

A
PROJECT REPORT ON

Event Management System

SUBMITTED IN
PARTIAL FULFILLMENT OF

DIPLOMA IN ADVANCED COMPUTING (PG-DAC)



BY

PRANAV ADAGONDA PATIL

UNDER THE GUIDENCE OF

Mrs. Lalita Shinde

AT

**SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY,
PUNE**

**SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY,
PUNE.**



CERTIFICATE

This is to certify that the project

Event Management System

Has been submitted by

PRANAV ADAGONDA PATIL

In partial fulfillment of the requirement for the Course of **PG Diploma in Advanced Computing (PG-DAC Sep 2023)** as prescribed by The **CDAC ACTS, PUNE.**

Place: Pune

Date: 22-FEB-2024

Mrs. Lalita Shinde
Project Guide

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my teacher Lalita Shinde as well as our Director Nitin Kudhale who gave me the golden opportunity to do this wonderful project on the topic User Analytics, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.

Pranav Patil

ABSTRACT

The Event Management System (EMS) is a comprehensive Java 2 Enterprise Edition (J2EE) project designed to streamline and automate the process of organizing and managing events. This system aims to facilitate event planners, organizers, and participants by providing a centralized platform for planning, scheduling, coordinating, and tracking various aspects of events.

INDEX

1.	INTRODUCTION	1
	1.1 Introduction	2
2.	PRODUCT OVERVIEW AND SUMMARY	
	2.1 Purpose	
	2.2 Scope	
	2.3 User Classes and Characteristics	
	2.4 Design and Implementation Constraints	
3.	REQUIREMENTS	
	3.1 Functional Requirements	
	3.1.1 Use case for Administrator.	
	3.1.2 Use case for User.	
	3.2 Non - Functional Requirements	
	3.2.1 Usability Requirement	
	3.2.2 Performance Requirement	
	3.2.3 Reliability Requirement	
	3.2.4 Portability Requirement	
	3.2.5 Security Techniques	
4.	PROJECT DESIGN	
	4.1 Data Model	
	4.1.1 Database Design	
	4.2 Process Model	
	4.2.1 Functional Decomposition Diagram	
	4.2.2 Data Flow Diagram (DFD)	
5.	PROJECT RELATED STATISTICS	
6.	CONCLUSION	

LIST OF TABLES

Section	Table Title	Page
Fig 3	Complete Database	

LIST OF FIGURES

[illegible]

INTRODUCTION

The Event Management System (EMS) is a sophisticated software solution developed using Java 2 Enterprise Edition (J2EE) technology, aimed at revolutionizing the way events are planned, organized, and executed. In today's fast-paced world, efficient event management is crucial for businesses, organizations, and individuals alike. Whether it's a corporate conference, a music concert, a sports tournament, or a community gathering, coordinating various aspects of an event can be complex and time-consuming. EMS addresses these challenges by providing a centralized platform that automates and streamlines the entire event management process.

This introduction will provide an overview of the objectives, significance, and key features of the EMS project, highlighting its potential benefits for event organizers, participants, and stakeholders.

Objectives:

The primary objective of the EMS project is to develop a comprehensive software solution that simplifies event management tasks while enhancing efficiency, accuracy, and productivity. Key objectives include:

Streamlining Event Planning: EMS aims to streamline the process of planning events by providing intuitive tools for creating event schedules, managing resources, and coordinating logistics.

Product Overview and Summary

Purpose:

The Event Management System (EMS) is designed to revolutionize the way events are planned, organized, and executed. Its primary purpose is to provide a comprehensive software solution that streamlines the entire event management process, from event creation and promotion to participant registration and feedback collection. EMS aims to enhance efficiency, accuracy, and productivity for event organizers while improving the overall experience for participants.

Scope:

EMS encompasses a wide range of features and functionalities aimed at meeting the diverse needs of event planners, organizers, participants, and stakeholders. Its scope includes event creation and management, registration and ticketing, venue management, communication and notifications, schedule integration, reporting and analytics, and security and access control. The system is scalable and adaptable to various types and sizes of events, from small corporate meetings to large-scale conferences or festivals.

User Classes and Characteristics:

1.Administrators:

Administrators have full access to the EMS system and are responsible for managing user accounts, roles, and permissions.

They can create and update events, manage venues, generate reports, and oversee system security and compliance.

Characteristics: Administrators possess advanced technical skills and have a deep understanding of event management principles and practices.

2.Event Organizers:

Event organizers are responsible for planning, coordinating, and executing events using the EMS platform.

They can create new events, specify event details, manage event schedules, communicate with participants, and analyze event performance.

Characteristics: Event organizers are detail-oriented individuals with strong organizational and communication skills. They have a creative mindset and thrive in fast-paced environments.

3.Participants:

Participants are individuals or groups who register for and attend events managed through the EMS system.

They can browse upcoming events, view event details, register for events, purchase tickets, receive event notifications, and provide feedback.

Characteristics: Participants come from diverse backgrounds and may have varying levels of technological proficiency. They value convenience, accessibility, and engagement in the events they attend.

| Design and Implementation Constraints

- User Interface

Intuitive Navigation: Clear and intuitive navigation aids user exploration.

Consistent Design: Uniform layout and visual elements ensure coherence.

Interactive Elements: Engaging UI components enhance user interaction.

Visual Feedback: Instant feedback informs users about actions and outcomes.

Mobile Responsiveness: Adaptable design ensures usability across devices.

- Tab Design:

Tabs are a common UI pattern used to organize content and facilitate navigation within a single interface. Key principles for designing tabs in the EMS UI include:

Clear Labeling:

Each tab should have a clear and descriptive label that succinctly conveys its purpose or content, helping users understand the context and relevance of each tab.

Logical Grouping:

Tabs should be logically grouped based on related functionality or content categories, making it easier for users to locate and access relevant information.

Visual Differentiation:

Active and inactive tabs should be visually differentiated to indicate the user's current location and provide feedback on their navigation actions.

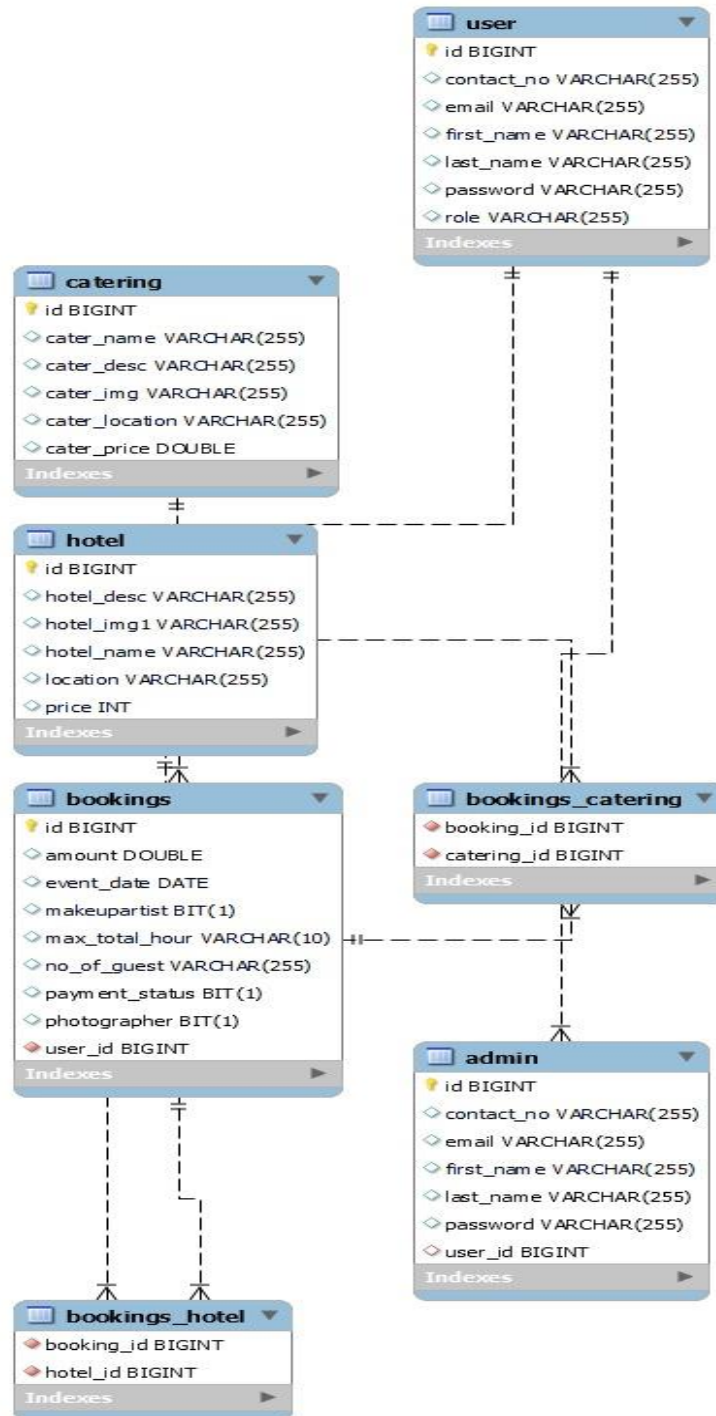
Tab Interaction:

Tabs should respond to user interaction, such as hovering or clicking, by changing appearance or revealing additional options or content.

Limited Number:

Functional Requirements

| Use Case Diagram:



Non - Functional Requirements

Usability Requirement:

Intuitive Interface: The UI should be easy to navigate, with clear labels and logically organized features.

Consistent Design: Maintain uniformity in layout, color schemes, and visual elements across all pages.

Minimal Learning Curve: Ensure new users can quickly understand and use the system with minimal training.

Error Prevention: Implement validation checks and clear error messages to help users avoid mistakes.

Responsive Design: Ensure the UI adapts seamlessly to different devices and screen sizes.

Accessibility Compliance: Follow accessibility standards to accommodate users with disabilities.

User Feedback: Incorporate mechanisms for users to provide feedback, enhancing continuous improvement.

Data Model

The data model for the Event Management System (EMS) encompasses various entities and their relationships to represent the system's data structure effectively. Here's a simplified representation of the data model:

Entities:

User:

(User ID, Username, Password, Email, Role)

Event:

(Event ID, Organizer ID, Event Name, Description, Date, Time, Location, Category, Status)

Venue:

(Venue ID, Venue Name, Address, Capacity)

Relationships:

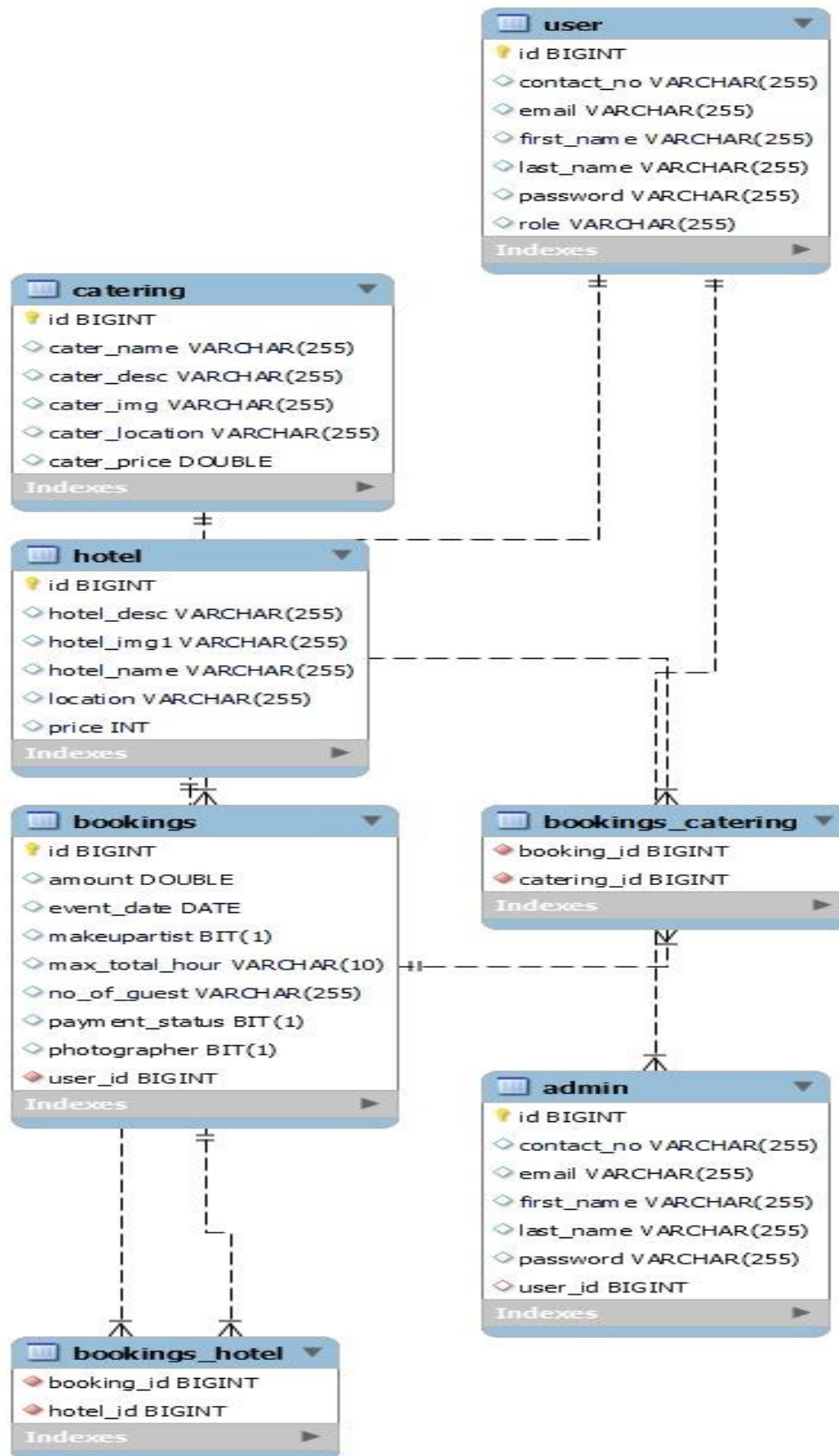
One-to-Many: User to Event (Organizer)

One-to-Many: Event to Venue

One-to-Many: Event to Ticket

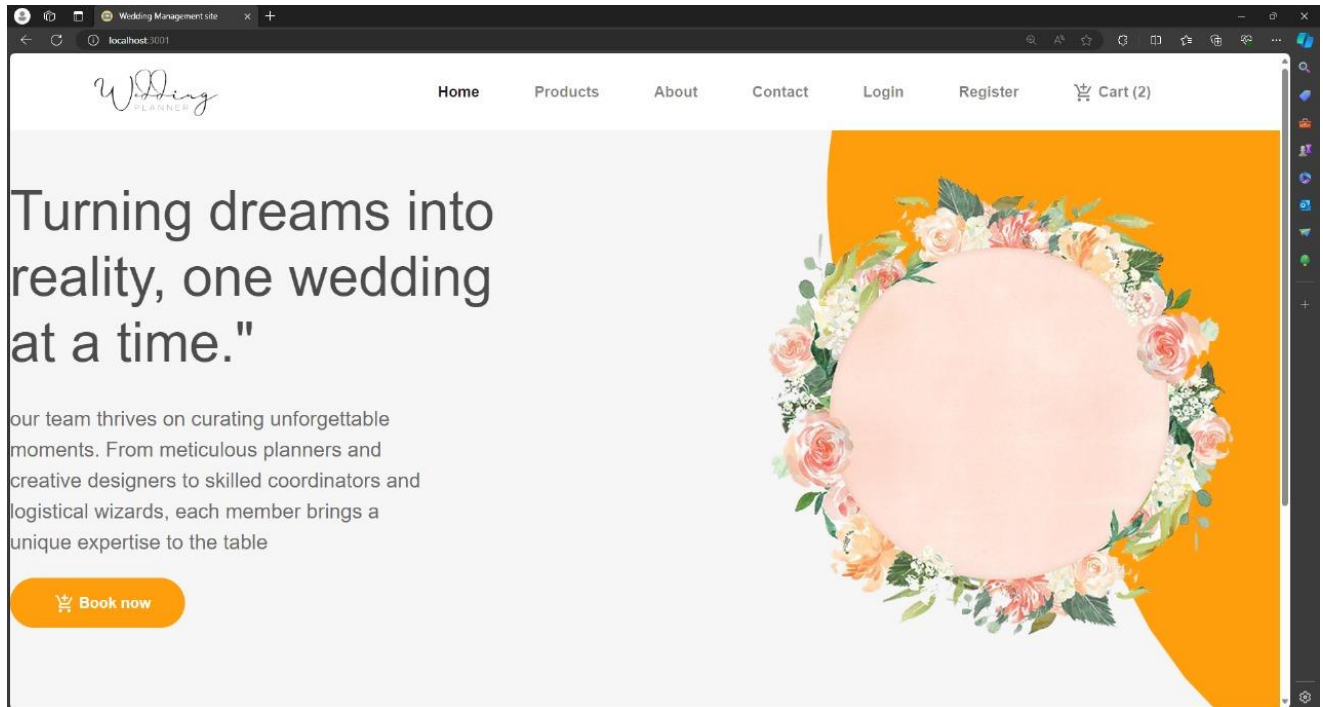
One-to-Many: User to Ticket (Participant)

This data model provides a foundation for storing and managing user accounts, events, venues, and tickets within the EMS. Additional attributes and entities may be incorporated based on specific requirements and functionalities of the system.



Complete Database

Screen Shots



[Home](#)[Services](#)[About](#)[Contact](#)[Login](#)[Register](#)[Cart \(2\)](#)

Login

Email address

Password

New Here? [Register](#)

Qualtiy

244-5333-7783

[Terms](#)

[Help](#)

[hello@gmail.com](#)

[Privacy](#)

Wedding Management site

localhost:3001/checkout

W.Ding
PLANNER

[Home](#)[Services](#)[About](#)[Contact](#)[Login](#)[Register](#)[Cart \(2\)](#)

Checkout

Booking Details

Event Date

dd-mm-yyyy

total hours

enter hours

no of guest

0-999

Makeupartist

☐

Photography Package

☐

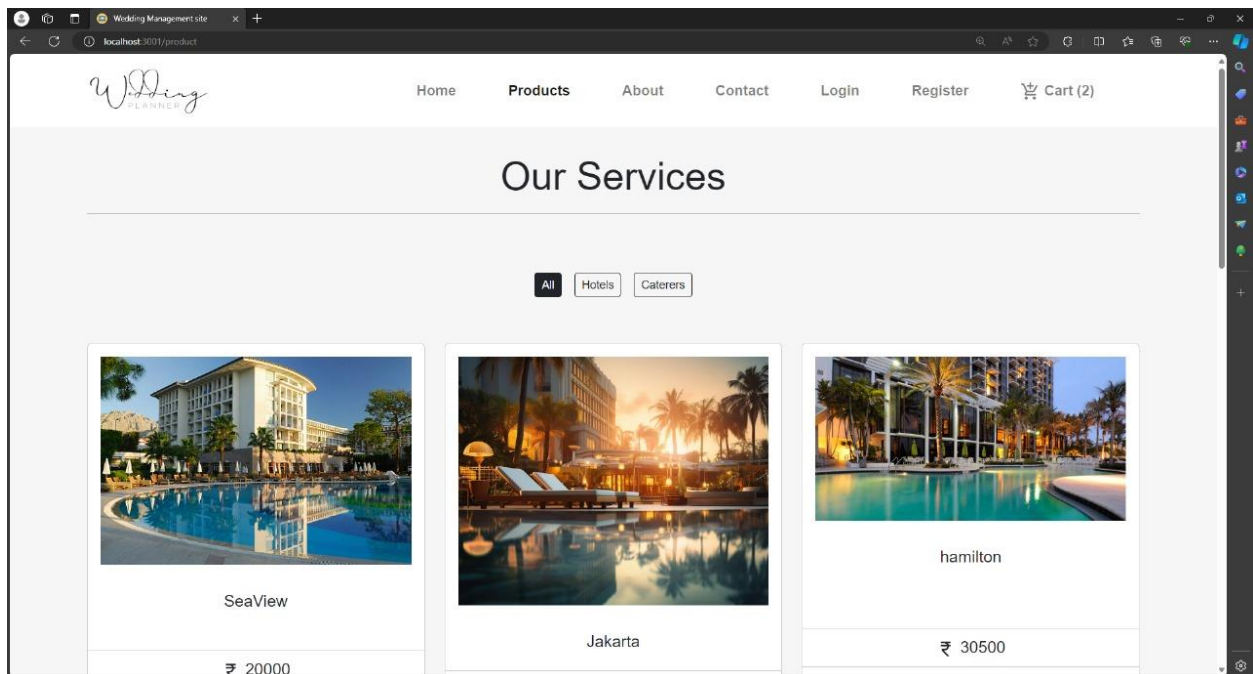
Payment

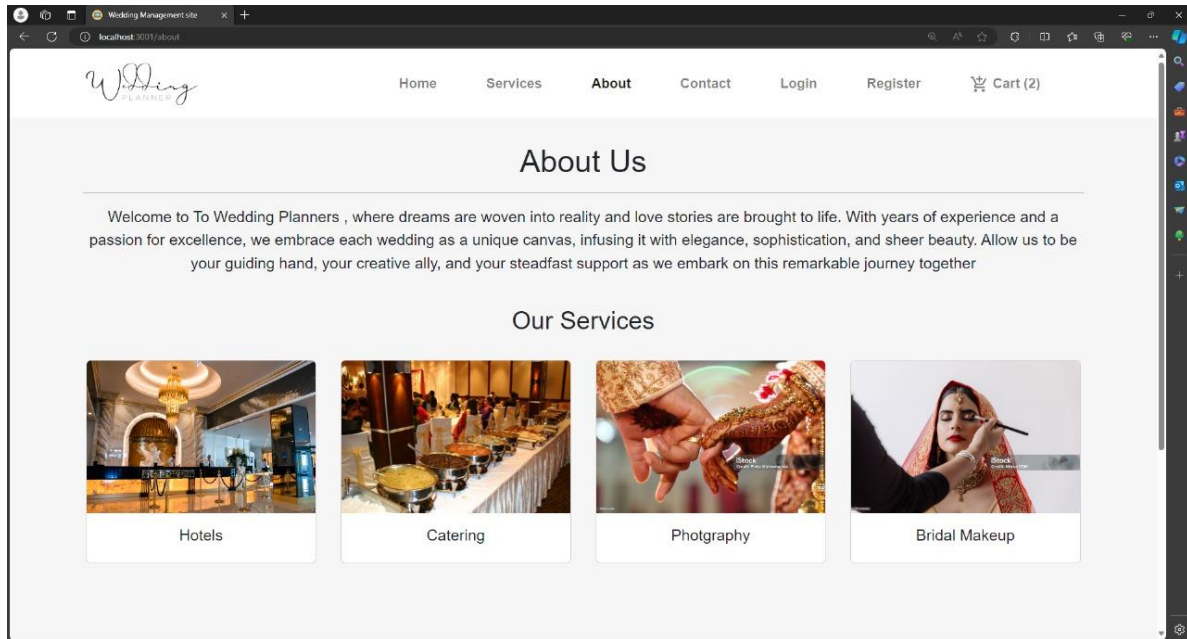
Name on card

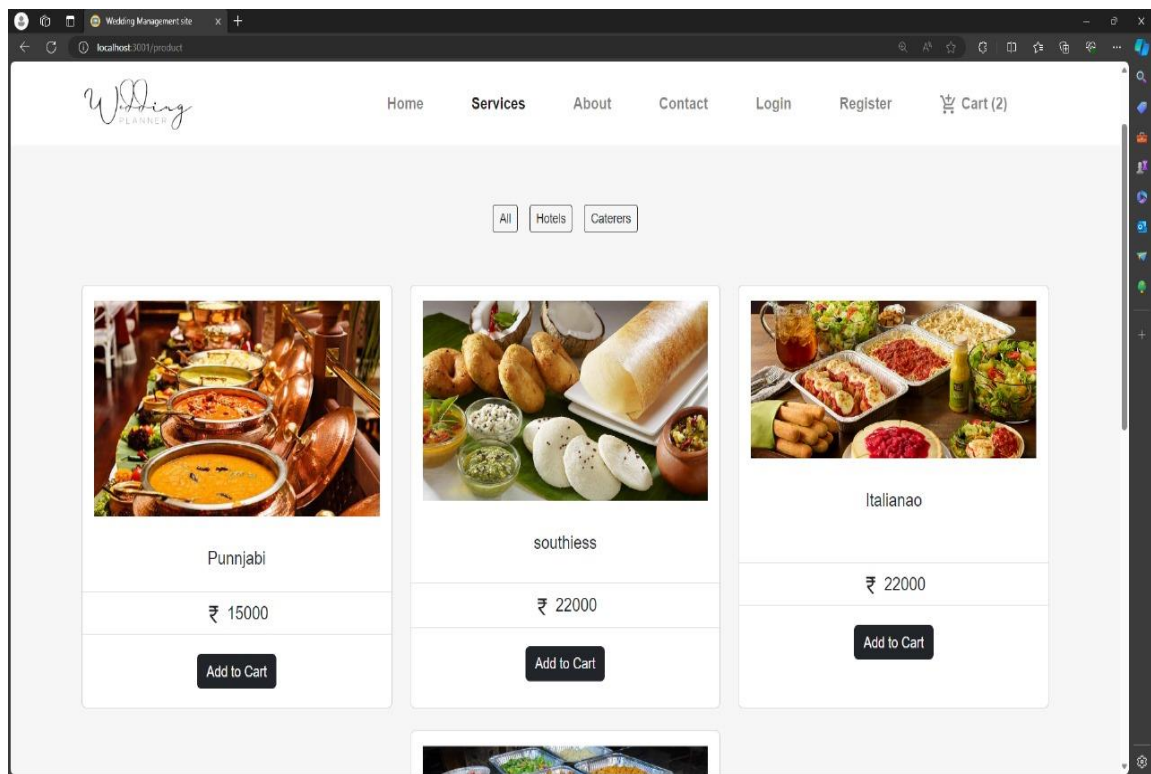
Credit card number

Order Summary

Products (2)	₹ 26500
Total amount	₹ 26500







Widling
PLANNING

[Home](#)[Services](#)[About](#)[Contact](#)[Login](#)[Register](#)[Cart \(2\)](#)

0-999

Makeupartist

☐

Photography Package

☐

Payment

Name on card

Credit card number

Full name as displayed on card

Expiration

CVV

Confirm Booking

| Conclusion: The Event Management System (EMS) offers a streamlined solution for organizing and managing events efficiently. Through its user-friendly interface and robust functionalities, EMS enhances efficiency for organizers and improves the overall experience for participants. With scalability, adaptability, and data-driven insights, EMS is poised to revolutionize event management practices, driving success and engagement in the industry.

| Future Scope:

Personalization: Implementing tailored features based on user preferences to enhance engagement.

Emerging Technologies Integration: Exploring AI, ML, and AR for innovative event experiences.

Social Media Integration: Strengthening social media connectivity for wider event reach and engagement.

IoT for Venue Management: Utilizing IoT devices for optimized venue logistics and resource management.

Sustainability Initiatives: Incorporating eco-friendly practices to promote sustainable events.

Mobile App Development: Creating dedicated mobile apps for convenient event access and interaction.

Market Expansion: Diversifying into new markets with customized offerings to attract a broader audience.

| References:

Oracle's Java EE Documentation: <https://www.oracle.com/java/technologies/java-ee-glance.html> - Oracle's official documentation provides comprehensive information on Java EE technologies, including Servlets, Java Server Pages (JSP), and Java Database Connectivity (JDBC).

W3Schools: <https://www.w3schools.com/> - W3Schools offers tutorials and references on web development technologies such as HTML, CSS, and JavaScript, which are essential for building the user interface of your EMS.

ACM Digital Library: <https://dl.acm.org/> - The ACM Digital Library contains a vast collection of academic journals, conference proceedings, and research papers on various topics, including event management systems and related fields.

IEEE Xplore Digital Library: <https://ieeexplore.ieee.org/> - IEEE Xplore provides access to a wide range of scholarly articles, conference papers, and standards in the field of technology and engineering, which may include relevant research on event management systems.

GitHub: <https://github.com/> - GitHub is a platform where you can find open-source projects related to event management systems, web development, and Java programming. You may find useful code snippets, libraries, and frameworks that can accelerate your project development.