A logo with blue text

Description automatically generated

**EVENT CRAFT**

**PROJECT SUPERVISOR**

**Mr. Syed Farooq Zaidi**

**PROJECT CO-SUPERVISOR**

**Mr. Fahad Hussain**

**PROJECT TEAM**

**Qazi Zain K20-1038**

**Zain ul Abdin K20-0185**

**Ali Kahfulwara K19-1066**

Submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science.

FAST SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES KARACHI CAMPUS

May 2024

|  |  |
| --- | --- |
| Project Supervisor | Mr. Syed Farooq Zaidi |
| Project Team | Qazi Zain K20-1038  Zain ul Abdin K20-1672  Ali Kahfulwara K19-1066 |
| Submission Date | Dec 12, 2024 |

Mr. Syed Farooq Zaidi

**Supervisor**

Dr. Atif Tahir

**Head of Department**

FAST SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES KARACHI CAMPUS

**Acknowledgement:**

All the thanks to the supervisor Mr. Syed Farooq Zaidi and co- supervisor Mr. Fahad Hussain, the members of the of the project Qazi Zain, Zain UL Abdin, Ali kahfulwara, and people who cooperated in providing the data for the project. All the mentioned people have contributed to the project and helped us on this journey. Without their help we could not have done this.

Table of Contents

[Abstract 5](#_heading=h.30j0zll)

[Introduction 6](#_heading=h.1fob9te)

[Related Work 7](#_heading=h.3znysh7)

[Requirements 8](#_heading=h.tyjcwt)

[Use Cases 8](#_heading=h.3dy6vkm)

[Registration 8](#_heading=h.1t3h5sf)

[Log-In 9](#_heading=h.4d34og8)

[Profile Management 10](#_heading=h.2s8eyo1)

[Add new Services. 11](#_heading=h.17dp8vu)

[Event Progress 12](#_heading=h.lnxbz9)

[E-Card Generation 13](#_heading=h.1ksv4uv)

[Invite Member 14](#_heading=h.2jxsxqh)

[Initialization budget 15](#_heading=h.z337ya)

[Event Progress 16](#_heading=h.3j2qqm3)

[Non-Functional Requirements are given below: 17](#_heading=h.1y810tw)

[Design 18](#_heading=h.2bn6wsx)

[UserApp 19](#_heading=h.qsh70q)

[Service Provider 20](#_heading=h.3as4poj)

[Implementation 23](#_heading=h.1pxezwc)

[Test cases 24](#_heading=h.49x2ik5)

[Conclusion 25](#_heading=h.2p2csry)

[References 26](#_heading=h.147n2zr)

# Abstract

EventCraft is a platform created to simplify the event planning experience by connecting customers with service providers, specifically tailored to their budget and unique event needs. In an industry often plagued by fragmented services and a lack of transparent pricing, EventCraft stands out by offering a cohesive solution that empowers users to plan their events with ease and confidence within budget and desired time. Ensuring that each recommendation aligns with the user’s style, preferences, and financial goals. This personalized approach enhances decision-making, allowing users to efficiently explore, compare, and book services in one centralized platform. Designed to support both individuals and organizations, EventCraft addresses the common challenges of event planning, such as managing budgets, discovering reliable vendors, and avoiding unexpected costs. With a strong focus on user experience, EventCraft simplifies every stage of the planning process, helping users bring their vision to life while staying within their budget, all in a seamless, efficient manner. Furthermore, For more Details you can visit our github repo [https://github.com/MirzaZain110/FYP-1] .

# Introduction

EventCraft is a user-centric platform designed to streamline and revolutionize the event planning process. It connects clients and event service providers through a comprehensive digital platform, enabling seamless collaboration tailored to the unique needs of each user. EventCraft ensures both clients and service providers can efficiently reach agreements, select services, and achieve their event goals without stress.

Clients can easily register on the platform by providing basic details such as their name, contact information, event type, location, and budget constraints. The platform’s intelligent recommendation system leverages these inputs to suggest suitable vendors and venues, ensuring personalized and relevant options. On the other hand, service providers can also register by providing their service offerings, pricing details, and past client reviews, which enhance their credibility and visibility to potential customers.

To further simplify the process, EventCraft incorporates features such as a budget management tool that helps users estimate and control costs, ensuring transparency and confidence in decision-making. By integrating reviews, ratings, and customizable options, the platform empowers users to select the most reliable and suitable vendors for their events.

EventCraft addresses common challenges in event planning, such as fragmented services, time inefficiencies, and unexpected costs, with an emphasis on delivering an exceptional user experience. Through its seamless and efficient approach, EventCraft ensures that clients can transform their event visions into reality, all while staying within budget and timelines. This personalized platform is equally valuable for individuals and organizations, setting a new standard for event planning efficiency and satisfaction.

For Citation we use the Zotero Chrome extension. Which is easy to use and workable to manage and save the citation.

# Related Work

Here is the list of the platforms that provided similar services which are:

* TopEvent
* MyWed
* EventBrite

**TopEvent:** This application provides features for event planning and management but lacks the option for selecting vendors or direct service providers. Users can plan events but have to manage vendor selection independently.

**MyWed:** This platform focuses on wedding planning and services, helping clients organize wedding events. However, it does not offer direct access to purchase services or materials for events through an integrated vendor system.

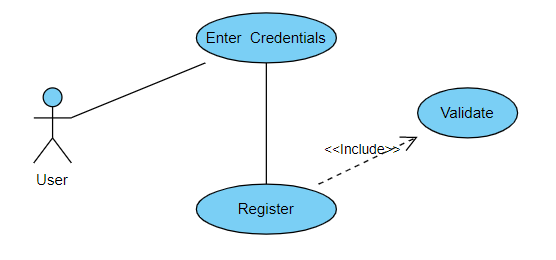
**EventBrite:** EventBrite is a well-known international platform for managing and organizing events. It allows users to create, manage, and promote events but does not provide a marketplace for material or vendor selection. Users need to rely on external sources for these needs.

# Requirements

Functional requirements and the diagrams are given below:

## Use Cases

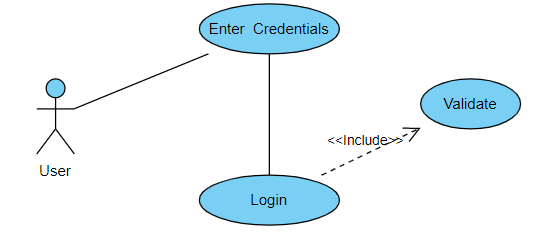
### Registration

**

**Figure 1: Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC1 : Registration*** | | | | |
| ***Use case Id:*** | | *Uc1* | | |
| ***Actors:***  *Users and Service providers* | | | | |
| ***Feature:*** *Sign Up* | | | | |
| ***Pre-condition:*** | | *If use is not register.* | | |
| ***Scenarios:*** *Before using the application, users and service providers have to registered.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *First of all the user must click on the sign-up to enter their details and register* | | | *Redirects the user to the registration form.* |
| ***2.*** | *Enter email* | | | *Email or roll number’s format will be checked.* |
| ***3.*** | *Enter password* | | | *Password format will be checked* |
| ***4.*** | *Enter Name* | | | *Checks if the field is not empty.* |
| ***5.*** | *Click the button to register* | | | *Data will be verified and the user will be redirected to a specific interface upon successful creation.* |
| ***Alternate Scenarios:*** | | | | |
| ***1:*** *Users use wrong or incorrect email to login.*    ***2:*** *User not entered strong password .* | | | | |
| ***Post Conditions:*** *Successful creation of an account which can be used to login to the system.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***Use Case Cross referenced*** | | | *-* | |

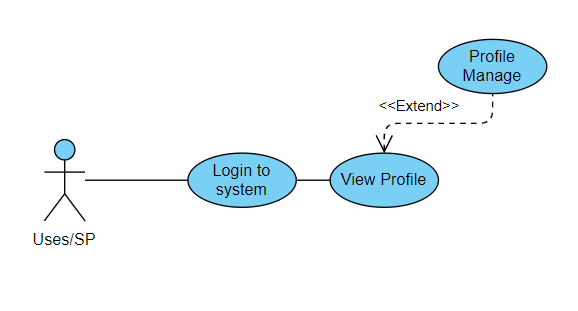
### Log-In



**Figure 2: Login Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC2 : Log-In to system*** | | | | |
| ***Use case Id:*** | | *Uc2* | | |
| ***Actors:***  *users and Service providers* | | | | |
| ***Feature:*** *Log-In* | | | | |
| ***Pre-condition:*** | | *Users and Service Provider should be registered on database before to Log-In.* | | |
| ***Scenarios:*** *Considering the user has registered already, he/she logs in to use the web application or mobile applications.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *First of all the user must click on the log-in to enter their details and process.* | | | *Redirects the user to the log-in page.* |
| ***2.*** | *Enter email address or roll number* | | | *Check if the email is correct and matches.* |
| ***3.*** | *Enter password* | | | *Check if the password matches.* |
| ***4.*** | *Click on the button to log-in* | | | *Authenticate and the user and proceed with the login process.* |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *Users or service provider login failed, then they should reentered password and email.* | | | | |
| ***Post Conditions:*** *Successful authentication will lead the user to log into the system and use services.* | | | | |
| ***Step#*** | ***Description*** | | | |
|  | *Access to use the system will be granted to the user according to its type.* | | | |
|  |  | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *Uc1* | |

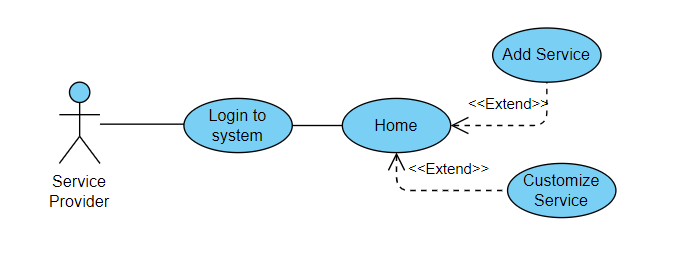
### Profile Management



**Figure 3: Profile Management Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC3: Manage Profile*** | | | | |
| ***Use case Id:*** | | *Uc3* | | |
| ***Actors:***  *users and Service providers* | | | | |
| ***Feature:*** *Manager Profile* | | | | |
| ***Pre-condition:*** | | *Users and Service Provider should be Log-In to the system.* | | |
| ***Scenarios:*** *user upload wrong data on system. He/She needed to be update that data.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Enter email password* | | | *Login to the system* |
| ***2.*** | *Click on profile* | | | *Update and manage profile* |
|  |  | | |  |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *Users or service provider login failed, then they should reentered password and email.* | | | | |
| ***Post Conditions:*** *Successful change the profile of user and service providers .* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *User's profile information is updated with the changes made.* | | | |
| ***2*** | *The system confirms the successful completion of the profile management task and displays a notification message to the user* | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *Uc2* | |

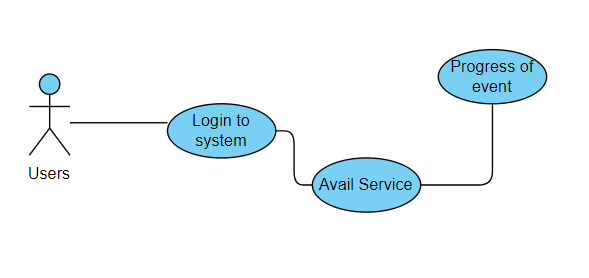
### Add new Services.



**Figure 4: Add new Services Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC4: Add new Services*** | | | | |
| ***Use case Id:*** | | *Uc4* | | |
| ***Actors:***  *Service providers* | | | | |
| ***Feature:*** *Add new Services* | | | | |
| ***Pre-condition:*** | | *Service Provider should be Log-In to the system.* | | |
| ***Scenarios:*** *Services provider want to add new services.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Enter email password* | | | *Login to the system* |
| ***2.*** | *Click on Add service button* | | | *Add new Services* |
|  |  | | |  |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *Service not added.* | | | | |
| ***Post Conditions:*** *Successful change the profile of user and service providers.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *Service will be available to users.* | | | |
| ***2*** | *Successful added service* | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *Uc3* | |

### Event Progress



**Figure 6:** Event Progress**Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC6: Event Progress*** | | | | |
| ***Use case Id:*** | | *Uc6* | | |
| ***Actors:***  *users* | | | | |
| ***Feature:*** *Event Progress* | | | | |
| ***Pre-condition:*** | | *Service Active by user* | | |
| ***Scenarios:*** *Services provider want to add new services.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Enter email password* | | | *Login to the system* |
| ***2.*** | *Click on Add service button* | | | *Add new Services* |
| ***3*** | *Open dashboard* | | | *See Progress* |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *. If the service provider tries to add a service with incomplete or invalid information, the system displays error messages prompting the provider to correct the entries.*   1. *Service is not started properly.* | | | | |
| ***Post Conditions:*** *View Services service providers.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *Progress will be seen by users and service provider.* | | | |
| ***2*** |  | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *Uc3,UC4 or UC5* | |

### 

### E-Card Generation

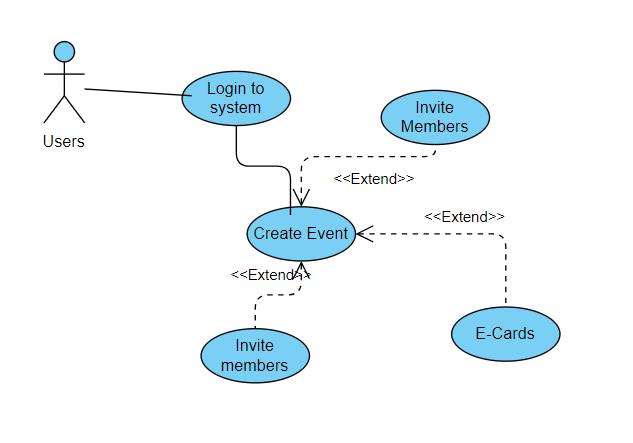
*A diagram of a function

Description automatically generated*

**Figure 7:** E-Card Generation**Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC7: E-cards*** | | | | |
| ***Use case Id:*** | | *UC7* | | |
| ***Actors:***  *users* | | | | |
| ***Feature:*** *E-cards* | | | | |
| ***Pre-condition:*** | | *Users should be login to the system.* | | |
| ***Scenarios:*** *user want to create e-cards and sent it to their guest.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Enter email password* | | | *Login to the system* |
| ***2.*** | *Click on generate cards* | | | *Cards generated with multiples templates* |
| ***3*** | *Sent to guest* | | | *Send to guest* |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *.* | | | | |
| ***Post Conditions:*** *E card generated and sended to the guests.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *Generate the card* | | | |
| ***2*** | *Select specific card* | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *UC2* | |

### Invite Member

**

**Figure 9:** Invite Member**Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC11: Invite helper*** | | | | |
| ***Use case Id:*** | | *Uc11* | | |
| ***Actors:***  *user and Guests* | | | | |
| ***Feature:*** *Invite Member (email)* | | | | |
| ***Pre-condition:*** | | *Contact number should be added. The event organizer must be logged in to the system and the event must be created and in the planning stage.* | | |
| ***Scenarios:*** *User want to help to create event or share their tasks* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Event organizer navigates to the event management section of the system* | | | *The system presents options for managing the selected event* |
| ***2.*** |  | | |  |
| ***3*** |  | | |  |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *.*  *If the event organizer enters an invalid email address or phone number, the system displays an error message prompting the organizer to correct the entry* | | | | |
| ***Post Conditions:*** *Now member will help to create event.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *Invitations are successfully sent to the specified guests.* | | | |
| ***2*** |  | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *UC2, UC10* | |

### Initialization budget

*A diagram of a process

Description automatically generated*

**Figure 10:** Initialization budget**Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC12: Initialization budget*** | | | | |
| ***Use case Id:*** | | *Uc12* | | |
| ***Actors:***  *users* | | | | |
| ***Feature:*** *Make budget* | | | | |
| ***Pre-condition:*** | | *The user must be logged in to the system.*  *The user must have created an event* | | |
| ***Scenarios:*** *Users want to make not to excite budget.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *User selects the event for which are under budget* | | | *The system presents progress bar which show budget and if excite it will become red* |
| ***2.*** |  | | |  |
| ***3*** |  | | |  |
|  |  | | |  |
| ***Alternate Scenarios: UC2, UC10*** | | | | |
| ***1 :*** *.*  *If the user tries to initialize the budget with an invalid amount (e.g., negative value or non-numeric input), the system displays an error message prompting the user to correct the entry.* | | | | |
| ***Post Conditions:*** *Maintain the event within budget* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *The budget for the event is successfully initialized with the specified amount.* | | | |
| ***2*** | *The user can start managing the budget by adding income and expenses.* | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *UC2, UC10* | |

### Event Progress

*A diagram of a system

Description automatically generated*

**Figure 12:** Event Progress**Use case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***UC14: Event Progress*** | | | | |
| ***Use case Id:*** | | *Uc14* | | |
| ***Actors:***  *Service providers and users* | | | | |
| ***Feature:*** *Event Progress* | | | | |
| ***Pre-condition:*** | | *Service Active by user* | | |
| ***Scenarios:*** *Services provider want to add new services.* | | | | |
| ***Step#*** | ***Action*** | | | ***Software Reaction*** |
| ***1.*** | *Enter email password* | | | *Login to the system* |
| ***2.*** | *Click on Add service button* | | | *Add new Services* |
| ***3*** | *Open dashboard* | | | *See Progress* |
|  |  | | |  |
| ***Alternate Scenarios:*** | | | | |
| ***1 :*** *. If the service provider tries to add a service with incomplete or invalid information, the system displays error messages prompting the provider to correct the entries.*   1. *Service is not started properly.* | | | | |
| ***Post Conditions:*** *View Services service providers.* | | | | |
| ***Step#*** | ***Description*** | | | |
| ***1*** | *Progress will be seen by users and service provider.* | | | |
| ***2*** |  | | | |
|  |  | | | |
| ***Use Case Cross referenced*** | | | *Uc3,UC4 or UC5* | |

## Non-Functional Requirements are given below:

**Operating Environment**

The project requires the users to only have a smartphone, computer, or laptop without any constraint on the operating system as the application will have the ability to run on Android, Windows or iOS devices and any web browser. However, users must have access to the internet as the application will not run without a stable WIFI connection.

**System Constraints**

The system has been developed by the project members of the university due to lack of the whole software team system is subject to some of the constraints. The system will not have any payment information. The system will not enable users to pay for a service. The system has only the web app and mobile app available for this time.

**Assumptions & Dependencies**

* We assume that the users off our website and mobile:
* Have a stable internet connection.
* Understand English
* Users of this website are local vendors and normal people.

**System Dependencies**:

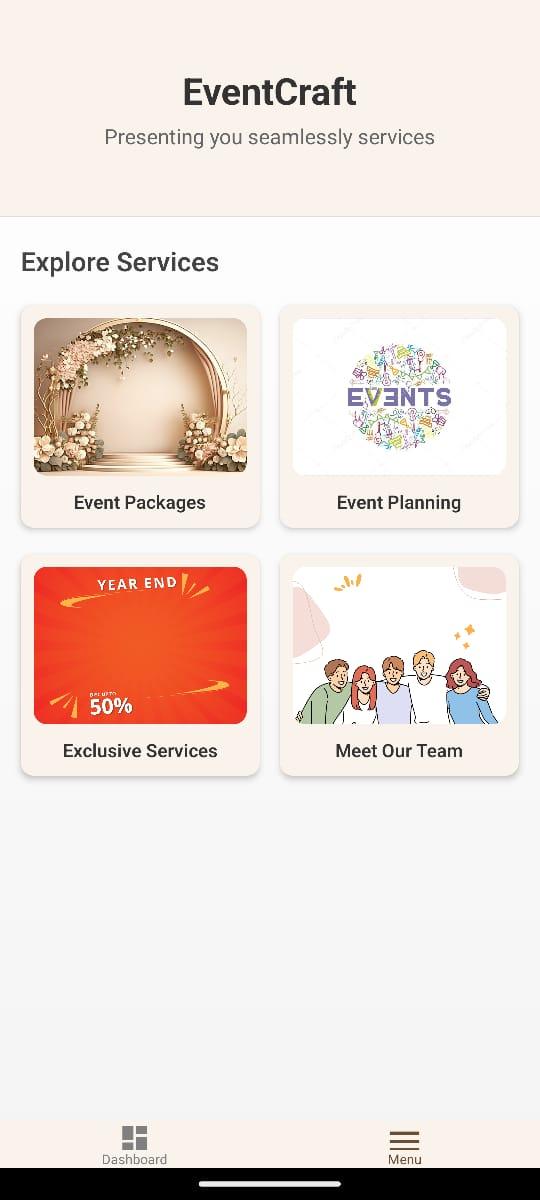
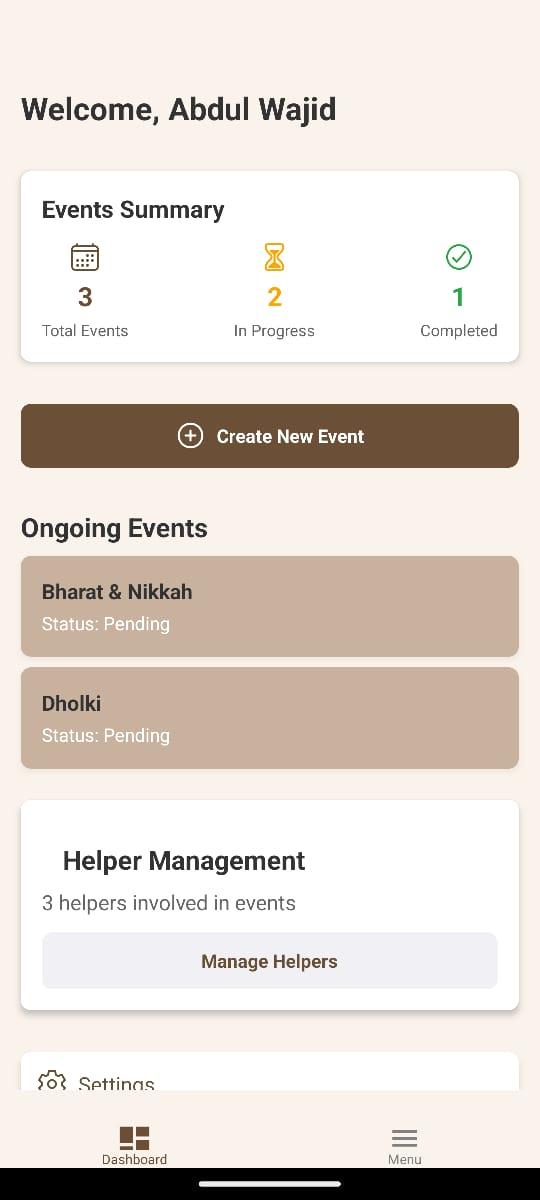
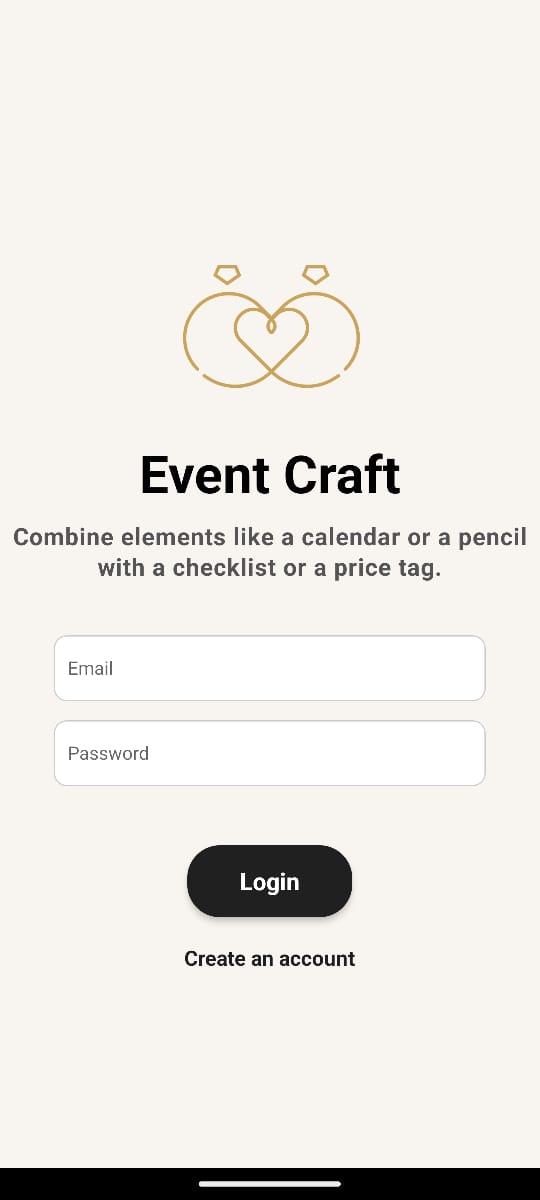
* Our system requires a hosting service that can connect to the servers.
* System depends on a stable internet connection.
* Requires an active database connection to save the changes mad

# Design

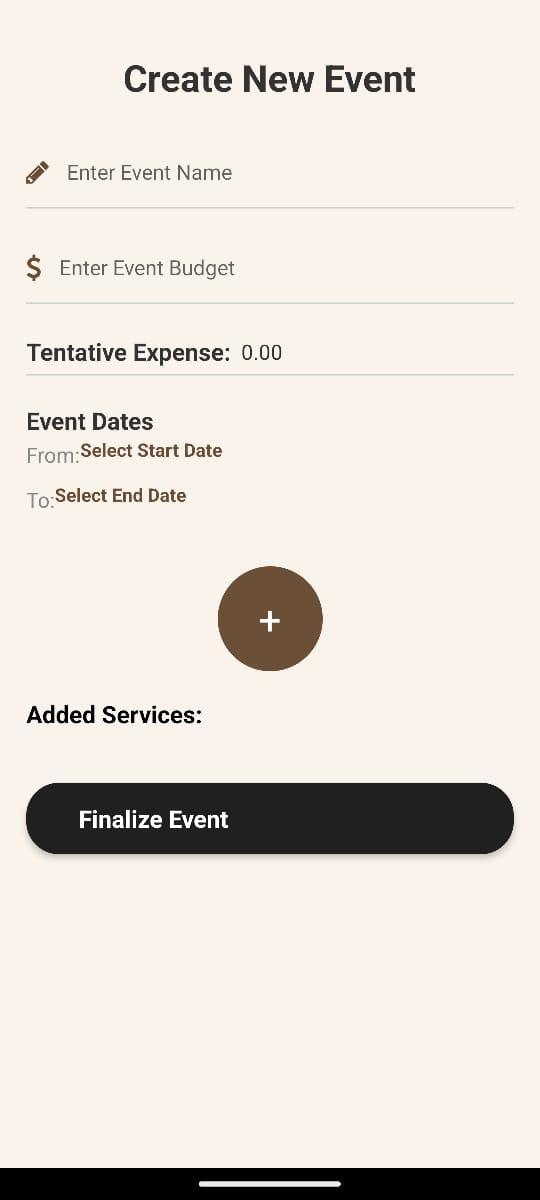
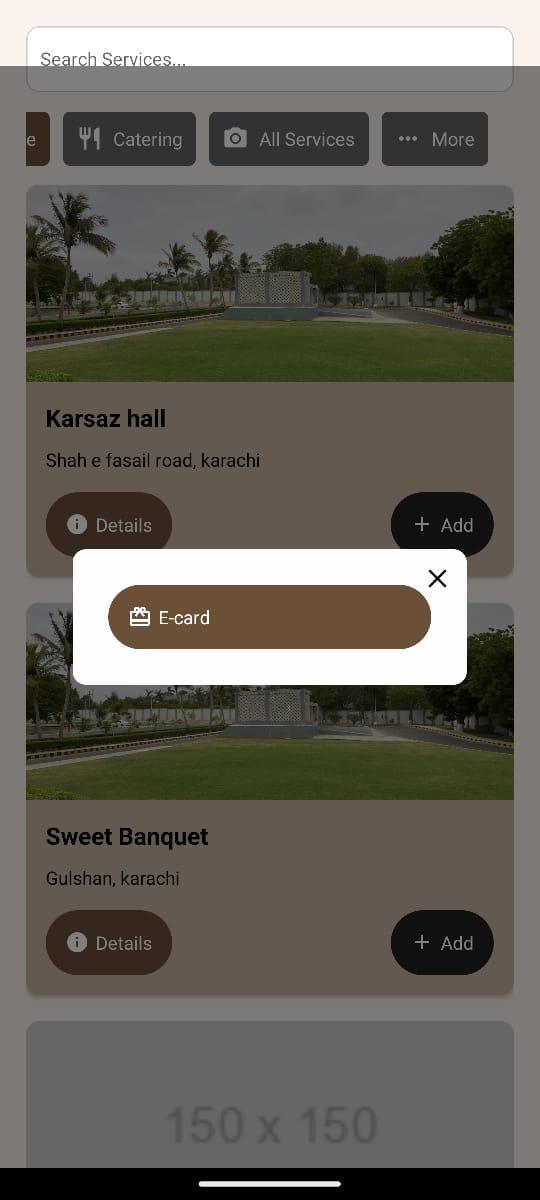
Our system is specifically designed to address the common challenges in event planning by connecting users and service providers on a unified platform. The goal is to create an intuitive, interactive, and user-friendly platform that simplifies the process for both clients and service providers. We have ensured seamless database connectivity to handle complex relationships between users, services, and events.

The user interface (UI) and user experience (UX) design are carefully crafted to deliver an aesthetically appealing and consistent layout, making the platform easy to navigate and engaging for all users. Input validation mechanisms and real-time feedback have been implemented to ensure data accuracy and enhance usability. A relational database is employed to efficiently manage and showcase intricate data relationships, enabling users to make informed decisions. Our platform is built to streamline event planning, saving time and reducing stress while providing an exceptional experience for all stakeholders.

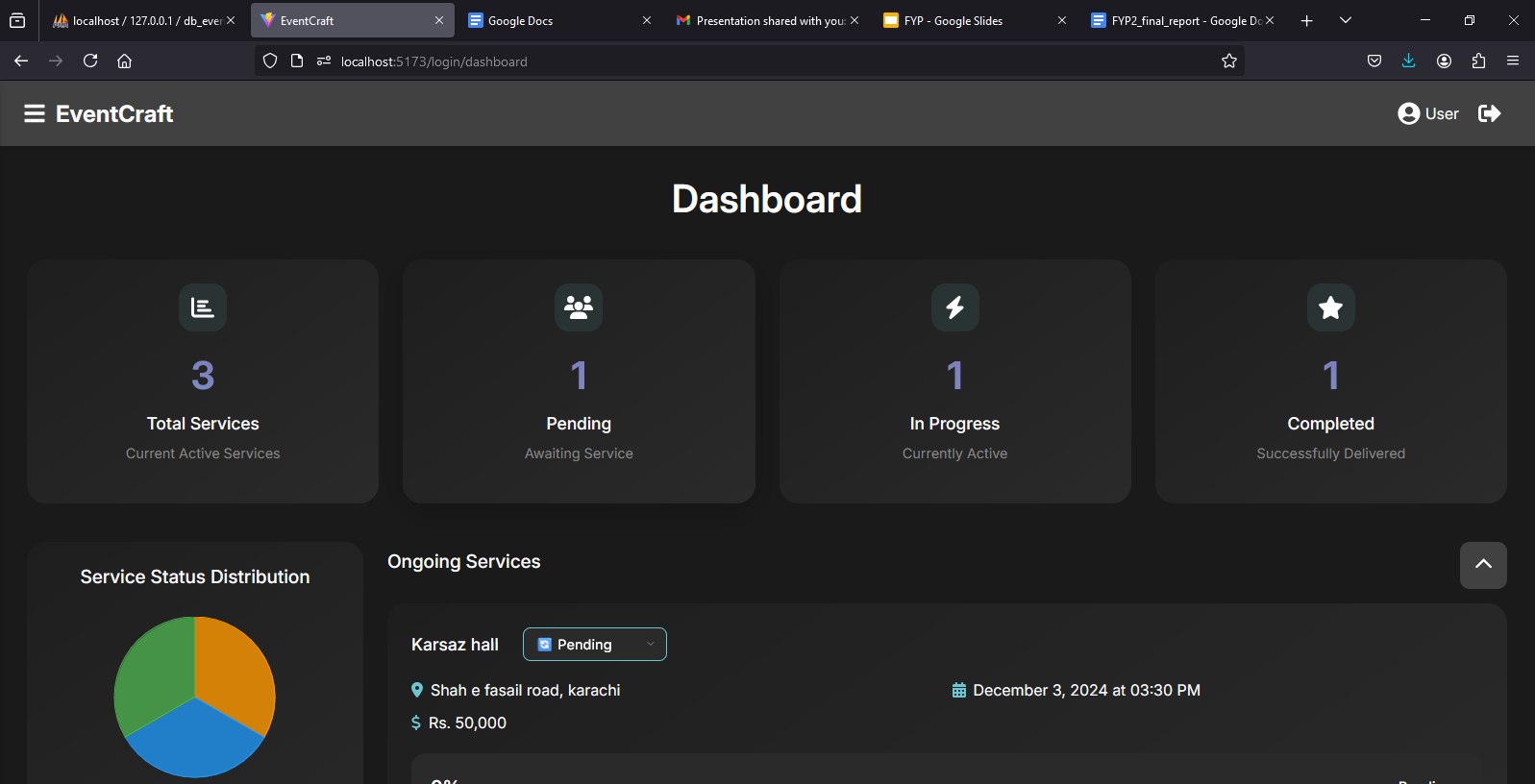
## UserApp

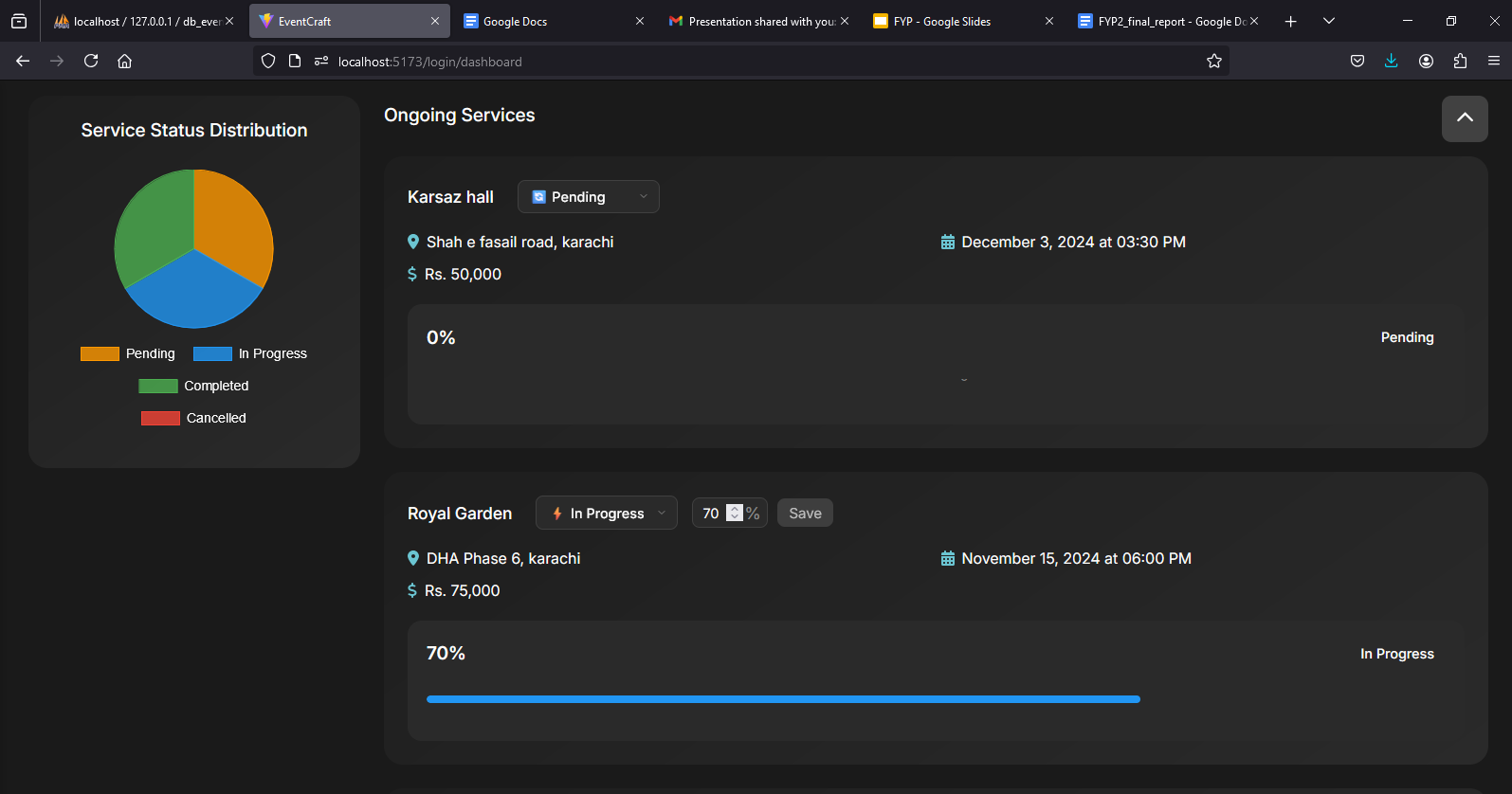
A screenshot of a cellphone

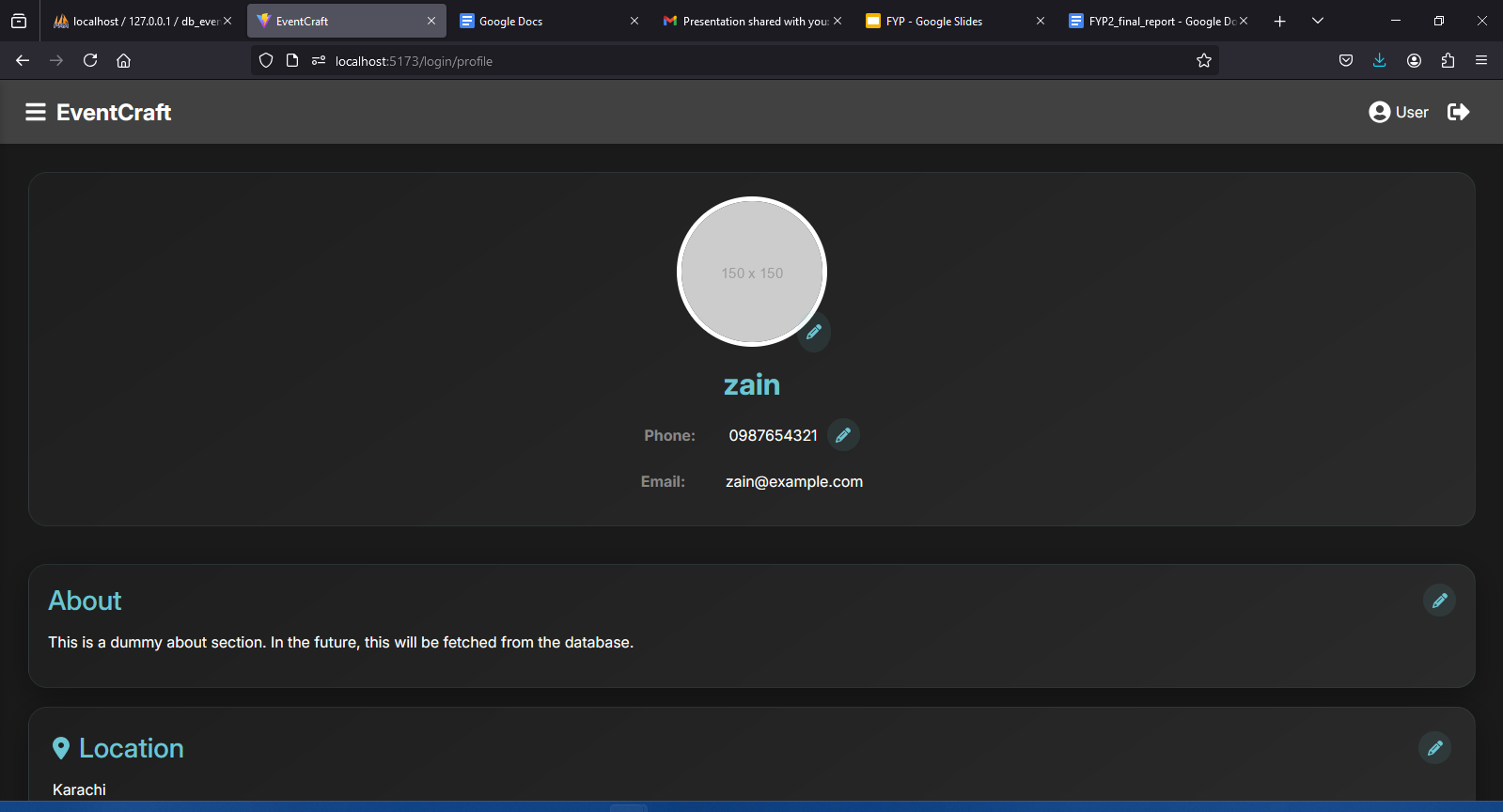
Description automatically generated

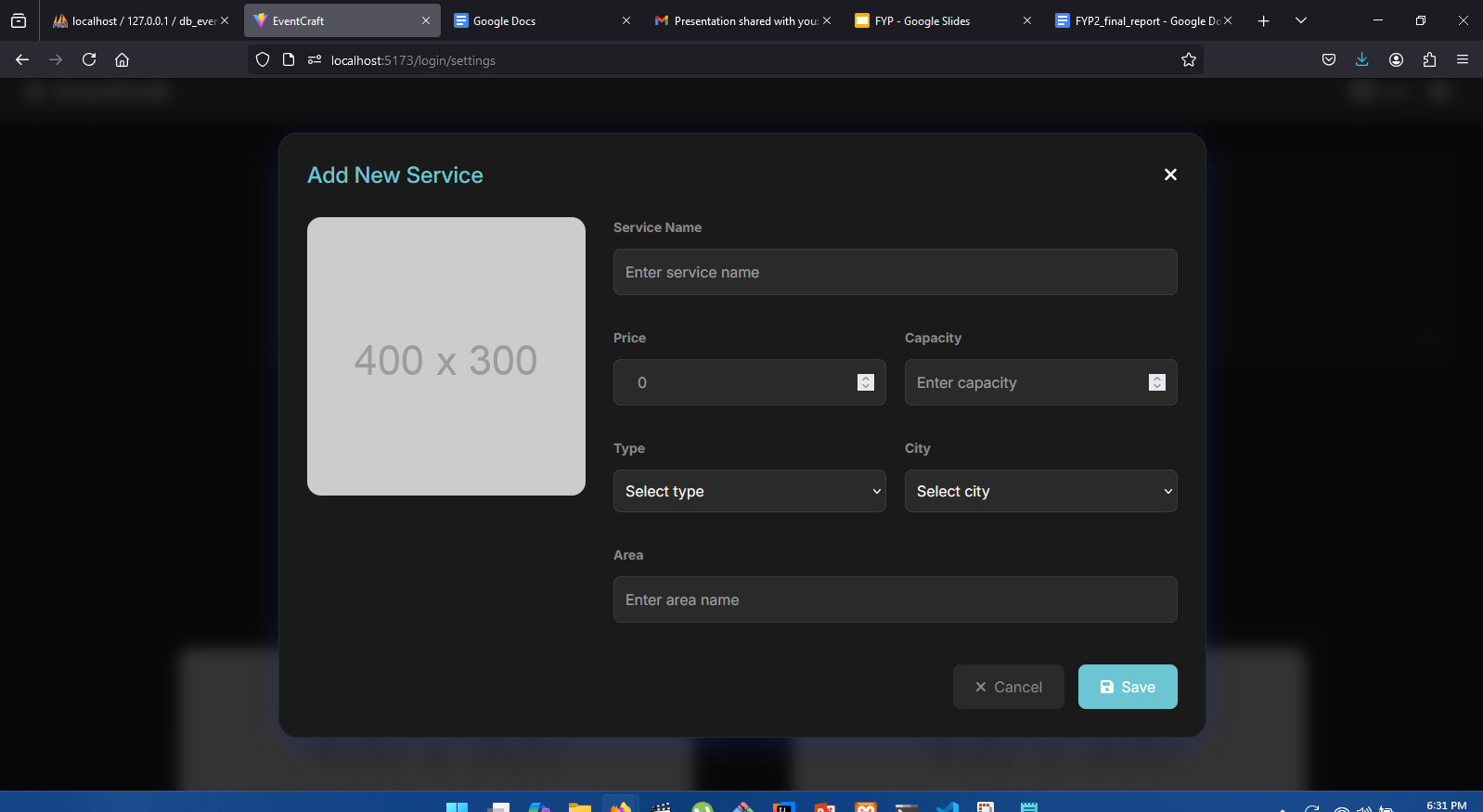


## Service Provider









# Implementation

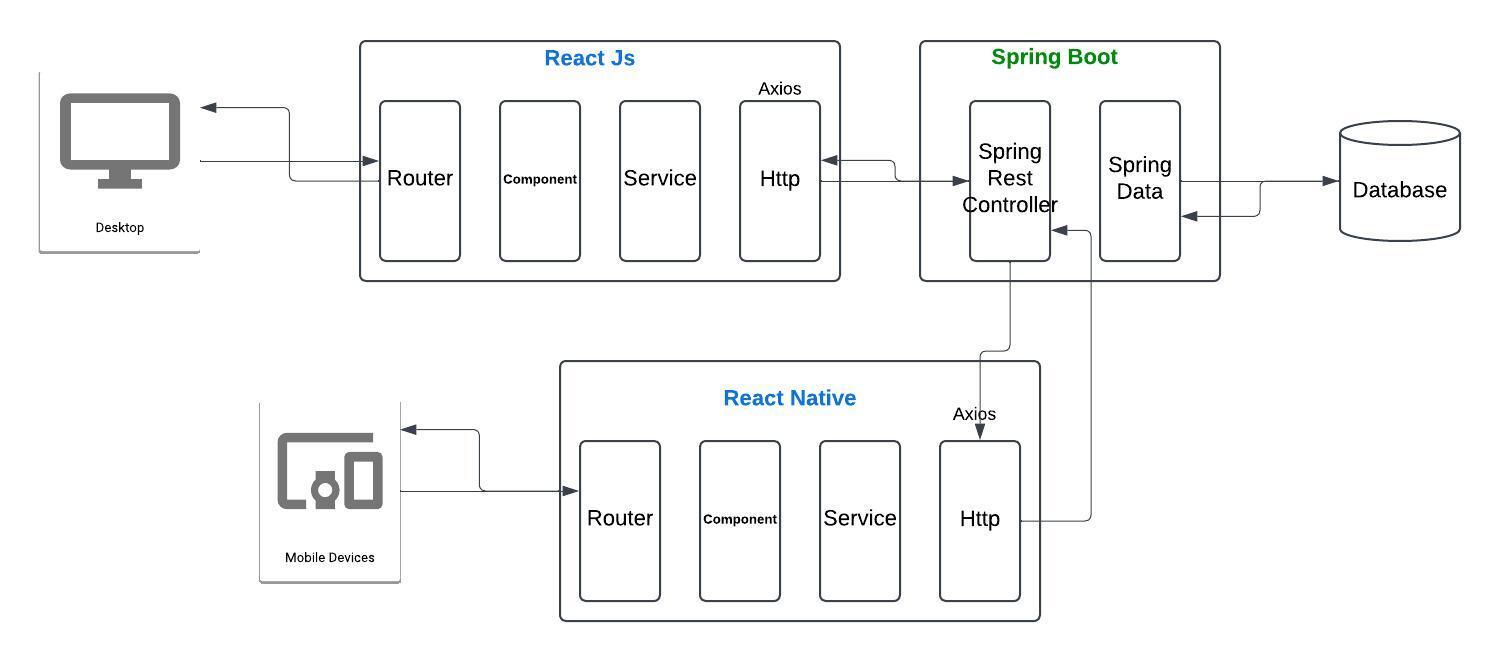
Our EventCraft application is based on a three-tier architecture, where the system is organized into three logical and physical layers. The application is compatible with any operating system that supports internet connectivity. The technology stack for the application includes React Native, Spring Boot, Hibernate, and MySQL, enabling seamless interaction between the layers.

In this architecture, React Native represents the Presentation Layer, where all user interfaces are defined. This framework simplifies frontend development and allows for building user-friendly and interactive interfaces for both web and mobile platforms. The Business Logic Layer is implemented using Spring Boot, which provides a robust server-side framework, coupled with Hibernate for efficient ORM (Object-Relational Mapping). APIs are used for seamless interaction between the frontend and backend layers.

For data storage and management, MySQL is used as the relational database. It ensures reliable and efficient handling of complex relationships between users, events, and service providers. The Data Access Layer leverages Hibernate to communicate with the database, ensuring secure and optimized data retrieval and storage.

In this architecture, the presentation layer communicates with the business logic layer, which in turn interacts with the data access layer to fetch or store the necessary information. This structure ensures scalability, maintainability, and seamless performance, making EventCraft a reliable and efficient platform for event planning and management.

Following diagram shows the implementation



**Figure 9: System architecture Diagram**

# Test cases

|  |  |  |
| --- | --- | --- |
| **Test Cases** | **Names** | **Expected Result** |
| TC1 | User and service provider can sign up by entering details. | User and service provider respective dashboards will be displayed after successful registration |
| TC2 | User and service provider can log in by entering email and password. | *Result:* User and service provider respective dashboards will be displayed. |
| TC3 | Service provider can add service listings. | Listings will be updated and displayed successfully. |
| TC4 | Service provider can accept or reject booking requests. | Booking requests will be either confirmed or removed based on the service provider's action. |
| *TC5* | Service provider can update the status of booked services. | Service status will be updated and displayed to the user. |
| TC6 | User and service provider can view available services based on their event type. | All relevant services for the selected event type will be displayed. |
| TC7 | User can view the list of available service providers. | A comprehensive list of service providers will be displayed, filtered by user preferences. |
| TC8 | User can view the progress of booked services. | Progress updates for each service will be displayed on the user dashboard. |
| TC9 | User can book a service by selecting from the available listings | Booking confirmation will be displayed, and details will be saved to the user's dashboard. |
| TC10 | User and service provider can provide feedback for completed events. | Feedback will be saved and visible on respective profiles. |

**Table 1: Test cases**

# Conclusion

We are proud to conclude that EventCraft is a platform designed to revolutionize the event planning process by bringing clients and service providers together on a unified, easy-to-use platform. Through our commitment to creating a user-friendly interface, robust backend architecture, and reliable data management, we have ensured that users can seamlessly explore, compare, and book services that align with their unique preferences and budgets. By addressing the common challenges of budget management, vendor discovery, and transparent pricing, we aim to empower users to plan and execute their events with confidence and ease. Our efforts have been focused on fostering collaboration between clients and service providers, creating a streamlined and efficient ecosystem that simplifies the event planning journey for everyone involved.

# References

[1] “Spring Initializr,” *Spring Initializr*. Available: https://start.spring.io. [Accessed: Dec. 17, 2024]

[2] “Spring Boot,” *Spring Boot*. Available: https://spring.io/projects/spring-boot. [Accessed: Dec. 17, 2024]

[3] “Get Started with React Native · React Native.” Available: https://reactnative.dev/docs/environment-setup. [Accessed: Dec. 17, 2024]

[4] “React.” Available: https://react.dev/. [Accessed: Dec. 17, 2024]

[5] “API Security Best Practices | Curity,” Dec. 03, 2024. Available: https://curity.io/resources/learn/api-security-best-practices/. [Accessed: Dec. 13, 2024]

[6] AyiFaruk, “Axios API request from React Native frontend to local spring backend service not working,” *Stack Overflow*, Feb. 07, 2024. Available: https://stackoverflow.com/q/71947163. [Accessed: Nov. 26, 2024]

[7] “Developing Mobile Apps with Spring Boot and React Native Jul 2024.” Available: https://developersappindia.com/blog/developing-mobile-apps-with-spring-boot-and-react-native. [Accessed: Nov. 26, 2024]

[8] “React Native DevTools · React Native,” Nov. 07, 2024. Available: https://reactnative.dev/docs/react-native-devtools. [Accessed: Nov. 25, 2024]

[9] “Upgrade Expo SDK,” *Expo Documentation*. Available: https://docs.expo.dev/workflow/upgrading-expo-sdk-walkthrough/. [Accessed: Nov. 24, 2024]

[10] “upload image springboot - YouTube.” Available: https://www.youtube.com/results?search\_query=upload+image+springboot. [Accessed: Nov. 19, 2024]

[11] “@expo/vector-icons@14.0.4.” Available: https://icons.expo.fyi/Index. [Accessed: Oct. 29, 2024]

[12] “javascript - Slider toggle button in react native? - Stack Overflow.” Available: https://stackoverflow.com/questions/65680441/slider-toggle-button-in-react-native. [Accessed: Oct. 27, 2024]

[13] “Animated - Snack.” Available: https://snack.expo.dev. [Accessed: Oct. 27, 2024]

[14]“Dashboard — Expo.” Available: https://expo.dev/. [Accessed: Sep. 24, 2024]

[15] “Common Application Properties :: Spring Boot.” Available: https://docs.spring.io/spring-boot/appendix/application-properties/index.html. [Accessed: Sep. 21, 2024]

[16] M. Babu, “Free Hosting Bliss: Deploying Your Spring Boot App on Render,” *JavaToDev*, Mar. 13, 2024. Available: https://medium.com/spring-boot/free-hosting-bliss-deploying-your-spring-boot-app-on-render-d0ebd9713b9d. [Accessed: Sep. 11, 2024]

[17] “What is collaborative filtering? | IBM,” Mar. 18, 2024. Available: https://www.ibm.com/topics/collaborative-filtering. [Accessed: Sep. 20, 2024]

[18] “Getting Started | Accessing data with MySQL,” *Getting Started | Accessing data with MySQL*. Available: https://spring.io/guides/gs/accessing-data-mysql. [Accessed: Sep. 19, 2024]

[19] “Postman API Platform | Sign Up for Free.” Available: https://www.postman.com/. [Accessed: Sep. 17, 2024]

[20] “Spring Boot Quick Start 4 - Spring and some of its problems - YouTube.” Available: https://www.youtube.com/watch?v=YXlSkWq04jk&list=PLqq-6Pq4lTTbx8p2oCgcAQGQyqN8XeA1x&index=5&ab\_channel=JavaBrains. [Accessed: Sep. 11, 2024]

**Similar Existing Apps**

Top Event

MyWed

EventBrite