based application developed to streamline the process of visiting temples by enabling online booking of darshan slots. It allows users to view temples based on location, choose a preferred temple, select a date and time slot, and book their darshan conveniently from anywhere. The system stores all bookings in a centralized database, ensuring organized and reliable record- keeping. Additionally, it includes an admin module for adding temple details and managing bookings. This project not only enhances user convenience but also helps reduce crowding and improves overall management at temples.

#### ABSTRACT WITH POs AND PSOs MAPPING

**CO 5 : BUILD DATABSE APPLICATIONS FOR SOLVING REAL-TIME PROBLEMS.**

|  |  |  |
| --- | --- | --- |
| **ABSTRACT** | **POs MAPPED** | **PSOs MAPPED** |
| **Engineering Knowledge**  Apply knowledge of computing, mathematics, and engineering fundamentals to design and implement a database-driven temple booking system.  **Problem Analysis**  Identify and analyze real-world challenges in manual darshan bookings and develop systematic solutions using web technologies.  **Design/Development of Solutions**  Design a user-friendly and efficient system that meets the needs of both temple authorities and devotees.  **Modern Tool Usage**  Use modern engineering tools like HTML, CSS, PHP, and MySQL for designing, coding, and testing the web application. **Communication**  Communicate technical details effectively through user interfaces and clear booking confirmations for end-users. | **PO1-3** | **PSO1 PSO2 PSO3** |
| **PO2-3**  **PO3-3 PO4-3** |
| **PO5-3**  **PO6-3** |
| **PO7-3**  **PO8-3 PO9-3** |
| **PO10-3** |
| **PO11-3** |
| **PO12-3** |

Note: 1- Low, 2-Medium, 3- High

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** |  |
| 1 | **INTRODUCTION** | 1 |
|  | 1.1 Objective | 1 |
|  | 1.2 Overview | 1 |
|  | 1.3 SQL and Database concepts | 2 |
| 2 | **PROJECT METHODOLOGY** | 5 |
|  | 2.1Proposed Work | 5 |
|  | 2.2 Block Diagram | 6 |
| 3 | **MODULE DESCRIPTION** | 7 |
|  | 3.1 User Interface Module | 7 |
|  | 3.2 Temple Management Module | 7 |
|  | 3.3 Darshan Booking Module | 7 |
|  | 3.4 View Temples Module | 7 |
|  | 3.5 Booking Status Module | 8 |
|  | 3.6 Admin Module | 8 |
|  | 3.7 Database Module | 8 |
| 4 | **CONCLUSION AND FUTURE ENHANCEMENT** | 9  9  9  9 |
| 5 | **5.1 APPENDIX A SOURCE CODE** | 10 |
|  | **5.2 APPENDIX B SCREEN SHOT** | 39 |
| 6 | **REFERENCES** | 42 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **TITLE** | **PAGE NO** |
| 2.2.1 | PROPOSED BLOCK DIAGRAM | 06 |
| 5.2  5.2.1  5.2.2  5.2.3  5.2.4  5.2.5 | APPENDIX B SCREENSHOT  REGISTER OR LOGIN HOME PAGE  BOOK DARSHAN PAGE  BOOKING SUCCESSFUL PAGE VIEW YOUR BOOKINGS PAGE | 39  39  39  40  40  41 |

**LIST OF ABBREVIATIONS**

**ABBREVIATIONS**

HTML - Hyper Text Markup Language PHP - Hypertext Preprocessor

CSS - Cascading Style Sheets TXT - Input Text

# CHAPTER 1 INTRODUCTION

* 1. **OBJECTIVE**

The main objective of the Temple Darshan Booking System is to develop a user- friendly web application that simplifies and digitizes the process of temple darshan bookings. The system aims to allow devotees to view temples based on their location and select suitable dates and time slots for darshan, ensuring a more organized and hassle-free visit. By providing an online booking platform, the project helps reduce overcrowding and long waiting times at temples. It incorporates a structured database to securely store all temple and booking-related data, enabling efficient record management and easy retrieval. The system also includes an admin module that allows administrators to add new temples and manage bookings dynamically. Users can view their bookings by entering their name and email, ensuring transparency and ease of access. The interface is designed to be simple and intuitive, catering to both tech-savvy and general users. Additionally, the system is scalable, allowing future enhancements such as login/registration features, feedback modules, and multi-language support. Overall, this project demonstrates the practical use of database and web development technologies in solving real-world problems related to crowd management and user convenience at places of worship.

# OVERVIEW

The Temple Darshan Booking System is a web-based application developed to modernize the way devotees plan and manage their temple visits. Traditionally, temple visits involve long queues and unorganized crowding, which can be stressful for devotees and difficult for temple authorities to manage. This system aims to provide a digital solution by enabling users to book darshan slots in advance from the comfort of their homes.

Users can select a location to view a list of available temples in that area. Once a temple is selected, the system allows them to choose a suitable date and time slot for their darshan. Upon entering personal details such as name and email, the booking is stored in a centralized database. Additionally, users can view their bookings later by entering their registered details.

The admin side of the system provides an interface to add new temples into the system, ensuring that temple listings stay up-to-date. Built using HTML, CSS, PHP, and MySQL, the project emphasizes ease of use, reliability, and data accuracy. This system can be deployed locally using platforms like XAMPP, and can be scaled for broader use in real- time temple management.

# SQL AND DATABASE CONCEPTS

This project is based on core database management principles and uses SQL (Structured Query Language) to interact with a MySQL database. Below are the fundamental SQL and database concepts applied in the Temple Darshan Booking System:

## Database:

A database is an organized collection of data. In this project, we use a MySQL database named temple\_darshan to store all information about temples, users, and their bookings.

## Tables:

A table is a collection of related data organized in rows and columns. The main tables used in this project are:

* + - * temples – stores temple details (ID, name, location)
      * bookings – stores booking information (temple ID, user name, email, date, time slot)

## PrimaryKey:

Each table uses a primary key (id) which uniquely identifies every record. This ensures data integrity and avoids duplication.

## ForeignKey:

The temple\_id in the bookings table is a foreign key that refers to the id in the temples table, establishing a relationship between bookings and their respective temples.

## Data Types:

SQL uses data types to define the nature of data in each column. Examples include:

* + - * INT for IDs
      * VARCHAR for text such as names and emails
      * DATE for booking dates

## SQL Queries:

Various SQL statements are used in the project:

* + - * CREATE TABLE to define the structure of tables
      * INSERT INTO to add data (temples and bookings)
      * SELECT to retrieve data (like viewing temples or user bookings)
      * WHERE clause to filter specific records (e.g., bookings by email and name)

## Normalization:

The database is designed to avoid redundancy and maintain consistency through normalized table structures, ensuring efficient data management.

## Connectivity with PHP:

PHP is used to connect the frontend with the backend database using mysqli\_connect() and query execution functions like query() and prepare().

## Security Considerations:

Basic input validation and use of POST methods help protect the data. (Further enhancements can include SQL injection prevention and password protection.)

## Local Server Environment:

The database and application are hosted using XAMPP on localhost, enabling easy testing and deployment without requiring internet access.

# CHAPTER 2 PROJECT METHODOLOGY

* 1. **PROPOSED WORK**

The Temple Darshan Booking System aims to provide a streamlined and digital solution for booking darshan slots in temples. The proposed work focuses on building an efficient, accessible, and user-friendly web application that manages temple visits through online reservations. The system is designed to benefit both devotees and temple administrators by reducing manual processes and improving overall coordination.

The proposed application will include the following key components:

## User Interface for Devotees

* + - A clean and intuitive web interface allowing users to select temple locations, view available temples, and choose suitable darshan slots.
    - Users will be able to enter their personal details (name and email) and book a slot for

a specific date and time.

## Temple Search and Filter Functionality

* + - Devotees can search temples based on selected geographical locations.
    - A dropdown or list view will allow quick access to temple options.

## Booking Management System

* + - Bookings will be stored in a structured database using SQL.
    - Each booking will be associated with the corresponding temple using foreign keys for easy tracking and management.

## Admin Panel

* + - Admins can log in and add new temples with details like name and location.
    - Admins will also have access to view all bookings in a tabular format for better oversight.

## Database Connectivity and Validation

* + - PHP will be used to connect the frontend with the MySQL backend.
    - Input data will be validated to ensure correctness and security.

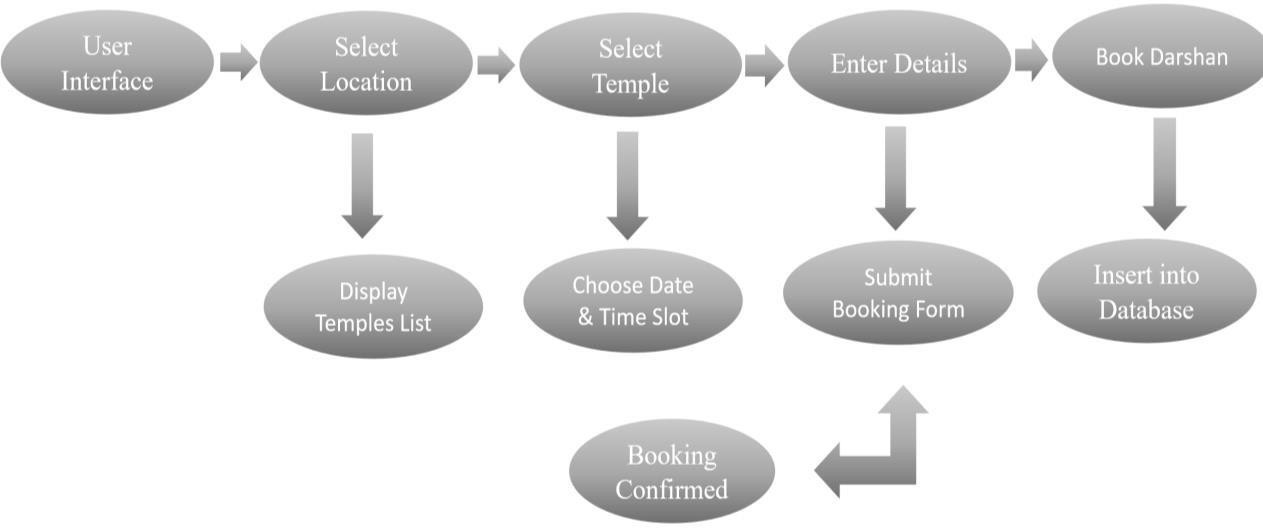
## Booking View for Users

* + - Users can enter their name and email to view previously made bookings.
    - This feature adds transparency and allows users to confirm or reference their slot.

## Styling and Usability

* + - The application will include consistent styling using CSS to enhance usability and visual appeal.
    - Pages such as index.php, add\_temple.php, book\_darshan.php, and view\_bookings.php will all maintain a uniform design.

# BLOCK DIAGRAM

****

**Fig 2.2.1 Block diagram of Temple darshan website**

# CHAPTER 3 MODULE DESCRIPTION

## User Interface Module

This module is the front-end of the application, where users interact with the system through web pages. It includes HTML forms, CSS for styling, and navigation elements. Pages like index.html, view\_temples.html, and book\_darshan.html are part of this module. The UI ensures users can easily navigate, select temples, and book darshan slots.

## Temple Management Module

This module allows the admin to add new temples to the database. Through the add\_temple.html form and its processing script add\_temple.php, admins can input temple names and their locations. This data is stored in the temples table in the database and dynamically displayed to users during booking.

## Darshan Booking Module

The core of the system, this module enables users to book darshan appointments. It handles input of user details, date selection, time slot choice, and submission. The backend (book\_darshan.php) processes the form, inserts the data into the bookings table, and confirms the booking.

## View Temples Module

This module displays a list of temples based on the user-selected location. It fetches temple data from the database (view\_temples.php) and shows it dynamically, helping users decide where they want to book their darshan.

## Booking Status Module

This module allows users to view their past bookings by entering their name and email. The backend (view\_bookings.php) fetches and displays bookings related to the user from the database, adding transparency and trust to the system.

## 

## Admin Module

Used only by authorized personnel, this module includes functionalities for viewing all bookings and managing temple data.

It helps ensure that the booking system is up-to-date and that temple authorities can monitor crowd expectations.

## 3.7 Database Module

This is the backend data handling layer where all temple and booking details are stored. It includes two main tables:

* + - temples for storing temple details
    - bookings for storing user booking information

This module is managed using SQL queries in PHP scripts.

# 

# CHAPTER 4

**CONCLUSION AND FUTURE ENHANCEMENT**

**4.1 CONCLUSION**

The Temple Darshan Booking System presents a streamlined and efficient solution for managing temple visit bookings. By digitizing the darshan appointment process, the system minimizes manual effort, reduces long queues, and provides a smooth experience for devotees and temple authorities alike. The integration of front-end and back-end technologies such as HTML, CSS, PHP, and MySQL ensures that temple data and bookings are handled securely and efficiently. Devotees can easily view temples by location, select convenient time slots, and book appointments with just a few clicks, while administrators can manage temples and monitor bookings from a centralized platform.

**4.2 FUTURE ENHANCEMENT**

Looking ahead, the system can be further enhanced with several advanced features. Incorporating user registration and login functionality would improve personalization and data security. Real-time slot availability and booking limits can prevent over-crowding and improve time management. Developing a mobile application would increase accessibility, especially for users on the go. Integrating a payment gateway would support paid bookings and donations, while SMS/email notifications can keep users informed. A feedback and rating system could help temple authorities improve services, and adding support for multiple Indian languages would make the platform more inclusive. These enhancements would make the system more comprehensive, scalable, and adaptable for future use in temples across the country.

**APPENDIX A (Project Source Code)**

### SQL

-- Step 1: Create the Database

CREATE DATABASE IF NOT EXISTS temple\_darshan; USE temple\_darshan;

-- Step 2: Create 'temples' table CREATE TABLE temples (

id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(255) NOT NULL,

location VARCHAR(255) NOT NULL

);

-- Step 3: Create 'bookings' table CREATE TABLE bookings (

id INT AUTO\_INCREMENT PRIMARY KEY,

temple\_id INT NOT NULL,

name VARCHAR(255) NOT NULL, email VARCHAR(255) NOT NULL, date DATE NOT NULL,

time\_slot VARCHAR(100) NOT NULL,

FOREIGN KEY (temple\_id) REFERENCES temples(id) ON DELETE CASCADE

);

-- Step 4: Create 'users' table (for login/registration) CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL UNIQUE, password VARCHAR(255) NOT NULL

);

-- Step 5: Insert sample temples (Optional demo data) INSERT INTO temples (name, location) VALUES ('Meenakshi Amman Temple', 'Madurai'), ('Brihadeeswara Temple', 'Thanjavur'),

('Kashi Vishwanath Temple', 'Varanasi'), ('Somnath Temple', 'Gujarat'), ('Jagannath Temple', 'Puri');

## Index.php

<?php

// Optionally, you can include a header or other common elements here.

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Temple Darshan System</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

nav {

margin-top: 20px;

}

nav a {

margin: 15px; padding: 10px 20px;

background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 18px;

}

nav a:hover {

background-color: #45a049;

}

footer {

background-color: #4CAF50; color: white;

padding: 10px; position: fixed; width: 100%;

bottom: 0;

}

</style>

</head>

<body>

<header>

<h1>Welcome to the Temple Darshan System</h1>

<p>Your online platform for temple darshan and bookings.</p>

</header>

<nav>

<a href="add\_temple.php">Add Temple</a>

<a href="view\_temples.php">View Temples</a>

<a href="view\_bookings.php">View Bookings</a>

</nav>

<footer>

<p>&copy; 2025 Temple Darshan System</p>

</footer>

</body>

</html>

## Login.php

<?php

session\_start(); // Start session to track login status

// Check if the user is already logged in, if so, redirect to dashboard if (isset($\_SESSION['username'])) {

header("Location: index.php"); // Replace with your logged-in page exit();

}

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$error\_message = "";

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

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

$password = md5($\_POST['password']); // Hashing password using MD5 (use bcrypt in production)

// Query to check the credentials

$sql = "SELECT \* FROM users WHERE username = '$username' AND password = '$password'";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

// User found, start session and redirect to dashboard

$\_SESSION['username'] = $username; header("Location: index.php");

exit();

} else {

// Invalid credentials

$error\_message = "Invalid username or password.";

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Login - Temple Darshan</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

.container {

margin-top: 50px; padding: 20px; background-color: #fff; border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); width: 300px;

margin-left: auto; margin-right: auto;

}

.container input[type="text"], .container input[type="password"] { width: 100%;

padding: 10px; margin: 10px 0; border-radius: 5px; border: 1px solid #ccc;

}

.container input[type="submit"] { background-color: #4CAF50; color: white;

padding: 10px 20px; border: none; border-radius: 5px; font-size: 16px; cursor: pointer;

}

.container input[type="submit"]:hover { background-color: #45a049;

}

.error {

color: red;

margin-top: 10px;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

</header>

<div class="container">

<h2>Login</h2>

<!-- Display error message if login fails -->

<?php if (isset($error\_message)): ?>

<div class="error"><?php echo $error\_message; ?></div>

<?php endif; ?>

<!-- Login form -->

<form method="POST" action="login.php">

<input type="text" name="username" placeholder="Enter your username" required>

<input type="password" name="password" placeholder="Enter your password" required>

<input type="submit" value="Login">

</form>

<p>Don't have an account? <a href="register.php">Register here</a></p>

</div>

</body>

</html>

## Register.php

<?php session\_start();

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$message = "";

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

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

$password = md5($\_POST['password']); // Hashing password using MD5 for simplicity, use bcrypt in production

// Check if the username already exists

$sql = "SELECT \* FROM users WHERE username = '$username'";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

$message = "Username already taken. Please choose another one.";

} else {

// Insert the new user into the database

$sql = "INSERT INTO users (username, password) VALUES ('$username', '$password')";

if ($conn->query($sql) === TRUE) {

$message = "Registration successful! You can now <a href='login.php'>login</a>.";

} else {

$message = "Error: " . $conn->error;

}

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Register - Temple Darshan</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

.container {

margin-top: 50px; padding: 20px; background-color: #fff; border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); width: 300px;

margin-left: auto; margin-right: auto;

}

.container input[type="text"], .container input[type="password"] { width: 100%;

padding: 10px; margin: 10px 0; border-radius: 5px; border: 1px solid #ccc;

}

.container input[type="submit"] { background-color: #4CAF50; color: white;

padding: 10px 20px; border: none; border-radius: 5px;

font-size: 16px;

cursor: pointer;

}

.container input[type="submit"]:hover { background-color: #45a049;

}

.error {

color: red;

margin-top: 10px;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

</header>

<div class="container">

<h2>Register</h2>

<!-- Display message after registration attempt -->

<?php if (isset($message)): ?>

<div class="error"><?php echo $message; ?></div>

<?php endif; ?>

<!-- Registration form -->

<form method="POST" action="register.php">

<input type="text" name="username" placeholder="Enter your username" required>

<input type="password" name="password" placeholder="Enter your password" required>

<input type="submit" value="Register">

</form>

<p>Already have an account? <a href="login.php">Login here</a></p>

</div>

</body>

</html>

## Add\_temple.php

<?php

// Check if the form is submitted

if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {

// Database connection details

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Get form data

$temple\_name = $\_POST['temple\_name'];

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

// Insert the temple into the database

$sql = "INSERT INTO temples (name, location) VALUES ('$temple\_name', '$location')";

if ($conn->query($sql) === TRUE) {

$message = "Temple added successfully!";

} else {

$message = "Error: " . $conn->error;

}

// Close the connection

$conn->close();

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Add Temple - Temple Darshan System</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

nav {

margin-top: 20px;

}

nav a {

margin: 15px; padding: 10px 20px;

background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 18px;

}

nav a:hover {

background-color: #45a049;

}

.form-container { background-color: #fff; border: 1px solid #ddd; margin-top: 30px; padding: 20px;

border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); max-width: 500px;

margin-left: auto;

margin-right: auto;

}

.form-container input[type="text"] { width: 100%;

padding: 10px; margin: 10px 0; border-radius: 5px;

border: 1px solid #ddd;

}

.form-container input[type="submit"] { background-color: #4CAF50;

color: white; border: none; padding: 10px 20px; border-radius: 5px; font-size: 16px; cursor: pointer;

}

.form-container input[type="submit"]:hover { background-color: #45a049;

}

.message {

margin-top: 20px; font-size: 18px;

}

footer {

background-color: #4CAF50; color: white;

padding: 10px; position: fixed; width: 100%;

bottom: 0;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

<p>Add a new temple to the system</p>

</header>

<nav>

<a href="index.php">Home</a>

<a href="view\_temples.php">View Temples</a>

<a href="view\_bookings.php">View Bookings</a>

</nav>

<div class="form-container">

<h2>Add Temple</h2>

<!-- Display the success or error message -->

<?php if (isset($message)): ?>

<p class="message"><?php echo $message; ?></p>

<?php endif; ?>

<!-- Temple addition form -->

<form action="add\_temple.php" method="POST">

<label for="temple\_name">Temple Name:</label><br>

<input type="text" id="temple\_name" name="temple\_name" required><br>

<label for="location">Location:</label><br>

<input type="text" id="location" name="location" required><br>

<input type="submit" value="Add Temple">

</form>

</div>

<footer>

<p>&copy; 2025 Temple Darshan System</p>

</footer>

</body>

</html>

## View\_temples.php

<?php

// Database connection

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Fetch all temples from the database

$sql = "SELECT \* FROM temples";

$result = $conn->query($sql);

// Close connection

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>View Temples - Temple Darshan System</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

nav {

margin-top: 20px;

}

nav a {

margin: 15px; padding: 10px 20px;

background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 18px;

}

nav a:hover {

background-color: #45a049;

}

.temple-list { margin-top: 30px; padding: 20px;

}

.temple-item { background-color: #fff; border: 1px solid #ddd; margin: 10px; padding: 15px;

border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1);

}

footer {

background-color: #4CAF50; color: white;

padding: 10px; position: fixed; width: 100%;

bottom: 0;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

<p>View all temples in your area</p>

</header>

<nav>

<a href="index.php">Home</a>

<a href="add\_temple.php">Add Temple</a>

<a href="view\_bookings.php">View Bookings</a>

</nav>

<div class="temple-list">

<h2>List of Temples</h2>

<?php

// Check if there are any temples in the database if ($result->num\_rows > 0) {

// Loop through all temples and display them while($row = $result->fetch\_assoc()) {

echo '<div class="temple-item">'; echo '<h3>' . $row['name'] . '</h3>';

echo '<p><strong>Location:</strong> ' . $row['location'] . '</p>'; echo '<a href="book\_darshan.php?temple\_id=' . $row['id'] . '">Book

Darshan</a>';

echo '</div>';

}

} else {

echo '<p>No temples found in the database.</p>';

}

?>

</div>

<footer>

<p>&copy; 2025 Temple Darshan System</p>

</footer>

</body>

</html>

## Select\_location.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Fetch unique locations

$sql = "SELECT DISTINCT location FROM temples";

$result = $conn->query($sql);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Select Location</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1>Temple Darshan - Select Location</h1>

</header>

<section>

<form action="select\_temple.php" method="GET">

<label>Select Location:</label><br>

<select name="location" required>

<option value="">--Select--</option>

<?php

while($row = $result->fetch\_assoc()) {

echo "<option value='" . $row['location'] . "'>" . $row['location'] . "</option>";

}

?>

</select><br><br>

<button type="submit">Show Temples</button>

</form>

</section>

<footer>

<p>© 2025 Temple Darshan</p>

</footer>

</body>

</html>

<?php

$conn->close();

?>

## Select\_temple.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$location = $\_GET['location'];

$sql = "SELECT \* FROM temples WHERE location = '$location'";

$result = $conn->query($sql);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Select Temple</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1>Temples in <?php echo htmlspecialchars($location); ?></h1>

</header>

<section>

<form action="book\_darshan.php" method="GET">

<input type="hidden" name="location" value="<?php echo $location; ?>">

<label>Select Temple:</label><br>

<select name="temple\_id" required>

<option value="">--Select Temple--</option>

<?php

while($row = $result->fetch\_assoc()) {

echo "<option value='" . $row['id'] . "'>" . $row['name'] . "</option>";

}

?>

</select><br><br>

<button type="submit">Book Darshan</button>

</form>

</section>

<footer>

<p>© 2025 Temple Darshan</p>

</footer>

</body>

</html>

<?php

$conn->close();

?>

## Book\_darshan.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$temple\_id = $\_GET['temple\_id'];

$sql = "SELECT name FROM temples WHERE id = $temple\_id";

$result = $conn->query($sql);

$row = $result->fetch\_assoc();

$temple\_name = $row['name'];

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Book Darshan</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1>Book Darshan at <?php echo htmlspecialchars($temple\_name); ?></h1>

</header>

<section>

<form action="confirm\_booking.php" method="POST">

<input type="hidden" name="temple\_id" value="<?php echo $temple\_id; ?>">

<label>Your Name:</label><br>

<input type="text" name="name" required><br><br>

<label>Your Email:</label><br>

<input type="email" name="email" required><br><br>

<label>Select Date:</label><br>

<input type="date" name="date" required><br><br>

<label>Choose Time Slot:</label><br>

<select name="time\_slot" required>

<option value="">--Select Slot--</option>

<option value="06:00 AM - 07:00 AM">06:00 AM - 07:00 AM</option>

<option value="07:00 AM - 08:00 AM">07:00 AM - 08:00 AM</option>

<option value="08:00 AM - 09:00 AM">08:00 AM - 09:00 AM</option>

<option value="09:00 AM - 10:00 AM">09:00 AM - 10:00 AM</option>

</select><br><br>

<button type="submit">Confirm Booking</button>

</form>

</section>

<footer>

<p>© 2025 Temple Darshan</p>

</footer>

</body>

</html>

<?php

$conn->close();

## Confirm\_booking.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$temple\_id = $\_POST['temple\_id'];

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

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

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

$time\_slot = $\_POST['time\_slot'];

$sql = "INSERT INTO bookings (temple\_id, name, email, date, time\_slot) VALUES ('$temple\_id', '$name', '$email', '$date', '$time\_slot')";

if ($conn->query($sql) === TRUE) {

$message = "Booking Confirmed!<br>Thank you, $name. Your darshan is booked for

$date at $time\_slot.";

} else {

$message = "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Booking Confirmation - Temple Darshan</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9; text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

nav {

margin-top: 20px;

}

nav a {

margin: 15px; padding: 10px 20px;

background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 18px;

}

nav a:hover {

background-color: #45a049;

}

footer {

background-color: #4CAF50; color: white;

padding: 10px; position: fixed; width: 100%;

bottom: 0;

}

.confirmation-message { margin-top: 30px; padding: 20px; background-color: #fff; border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); max-width: 500px;

margin-left: auto; margin-right: auto;

}

.confirmation-message h2 { color: #4CAF50;

}

.confirmation-message p { font-size: 16px; margin: 15px 0;

}

.confirmation-message a { padding: 10px 20px; background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 16px; display: inline-block;

}

.confirmation-message a:hover { background-color: #45a049;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

</header>

<nav>

<a href="index.php">Home</a>

<a href="view\_temples.php">View Temples</a>

<a href="add\_temple.php">Add Temple</a>

</nav>

<div class="confirmation-message">

<h2>Booking Status</h2>

<p><?php echo $message; ?></p>

<a href="index.php">Back to Home</a>

</div>

<footer>

<p>&copy; 2025 Temple Darshan System</p>

</footer>

</body>

</html>

## View\_bookings.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "temple\_darshan";

$conn = new mysqli($servername, $username, $password, $dbname); if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$message = "";

$bookings = [];

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

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

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

// Query to get bookings based on the user's name and email

$sql = "SELECT b.id, t.name AS temple\_name, b.date, b.time\_slot FROM bookings b

JOIN temples t ON b.temple\_id = t.id

WHERE b.name = '$name' AND b.email = '$email'";

$result = $conn->query($sql); if ($result->num\_rows > 0) {

// Fetch all bookings

while ($row = $result->fetch\_assoc()) {

$bookings[] = $row;

}

} else {

$message = "No bookings found for $name with email $email.";

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>View Bookings - Temple Darshan</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f9;

text-align: center;

}

header {

background-color: #4CAF50; color: white;

padding: 15px;

}

nav {

margin-top: 20px;

}

nav a {

margin: 15px; padding: 10px 20px;

background-color: #4CAF50; color: white;

text-decoration: none; border-radius: 5px; font-size: 18px;

}

nav a:hover {

background-color: #45a049;

}

footer {

background-color: #4CAF50; color: white;

padding: 10px; position: fixed; width: 100%;

bottom: 0;

}

.container {

margin-top: 30px; padding: 20px; background-color: #fff; border-radius: 5px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); max-width: 500px;

margin-left: auto; margin-right: auto;

}

.container input[type="text"], .container input[type="email"] { width: 100%;

padding: 10px; margin: 10px 0; border-radius: 5px; border: 1px solid #ccc;

}

.container input[type="submit"] { background-color: #4CAF50; color: white;

padding: 10px 20px; border: none; border-radius: 5px; font-size: 16px; cursor: pointer;

}

.container input[type="submit"]:hover { background-color: #45a049;

}

.bookings-table { margin-top: 30px; width: 100%;

border-collapse: collapse;

}

.bookings-table th, .bookings-table td { padding: 12px;

text-align: left;

border: 1px solid #ddd;

}

.bookings-table th { background-color: #4CAF50; color: white;

}

.bookings-table tr:nth-child(even) { background-color: #f2f2f2;

}

.message {

margin-top: 20px; color: red;

}

</style>

</head>

<body>

<header>

<h1>Temple Darshan System</h1>

</header>

<nav>

<a href="index.php">Home</a>

<a href="view\_temples.php">View Temples</a>

<a href="add\_temple.php">Add Temple</a>

</nav>

<div class="container">

<h2>View Your Bookings</h2>

<!-- Form to enter name and email -->

<form method="POST" action="view\_bookings.php">

<input type="text" name="name" placeholder="Enter your name" required>

<input type="email" name="email" placeholder="Enter your email" required>

<input type="submit" value="View Bookings">

</form>

<!-- Display message if no bookings found -->

<div class="message">

<?php echo $message; ?>

</div>

<!-- Display bookings if available -->

<?php if (!empty($bookings)): ?>

<table class="bookings-table">

<tr>

<th>Temple Name</th>

<th>Date</th>

<th>Time Slot</th>

</tr>

<?php foreach ($bookings as $booking): ?>

<tr>

<td><?php echo $booking['temple\_name']; ?></td>

<td><?php echo $booking['date']; ?></td>

<td><?php echo $booking['time\_slot']; ?></td>

</tr>

<?php endforeach; ?>

</table>

<?php endif; ?>

</div>

<footer>

<p>&copy; 2025 Temple Darshan System</p>

</footer>

</body>

</html>

# APPENDIX B (OUTPUT SCREENSHOTS)

## 1 ) Register or Login

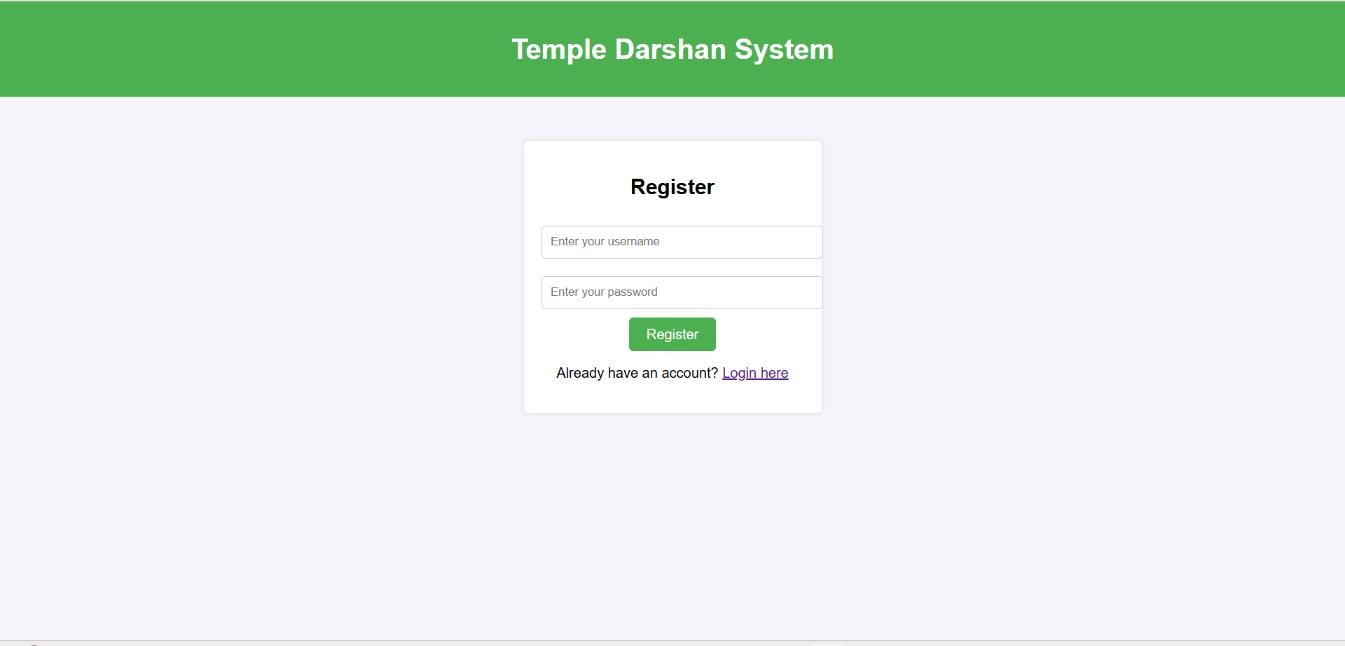
****

Fig 5.2.1 Register or login

## Choose your option

****

Fig 5.2.2 Home page

## Book Darshan

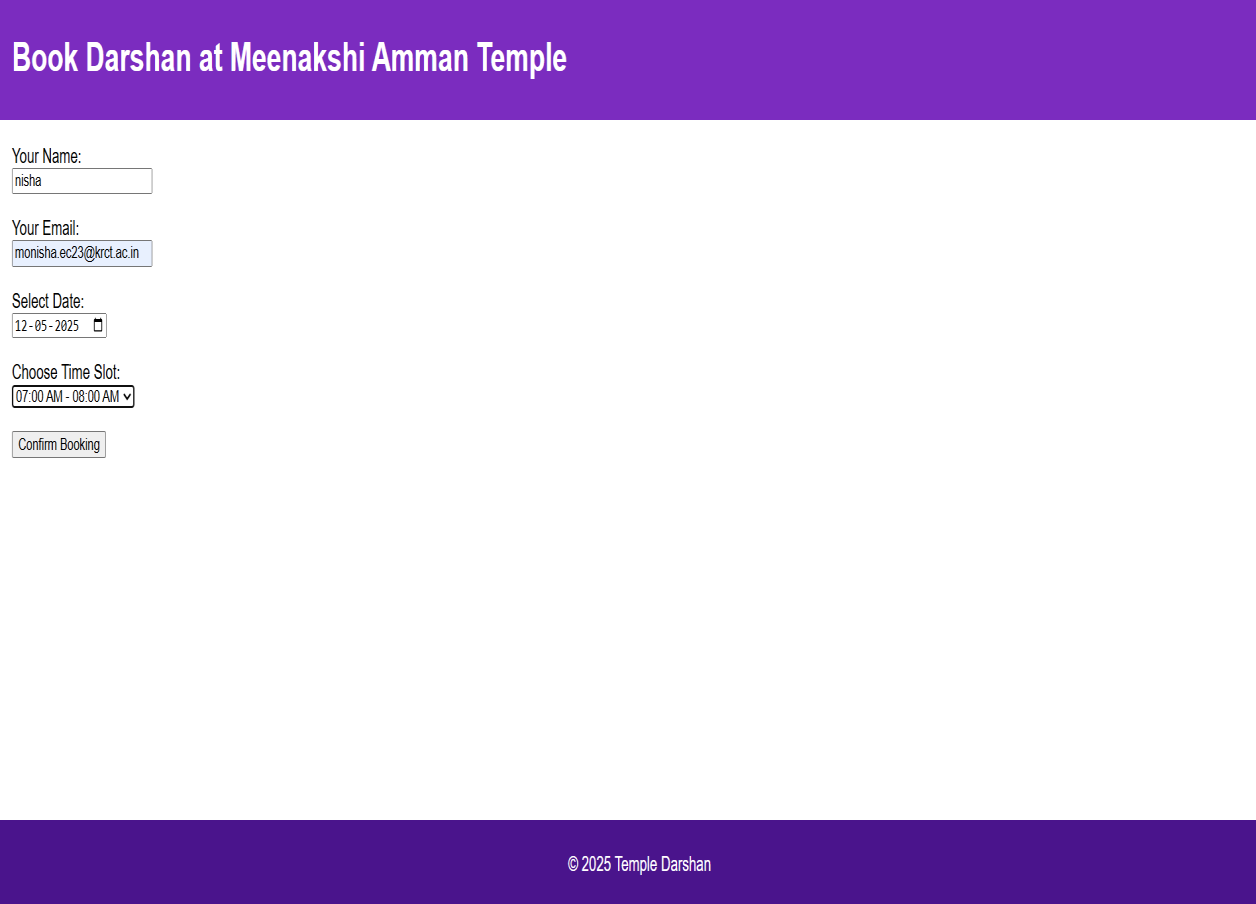
****

Fig 5.2.3 Book darshan page

## Booking Successful

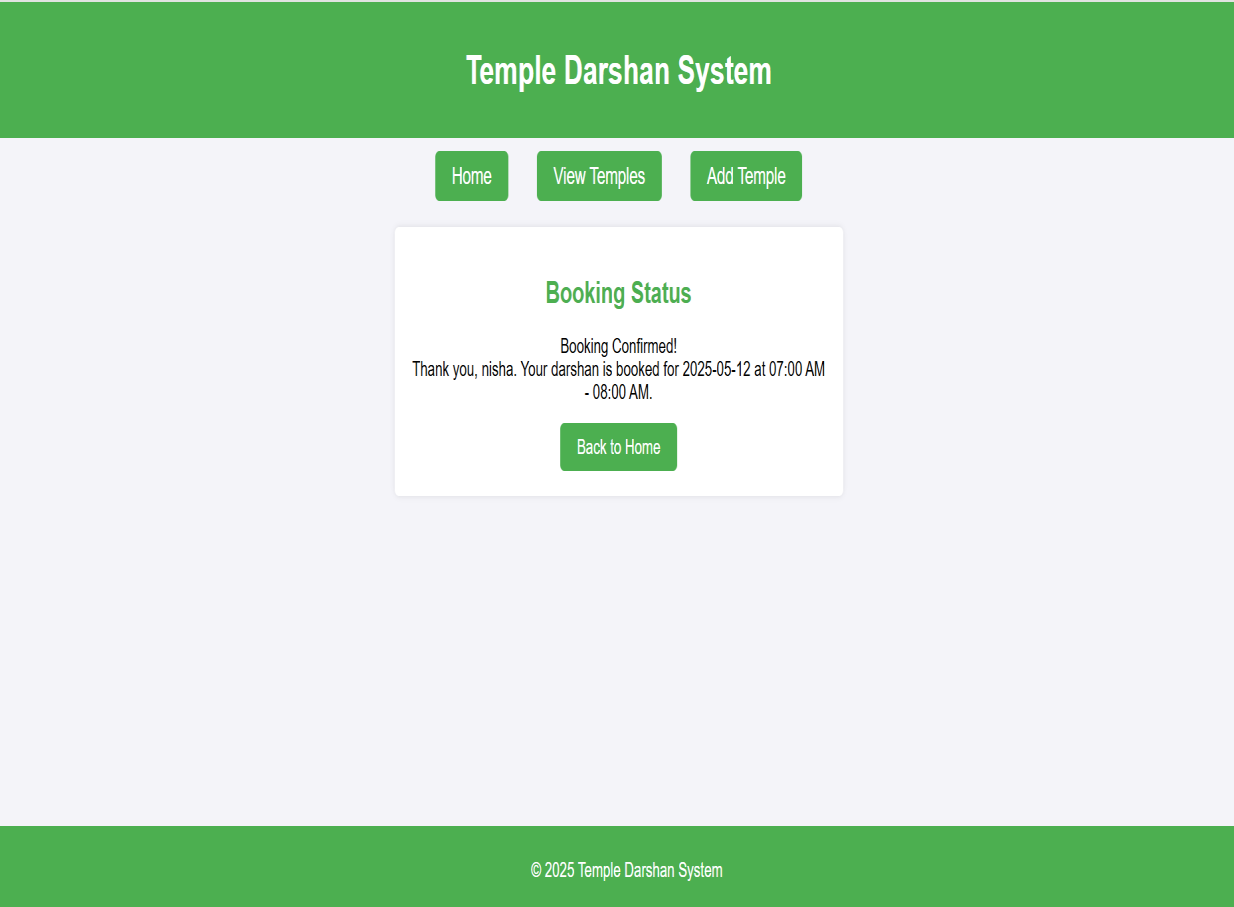
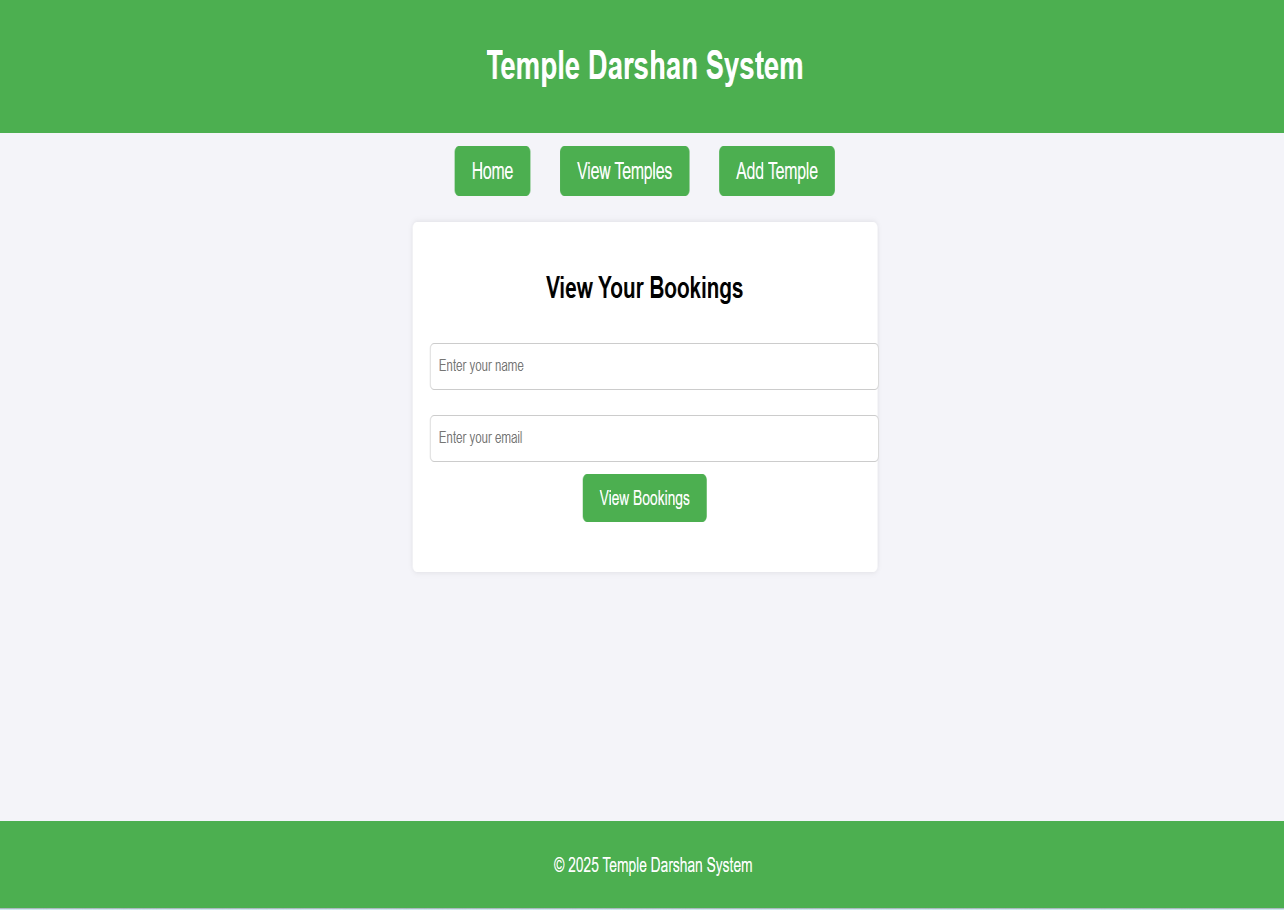
****

Fig 5.2.4 Booking successful page

## View your Bookings

****

**Fig 5.2.5 View your Bookings**

# REFERENCES

* 1. **W3Schools** – [https://www.w3schools.com](https://www.w3schools.com/)

(For learning and referencing HTML, CSS, PHP, and MySQL basics)

* 1. **PHP Manual** – <https://www.php.net/manual/en/>

(Official documentation for PHP functions and database connectivity)

* 1. **MySQL Documentation** – <https://dev.mysql.com/doc/>

(For understanding SQL queries, data types, and database structure)

* 1. **Stack Overflow** – [https://stackoverflow.com](https://stackoverflow.com/)

(For solutions to common coding issues and debugging tips during development)

* 1. **XAMPP Official Website** – [https://www.apachefriends.org](https://www.apachefriends.org/)

(Used for setting up the local server environment for PHP and MySQL integration)

* 1. **Online Tutorials and Guides** from sources like GeeksforGeeks, TutorialsPoint, and YouTube

(To understand web application architecture and dynamic form handling with PHP)