**GB EVENT MANAGEMENT WEB APPLICATION**



Student Name: Shafa Ali

Registration Number: 2019-KIU-BS-984

Student Name: Khush Kabeer

Registration Number: 2019-KIU-BS-961

**Department of Computer Science, Faculty of Natural Science and Engineering, KIU, Gilgit**

September, 2023

GB EVENT MANAGEMENT WEB APPLICATION

By

Shafa Ali

2019-KIU-BS-918

Khush Kabeer

2019-KIU-BS-961

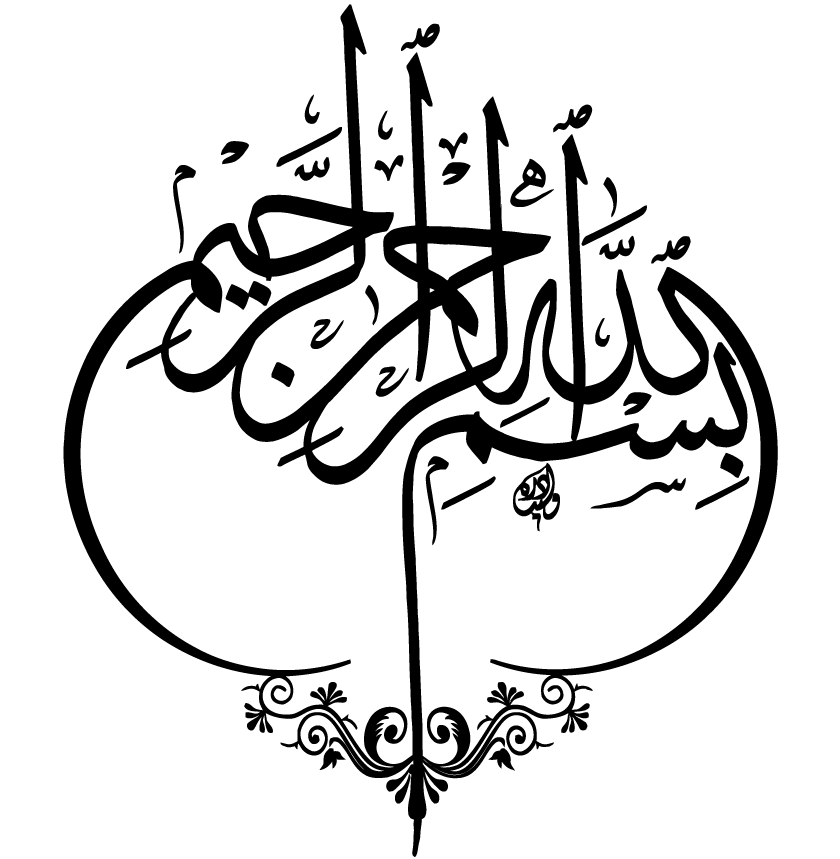
A thesis submitted in partial fulfillment of the requirement for the degree of BS

In

Information Technology

Department of Computer Science, Faculty of Natural Science and Engineering, KIU, Gilgit.

August, 2023

In the Name of Allah Almighty

The Most Beneficent

The Most Merciful

# CERTIFICATE OF APPROVAL

This Project “ GB Event Management web application” submitted by Shafa Ali (2019-KIU-BS-984) and Khush Kabeer (2019-KIU-BS-961) is hereby approved in partial fulfillment of the requirement for the degree of Bachelor of Science in Computer Sciences/Information Technology/Software Engineering” Name of department and University”

Thesis Supervisor Ms. Ponum Almas

Co-supervisor \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Chairman Dr. Zarnawab Khan Swati

External Examiner \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# DEDICATION

To our respected parents, teachers, dear friends on being there for us throughout the entire process and specially dedicated to our Respectable Supervisor Mam Poonam Almas for his encouragement, help, and support for our project.

**DECLARATION**

|  |  |
| --- | --- |
| Signature  Scholar Name: Shafa Ali  Reg: 2019-KIU-BS-984  Scholar Name: Khush Kabeer  Reg: 2019-KIU-BS-961  Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature  Supervisor Name: Mam Poonam Almas  Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

We hereby declare that this thesis/project is a presentation of our own work and that it has not been submitted anywhere for any award. We also warrant, that we have not received outside assistance or involved the external contributions, if received/involved we will acknowledge in written statement to authorities, otherwise we will be liable for the cancellation of our thesis thereby the degree that will be awarded.

# ACKNOWLEDGEMENTS

**“To him belongs the dominion of the Heaven and the Earth, it is He who gives life and death and He has power over all things” (AL-QURAN)**

All praise to Almighty Allah who gave us the understanding, courage and patience to complete this project.

We express our gratitude to our project supervisor **Mam Poonam Almas** who provide us the opportunity to learn and enhance knowledge. As the project supervisor, Mam Poonam Almas been ready to help and guide us throughout the project development. We would also like to thank our teachers in the department and friends for their moral and technical support.

We would like to acknowledge the support of our family members we would like to admit that we owe all our achievements to our truly, sincere and loving parents whose prayers are a source of determination for us.

Last but not least, we would like to extend our gratitude to everyone who has been helping us directly or indirectly from the beginning until the final stage of this project. All the helps and cooperation from various parties are truly appreciated.

# ABSTRACT

The title of our project is the “**GB** **Event Management Web Application**”. Event management web application is the strategic planning, organization, and execution of events to achieve specific objectives and create memorable experiences. It involves a range of tasks, from conceptualizing and designing an event to coordinating logistics, marketing, and post-event evaluation. Event management professionals work in various industries, including corporate, entertainment, weeding, birthdays and social events. They focus on ensuring that events run smoothly, meet their goals, and leave a positive impression on attendees. Event management is a multifaceted discipline that involves meticulous planning, organization, marketing, and execution to create successful and impactful events. It requires a combination of creativity, attention to detail, and effective communication to bring visions to life and deliver memorable experiences

# TABLE OF CONTENTS

Chapter Page

[CERTIFICATE OF APPROVAL iv](#_Toc145020858)

[DEDICATION v](#_Toc145020859)

[DECLARATION vi](#_Toc145020860)

[ACKNOWLEDGEMENTS vii](#_Toc145020861)

[ABSTRACT viii](#_Toc145020862)

[TABLE OF CONTENTS x](#_Toc145020863)

[LIST OF FIGURES xiii](#_Toc145020864)

[ACRONYMS AND ABBRIVIATION xv](#_Toc145020865)

[Chapter 1 1](#_Toc145020866)

[INTRODUCTION 1](#_Toc145020867)

[1.1.Background 1](#_Toc145020868)

[1.2.Study rationale and relevance 2](#_Toc145020869)

[1.3.Aims and objectives 2](#_Toc145020870)

[1.4.Objective 2](#_Toc145020871)

[1.5.Problem Statement 3](#_Toc145020872)

[Chapter 2 4](#_Toc145020873)

[REVIEW OF LITERATURE 4](#_Toc145020874)

[2.1.Literature Review 4](#_Toc145020875)

[2.2.Online Payment Systems in Event Management Application 4](#_Toc145020876)

[2.3.Existing systems 5](#_Toc145020877)

[2.4 Conventional Event Financial Handling and Administrative Procedures in Event Management………………………………………………………………………………………](#_Toc145020878)5

[2.5.Websites 5](#_Toc145020879)

[2.6. Event Mangement Web Application 5](#_Toc145020880)

[2.7. Limitations 6](#_Toc145020882)

[Chapter 3](#_Toc145020885) 7

[REQUIREMENT SPECIFICATION](#_Toc145020886) 7

[3.1.Functional requirements](#_Toc145020887) 7

[3.1.1.Functions](#_Toc145020888) 7

[3.2.Nonfunctional requirement](#_Toc145020889) 8

[3.2.1.Performance](#_Toc145020890) 8

[3.2.2.Usability 10](#_Toc145020891)

[3.2.3.Reliability 10](#_Toc145020892)

[3.2.4.Security](#_Toc145020893) 9

3.2.5 Scalability 9

3.2.6 Compatibility 9

3.2.7 Documentation and Training 9

[3.3.Required tools and technologies 1](#_Toc145020894)0

[3.3.1.Programming Language 1](#_Toc145020895)1

[3.3.2. Frontend Framework and libraries 1](#_Toc145020896)1

3.3.3 Database Management System 10

3.3.4 UI/UX design.................................................................................................10

[Chapter 4](#_Toc145020897) 11

[DESIGN](#_Toc145020898) 11

[4.1.System architecture…………..](#_Toc145020899) 11

[4.2.Design Constraints………………](#_Toc145020900) 11

[4.3.Design methodology…………..](#_Toc145020901) 11

[4.4.High Level Design…………… … 1](#_Toc145020902)2

[4.4.1.Use case Models…………… 1](#_Toc145020903)3

[4.4.2.Data Flow ……….………….. 18](#_Toc145020904)

[4.5.Low-Level Design………………](#_Toc145020905) 20

[4.6.Database Design………………….. 22](#_Toc145020906)

[4.6.1.Some of the basics of database: 22](#_Toc145020907)

[4.6.2.Design Database:……………..](#_Toc145020908) 20

[4.7.GUI Design……………………](#_Toc145020909) 21

[4.7.1.Home Screen:………………….](#_Toc145020910) 22

[4.7.2.Login Screen:………………..](#_Toc145020911) 23

[4.7.3.About us:………………………..](#_Toc145020912) 23

[4.7.4.What to do:……………………..](#_Toc145020913) 24

[4.7.5.Registration Screen:…………](#_Toc145020914) 24

[4.7.6.Management Team Dashboard: 26](#_Toc145020916)

[Chapter 5](#_Toc145020920) 25

[System Implementation](#_Toc145020921) 25

[5.1 Implementation](#_Toc145020922) 25

[5.2 Tools and Technologies](#_Toc145020923) 25

[5.3 Frontend](#_Toc145020924) 25

[5.4 Backend](#_Toc145020925) 26

[5.5 Database](#_Toc145020926) 26

[5.6 Testing and Quality Assurance](#_Toc145020927) 26

[5.7 Deployment and User Training](#_Toc145020928) 27

[5.8 Maintenance and Updates](#_Toc145020929) 28

[Chapter 6](#_Toc145020930) 29

[SYSTEM TESTING AND EVALUTION](#_Toc145020931) 29

[6.1 Manual Testing](#_Toc145020932) 29

[6.1.1Unit testing](#_Toc145020933) 29

[Chapter 7](#_Toc145020934) 30

[CONCLUSION](#_Toc145020935) 30

[FUTURE WORK](#_Toc145020936) 31

[REFERENCES](#_Toc145020937) 32

[APPENDICES](#_Toc145020938) 33

[Appendix A](#_Toc145020939) 33

[A ppendix B](#_Toc145020940) 34

[Main frontend code:](#_Toc145020941) 34

[Main Backend Code:](#_Toc145020942) 35

# LIST OF FIGURES

Figure Page

Figure 4.1: System use case diagram…………………………………………………………………….....14

Figure 4.4.1: Use Case Diagram for Users………………………………………………………………….15

Figure 4.4.2: ER Diagram for Event Management………………………………………………………....16

Figure 4.7: Home Screen……………………………………………………………………………………22

Figure 4.7.2: Log in screen……………………………………………………………………………….....23

Figure 4.7.3: About us………………………………………………………………………………………23

Figure 4.7.4: what we do……………………………………………………………………………………24

Figure 4.7.5: Registration screen…………………………………………………………………………....24

|  |  |  |
| --- | --- | --- |
| IT |  | INFORMATION TECHNOLOGY |
| KIU |  | KARAKORAM INTERNATIONAL UNIVERSITY |
| CS |  | COMPUTER SCIENCES |
| SE |  | SOFTWARE ENGINEERING |
| APP |  | APPLICATION |
| ICT |  | INFORMATION AND COMMUNICTION TECHNOLOGY |
| DFD |  | DATA FLOW DIAGRAM |
| MHz |  | MEGA HERTZ |
| GHz |  | GIGA HERTZ |
| DFD |  | DATA FLOW DIAGRAM |
| GB |  | GIGA BYTE |
| RAM |  | RANDOM ACCESS MEMORY |
| UID |  | USER IDENTITY |
| ERD |  | ENTIT RELATIONAL DIAGRAM |
| GUI |  | GRAPHICAL USER INTERFACE |
| RDBMS |  | RELATIONAL DATABASE MANAGEMENT SYSTEM |
| UI |  | USER INTERFACE |

# ACRONYMS AND ABBRIVIATION

# Chapter 1

## INTRODUCTION

## Background

## This is an event management web application software project that serves the functionality of an event manager. The system allows only registered user to login and new users are allowed to register on the application. This is a web application developed in Asp.net and Sql but desktop application of the same application is also available. The project provides most of the basic functionality required for an event. It allows the user to select from a list of event types. Once the user enters an event type e.g. (Marriage, Dance Show etc.) the system then allows the user to select the date and time of event, place and the event equipment’s. All this data is logged in the database and the user is given a receipt number for his booking. This data is then sent to the administrator (website owner) and they may interact with the client as per his requirements and his contact data stored in the database. The aim of the Event Management Process is identifying events and determining corresponding control measures. There can be several updates or changes in a service or configuration item. Some of these changes can be critical while some changes can be minor without impacting other aspects of the IT services. The categorization of these events and defining appropriate control measures for these different events is an objective of the Event Management Process. Event Management Process is providing a basis for service assurance, reporting and service improvement. IT service providers aim for service improvement to improve the provided services consistently to increase the value provided to the employee. The Event Management Process helps to increase this value delivered to the employee.

## 

## Study rationale and relevance

Event management is a dynamic field that necessitates efficient coordination and organization. A web application tailored for event management offers a centralized platform to streamline event planning, scheduling, and execution. In an increasingly digital world, such a tool aligns with the evolving needs of event organizers, providing real-time collaboration and accessibility. By integrating features like attendee registration, budget tracking, task delegation, and communication tools, the application enhances efficiency, minimizes errors, and promotes effective decision-making. Moreover, it facilitates data analysis, offering insights into attendee preferences and event success metrics. In essence, this web application is pertinent, addressing the modern demand for enhanced event coordination, ultimately elevating event experiences and optimizing resource utilization.

## Aims and objectives

**Aims:**

1. Streamline event planning and organization processes.
2. Enhance user experience for event organizers and attendees.
3. Facilitate real-time communication and collaboration.
4. Automate repetitive tasks and workflows in event management.
5. Enable data analytic and insights for informed decision-making.
6. Ensure scalability and flexibility to handle various events.
7. Foster integration and connectivity with other platforms.

**Objectives:**

1. Allow easy event creation and management by organizers.
2. Enable attendee registration and engagement with event details.
3. Implement a secure ticketing and payment system for events.
4. Enable efficient communication and notifications for event updates.
5. Facilitate vendor and sponsor management for event organizers.
6. Gather feedback and conduct surveys for event evaluation and improvement.
7. Provide comprehensive reporting and analytic on event performance.
8. Prioritize security and data privacy in handling user information.
9. Optimize search and navigation for quick access to event information

**Problem Statement:**

The event management landscape often grapples with inefficiencies and challenges that can be effectively addressed through a well-designed web application. Currently, manual and disjointed processes dominate the field, leading to cumbersome event planning, data discrepancies, and delays. Organizers struggle with scattered communication channels, registration complexities, and inadequate real-time tracking of event progress and resource allocation. Attendees face difficulties in seamless registration, obtaining event updates, and providing feedback. Budget management and financial transparency are often lacking, hindering optimal resource utilization. Moreover, the absence of comprehensive analytic limits the ability to make data-driven decisions for future events. These challenges highlight the urgent need for an integrated event management web application that centralizes operations, enhances communication, streamlines processes, improves attendee experiences, ensures financial prudence, and provides insightful data analytic to elevate the event management domain.

Chapter 2

## REVIEW OF LITERATURE

## Literature Review

The literature on event management web applications highlights their significance in modern event coordination. These robust platforms leverage JavaScript, React JS, HTML, CSS, and Bootstrap to provide intuitive interfaces, simplifying event type selection, date scheduling, venue management, and equipment coordination. Database integration ensures secure and organized data handling. Noteworthy features include user registration protocols, stringent authentication, and responsive design implementation, all contributing to an enhanced user journey. These applications prioritize streamlining event planning processes, improving service quality, and optimizing stakeholder engagement through dynamic event control measures, underscoring their role in delivering exceptional value.

## Online Payment Systems in Event management

## Online payment systems have revolutionized the landscape of event management, offering a seamless and secure means of financial transactions. These systems facilitate convenient and swift processing of event payments, ranging from ticket purchases to event registrations and merchandise sales. Employing encryption and authentication protocols, they ensure robust security, instilling trust among users. Furthermore, online payment systems enable event organizers to efficiently track and manage transactions, providing real-time insights into revenue generation. The integration of diverse payment methods caters to a broader audience, enhancing the accessibility and inclusivity of events. Overall, online payment systems have become an indispensable component of event management, enhancing the efficiency, transparency, and overall experience for both organizers and attendees alike.

## Existing systems

Current event management web applications encompass a range of robust systems designed to streamline the entire event organization process. These existing platforms integrate modern technologies such as JavaScript, React JS, HTML, CSS, and Bootstrap to deliver intuitive and user-friendly interfaces. They typically offer features like event type selection, date and time scheduling, venue management, and equipment coordination. Incorporation of databases ensures efficient data storage and retrieval, enabling organizers to manage attendee information and event logistics effectively. Moreover, user authentication, registration, and secure payment gateways are integral aspects, ensuring a secure and personalized experience for users. These systems often provide event analytics and reporting features, empowering organizers with valuable insights for future event planning and strategy. The ever-evolving nature of these applications constantly incorporates new functionalities and optimizations to enhance event planning and execution processes.

## Conventional Event Financial Handling and Administrative Procedures in Event Management

## Websites

We will discuss the diverse array of web-based services and programs utilized in event management.

**Tailored Event Management Portals**

In the realm of event management tailored web portals play a pivotal role. Event organizers often develop customized web portals to efficiently handle a range of event-related tasks. These portals are crafted to align with the specific event requirements and branding. For instance, an event management team might create a dedicated portal where participants can log in, register for the event, make payments, and access essential event documents.

* 1. **Event Management Platforms**

Modern event management heavily relies on specialized platforms that streamline the planning and execution process. These platforms offer comprehensive features for event organization, attendee registration, fee processing, and document management. Originally designed for event planning, these platforms have evolved to integrate administrative functionalities, enhancing the overall efficiency of event management. From event logistics to attendee engagement, these platforms serve as centralized hubs for successful event coordination.

## Limitations

Event management systems, despite their many advantages, have notable limitations. Firstly, they often require a learning curve, especially for users unfamiliar with the specific software, which can lead to potential delays and errors during implement action. Additionally, customization can be limited, making it challenging to tailor the system to unique event requirements. Scalability can also be an issue, as the system may struggle to handle a sudden increase in event complexity or volume. Another limitation lies in potential dependencies on internet connectivity; if the system is primarily web-based, disruptions in internet access can hinder real-time updates and communications. Lastly, cost can be a barrier, especially for smaller organizations or events with limited budgets, as robust event management systems often entail significant upfront and ongoing expenses. Despite these limitations, careful selection, training, and adaptation strategies can mitigate these challenges and maximize the benefits of event management systems

# Chapter 3

## REQUIREMENT SPECIFICATION

## 3.1 Functional requirements

A comprehensive event management web application should encompass several key functional requirements to streamline the event planning and execution process effectively. First and foremost, it should provide a user-friendly interface for user registration and authentication, enabling different levels of access based on user roles. The application should facilitate event creation and management, allowing organizers to define event details, upload event-related media, and edit or delete events as needed. Attendee management features should enable easy registration and communication with attendees, along with agenda and schedule management for both organizers and attendees to keep track of event activities. Seamless communication through automated notifications and updates is essential. Additionally, the application should offer capabilities for venue and resource management, budget and financial oversight, sponsor and partner management, as well as feedback collection and analysis. Robust reporting and analytics features should be incorporated to provide valuable insights for organizers. Moreover, ensuring accessibility, multilingual support, and effective search and filtering options contribute to an inclusive and efficient event management web application. These functional requisites collectively empower event organizers and enhance the event experience for all stakeholders involved

### **Functions**

The main functions of “**Event management web application**”

An event management web application encompasses a set of essential functions that streamline the entire event life cycle. Firstly, it offers event creation and editing capabilities, allowing organizers to input event details and modify them as needed, ensuring accurate and up-to-date event information. Attendee registration and management functions facilitate smooth attendee sign-up processes and enable organizers to efficiently manage attendee lists, enhancing overall event coordination. Agenda and schedule management features allow for the creation of event agendas, including sessions and speakers, providing attendees with a clear event schedule and customization options. Communication and notifications functions ensure effective dissemination of event updates and announcements to keep attendees informed. Venue and resource management functions assist in selecting event venues and tracking necessary resources, optimizing event logistics. Budget and financial tracking features enable budget planning, monitoring, and financial reporting, promoting financial transparency and prudent expenditure. Collaboration with sponsors and partners is facilitated through functions that manage agreements, packages, and branding opportunities. Feedback and survey collection functions aid in gathering valuable insights from attendees, enabling organizers to make data-driven improvements for future events. Reporting and analytics functions generate comprehensive event reports and provide data-driven insights, empowering organizers with valuable information for decision-making. Lastly, search and filtering functions enhance attendee experience by facilitating easy event discovery and navigation based on various criteria, ensuring a user-friendly interface and seamless event exploration. Together, these functions contribute to the effectiveness, efficiency, and success of event management within the web application.

## Nonfunctional requirement

### **Performance:**

Ensure fast response times for all actions within the application, aiming for minimal loading and processing delays.

Support scalability to handle a significant increase in users and data without compromising performance

### **Usability**

Design an intuitive and user-friendly interface to enhance ease of use for both organizers and attendees

Ensure consistent design elements and navigation across the application to provide a cohesive user experience.

### **Reliability**

Maintain a high level of availability to ensure the application is accessible and operational at all times, minimizing downtime.

Implement backup and recovery strategies to quickly restore the system in case of failures or data loss.

### **Security**

Implement robust authentication and authorization mechanisms to ensure secure access to the application and its data.

Encrypt sensitive data (e.g., financial information, personal details) to protect against unauthorized access or breaches

### **Scalability**

Design the application architecture to scale horizontally and vertically, accommodating a growing number of events, attendees, and organizers.

Ensure efficient database scaling to handle increased data storage demands.

### **Compatibility**

Ensure compatibility with a variety of devices, browsers, and operating systems to cater to a broad user base

Optimize the application for various screen sizes, including mobile devices, tablets, and desktops.

### **Documentation and Training**

Provide comprehensive documentation, user guides, and tutorials to assist users in understanding and effectively using the application.

Offer training and support resources to educate users on utilizing the application's features optimally

## 3.3 Required tools and technologies

The required tools and technologies to run this web app are as under.

Developing an event management web application involves using a combination of tools and technologies to ensure a robust, scalable, and user-friendly product. Here are the key tools and technologies required for creating an event management web application:

### 3.3.1 Programming Languages

* **JavaScript**: A fundamental language for web development, used for both fronted and back end development
* **HTML/CSS**: Basic building blocks for creating the structure and styling of web pages.

### 3.3.2 Fronted Frameworks and Libraries

* **React:** A popular JavaScript library for building interactive user interfaces and single-page applications.

### 3.3.3 Database Management Systems

* **MySQL, PostgreSQL, MongoDB:** Popular database systems for storing and managing application data, each with its own strengths based on the application requirements

### 3.3.4 UI/UX Design

* **Fig-ma, Adobe XD, Sketch:** Design tools to create wire frames, prototypes, and the overall user interface of the application

# Chapter 4

## DESIGN

## System architecture

The system architecture of an event management web application is a multi-tiered structure that seamlessly integrates the fronted, back end, and database components. At the fronted, user interfaces are designed using HTML, CSS, and JavaScript frameworks like React or Angular to create an intuitive user experience. On the server-side, an HTTP server, often Node.js with Express, handles requests, while application logic manages functionalities like event creation, registration, and database interactions. The database layer employs a Database Management System (DBMS) such as MySQL or MongoDB to store and manage data. REST ful API s serve as the bridge between fronted and back end, allowing communication and data retrieval. Middleware, authentication services, and third-party integration enhance application security and functionality. The system is deployed on cloud servers, ensuring scalability and performance, and orchestrated through containerization and continuous integration/continuous deployment (CI/CD) pipelines for efficient management and updates. This architecture culminates in a comprehensive, reliable event management web application.

## Design Constraints

Designing an event management web application involves navigating various design constraints. Firstly, user experience is critical, necessitating an intuitive interface for users of varying technical proficiency. Achieving cross-platform compatibility and responsiveness across devices adds to the complexity, ensuring the application functions seamlessly on computers, tablets, and Smartphone. Accessibility is another constraint, requiring adherence to accessibility standards to cater to individuals with disabilities. Striking a balance between rich features and simplicity is challenging; too many features can overwhelm users, while too few may limit functionality. Security is paramount, demanding robust measures to protect sensitive user data and financial transactions. Scalability is a key concern as the application must handle varying event sizes and complexities without performance degradation. Finally, integration capabilities with other systems and API s, such as payment gateways and social media platforms, are essential for a well-rounded event management web application. Navigating and addressing these constraints are fundamental to designing an effective and successful event management web application

## Design methodology

The design methodology for developing an event management web application involves a structured approach to create an efficient and user-centric system. It commences with a thorough understanding of requirements through stakeholder collaboration and analysis, defining the project scope and objectives. Comprehensive market analysis and research guide conceptualization, wherein architecture, features, and functionalities are ideate. Selection of an appropriate technology stack based on scalability, performance, and other criteria is crucial. The architectural design delineates the system's structure and components, integrating databases and UI/UX design. Database modeling ensures optimal data storage and retrieval. Development and implementation follow an agile approach, dividing the project into iterations. Rigorous testing and quality assurance are integral to deliver a robust, bug-free application. Deployment on chosen platforms and post-launch evaluation through user feedback further refine the application. Continuous iteration based on feedback ensures the application remains adaptable to evolving needs and technologies, resulting in a successful event management web application.

## High Level Design

In the high-level architecture of the event management application, we outline fundamental components and their interactions to provide an overview of the system's structure and flow.

**User Management System:** Responsible for user registration, login, and session management functionalities. It ensures a secure and seamless user experience throughout the application.

**Event Creation and Management Module:** Enables event organizers to create and manage events. This includes defining event details, setting up registration forms, and managing attendee data.

**Payment Processing Module:** Manages payments related to event registration, ticketing, and other transactions. It integrates with payment gateways for secure and efficient financial transactions.

**Attendee Engagement Module:** Facilitates interaction and engagement with event attendees. It may include features like event updates, attendee networking, and Q&A sessions.

All these components are intricately connected to the application's database, which centralizes data related to user profiles, event details, payment records, attendee information, and engagement activities.

* + 1. **Use case Models**

Step into the dynamic world of our event management web application, where the user experience is brought to life through engaging use cases. The Event Organizer takes the reins, effortlessly creating and modifying events with vital details like date, time, and location. Meanwhile, the Attendee enjoys a seamless journey, effortlessly registering for events, making secure payments for tickets, and immersing themselves in the event. The Administrator dons the hat of oversight, wielding the power to analyze event data, manage user accounts, and ensure the seamless orchestration of the event. And through our integrated Payment Gateway, users experience swift and secure transactions, enhancing the overall event participation. Embark on an exciting event management journey, experiencing the app's versatility through these vibrant and essential user interactions.

**Actors:**

**Admin:** Event management admins handle administrative, logistical, and coordination tasks such as scheduling, vendor coordination, budget management, and problem-solving to ensure successful event planning and execution.

**Event Manager:** Event manager is responsible to manage all events.

**Event Customer:** The work of event customers, or attendees, primarily involves registering for and attending the event, actively participating in its activities, adhering to event rules, and providing feedback to help organizers improve future events.

**(Admin, Events, and Customer):** Users responsible for approving document applications.

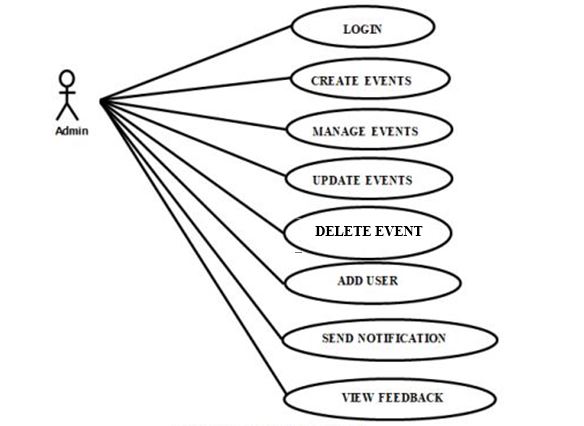


Figure 4.1: System Use Case Diagram

**Actions in Admin Use Case:**

1. **Login:** Once they have registered they need to login to avail the service at the needy time.
2. **Create Event:** An admin can be create the events and send to the employees.
3. **Manage Events:** Event managers are responsible for making sure events go smoothly. They handle everything from planning and budgets to vendors and guest experiences to meet the event's goals**.**
4. **Update Events:** An admin can be updated the new events or new sub events.
5. **Delete Events:** An admin will be delete the older events andnon-conduct events.
6. **Add Users:** An admin can be add new employee and their details.
7. **Send Notification:** An admin add the new events, after send to the notifications to the employee.
8. **View Feedback**: An admin view the feedback from the employee.

**Use Case for Users:**

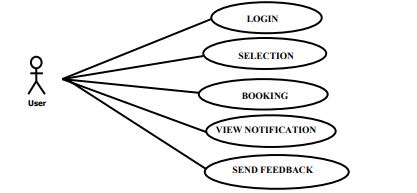
****

Figure 4.4.1: Use Case Diagram for Users

### **User**

**1. Login:** Once they have registered they need to login to avail the service at the needy time.

**2. View notification**: An employee can view event notification.

**3. View events:** An employee can view new events.

**4. Send feedback:** An employee will be send the feedback about recent events.

**ER Diagram for Event Management**

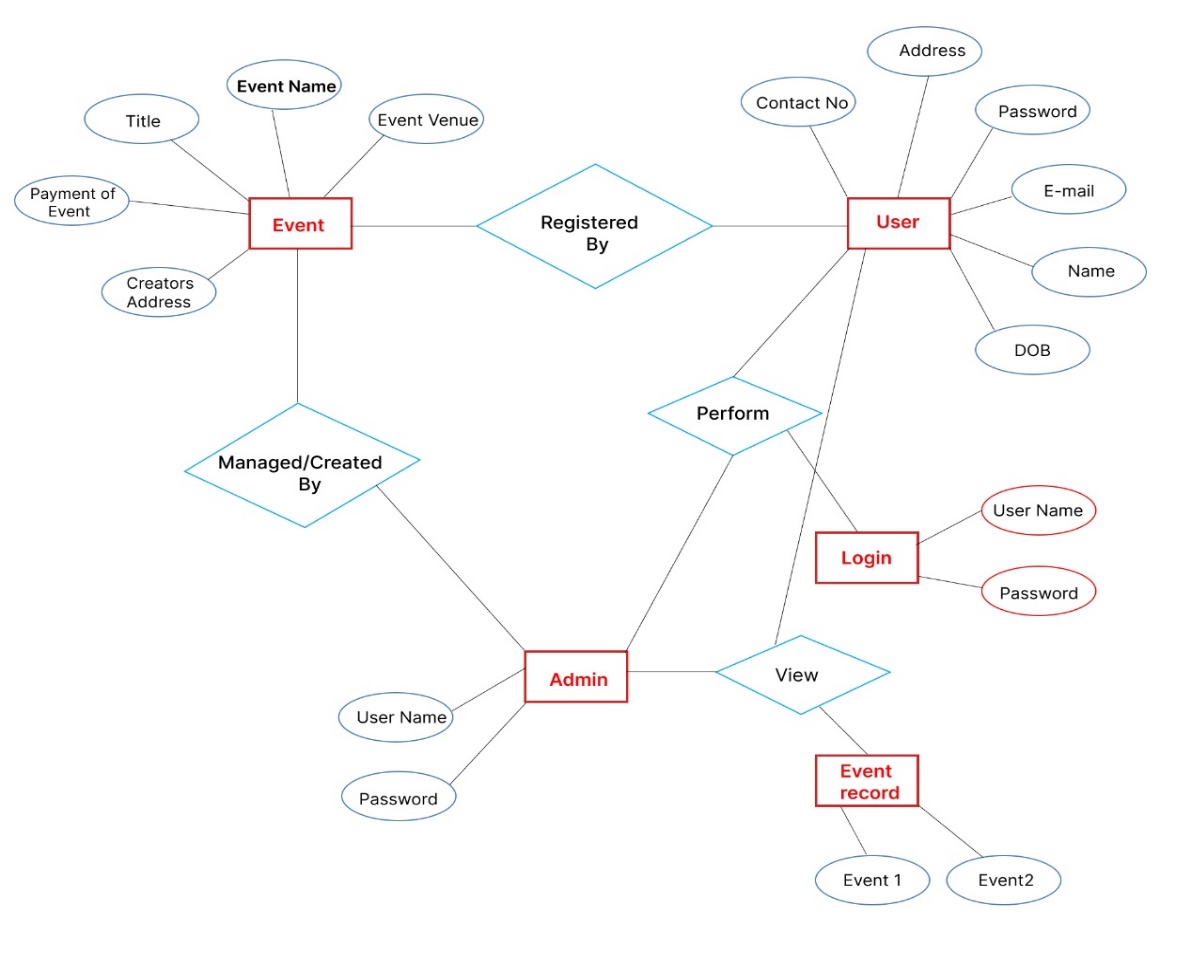
****

Figure 4.4.2: ER Diagram for Event Management

1. **Event**

* Event ID (Primary Key)
* Event Name
* Event Date
* Event Title
* Event Venue
* Event Creators
* Event Cost

1. **User**

* User ID (Primary Key)
* First Name
* Last Name
* Email
* Phone

1. **Login**

* Login ID (Primary Key)
* Username
* Password

1. **Admin**

* Admin ID (Primary Key)
* Admin Name
* Admin Email

1. **Event Record**

* Record ID (Primary Key)
* Event ID (Foreign Key)
* User ID (Foreign Key)
* Attendance Status
* Registration Date

**Relationships:**

**Organizes (One-to-Many)**

An Event can be organized by one User (who acts as an organizer), but a User can organize multiple events.

Connects User (1) to Event (Many) using the User ID as a foreign key in the Event entity.

**Registers (Many-to-Many)**

A User can register for multiple events, and an event can have multiple attendees (Users).

To represent a many-to-many relationship, create a junction table (Event Record) with foreign keys to both Event and User entities. This table also includes attributes like "Attendance Status" and "Registration Date" to track event attendance and registration dates.

**Manages (One-to-Many)**

An Admin can manage multiple events, but each event has one Admin responsible for it. Connects Admin (1) to Event (Many) using the Admin ID as a foreign key in the Event entity.

**Performs (One-to-Many)**

A User (an organizer) can perform multiple logins to access the event management system, but each login record is associated with one User.

Connects User (1) to Login (Many) using the User ID as a foreign key in the Login entity.

**Belongs To (One-to-Many)**

Each Event Record belongs to one Event and one User.

Connects Event (1) to Event Record (Many) using the Event ID as a foreign key in the Event Record entity, and connects User (1) to Event Record (Many) using the User ID as a foreign key in the Event Record entity.

**Data Flows:**

**Event Registration Data**: User data for event registration, including personal information such as name, address, and identification details, flows from event participants to the "Registration Process."

**Payment Details**: Payment information, including transaction details and payment methods, is exchanged between the system and the payment gateway used for event ticket purchases.

**Event Information**: Details about the events, including event names, dates, and descriptions, flow from the "Event Database" to various parts of the application for event listing, promotion, and registration.

**Participant Data**: Information about event participants, including their registration details and preferences, is stored in the "Participant Records" data store and is used for event management and communication.

**Explanation:**

**External Entities:** External entities in our event web application are the various individuals and partners who engage with the system. This encompasses event participants, event organizers (admins), and external collaborators. Each of these entities fulfills unique roles within the application, such as event registration, event management, and offering event-related services.

**Processes:** In our event web application, processes represent essential functions and tasks. These include event registration, creation, payment processing, and communication with external services. These processes work together to ensure the application operates seamlessly.

**Data Stores:** Data stores in the event web application store diverse information, including event details, participant records, and possibly documents, event history, or external partner data based on the application's needs.

**Data Flows:** Data flows in the event web application illustrate how information moves between users, processes, and data stores, enabling the application's functionality.

This Level 0 DFD offers a high-level overview of the event web application's functionality and the interactions among its various components. To provide more detailed insights into each process or component, you can expand this DFD into Level 1 or Level 2 diagrams as needed.

## Low-Level Design

## The low-level design for our event web application encompasses the granular

## Implementation details of critical modules:

## Event Registration Flow: This module focuses on the step-by-step process of how users register for events on the application. It covers aspects like user interactions, data validation, and communication with external systems if needed.

## Event Creation Workflow: This module explains the process of creating events within the application, including defining event details, registration parameters, and categories. It also outlines how event organizers can initiate approval workflows for submitted event proposals.

## Approval Workflow Logic: This module focuses on how events get approved. It explains the decision-making process, how it works with different roles (like event organizers or administrators), and how it tells event creators and participants about the approval status.

## The low-level design serves as a detailed technical plan, ensuring that the event web application's implementation runs smoothly and effectively.

## Database Design

Efficient data storage and management are vital for any software or application, and the database serves as the central repository for project data.

### Basics of database:

Normalization: In the context of our Events Management Web Application, normalization is the practice of organizing data in a professional and structured manner within the database. The normalization process serves two main objectives: eliminating duplicate data and ensuring that data relationships are logically sound. This process involves three key levels:

First Normal Form**:** Ensuring that every table contains only atomic attributes (indivisible data elements).

* Second Normal Form**:** Ensuring that every non-key attribute in each table is entirely and directly dependent on the primary key attribute.
* Third Normal Form**:** Ensuring that no transitive dependencies exist within any table.

Normalization plays a critical role in maintaining a well-structured and efficient database for our Events Management Web Application.

### Design Database:

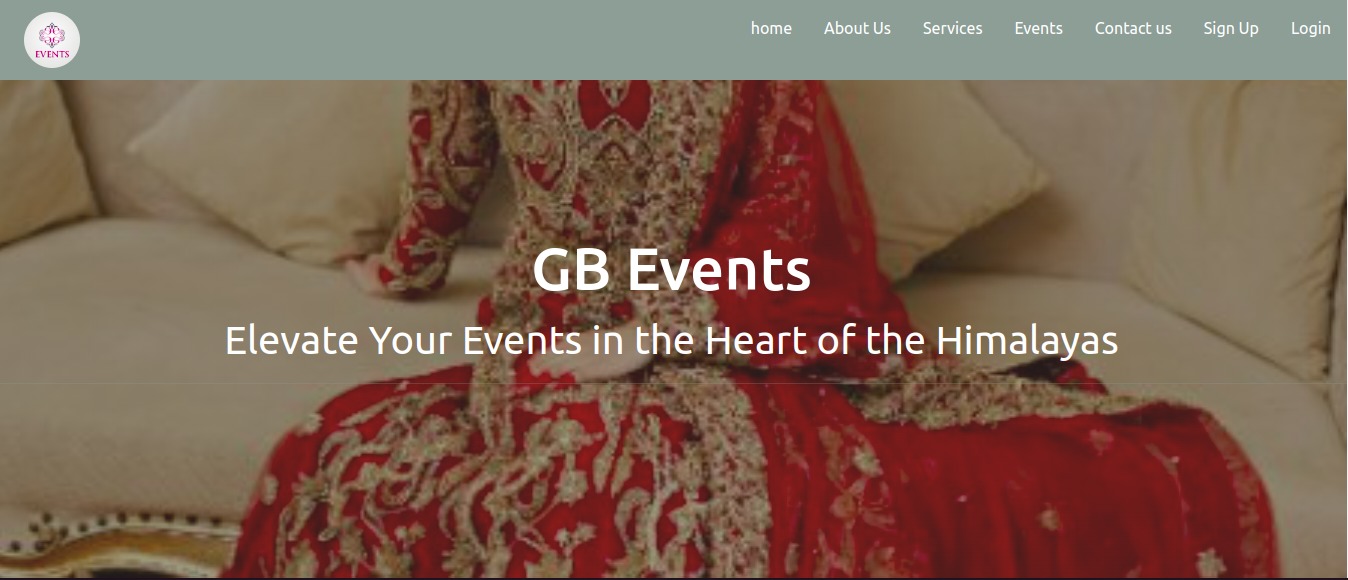
The database design for the GB Event Web Application is structured as follows:

* **Event Attendees Table:** Stores user profiles, including their information, authentication credentials, and access permissions.
* **Payment Records Table:** Maintains a record of payment transactions, various fee types, and payment statuses for event attendance.
* **Event Applications Table:** Stores detailed information about applications submitted by users, including timestamps and applicant data.
* **Approval Tracking Table:** Monitors the progress of application approvals, documenting decisions made by different stakeholders in the GB Event Web Application.

## GUI Design

The graphical user interface (GUI) design focuses on simplicity, usability, and a consistent user experience. Wire frames and mock ups have been created for key screens:

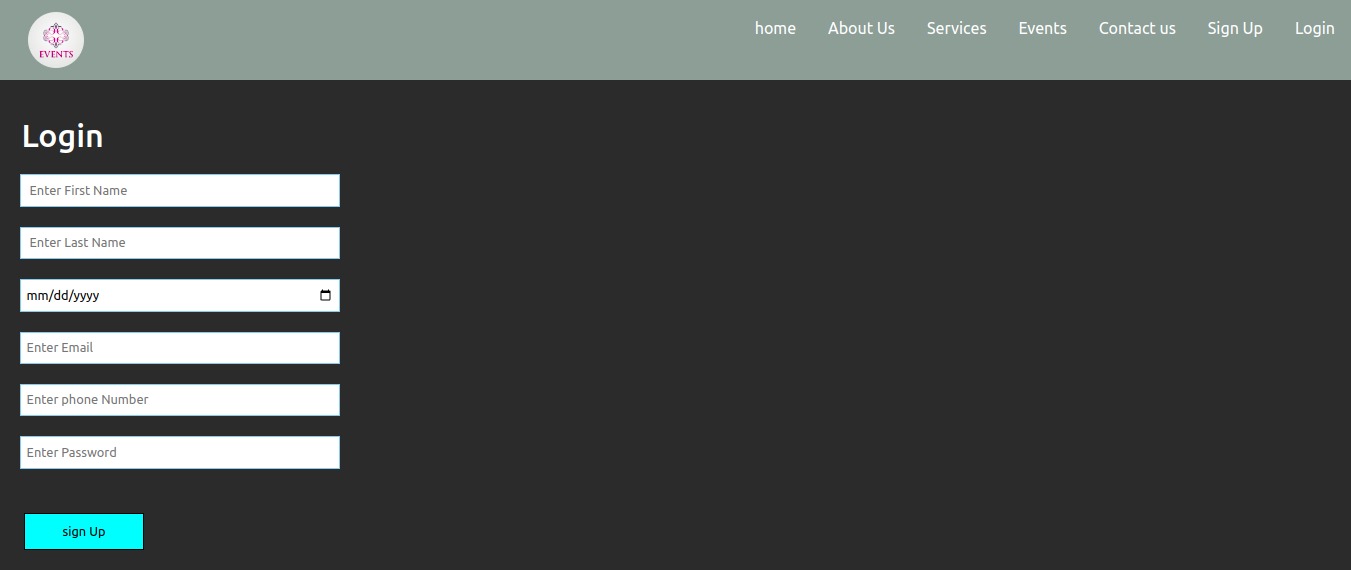
### **Home Screen:**



Displays the Events name, logo, and login options for users.

Figure 4.7 Home screen

### **Log in Screen:**

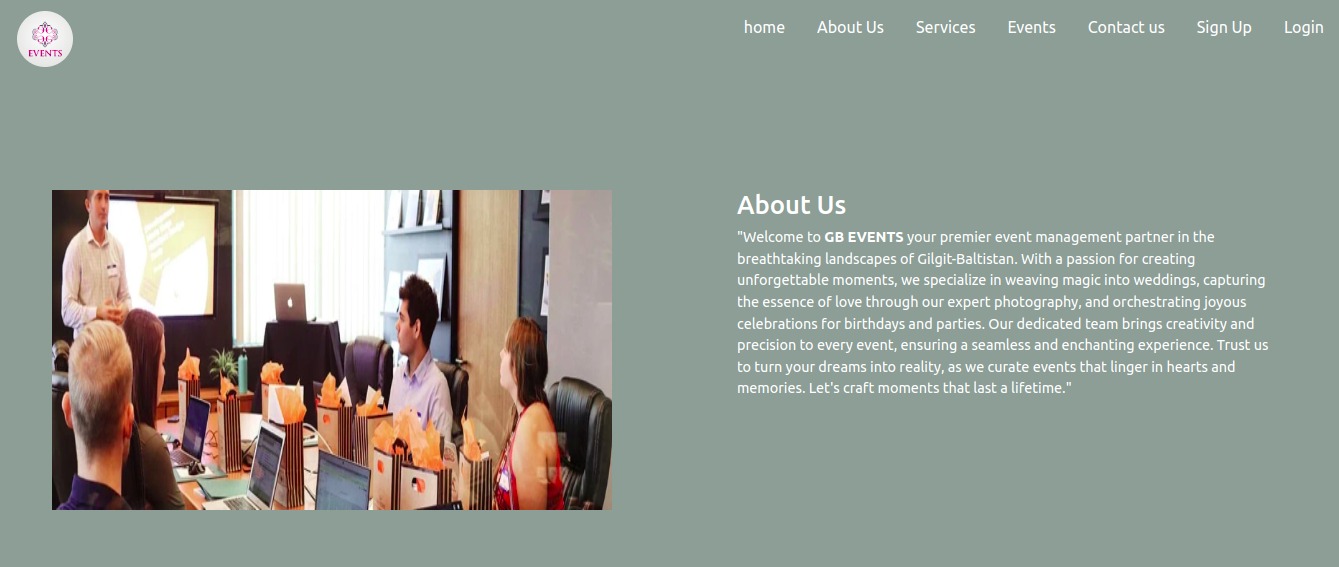


Displays the Events name, logo, two field for login credentials (username and password) and a Sign In button

Figure 4.7.2 log in screen

Top of Form

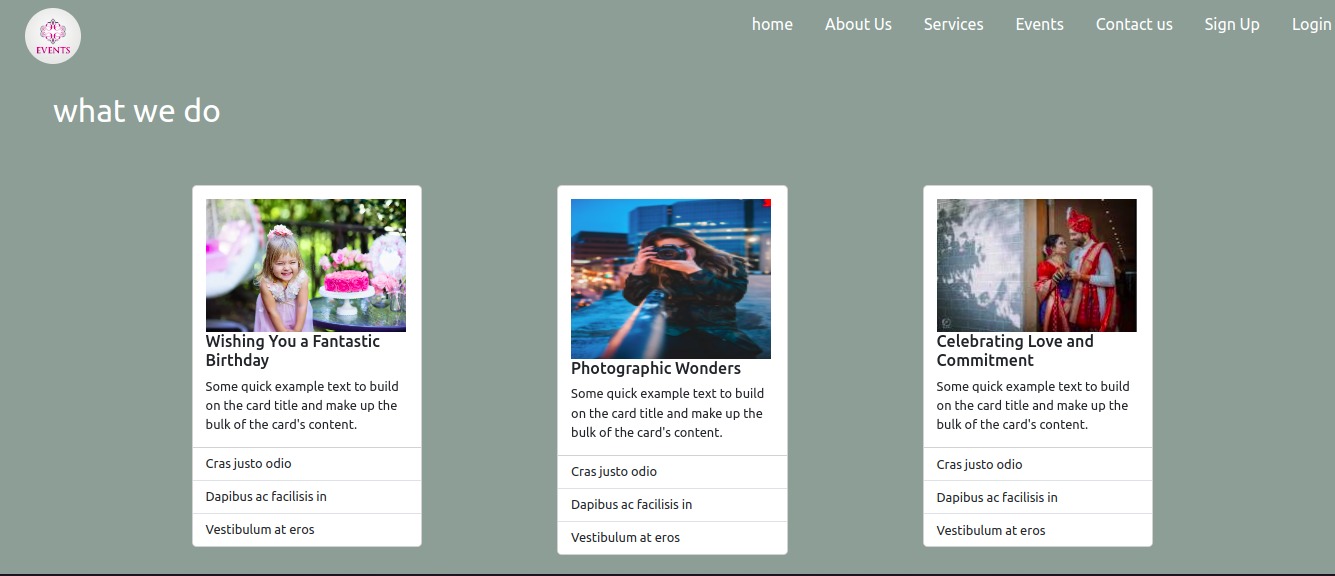
### **4.7.3:** **About Us:**



This section introduces our team and highlights what makes GB Events unique in the event management field. It aims to give potential clients a clear understanding of our identity and expertise

Figure 4.7.3 About us

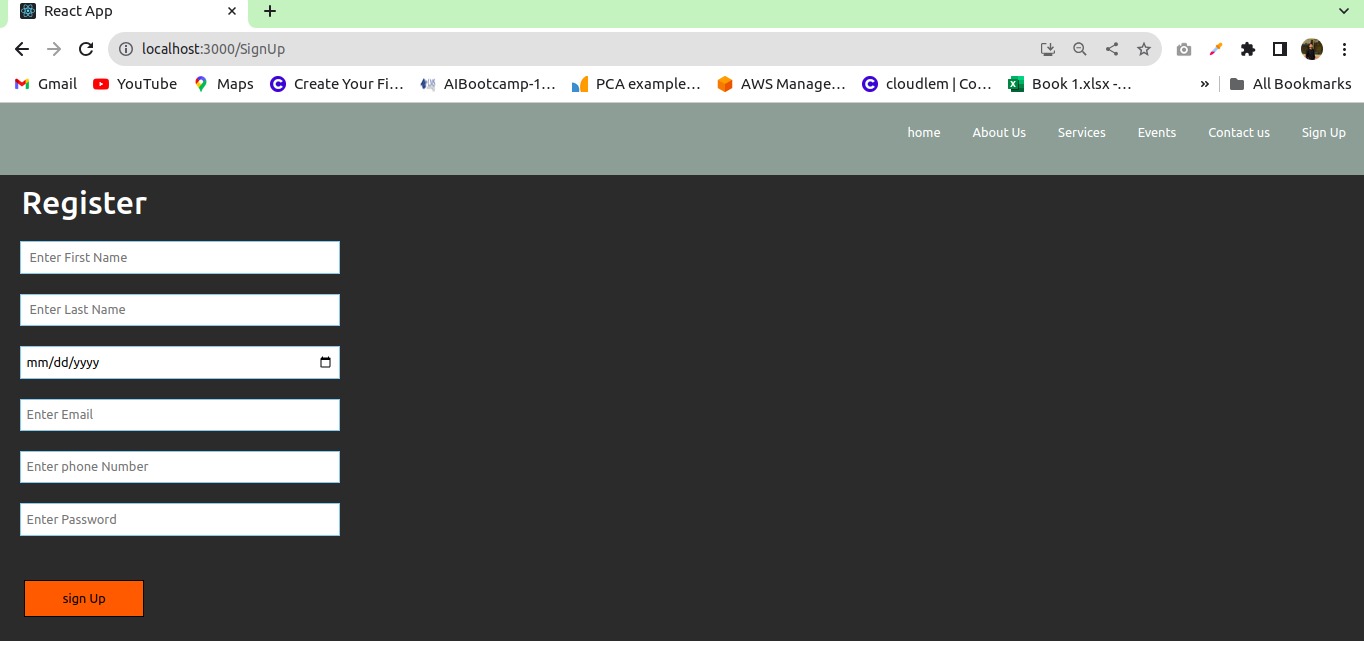
### **What we do:**



It provides users with a quick overview of the app's key features, services, or helping them grasp its value and how it can meet their needs. This section serves as a user guide, making it easier for individuals to navigate the app and make informed decisions about how to leverage its features to their advantage.

Figure: 4.7.4 what we do

**4.7.5 Registration Screen:**



Users provide essential information like their name, email address, and password, which is then stored securely for future access. Registration screens often include validation mechanisms to ensure data accuracy, like email verification.

Figure 4.7.5 Registration Screen

# Chapter 5

## System Implementation

## Implementation

The implementation phase involves developing and integrating the various components of the web application to create a functional and user-friendly online platform. This phase will encompass fronted and backend development, as well as database setup and integration. Additionally, it will involve rigorous testing to ensure the system's functionality, security, and performance meets the specified requirements.

## Tools and Technologies

In the development of a Web application like this, tool selection is critical. The tools used in the development are listed below.

* HTML
* CSS
* JavaScript
* React JS
* Node JS
* Express JS
* MySQL

## Fronted

For the fronted development of the Event Management web application, we will be using React JS, a popular JavaScript library for building user interfaces. React JS provides a component-based architecture that facilitates the creation of interactive and dynamic user interfaces. With its virtual DOM and efficient rendering, React JS ensures a smooth user experience. The fronted will be responsible for displaying the user interface, handling user interactions, and communicating with the back-end API.

## Backend

The back end of the Web Application will be developed using Node.js and Express.js. Node.js is a runtime environment that allows us to build server-side applications using JavaScript. Express.js is a web application framework for Node.js that simplifies the process of building robust and scalable API s. The backend will handle user authentication, payment processing through the Stripe payment gateway, document application workflows, and interaction with the database.

## Database

The database for an event management web application is the cornerstone of efficient data organization and retrieval. It comprises several interlinked tables, each housing critical information. The Event table holds essential event details like event name, description, dates, location, and references the Organizer table. The Organizer table stores organizer information such as name, contact details, and links back to their respective events. Attendee table contains attendee information and their associated events, facilitating event attendance tracking. The Venue table holds venue data, including venue name and address, associated with events. The Registration table links events and attendees, tracking registrations and their status. The Task table manages event-related tasks, monitoring their progress and deadlines. Lastly, the Sponsor table stores sponsor details and links them to specific events, supporting event sponsorship management. Altogether, this database structure forms a robust foundation, enabling the event management web application to efficiently manage event-related data, streamline operations, and enhance the overall event management process

## Testing and Quality Assurance

Testing and quality assurance for an event management web application involves a meticulous process to ensure its functionality and performance meet predefined standards. Initially, the requirements and scope of the application are analyzed thoroughly. Functional testing is then conducted to validate features such as event creation, attendee registration, and task management. Usability testing ensures an intuitive user interface and a positive user experience. Performance testing evaluates the application's response under varying conditions like load, stress, and scalability to ensure optimal performance. Security testing identifies and mitigates vulnerabilities to safeguard sensitive data. Compatibility testing verifies the application's functionality across multiple devices and browsers. Integration and regression testing ensure seamless interaction of different components and prevent new updates from disrupting existing features. User acceptance testing involves real users to validate that the application meets their needs and expectations. Automated testing may also be employed to streamline repetitive testing procedures. This rigorous testing and quality assurance process culminates in a robust and reliable event management web application that provides a smooth and efficient event management experience for users

## Deployment and User Training

Deployment of the event management web application entails setting up the necessary infrastructure, configuring databases, hosting the application, and optimizing its performance and security. This involves tasks like domain configuration and SSL setup for a secure environment. On the other hand, user training involves preparing training materials, scheduling and conducting training sessions, providing hands-on practice, addressing queries, and ensuring users are proficient in utilizing the application effectively. Ongoing support and feedback mechanisms are crucial components of successful user training to enhance application adoption and usage.

## Maintenance and Updates

Maintenance and updates of the event management web application are ongoing processes vital for its functionality and relevance. Maintenance involves monitoring performance, ensuring data integrity, and addressing bugs or issues promptly to guarantee a seamless user experience. Regular backups and security audits are essential aspects to safeguard against data loss and maintain system security. Furthermore, user feedback and analytics are valuable sources for identifying areas of improvement and user needs. Updates involve implementing new features, improving existing functionalities, and applying security patches. Communication with users regarding updates, providing clear release notes, and offering supports during transitions are crucial to ensure a smooth and successful update process. Continuous maintenance and updates are fundamental to keeping the event management web application efficient, secure, and aligned with user requirements and technological advancements.

# Chapter 6

## SYSTEM TESTING AND EVALUTION

System testing and evaluation of an event management web application are pivotal to ensuring its reliability, functionality, and user satisfaction. Functional testing is performed to validate that all features and functions, including event creation, attendee registration, and task management, operate correctly. Usability testing focuses on the application's user interface, ensuring it is intuitive, easy to navigate, and provides a positive user experience. Performance testing involves assessing the application's responsiveness and stability under various conditions, ensuring it can handle different levels of usage effectively. Security testing is critical to identifying vulnerabilities and ensuring data privacy and protection measures are in place. Compatibility testing guarantees the application works seamlessly across different devices, browsers, and platforms. Integration and regression testing ensure that the application's components interact smoothly and that new updates do not disrupt existing functionalities. User acceptance testing involves real users to validate that the application meets their expectations and requirements. Collecting and incorporating feedback from these tests ensures the event management web application is robust, reliable, and aligned with user needs, providing an optimal platform for efficient event management.

# Chapter 7

## CONCLUSION

The proposed of our event management web application stands as a powerful tool designed to revolutionize event organization and attendee experiences. The comprehensive features and user-friendly interface cater to event organizers, attendees, and administrators alike. From seamless event creation and attendee registration to efficient payment processing and insightful analytics, our application streamlines the event lifecycle. The integration of secure payment gateways ensures trust and reliability, while the dynamic use cases showcase the app's versatility. Emphasizing user engagement and satisfaction, our event management web application represents a leap forward in event coordination, promising enhanced efficiency and an enriched event experience for all stakeholders.

As we continue to innovate and evolve, our commitment to providing a seamless and dynamic platform remains unwavering. Future updates will further enhance the application, incorporating advanced features based on user feedback and industry trends. We strive to redefine event management, ensuring that every event, big or small, is a resounding success. Join us on this journey of transformation and stay tuned for an even more exceptional event management experience.

## FUTURE WORK

Looking forward, our event management web application is primed for ongoing improvements and expansions, aiming to heighten user satisfaction and extend its functionality. Envisioned future work includes.

* 1. **Enhanced User Interface**: Innovating the interface to be more intuitive and visually appealing, promoting seamless navigation and optimal event planning.
  2. **Integration with Various Payment Gateways**: Broadening payment alternatives by integrating with multiple payment gateways, ensuring secure and flexible transaction processes for event attendees.
  3. **Seamless Digital Wallet Integration**: Incorporating digital wallets seamlessly for swift and convenient payment transactions, enhancing the overall user experience in event registration and purchases.
  4. **Advanced Notification System:** Implementing an advanced notification system that provides real-time alerts and updates to keep attendees informed about event activities and changes.
  5. **Optimized Event Analytics:** Enhancing the analytical capabilities to provide valuable insights into event performance, attendee engagement, and areas for improvement.
  6. **Social Media Integration:** Integrating social media platforms to facilitate event promotion, allowing attendees to easily share and invite others to events.

Top of Form

# REFERENCES

Includes all references: articles, media facts, books, reports, regulations, internet articles, papers that you referenced from the text. Use the APA style of referencing ([www.apastyle.org](http://www.apastyle.org)).

Or any other standard format as per requirement of the subject.

# APPENDICES

# Appendix A

USER MANUALS

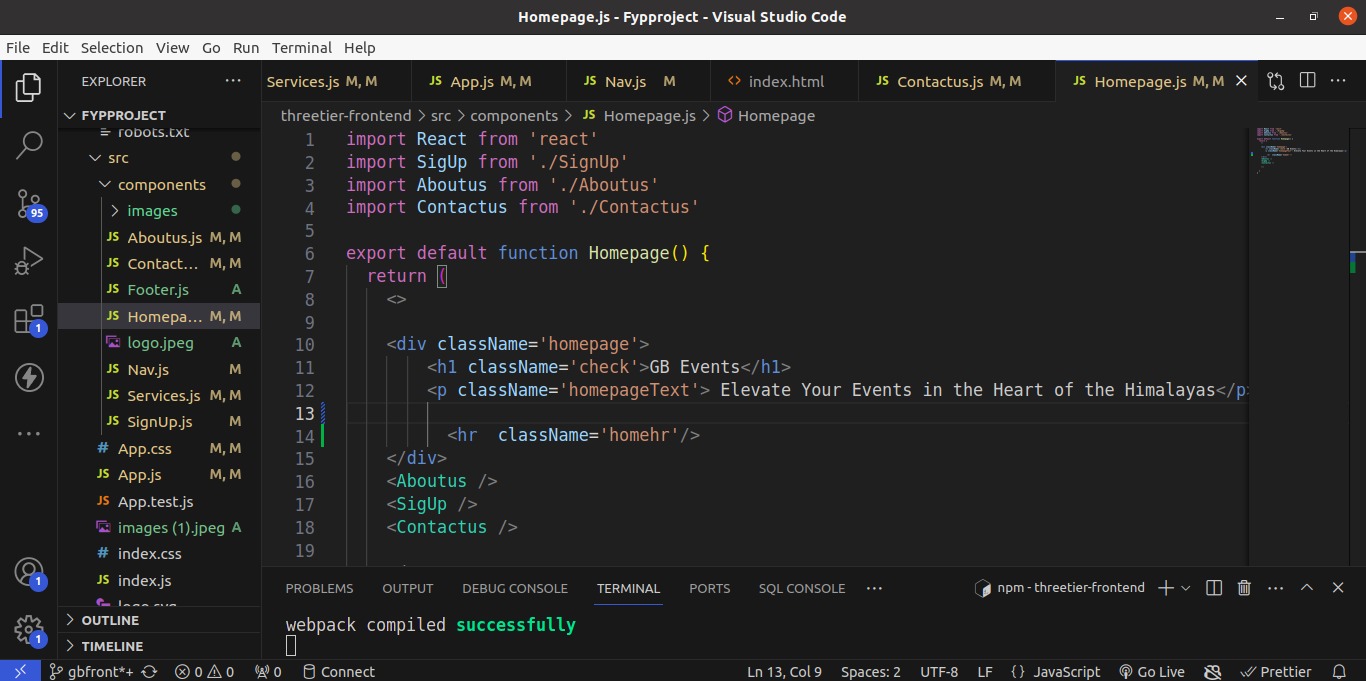
.

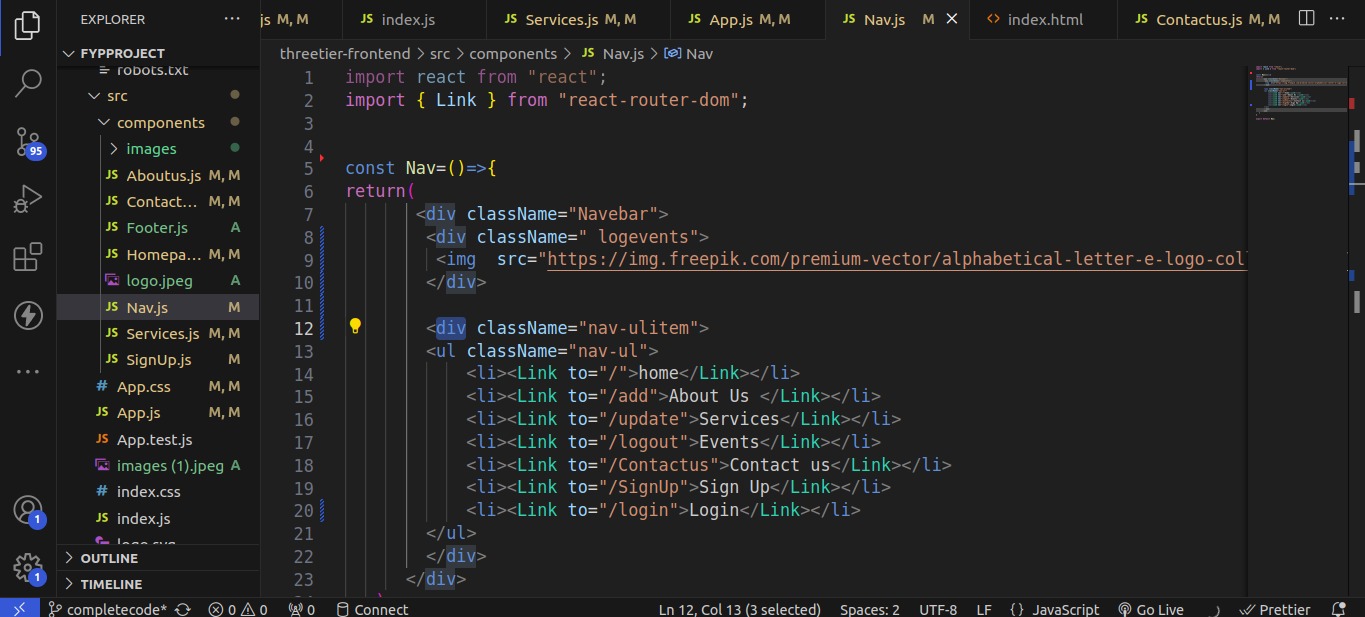
|  |  |
| --- | --- |
|  | To use the system properly user must have active internet connection |
|  | User can use any internet browser on any Device. |
|  | Do not share your banking credentials with anyone you do not trust. |
|  | Any digital payment gateway takes a while to load, so wait patiently. |
|  | Don’ t pay your fee using anyone’s device |
|  | Your credentials and data will not be shared or accessed by anyone. |

# Appendix B

SOURCE CODE

## Main frontend code:





It does not contain all the source code but just a sample code

## Main Backend Code: ­­

