EVENTOPLANNEUR
OFFICIAL EVENT MANAGEMENT SYSTEM

Mini Project Report

Submitted by

ELIZEBATH THOMAS

Reg. No.: AJC19MCA-I024

In Partial fulfillment for the Award of the Degree of

INTEGRATED MASTER OF COMPUTER APPLICATIONS

(INMCA)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with ‘A’ grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

2023-2024

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

CERTIFICATE

This is to certify that the Project report, "**EVENTOPLANNEUR: OFFICIAL EVENT MANAGEMENT SYSTEM**" is the bona fide work of **ELIZEBATH THOMAS (Reg No: AJC19MCA-I024)** in partial fulfillment of the requirements for the award of the Degree of Integrated Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2023-24.

Dr. Paulin Paul	**Ms. Meera Rose Mathew**
Internal Guide	**Coordinator**

Rev. Fr. Dr. Rubin Thottupurathu Jose

Head of the Department

DECLARATION

I hereby declare that the project report **"EVENTOPLANNEUR: OFFICIAL EVENT MANAGEMENT SYSTEM"** is a bona fide work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Integrated Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2023-2024.

Date: 30-10-2023 **ELIZEBATH THOMAS**

KANJIRAPPALLY **Reg: AJC19MCA-I024**

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the project. I take this opportunity to express my gratitude to all who helped me in completing this project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to beloved Manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Lillykutty Jacob** for providing good faculty for guidance.

I owe a great depth of gratitude and appreciation towards our Head of the Department **Rev.Fr.Dr. Rubin Thottupurathu Jose** for helping us. I extend my whole hearted thanks to the project coordinator **Ms. Meera Rose Mathew** for her valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the project. I would also express sincere gratitude to my guide **Dr. Paulin Paul** for her inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the project. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the project. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

ELIZEBATH THOMAS

ABSTRACT

EventoPlanneur is created to organize four categories of events such as webinars, conferences, exhibitions and general social events like campaigns, reunions, awareness programs. The webinars, conferences and exhibitions are meant for academic, research and technical audience. The proposed system EventoPlanneur is a comprehensive platform which facilitates planning, registration, publicity of professional events (webinars, conferences, exhibitions) by registered organizations/agencies to registered audience by matching the domain of registered events and the specific title among the internet audience using techniques like Natural Language Processing. In addition to registered users, organizers/agencies/normal users(customers) also have the provision to view and book facilities like Venue, Sound and Lighting, Caterers, Media for social events like public meetings and get-togethers. Based on the need auto recommendation of list of services based on location will be intimated to the customer.

This project is mainly divided into four modules:

- Admin
- Organizer/Agency
- Service provider
- Attendee

The system has many advantages like efficient event planning and real-time communication. NLP enhances search, recommendations, sentiment analysis, and automated information extraction. The entire draft is for main project out of which the mini project includes functions such as user (Admin, Organizer, Attendee, Service Provider) registration and profile management, event registration, an automated email system, display organizers and list events.

.

CONTENT

List of Abbreviation

HTML - Hyper Text Markup Language

CSS - Cascading Style Sheet

UML - Unified Modeling Language

JS - JavaScript

AJAX - Asynchronous JavaScript and XML

DOM - Document Object Model

DBMS - Database Management System

1NF – First Normal Form

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

The website is a complete platform designed to make event management and coordination easier. It serves event organizers, attendees, and service providers and offers user-friendly interfaces for easy navigation. Key features include event creation, registration, and service booking, facilitated by real-time notifications and advanced search algorithms. The technical and operational feasibility of the project is ensured by a skilled development team and a thorough feasibility analysis. With financial viability supported by a cost-benefit assessment, the site promises to revolutionize event planning and provide a seamless experience for all users.

1.2 PROJECT SPECIFICATION

The Event Management Website is a dynamic platform designed to connect attendees, organizers, and service providers for various events. The primary objectives are to allow organizers to register events, attendees to register for recommended events based on their interests, and to ensure the verification of college organizers using AICTE (All India Council for Technical Education) ID.

The system modules are:

1. **Event Organizer Module**

   An essential part of the Event Management Website for people or organizations wishing to organize events is the Organizer Module. Event planners can effortlessly register and manage their events using this module. They may include essential event information, such event titles, dates, time, deadline, description, speakers, venues. In addition, event listings can be edited by organizers to guarantee correctness and relevancy. All things considered, the Organizer Module is the foundation for organizing and creating events. With the help of the AICTE ID verification for college organizers, it provides a highly credible interface that is easy to use for organizers to establish, manage, and promote their events.

2. Attendee Module

The Attendee Module is a user-centric component of the Event Management Website, tailored to individuals seeking to explore and engage with a wide array of events. Attendees play a pivotal role in the platform's ecosystem, and this module provides them with a seamless and enriching experience. Key functionalities of the Attendee Module include user registration and event discovery. Attendees can register on the platform, creating personalized profiles that reflect their interests and preferences. These profiles serve as the foundation for the platform's event recommendation system. Attendees can effortlessly browse, search, and discover events based on their specified interests. The system employs NLP based recommendation algorithms to suggest events that align with their preferences, ensuring that attendees can easily find events that resonate with their interests.

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

In today's fast-paced digital landscape, the need for efficient and user-friendly event management solutions is more pressing than ever. EventoPlanneur, our innovative event management platform, aspires to provide an all-encompassing experience for attendees, organizers, and service providers.

2.2 EXISTING SYSTEM

The existing event management system is a basic framework for event hosting and user interaction. Event organizers can list events with limited customization, while attendees have minimal personalization options. The system lacks advanced recommendation algorithms and a verification process for college organizers. Data handling, user experience, and security measures need enhancement. The proposed system study aims to address these limitations and pave the way for a more user-centric, efficient, and secure event management platform.

2.3 DRAWBACKS OF EXISTING SYSTEM

- **Customization for Event Organizers:** Event organizers face limitations in customizing and promoting their events, reducing their control and event visibility.

- **Personalization for Attendees:** Attendees have limited options for personalization, hampering their ability to tailor their event experiences.
- **Event Discovery:** Event discovery is hindered by basic recommendation algorithms, limiting the platform's ability to suggest relevant events to users.
- **Verification for College Organizers:** The absence of a verification process for college organizers, particularly using AICTE IDs, compromises the system's trustworthiness.
- **User Experience and Engagement**: Comprehensive improvements are necessary due to the system's poor user experience, weak security protocols, poor data management, scalability issues, and minimal user interaction.

2.4 PROPOSED SYSTEM

The proposed system is defined to meet all the disadvantages of the existing system. With respect to the current configuration, the suggested event management system offers a significant improvement. With strong security features and a user-centric design, it will provide an improved user experience. An increased user base and event listings will be aided by enhanced data management and scalability. Highly relevant event recommendations will be suggested by advanced recommendation algorithms, and confidence will be increased by the process of verification for college organizers and enhancements to legal and security. The platform will be enhanced with additional capabilities for user interaction and service provider integration, guaranteeing a safe and dynamic event management experience.

2.5 ADVANTAGES OF PROPOSED SYSTEM

- **Personalization and Customization:** More choices will be available for attendees and event organizers to customize and personalize their interactions, making the event more enjoyable in the end.
- **Advanced Recommendation Algorithms:** By offering consumers very relevant event ideas, advanced recommendation algorithms will raise the possibility that users will attend and enjoy the event.
- **Trust and Verification:** By adding a thorough verification procedure, such as AICTE ID verification for college organizers, the platform will become more legitimate and trustworthy, which will lower fraudulent activity.
- **Better User Experience:** All users will have a more satisfying and effective experience due to the user-centric design and improved interface, which will raise satisfaction and engagement.
- **Scalability and Performance:** A bigger audience and more event listings will be supported by the system's optimized scalability and performance, guaranteeing a smooth and responsive platform.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

Feasibility is the practical extent to which a project can be successfully executed. It involves performing a feasibility study to determine the viability and workability of a proposed solution for meeting the project's requirements. The feasibility study assesses various factors, including resource availability, software development cost estimation, post-development benefits to the organization, and maintenance costs. During the feasibility study, information is gathered to evaluate whether the proposed solution aligns with the business objectives. If the system does not support the business objectives, it lacks real value to the organization. Many organizations develop systems that fail to contribute to their objectives due to unclear statements of objectives, undefined business requirements, or influence from political or organizational factors in the system procurement process. To address these challenges, the feasibility study is a crucial step in the requirements engineering and system development process. It helps in determining the practicality and effectiveness of the proposed solution, ensuring that it aligns with the organization's goals and justifies the investment of resources and efforts.

3.1.1 Economical Feasibility

Economic feasibility refers to the evaluation of whether a proposed system is financially viable and capable of generating sufficient returns to justify its implementation. It involves analyzing the costs and benefits associated with the project, including the initial investment, ongoing expenses, potential revenues, and overall financial impact on the organization.

The economic feasibility of the discussed event management site seems promising, given its comprehensive functionalities. The cost-benefit analysis considered factors such as software development, hardware, staffing, and maintenance costs, while also assessing potential revenue streams from event registrations and service bookings. With user-friendly interfaces, real-time notifications, and advanced search algorithms, the site is expected to attract event organizers, attendees, and service providers, contributing to its financial viability.

3.1.2 Technical Feasibility

Technical feasibility evaluates product or service development with existing technology and resources. It deals with tools, materials, labour, logistics and technology and proactively addresses potential problems. Assessing hardware, software, and the development team ensures stability and user support. Visualizing the process with a flowchart aid in handling technical challenges. Comprehensive assessment ensures successful project implementation within the allocated time and budget.

The event management website's technical feasibility is backed by its user-friendly interfaces, efficient event registration and booking processes, real-time notifications, and advanced search algorithms. These features, coupled with reliable technologies, ensure its successful implementation.

3.1.3 Behavioral Feasibility

Behavioral feasibility is a vital aspect of the event management website's success. It involves ensuring that users, including event organizers, attendees, and service providers, are ready and willing to embrace the new system. To achieve this, comprehensive user training and support, change management strategies, user feedback mechanisms, and a user-centric design approach will be employed. These measures aim to facilitate a smooth transition to the new platform, minimize resistance, and ensure a positive user experience. Continuous user feedback and iterative improvements will further enhance the system's alignment with user needs and expectations.

3.1.4 Feasibility Study Questionnaire

I. What types of events do you manage?

- Diverse range of events, including conferences, corporate meetings, weddings, trade shows, exhibitions, social gatherings, and more.

II. How do you ensure smooth event coordination for large gatherings?

- For large gatherings, focus on planning, efficient logistics, and gather a capable team.

III. How do you handle unexpected challenges during events?

- Rely on experience and quick decision-making skills. Have possibility plans in place to address issues on time while maintaining event quality.

IV. How do you manage event budgets and cost optimization?

- Carefully plan the budget with the client's goals in mind and negotiate with vendors for cost-effective solutions. Track expenses, prioritize spending, and find creative ways to optimize costs.

V. How do you handle vendor and service provider coordination?

- Establish strong relationships with reliable vendors and service providers. Clear communication, contracts, and regular updates ensure seamless coordination and smooth execution.

VI. How do you ensure client satisfaction?

- Client satisfaction is supreme. Maintain open communication, understand their vision, and deliver personalized solutions. Post-event feedback and evaluations help to continuously improve services.

VII. What are the challenges faced?

- The biggest challenges include manual processes, limited reach, communication gaps, data management issues, and resource optimization difficulties.

VIII. How to attract new customers?

- To attract new customers for an event management business, leverage online presence, referrals, networking, email marketing, and partnerships while providing excellent customer service.

IX. How do you handle last-minute changes or emergencies during events?

- During events, maintain open communication, implement likelihood plans, and rely on a strong network of service providers to handle last-minute changes and emergencies efficiently.

X. What are the key factors you consider when selecting event venues and service providers?

- When selecting event venues and service providers, prioritize factors such as location, capacity, facilities, reputation, cost-effectiveness, reliability, past client reviews, and their ability to meet specific event requirements and preferences.

3.1 SYSTEM SPECIFICATION

3.2.1 Hardware Specification

Processor – Intel core i3

RAM – 4 GB

Hard disk - 1 T B

3.2.2 Software Specification

Front End	-	HTML, CSS
Back End	-	Django
Database	-	SQLite
Client on PC	-	Windows 7 and above.
Technologies used	-	JS, HTML5, AJAX, Django, CSS

3.3 SOFTWARE DESCRIPTION

3.3.1 HTML

HTML (Hypertext Markup Language) is the most basic building block of the Web. HTML is a simple language made up of elements, which can be applied to pieces of text to give them different meanings in a document, structure a document into logical sections, and embed content such as images and videos into a page. In the proposed system, HTML 5 is used to carry out the design part of the webpage.HTML validations are also used.

3.3.2 CSS

While HTML is used to define the structure and semantics of your content, CSS is used to style it and lay it out. For example, you can use CSS to alter the font, colour, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features. In the proposed system CSS 3 is used. Animation using CSS is also used to give the website a unique look & feel.

3.3.3 JavaScript

JavaScript was initially created to “make web pages alive”. The programs in this language are called scripts. They can be written right in a web page’s HTML and run automatically as the page loads. In the proposed system loading screen, different popup animations are implemented using Java Script.

3.3.4 Django

Django is a powerful, open-source Python web framework used for creating dynamic, database-driven websites and web applications. Known for its versatility and rapid development capabilities, it simplifies complex, data-driven projects and follows the model-view-controller (MVC) architectural pattern. Django provides a comprehensive toolkit, including authentication, database handling, and a built-in admin panel. Its emphasis on clean, pragmatic code and the "Don't Repeat Yourself" (DRY) principle makes it a preferred choice for developers.

3.3.5 SQLite

SQLite is a serverless, self-contained, and lightweight relational database management system (RDBMS). It is embedded directly into applications, making it ideal for small to medium-sized projects, mobile apps, and testing. SQLite is known for its simplicity, speed, and portability, offering SQL-based data storage without requiring a separate database server.

3.3.6 AJAX

AJAX (Asynchronous JavaScript and XML) is a web technology that enables real-time data updates on web pages without full page reloads. It enhances user interactivity and reduces server load by allowing asynchronous data exchange.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

A crucial stage in creating an event management website is system design, when the requirements and ideas for the system are turned into an elaborate technological blueprint. This stage includes choices for architecture, database design, module requirements, and the system's general structure. Ensuring that the system is scalable, effective, and in line with the project's goals is the goal of the design process. It shapes the future of your event management platform and lays the groundwork for effective deployment and evolution. The main features of the system design that will result in a reliable, intuitive, and feature-rich website are described in this paper.

4.2 UML DIAGRAM

Unified Modeling Language (UML) diagrams are a visual representation of the structural and behavioral aspects of the system. UML diagrams offer a standardized and intuitive way to plan, document, and visualize the intricate aspects of an event management website's architecture and functionality.

UML includes the following nine diagrams:

- Class diagram
- Object diagram
- Use case diagram
- Sequence diagram
- Collaboration diagram
- Activity diagram
- Statechart diagram
- Deployment diagram
- Component diagram

4.2.1 USE CASE DIAGRAM

Use case diagrams describe the interactions between users (actors) and the system, focusing on what the system does from a user's perspective. For event management website, we can represent actors like "Organizer," "Attendee," and "Service Provider" and their interactions with the system. It is divided into three sections: Attendee, Event Organizer, and Service Provider. The Attendee section includes steps such as "Search and Book Services" and "Review and Ratings". The Event Organizer section includes steps such as "Organizer Registration" and "Book Services". The Service Provider section includes steps such as "Manage Booking" and "Update Availability". The boxes represent different steps in the process, while the arrows represent the flow of the process.

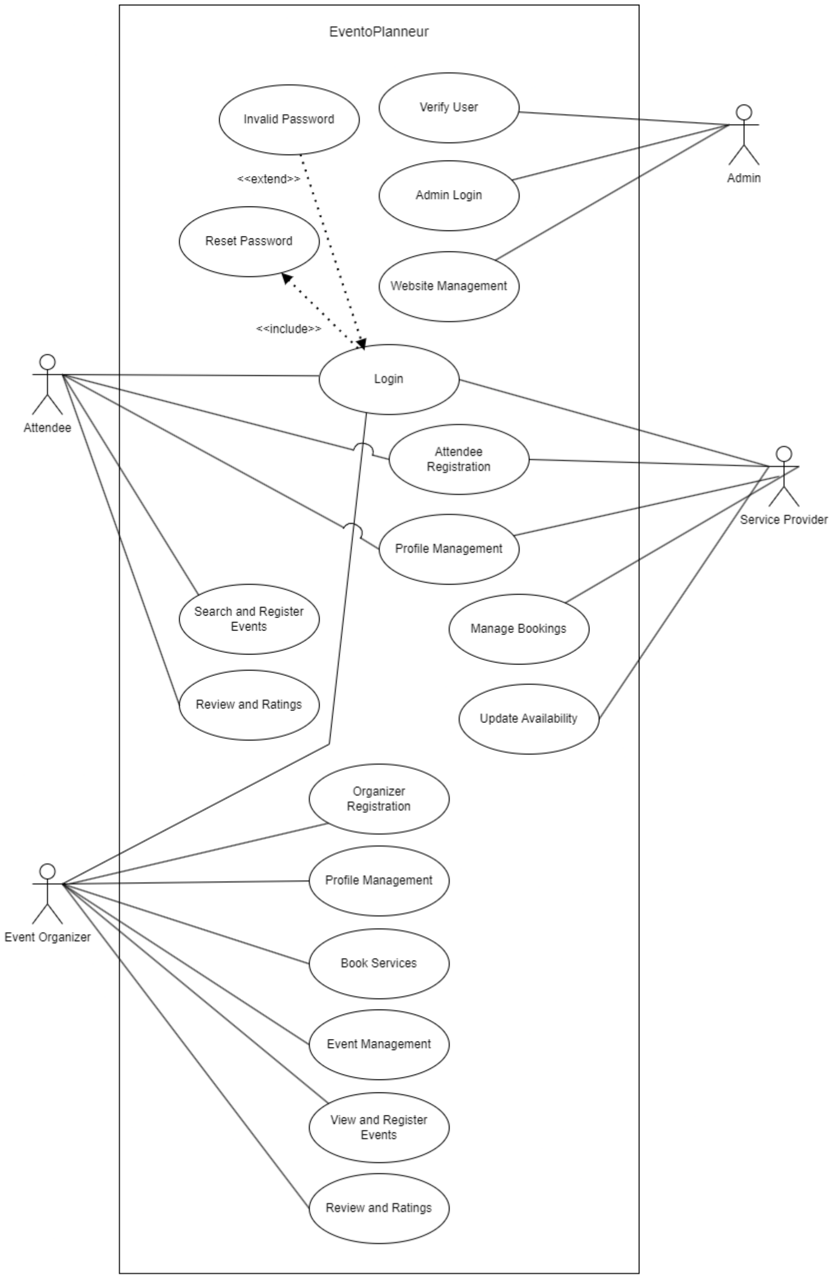


Figure 1: Use case diagram

4.2.2 SEQUENCE DIAGRAM

Sequence diagrams show how objects in the system interact and in what order. These are useful for visualizing the flow of events in the system, such as the process of event registration, user authentication, or payment processing. The process of registering, searching, and booking events in a system. It consists of rectangular boxes with text inside them, connected by lines with arrows. The boxes represent different steps in the process, and the lines represent the flow of information between them. The flowchart starts at the top with the “Event Organizer” box and ends at the bottom with the “Logout” box. It also includes a “Database” and a “Bank” box, which represent external systems that the process interacts with.

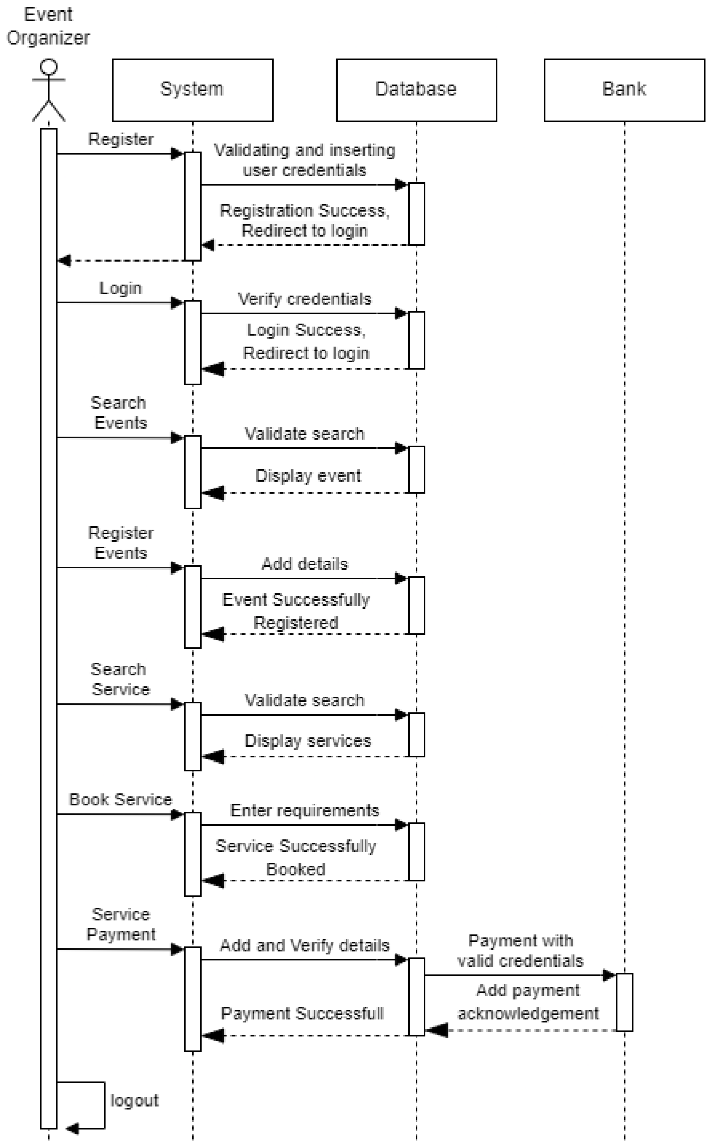


Figure 2: Sequence diagram

4.2.3 State Chart Diagram

State diagrams depict the different states an object can be in and the transitions between these states. You model the state changes of an event, user account, or booking. The diagram explains the process of registering, searching, and booking events in a system. It consists of rectangular boxes with text inside them, connected by lines with arrows. The boxes represent different steps in the process, and the lines represent the flow of information between them. The flowchart starts at the top with the “Event Organizer” box and ends at the bottom with the “Logout” box. The flowchart also includes a “Database” and a “Bank” box, which represent external systems that the process interacts with. A state chart is a type of diagram that represents a workflow or process. It can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. They are mainly used to document, study, plan, improve, and communicate often complex processes in clear, easy-to-understand diagrams.

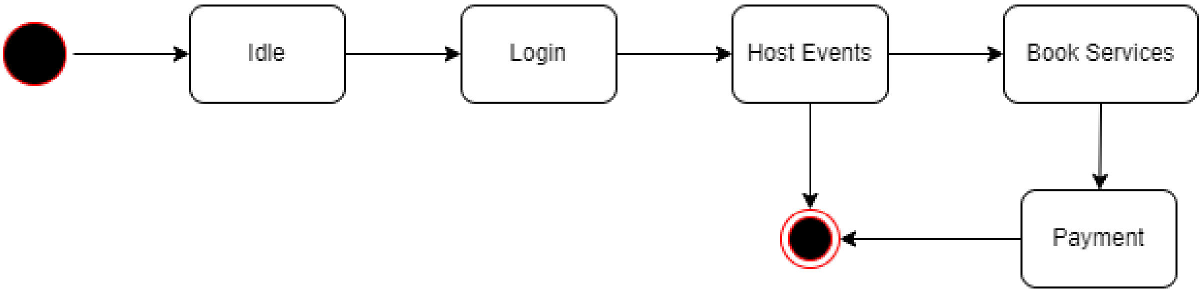


Figure 3: State Chart Diagram

4.2.4 Activity Diagram

Activity diagrams represent the workflow or business processes within the system. It depicts the flow of actions when an organizer creates an event, when attendees register for an event, or when a service provider offers their services. The diagram illustrates the flow of control within a system rather than the implementation. It models the concurrent and sequential activities and helps in envisioning the workflow from one activity to another. The flow can be sequential, branched, or concurrent, and to deal with such kinds of flows, the activity diagram has come up with a fork, join, etc. It is also termed as an object-oriented flowchart. It encompasses activities composed of a set of actions or operations that are applied to model the behavioral diagram. The diagram is made up of boxes and arrows. The boxes represent different actions that a user can take within the system, while the arrows represent the flow of information between them. The control flow of activity is represented by control nodes and object nodes that illustrate the objects used within an activity. The activities are initiated at the initial node and are terminated at the final node. It is a visual representation of the different steps and paths that a user can take within the system. In this case, the system is an event management system and the users are the admin, attendee, event organizer, and service provider. The actions include login, registering, manage website, searching and register for events, book services, manage booking, and payment

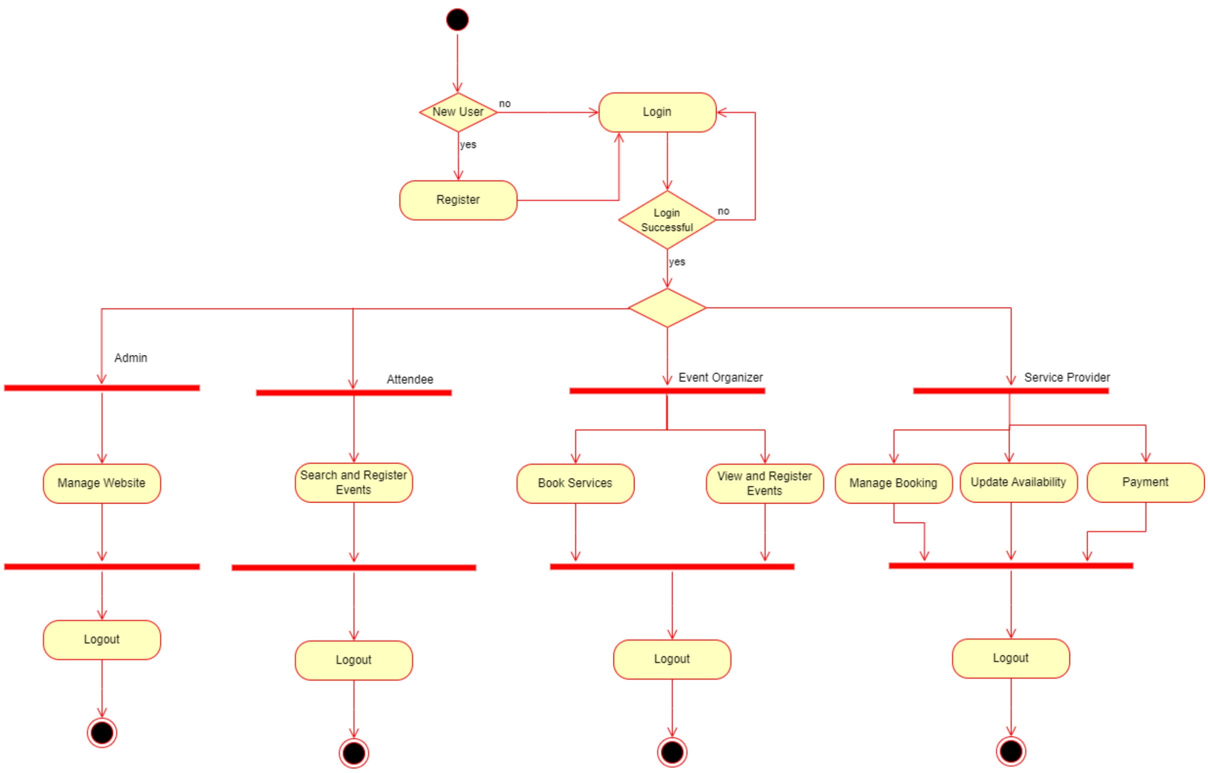


Figure 4: Activity Diagram

4.2.5 Component Diagram

A component diagram is a type of structural diagram in the Unified Modeling Language (UML) that shows the high-level components or modular parts of a system, along with their dependencies and interactions. In software engineering, it is used to represent the organization and relationships between the components or modules that make up a software system. Each component typically represents a self-contained and reusable part of the system, such as classes, packages, libraries, or even physical components. In the context of the event management site, a component diagram can help you visualize and understand the major software components or modules that form the system.

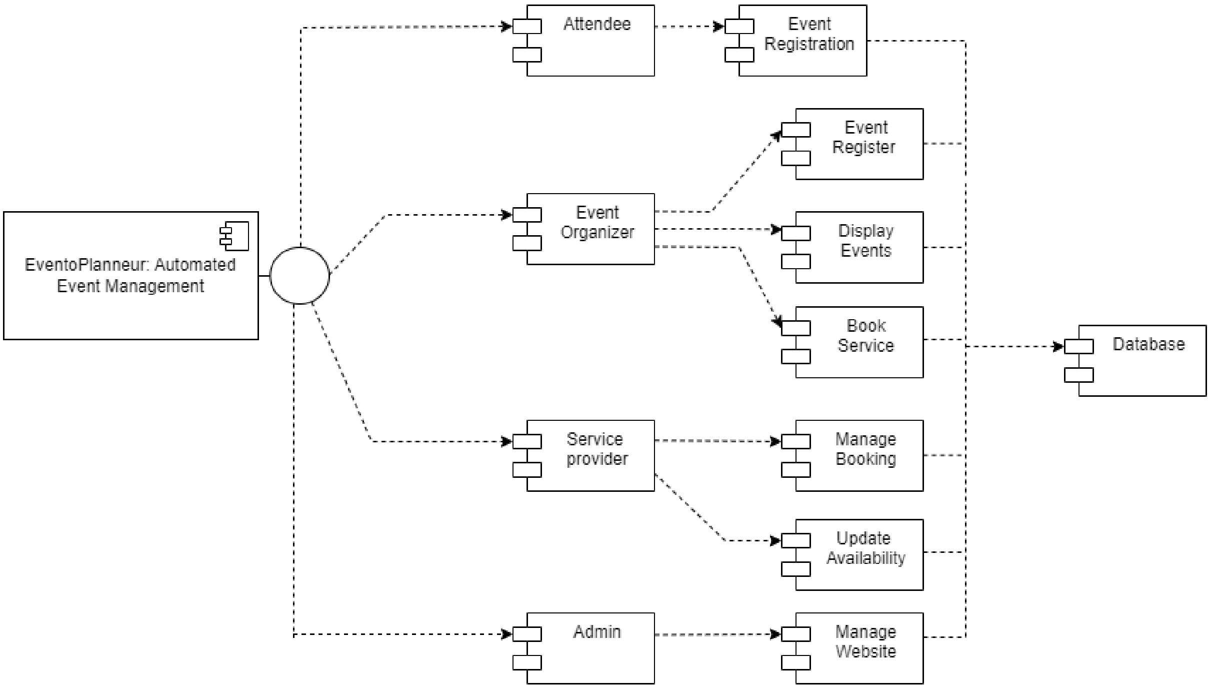


Figure 5: Component Diagram

4.2.6 Class Diagram

A class diagram is used to show the classes, relationships, interface, association, and collaboration. It describes various kinds of objects and the static relationship between them. The class diagram can be used to represent the object model for complex systems. It reduces the maintenance time by providing an overview of how an application is structured before coding. It is helpful for the stakeholders and developers. Here how to create a class diagram: Identify the objects in the problem domain and create classes for each of them. Add attributes for those classes. Add operations for those classes. Each class is represented by a rectangle having a subdivision of three compartments name, attributes, and operation. There are three types of modifiers that are used to decide the visibility of attributes and operations: + is used for public visibility (for everyone), - is used for private visibility (for only me), and # is used for protected visibility (for friend and derived)

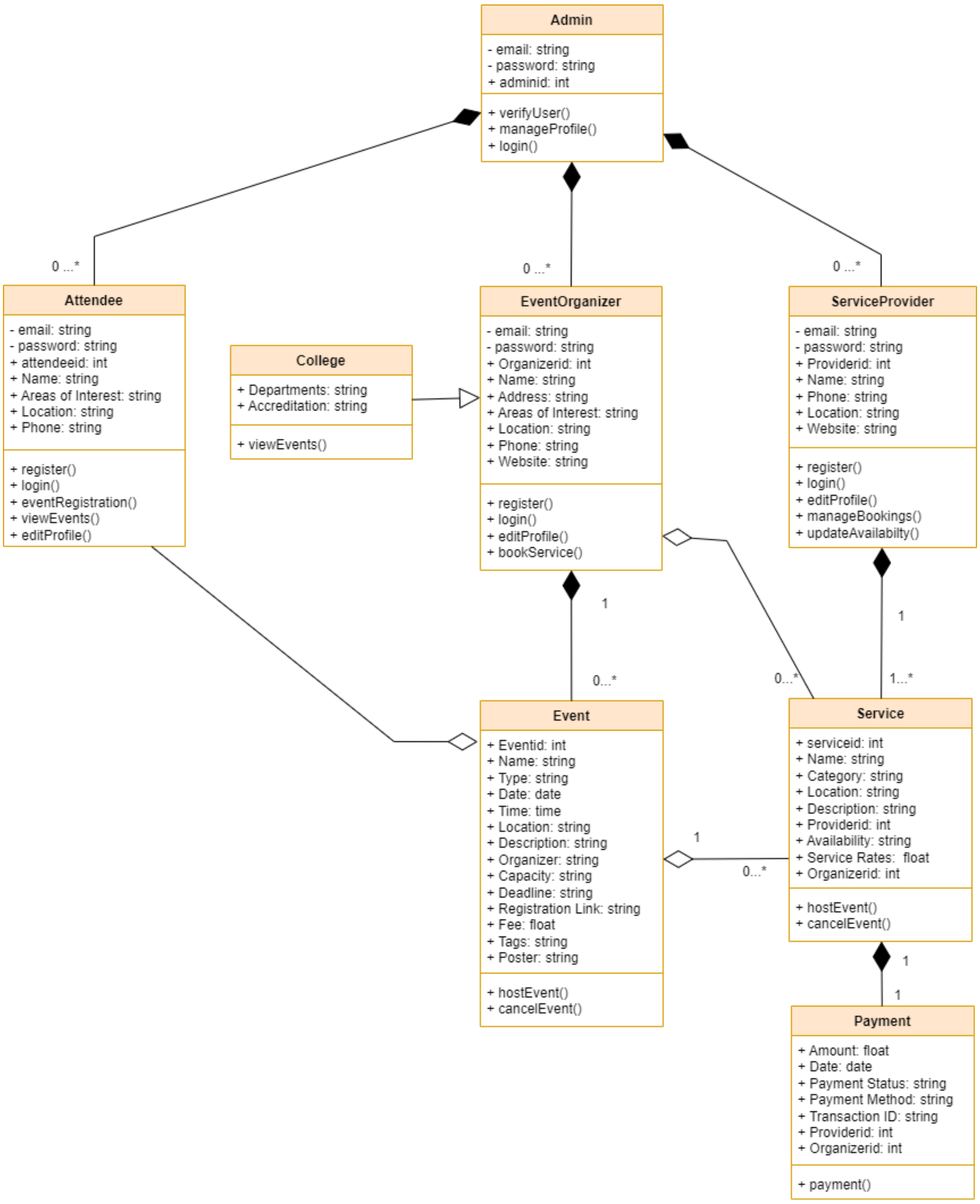


Figure 6: Class Diagram

4.2.7 Object Diagram

The diagram shows a snapshot of the instances in a system and the relationships between the instances. It is used to depict the nature and structure of the system at a particular point in time. The diagram consists of rectangular boxes and circular nodes. The rectangular boxes are labeled “Idle,” “Login”, “Host Events”, “Book Services”, and “Payment”. The flowchart shows the process of booking services and making payments. The flowchart starts at the “Idle” box and ends at the “Payment” box. The flowchart has arrows connecting the boxes and nodes, showing the direction of the process. Object diagrams use real-world examples to depict the nature and structure of the system at a particular point in time. They are useful to explain smaller portions of your system when your system class diagram is complex.

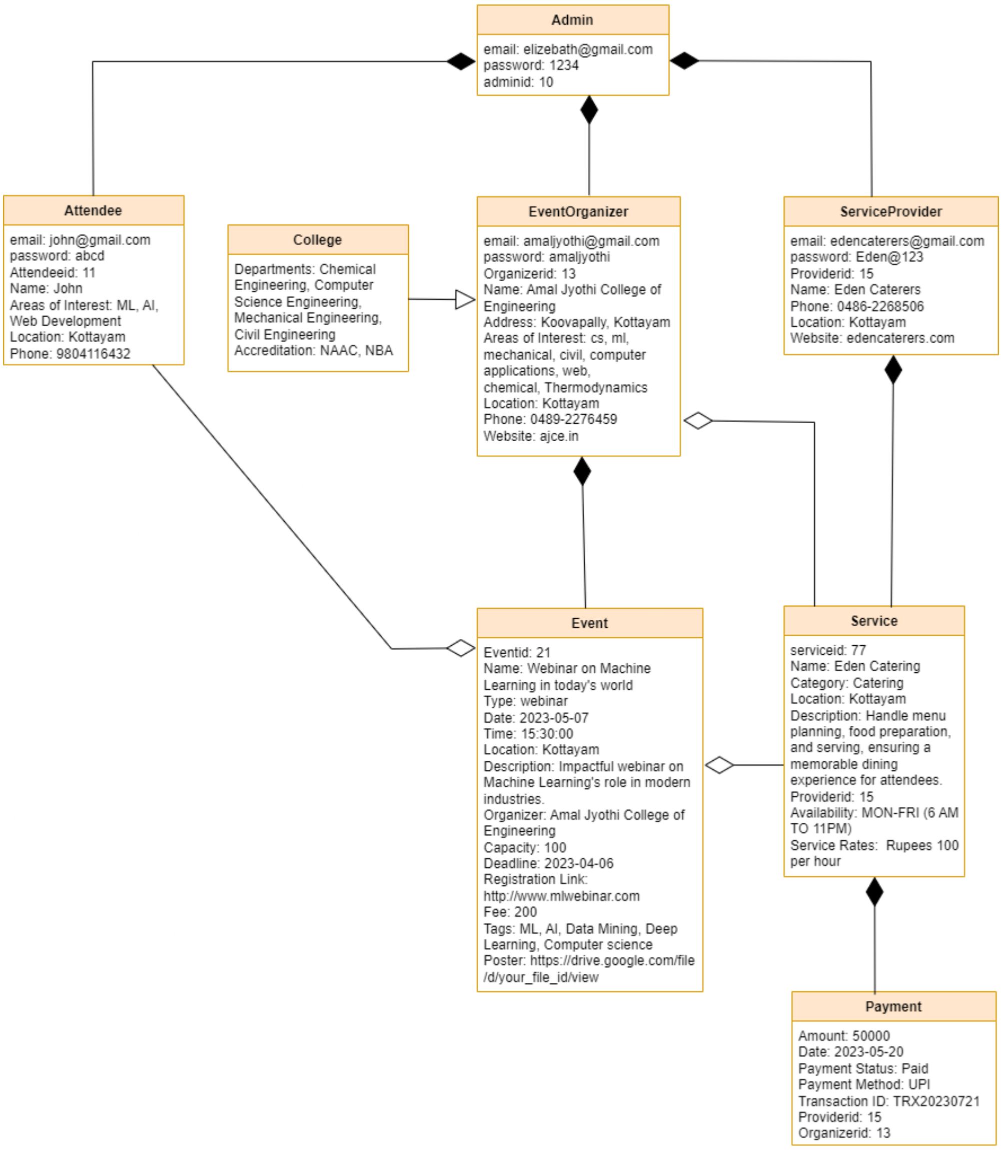

Admin
email: elizebath@gmail.com
password: 1234
adminid: 10
Attendee
email: john@gmail.com
password: abcd
Attendeeid: 11
Name: John
Areas of Interest: ML, AI, Web Development
Location: Kottayam
Phone: 9804116432
College
Departments: Chemical Engineering, Computer Science Engineering, Mechanical Engineering, Civil Engineering
Accreditation: NAAC, NBA
EventOrganizer
email: amaljyothi@gmail.com
password: amaljyothi
Organizerid: 13
Name: Amal Jyothi College of Engineering
Address: Koovapally, Kottayam
Areas of Interest: cs, ml, mechanical, civil, computer applications, web, chemical, Thermodynamics
Location: Kottayam
Phone: 0489-2276459
Website: ajce.in
ServiceProvider
email: edencaterers@gmail.com
password: Eden@123
Providerid: 15
Name: Eden Caterers
Phone: 0486-2268506
Location: Kottayam
Website: edencaterers.com
Event
Eventid: 21
Name: Webinar on Machine Learning in today's world
Type: webinar
Date: 2023-05-07
Time: 15:30:00
Location: Kottayam
Description: Impactful webinar on Machine Learning's role in modern industries.
Organizer: Amal Jyothi College of Engineering
Capacity: 100
Deadline: 2023-04-06
Registration Link: http://www.mlwebinar.com
Fee: 200
Tags: ML, AI, Data Mining, Deep Learning, Computer science
Poster: https://drive.google.com/file/d/your_file_id/view
Service
serviceid: 77
Name: Eden Catering
Category: Catering
Location: Kottayam
Description: Handle menu planning, food preparation, and serving, ensuring a memorable dining experience for attendees.
Providerid: 15
Availability: MON-FRI (6 AM TO 11PM)
Service Rates: Rupees 100 per hour
Payment
Amount: 50000
Date: 2023-05-20
Payment Status: Paid
Payment Method: UPI
Transaction ID: TRX20230721
Providerid: 15
Organizerid: 13

Figure 7: Object Diagram

4.2.8 Deployment Diagram

A deployment diagram can help to understand how our application is physically distributed, which can be useful for system administrators and DevOps teams responsible for deploying and maintaining the application. It can also be used to plan and document the hardware and software infrastructure of your event management system. Visualizing this information in a graphical diagram would provide a more intuitive and comprehensive view of the deployment.

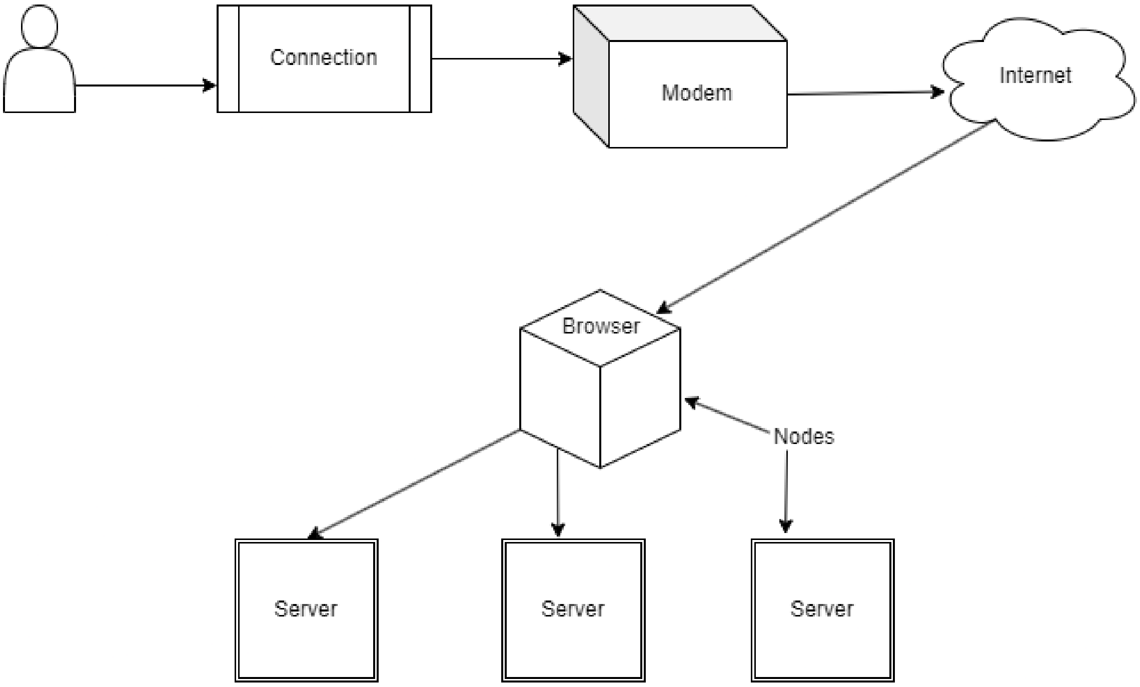
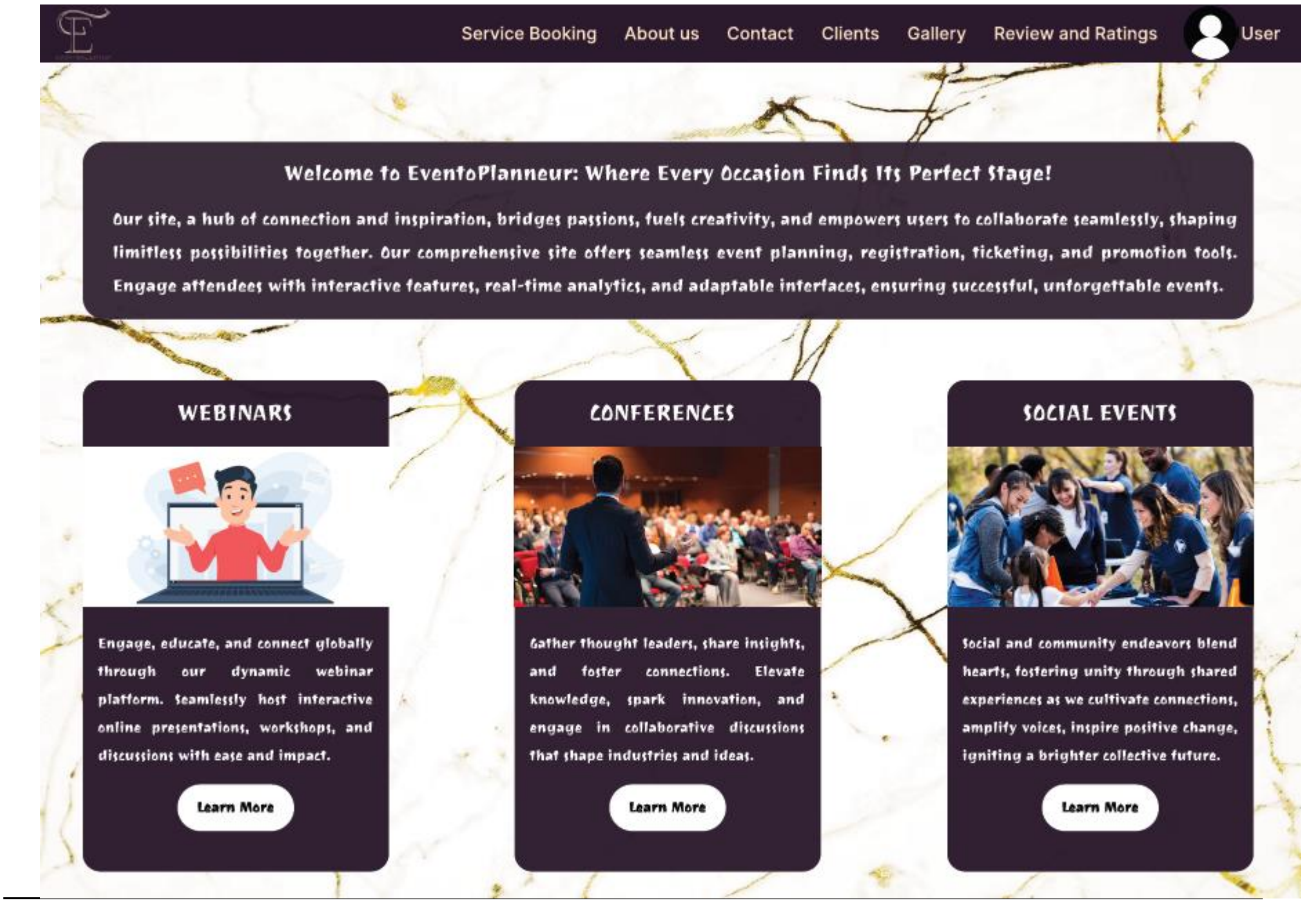


Figure 8: Deployment Diagram

4.3 USER INTERFACE DESIGN USING FIGMA

Form Name: Homepage



Welcome to EventoPlanneur: Where Every Occasion Finds Its Perfect Stage!

Our site, a hub of connection and inspiration, bridges passions, fuels creativity, and empowers users to collaborate seamlessly, shaping limitless possibilities together. Our comprehensive site offers seamless event planning, registration, ticketing, and promotion tools. Engage attendees with interactive features, real-time analytics, and adaptable interfaces, ensuring successful, unforgettable events.

WEBINARS

Engage, educate, and connect globally through our dynamic webinar platform. Seamlessly host interactive online presentations, workshops, and discussions with ease and impact.

Learn More

CONFERENCES

Gather thought leaders, share insights, and foster connections. Elevate knowledge, spark innovation, and engage in collaborative discussions that shape industries and ideas.

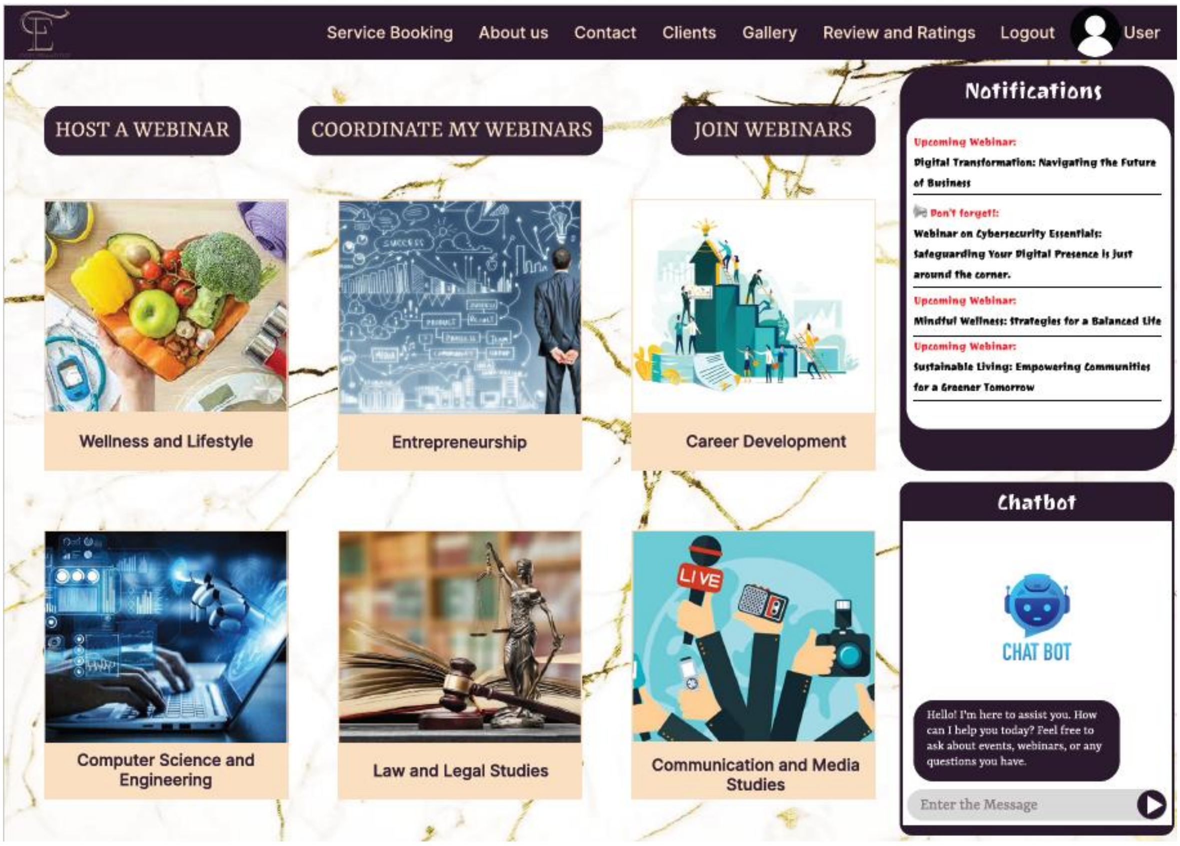
Learn More

SOCIAL EVENTS

Social and community endeavors blend hearts, fostering unity through shared experiences as we cultivate connections, amplify voices, inspire positive change, igniting a brighter collective future.

Learn More

Form Name: Webinar page



Upcoming Webinar:
Digital Transformation: Navigating the Future of Business

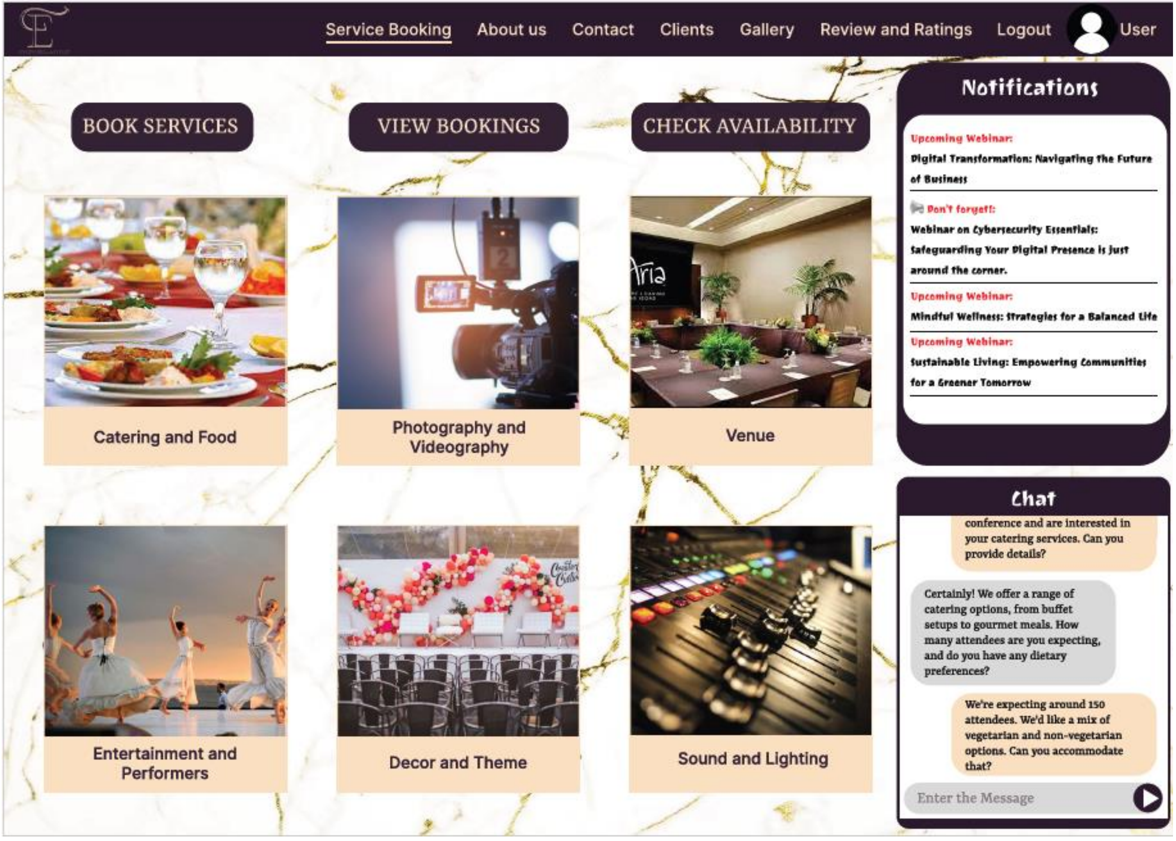
Don't forget!:
Webinar on Cybersecurity Essentials: Safeguarding Your Digital Presence is just around the corner.

Upcoming Webinar:
Mindful Wellness: Strategies for a Balanced Life

Upcoming Webinar:
Sustainable Living: Empowering Communities for a Greener Tomorrow

Hello! I'm here to assist you. How can I help you today? Feel free to ask about events, webinars, or any questions you have.

Form Name: Service Booking page



Upcoming Webinar:
Digital Transformation: Navigating the Future of Business

Don't forget!:
Webinar on Cybersecurity Essentials: Safeguarding Your Digital Presence is just around the corner.

Upcoming Webinar:
Mindful Wellness: Strategies for a Balanced Life

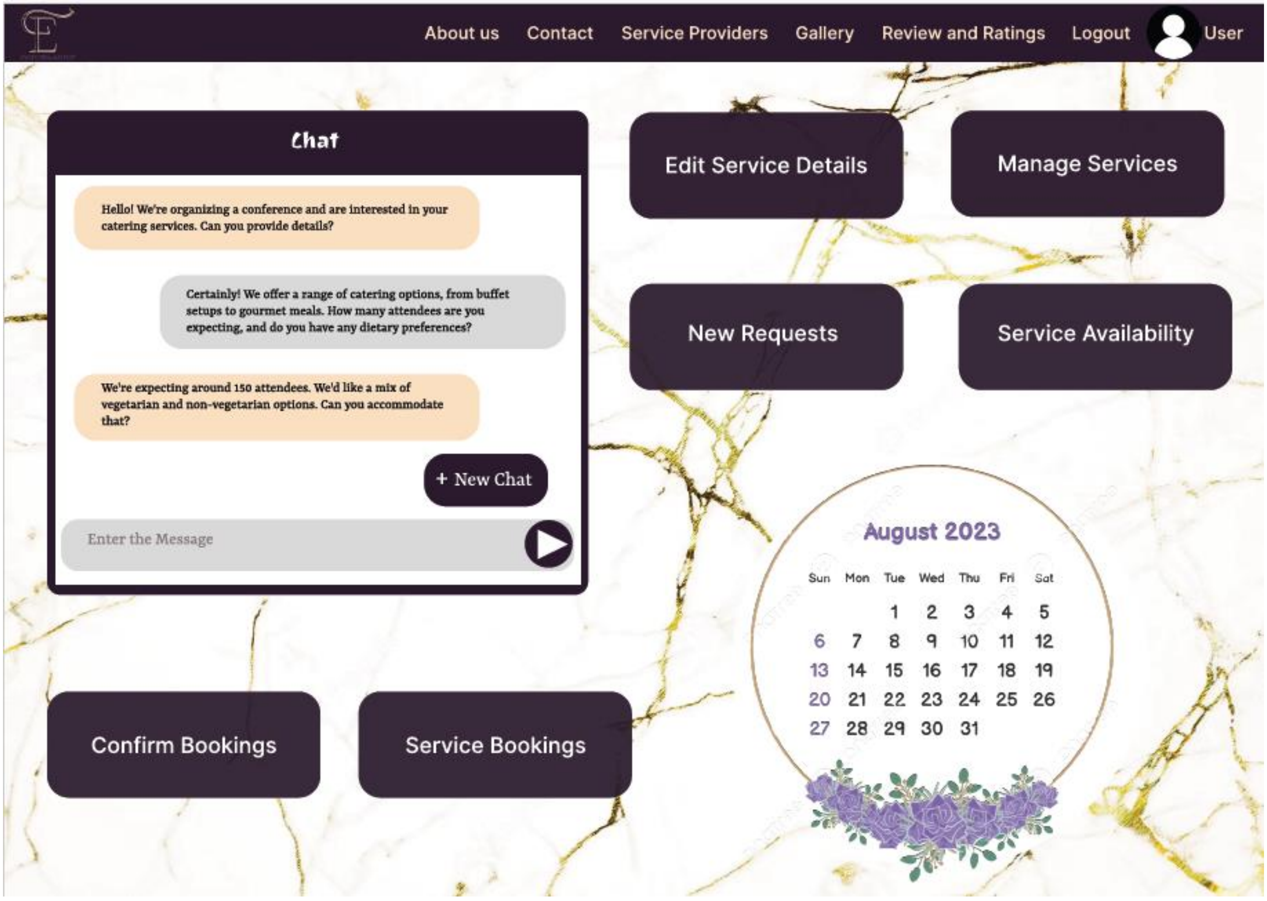
Upcoming Webinar:
Sustainable Living: Empowering Communities for a Greener Tomorrow

conference and are interested in your catering services. Can you provide details?

Certainly! We offer a range of catering options, from buffet setups to gourmet meals. How many attendees are you expecting, and do you have any dietary preferences?

We're expecting around 150 attendees. We'd like a mix of vegetarian and non-vegetarian options. Can you accommodate that?

Form Name: Service page



Hello! We're organizing a conference and are interested in your catering services. Can you provide details?

Certainly! We offer a range of catering options, from buffet setups to gourmet meals. How many attendees are you expecting, and do you have any dietary preferences?

We're expecting around 150 attendees. We'd like a mix of vegetarian and non-vegetarian options. Can you accommodate that?

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

4.4 DATABASE DESIGN

An information-storing device that is well-organized and capable of efficiently and effectively retrieving stored data is called a database. Any database's goal is its data, which is why it must be safeguarded.

There are two stages to the database design process. The first phase involves gathering user needs and designing a database that as clearly as feasible satisfies them. Information Level Design is the name of this stage, which is carried out without regard to any specific DBMS.

The information level design is then translated into a design for the database management system (DBMS) that will be utilized to put the system into operation in the second stage. The Physical Level Design stage addresses the features of the DBMS that will be utilized. The system design and the database design are executed concurrently.

4.4.1 Relational Database Management System (RDBMS)

A relational model uses a set of relations to describe the database. Every relation is like a file of records or a table of values. A table is referred to as a relation, a row as a tuple, and a column header as an attribute in formal relational model language. A relational database is made up of several tables, each given its own name. In a story, a row stands for a group of connected values.

Domains, Relations, and Attributes

A relation is a table. Tuples are the rows that make up a table. An ordered set of n elements is called a tuple. Attributes are the names given to columns. In the database, relationships have been established between each table. Referential and entity relationship integrity are therefore guaranteed. An atomic value set is called a domain D. One popular way to define a domain is to define the data type that the domain's data values are derived from. Giving the domain a name will also be helpful in deciphering its values. In a relation, each value is atomic and cannot be broken down.

Relationships

The key is used to build relationships between tables. Primary Key and Foreign Key are the two primary keys that are most important. These keys can be used to create relationships based on entity integrity and referential integrity. Entity Integrity mandates that null values cannot exist in any Primary Key. It is required by Referential Integrity that no Primary Key has null values. Referential Integrity: A corresponding Primary Key value must be present in the same domain for every unique Foreign Key value. Super Key and Candidate Keys are additional keys.

4.4.2 Normalization

To minimize the impact of future changes on data structures, data are put together in the simplest possible manner. The formal method of organizing data structures to remove duplication and enhance integrity is called normalization. By using the normalization approach, superfluous fields may be divided and a huge table can be divided into smaller ones. Additionally, it helps prevent anomalies related to insertion, deletion, and updating. Keys and relationships are the two notions used in normal form data modeling.

A row in a table is uniquely identified by a key. Primary keys and foreign keys are the two different categories of keys. An element, or a set of elements, in a table whose function is to identify entries from the same table is called a primary key. A column in one table that uniquely identifies a record from another table is called a foreign key. Up to the third normal form, every table has been normalized.

It means placing things in their natural form, as the name suggests. Through normalizing, the application developer hopes to obtain a logical arrangement of the data into appropriate tables and columns where names may be readily associated with the data by the user. By removing repetitive groupings from the data, normalization helps to prevent data duplication, which puts a significant strain on computer resources. These consist of: Data normalization, Table, and column proper names.

First Normal Form

According to the First Normal Form, each attribute's domain can only include atomic values, and each attribute value in a tuple can only contain one value from the attribute's domain. Stated differently, 1NF prohibits "relations as attribute values within tuples" or "relations within relations." 1NF only allows single atomic or indivisible values for attribute values. Putting the data into the First Normal Form is the first step. This may be accomplished by transferring data into several tables, each containing data of a similar type. Depending on what the project requires, a Primary Key or Foreign Key is assigned to each table. In doing so, we create new relations for every nested relationship or non-atomic property. This removed data sets that were repeated. If a relation solely meets the constraints that contain the main key, it is said to be in first normal form.

Second Normal Form

No non-key attribute should be functionally reliant on any portion of the main key for relations when the primary key has several attributes, according to the Second Normal Form. This involves breaking down each partial key and setting up a new connection with its dependent properties. Maintaining a relationship with the initial primary key and any properties that rely entirely on it for functionality is important. This stage aids in removing information that depends only partially on the key. If and only if all the first normal form requirements for the main key are met by the relation, and all of its non-primary key properties are entirely reliant on the primary key, then the relation is said to be in second normal form.

Third Normal Form

Third Normal Form states that a relation's non-key attribute cannot be functionally defined by a group of non-key attributes or by another non-key attribute. In other words, there should not be any reliance on the main key that is transitive. The non-key qualities that functionally determine other non-key attributes are broken down and put up in this way. The purpose of this phase is to eliminate anything that is not totally dependent on the Primary Key. If a relationship is solely in the second normal form and its non-key qualities do not depend on any other non-key attribute, it is said to be in the third normal form.

4.4.3 Sanitization

Web development requires data validation, particularly when working with forms where users submit their personal information first and then it is sent to the database. DBMS security issues may arise from data that is delivered in an incorrect format. SQL injections are frequently used by hackers to introduce rogue SQL instructions into databases. Once injected, SQL injections can potentially completely corrupt the database. Thus, before submitting user-entered data to the database, it must be sanitized and filtered to protect it from hackers. Purposefully erasing or destroying data from a storage device to assure that it cannot be retrieved is known as data sanitization. Typically, when data is removed from storage media, the medium itself is not truly wiped, and an attacker with access to the device can retrieve the data.

4.4.4 Indexing

To swiftly obtain specific data from the database, indexing is utilized. It is a method that maximizes a database query's searching time by utilizing data structures. Indexing creates an index table internally, hence reducing the number of disks needed to retrieve a certain piece of data. A copy of the database table's primary key or candidate key is contained in the search key. To shorten the total query or search time, we often keep the chosen Primary or Candidate keys in a sorted order. An index effectively returns a set of records that match a search key input. A database's indexing is specified by its indexing properties.

Various indexing techniques include:

Primary Indexing: The Primary Index is a fixed-length, ordered file containing two fields. The first field points to that data block, while the second field is the same as the main key. The relationships between the elements in the index table in the primary index are always one to one.

Secondary indexing: A field that is a candidate key and has a distinct value for every record in a database management system (DBMS) can produce the secondary index. Another name for it is a non-clustering index. The first level's mapping size is decreased by using this two-level database indexing method. This leads to the selection of a wide range of numbers for the first level; the mapping size is always modest.

Multilevel Index: Multilevel Indexing in a Database is created when a primary index does not fit in memory. In this type of indexing method, you can reduce the number of disk accesses to short any record kept on a disk as a sequential file and create a sparse base on that file.

4.5 TABLE DESIGN

1. tbl_login

Primary key: **login_id**

Foreign key: **login_id** references table **tbl_organizer, tbl_attendee, tbl_provider**

No:	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	login_id	int (10)	Primary Key	Login Id
2	email	varchar (30)	NULL	Email
3	password	varchar (20)	NOT NULL	Password
4	role	int (10)	NOT NULL	Role of actors
5	Status	Boolean	Not null	Set user active

2. tbl_attendee

Primary key: **attendee_id**

Foreign key: **login_id** references table **tbl_login**

No:	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	attendee_id	int (10)	Primary Key	Attendee Registration Id
2	name	varchar (30)	NOT NULL	Attendees name
3	contactno	varchar (20)	NOT NULL	Attendees contact number
4	login_id	int (10)	Foreign Key	Login id
5	email	varchar (30)	NOT NULL	Email
6	location	varchar (20)	NOT NULL	Location
7	organisation	varchar (20)	NOT NULL	Name of Organization/Institution
8	interest	varchar (100)	NOT NULL	Interests/Preferences
9	Status	Boolean	Not null	Set user active

3. tbl_organizer

- Primary key: organizer_id
- Foreign key: login_id references table tbl_login

No	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	organizer_id	int (10)	Primary Key	Organizer Registration ID
2	name	varchar (30)	NOT NULL	Organizer's name
3	contactno	varchar (20)	NOT NULL	Organizer's contact number
4	login_id	int (10)	Foreign Key	Login ID of the organizer
5	email	varchar (30)	NOT NULL	Email of the organizer
6	organization	varchar (50)	NOT NULL	Name of Organization/Company
7	location	varchar (30)	NOT NULL	Location of the organizer
8	Status	Boolean	Not null	Set user active

4. tbl_service_availability

- Primary key: availability_id
- Foreign key: service_id references table tbl_service

No	Fieldname	Datatype	Key Constraints	Description of the Field
1	availability_id	INTEGER (10)	Primary Key	Availability ID
2	service_id	INTEGER (10)	Foreign Key	Service ID of the service
3	day	VARCHAR (20)	NOT NULL	Day of the week
4	start_time	VARCHAR (10)	NOT NULL	Start time of availability
5	end_time	VARCHAR (10)	NOT NULL	End time of availability
6	Status	Boolean	Not null	Set user active

5. tbl_provider

- Primary key: provider_id
- Foreign key: login_id references table tbl_login

No	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	provider_id	int (10)	Primary Key	Provider Registration ID
2	name	varchar (30)	NOT NULL	Provider's name
3	contactno	varchar (20)	NOT NULL	Provider's contact number
4	login_id	int (10)	Foreign Key	Login ID of the provider
5	email	varchar (30)	NOT NULL	Email of the provider
6	organization	varchar (50)	NOT NULL	Organization/Company
7	location	varchar (20)	NOT NULL	Location of the provider
8	Status	Boolean	Not null	Set user active

6. tbl_payment

- Primary key: payment_id
- Foreign key: service_id, organizer_id references table tbl_service, tbl_organizer

No	Field Name	Datatype	Key Constraints	Description
1	payment_id	int (10)	Primary Key	Payment ID
2	service_id	int (10)	Foreign Key	ID of the service being paid for
3	organizer_id	int (10)	Foreign Key	ID of the organizer making the payment
4	payment_date	datetime	NOT NULL	Date and time of the payment
5	amount	decimal (10,2)	NOT NULL	Payment amount
6	payment_method	varchar (20)	NOT NULL	Payment method
7	Status	Boolean	Not null	Set user active

7.tbl_service

- Primary key: service_id
- Foreign key: provider _id references table tbl_ provider

No	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	service_id	int (10)	Primary Key	Service ID
2	provider_id	int (10)	Foreign Key	Provider ID
3	name	varchar (100)	NOT NULL	Service Name
4	description	text	NOT NULL	Service Description
5	price	decimal (10,2)	NOT NULL	Price of the Service
6	provider	varchar (30)	NOT NULL	Service Provider Name
7	location	varchar (30)	NOT NULL	Service Location
8	Status	Boolean	Not null	Set user active

8. tbl_event

- Primary key: event_id
- Foreign key: organizer_id references table tbl_organizer

No	Fieldname	Datatype (Size)	Key Constraints	Description of the Field
1	event_id	int (10)	Primary Key	Event ID
2	organizer_id	int (10)	Foreign Key	Organizer ID of the event
3	title	varchar (100)	NOT NULL	Event title
4	description	text	NOT NULL	Event description
5	start_date	datetime	NOT NULL	Start date and time of the event
6	end_date	datetime	NOT NULL	End date and time of the event
7	location	varchar (50)	NOT NULL	Location of the event
8	capacity	int (10)	NOT NULL	Maximum capacity of the event
9	event_category	varchar (50)	NOT NULL	Type of event
10	link	varchar (100)	NOT NULL	Registration Link
11	deadline	date	NOT NULL	Registration Deadline
12	organizer	varchar (50)	NOT NULL	Organizer's name
13	Status	Boolean	Not null	Set user active

9. tbl_departments

- Primary key: dept_id
- Foreign key: organizer _id references table tbl_ organizer

No	Field Name	Datatype	Key Constraints	Description
1	dept_id	int (10)	Primary Key	Department ID
2	organizer _id	int (10)	Foreign Key	ID of the corresponding college organizer
3	department_name	varchar (50)	NOT NULL	Name of the department
4	department_head	varchar (50)	NOT NULL	Name of the department head
5	department_email	varchar (50)	NOT NULL	Email of the department head
6	Status	Boolean	Not null	Set user active

10. **tbl_attendee_interest**

- Primary key: attendee_interest_id
- Foreign key: attendee_id references table tbl_attendee

No	Fieldname	Datatype	Key Constraints	Description of the Field
1	attendee_interest_id	INTEGER (10)	Primary Key	Attendee Interest ID
2	attendee_id	INTEGER (10)	Foreign Key	Attendee Registration ID
3	interest	VARCHAR (100)	NOT NULL	Interest/Preference of Attendee
4	Status	Boolean	Not null	Set user active

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

The process of running software under controlled conditions to find out if it behaves as intended is known as software testing. Software testing is frequently used interchangeably with validation and verification. The process of verifying or examining something comprises software, to ensure compliance and coherence with a related standard. Testing software is only one type of verification, which further makes use of methods like inspections, analyses, reviews, and tutorials. The process of confirming that the information provided matches what the user genuinely desires. Additional tasks that are frequently connected to software testing include static dynamic analysis as well as analysis. Without really running the code, static analysis examines the source code of software to find issues and collect metrics. Software behavior during execution is examined via dynamic analysis, which yields details like timing profiles, execution traces, and test coverage data.

Testing is a series of activities that can be methodically and pre-planned. Module-by-module testing is the first step in the process of testing a whole computer-based system. Nothing is finished without testing, and since system testing objectives are essential to the system's success, there are several rules that may be used as testing objectives. These are: The process of running software with the goal of identifying errors is called testing. A successful test is one that finds an undetected error; a good test case is one that has a high probability of detecting an error. If the testing is carried out in accordance with the above-mentioned aims, it will reveal software bugs. Additionally, testing shows that the performance criterion appears to have been satisfied and that the software function appears to be operating in accordance with the specification.

5.2 TEST PLAN

A test plan is a set of instructions intended to be followed to carry out different testing techniques. The activity that is to be taken is outlined in the Test Plan. A computer program, its documentation, and any associated data structures are made by software developers. Software developers are always in charge of testing the different program components to make sure each one serves the purpose for which it was intended. The purpose of the independent test group (ITG) is to eliminate the inherent issues that arise from allowing the builder to evaluate the completed product. Measurable language should be used to express the testing's precise objectives. Thus, the test strategy should include information on the mean time to failure, the cost of identifying and fixing flaws, the residual defect density or frequency of occurrence, and the number of test labor hours required for each regression test. The levels of testing include:

- Unit testing
- Integration Testing
- Data Validation Testing
- Output Testing

5.2.1 Unit Testing

Unit testing concentrates verification efforts on the software component or module, which is the smallest unit of software design. To find faults inside the module border, key control pathways are examined using the component-level design description as a reference. The close friend complexity of the tests revealed the unit testing scope. White-box-oriented unit testing can be carried out in parallel for numerous components. The divide interface is examined to make sure data enters and exits the test program unit correctly. To make sure that data that has been temporarily saved retains its integrity during every stage of an algorithm's execution, the local data structure is checked. To make sure that every statement in a module has been run at least once, boundary conditions are verified. All error-handling pathways are evaluated in the end.

Before starting any further tests, data flow tests via a module interface must be completed. All other tests are meaningless if data do not enter and depart correctly. One of the most important tasks in the unit test is to selectively test the execution pathways. Error circumstances must be expected in good design, and error handling pathways must be configured to redirect or cleanly end execution when an error does occur. The final phase in the unit testing process is boundary testing. Software frequently breaks down at its limits.

.

5.2.2 Integration Testing

Integration testing is an organized approach to building the program's structure while simultaneously running tests to find interface-related problems. The goal is to take components that have undergone unit testing and use them to construct a program structure that is determined by design. The program is tested in its entirety. Correction is challenging since the large scope of the overall program makes it difficult to isolate the reasons. After these mistakes are fixed, new ones crop up, and the process carries on in what seems like an unending circle. All the modules were merged to test for interface consistency issues after unit testing was completed in the system.

5.2.3 Validation Testing or System Testing

This completes the testing process. This involved testing the system in its entirety, including all forms, codes, modules, and class modules. This type of testing is sometimes referred to as system testing or black box testing. The functional requirements of the program are the main emphasis of the black box testing approach. To put it another way, Black Box testing gives software engineers the ability to determine input condition sets that will completely test all a program's functional requirements. issues in data structures or external data access, erroneous or missing functions, interface issues, performance flaws, initialization mistakes, and termination faults are the types of defects that black box testing looks for.

5.2.4 Output Testing or User Acceptance Testing

When a system is put to the test for user approval, it should meet the needs of the company. When the program is being developed, it should stay in communication with the user and the perspective system, adjusting as needed. Considering the following, this is done:

- Designs for Input Screens
- Designs for Output Screens

The testing mentioned above uses a variety of test data types. In order to test the system, test data preparation is essential. The system under investigation is tested using the test data that has been prepared. Errors in the test data are found during system testing and fixed using the previously mentioned testing procedures; the adjustments are also recorded for further use.

5.2.5 Automation Testing

Software and other digital goods are tested using automation to make sure they adhere to stringent specifications. Basically, a test to confirm that the hardware or software performs precisely as intended. It checks for errors, flaws, and any other problems that could occur during the creation of a product. Using automated testing in software development has the following advantages: Comprehensive reporting features, better bug detection, testing simplification, testing process acceleration, and a decrease in human intervention.

5.2.6 Selenium Testing

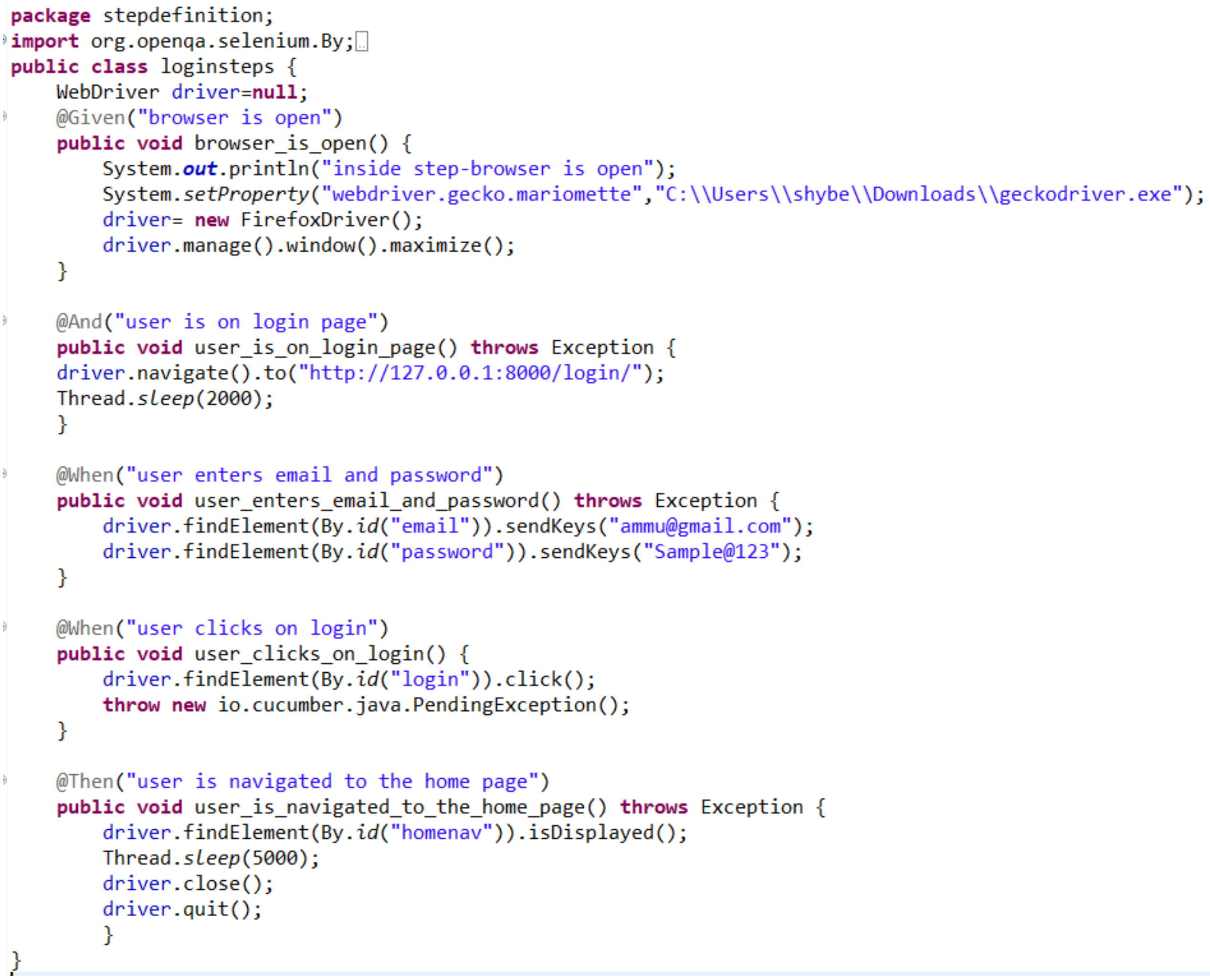
One of the most popular open-source Web UI (User Interface) automation testing suites is called Selenium. Jason Huggins created it in the beginning as a Thought Works internal tool in 2004. Selenium facilitates automation for many platforms, programming languages, and browsers. Selenium is readily deployable on Windows, Linux, Solaris, and Macintosh operating systems. Additionally, it supports Android, Windows Mobile, and iOS as OS (Operating System) for mobile apps.

Selenium uses language-specific drivers to support a wide range of programming languages. Selenium supports the following languages: Java, C#, Perl, PHP, Python, and Ruby. The Selenium Web driver is now most often used with C# and Java. Any of the supported programming languages may be used to write Selenium test scripts, which can then be executed directly in most contemporary web browsers. Selenium supports the following browsers: Google Chrome, Safari, Mozilla Firefox, and Internet Explorer.

Functional tests can be automated with Selenium, and continuous testing may be accomplished by integrating it with other automation test technologies like Maven, Jenkins, and Docker. Additionally, it may be used with JUnit and TestNG to manage test cases and provide results.

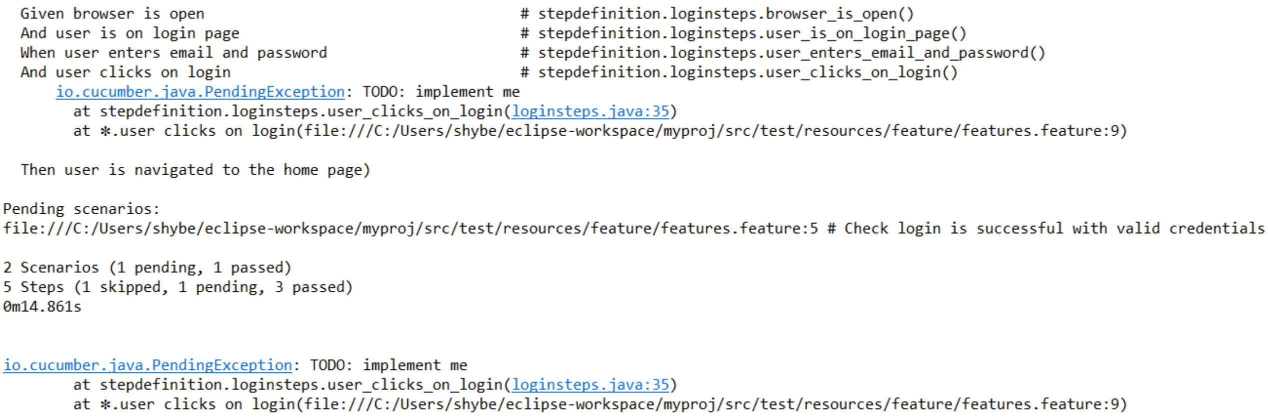
Example: Login Page Test Case

Code



```
package stepdefinition;
import org.openqa.selenium.By;
public class loginsteps {
    WebDriver driver=null;
    @Given("browser is open")
    public void browser_is_open() {
        System.out.println("inside step-browser is open");
        System.setProperty("webdriver.gecko.mariomette","C:\\Users\\shybe\\Downloads\\geckodriver.exe");
        driver= new FirefoxDriver();
        driver.manage().window().maximize();
    }

    @And("user is on login page")
    public void user_is_on_login_page() throws Exception {
    driver.navigate().to("http://127.0.0.1:8000/login/");
    Thread.sleep(2000);
    }

    @When("user enters email and password")
    public void user_enters_email_and_password() throws Exception {
        driver.findElement(By.id("email")).sendKeys("ammu@gmail.com");
        driver.findElement(By.id("password")).sendKeys("Sample@123");
    }

    @When("user clicks on login")
    public void user_clicks_on_login() {
        driver.findElement(By.id("login")).click();
        throw new io.cucumber.java.PendingException();
    }

    @Then("user is navigated to the home page")
    public void user_is_navigated_to_the_home_page() throws Exception {
        driver.findElement(By.id("homenav")).isDisplayed();
        Thread.sleep(5000);
        driver.close();
        driver.quit();
        }
}
```

Eg.Screenshot



```
  Given browser is open                                    # stepdefinition.loginsteps.browser_is_open()
  And user is on login page                                # stepdefinition.loginsteps.user_is_on_login_page()
  When user enters email and password                      # stepdefinition.loginsteps.user_enters_email_and_password()
  And user clicks on login                                 # stepdefinition.loginsteps.user_clicks_on_login()
      io.cucumber.java.PendingException: TODO: implement me
        at stepdefinition.loginsteps.user_clicks_on_login(loginsteps.java:35)
        at ✽.user clicks on login(file:///C:/Users/shybe/eclipse-workspace/myproj/src/test/resources/feature/features.feature:9)

  Then user is navigated to the home page)

Pending scenarios:
file:///C:/Users/shybe/eclipse-workspace/myproj/src/test/resources/feature/features.feature:5 # Check login is successful with valid credentials

2 Scenarios (1 pending, 1 passed)
5 Steps (1 skipped, 1 pending, 3 passed)
0m14.861s


io.cucumber.java.PendingException: TODO: implement me
        at stepdefinition.loginsteps.user_clicks_on_login(loginsteps.java:35)
        at ✽.user clicks on login(file:///C:/Users/shybe/eclipse-workspace/myproj/src/test/resources/feature/features.feature:9)
```

Eg.Test Report

Test Case 1

Project Name: EventoPlanneur: Event Management System	
Login Test Case	
Test Case ID: Test_1	**Test Designed By: Elizebath Thomas**
Test Priority (Low/Medium/High): High	**Test Designed Date: 12-10-2023**
Module Name: Login Screen	**Test Executed By: Dr. Paulin Paul**
Test Title: Verify login page with email and password	**Test Execution Date: 13-10-2023**
Description: Test the Login Page	
Pre-Condition: User has valid email and password	

Step	Test Step	Test Data	Expected Result	Actual Result	Status (Pass/ Fail)
1	Navigate to Login Page		Login Page should be displayed	Login Page displayed	Pass
2	Provide valid e-mail ID	E-mail ID: elizatom9@gmail.com	User should be able to login	User logged in and navigated to user dashboard	Pass
3	Provide valid password	Password: Eliza@123			
4	Click sign In Button				
5	Provide Invalid Email –ID or password	E-mail ID: elizebath@gmail.com Password: E@123	User should not be able to Login	Message "Enter valid email - ID or password displayed"	Pass
6	Provide Null Phone number or Password	E-mail ID: null Password: null			
7	Click on Sign In button				

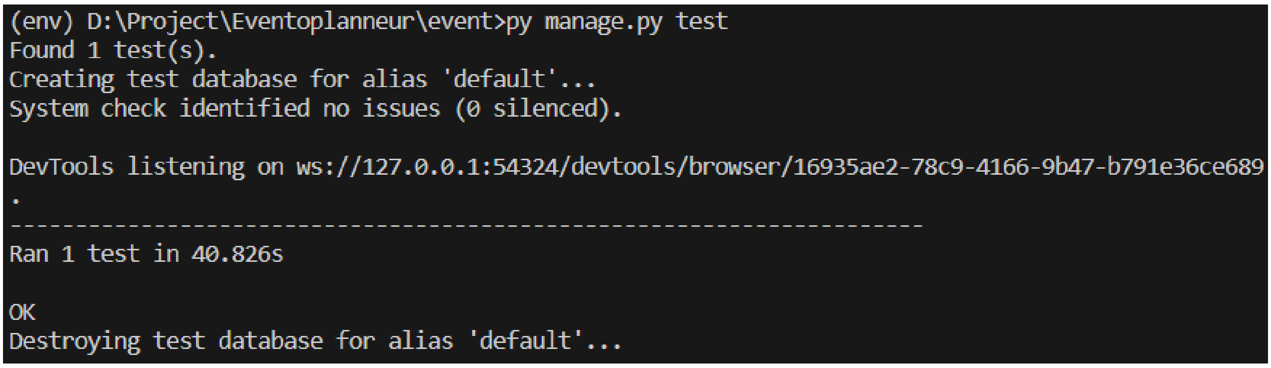
Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database

Example: Event Registration Test Case

Code

```
from datetime import datetime
from django.test import TestCase
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
class Hosttest(TestCase):
    def setUp(self):
        self.driver = webdriver.Chrome()
        self.driver.implicitly_wait(10)
        self.live_server_url = 'http://127.0.0.1:8000/'
    def tearDown(self):
        self.driver.quit()
    def test_01_login_page(self):
        driver = self.driver
        driver.get(self.live_server_url)
        driver.maximize_window()
        time.sleep(1)
        login = driver.find_element(By.CSS_SELECTOR, 'a.nav-link[href="/login/"]')
        login.click()
        time.sleep(2)
        email = driver.find_element(By.CSS_SELECTOR, 'input[name="email"]')
        email.send_keys("waremo9428@twugg.com")
        password = driver.find_element(By.CSS_SELECTOR, 'input[name="password"]')
        password.send_keys("Sample@123")
        time.sleep(2)
        submit = driver.find_element(By.CSS_SELECTOR, 'button#login.btn.btn-primary.btn-block.register-btn')
        submit.click()
        time.sleep(2)
        driver.get("http://127.0.0.1:8000/register_webinar/")
        time.sleep(3)
        title = driver.find_element(By.NAME, "title")
        title.send_keys("Webinar on Global Warming")
        fee = driver.find_element(By.NAME, "fee")
        fee.send_keys("100")  # You can change this value

        start_time = driver.find_element(By.NAME, "start_time")
        start_time.send_keys("09:00 AM")  # Adjust the time as needed
        end_time = driver.find_element(By.NAME, "end_time")
        end_time.send_keys("11:00 AM")
        date = driver.find_element(By.NAME, "date")
        date.send_keys("26-11-2023")
        deadline = driver.find_element(By.NAME, "deadline")
        deadline.send_keys("23-11-2023")
        event_type = driver.find_element(By.NAME, "event_type")
        event_type.send_keys("Online
        organizer_name = driver.find_element(By.NAME, "organizer_name")
        organizer_name.send_keys("Dept. of Environment Studies")

        phone_number = driver.find_element(By.NAME, "phone_number")
        phone_number.send_keys("8978537890")
        poster = driver.find_element(By.NAME, "poster")
        poster.send_keys("https://example.com/poster.jpg")  # Use the desired image URL
        max_participants = driver.find_element(By.NAME, "max_participants")
        max_participants.send_keys("100")  # Set the maximum number of participants
        # Find the "Speakers" section and its elements
        speakers_container = driver.find_element(By.ID, "speakers-container")
        speaker_designation = driver.find_element(By.NAME, "speakers_designation[]")
        speaker_name = driver.find_element(By.NAME, "speakers_name[]")
        speaker_designation.send_keys("Mr")  # You can change the designation
        speaker_name.send_keys("John Kim")  # Enter the speaker's name
        description = driver.find_element(By.NAME, "description")
        description.send_keys("Are you concerned about the pressing issue of global warming and its far-reaching consequences? Join us for an insightful webinar where we dive deep into the science, impact, and sustainable solutions related to this critical environmental challenge.")
        time.sleep(3)
        submit_button = driver.find_element(By.XPATH, "//button[@type='submit']")
        submit_button.click()
        time.sleep(3)
        logout = driver.find_element(By.CSS_SELECTOR, "a.nav-link[href='/logout/']")
        logout.click()
        time.sleep(2)

if __name__ == '__main__':
    import unittest
    unittest.main()
```

Eg.Screenshot



```
(env) D:\Project\Eventoplanneur\event>py manage.py test
Found 1 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).

DevTools listening on ws://127.0.0.1:54324/devtools/browser/16935ae2-78c9-4166-9b47-b791e36ce689
.
----------------------------------------------------------------------
Ran 1 test in 40.826s

OK
Destroying test database for alias 'default'...
```

Test Case 2

Project Name: EventoPlanneur: Event Management System	
Event Registration Test Case	
Test Case ID: Test_2	**Test Designed By: Elizebath Thomas**
Test Priority (Low/Medium/High): High	**Test Designed Date: 13-10-2023**
Module Name: Event Registration Screen	**Test Executed By: Dr. Paulin Paul**
Test Title: Event Registration Testing	**Test Execution Date: 14-10-2023**
Description: Login to event organizer user and register a webinar	
Pre-Condition: Users has registered as event organizer user	

Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/ Fail)**
1	Navigate to Login Page		Login Page should be displayed	Login Page displayed	Pass
2	Provide valid e-mail ID	E-mail ID: elizatom9@gmail.com	User should be able to login	User logged in and navigated to user dashboard	Pass
3	Provide valid password	Password: Eliza@123			
4	Click sign In Button				
5	Click on Host a Webinar		Webinar registration must be displayed	Webinar registration page is displayed	Pass
6	Provide valid title	Tile: Webinar on Global Warming	User should be able to enter valid event details	Event is added to the database	Pass
7	Provide valid fee	Fee: 100			
8	Provide start time	Start time: 10:00:00			
9	Provide date	Date: 11-11-2023			
10	Provide event type	Event type: Online			
11	Provide deadline	Deadline: 05-11-2023			

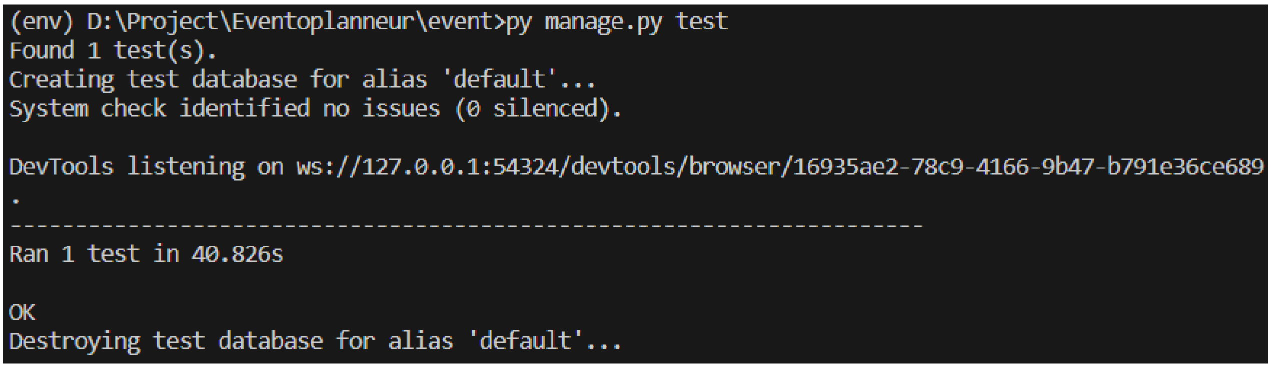
12	Provide organizer name	Organizer: Dept. of Environment Studies			
13	Provide end time	End time: 16:00:00			
14	Provide poster URL	Poster: https://example.com/poster.jpg			
15	Provide description	Description: Are you concerned about the pressing issue of global warming and its far-reaching consequences? Join us for an insightful webinar where we dive deep into the science, impact, and sustainable solutions related to this critical environmental challenge.			
16	Click Register Button				
Post-Condition: The webinar is successfully registered n database and visible to other users.					

Example: Event Organizer Profile Test Case

Code

```
from datetime import datetime
from django.test import TestCase
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
class Hosttest(TestCase):

    def setUp(self):
        self.driver = webdriver.Chrome()
        self.driver.implicitly_wait(10)
        self.live_server_url = 'http://127.0.0.1:8000/'
    def tearDown(self):
        self.driver.quit()

    def test_01_login_page(self):
        driver = self.driver
        driver.get(self.live_server_url)
        driver.maximize_window()
        time.sleep(1)
        login = driver.find_element(By.CSS_SELECTOR, 'a.nav-link[href="/login/"]')
        login.click()
        time.sleep(2)
        email = driver.find_element(By.CSS_SELECTOR, 'input[name="email"]')
        email.send_keys("elizebaththomas2024@mca.ajce.in")
        password = driver.find_element(By.CSS_SELECTOR, 'input[name="password"]')
        password.send_keys("Eliza@123")
        time.sleep(2)
        submit = driver.find_element(By.CSS_SELECTOR, 'button#login.btn.btn-primary.btn-block.register-btn')
        submit.click()
        time.sleep(2)
        driver.get("http://127.0.0.1:8000/org_profile/")
        aicte = driver.find_element(By.NAME, "aicte_id")
        aicte.clear()
        aicte.send_keys("1-36488387417")
        phone_number = driver.find_element(By.NAME, "phone_number")
        phone_number.clear()
        phone_number.send_keys("9034567890")
        website = driver.find_element(By.NAME, "website")
        website.clear()
        website.send_keys("https://www.model.in")
        submit_button = driver.find_element(By.CSS_SELECTOR, 'input[type="submit"]')
        submit_button.click()
        time.sleep(5)
if __name__ == '__main__':
    import unittest
    unittest.main()
```

Eg.Screenshot



```
(env) D:\Project\Eventoplanneur\event>py manage.py test
Found 1 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).

DevTools listening on ws://127.0.0.1:54324/devtools/browser/16935ae2-78c9-4166-9b47-b791e36ce689
.
----------------------------------------------------------------------
Ran 1 test in 40.826s

OK
Destroying test database for alias 'default'...
```

Test Case 3

Project Name: EventoPlanneur: Event Management System	
Event Organizer Profile Test Case	
Test Case ID: Test_3	**Test Designed By: Elizebath Thomas**
Test Priority (Low/Medium/High): High	**Test Designed Date: 13-10-2023**
Module Name: Event Organizer Profile Screen	**Test Executed By: Dr. Paulin Paul**
Test Title: Event Organizer Profile page Testing	**Test Execution Date: 14-10-2023**
Description: Login to event organizer user and edit their profile	
Pre-Condition: Users has registered as organizer	

Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/ Fail)**
1	Navigate to Login Page		Login Page should be displayed	Login Page displayed	Pass
2	Provide valid e-mail ID	E-mail ID: elizebaththomas2024@mca.ajce.in	User should be able to login	User logged in and navigated to user dashboard	Pass
3	Provide valid password	Password: Eliza@123			
4	Click sign In Button				
5	Click on Profile tab		Profile page must be displayed	Profile page is displayed	Pass
6	Provide valid AICTE id	AICTE id: 1-36488387417	Checks if AICTE id entered is valid and auto-fills college name, location, and address	Auto-fill college name, location, and address	Pass
7	Provide valid phone number	Phone number: 7678965100	User should be able to enter valid college details	College details is added to the database	Pass
8	Provide website	Website: https://www.adishankara.in			
10	Click Submit Button				

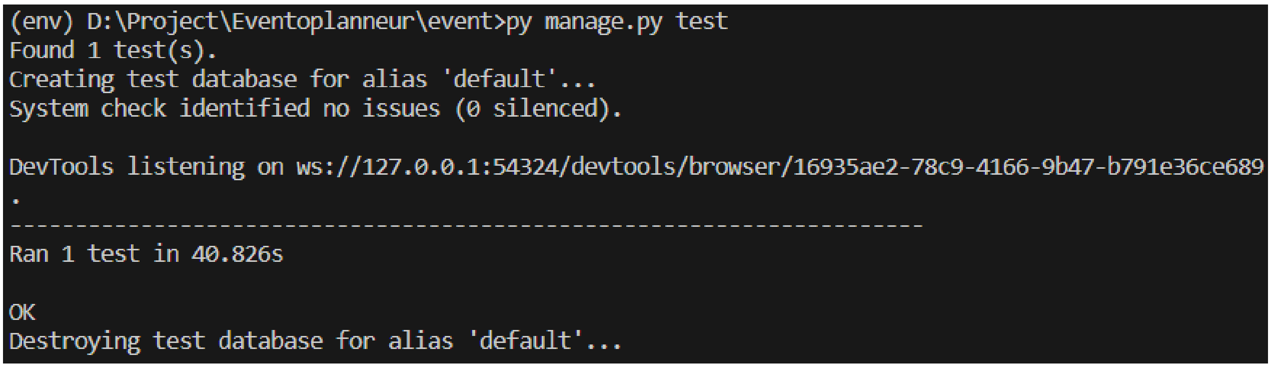
Post-Condition: The profile is successfully updated in database

Example: Attendee Event Recommendation Test Case

Code

```
from datetime import datetime
from django.test import TestCase
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
class Hosttest(TestCase):
    def setUp(self):
        self.driver = webdriver.Chrome()
        self.driver.implicitly_wait(10)
        self.live_server_url = 'http://127.0.0.1:8000/'
    def tearDown(self):
        self.driver.quit()
    def test_01_login_page(self):
        driver = self.driver
        driver.get(self.live_server_url)
        driver.maximize_window()
        time.sleep(1)
        login = driver.find_element(By.CSS_SELECTOR, 'a.nav-link[href="/login/"]')
        login.click()
        time.sleep(2)
        email = driver.find_element(By.CSS_SELECTOR, 'input[name="email"]')
        email.send_keys("ammu@gmail.com")
        password = driver.find_element(By.CSS_SELECTOR, 'input[name="password"]')
        password.send_keys("Sample@123")
        time.sleep(2)
        submit = driver.find_element(By.CSS_SELECTOR, 'button#login.btn.btn-primary.btn-block.register-btn')
        submit.click()
        time.sleep(2)
        events = driver.find_element(By.CSS_SELECTOR, 'a.nav-link[href="/recommendations/"]')
        events.click()
        time.sleep(2)
        view = driver.find_element(By.CSS_SELECTOR, "a.btn[href='/view_webinar/68']")
        view.click()
        time.sleep(2)
        logout = driver.find_element(By.CSS_SELECTOR, "a.nav-link[href='/logout/']")
        logout.click()
        time.sleep(2)
if __name__ == '__main__':
    import unittest
    unittest.main()
```

Eg.Screenshot



```
(env) D:\Project\Eventoplanneur\event>py manage.py test
Found 1 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).

DevTools listening on ws://127.0.0.1:54324/devtools/browser/16935ae2-78c9-4166-9b47-b791e36ce689
.
----------------------------------------------------------------------
Ran 1 test in 40.826s

OK
Destroying test database for alias 'default'...
```

Test Case 4

Project Name: EventoPlanneur: Event Management System					
Event Recommendation Test Case					
Test Case ID: Test_4			**Test Designed By: Elizebath Thomas**		
Test Priority (Low/Medium/High): High			**Test Designed Date: 13-10-2023**		
Module Name: Event Recommendation Screen			**Test Executed By: Dr. Paulin Paul**		
Test Title: Event Recommendation Testing			**Test Execution Date: 14-10-2023**		
Description: Login to attendee user and view webinars					
Pre-Condition: Users has registered as attendee user					
Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/ Fail)**
1	Navigate to Login Page		Login Page should be displayed	Login Page displayed	Pass
2	Provide valid e-mail ID	E-mail ID: elizatom9@gmail.com	User should be able to login	User logged in and navigated to user dashboard	Pass
3	Provide valid password	Password: Eliza@123			
4	Click sign In Button				
5	Click on Events		Webinar recommendations must be displayed	Webinar recommendations page is displayed	Pass
6	Click on View button		Corresponding webinar details must be displayed	Webinar details is displayed	Pass
Post-Condition: The webinars are successfully displayed from database and recommended to the user using NLP.					

CHAPTER 6
IMPLEMENTATION

6.1INTRODUCTION

The project's implementation phase is when the theoretical design is transformed into a functional system. Gaining users' trust that the new system will function and be accurate, efficient, and successful might be seen as the most important step in the process of creating a successful new system. Documentation and user training are its main concerns. Conversion often occurs later, or around the time the user is receiving training. Implementation is the process of turning a newly revised system design into an operational one. It is simply defined as calling a new system design into operation.

At this point, the user department is responsible for most of the workload, the most disruption, and the biggest influence on the current system. An unplanned or poorly managed installation can lead to confusion and mayhem.

All of the actions necessary to switch from the current system to the new one is included in the implementation. The new system might be an entirely new one that replaces the current automated or manual system, or it could be an existing system that has been modified. To deliver a dependable system that satisfies organizational needs, proper implementation is necessary. System implementation is the process of implementing the developed system. This covers all the procedures involved in switching from the old to the new system. Only after extensive testing and confirmation that the system is operating in accordance with the requirements can the system be put into use. The system staff verifies the system's viability. The system analysis and design work needed to accomplish the three primary components—education and training, system testing, and changeover—will increase with the complexity of the system being implemented.

The following tasks are included in the implementation state:

- Meticulous planning.
- Examining the restrictions and system.
- Creating the strategies to make the transition.

6.2 IMPLEMENTATION PROCEDURES

Software implementation is the process of installing a package in its final form in a real-world setting and making sure it functions as intended. The software development project will often be commissioned by a person who will not be using it. People have doubts about the software at first, but we need to make sure that resistance does not grow. To achieve this, we need to make sure that:

- The active user is aware of the advantages of utilizing the new system.
- Their faith in the program is bolstered.
- The user receives appropriate instruction to ensure his comfort level when utilizing the program.

The user must be aware that the server software must be running on the server to examine the system's results before proceeding. The real procedure will not occur if the server object is not up and running on the server.

6.2.1 User Training

The purpose of user training is to get the user ready for system conversion and testing. The individuals who will be participating must feel secure in their function inside the new system to achieve the goal and reap the benefits anticipated from a computer-based system. Training becomes increasingly necessary as systems get more complicated. Through user training, the user learns how to query the database, enter data, respond to error signals, call up routines that generate reports, and carry out other essential tasks.

6.2.2 Training on the Application Software

The user must get training on the new application software after receiving the required foundational instruction in computer awareness. This will provide the fundamental ideas behind using the new system, including the screen flow, screen design, types of on-screen help, types of data entry problems, the matching validation check at each entry, and methods for entering dates correctly. After providing the program's training on the application, it should go over the information required by that user or group to utilize the system or a portion of it. The nature of this training may vary depending on the user group and hierarchical level.

6.2.3 System Maintenance

The mystery of system development is maintenance. A software product's useful work is done during the maintenance phase of its lifecycle. Following a system's successful implementation, it must be properly maintained. An essential part of the software development life cycle is system maintenance. System maintenance is necessary to keep the system flexible enough to respond to changes in its surroundings. Of fact, software maintenance involves much more than just "Finding Mistakes".

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

"EventoPlanneur: Official Events Management System" is a groundbreaking platform with a singular goal: to revolutionize event management for colleges, organizers, service providers, and attendees. It offers a comprehensive solution to simplify event planning and participation, enhancing the experience for all stakeholders involved. This innovative system boasts user-friendly interfaces tailored to the unique needs of admins, colleges, organizers, and attendees, streamlining event registration, management, and engagement processes. Admins gain the tools they need to efficiently oversee events, colleges benefit from simplified registration and enhanced promotion capabilities, organizers enjoy intuitive event planning resources, and attendees revel in a seamless event participation journey. Prioritizing security and scalability, the platform ensures the protection of user data and accommodates growth seamlessly. The success of EventoPlanneur will be cemented through astute marketing strategies aimed at attracting and engaging colleges, organizers, and attendees, establishing it as the go-to hub for all things related to events.

7.2 FUTURE SCOPE

The future of "EventoPlanneur: Official Events Management System" is bright, with constant growth and development anticipated. With the addition of features like AI-driven suggestions and integrated payments, this versatile platform hopes to advance and increase the effectiveness of event management. Creating specialized mobile applications will improve user accessibility. Increased event marketing and attendance may be achieved through a smooth interface with social media platforms. The platform's reach grows as it develops, going beyond universities to include a range of event kinds. The user experience will be further enhanced with the introduction of AI chatbots, user-generated content, and powerful data analytics. EventoPlanneur has the potential to develop into a comprehensive and cutting-edge event management and planning solution.

.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

- Web development using Python and Django with Sqlite3: This Stack Overflow post discusses building a web-based dashboard using Django framework with Sqlite3 database. It also covers topics such as using HTML, CSS for front-end designs, and plotly library for data visualization
- Working with AJAX in Django on Pluralsight: This guide provides an overview of how to work with AJAX in Django. It covers topics such as how to create an AJAX view, how to handle AJAX requests, and how to use AJAX with forms 3.
- Working with AJAX in Django on Pluralsight: This guide provides an overview of how to work with AJAX in Django. It covers topics such as how to create an AJAX view, how to handle AJAX requests, and how to use AJAX with forms 3.
- Working with AJAX in Django on Pluralsight: This guide provides an overview of how to work with AJAX in Django. It covers topics such as how to create an AJAX view, how to handle AJAX requests, and how to use AJAX with forms 3.

WEBSITES:

- https://stackoverflow.com/questions/52585028/web-development-using-python-and-django-with-sqlite3-inbuilt-db-in-django

- https://www.w3schools.com/xml/ajax_database.asp

- https://www.pluralsight.com/guides/work-with-ajax-django

- https://www.w3schools.com/django/index.php

CHAPTER 9
APPENDIX

9.1 Sample Code

Views.py

```
from django.contrib import messages
from django.shortcuts import render, redirect,get_object_or_404
from .models import CustomUser,Webinar,EventOrganizer,AICTE,Speaker,Conference,WebinarRegistration,Attendee
import re
from django.conf import settings
from django.contrib.auth import authenticate, login as auth_login
from django.contrib.auth.models import User,auth
from .forms import WebinarForm, Organizer,ConferenceForm,AttendeeForm
from django.contrib.auth.decorators import login_required
from django.contrib.auth import logout as auth_logout
from django.forms import modelformset_factory
from django.http import HttpResponse
from django.urls import reverse
from django.core.mail import send_mail
from django.utils import timezone
# import requests
from django.http import JsonResponse
from django.utils.crypto import get_random_string
def is_valid_email(email):
    pattern = r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'
    if not re.match(pattern, email):
        return False
    return True
def is_valid_password(password):
    pattern = r'^(?=.*[0-9])(?=.*[!@#$%^&*])(?=.*[A-Z]).{8,}$'
    if not re.match(pattern, password):
        return False
    return True
def is_valid_name(name):
    pattern = r'^[A-Za-z\s]+$'
    if not re.match(pattern, name):
        return False
    return True
def registration(request):
    if request.method == 'POST':
        email = request.POST.get('email')
        password = request.POST.get('password')
        user_type = request.POST.get('userType')
        if not is_valid_email(email):
            messages.error(request, 'Invalid email')
            return render(request, 'register.html')
        if not is_valid_password(password):
            messages.error(request, 'Password must be at least 8 characters long and contain at least one number, one
symbol, and one capital letter')
            return render(request, 'register.html')
        if CustomUser.objects.filter(email=email).exists():
            messages.error(request, "Email already exists.")
        user = CustomUser.objects.create_user(email=email, password=password)
        token = get_random_string(length=32)
        user.verification_token = token
        user.is_verified = False
        if user_type == 'eventOrganizer':
            user.is_organizer = True
        elif user_type == 'serviceProvider':
            user.is_provider = True
        elif user_type == 'attendee':
            user.is_provider = True
        user.save()
        send_mail(
            'Email Verification',
            f'Click the following link to verify your email: {request.build_absolute_uri("/verify/")}?token={token}',
            'eventoplanneur@gmail.com',
            [email],
            fail_silently=False,
        )
        return redirect('eventapp:verify')
    return render(request, 'register.html')
def verify(request):
    token = request.GET.get('token')
    user = CustomUser.objects.filter(verification_token=token).first()
    if user:
        user.is_verified = True
        user.verification_token = None
        user.save()
        return redirect('/')  # Redirect to login page after successful verification
    else:
        return render(request, 'invalid_token.html')  # Handle invalid token
def login(request):
    if request.method == 'POST':
        email = request.POST.get('email')
        password = request.POST.get('password')
        if email and password:
            user = authenticate(request, email=email, password=password)
            if user is not None and user.is_verified:
                auth_login(request, user)
                request.session['user_id'] = user.id
                # request.session['user_type'] = 'organizer'
                if user.is_organizer:
                    return redirect('eventapp:orghome')
                if user.is_attendee:
                    return redirect('eventapp:attendeehome')
                if user.email=="admin@gmail.com":
                    return redirect('eventapp:admindash')
                else:
                    return redirect('/')
            else:
                try:
                    user = CustomUser.objects.get(email=email)
                    messages.error(request, "Email not verified or Incorrect password")
                except CustomUser.DoesNotExist:
                    messages.error(request, "Email not registered")
        else:
            messages.error(request, "Please provide both email and password")
    return render(request, 'login.html')
@login_required
def webinar(request):
    orgs=request.user
    update_webinar=Webinar.objects.filter(org_user=orgs, status=1)
    context = {'update_webinar': update_webinar}
    return render(request, 'webinar.html', context)
def view_webinar(request,update_id):
    task=Webinar.objects.get(id=update_id)
    form=WebinarForm(request.POST or None,instance=task)
    speakers = task.speakers.all()
    return render(request,'view_webinar.html',{'form':form,'speakers': speakers})

@login_required
def delete_webinar(request, del_id):
    webinar = Webinar.objects.get(id=del_id)
    organizer_email = webinar.org_user.email
    subject = 'Webinar Deleted'
    message = f'The webinar "{webinar.title}" on {webinar.date} at {webinar.start_time} has been deleted.'
    from_email = 'eventoplanneur@gmail.com'
    recipient_list = [organizer_email]
    # participants also
    send_mail(subject, message, from_email, recipient_list)
    webinar.status=0
    webinar.save()
    return redirect('eventapp:webinar')
@login_required
def register_webinar(request):
    if request.method == 'POST':
        form = WebinarForm(request.POST)
        if form.is_valid():
            title=request.POST.get('title')
            date=request.POST.get('date')
            existing_webinar = Webinar.objects.filter(title=title, date=date)
            if existing_webinar:
                # If a webinar with the same name and date exists, show an error message
                messages.error(request, "A webinar with the same name and date already exists.")
            else:# Save the webinar without committing to the database yet
                webinar = form.save(commit=False)
                # Set the organizer to the currently logged-in user
                webinar.org_user = request.user
                webinar.save()  # Commit the webinar to the database
                speakers_designation = request.POST.getlist('speakers_designation[]')
                speakers_name = request.POST.getlist('speakers_name[]')
                for i in range(len(speakers_designation)):
                    speaker = Speaker.objects.create(
                        designation=speakers_designation[i],
                        speaker_name=speakers_name[i]
                    )
                    webinar.speakers.add(speaker)  # Add the speaker to the webinar
                recipient_email = request.user.email
                subject = 'Webinar Registration Confirmation'
                message = f'Thank you for registering the webinar: {webinar.title} on {webinar.date} from
{webinar.start_time} to {webinar.end_time}.'
                from_email = 'eventoplanneur@gmail.com'  # Replace with your email address
                recipient_list = [recipient_email]  # Use the organizer's email or another recipient
                send_mail(subject, message, from_email, recipient_list)
                messages.success(request, "Webinar saved successfully")
                return redirect('eventapp:register_webinar')
        else:
            print(form.errors)
            messages.error(request, form.errors)
    else:
        form = WebinarForm()
    return render(request, 'register_webinar.html', {'form': form})
@login_required
def org_profile(request):
    org_user = request.user
    try:
        organizer_instance = EventOrganizer.objects.get(org_user=org_user)
    except EventOrganizer.DoesNotExist:
        organizer_instance = None
    if request.method == "POST":
        if organizer_instance:
            form = Organizer(request.POST, instance=organizer_instance)
        else:
            form = Organizer(request.POST)

        if form.is_valid():
            event_organizer = form.save(commit=False)
            event_organizer.org_user = org_user
            event_organizer.save()
            return redirect('eventapp:orghome')
    else:
        form = Organizer(instance=organizer_instance)
    messages.error(request, form.errors)
    return render(request, 'org_profile.html', {'form': form})
def update_org_profile(request):
    orgs=request.user
    task=EventOrganizer.objects.get(org_user=orgs)
    form=Organizer(request.POST or None,instance=task)
    if form.is_valid():
        form.save()
        return redirect('eventapp:update_org_profile')
    return render(request,'update_webinar.html',{'form':form})
def update_webinar(request, update_id):
    webinar = Webinar.objects.get(id=update_id)
    speakers = webinar.speakers.all()
    if request.method == 'POST':
        form = WebinarForm(request.POST, instance=webinar)
        if 'event_type' in request.POST and request.POST['event_type'] == 'Offline':
            form.fields['livestream'].initial = 'None'
        if form.is_valid():
            new_date=request.POST.get('date')
            new_time=request.POST.get('start_time')
            if webinar.date != new_date or webinar.time != new_time:
                subject = 'Webinar Date and Time Update'
                message = f'The date and time for the webinar "{webinar.title}" have been updated.\n\nNew Date:
{new_date}\nNew Time: {new_time}'
                from_email = 'eventoplanneur@gmail.com'  # Replace with your email address
                recipient_email = ['elizebaththomasv@gmail.com']
                send_mail(subject, message, from_email, recipient_email)
              web = form.save(commit=False)
            # web.max_participants=max
            web.org_user = request.user
            web.save()
            for speaker in speakers:
                designation_field = f"speakers-{speaker.id}-designation"
                name_field = f"speakers-{speaker.id}-speaker_name"
                speaker.designation = request.POST.get(designation_field)
                speaker.speaker_name = request.POST.get(name_field)
                speaker.save()
            messages.success(request, "Webinar updated successfully")
            return redirect('update_webinar', update_id=update_id)
        else:
            messages.error(request, form.errors)
    else:
        form = WebinarForm(instance=webinar)  # Initialize form with webinar data
    return render(request, 'update_webinar.html', {'form': form, 'speakers': speakers, 'webinar': webinar})
def check_aicte_id(request):
    aicte_id = request.GET.get('aicte_id')
    try:
        aicte = AICTE.objects.get(aicte_id=aicte_id)
        return JsonResponse({'valid': True, 'name': aicte.name, 'location': aicte.location,'address': aicte.address})
    except AICTE.DoesNotExist:
        return JsonResponse({'valid': False, 'name': None, 'location': None,'address': None})
```

```
@login_required
def conference(request):
    orgs=request.user
    con=Conference.objects.filter(org_user=orgs)
    context = {'con': con}
    return render(request, 'conference.html', context)
def view_conference(request,view_id):
    task=Conference.objects.get(id=view_id)
    form=ConferenceForm(request.POST or None,instance=task)
    speakers = task.speakers.all()
    return render(request,'view_conference.html',{'form':form,'speakers': speakers})
@login_required
def delete_conference(request, del_id):
    conference = Conference.objects.get(id=del_id)
    organizer_email = conference.org_user.email
    subject = 'Conference Deleted'
    message = f'The conference "{ conference.title}" planned from { conference.start_date} to { conference.end_date}
has been deleted.'
    from_email = 'eventoplanneur@gmail.com'
    recipient_list = [organizer_email]
    send_mail(subject, message, from_email, recipient_list)
    conference.delete()
    return redirect('eventapp:conference')
@login_required
def register_conference(request):
    if request.method == 'POST':
        form = ConferenceForm(request.POST)
        if form.is_valid():
            conference = form.save(commit=False)
            conference.org_user = request.user
            # Save the conference object to the database first
            conference.save()

            speakers_designation = request.POST.getlist('speakers_designation[]')
            speakers_name = request.POST.getlist('speakers_name[]')
            for i in range(len(speakers_designation)):
                speaker = Speaker.objects.create(
                    designation=speakers_designation[i],
                    speaker_name=speakers_name[i]
                )
                # Now that the conference is saved, you can add speakers to it
                conference.speakers.add(speaker)

            recipient_email = request.user.email
            subject = 'Conference Registration Confirmation'
            message = f'Thank you for registering the conference: {conference.title} from {conference.start_date} to
{conference.end_date}.'
            from_email = 'eventoplanneur@gmail.com'  # Replace with your email address
            recipient_list = [recipient_email]  # Use the organizer's email or another recipient

            send_mail(subject, message, from_email, recipient_list)

            interested_users = ['elizebaththomasv@gmail.com', 'elizatom9@gmail.com']  # Replace with actual email
addresses
            conference_link = 'http://127.0.0.1:8000/'  # Replace with the actual webinar details page URL
            email_subject = f'Upcoming Conference: {conference.title}'
            email_message = f'Hello,\n\nThere is an upcoming conference that you may be interested in: {conference.title}
from {conference.start_date} to {conference.end_date}.\n\nYou can find more details and register for the  conference
here: {conference_link}.\nRegistration closes by: {conference.deadline}.\n\nPoster Link: {conference.poster}'
            from_email = 'eventoplanneur@gmail.com'
            send_mail(email_subject, email_message, from_email, interested_users)
            messages.success(request, "Conference saved successfully")
            return redirect('eventapp:register_conference')
        else:
            print(form.errors)
            messages.error(request, form.errors)
    else:
        form = ConferenceForm()
    return render(request, 'register_conference.html', {'form': form})
def update_conference(request, update_id):
    conference = Conference.objects.get(id=update_id)
    if request.method == 'POST':
        form = ConferenceForm(request.POST, instance=conference)
        if form.is_valid():
            form.save()
            for speaker in conference.speakers.all():
                designation_field = f"speakers-{speaker.id}-designation"
                name_field = f"speakers-{speaker.id}-speaker_name"
                speaker.designation = request.POST.get(designation_field)
                speaker.speaker_name = request.POST.get(name_field)
                speaker.save()
            messages.success(request, "Conference and speakers updated successfully")
            return redirect('eventapp:update_conference', update_id=update_id)
        else:
            messages.error(request, form.errors)
    else:
        form = ConferenceForm(instance=conference)
    speakers = conference.speakers.all()
    return render(request, 'update_conference.html', {'form': form, 'speakers': speakers})

def listwebinars(request):
    # Filter webinars whose date is greater than today
    today = timezone.now().date()
    allwebinars = Webinar.objects.filter(deadline__gt=today, status=1)
    context = {'allwebinars': allwebinars}
    return render(request, 'listwebinars.html', context)

def register_for_webinar(request, webinar_id):
    user = request.user
    webinar = Webinar.objects.get(pk=webinar_id)
    registration_count = WebinarRegistration.objects.filter(webinar=webinar).count()
    if registration_count < webinar.max_participants:
        if not WebinarRegistration.objects.filter(user=user, webinar=webinar).exists():
            WebinarRegistration.objects.create(user=user, webinar=webinar)
            messages.success(request, "Webinar registered successfully")
            recipient_email = request.user.email
            subject = 'Webinar Registration Confirmation'
            message = f'Thank you for registering the webinar: {webinar.title} on {webinar.date} '
            from_email = 'eventoplanneur@gmail.com'
            recipient_list = [recipient_email]

            send_mail(subject, message, from_email, recipient_list)
            pay_id=webinar_id
            return redirect('payment', pay_id=pay_id)
        else:
            messages.success(request, "You are already registered for this webinar.")
            return redirect('eventapp:events')
    else:
        messages.error(request, "Webinar reached maximum number of participants")
        return redirect('eventapp:events')
@login_required
```

```
def registered_webinar(request):
    user = request.user
    webinar_registrations = WebinarRegistration.objects.filter(user=user)
    context = {'webinar': webinar_registrations}
    return render(request, 'registered_webinar.html', context)
from django.shortcuts import render
import razorpay
from django.conf import settings
from django.views.decorators.csrf import csrf_exempt
from django.http import HttpResponseBadRequest
razorpay_client = razorpay.Client(
    auth=(settings.RAZOR_KEY_ID, settings.RAZOR_KEY_SECRET))
def payment(request,pay_id):
    webinar = Webinar.objects.get(pk=pay_id)
    currency = 'INR'
    amount = int(webinar.fee)*100
    razorpay_order = razorpay_client.order.create(dict(amount=amount,
                                                       currency=currency,
                                                       payment_capture='0'))
    razorpay_order_id = razorpay_order['id']
    callback_url = 'eventapp:paymenthandler'
    context = {}
    context['razorpay_order_id'] = razorpay_order_id
    context['razorpay_merchant_key'] = settings.RAZOR_KEY_ID
    context['razorpay_amount'] = amount
    context['currency'] = currency
    context['callback_url'] = callback_url

    return render(request, 'payment.html', context=context)

@csrf_exempt
def paymenthandler(request):

    # only accept POST request.
    if request.method == "POST":
        try:
            payment_id = request.POST.get('razorpay_payment_id', '')
            razorpay_order_id = request.POST.get('razorpay_order_id', '')
            signature = request.POST.get('razorpay_signature', '')
            params_dict = {
                'razorpay_order_id': razorpay_order_id,
                'razorpay_payment_id': payment_id,
                'razorpay_signature': signature
            }
            result = razorpay_client.utility.verify_payment_signature(
                params_dict)
            if result is not None:
                amount = 20000  # Rs. 200
                try:
                    razorpay_client.payment.capture(payment_id, amount)
                    return render(request, 'paymentsuccess.html')
                except:
                    return render(request, 'paymentfail.html')
            else:
                return render(request, 'paymentfail.html')
        except:
            return HttpResponseBadRequest()
    else:
        return HttpResponseBadRequest()

def paymentsuccess(request):
    return render(request, 'paymentsuccess.html')

def paymentfail(request):
    return render(request, 'paymentfail.html')

@login_required
def attendee_profile(request):
    org_user = request.user

    try:
        organizer_instance = Attendee.objects.get(org_user=org_user)
    except Attendee.DoesNotExist:
        organizer_instance = None

    if request.method == "POST":
        if organizer_instance:
            form = AttendeeForm(request.POST, instance=organizer_instance)
        else:
            form = AttendeeForm(request.POST)

        if form.is_valid():
            attendee = form.save(commit=False)
            interests = request.POST.get('interests', '')  # Get the interests as a comma-separated string
            attendee.interests = interests
            attendee.org_user = org_user
            attendee.save()
            return redirect('eventapp:attendeehome')

    else:
        initial_data = {}  # Create a dictionary to store initial data for the form
        if organizer_instance and organizer_instance.interests:
            # Split the comma-separated interests string into a list
            initial_data['interests'] = organizer_instance.interests.split(', ')
        form = AttendeeForm(instance=organizer_instance, initial=initial_data)

    return render(request, 'attendee_profile.html', {'form': form})

from django.shortcuts import render
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from .models import Webinar, Attendee

def recommendations(request, N=6):
    try:
        attendee_id=request.user.id
        attendee = Attendee.objects.get(org_user=attendee_id)
        attendee_interests = attendee.interests

        all_events = Webinar.objects.all()

        event_description_to_webinar = {event.description: event for event in all_events}
        tfidf_vectorizer = TfidfVectorizer()
        tfidf_matrix_events = tfidf_vectorizer.fit_transform(all_events.values_list('description', flat=True))
        tfidf_matrix_attendee = tfidf_vectorizer.transform([attendee_interests])
        similarity_scores = cosine_similarity(tfidf_matrix_attendee.toarray(), tfidf_matrix_events)

        sorted_event_indices = similarity_scores.argsort()[0][::-1]

        sorted_event_indices = sorted_event_indices.tolist()

        top_N_events = [event_description_to_webinar[all_events[i].description] for i in sorted_event_indices[:N]]
        return render(request, 'recommendations.html', {'recommended_events': top_N_events})

    except Attendee.DoesNotExist:
        # Handle the case where the attendee doesn't exist
        return render(request, 'recommendations.html', {'recommended_events': []})
```

index.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>EventoPlanneur</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="icon" href="{% static 'images/logo.jpg' %}" type="image/x-icon">
  <link rel="stylesheet" href="{% static 'css/index.css' %}">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">
</head>
<body>
  <nav class="navbar navbar-expand-md bg-dark navbar-dark">
    <div class="container">
      <a class="navbar-brand" >
        <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
      </a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarCollapse">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item"><a class="nav-link" href="#about">About Us</a></li>
          <li class="nav-item"><a class="nav-link" href="#contact">Contact</a></li>
          <li class="nav-item"><a class="nav-link" href="#gallery">Gallery</a></li>
          <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
          {% if user.is_authenticated %}
          {% if user.is_organizer %}
            <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:orghome' %}">Home Page</a></li>
          {% endif %}
          {% if user.is_attendee %}
            <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:attendeehome' %}">Home Page</a></li>
          {% endif %}
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:logout' %}">Logout</a></li>
        {% else %}
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:login' %}">Login</a></li>
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:registration' %}">Register</a></li>
        {% endif %}
        </ul>
      </div>
    </div>
  </nav>
  <section class="hero-section">
    <h1 class="hero-heading">Welcome to Our Event Planning Service</h1>
    <p class="hero-tagline">Creating unforgettable experiences for all occasions</p>
  </section>
  <section class="about-section" id="about">
    <p class="about-description">
      Welcome to EventoPlanneur! We are passionate about creating unforgettable experiences for your special
occasions. Our mission is to bring your vision to life and make every moment truly remarkable. With a team of
dedicated professionals, we are committed to turning your dreams into reality.
    </p><br>
    <div class="team-members">
    <a href="{% url 'eventapp:reg_provider' %}">
      <div class="team-member">
        <img class="team-image" src="{% static 'images/user1.jpg' %}" alt="Service Provider">
        <div class="team-description">
          <h3 class="team-name">Service Provider</h3>
          <p>Do you offer unique services that can enhance events and create memorable experiences? Join as a Service Provider and showcase your expertise to a wide audience. Whether you specialize in catering, photography, entertainment, decor, or any other event-related service, our platform provides you with a platform to connect with event organizers and showcase your offerings. Collaborate, build your portfolio, and become an integral part of creating extraordinary events. Sign up today and be a part of making every event unforgettable!</p>
        </div>
      </div>
    </a>

    <a href="{% url 'eventapp:reg_organizer' %}">
      <div class="team-member">
        <img class="team-image" src="{% static 'images/user2.jpg' %}" alt="Event Organizer">
        <div class="team-description">
          <h3 class="team-name">Event Organizer</h3>
          <p>Are you passionate about planning and executing unforgettable events? Join as an Event Organizer and bring your creative ideas to life. Collaborate with clients, vendors, and partners to create seamless and memorable experiences for a wide range of events, from weddings and conferences to festivals and corporate gatherings. Whether you're an experienced event planner or looking to launch your career in the dynamic world of event management, our platform provides the tools and resources you need to succeed. Get started today and make every event a masterpiece!</p>
        </div>
      </div>
    </a>
    <a href="{% url 'eventapp:reg_attendee' %}">
      <div class="team-member">
        <img class="team-image" src="{% static 'images/user3.jpg' %}" alt="Attendee">
        <div class="team-description">
          <h3 class="team-name">Attendee</h3>
          <p>Experience the excitement of being part of incredible events as an Attendee. Explore a variety of events, from conferences and workshops to concerts and festivals. Register and participate in events that match your interests and preferences. Connect with fellow attendees, engage in meaningful discussions, and create unforgettable memories. Our platform offers a seamless experience to discover, register, and engage in events that matter to you. Join us today and be a part of the event community!</p>
        </div>
      </div>
    </a>
  </div>
 </section>
<section class="cta-section">
  <h2 class="cta-heading">Ready to Get Started?</h2>
  <p class="cta-description">
    Explore our services and make your next event extraordinary. Join us in creating memorable moments together!
  </p>
  <a id="services" class="cta-button" href="">Explore Services</a>
</section>
<section class="featured-section">
  <div class="feature">
    <img class="feature-image" src="{% static 'images/plan.jpg' %}" alt="Feature 1">
    <h2 class="feature-title">Event Planning</h2>
  </div>
  <div class="feature">
    <img class="feature-image" src="{% static 'images/booking.jpg' %}" alt="Feature 2">
    <h2 class="feature-title">Service Booking</h2>
  </div>
  <div class="feature">
    <img class="feature-image" src="{% static 'images/register.png' %}" alt="Feature 3">
    <h2 class="feature-title">Event Registration</h2>
  </div>
</section>
<section class="testimonials-section" id="reviews">
  <div class="testimonial">
    <p class="testimonial-text">EventoPlanneur made family meet absolutely magical! Their attention to detail and
creative ideas exceeded our expectations.</p>
    <p class="testimonial-author">Sarah and John</p>
  </div>
  <div class="testimonial">
    <p class="testimonial-text">We couldn't have asked for a better event planning team. They turned our corporate
event into a memorable experience for all attendees.</p>
    <p class="testimonial-author">Michael Smith</p>
  </div>
  <div class="testimonial">
    <p class="testimonial-text">Exceptional service from start to finish. EventoPlanneur's dedication and
professionalism made our company anniversary celebration truly special.</p>
    <p class="testimonial-author">Emily and David</p>
  </div>
</section>
<section class="blog-section">
  <div class="blog-post">
    <h2 class="post-title">10 Tips for Planning a Successful Corporate Event</h2>
    <p class="post-date">Published on August 10, 2023</p>
    <p class="post-content">Planning a corporate event requires careful attention to detail and a strategic approach. In
this article, we provide you with valuable tips to ensure your next corporate event is a resounding success...</p>
  </div>
  <div class="blog-post">
    <h2 class="post-title">Navigating Event Venue Selection: Finding the Ideal Space</h2>
    <p class="post-date">Published on July 28, 2023</p>
    <p class="post-content">Choosing the right venue for your event is a crucial decision that can greatly impact its
success. Discover key factors to consider when selecting the perfect event venue...</p>
  </div>
</section>
<section class="contact-section" id="contact">
  <div class="contact-details">
    <h2 class="contact-heading">Contact Us</h2>
    <p class="contact-info">Have questions or need assistance? Feel free to get in touch with us.</p>
    <div class="contact-form">
      <form action="submit_form.php" method="post">
        <div class="form-group">
          <label class="form-label" for="name">Your Name:</label>
          <input class="form-input" type="text" id="name" name="name" required>
        </div>
        <div class="form-group">
          <label class="form-label" for="email">Your Email:</label>
          <input class="form-input" type="email" id="email" name="email" required>
        </div>
        <div class="form-group">
          <label class="form-label" for="message">Message:</label>
          <textarea class="form-message" id="message" name="message" rows="4" required></textarea>
        </div>
        <button class="form-button" type="submit">Send Message</button>
      </form>
    </div>
    <div class="social-media">
      <a class="social-link" href="https://www.facebook.com/yourpage" target="_blank"><i class="fab fa-
facebook"></i></a>
      <a class="social-link" href="https://www.twitter.com/yourpage" target="_blank"><i class="fab fa-
twitter"></i></a>
      <a class="social-link" href="https://www.instagram.com/yourpage" target="_blank"><i class="fab fa-
```

```
instagram"></i></a>
          <a class="social-link" href="https://www.linkedin.com/company/yourpage" target="_blank"><i class="fab fa-
linkedin"></i></a>
      </div>
    </div>
</section>
  <footer class="footer">
    <div class="footer-nav">
      <a href="terms.html">Terms of Use</a>
      <a href="privacy.html">Privacy Policy</a>
      <a href="faq.html">FAQ</a>
    </div>
    <p class="copyright">&copy; 2023 EventoPlanneur. All rights reserved.</p>
    <p class="disclaimer">Disclaimer: The information provided on this website is for general informational purposes
only and should not be considered as professional advice.</p>
</footer>
</body>
</html>
```

Login.html

```
{% load socialaccount %}
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="icon" href="{% static 'images/Eventoplanneur.png' %}" type="image/x-icon">
  <link rel="stylesheet" href="{% static 'css/style.css' %}">
</head>
<body>
  <div class="login-box">
    <div class="image-container">
      <img src="{% static 'images/logo.jpg' %}" alt="Site Logo">
    </div>
    <h2>Login</h2>
    <form method="POST" action="{% url 'eventapp:login' %}">
      {% csrf_token %}
      <div class="form-group">
        <input type="text" class="form-control" id="email" name="email" placeholder="Email">
        <span id="emailError" class="error-text"></span>
    </div>
    <div class="form-group">
      <input type="password" class="form-control" id="password" name="password" placeholder="Password">
      <span id="passwordError" class="error-text"></span>
  </div>
  <p class="forgot-password"><a href="{% url 'eventapp:reset_password' %}">Forgot Password?</a></p>
      <button type="submit" id="login" class="btn btn-primary btn-block register-btn">Login</button>
      <p class="text-center mt-3"> <a href="{% url 'eventapp:registration' %}">Don't have an account?
<b>Register</b></a></p>
      <button class="signupBtn" ><a href="{% provider_login_url 'google'%}?next=/">
        Login with Google</a>
        <span class="arrow">
           <svg xmlns="http://www.w3.org/2000/svg" height="1em" viewBox="0 0 320 512" fill="rgb(183, 128,
255)"><path d="M278.6 233.4c12.5 12.5 12.5 32.8 0 45.3l-160 160c-12.5 12.5-32.8 12.5-45.3 0s-12.5-32.8 0-
45.3L210.7 256 73.4 118.6c-12.5-12.5-12.5-32.8 0-45.3s32.8-12.5 45.3 0l160 160z"></path></svg>
        </span>
      </button>
      {% comment %} <button type="submit" class="btn btn-primary btn-block register-btn"<a href="{%
provider_login_url 'google'%}?next=/">Login with Google</a><button> {% endcomment %}
      {% if messages %}
<div class="alert alert-danger">
  <ul>
    {% for message in messages %}
    <li>{{ message }}</li>
    {% endfor %}
  </ul>
</div>
{% endif %}
    </form>
  </div>
  <script src="{% static 'js/validation.js' %}"></script>
</body>
</html>
```

Register.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Registration</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="icon" href="{% static 'images/logo.jpg' %}" type="image/x-icon">
  <link rel="stylesheet" href="{% static 'css/style.css' %}">
</head>
<body>
  <div class="login-box">
    <div class="image-container">
        <img src="{% static 'images/logo.jpg' %}" alt="Site Logo">
      </div>
    <h2>Register</h2>
    <form id="login-form" method="POST" action="{% url 'eventapp:reg_organizer' %}">
      {% csrf_token %}
      <div class="form-group">
        <input type="text" class="form-control" id="email" name="email" placeholder="Email">
        <span id="emailError" class="error-text"></span>
    </div>
    <div class="form-group">
        <input type="password" class="form-control" id="password" name="password" placeholder="Password">
        <span id="passwordError" class="error-text"></span>
    </div>
    <div class="form-group">
        <input type="password" class="form-control" id="confirm_password" name="confirm_password"
placeholder="Confirm Password">
        <span id="confirmPasswordError" class="error-text"></span>
    </div>
      <button type="submit" class="btn btn-primary btn-block register-btn">Register</button>
      <p class="text-center mt-3"><a href="{% url 'eventapp:login' %}">Already have an account?
<b>Login</b></a></p>
    </form>
    {% if messages %}
<div class="alert alert-danger">
    {% for message in messages %}
      {{ message }}
    {% endfor %}
</div>
{% endif %}
  </div>
  <script src="{% static 'js/validation.js' %}"></script>
</body>
</html>
```

Organizer_home.html

```
{% load static %}
{% if user.is_authenticated %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Homepage</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet" href="{% static 'css/home.css' %}">
  <link rel="icon" href="{% static 'images/eventoplanneur.png' %}" type="image/x-icon">
</head>
<body>
  <nav class="navbar navbar-expand-md bg-dark navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="#">
        <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
      </a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarCollapse">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item"><a class="nav-link" href="#services">Services</a></li>
          <li class="nav-item"><a class="nav-link" href="#contact">Contact</a></li>
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:gallery' %}">Gallery</a></li>
          <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:org_profile' %}">Profile</a></li>
          <li class="nav-item"><a class="nav-link" href="#notifications">Notifications</a></li>
          <li class="nav-item">
              {% if user.is_authenticated %}
              <a class="nav-link" >User: {{ user.email }}</a>
              <li class="nav-item"><a href="{% url 'eventapp:logout' %}" class="nav-link">Logout</a></li>
              {% endif %}
            </li>
        </ul>
      </div>
    </div>
  </nav>
  <section class="intro" style="background:linear-gradient(to left top, #DD2476 10%, #FF512F 90%);">
    <div class="intro-content">
      <h2>Welcome to EventoPlanneur</h2>
      <p>Where Every Occasion Finds Its Perfect Stage!</p>
      <p>Our site, a hub of connection and inspiration, bridges passions, fuels creativity, and empowers users to collaborate seamlessly, shaping limitless possibilities together. Our comprehensive site offers seamless event planning, registration, ticketing, and promotion tools. Engage attendees with interactive features, real-time analytics, and adaptable interfaces, ensuring successful, unforgettable events.</p>
    </div>
  </section>
  <section id="event-listing" class="event-listing-section">
```

```
    <div class="container">
      <div class="search-filter">
        <input type="text" id="search-input" placeholder="Search events">
        <select id="category-select">
          <option value="">All Categories</option>
          <option value="conference">
            <a href="">Conference</a>
        </option>
        <option value="social">
            <a href="">Social Events</a>
        </option>
            <a href="{% url 'eventapp:webinar' %}"><option value="webinar">Webinar</option></a>
        </select>
        <button id="apply-filter" class="btn">Apply Filter</button>
      </div>

      <div class="event-list">
      </div>
    </div>
  </section>
  <section id="featured-events" class="featured-section">
    <div class="container">
      <h2 class="text-center mb-4">Featured Events</h2>
      <div class="row">
        <div class="col-md-4 mb-4" ><a href="{% url 'view_webinar' 66 %}">
          <div class="feature">
            <img src="{% static 'images/conference1.jpg' %}" alt="Featured Event 1" class="feature-image">
            <h3 class="feature-title">Cyber Security Asia 2023</h3>
            <p class="feature-description">Join industry leaders for a groundbreaking summit on the future of
technology.</p>
          </a></div>
        </div>
        <div class="col-md-4 mb-4">
          <div class="feature">
            <img src="{% static 'images/conference2.png' %}" alt="Featured Event 2" class="feature-image">
            <h3 class="feature-title">UX India</h3>
            <p class="feature-description">Learn from experts in our engaging and informative webinar
series.</p>
          </div>
        </div>
        <div class="col-md-4 mb-4">
          <div class="feature">
            <img src="{% static 'images/conference3.jfif' %}" alt="Featured Event 3" class="feature-image">
            <h3 class="feature-title">Nullcon Goa</h3>
            <p class="feature-description">Celebrate the season with an unforgettable beachside party.</p>
          </div>
        </div>
      </div>
    </div>
  </section>
  <section id="categories" class="py-5">
    <div class="container">
      <h2 class="text-center mb-4">Explore Categories</h2>
      <div class="row">
        <div class="col-md-4 mb-4">
      <div class="card h-100">
        <a href="{% url 'eventapp:listwebinars' %}">
        <img src="{% static 'images/webinar.jfif' %}" class="card-img-top" alt="Webinar">
        <div class="card-body d-flex flex-column">
          <h5 class="card-title">Webinar</h5>
        </div>
       </a>
      </div>
    </div>
    <div class="col-md-4 mb-4">
      <div class="card h-100">
       <a href="{% url 'eventapp:conference' %}">
        <img src="{% static 'images/conference.jpg' %}" class="card-img-top" alt="Conference">
        <div class="card-body d-flex flex-column">
          <h5 class="card-title">Conference</h5>
        </div>
       </a>
      </div>
    </div>
    <div class="col-md-4 mb-4">
      <div class="card h-100">
        <img src="{% static 'images/social.webp' %}" class="card-img-top" alt="Social">
        <div class="card-body d-flex flex-column">
          <h5 class="card-title">Social Events</h5>
        </div>
      </div>
    </div>
  </div>

  </div>
 </section>
 <section id="introduction" class="introduction-section" style="background:linear-gradient(to left top,
#DD2476 10%, #FF512F 90%);">
  <div class="container">
   <p>Experience seamless event planning with our comprehensive range of services. From conference
management to social events, we've got you covered.
  Our team of dedicated event professionals is here to help you create unforgettable moments that leave a
lasting impression.
  Ready to make your event a success? Book your desired service now and let us bring your vision to
life!</p>
  <button id="book-service-btn" class="btn">Book a Service</button>
  </div>
 </section>
 <section id="services" class="our-services-section">
  <div class="container">
   <h2 class="text-center mb-4">Our Services</h2>

   <div class="service-card-container">
    <div class="service-card">
     <img src="{% static 'images/service1.jpg' %}" alt="Service 1" class="service-image">
     <h3 class="service-title">Catering Service</h3>
     <p class="service-description">Plan and execute successful conferences with precision and
expertise.</p>
    </div>
```

```
        <div class="service-card">
          <img src="{% static 'images/service2.jpg' %}" alt="Service 2" class="service-image">
          <h3 class="service-title">Light and Sound</h3>
          <p class="service-description">Create memorable social gatherings that leave a lasting
impression.</p>
        </div>
        <div class="service-card">
          <img src="{% static 'images/service3.jpg' %}" alt="Service 3" class="service-image">
          <h3 class="service-title">Venue Booking</h3>
          <p class="service-description">Host engaging webinars that connect and educate your audience.</p>
        </div>
      </div>
    </div>
  </section>
  <section class="contact-section" id="contact">
    <div class="contact-details">
        <h2 class="contact-heading">Contact Us</h2>
        <p class="contact-info">Have questions or need assistance? Feel free to get in touch with us.</p>
        <div class="contact-form">
            <form action="submit_form.php" method="post">
                <div class="form-group">
                    <label class="form-label" for="name">Your Name:</label>
                    <input class="form-input" type="text" id="name" name="name" required>
                </div>
                <div class="form-group">
                    <label class="form-label" for="email">Your Email:</label>
                    <input class="form-input" type="email" id="email" name="email" required>
                </div>
                <div class="form-group">
                    <label class="form-label" for="message">Message:</label>
                    <textarea class="form-message" id="message" name="message" rows="4" required></textarea>
                </div>
                <button class="form-button" type="submit">Send Message</button>
            </form>
        </div>
        <div class="social-media">
            <a class="social-link" href="https://www.facebook.com/yourpage" target="_blank"><i class="fab fa-
facebook"></i></a>
            <a class="social-link" href="https://www.twitter.com/yourpage" target="_blank"><i class="fab fa-
twitter"></i></a>
            <a class="social-link" href="https://www.instagram.com/yourpage" target="_blank"><i class="fab
fa-instagram"></i></a>
            <a class="social-link" href="https://www.linkedin.com/company/yourpage" target="_blank"><i
class="fab fa-linkedin"></i></a>
        </div>
    </div>
</section>
  <footer class="bg-dark text-white text-center py-3">
    <div class="container">
      &copy; 2023 EventoPlanneur. All rights reserved.
    </div>
  </footer>
</body>
</html>
{% endif %}
```

Register_webinar.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Host Webinar</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet"type="text/css" href="{% static 'css/webinar.css' %}">
  <link rel="icon" href="{% static 'images/eventoplanneur.png' %}" type="image/x-icon">
</head>
<body>
  <nav class="navbar navbar-expand-md bg-dark navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="{% url 'eventapp:orghome' %}">
        <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
      </a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarCollapse">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item"><a class="nav-link" href="#services">Services</a></li>
          <li class="nav-item"><a class="nav-link" href="#contact">Contact</a></li>
          <li class="nav-item"><a class="nav-link" href="#clients">Clients</a></li>
          <li class="nav-item"><a class="nav-link" href="#gallery">Gallery</a></li>
          <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:org_profile' %}">Profile</a></li>
          <li class="nav-item"><a class="nav-link" href="#notifications">Notifications</a></li>
          <li>
                <a href="{% url 'eventapp:logout' %}" class="nav-link">Logout</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
  <div class="container">
    <div class="events">
        <h4>Host a Webinar</h4>
        {% if messages %}
    <ul class="messages">
        {% for message in messages %}
            <li {% if message.tags %} class="{{ message.tags }}"{% endif %}>{{ message }}</li>
        {% endfor %}
    </ul>
{% endif %}
        <form method="post" id="webinar" action="{% url 'eventapp:register_webinar' %}">
          {% csrf_token %}
          <table>
          <tr>
            <td>
              <div class="mb-3">
              <label for="title" class="form-label">Title<span>*</span>:</label>
              <input type="text" id="title" name="title" class="form-control" required>
              <span id="titleError" class="error-text"></span>
          </div>
        </td>
        <td>
          <div class="mb-3">
              <label for="fee" class="form-label">Fee<span>*</span>:</label>
              <input type="number" id="fee" name="fee" class="form-control" required>
              <span id="feeError" class="error-text"></span>
          </div></td></tr><tr>
        <td>
          <div class="mb-3">
            <label for="start_time" class="form-label">Start Time<span>*</span>:</label>
            <input type="time" id="start_time" name="start_time" class="form-control">
          </div></td>
          <td>
            <div class="mb-3">
              <label for="end_time" class="form-label">End Time<span>*</span>:</label>
              <input type="time" id="end_time" name="end_time" class="form-control">
              <span id="timeError" class="error-text"></span>
            </div></td>

        </tr><tr><td>
          <div class="mb-3">
              <label for="date" class="form-label">Date<span>*</span>:</label>
              <input type="date" id="date" name="date" class="form-control" required>
              <span id="event_dateError" class="error-text"></span>
          </div></td><td>
            <div class="mb-3">
                <label for="deadline" class="form-label">Deadline<span>*</span>:</label>
                <input type="date" id="deadline" name="deadline" class="form-control" required>
              <span id="deadlineError" class="error-text"></span>
            </div></td></tr><tr><td>
            <div class="mb-3">
                <label for="event_type"  class="form-label">Event Type<span>*</span>:</label>
                <select id="event_type" name="event_type" class="form-control">
                <option value="Online">Online</option>
                <option value="Offline">Offline</option>

                </select>
                <span id="event_typeError" class="error-text"></span>
            </div></td><td>
              <div class="mb-3" id="url_div" >
                <label for="livestream" class="form-label">Livestream Link:</label>
                <input type="text" id="livestream" name="livestream" class="form-control" default="None" >
                <span id="urlError" class="error-text"></span>
            </div>
          <div class="mb-3" id="loc" style="display:none">
              <label for="location" class="form-label" >Location:</label>
              <input type="text" id="location" name="location" class="form-control" default="None">
              <span id="locationError" class="error-text"></span>
          </div>
            </td>
          </tr><tr><td>
          <div class="mb-3">
              <label for="organizer_name" class="form-label">Organizer Name<span>*</span>:</label>
              <input type="text" id="organizer_name" name="organizer_name" class="form-control" required>
              <span id="organizer_nameError" class="error-text"></span>
          </div></td><td>
          <div class="mb-3">
              <label for="phone_number" class="form-label">Contact number<span>*</span>:</label>
              <input type="text" id="phone_number" name="phone_number" class="form-control" required>
              <span id="phone_numberError" class="error-text"></span>
          </div></td></tr><tr><td>
            <div class="mb-3">
                <label for="poster" class="form-label">Poster( Image URL )<span>*</span>:</label>
                <input type="text" id="poster" name="poster" class="form-control" required>
                <span id="posterError" class="error-text"></span>
              </div></td>
              <td>
                <div class="mb-3">
                    <label for="max_participants" class="form-label">Maximum Participants<span>*</span>:</label>
                    <input type="number" id="max_participants" name="max_participants" class="form-control" required>
                    <span id="maxError" class="error-text"></span>
                </div></td></tr>
              <tr><td><div class="form-group">
                    <label for="speakers">Speakers:</label>
                    <div id="speakers-container">
                      <div class="speaker" style="display:flex;">
                        <select name="speakers_designation[]" class="form-control" style="width:100px;">
                          <option value="Mr">Mr</option>
                          <option value="Ms">Ms</option>
                          <option value="Mrs">Mrs</option>
                          <option value="Dr">Dr</option>
                        </select>
                        <input type="text" name="speakers_name[]" id="speaker_name" class="form-control"
style="width:250px;"required>
                </div>
              </div>
              <span id="speakerError" class="error-text"></span>
            <button type="button" id="add-speaker" class="btn btn-secondary">+</button>
        </div></td>
          <td>
            <div class="mb-3">
                <label for="description" class="form-label">Description<span>*</span>:</label>
                <textarea id="description" name="description" class="form-control" rows="4" cols="50"
required></textarea>
            <span id="descriptionError" class="error-text"></span>
            <span class="error-text">Please provide a detailed description of your webinar, including the agenda, learning
objectives, and target audience.
              Use this space to introduce your webinar, describe its key topics, and explain why attendees should
join.</span>
            </div></td>
        </tr><tr><td>
          <span>* Required fields</span><br>
      <button  type="submit"><a class="btn mr-2">Register</a></button></td><td>
        <button><a href="{% url 'eventapp:webinar' %}" class="btn mr-2"> Back </a></button>
      {% comment %} <button id="sendButton"><a style="text-decoration:none;color:white;" href="{% url
'eventapp:webinar' %}">Back</a></button></td></tr> {% endcomment %}
    </form>
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <script>
      $(document).ready(function() {
          var event_type = document.getElementById("event_type");
          var loc = document.getElementById("loc");
          var url_div = document.getElementById("url_div");
          var livestream=document.getElementById("livestream");
          if (event_type.value === 'Online') {
            loc.style.display = 'none';
            url_div.style.display = 'block';
        } else {
            loc.style.display = 'block';
            url_div.style.display = 'none';
            livestream.innerText='https://www.xyz.com';
        }
          handleLocationChange(event_type,loc,url_div);
      });
      function handleLocationChange(event_type,loc,url_div) {
        if (event_type.value === 'Online') {
            loc.style.display = 'none';
            url_div.style.display = 'block';

        } else {
            loc.style.display = 'block';
            url_div.style.display = 'none';
        }
    }
    // JavaScript to dynamically add more speaker fields
    document.getElementById('add-speaker').addEventListener('click', function() {
        const speakersContainer = document.getElementById('speakers-container');
        const newSpeaker = document.createElement('div');
        newSpeaker.classList.add('speaker');
        newSpeaker.innerHTML = `
        <div class="speaker" style="display:flex;">
          <select name="speakers_designation[]" class="form-control" style="width:100px;">
            <option value="Mr">Mr</option>
            <option value="Ms">Ms</option>
            <option value="Mrs">Mrs</option>
            <option value="Dr">Dr</option>
          </select>
          <input type="text" name="speakers_name[]" id="speaker_name" class="form-control"
style="width:250px;"required>
        </div>
        `;
        speakersContainer.appendChild(newSpeaker);
    });
</script>
<script src="{% static 'js/webinar.js' %}"></script>
</div>
</div>
</body>
</html>
```

View_webinar.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>View Webinar</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet"type="text/css" href="{% static 'css/webinar.css' %}">
  <link rel="icon" href="{% static 'images/eventoplanneur.png' %}" type="image/x-icon">
</head>
<body>
  <nav class="navbar navbar-expand-md bg-dark navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="{% url 'eventapp:orghome' %}">
        <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
      </a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarCollapse">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item"><a class="nav-link" href="#services">Services</a></li>
          <li class="nav-item"><a class="nav-link" href="#contact">Contact</a></li>
          <li class="nav-item"><a class="nav-link" href="#clients">Clients</a></li>
          <li class="nav-item"><a class="nav-link" href="#gallery">Gallery</a></li>
          <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
          <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:org_profile' %}">Profile</a></li>
          <li class="nav-item"><a class="nav-link" href="#notifications">Notifications</a></li>
          <li>
                <a href="{% url 'eventapp:logout' %}" class="nav-link">Logout</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
  <div class="container">
  <div class="events">
    <form method="POST" id="webinar">
        {% csrf_token %}
        <table>
          <tr>
            <td>
              <div class="mb-3">
              <label for="title" class="form-label">Title:</label>
              <input type="text" id="title" name="title" class="form-control" value="{{form.title.value}}" required>
              <span id="titleError" class="error-text"></span>
          </div>
        </td>
        <td>
          <div class="mb-3">
              <label for="fee" class="form-label">Fee:</label>
              <input type="number" id="fee" name="fee" class="form-control" value="{{form.fee.value}}" required>
              <span id="feeError" class="error-text"></span>
          </div></td></tr><tr>
        <td>

          <div class="mb-3">
            <label for="start_time" class="form-label">Start Time:</label>
            <input type="time" id="start_time" name="start_time" class="form-control"
value="{{form.start_time.value|date:'H:i' }}">

          </div></td>
          <td>
            <div class="mb-3">
              <label for="end_time" class="form-label">End Time:</label>
              <input type="time" id="end_time" name="end_time" class="form-control"
value="{{form.end_time.value|date:'H:i' }}">
              <span id="timeError" class="error-text"></span>
            </div></td>
        </tr><tr><td>
          <div class="mb-3">
              <label for="date" class="form-label">Date:</label>
              <input type="date" id="date" name="date" class="form-control" value="{{ form.date.value|date:'Y-m-d' }}"
required>
              <span id="event_dateError" class="error-text"></span>
          </div></td><td>
            <div class="mb-3">
                <label for="deadline" class="form-label">Deadline:</label>
                <input type="date" id="deadline" name="deadline" class="form-control"
value="{{form.deadline.value|date:'Y-m-d'}}" required>
              <span id="deadlineError" class="error-text"></span>
            </div></td></tr><tr><td>
            <div class="mb-3">
                <label for="event_type"  class="form-label">Event Type:</label>
                {% comment %} <select id="event_type" name="event_type" value="{{form.event_type.value}}"
```

```
class="form-control">
                <option value="Offline">Offline</option>
                <option value="Online">Online</option>
              </select> {% endcomment %}
              <input type="text" id="event_type" name="event_type" class="form-control"
value="{{form.event_type.value}}" required>
              <span id="event_typeError" class="error-text"></span>
          </div></td><td>
        <div class="mb-3" id="loc" >
            <label for="location" class="form-label">Location:</label>
            <input type="text" id="location" name="location" value="{{form.location.value}}" class="form-control">
            <span id="locationError" class="error-text"></span>
        </div>
       <div class="mb-3" id="url_div">
              <label for="livestream" class="form-label">Livestream Link:</label>
              <input type="text" id="livestream" name="livestream" value="{{form.livestream.value}}"class="form-
control">
              <span id="urlError" class="error-text"></span>
          </div></td>
        </tr><tr><td>
        <div class="mb-3">
            <label for="organizer_name" class="form-label">Organizer Name:</label>
            <input type="text" id="organizer_name" name="organizer_name" class="form-control"
value="{{form.title.value}}" required>
            <span id="organizer_nameError" class="error-text"></span>
        </div></td><td>
        <div class="mb-3">
            <label for="phone_number" class="form-label">Contact number:</label>
            <input type="text" id="phone_number" name="phone_number" value="{{form.phone_number.value}}"
class="form-control" required>
            <span id="contactError" class="error-text"></span>
        </div></td></tr><tr><td>
          <div class="mb-3">
              <label for="poster" class="form-label">Poster( Image URL ):</label>
              <input type="text" id="poster" name="poster" class="form-control" value="{{form.poster.value}}" required>
            <span id="posterError" class="error-text"></span>
          </div></td></tr>
          <tr><td>
              <div class="form-group">
                <label for="speaker_name">Speakers:</label>
                <div id="">
                  {% for speaker in speakers %}
                      <div class="speaker" style="display:flex;">
                          <input type="text" style="width:60px;" id="designation_{{ speaker.id }}" name="speakers-{{
speaker.id }}-designation" class="form-control" value="{{ speaker.designation }}" required>
                          <input type="text" id="speaker_name_{{ speaker.id }}" name="speakers-{{ speaker.id }}-
speaker_name" class="form-control" value="{{ speaker.speaker_name }}" required>
                      </div>
                  {% endfor %}
                </div>
            </div>
            <td>
              <div class="mb-3">
                  <label for="description" class="form-label">Description:</label>
                  <textarea id="description" name="description" class="form-control" rows="4" cols="50"
>{{form.description.value}}</textarea>
              <span id="descriptionError" class="error-text"></span>
              </div></td>
          </tr><tr><td></table>
    </form>
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <script>
      $(document).ready(function() {
        var event_type = document.getElementById("event_type");
        var loc = document.getElementById("loc");
        var url_div = document.getElementById("url_div");
        if (event_type.value === 'Online') {
          loc.style.display = 'none';
          url_div.style.display = 'block';
      } else {
          loc.style.display = 'block';
          url_div.style.display = 'none';
      }
        handleLocationChange(event_type,loc,url_div);
      });
      function handleLocationChange(event_type,loc,url_div) {
      if (event_type.value === 'Online') {
          loc.style.display = 'none';
          url_div.style.display = 'block';
      } else {
          loc.style.display = 'block';
          url_div.style.display = 'none';
      }
    }
  </script>
  </div>
</div>
<footer class="bg-dark text-white text-center py-3">
    <div class="container">
      &copy; 2023 EventoPlanneur. All rights reserved.
    </div>
  </footer>
</body>
</html>
```

Update_webinar.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Update Webinar</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet"type="text/css" href="{% static 'css/webinar.css' %}">
  <link rel="icon" href="{% static 'images/eventoplanneur.png' %}" type="image/x-icon">
</head>
<body>
  <nav class="navbar navbar-expand-md bg-dark navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="{% url 'eventapp:orghome' %}">
        <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
      </a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarCollapse">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item"><a class="nav-link" href="#services">Services</a></li>
          <li class="nav-item"><a class="nav-link" href="#contact">Contact</a></li>
          <li class="nav-item"><a class="nav-link" href="#clients">Clients</a></li>
          <li class="nav-item"><a class="nav-link" href="#gallery">Gallery</a></li>
          <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
          <li class="nav-item"><a class="nav-link" href="#profile">Profile</a></li>
          <li class="nav-item"><a class="nav-link" href="#notifications">Notifications</a></li>
          <li>
                <a href="{% url 'eventapp:logout' %}" class="nav-link">Logout</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
  <div class="container">
  <div class="events">
    <h4>Update Webinar</h4>
    <form method="POST" id="webinar" >
      {% csrf_token %}
      <table>
        <tr>
          <td>
            <div class="mb-3">
            <label for="title" class="form-label">Title:</label>
            <input type="text" id="title" name="title" class="form-control" value="{{form.title.value}}"
required>
            <span id="titleError" class="error-text"></span>
        </div>
      </td>
      <td>
        <div class="mb-3">
            <label for="fee" class="form-label">Fee:</label>
            <input type="number" id="fee" name="fee" class="form-control" value="{{form.fee.value}}"
required>
            <span id="feeError" class="error-text"></span>
        </div></td></tr><tr>
          <td>

            <div class="mb-3">
              <label for="start_time" class="form-label">Start Time:</label>
              <input type="time" id="start_time" name="start_time" class="form-control"
value="{{form.start_time.value|date:'H:i' }}">

            </div></td>
            <td>
              <div class="mb-3">
                <label for="end_time" class="form-label">End Time:</label>
                <input type="time" id="end_time" name="end_time" class="form-control"
value="{{form.end_time.value|date:'H:i' }}">
                <span id="timeError" class="error-text"></span>
              </div></td>
          </tr><tr><td>
        <div class="mb-3">
            <label for="date" class="form-label">Date:</label>
            <input type="date" id="date" name="date" class="form-control" value="{{ form.date.value|date:'Y-
m-d' }}" required>
```

```
      <span id="event_dateError" class="error-text"></span>
    </div></td><td>
     <div class="mb-3">
       <label for="deadline" class="form-label">Deadline:</label>
       <input type="date" id="deadline" name="deadline" class="form-control"
value="{{form.deadline.value|date:'Y-m-d'}}" required>
      <span id="deadlineError" class="error-text"></span>
     </div></td></tr><tr><td>
     <div class="mb-3">
       <label for="event_type"  class="form-label">Event Type:</label>
       <input type="text" id="event_type" name="event_type" class="form-control"
value="{{form.event_type.value}}" required>
       <span id="event_typeError" class="error-text"></span>
     </div></td><td>
    <div class="mb-3" id="loc">
      <label for="location" class="form-label">Location:</label>
      <input type="text" id="location" name="location" value="{{form.location.value}}" class="form-
control" >
      <span id="locationError" class="error-text"></span>
    </div>
    <div class="mb-3" id="url_div">
        <label for="livestream" class="form-label">Livestream Link:</label>
        <input type="text" id="livestream" name="livestream"
value="{{form.livestream.value}}"class="form-control">
        <span id="urlError" class="error-text"></span>
      </div></td>
    </tr><tr><td>
    <div class="mb-3">
      <label for="organizer_name" class="form-label">Organizer Name:</label>
      <input type="text" id="organizer_name" name="organizer_name" class="form-control"
value="{{form.organizer_name.value}}" required>
      <span id="organizer_nameError" class="error-text"></span>
    </div></td><td>
    <div class="mb-3">
      <label for="phone_number" class="form-label">Contact number:</label>
      <input type="text" id="phone_number" name="phone_number"
value="{{form.phone_number.value}}" class="form-control" required>
      <span id="contactError" class="error-text"></span>
    </div></td></tr><tr><td>
     <div class="mb-3">
       <label for="poster" class="form-label">Poster( Image URL ):</label>
       <input type="text" id="poster" name="poster" class="form-control" value="{{form.poster.value}}"
required>
      <span id="posterError" class="error-text"></span>
     </div></td><td>
      <div class="mb-3">
        <label for="max_participants" class="form-label">Maximum Participants:</label>
        <input type="number" id="max_participants" name="max_participants" class="form-control"
value="{{form.max_participants.value}}"required>
        <span id="maxError" class="error-text"></span>
      </div></td></tr>
     <tr><td>
      <div class="form-group">
       <label for="speakers">Speakers:</label>
       <div id="speakers-container">
```

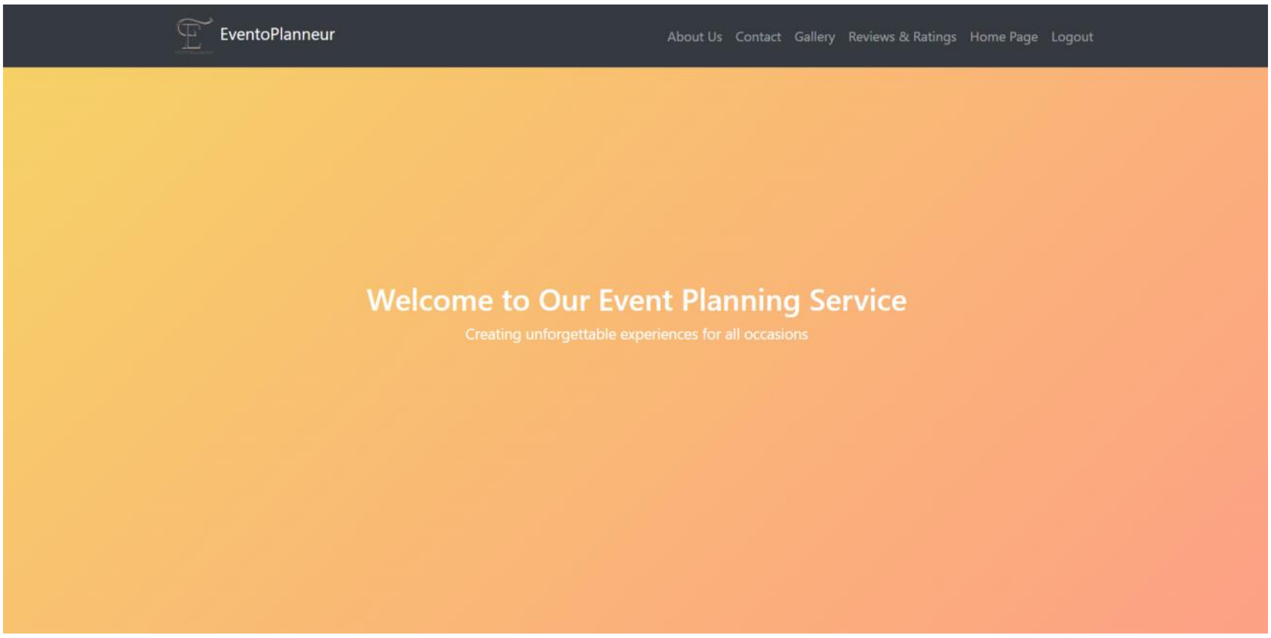
```
               {% for speaker in speakers %}
                 <div class="speaker" style="display: flex;">
                   <input type="text" style="width: 60px;" id="designation_{{ speaker.id }}" name="speakers-{{
speaker.id }}-designation" class="form-control" value="{{ speaker.designation }}" required>
                   <input type="text" id="speaker_name_{{ speaker.id }}" name="speakers-{{ speaker.id }}-
speaker_name" class="form-control" value="{{ speaker.speaker_name }}" required>
                 </div>
               {% endfor %}
             </div>
             <button type="button" id="add-speaker" class="btn btn-secondary">+</button>
           </div></td>
           <td>
             <div class="mb-3">
                 <label for="description" class="form-label">Description:</label>
                 <textarea id="description" name="description" class="form-control" rows="4" cols="50"
>{{form.description.value}}</textarea>
             <span id="descriptionError" class="error-text"></span>
             </div></td>
         </tr><tr><td></table>
           <button id="sendButton" type="submit">Update</button></td><td>
           <button id="sendButton"><a style="text-decoration:none;color:white;" href="{% url
'eventapp:webinar' %}">Back</a></button></td></tr>
  </form>
  {% if messages %}
  <ul class="messages">
   {% for message in messages %}
   <li {% if message.tags %} class="{{ message.tags }}"{% endif %}>{{ message }}</li>
   {% endfor %}
   </ul>
  {% endif %}
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="{% static 'js/webinar.js' %}"></script>
 <script>
  $(document).ready(function() {
      var event_type = document.getElementById("event_type");
      var loc = document.getElementById("loc");
      var url_div = document.getElementById("url_div");
      var livestream=document.getElementById("livestream");
      if (event_type.value === 'Online') {
        loc.style.display = 'none';
        url_div.style.display = 'block';
    } else {
        loc.style.display = 'block';
        url_div.style.display = 'none';
        livestream.innerText="None";
    }
      handleLocationChange(event_type,loc,url_div);
  });
  function handleLocationChange(event_type,loc,url_div) {
    if (event_type.value === 'Online') {
        loc.style.display = 'none';
        url_div.style.display = 'block';

    } else {
        loc.style.display = 'block';
        url_div.style.display = 'none';
        livestream.innerText='https://www.xyz.com';
    }
}
</script>
  </div>
</div>

<footer class="bg-dark text-white text-center py-3">
    <div class="container">
      &copy; 2023 EventoPlanneur. All rights reserved.
    </div>
  </footer>
</body>
</html>
```

Payment.html

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>GFG</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet"type="text/css" href="{% static 'css/webinar.css' %}">
  <link rel="icon" href="{% static 'images/eventoplanneur.png' %}" type="image/x-icon">
</head>
<body>
              <nav class="navbar navbar-expand-md bg-dark navbar-dark">
        <div class="containers">
          <a class="navbar-brand" href="{% url 'eventapp:attendeehome' %}">
            <img src="{% static 'images/Eventoplanneur.png' %}" alt="Logo" width="50"> EventoPlanneur
          </a>
          <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse" id="navbarCollapse">
            <ul class="navbar-nav ml-auto">
              <li class="nav-item"><a class="nav-link" href="#gallery">Gallery</a></li>
              <li class="nav-item"><a class="nav-link" href="#reviews">Reviews & Ratings</a></li>
              <li class="nav-item"><a class="nav-link" href="{% url 'eventapp:org_profile' %}">Profile</a></li>
              <li class="nav-item"><a class="nav-link" href="#notifications">Notifications</a></li>
              <li class="nav-item">
                {% if user.is_authenticated %}
                <a class="nav-link" >User: {{ user.email }}</a>
                <li class="nav-item"><a href="{% url 'eventapp:logout' %}" class="nav-link">Logout</a></li>
                {% endif %}
              </li>
            </ul>
          </div>
        </div>
      </nav><br>
              <div class="card">
              <h1 class="title">WEBINAR REGISTRATION </h1>
              <small class="muted"
                          >Proceed to pay the registration fee</small
              >
              <div class="btn_container">
                          <!-- Payment Button -->
                          <button class="btn" id="pay-btn">Pay</button>
        <a href="{% url 'eventapp:events' %}" class="btn btn-primary">Go Back to Events</a>
              </div>
              </div>
</body>
<!-- Razorpay's Javascript code. -->
<script src="https://checkout.razorpay.com/v1/checkout.js"></script>
<script>
              var options = {
              // Enter the Key ID generated from the Dashboard
              key: "{{ razorpay_merchant_key }}",
              amount: "{{ razorpay_amount }}",
              currency: "{{ currency }}",
              name: "Dj Razorpay",
              order_id: "{{ razorpay_order_id }}",
              callback_url: "{{ callback_url }}",
              };
              var rzp1 = new Razorpay(options);
              document.getElementById("pay-btn").onclick = function (e) {
        rzp1.open();
        e.preventDefault();
        };
</script>
</html>
```

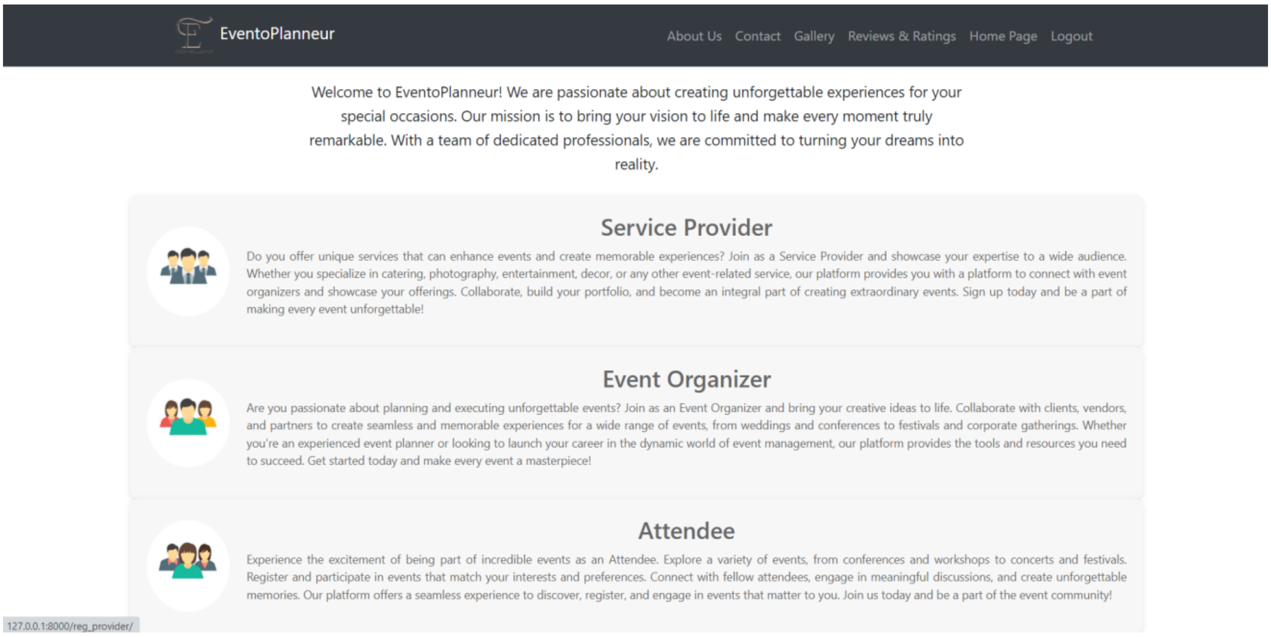
9.2 Screen Shots

Index page



Welcome to Our Event Planning Service

Creating unforgettable experiences for all occasions



Welcome to EventoPlanneur! We are passionate about creating unforgettable experiences for your special occasions. Our mission is to bring your vision to life and make every moment truly remarkable. With a team of dedicated professionals, we are committed to turning your dreams into reality.

Service Provider

Do you offer unique services that can enhance events and create memorable experiences? Join as a Service Provider and showcase your expertise to a wide audience. Whether you specialize in catering, photography, entertainment, decor, or any other event-related service, our platform provides you with a platform to connect with event organizers and showcase your offerings. Collaborate, build your portfolio, and become an integral part of creating extraordinary events. Sign up today and be a part of making every event unforgettable!

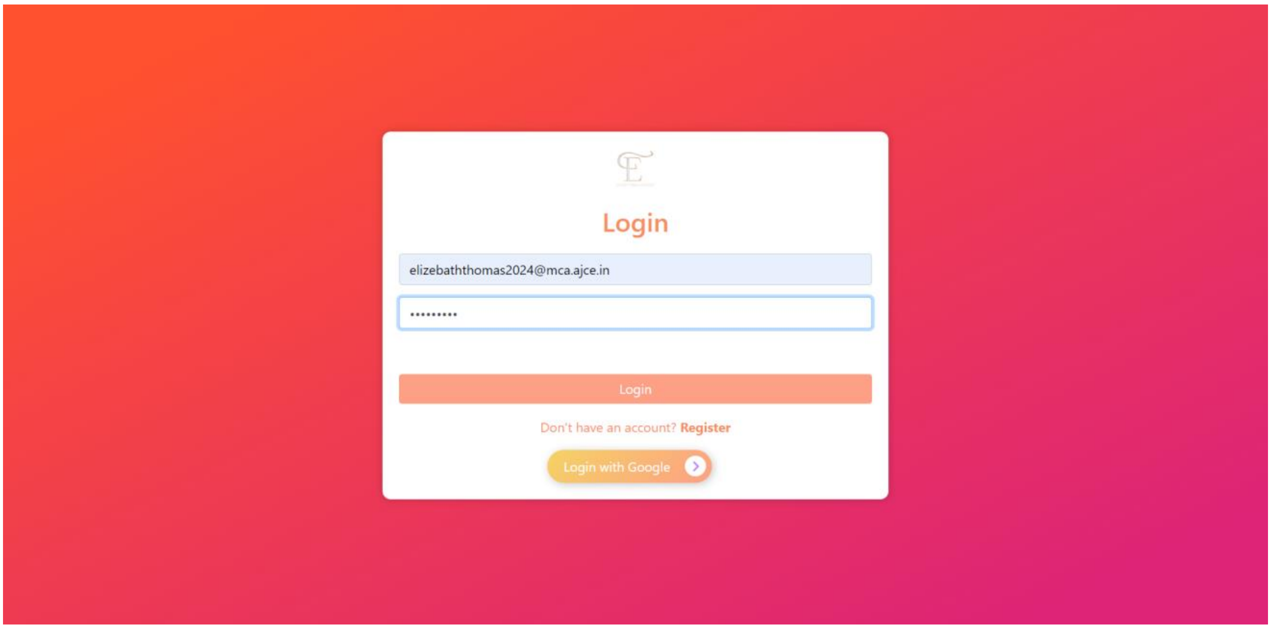
Event Organizer

Are you passionate about planning and executing unforgettable events? Join as an Event Organizer and bring your creative ideas to life. Collaborate with clients, vendors, and partners to create seamless and memorable experiences for a wide range of events, from weddings and conferences to festivals and corporate gatherings. Whether you're an experienced event planner or looking to launch your career in the dynamic world of event management, our platform provides the tools and resources you need to succeed. Get started today and make every event a masterpiece!

Attendee

Experience the excitement of being part of incredible events as an Attendee. Explore a variety of events, from conferences and workshops to concerts and festivals. Register and participate in events that match your interests and preferences. Connect with fellow attendees, engage in meaningful discussions, and create unforgettable memories. Our platform offers a seamless experience to discover, register, and engage in events that matter to you. Join us today and be a part of the event community!

Login page

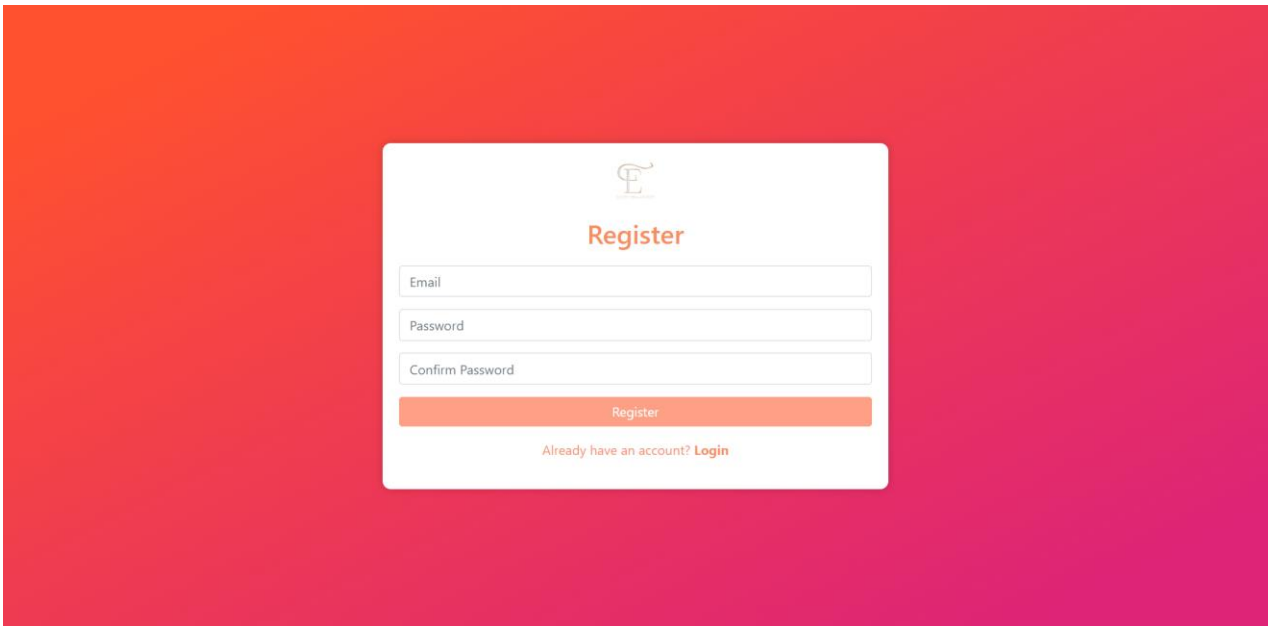


Login

elizebaththomas2024@mca.ajce.in

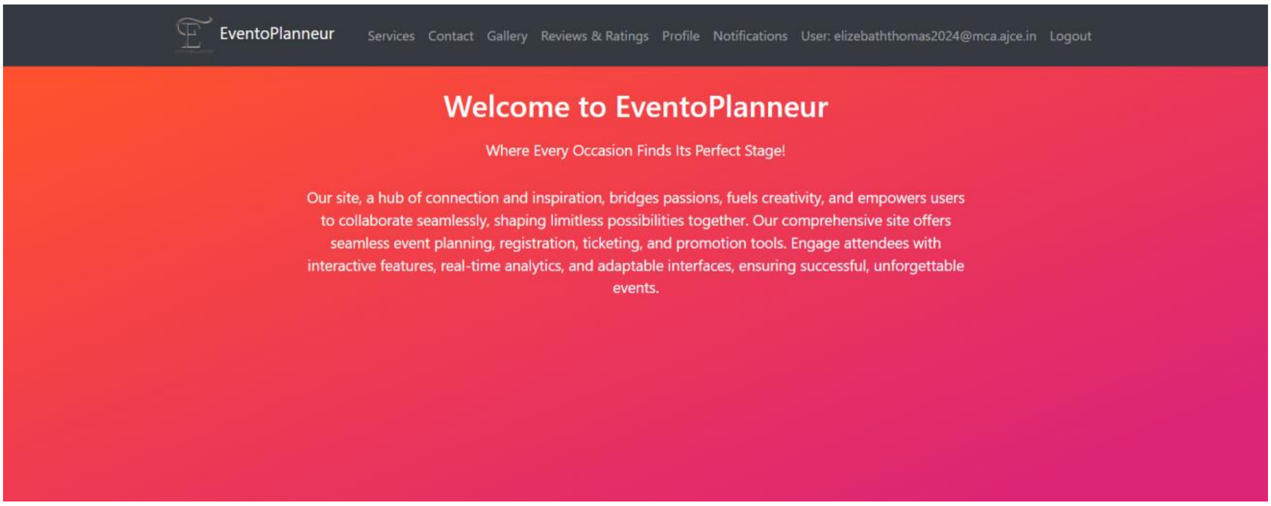
•••••••••

Register page



Register

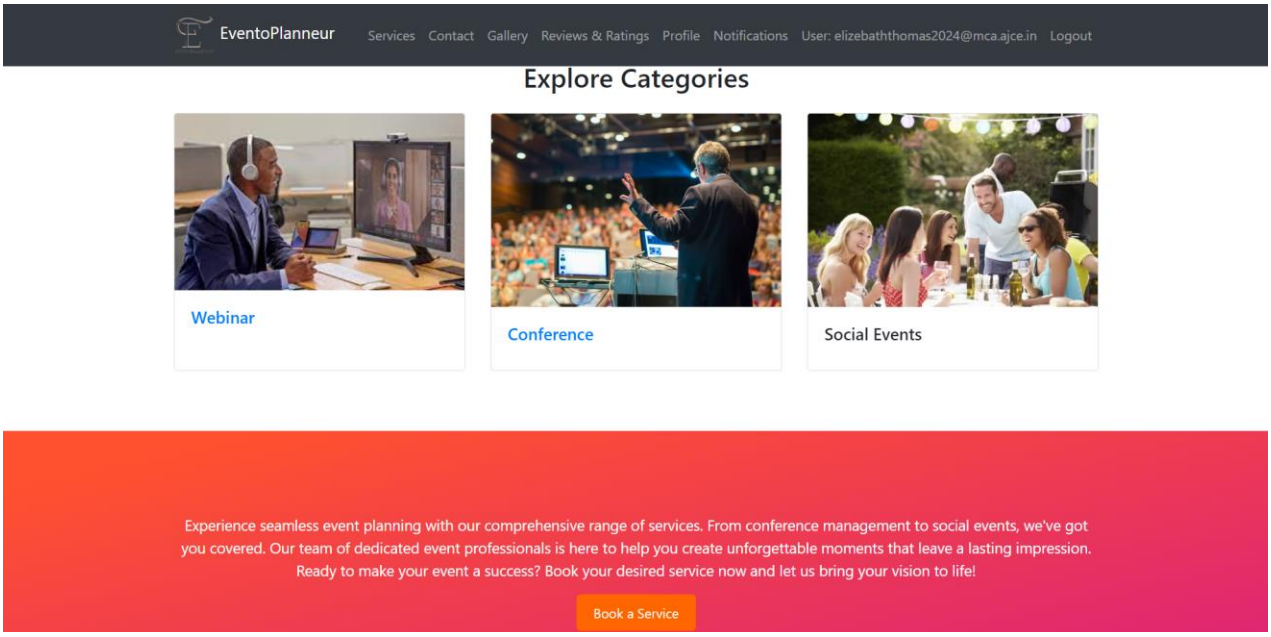
Event Organizer Homepage



Welcome to EventoPlanneur

Where Every Occasion Finds Its Perfect Stage!

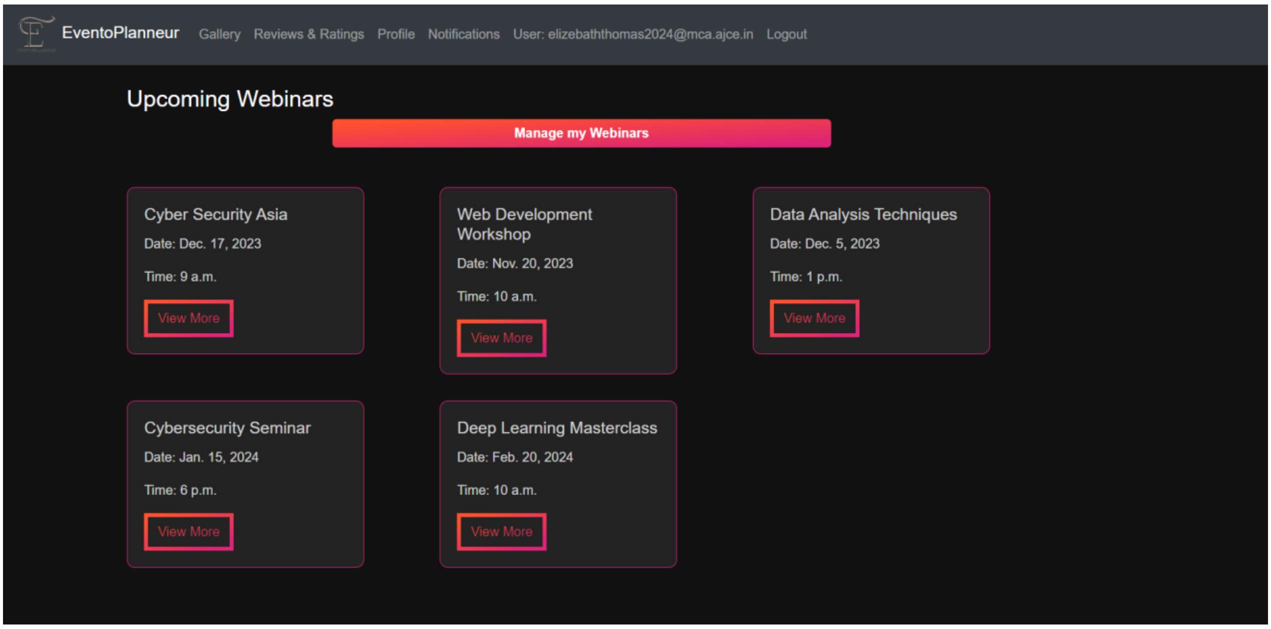
Our site, a hub of connection and inspiration, bridges passions, fuels creativity, and empowers users to collaborate seamlessly, shaping limitless possibilities together. Our comprehensive site offers seamless event planning, registration, ticketing, and promotion tools. Engage attendees with interactive features, real-time analytics, and adaptable interfaces, ensuring successful, unforgettable events.



Explore Categories

Experience seamless event planning with our comprehensive range of services. From conference management to social events, we've got you covered. Our team of dedicated event professionals is here to help you create unforgettable moments that leave a lasting impression. Ready to make your event a success? Book your desired service now and let us bring your vision to life!

Webinar page



Upcoming Webinars

Cyber Security Asia
Date: Dec. 17, 2023
Time: 9 a.m.

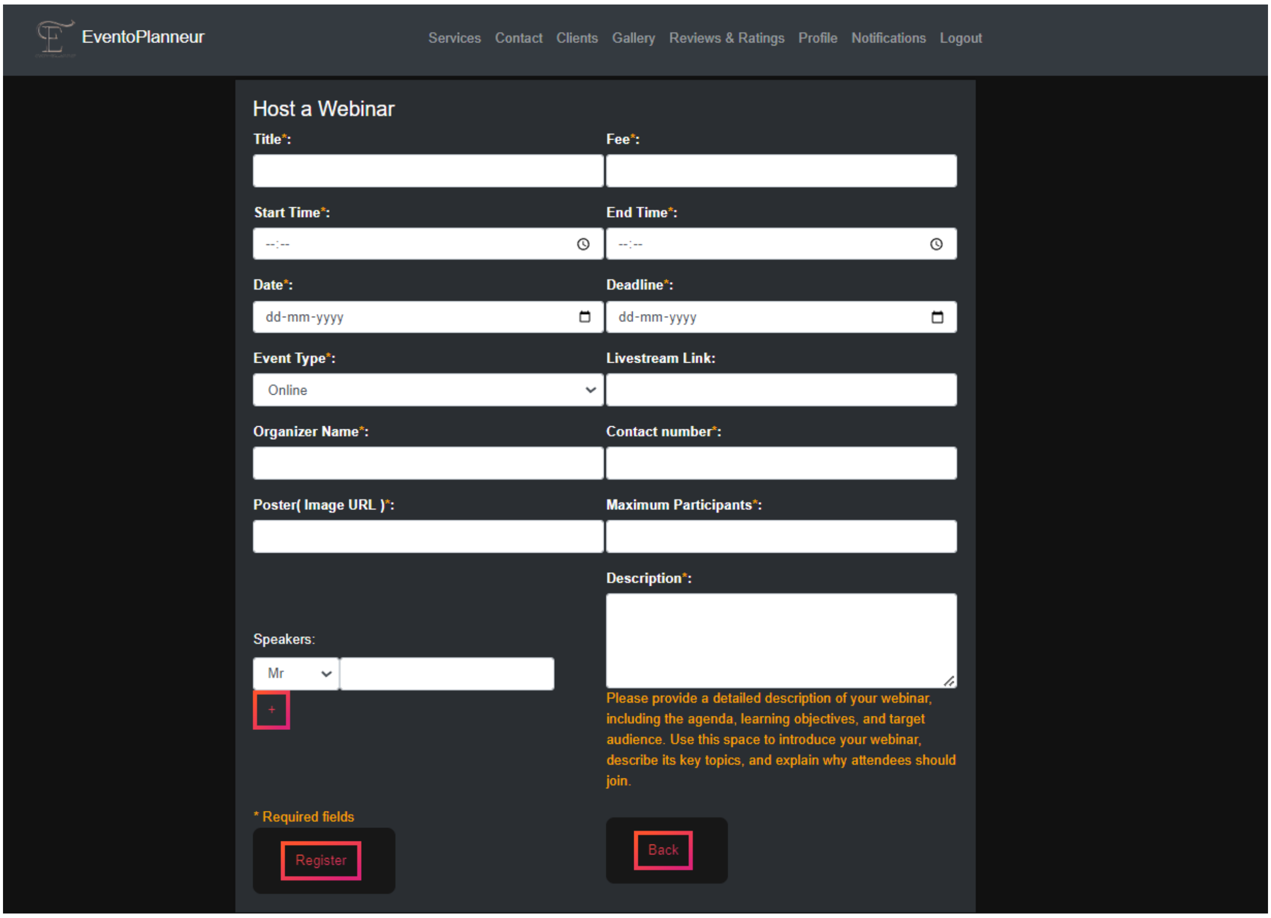
Web Development Workshop
Date: Nov. 20, 2023
Time: 10 a.m.

Data Analysis Techniques
Date: Dec. 5, 2023
Time: 1 p.m.

Cybersecurity Seminar
Date: Jan. 15, 2024
Time: 6 p.m.

Deep Learning Masterclass
Date: Feb. 20, 2024
Time: 10 a.m.

Webinar Register Page



Host a Webinar

Title*:

Fee*:

Start Time*: --:--

End Time*: --:--

Date*: dd-mm-yyyy

Deadline*: dd-mm-yyyy

Event Type*: Online

Livestream Link:

Organizer Name*:

Contact number*:

Poster(Image URL)*:

Maximum Participants*:

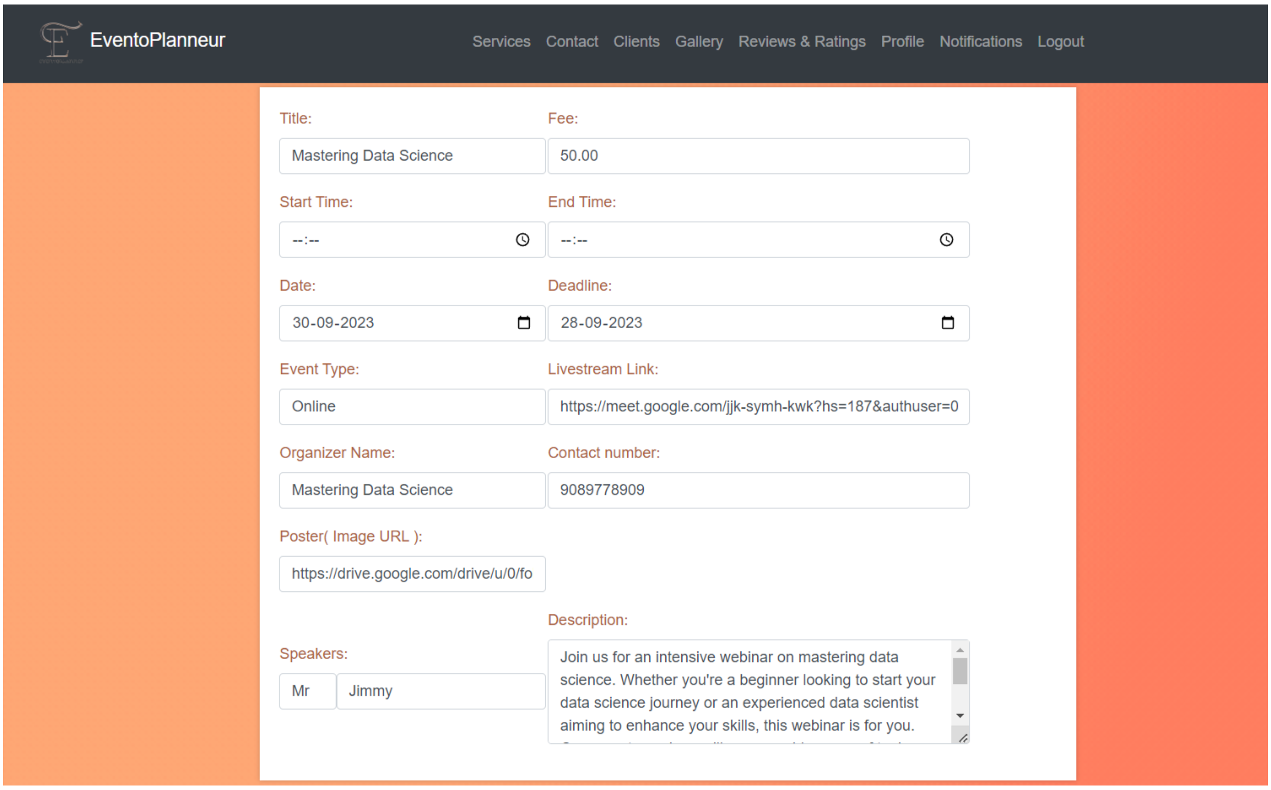
Description*:

Please provide a detailed description of your webinar, including the agenda, learning objectives, and target audience. Use this space to introduce your webinar, describe its key topics, and explain why attendees should join.

Speakers: Mr

* Required fields

Update Webinar Page



Title: Mastering Data Science

Fee: 50.00

Start Time: --:--

End Time: --:--

Date: 30-09-2023

Deadline: 28-09-2023

Event Type: Online

Livestream Link: https://meet.google.com/jjk-symh-kwk?hs=187&authuser=0

Organizer Name: Mastering Data Science

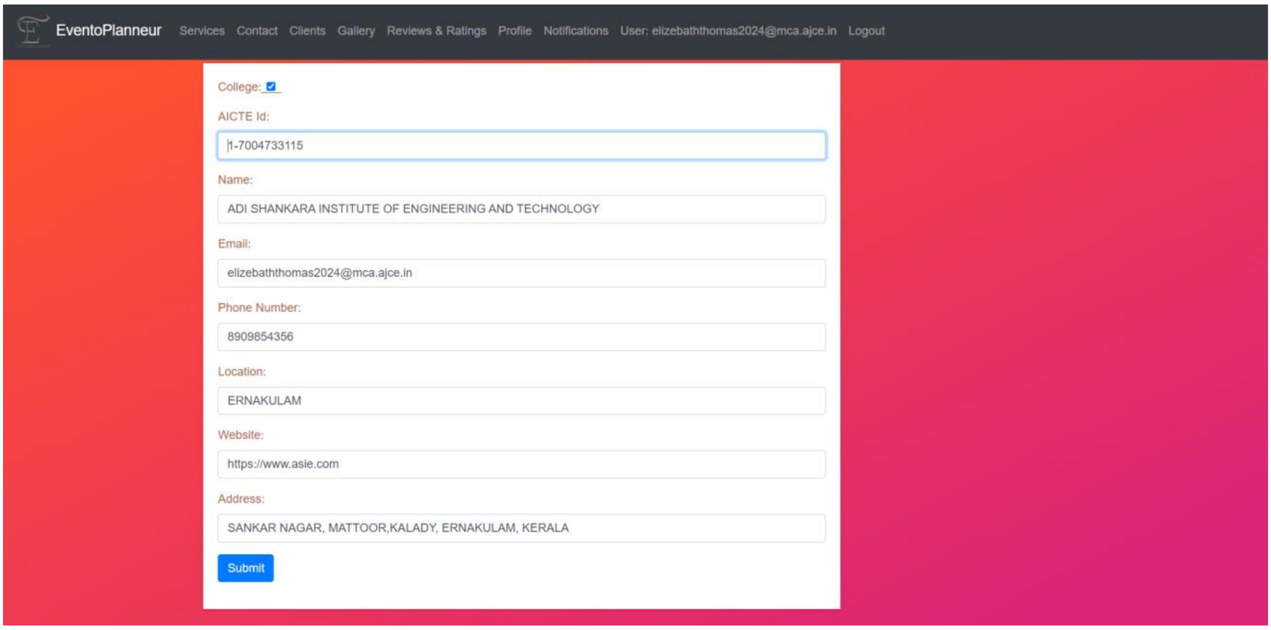
Contact number: 9089778909

Poster(Image URL): https://drive.google.com/drive/u/0/fo

Speakers: Mr Jimmy

Description: Join us for an intensive webinar on mastering data science. Whether you're a beginner looking to start your data science journey or an experienced data scientist aiming to enhance your skills, this webinar is for you.

Profile page



College: ☑

AICTE Id: 1-7004733115

Name: ADI SHANKARA INSTITUTE OF ENGINEERING AND TECHNOLOGY

Email: elizebaththomas2024@mca.ajce.in

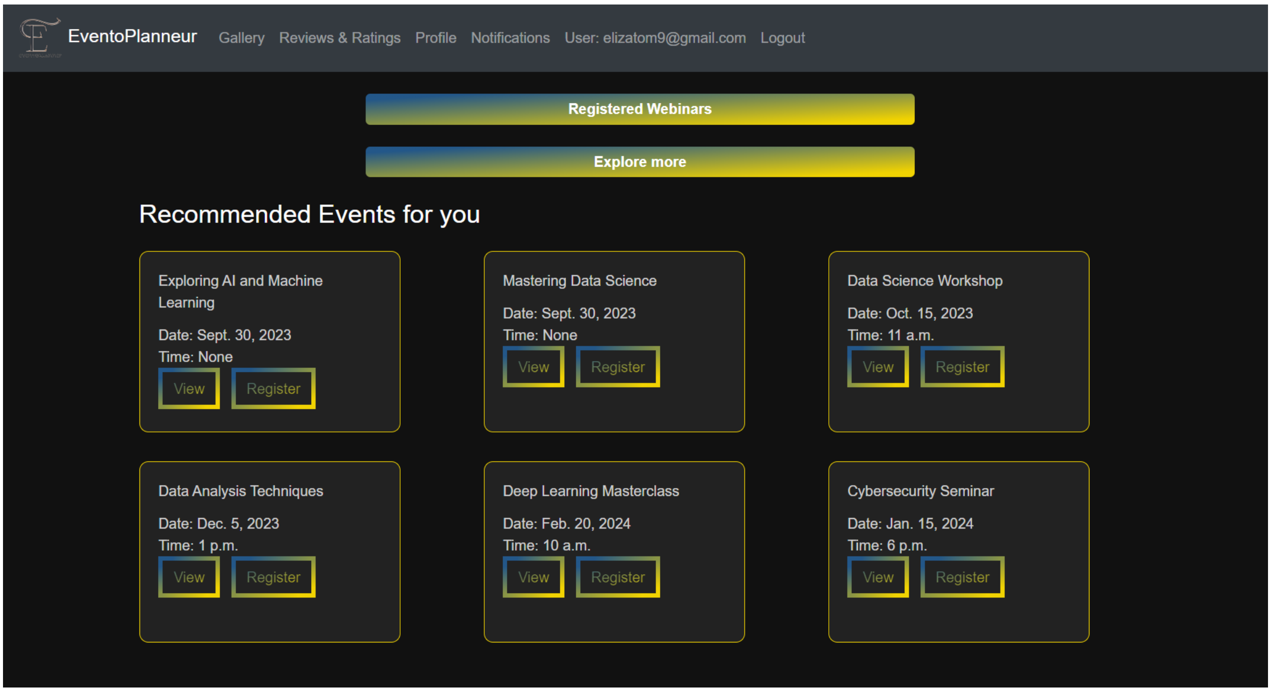
Phone Number: 8909854356

Location: ERNAKULAM

Website: https://www.asie.com

Address: SANKAR NAGAR, MATTOOR,KALADY, ERNAKULAM, KERALA

Recommendation Page



Recommended Events for you

Exploring AI and Machine Learning
Date: Sept. 30, 2023
Time: None
View Register

Mastering Data Science
Date: Sept. 30, 2023
Time: None
View Register

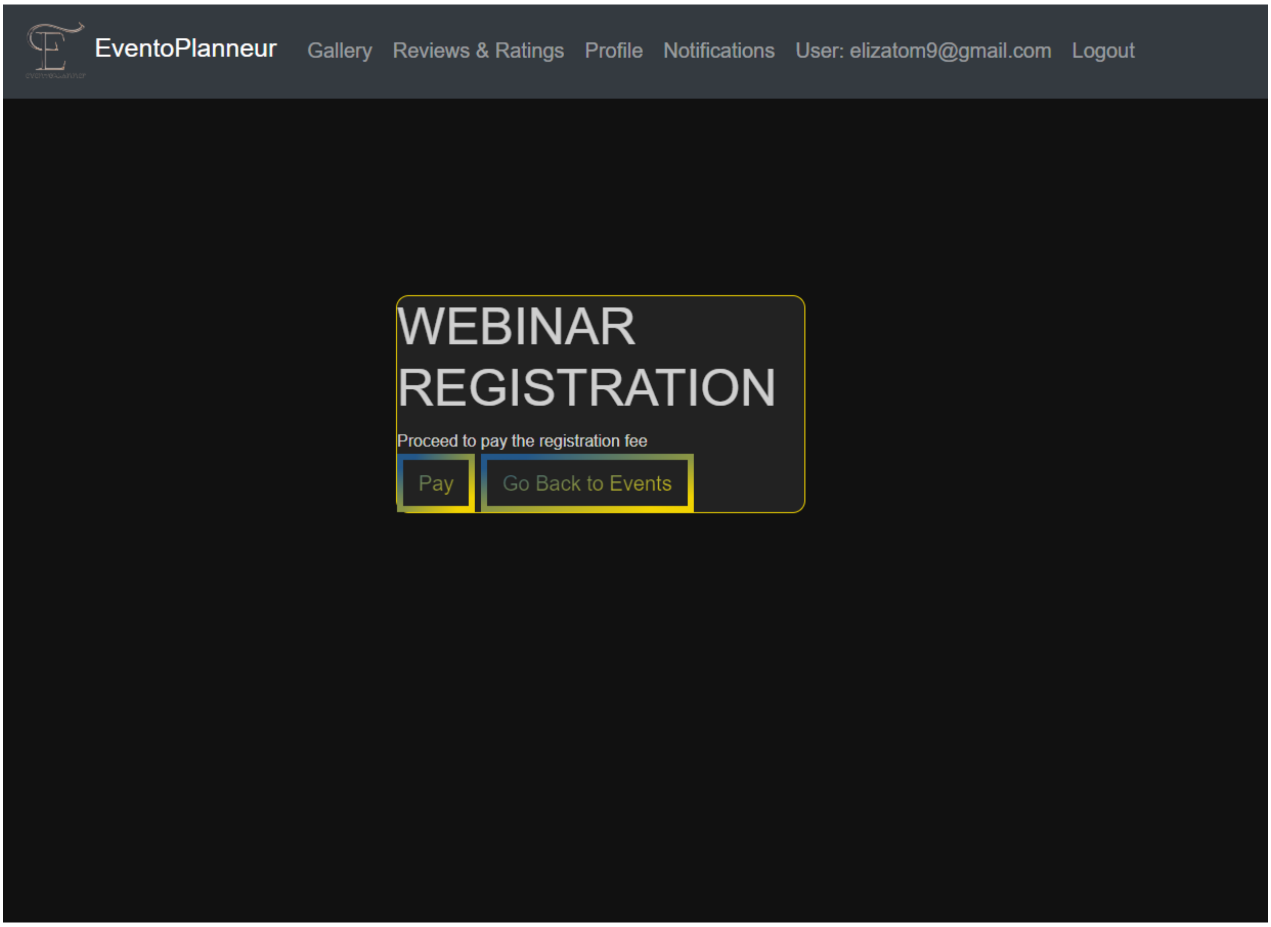
Data Science Workshop
Date: Oct. 15, 2023
Time: 11 a.m.
View Register

Data Analysis Techniques
Date: Dec. 5, 2023
Time: 1 p.m.
View Register

Deep Learning Masterclass
Date: Feb. 20, 2024
Time: 10 a.m.
View Register

Cybersecurity Seminar
Date: Jan. 15, 2024
Time: 6 p.m.
View Register

Payment Page

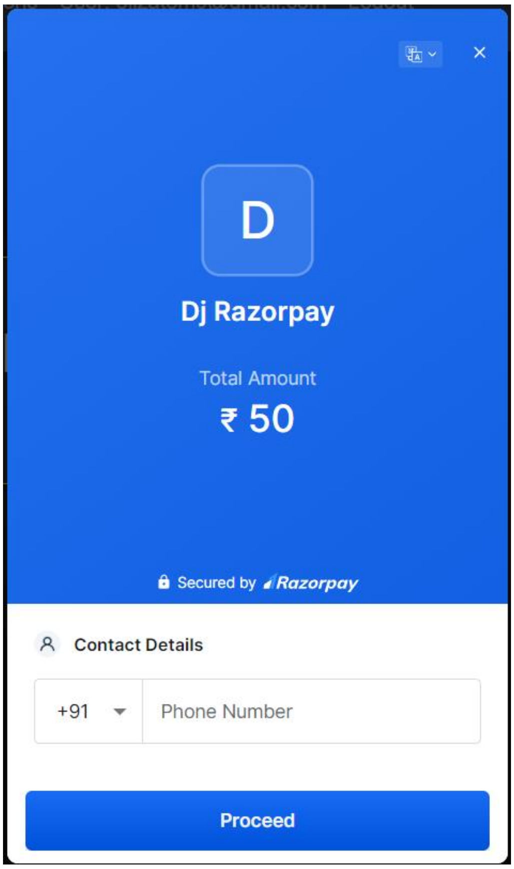


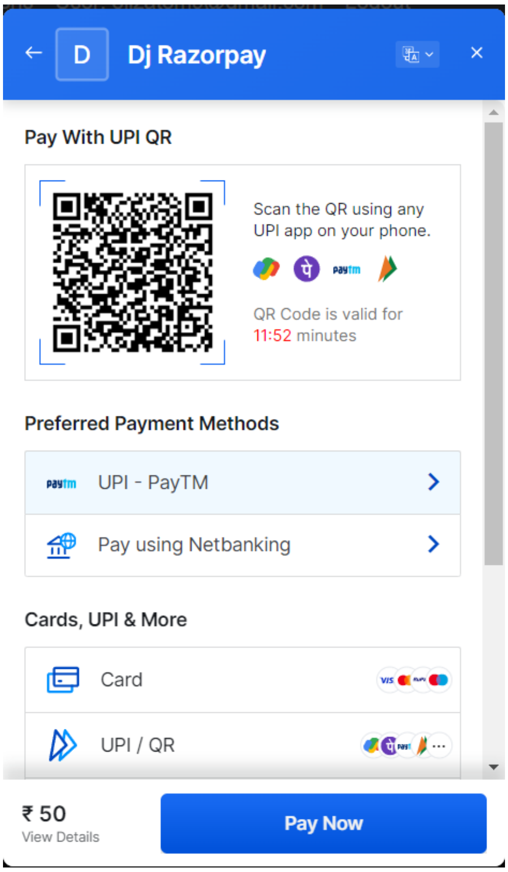
WEBINAR REGISTRATION

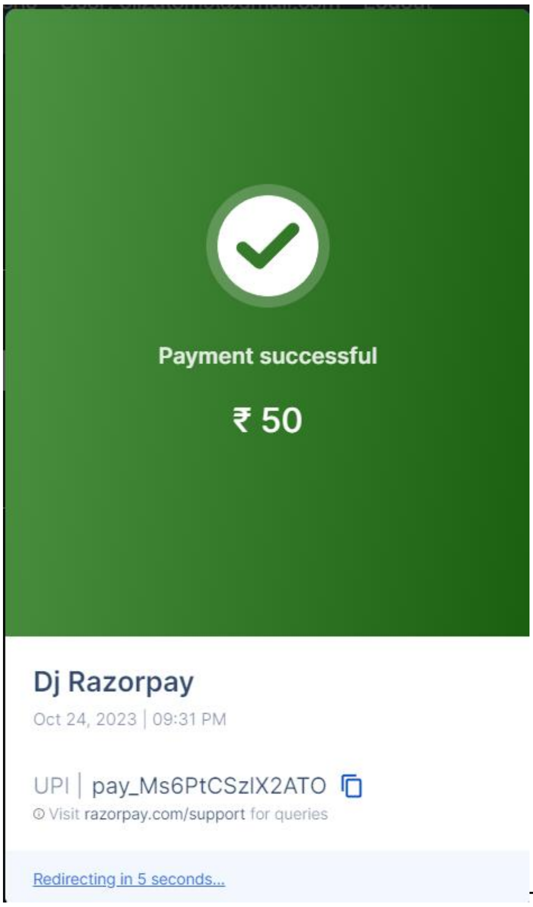
Proceed to pay the registration fee

Pay Go Back to Events

Payment Interface


D
Dj Razorpay
Total Amount
₹ 50
Secured by Razorpay
Contact Details
+91
Phone Number
Proceed


D
Dj Razorpay
Pay With UPI QR
Scan the QR using any UPI app on your phone.
paytm
QR Code is valid for 11:52 minutes
Preferred Payment Methods
UPI - PayTM
Pay using Netbanking
Cards, UPI & More
Card
UPI / QR
₹ 50
View Details
Pay Now


Payment successful
₹ 50
Dj Razorpay
Oct 24, 2023 | 09:31 PM
UPI | pay_Ms6PtCSzIX2ATO
Visit razorpay.com/support for queries
Redirecting in 5 seconds...