Software Design Specification

Prepared by :-

IIT2021001 - Kshitij Chaturvedi
IIT2021025 - Anand Choudhary
IIT2021080 - Dheeraj Devnani
IIT2021082 - Indrajeet Garse
IIT2021084 - Md Shaheer
IIT2021095 - Samyak Jain

Table of Contents

1 Introduction....3

1.1 Purpose of Document....3

1.2 Document Scope....3

1.2.1 In-Scope....4

1.2.2 Out-of-Scope....4

1.2.3 Assumptions....4

1.3 Methodology and Tools....4

2 Design Overview....5

2.1 Background Information....5

2.2 System Evolution Description....5

2.3 Required Environment....6

2.4 Constraints....7

2.5 Design Trade-offs....7

3 Logical Architecture....7

3.1 Application Architecture....7

3.2 Communication Architecture....7

4 Data Model....8

4.1 Database Management System Files....8

4.2 Non-Database Management System Files....9

5 Detailed Design....9

5.1 Application Detailed Design....9

5.2 Communication Detailed Design....11

6 Graphical User Interface (GUI)....12

6.1 Navigation Hierarchy....12

1 Introduction

The "Event Expo" website for event management is a software platform that enables users to reserve venues and assess costs for a variety of events, including weddings, birthday parties, product launches, and anniversaries. Depending on the availability of particular venues in that location, the administrator may accept or refuse the booking.This document explains the system's architecture, design, and implementation as well as its functional and non-functional requirements.

1.1 Purpose of Document

The document offers in-depth information on how the software was built. It includes comprehensive information about the features of the software and how they have been precisely implemented.Additionally, it draws attention to the key aspects of the system by displaying the GUI.

1.2 Document Scope

The following document is expected to give a detailed description of the system requirements for running the project, operating environment for the project, architecture used, files used, and database design of the software design implemented.

1.2.1 In-Scope

- User account creation, login, and logout of user/admin/venue owner.
- Searching for different venues.
- Booking of venues.
- Comparison of prices on the basis of venues.
- Receiving feedback through Emal(EventExpo8@gmail.com).
- Allotment of venues on basis of functions and required event

1.2.2 Out-of-Scope

- Payment processing by user for booking of venues.
- Integration with external services or APIs.
- Advanced project management features such as conduction of online events.
- Automatic allotment of venues on the basis of availability.
- Mobile app development or support.

1.2.3 Assumptions

- All users have a reliable internet connection and access to a modern web browser.
- The system will be used primarily by event organizers and venue owners..
- Users will provide accurate and complete information after login.
- Users will be able to book particular venues for particular events.

1.3 Methodology and Tools

The software has been produced by implementing the incremental methodology. It has been built and delivered in small increments, each of which adds new features or functionality. Each increment is developed, tested, and delivered independently, and the final product is created

by integrating all the increments. The requirements were divided into small chunks, and each chunk was developed and tested before moving on to the next chunk. This approach allowed feedback on the software at each stage of the development procedure, and to make changes or improvements based on that feedback making it more convenient to detect flaws during execution and testing of each chunk.

The tools used for implementing this software are PHP and MySQL. PHP has been used as it is easy to learn and implement, open source and due to simpler database connectivity. MySQL has been used as it is easy to learn and implement, open source and due to its compatibility with PHP.

2 Design Overview

2.1 Background Information

The key user groups for the Anusandhaan project include students and admin. Students will be the primary users of the system and will be able to create an account, browse projects, and submit publishing requests. Admin will be responsible for managing the projects, deciding whether to accept or reject publishing requests, and will be responsible for maintaining the system and providing support to users as needed.

2.2 System Evolution Description

The development of the website had been carried out by dividing it into various subtasks. The subtasks are listed below:

- **Event Organizer Sign Up / Sign In:** It is mandatory for any user to make an account to access the functionalities of user provided such as *venues comparison,venues booking,providing feedback ,access to gallery section.*
- **Owner Sign Up / Sign In:** It is mandatory for owners to make an account on the website and provide information about the venue such as the availability status,booking prices,update gallery section etc

- **Upload Images:** This functionality has been made available only for the ***Venue Owner*** where they could upload the images with the type of event.
- **Feedback form:** This functionality would help the event organizers to provide feedback about the website and give their reviews about the ease of reserving events to fulfill their purpose.
- **Venue Approval:** This functionality has been made available at the time of booking of venues where the user will be available with the complete details and status of venues.
- **Project Status:** This functionality has been made available only for the Users

  wherein a user could view the current availability status..
- **Search venues:** This functionality has been made available to both the ***user*** and the ***owner***.
- **Contact Form:** This functionality would help the event organizers to get in touch with Event Expo to solve any query that they may have.

2.3 Required Environment

Functionality	Requirements
Operating System	• Windows • Linux • macOS
Web and Database Server	• Mongoose • Nodejs
Development Tools	• Visual Studio Code

2.4 Constraints

- It is assumed that the admin responds to the various venues requests made by the user for the particular date just by checking the availability status and clicking, making it unavailable for the other user.
- It is assumed that the information shared by the venue owner is correct.
- There is no money transaction feature added that would be needed to complete the venue booking.
- Search Engine Optimization could be implemented for better searching the particular venues according to the needs of the user.

2.5 Design Trade-offs

The Design Trade-offs are mentioned as follows:

1. Innovation vs Practicality: It faces a trade-off between innovation and practicality. While it may be tempting to incorporate the latest technologies and features, doing could prove challenging to implement it righteously by the specified deadline.
2. Security vs Usability: A trade-off is the balance between security and usability. While it's important to ensure that the platform is secure, too many security measures may make it difficult for users to access and use the system.

3 Logical Architecture

3.1 Application Architecture

It has been implemented using Mongoose and MongoDB. Mongoose is responsible for storing user data, accessing the database, and generating appropriate responses whereas MongoDB is responsible for storing the data by providing the database.

3.2 Three-Tier Architecture

The most common architectural model used in event management websites is the three-tier architecture. This architecture divides the website into three layers: the presentation layer, the application layer, and the data layer.

The presentation layer is responsible for displaying the website's user interface to the user. It is implemented using HTML, CSS, and JavaScript.

The application layer is responsible for processing the user's requests and returning the appropriate response. It is typically implemented using a programming language such as NodeJS.

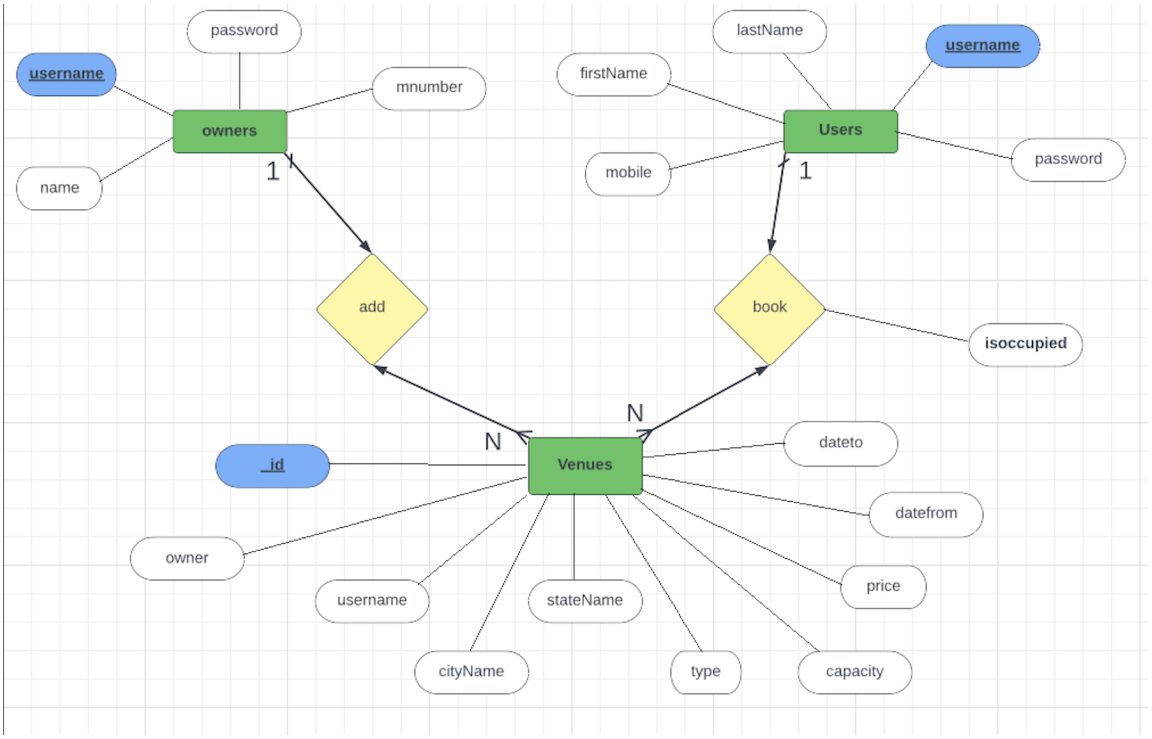
The data layer is responsible for storing the website's data. It is implemented using a database management system such as MongoDB.

4 Data Model

4.1 Database Management System Files

The tables created for Database Management are as follows:

1. Owners
    - username
    - name
    - mnumber
    - password
2. Admins
    - username
    - lastName
    - firstName
    - password
    - mobile_no.
3. Venues
    - dateto
    - datefrom
    - price
    - capacity
    - type
    - stateName
    - cityName
    - username
    - _id
    - owner



4.2 Non-Database Management System Files

The venue images have been uploaded by the venue owners during adding of venus is stored in database as .png, which can be directly displayed on website in gallery section.Any user whosoever login's will be able to access the images in gallery.

5 Detailed Design

5.1 Application Detailed Design

The application design is intended to provide a user-friendly interface which would allow the event organizers to get in touch with the venue owners and search from the list of available venues for the type of event that the organizer wants to conduct while also providing the owners with the necessary to manage the availability of the venue and the type of event.

1. Login and Registration Module: This module will allow both event organizer and owner to create an account in the system and log in using their credentials. It will consist of the following components:

- Registration Form: A web form that allows users to enter their name, email address, password, and other information required to create an account.
- Login Form: A web form that allows users to enter their credentials and log in to their account if valid credentials are entered.

2. Project Management Module: This module will allow users to view and select venues available in the venue list. It will consist of the following components:

- Venue Listing: A web page that lists all the available venues with details like name, availability status, booking price and and for particular date for which venue is required.
- Project Details Page: A web page that displays detailed information about a selected venue about the type of event and it's venue locality.
- Feedback Creation Form: A web form that allows users to give feedback through email where they can send their reviews about booking of venues.
- Gallery Page: This module will allow users to feel the environment and give rough idea about the decoration and aura for the venue they are booking.
- Review page: A web page where the users image along with their experience is mentioned.

3. Owner Module: This module will allow owner to manage venues, and add new venues along with the strength,price,gallery and location. It will consist of the following components:

- Venue Management: A web page that list of venues available for the particular date and type of event according to the need of user.

- Venue Request Management: A web page that lists all the requests made by users for the particular date, availability status, and associated information. Admins can manage by proper integration of requests of both owner and user.

4. Admin Module: This module will allow the admin to administer contact between the user and the venue owner. It will consist of the following components:

- Admin Services : A service that generates the proper list of available venues with the details provided by the venue's owner .
- Admin Verification Service: A service that checks whether the information provided by the owner is correct or not and shares the feedback and experience to the venue owner.

5.2 Communication Detailed Design

- **Three-Tier Architechture:** EventExpo is a web-based system that divides the website into three layers: the presentation layer, the application layer, and the data layer.

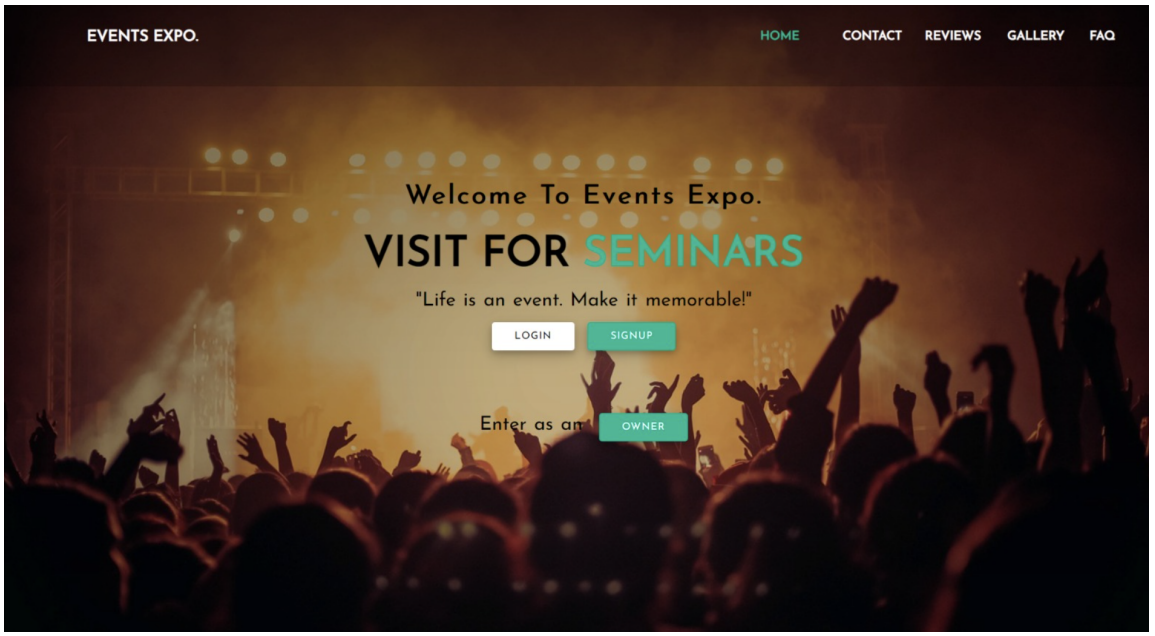
- The presentation layer is responsible for displaying the website's user interface to the user. It is implemented using HTML, CSS, and JavaScript. The application layer is responsible for processing the user's requests and returning the appropriate response. It is typically implemented using a programming language such as NodeJS. The data layer is responsible for storing the website's data. It is implemented using a database management system such as MongoDB.

- **Email Communication:** EventExpo has an email system which can be used to establish communication between the admin and the venue owners and users.

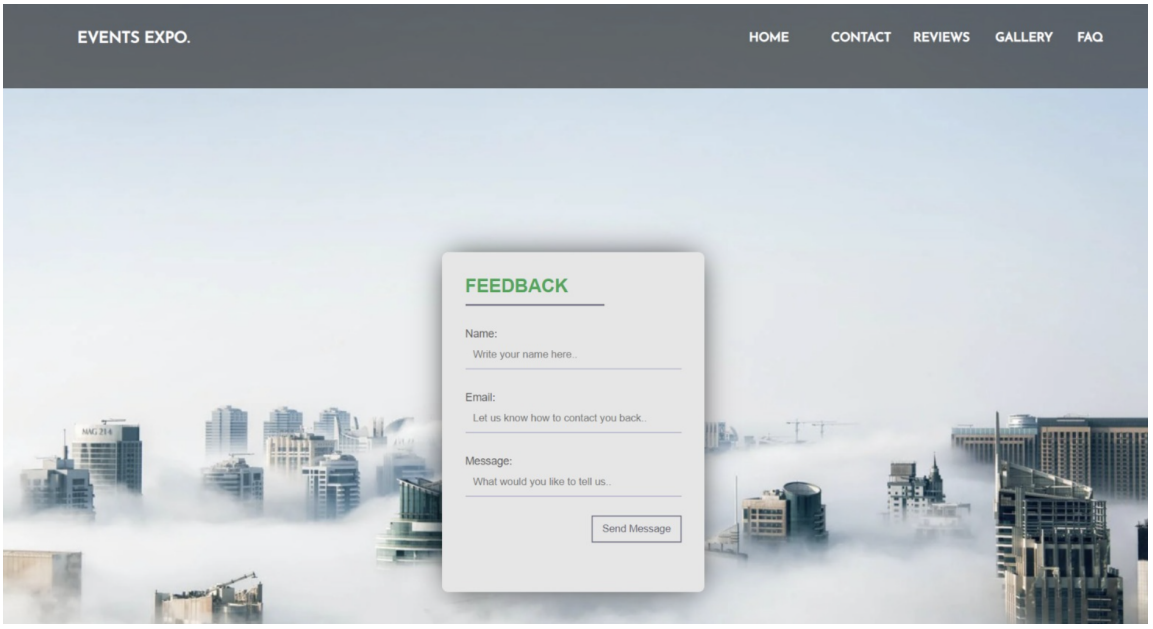
6 Graphical User Interface

6.1 Navigation Hierarchy

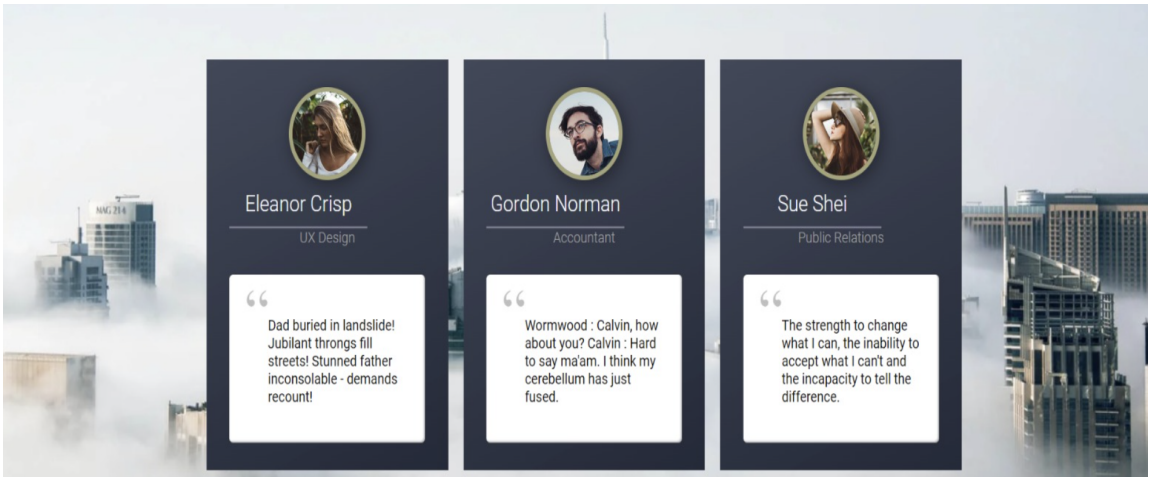
6.1.1 Home Page



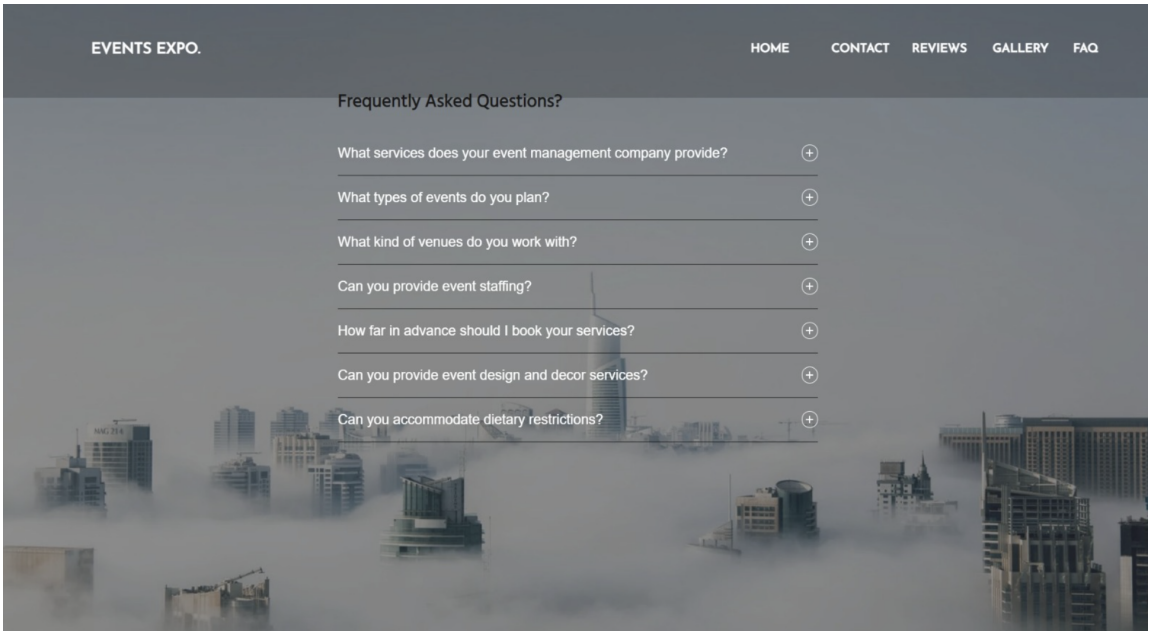
6.1.2 Feedback Page



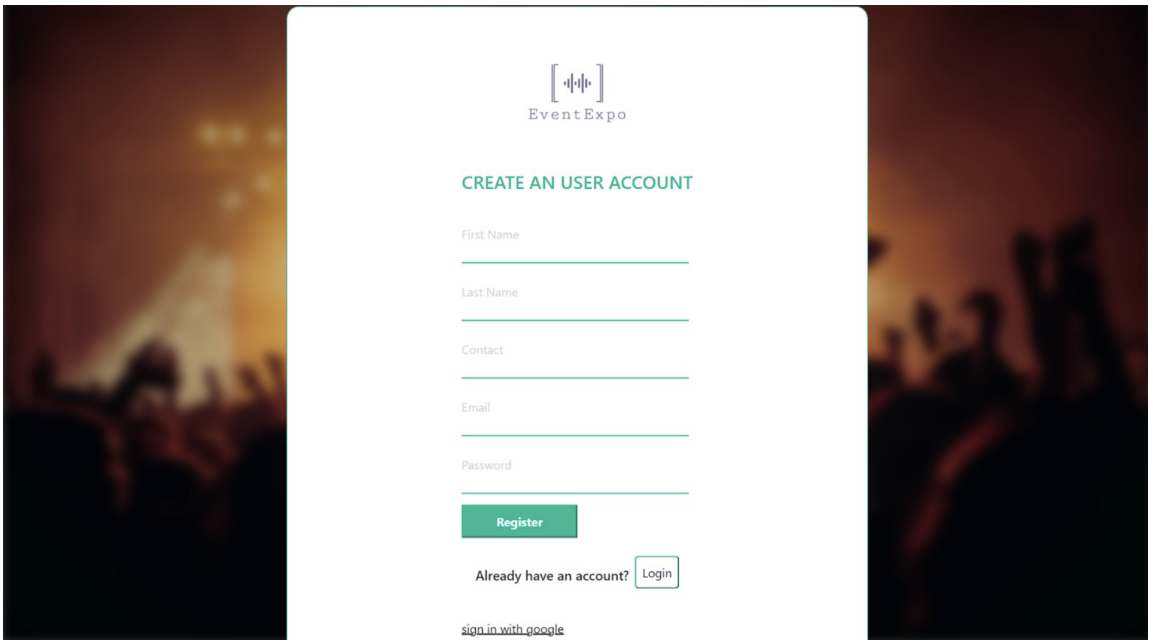
6.1.3 Review Page



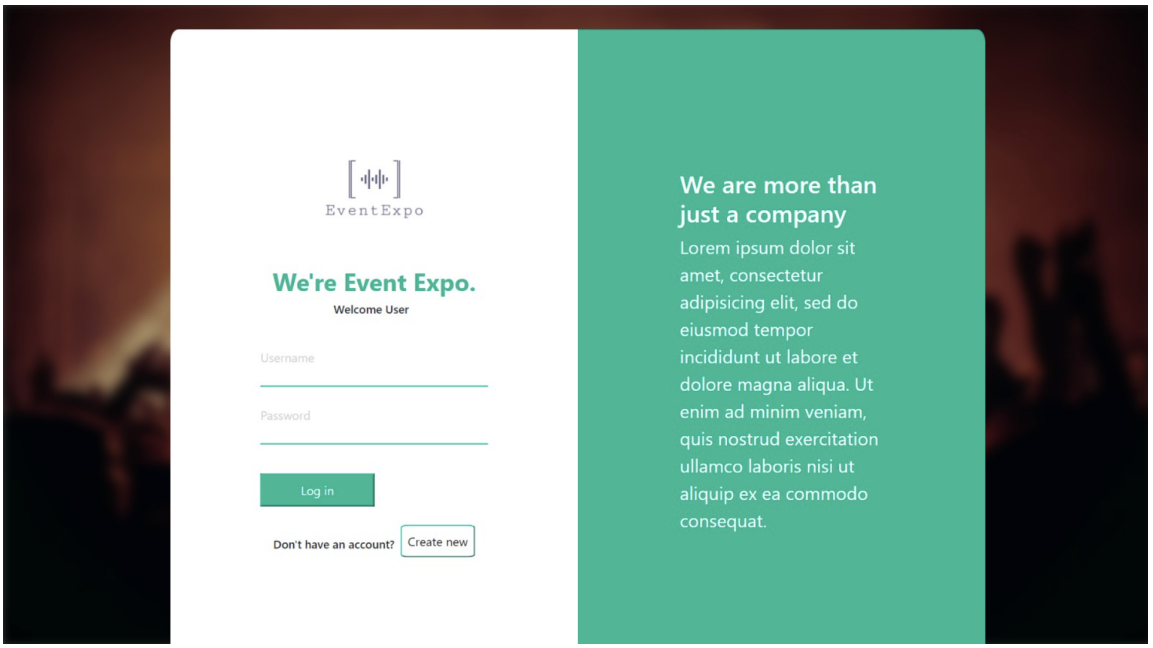
6.1.4 Frequently asked questions



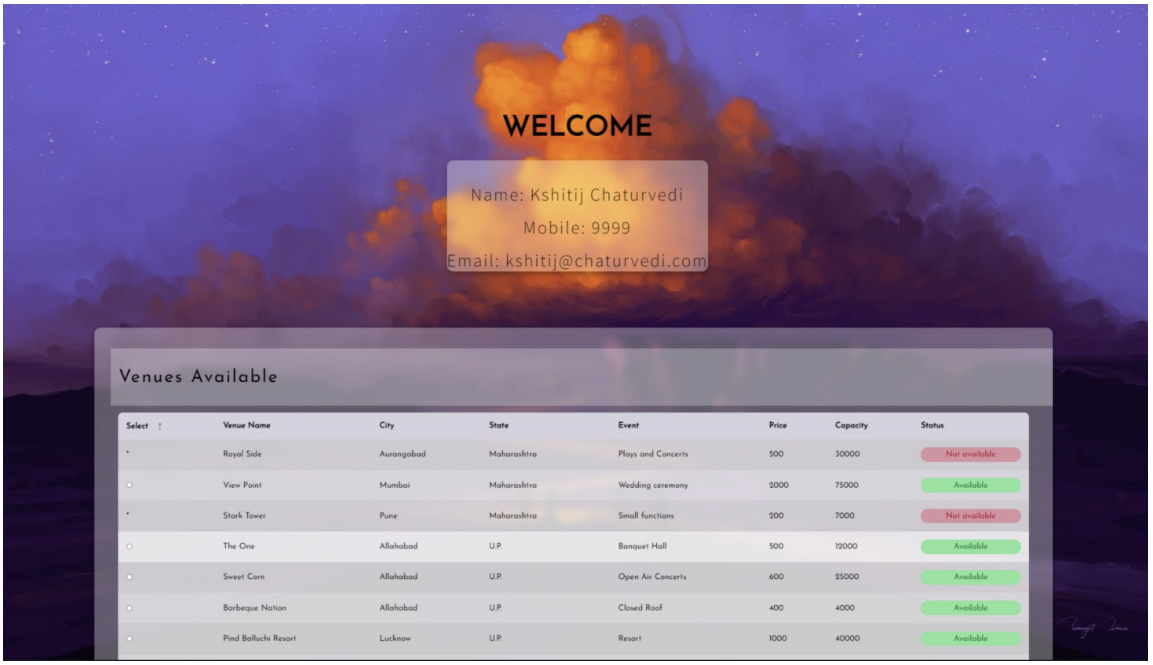
6.1.5 User Account creation Page



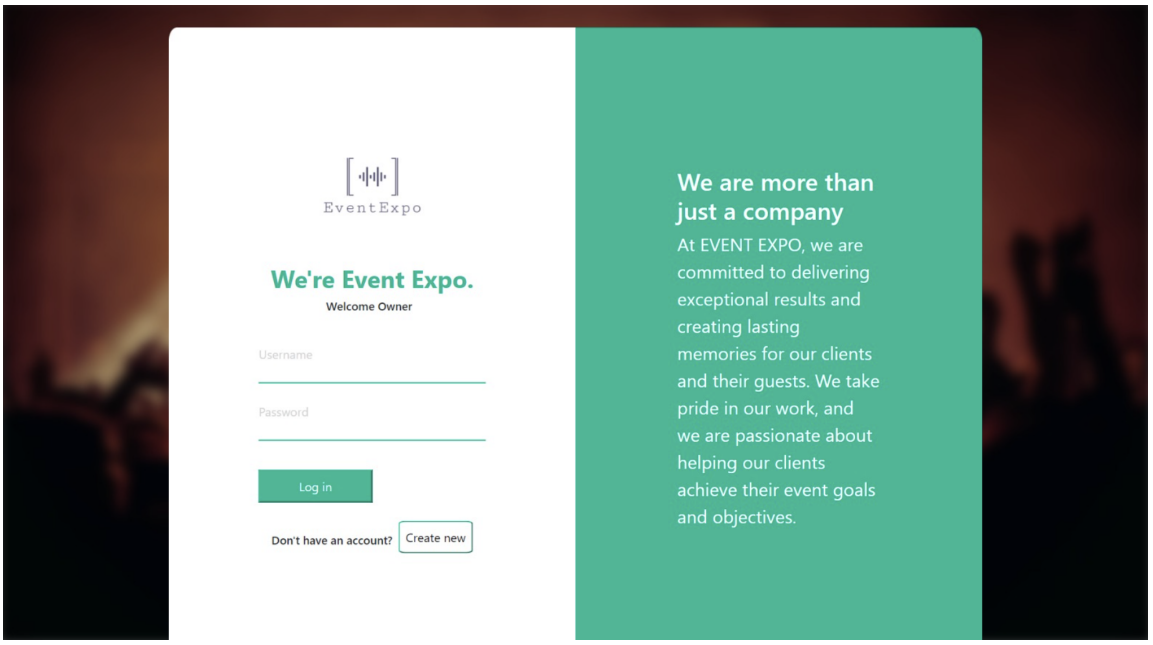
6.1.6 User Login Page



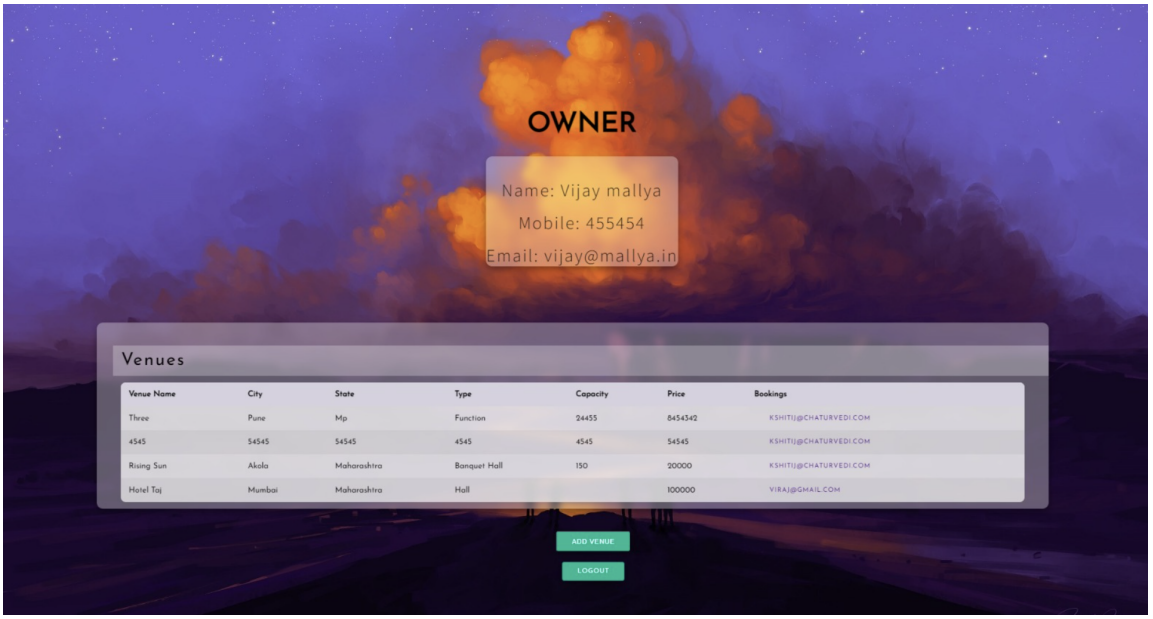
6.1.7 Available venue page



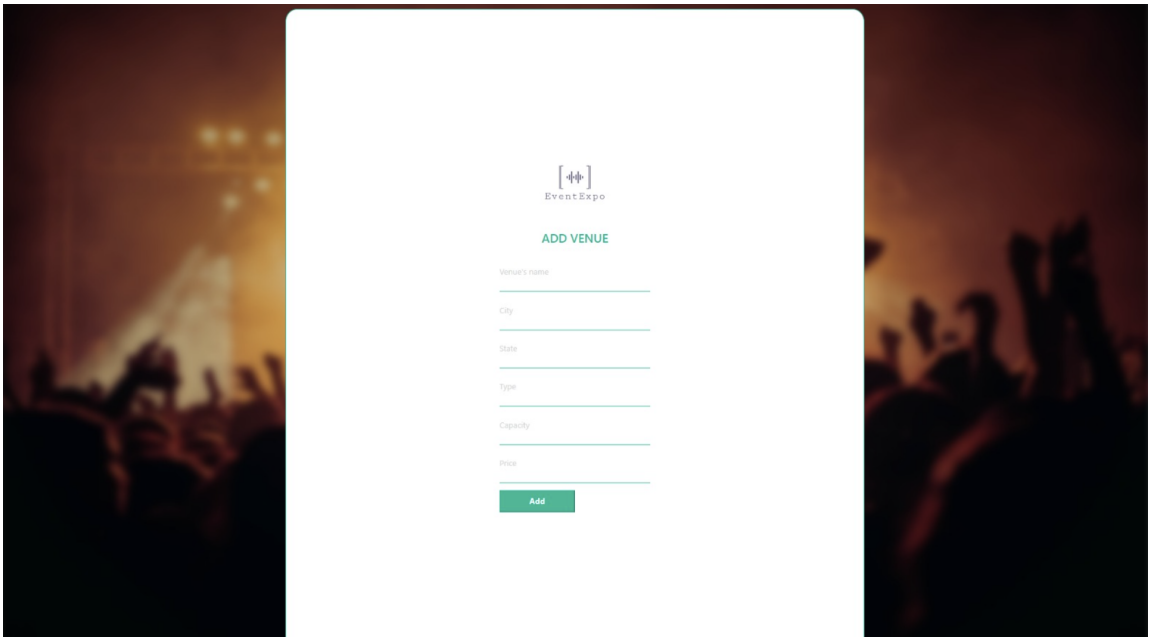
6.1.8 Owner login page



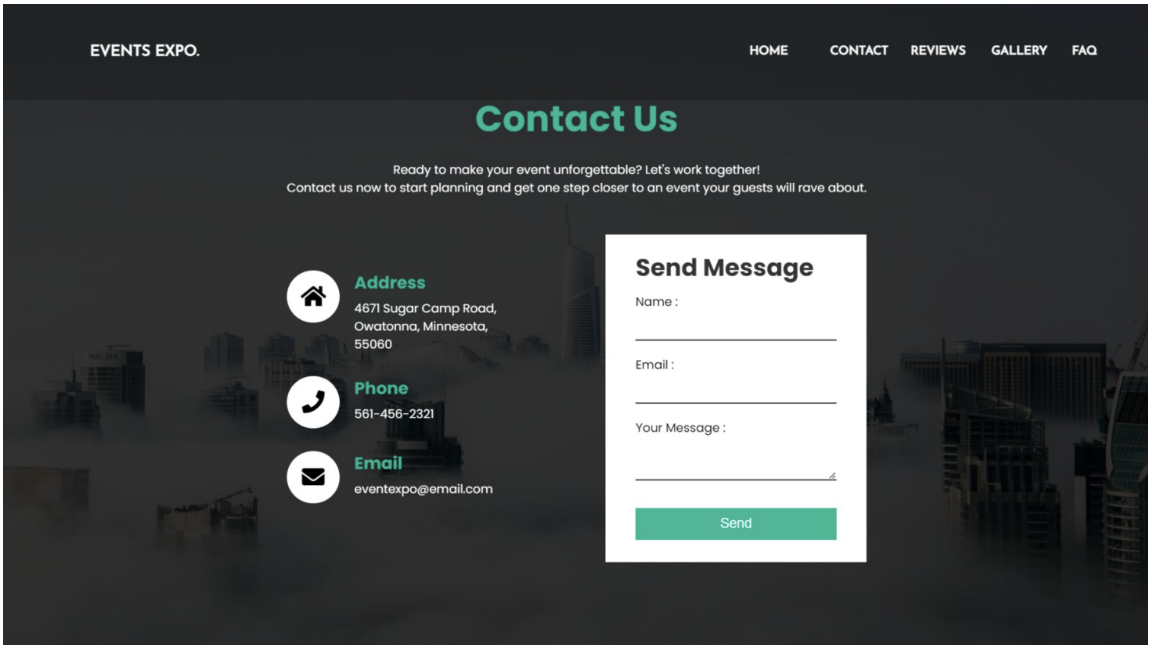
6.1.9 Owner detail Page



6.1.10 Add venue Page



6.1.11 Contact us Page



6.1.12 Transaction Page