Avenue Event Management Website

A PROJECT REPORT
Submitted by

Adithyan M S	**PRP19CS006**
Aswani N K	**LPRP19CS056**
Devanand T	**PRP19CS022**
Krishna P	**LPRP19CS057**

to

the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree

of

Bachelor of Technology
In
Computer Science and Engineering

Department of Computer Science and Engineering
COLLEGE OF ENGINEERING & MANAGEMENT PUNNAPRA
PUNNAPRA, ALAPPUZHA

JULY 2022

DECLARATION

I hereby declare that the project report "Avenue Event Management Website " , submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Mrs. Ansu Mirian Varkey. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Place: Punnapra
Date: 06-07-2022

ADITHYAN M S
ASWANI N K
DEVANAND T
KRISHNA P

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

COLLEGE OF ENGINEERING AND MANAGEMENT PUNNAPRA

CERTIFICATE

This is to certify that the project report entitled " **Avenue Event Management Website** " submitted by : **ADITHYAN M S (KTU ID : PRP19CS006), ASWANI N K (KTU ID : LPRP19CS056), DEVANAND T (KTU ID : PRP19CS022), KRISHNA P (KTU ID : LPRP19CS057)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Ms. ANSU MIRIAM VARKEY Asst. Professor Dept. of CSE (Project Guide)	Ms. AGGIE VARGHESE Asst. Professor Dept. of CSE (Project Co-Ordinator)	Ms. SONY M S Asst. Professor Dept. of CSE (Project Co-Ordinator)	Ms. SMITHA M JASMINE Asst. Professor Dept. of CSE (HOD)

Place: Punnapra

Date: 06-07-2022

CONTENTS

ACKNOWLEDGEMENT

First and foremost I thank God Almighty for his blessings for this project. I take this opportunity to express my gratitude to all those who have guided in the successful completion of this Endeavor. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our Principal Dr. ROOBIN V VARGHESE for providing infrastructural facilities and for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the department, CSE, Mrs. SMITHA M JASMINE, Assistant Professor for her full-fledged support.I owe Mr. SURESH KUMAR N, Associate Professor & Senior Faculty, Department of Computer Science, a deep sense of gratitude for his unwavering support. I am also deeply indebited to my project guide Mrs.ANSU MIRIAN VARKEY , Asst. Professor, Dept. of CSE and the project co-ordinators Ms. AGGIE VARGHESE and Ms. SONY M S , Asst. Professors , Dept. of CSE for their keen interest and ample guidance throughout the project.

I am indebted to my beloved teachers whose cooperation and suggestions throughout the project which helped me a lot. I also thank all my friends and classmates for their interest, dedication and encouragement shown towards the project. I convey my hearty thanks to my parents for the moral support, suggestions and encouragement to make this venture a success.

ADITHYAN M S
ASWANI N K
DEVANAND T
KRISHNA P

ABSTRACT

Online event management system is a best way to keep clients engaged with the service as they are on the move. As technology is growing rapidly we are also moving to a technical world where everything we want is to be online. The main aim of this proposal is to develop an online event management system. To gather requirements for designing an online event management system. To design or model an online event management system and to test, validate and implement the designed system. We are planning to build a Event Planners/Management Website through this project. This acts mostly as a business website which serve as a space to provide general information about our company or a direct platform for e-commerce. The services provided by us are small and/or large-scale personal or corporate events such as festivals ,conferences ,ceremonies ,wedding ,formal parties ,concerts or conventions.

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

OEMS	Online Event Management System
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
DBMS	DataBase Management System
JSON	JavaScript Object Notation

Chapter 1

INTRODUCTION

1.1 Web Development

Web development is the work involved in developing a website for the Internet (World Wide Web) or an intranet (a private network).[1] Web development can range from developing a simple single static page of plain text to complex web applications, electronic businesses, and social network services. A more comprehensive list of tasks to which Web development commonly refers, may include Web engineering, Web design, Web content development, client liaison, client-side/server-side scripting, Web server and network security configuration, and e-commerce development. Web Development consists of four architectural layers:

- Presentation layer (PL)
- Business logic layer (BLL)
- Data service layer (DSL)
- Data access layer (DAL)

1.2 Online Event management system

Online Event management system was web-based application that enhanced project management to the creation and development of large scale events such as festivals, conferences, ceremonies, weddings, formal parties, concerts, or conventions. It involved studying the brand, identifying its target audience, devising the event concept, and coordinating the technical aspects before actually launching the event. The Online Event management system (OEMS) enabled customers/ clients view various packages/products about the event and make booking through the online

platform. The process of planning and coordinating the event is usually referred to as event planning and it included budgeting, scheduling, site selection, acquiring necessary permits, coordinating transportation and parking, arranging for speakers or entertainers, arranging decor, event security, catering, coordinating with third party vendors, and emergency plans. Each event was different in its nature so process of planning execution of each event differed on basis of type of event.

1.3 Benefits of OEMS

1.3.1 Reduce your workload

If you don't already use an online event management system then chances are you spend most of your day tackling a mountain of paperwork. From mailed in (and frequently iJlegible) registration forms needing to be entered onto your spreadsheet to invoices and confirmation letters that need to be sent out, the paperwork seems never ending. OEMS automates all these tasks meaning no more boring paperwork which lets you devote time to more important tasks that really make a difference to your attendees.

1.3.2 Reduce your costs

If you have employed expensive temporary staff to help with data entry and handling registration queries you can save here too. Some OEMS clients have reported immediate savings of up to 60staffing costs! Instead of paying for printed brochures or leaflets, OEMS allows you to place your entire event information package online with no cost until you start accepting registrations. During the registration phase, OEMS uses email notifications and generates mvo1ces automatically for attendees to download, again saving you mailing costs.

1.3.3 Get paid faster

OEMS allows you to accept credit and debit cards online as well as more traditional forms of payment such as cheques which means you get paid straight away. Unlike our competitors, we connect OEMS to your payment processor for free so all the money goes straight to you. OEMS even helps you keep track of who still owes you money too, helpfully flagging up overdue payments for you to chase.

1.3.4 Enhanced perception of your organization

Your attendees can access your online event registration pages at their convenience and not just during office hours. Not only can they register themselves, but they get virtually instant confirmation emails, invoices and receipts which all helps to reassure. As an important point of contact between your organization and potential customers, members and donors, you want them to have a great experience and this is what OEMS delivers in spades. As an additional bonus, the OEMS allows your attendees to come back in at any time and make changes to their registration and/or print out their event documentation.

1.4 Problem Statement

Curently Event Management system is manual and only accessible to staff. The client has to travel to the company offices in order to schedule, book and organize an event such as Birthday Party, Marriage, Reception, Ring Ceremony. Clients pay cash to book for an event which is inconveniencing when customers are many at the company. It takes lots of time of customer because they have to search such event organizer and contact them individually so an online event management system is needed which will enable the customer make booking, schedule events online at any preferred time

Chapter 2

LITERATURE SURVEY

2.1 Introduction

This section summarizes the contents of the literature review that is event management systems defined; online event management systems evolution and trend; Benefits of OEMS (proposed system): Comparison between related works.

2.2 Analyzing the existing system

In the existing system customer contacts the company for event management. He provides the details of the event and its requirements. He explains its aims, how long it will last, its format (Presentation/Workshop and/or Exhibition etc.), expected number of delegates/guests, equipment and furniture required, whether any delegate pack or promotional material is to be distributed, and other facilities required. The Event Manager studies the requirements of the event carefully and using the event management system The company offers some readymade packages to choose from. If the customer agrees, the event is booked and the advance deposit is taken by the company. According to the requirements of the event, different bookings are made. A strategic schedule is prepared for smooth conduct of the event. The Event Management System helps the manager in different tasks of planning, scheduling and Conducting the event. This system provides instant access to event-related information.

Nowadays, the event industry has played a vital role in our society. People have come up with a lot of occasions for organizing events such as educational events, birthdays, international conferences, company parties etc. Generally speaking, events are also a pmi of human beings' social life because they get to know and

talk with different people with different backgrounds. However, in order to organize a good and successful event, it requires a thorough and detailed planning process. Event managers and event coordinators need to collaborate in order to formulate the most viable plan for events. In order to make events successful and well organized, all of the stages in the entire planning process also need to be in harmony and be corelated with one another. Risk Management has been of great importance during the entire planning process. Under no circumstances should risk management be underestimated. Additionally, evaluation process helps event organizers and event managers to realize which aspects should be improved and need better development. The event management industry is a dapper of a dandy. Due to the endless changes and sweeping improvements in the cunent events technology, there's no place for stability.

2.3 Comparison between related works

"Event management - A special kind of project management " (Peter J.A Reusch, Pascal Reusch,.) published in 2019.[1]Event management is a strong and fast growing profession with a rather low level of standardization. Often we take event management as a part of project management, but we have to consider that event management has very specific concepts and issues, and needs further developed methods and tools. We classify events, we compare project management and event management, we reconsider standards in both areas, and discuss perspectives for a stronger standardization of event management in the future.

"Development of a mobile based birth and funeral event planning application in Bahrain " (Ehab Juma Adwan, Manal Zulfiqar, Jeehan Malik, Ayesha Arif. [2021])[2]This research is aimed at developing a mobile application "Emotive Events", acting as an event planner that maintains time and budget for a private event including funeral considering multiple religions and birth parties and addresses many features such as checklists creation, guest list, religion-based services, online payments, a collaboration of service providers/services that most existing applications typically lack all in one place.

"Factors Affecting Customer Satisfaction with Ecommerce Websites - An Omani Perspective " (Shouvik Sanyala , Mohammed Wamique Hisamb. [2019])[3]Attaining customer satisfaction is one of the keys to success in today's crowded and competitive online market. This study attempts to find and analyse the impor-

tant factors affecting customer satisfaction with Ecommerce websites and online purchasing in Oman. The study shows that Price and Ease of Use and availability of multiple payment options are the important factors that positively influence customer satisfaction.

"Transforming a website from desktop to mobile a cross platform viewpoint " (Leeladevi B, Pinku Hazarika, Hemadri Pavan Kumar Nanyan [2020])[4]The paper discusses the different migration facets like the application types to be considered when migrating a desktop website to a mobile site, best practices, supporting a wide range of devices, design approaches, maintaining a single codebase for all devices, etc. It presents a criteria sheet to help a developer choose the right development approach for migration.

"Twitter bootstrap and AngularJS: Frontend frameworks to expedite science gateway development " (Viknes Balasubramanee, Chathuri Wimalasena, Raminder Singh,Marlon Pierce. [2022])[5]To enable the goals of a science gateway and the communities of scientists it supports, gateway developers need to be able to spend more time on designing and developing the user experience and less time on wrestling with the underlying technology (such as HTML5, CSS, and JavaScript). In this poster, we describe our experiences using Twitter Bootstrap and AngularJS frameworks to address this balance between design and implementation, empowering developers to create better styled and easily maintainable websites.

"Web Development and performance comparison of Web Development Technologies in Node.js and Python " (Sai Sri Nandan Challapalli, Prakarsh Kaushik, Shashikant Suman, Basu Dev Shivahare, Vimal Bibhu, Amar Deep Gupta. [2021])[6]This research paper discusses the process involved in developing a website in past and present, development of content delivery over the years, the website uses, a website for mobile devices, and performance comparison between two of the most used web backend development technologies, i. e, Node.js and Python. For comparing performance, we have used Locust – an open-source software and Autocannon and tested both of them under similar conditions

"Express supervision system based on NodeJS and MongoDB " (Li Liang, Ligu Zhu, Wenqian Shang, Dongyu Feng, Zida Xiao. [2019])[7]This paper discusses the advantages of using AngularJS to build the front-end framework, the advantages of using NodeJS to construct the back-end Web server, and the performance ad-

vantages of storing data based on MongoDB. This paper focuses on the storage solutions of using MongoDB to store large data and the statistical analysis solutions based on MapReduce. This paper argues on how to build Web services that meet the requirements of large data visualization based on NodeJs.

"Study on Website Search Engine Optimization " (Zhou Hui, Qin Shingang, Liu Jinhua, Chen Jianli [2020])[8]With the rapid development of information technology, search engine optimization (SEO) technology has attracted more and more attentions. In order to improve their website visit quantity, SEO techniques can make a better ranking in the search result using the keyword selection and deployment, high quality back links, rational website constitution, and rich content, etc. This paper discusses in detail the technical process of website search engine optimization in terms of the search engine work principle, factors affecting search ranking, and website search engine optimization method.

Table 2.1: **Comparison between Related Works**

Sl No	Name of Paper	Paper type	Year	Description
1	Event management - A special kind of project management	IEEE paper	2019	Compare project management and event management, to reconsider standards in both areas, and discuss perspectives for a stronger standardization of event management in the future.
2	Development of a mobile based birth and funeral event planning application in Bahrain	IEEE paper	2021	mobile application "Emotive Events", acting as an event planner that maintains time and budget for a private event including funeral considering multiple religions and birth parties and others
3	Factors Affecting Customer Satisfaction with Ecommerce Websites - An Omani Perspective	IEEE	2019	Study shows that Price and Ease of Use and availability of multiple payment options are the important factors that positively influence customer satisfaction.
4	Transforming a website from desktop to mobile a cross platform viewpoint	IEEE	2020	Migrating a desktop website to a mobile site
5	Twitter bootstrap and AngularJS: Frontend frameworks to expedite science gateway development	IEEE	2022	Empowering developers to create better styled and easily maintainable websites.
6	Web Development and performance comparison of Web Development Technologies in Node.js and Python	IEEE	2021	Performance comparison between two of the most used web backend development technologies, i. e, Node.js and Python.

7	Express supervision system based on NodeJS and MongoDB	IEEE	2019	The advantages of using NodeJS to construct the back-end Web server, and the performance advantages of storing data based on MongoDB.
8	Study on Website Search Engine Optimization	IEEE	2020	Improve the website visit quantity, SEO techniques can make a better ranking in the search result.

Chapter 3

METHODOLOGY

3.1 Introduction

This chapter discusses the methodology that was used in gathering the data and implementing a event management website. Here the researcher aimed at identifying the objectives to be carried out and the methods and tools to be used to present and analyze data to develop the event management website.

3.2 Target Population of study

The target population for this study will be FEP finance managers (I 0),event planners (40),human resource manager (9) and clients (200).

3.2.1 Sample size and sampling techniques

The sample size was estimated using the formula (Sloven's formula).

$$\text{n}= N/(1 + N(e^2))$$

where n is the sample size;N is the sample population; e is the marginal error which is constantly 0.05

According to Sloven's formula the sample size for finance managers is equal to 10, event planners is equal to 40, human resource manager is equal to 9 and clients is equal to 133.

3.3 Data Collection Techniques

3.3.1 Interviews

An interview is a process of conducting intensive individual interviews with a small number of respondents to explore their perspectives on a particular idea, program or situation. Face to face method of discussion was used to gather information from people who work in the company / organization and get the knowledge to design a responsive system.

3.3.2 Observation

This technique was used to gather accurate information about how the system actually operates, particularly about processes. This involves the researcher to systematically watch and record the behaviors and characteristics of operations and processes in the company. Although the method is time consuming, it has a number of advantages, which include: It gives more detailed and context related information, It permits the collection of information on facts not mentioned in the interview, It permits tests of the reliability of the responses to the questionnaires, observe operations of a program as they are actually occurring and can adapt to events as they occur.

3.3.3 Design

In systems design the design functions and operations are described in detail, including business rules, process diagrams and other documentation. The output of this stage was describe the new system as a collection of modules or subsystems. The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will be produced as a result of interviews, workshops, and/or prototype efforts.

3.4 Study Findings

According to the data collection methods the following was thought to be done to improve on the gaps found in the current system

3.4.1 Weaknesses of the Current System

It was through the information got about the current system that the researcher was able to identify its weaknesses. This helped in understanding what was to be done in order to develop a new system.
The following weaknesses were found in the current system:

- The use of a paper-file system to record, process and monitor event booking data, financial data and personnel data causes a delay in decision making. This is so because managers get limited time to analyze the information for proper decision making process.
- The current method of event booking is time wasting, ineffective and inefficient to the clients who would have rather booked for the events from wherever they are by using online platforms, online payments and mobile money payments.
- Access to Information is very difficult especially when it comes transactions for bookings
- It's very difficult and time consuming to generate repo1is at the head office since all support branches must submit their reports first in order to generate the general / final rep01i.

3.4.2 Presentation and Analysis of the findings

From the analysis made, there is need for an online event management system to support the carrying out, monitoring and recording of business transactions of Event Planners. The current system is unreliable, inefficient and costly. Three categories of stakeholders were interviewed and these included;

3.5 Designing or modeling the system

System design tools to create systems that meet the needs of stake holders. The tools are used in system designing and modeling are flow charts, use case diagrams, context diagrm11, flow chaiis and use case diagrams. A system flow chart is a way of displaying how data flows !n a system and how decisions are made to control events

3.5.1 MongoDB

MongoDB, the most popular NoSQL database, is an open-source document-oriented database. The term 'NoSQL' means 'non-relational'. It means that MongoDB isn't based on the table-like relational database structure but provides an altogether different mechanism for storage and retrieval of data. This format of storage is called BSON (similar to JSON format).

3.5.2 NodeJS

Node.js is an open-source and cross-platform runtime environment built on Chrome's V8 JavaScript engine for executing JavaScript code outside of a browser. You need to recollect that NodeJS isn't a framework, and it's not a programing language. It provides an event-driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side applications using JavaScript.

3.5.3 HTML and CSS

Hypertext Markup Language (HTML) is based on the Standard Generalized Language (SGML). HTML is a language for describing the structure of a document, not its presentation (Lemay, 2001). Cascading style sheets will be used in designing common application and systems. These provide specific design of web pages. HTML defines a set of common styles for web pages: headings, paragraphs, lists and tables. HTML provides a means by which a documents main content can be am1otated with various kinds of meta-data and rendering hits. The rendering hints include specifying scripts, image maps and form definitions for web browsers. Macromedia Dreamweaver is the leading software tools for editing HTML. Content and presentation will be combined using server side scripting languages like PHP to make the final HTML.

3.6 Requirements Analysis (using a use case diagram).

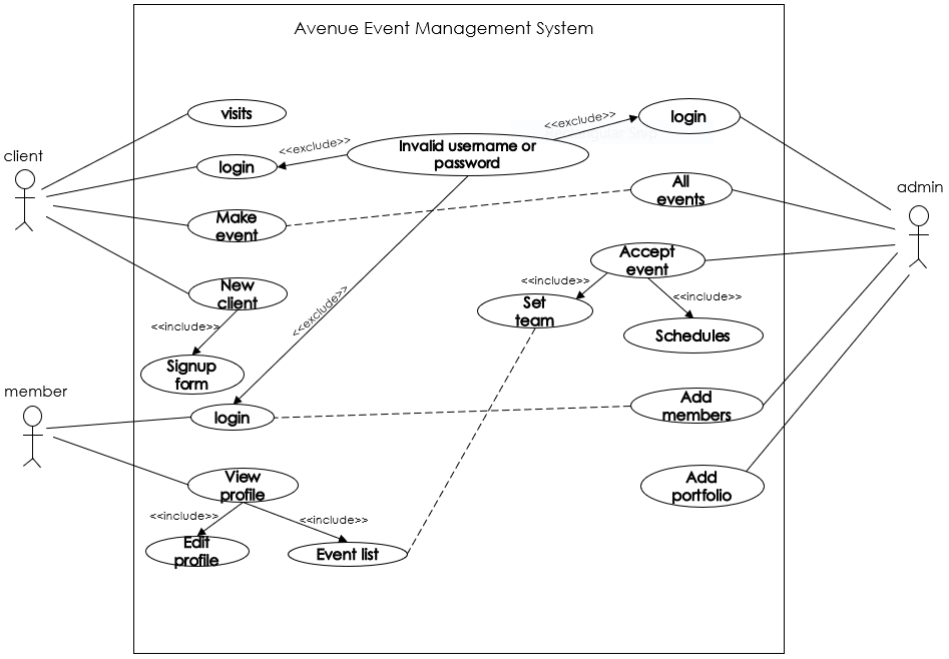


Figure 3.1: Use case diagram

3.6.1 Functional Requirements

Functional requirements define the specific functions that the system perfoms, along with the data operated on by the functions. The functional requirements are presented in 19 scenarios that depict an operational system from the perspective of its end users. Included are system features and an enumeration of all the specific requirements associated with these features.

- The system incorporated mechanism to authenticate its clients , members and admins
- The system verifies and validates all user input and notifies m case of error detection and helps the user in error correction
- The system allows quick messages to be exchanged without face to face interaction
- clients will be able to use the system to make booking of events online.

- The system will allow each members to have profile page, where he/she can see the scheduled events which are to be handled by them.
- The system provides a admin panel in which the administrator can view all the event requests and accept/reject them. The admin can also add new Member details and Portfolio images in the admin panel itself.

3.6.2 Non-functional Requirements

Non-functional requirements address aspects of the system other than the specific functions it performs. These aspects include system performance, costs, and such general system characteristics as reliability, security, and portability. The non-functional requirements also address aspects of the system development process and operational personnel. It includes the following:

- The system was user friendly and consistent
- The system provided attractive graphical interface for the user
- The system allowed developer access to installed environment
- The system targeted customer base

3.6.3 System Requirements

To be used efficiently, the online bus ticket booking system will need certain hardware components and software resources to be present on a computer. These requirements are regarded as minimum for the sake of running the system:

Hardware Requirements

- **Server computer** Intel Processor - 2.7GHz, Memory - 16 GB, Disk Space - 500 GB.
- **Client computer** Intel Processor - 1.6 GHz, Memory - 4 GB, Disk Space - 250 GB.

Software Requirements

- **Server computer** Windows 2008 Server, Web Server- Wamp5, DBMS - Mongodb.
- **Client computer** Windows XP, Mozilla Firefox 54.6.

3.7 System Design

The new online event management system has been designed in line with the user and system requirements that were identified during the data collection and analysis stage. The system will be used by the Event Mangers, Event members and the customers.

3.7.1 Network and System Architecture

It is represented using a three tier architecture that comprises of user interface, process management and Database Management System (DBMS). It shows the components of the system, the services they provide and the way they communicate to bring about the system functionality.

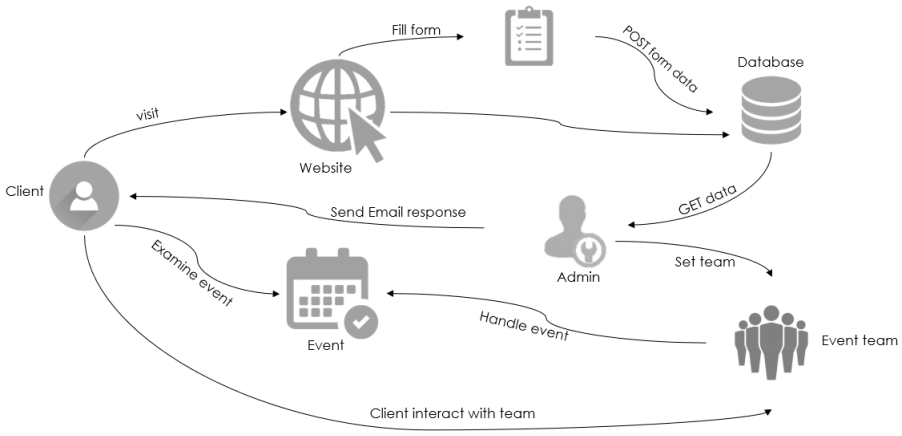
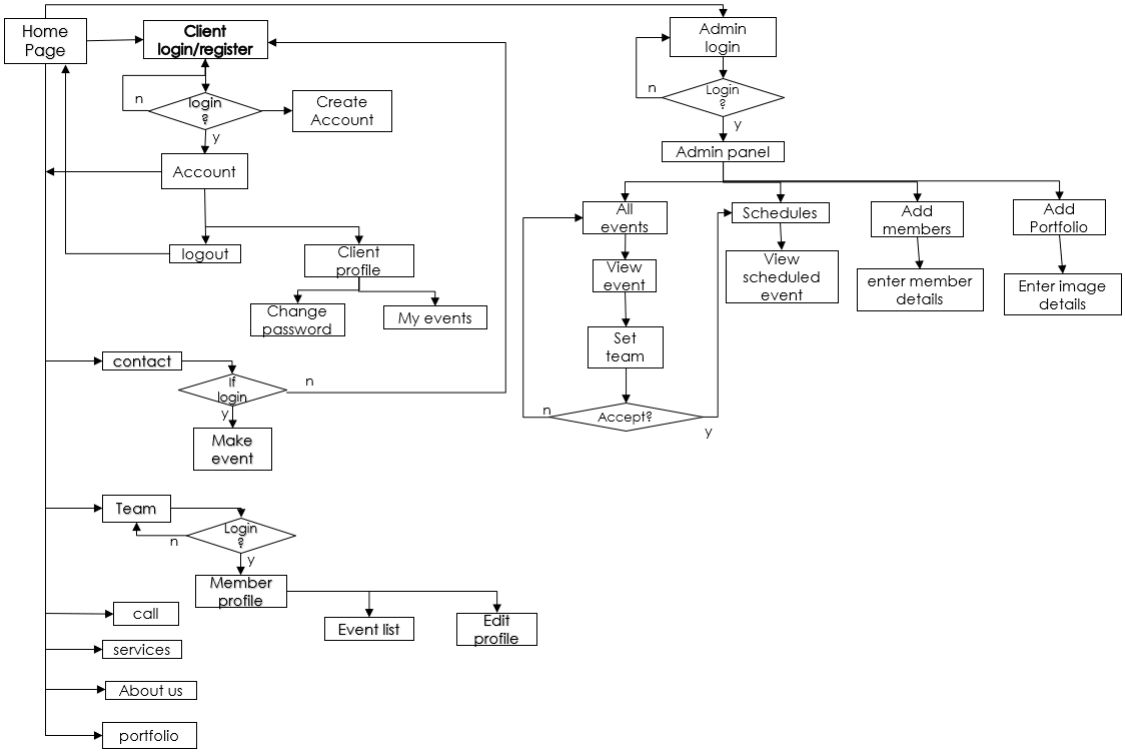


Figure 3.2: System Architecture

3.7.2 System Flow Diagram

The system flow diagram is one of the graphical representations of the flow of data in a system in software engineering. The diagram consists of several steps that identify where the input is coming to the system and output going out of the system. With the help of the diagram, it is possible to control the event decisions of the system and how data is flowing to the system.



3.7.3 Database Design

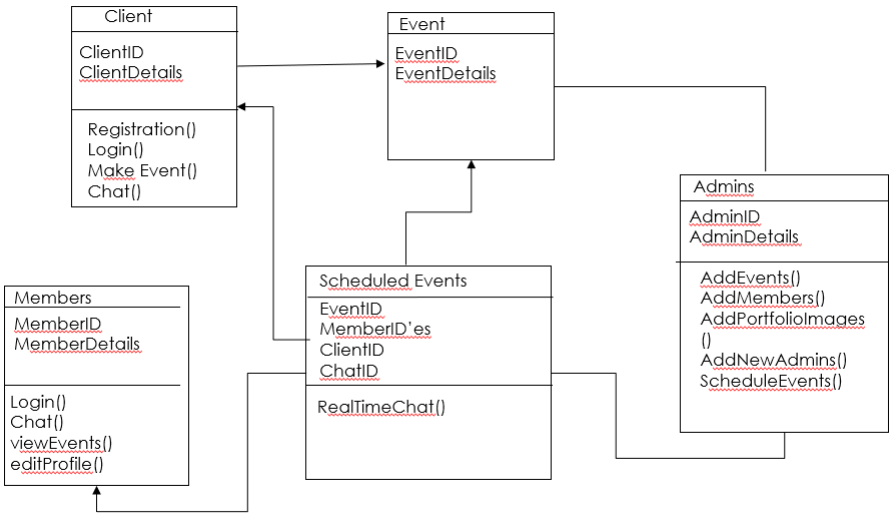


Figure 3.3: Database Design

Chapter 4

SYSTEM IMPLEMENTATION

4.1 introduction

The data analysis done in the previous chapter indicates that the automation of the processes of carrying out, monitoring and recording business transactions of Avenue Event Planners can lead to reduction in cost of doing business, improved data management and quick decision-making process. Details of reliable and timely information about the different events scheduled. The database used to store the information was separated from the user interface and business logic. MongoDB was used to implement the database and a combination of HTML ,CSS and JavaScript were used for the implementation of the interfaces.NodeJS was used for implementing the backend logic .

4.2 Implementation

The table below shows the different activities and deliverables of the online Event Management System.

Table 4.1: **System implementation plan**

Activity	Tools Used	Deliverables
Database Implementation	MongoDB 6.0 ,MongoDB Atlas	Fully operational and working Database.
Implementation of Interfaces	HTML5, CSS ,Bootstrap	User friendly interfaces
Implementation of business Logic	JavaScript, NodeJS 18	System code,

System Testing	NodeJS 18, MongoDB, HTML5, CSS , Apache 2.0.22	System reviewed, System security tested.

4.3 Database Implementation

MongoDB DBMS and MongoDB Atlas interface were used to create the database and all tables. The figure below shows a screen shot of the database and all the collections in the database.

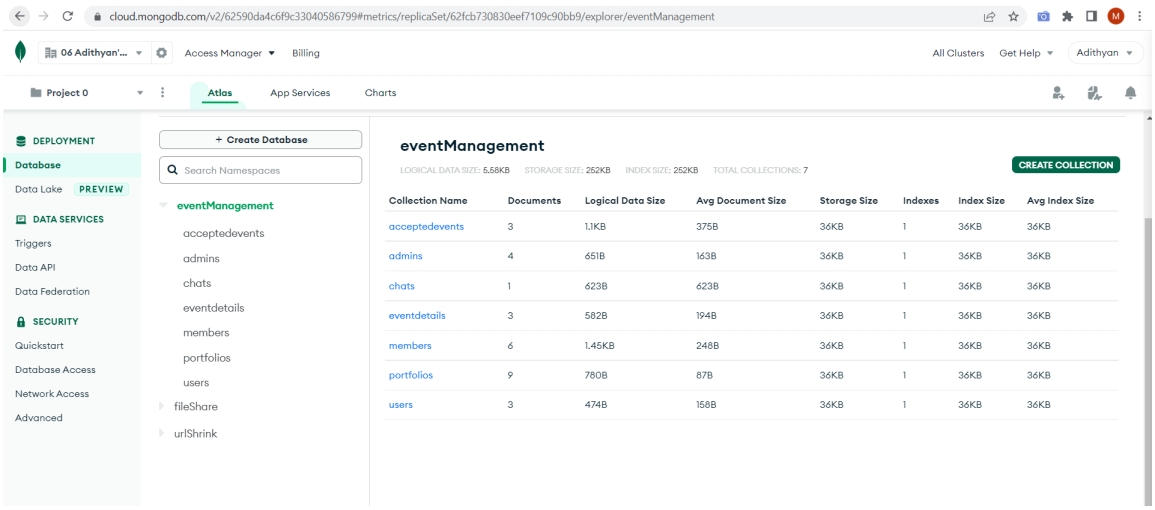


Figure 4.1: Database with all the Collections

```
JS db.js > ...
require('dotenv').config();
const mongoose = require('mongoose');

function connectDB(){
    mongoose.connect(process.env.MONGO_CONNECTION_URL,
        { useNewUrlParser: true,useUnifiedTopology: true });

    const connection=mongoose.connection;

    connection.once('open', () => {
        console.log('Database connected');
    }).on('error',(err)=> {
        console.log(err);
    });

}
module.exports = connectDB;
```

Figure 4.2: Database Connection

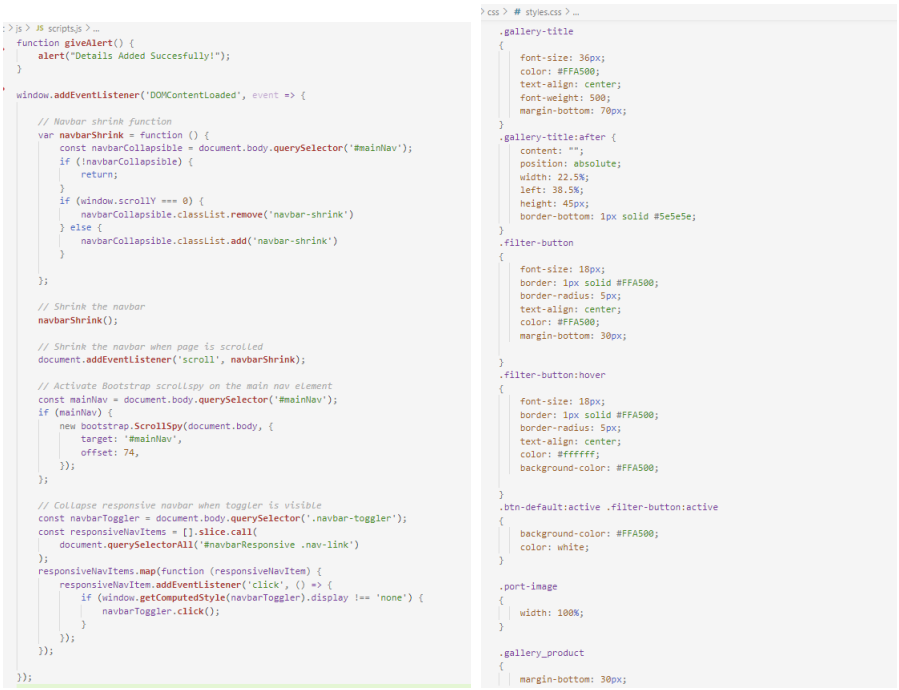
4.4 Implementation of User Interfaces

The UI of the website is developed using html5, css and javascript, Bootstrap framework is also used to make our website more responsive.

```
index.ejs > ...
<!-- Masthead-->
<header class="masthead mb-1">
    <div class="dropdown pb-5">
        <button class="dropbtn">
            <% if(user){ %>
                <%= user.name %>
            <% }else { %>
                Account
            <% } %>
        </button>
        <div class="dropdown-content">
            <% if(user){ %>
                <a href="/my-profile">My Profile</a>
                <a href="/logout">logout</a>
            <% }else { %>
                <a href="/login">login</a>
            <% } %>
        </div>
    </div>
    <div class="container">
        <div class="masthead-subheading">Avenue Event Planners</div>
        <div class="masthead-heading text-uppercase">It's Nice To Meet You</div>
        <a class="btn btn-primary btn-xl text-uppercase" href="tel:928-816-8563">Call Now</a>
    </div>
</header>
<section class="page-section" style="background-color: #ff99bb;">
    <div class="container">
        <div class="text-center pt-2">
            <h2 class="" style="font-family: 'Satisfy',cursive;">Avenue</h2>
            <h2 class="section-heading text-uppercase">Event Planning & Design</h2>
            <div class="row pt-5 section-para">
            <p style="text-align: center;">The Avenue Hospitality experience is unlike any other.
            <p style="text-align: center;">With us, customization is guaranteed. From the very beg
            <p style="text-align: center;">When it's showtime, our staff will work tirelessly to g
            </div>
        </div>
        <div class="d-flex justify-content-center pt-3">
            <a href="/about" class="btn btn-info btn-xl" >FIND OUT MORE</a>
        </div>
    </div>
</section>
```

```
<section class="page-section p-5 bg-light" id="services">
    <div class="container">
        <div class="text-center pt-5">
            <h2 class="section-heading text-uppercase" >Services</h2>
        </div>
        <div class="row text-center pb-5" style="color: #093a15">
            <div class="row col-lg-4 d-flex justify-content-center pt-5">
                <h3 class="serviceHeading">Social Gathering</h3>
                <a href="/socialGathering"><img src="/images/social-gathering.jpg" class="sqs-blo
            </div>
            <div class="row col-lg-4 d-flex justify-content-center pt-5">
                <h3 class="serviceHeading">Wedding</h3>
                <a href="/wedding"><img src="/images/weddingService.jpg" class="sqs-block-image b
            </div>
            <div class="row col-lg-4 d-flex justify-content-center pt-5">
                <h3 class="serviceHeading">Corporate Events</h3>
                <a href="/corporateEvent"><img src="/images/corporateEvent.jpg" class="sqs-block-
            </div>
        </div>
    </div>
    <div style="height: 100px;"></div>
</section>
<!--get in touch-->
<section class="page-section bg-info getInTouch-section">
    <div class="container ">
        <div class="row">
            <div class="section-para getInTouch-para pt-4">
                <p class="d-flex justify-content-center">Ready to start planning an event that is
                <p class="d-flex justify-content-center">Let's work together.</p>
            </div>
            <div class="d-flex justify-content-center pt-3">
                <a href="/contact" class="btn btn-xl btn-primary text-center">GET IN TOUCH</a>
            </div>
        </div>
    </div>
</section>
```

Figure 4.3: Implementation of home page



```
js > JS scripts.js > ...
function giveAlert() {
    alert("Details Added Succesfully!");
}

window.addEventListener('DOMContentLoaded', event => {

    // Navbar shrink function
    var navbarShrink = function () {
        const navbarCollapsible = document.body.querySelector('#mainNav');
        if (!navbarCollapsible) {
            return;
        }
        if (window.scrollY === 0) {
            navbarCollapsible.classList.remove('navbar-shrink')
        } else {
            navbarCollapsible.classList.add('navbar-shrink')
        }

    };

    // Shrink the navbar
    navbarShrink();

    // Shrink the navbar when page is scrolled
    document.addEventListener('scroll', navbarShrink);

    // Activate Bootstrap scrollspy on the main nav element
    const mainNav = document.body.querySelector('#mainNav');
    if (mainNav) {
        new bootstrap.ScrollSpy(document.body, {
            target: '#mainNav',
            offset: 74,
        });
    };

    // Collapse responsive navbar when toggler is visible
    const navbarToggler = document.body.querySelector('.navbar-toggler');
    const responsiveNavItems = [].slice.call(
        document.querySelectorAll('#navbarResponsive .nav-link')
    );
    responsiveNavItems.map(function (responsiveNavItem) {
        responsiveNavItem.addEventListener('click', () => {
            if (window.getComputedStyle(navbarToggler).display !== 'none') {
                navbarToggler.click();
            }
        });
    });

});
```

```
css > # styles.css > ...
.gallery-title
{
    font-size: 36px;
    color: #FFA500;
    text-align: center;
    font-weight: 500;
    margin-bottom: 70px;
}
.gallery-title:after {
    content: "";
    position: absolute;
    width: 22.5%;
    left: 38.5%;
    height: 45px;
    border-bottom: 1px solid #5e5e5e;
}
.filter-button
{
    font-size: 18px;
    border: 1px solid #FFA500;
    border-radius: 5px;
    text-align: center;
    color: #FFA500;
    margin-bottom: 30px;

}
.filter-button:hover
{
    font-size: 18px;
    border: 1px solid #FFA500;
    border-radius: 5px;
    text-align: center;
    color: #ffffff;
    background-color: #FFA500;

}
.btn-default:active .filter-button:active
{
    background-color: #FFA500;
    color: white;
}

.port-image
{
    width: 100%;
}

.gallery_product
{
    margin-bottom: 30px;
```

Figure 4.4: JavaScripts and CSS Implementation

4.5 Implementation of Business Logic(BackEnd)

The backend handle everything that doesn't involve providing a user interface. This can include writing APIs, building libraries, and utilities.Nodejs is used as a language for the backend. Express.js which is a framework for nodejs is used building the website and APIs.

```
const express = require('express')
const expressLayout = require('express-ejs-layouts')
const path=require('path')
const indexRouter=require('./routes/index')
const adminRouter=require('./routes/admin')

const app=express()
const http = require('http');
const server = http.createServer(app);
const { Server } = require("socket.io");
const io = new Server(server);

const bodyParser = require('body-parser')
var session=require('express-session')
var fileUpload=require('express-fileupload')

app.use(bodyParser.urlencoded({
    extended: true
}));
app.use(bodyParser.json());
app.use(session({secret:"key",cookie:{maxAge:600000}}))

const PORT = process.env.PORT || 3000

app.use(expressLayout)

app.set('layout','./layouts/layout')
app.set('view engine','ejs')
app.use(fileUpload())
app.use('/',indexRouter)
app.use('/admin',adminRouter)
app.use(express.static(path.join(__dirname, 'public')));

const connectDB = require('./config/db')
connectDB()

//chat implementation....
```

```
const ChatDataModel = require("./models/chatdata")
app.get('/chatwindow/:eventid/:user',async(req,res)=>{
    res.render('members/chat',
    { roomName: req.params.eventid,user:req.params.user })
})

io.on('connection',(socket)=>{
  console.log("connected...");
  socket.on('join',async(room)=>{
    let result = await ChatDataModel.findOne({"roomId":room})
    if(!result){
      await ChatDataModel.create({"roomId":room,messages:[]})
    }
    result = await ChatDataModel.findOne({"roomId":room})
    socket.join(room)
    socket.emit("joined",room,result)
    socket.activeRoom=room
  })
  socket.on('message',async(room,msg)=>{
   let msgdata={
     user:msg.user,
     message:msg.message
   }
   await ChatDataModel
     .updateOne({roomId:room},
     {
       $push:{messages:msgdata}
     })
    socket.to(room).emit('message',msg)
  })
})

//chat implementation....

server.listen(PORT,()=>{
    console.log(`Listening on port ${PORT}`)
})
```

Figure 4.5: BackEnd Server.js

```
const eventDetailModel = require("../models/eventdetails")
const UserDetailModel = require("../models/userdetails")
const bcrypt=require('bcrypt')
const EventDetailModel = require("../models/eventdetails")
const MemberDetailModel = require("../models/memberdetails")
const PortfolioDetailModel = require("../models/portfoliodetails")
const { reject } = require("bcrypt/promises")
const AcceptedEventModel = require("../models/acceptedEvent")
var ObjectId=require('mongodb').ObjectID



module.exports={
    addDetails:(detail,userId)=>{...
    },
    doSignup:(data)=>{...
    },
    doLogin:(data)=>{...
    },
    doMemberLogin:(data)=>{...
    },
    getPortfolio:()=>{...
    },
    getMembersEvent:(member)=>{...
    },
    getUsersAcceptedEvent:(user)=>{...
    },
    getMember:(memId)=>{...
    },
    changeEmail:(memId,newEmail)=>{...
    },
    changePassword:(memId,passData)=>{...
    }
}
```

```
const eventDetailModel = require("../models/eventdetails")
const AdminDetailModel = require("../models/admindetails")
const AcceptedEventModel = require("../models/acceptedEvent")
const UserDetailModel = require("../models/userdetails")
const MemberDetailModel = require("../models/memberdetails")
const PortfolioDetailModel = require("../models/portfoliodetails")
const ChatDataModel = require("../models/chatdata")
const bcrypt = require('bcrypt')
var ObjectId=require('mongodb').ObjectID


module.exports={
    doLogin:(data)=>{...
    },
    doSignup:(data)=>{...
    },
    getAllDetails:()=>{...
    },
    getEventDetails:(userId,eventId)=>{...
    },
    getAllMembers:()=>{...
    },
    acceptDetail:(data)=>{...
    },
    getAcceptedDetail:()=>{...
    },
    addMemberDetails:(data)=>{...
    },
    getScheduleDetail:(dataId)=>{...
    },
    addPortfolio:(data)=>{...
    },
    deleteEvent:(dataId,eventId)=>{...
    },
    deleteSchedule:(id)=>{...
    }
}
```

Figure 4.6: Backend Helper Functions

4.6 Implementation of Mailing Service

Mail service is also included in the Online Event Management System. SendInBlue mail service is used to sent mail to the client, members and the admin. It's one of the best tools for trigger-based and transactional emails. Its automation workflow designer allows you to build campaigns triggered by clicks, opens and even webpage visits.

```
> JS emailService.js > ...
const nodemailer = require('nodemailer')
async function sendMail({from,to,subject,text,html}){
    let transporter=nodemailer.createTransport({

        service:'SendinBlue',
        auth:{
            user:process.env.MAIL_USER,
            pass:process.env.MAIL_PASSWORD
        }
    });
    let info=await transporter.sendMail({
        from: `Avenue Event Management <${from}>`,
        to:to,
        subject:subject,
        text:text,
        html:html
    })
}
module.exports = sendMail;
```

Figure 4.7: Mailing service.js

Chapter 5

RESULTS AND DISCUSSIONS

5.1 Results

5.1.1 Home Page

This is the first interface for the Avenue event management system. This interface can be accessed by all the system users and has information about what has to be done to access the different system functionalities

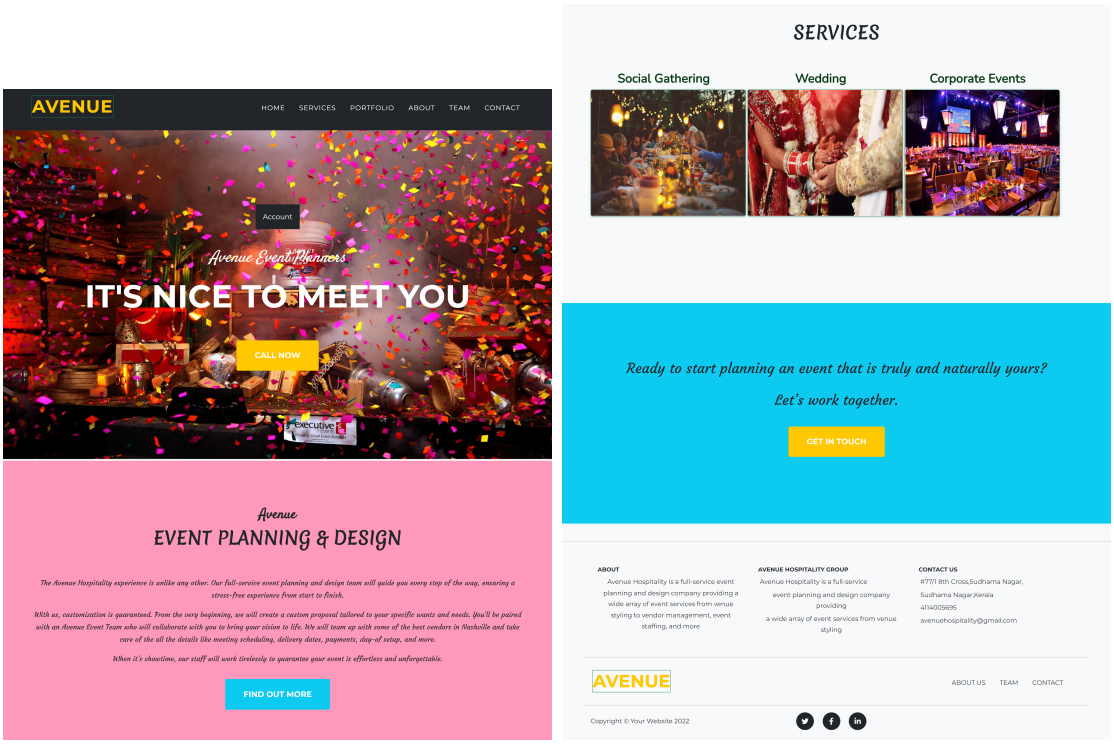


Figure 5.1: Home Page

5.1.2 Client Login/Signup Page

The client must login into his account for booking any events. If he does'nt have an account, he must create one by signing up.

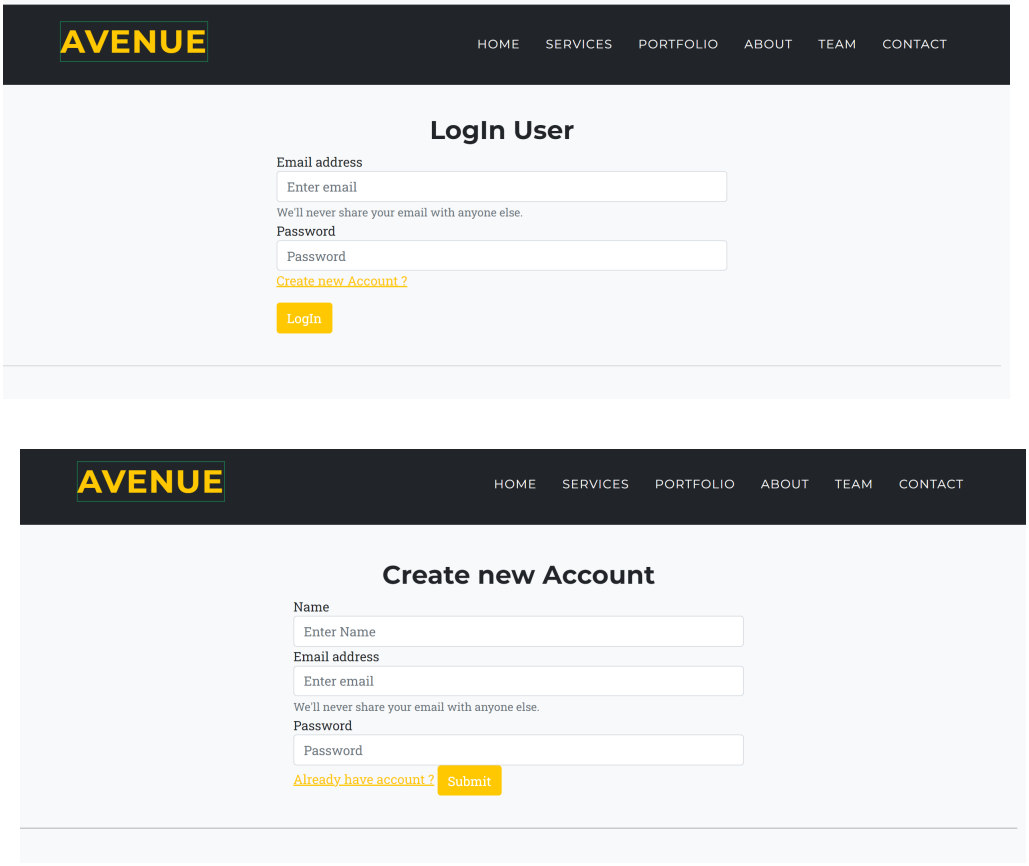


Figure 5.2: Login/Signup Page

5.1.3 Client Booking Page

The client will login after signing up and make a booking through this page as indicated in the figure below.

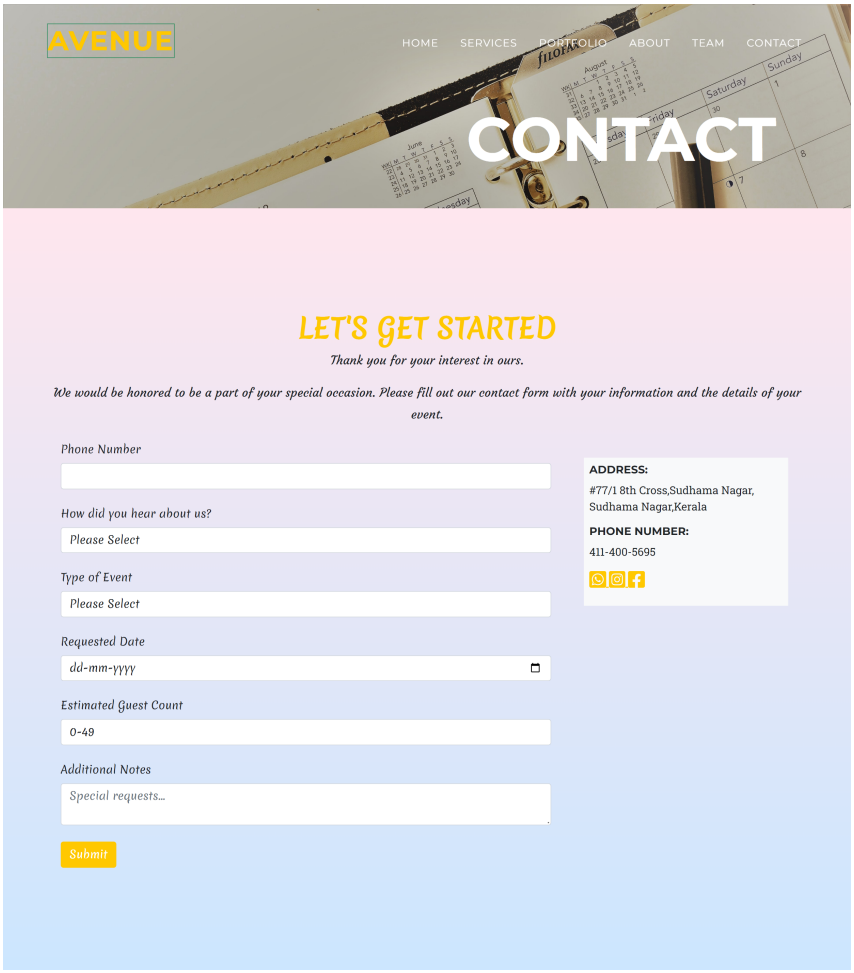
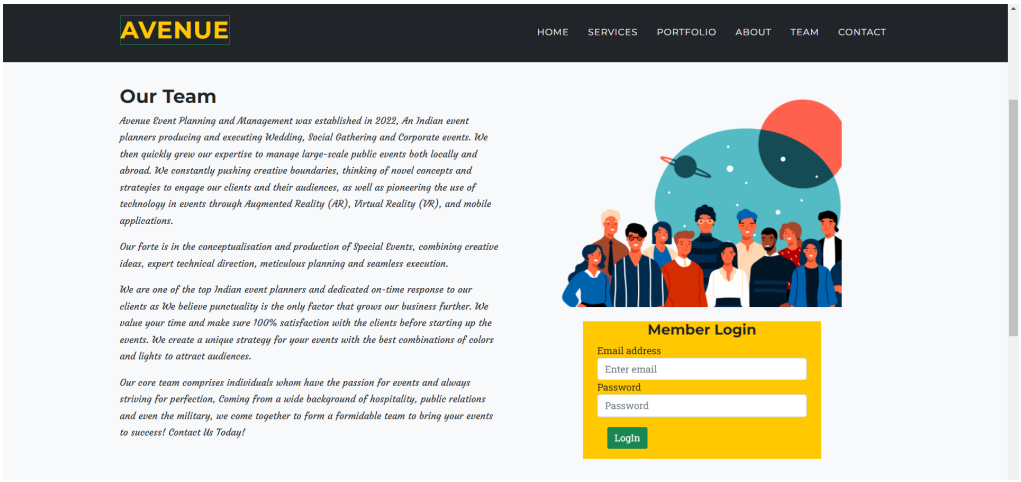


Figure 5.3: Booking Page

5.1.4 Event Planners Login and Account

Event planners can login to their account where he can view his events and chat with his clients. He can also edit his profile details int the member account page.



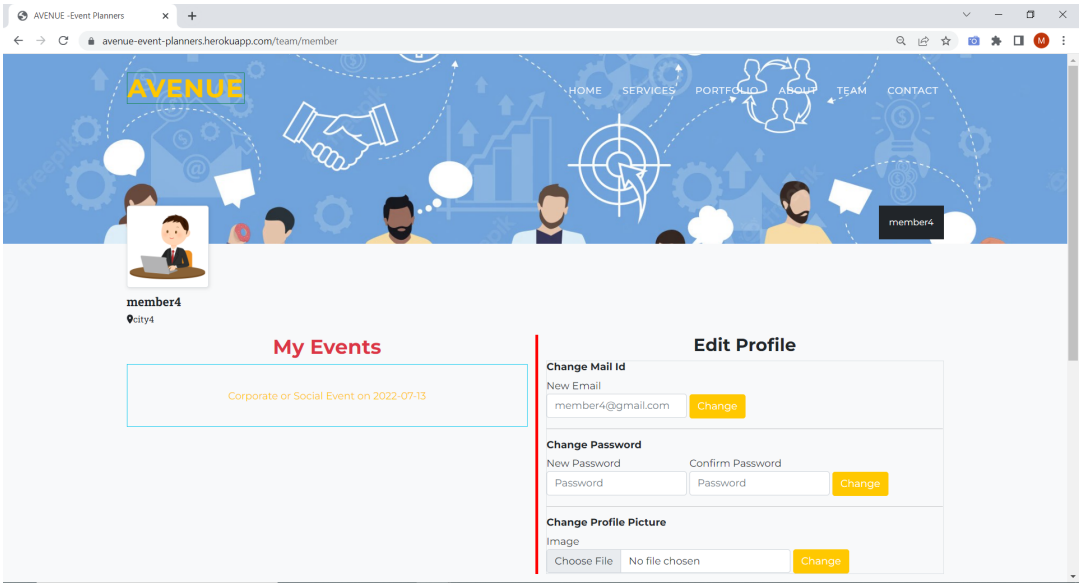


Figure 5.4: Event Planners Login and Account Pages

5.1.5 Admin Login Interface

Admin Login page is available in ' /admin ' route ,i.e, "BASE-URL/admin".Admin login details are matched with the login credentials already saved in the database.

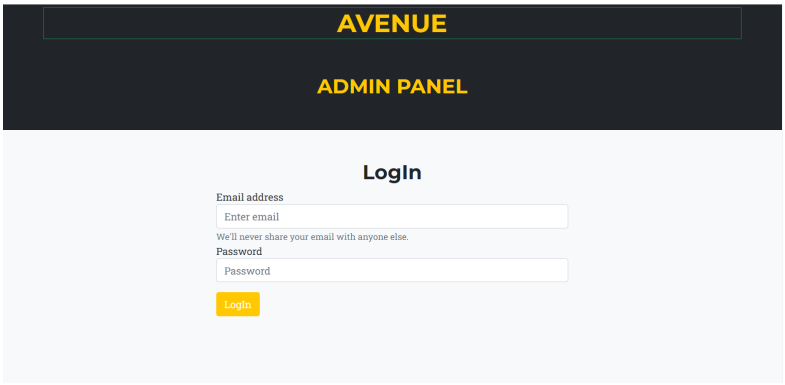


Figure 5.5: Admin Login Page

5.1.6 Admin Panel Interface

When the Username and Password is correct and the user privilege level is Administrator, the user is then directed to the admin main switch board as indicated below.

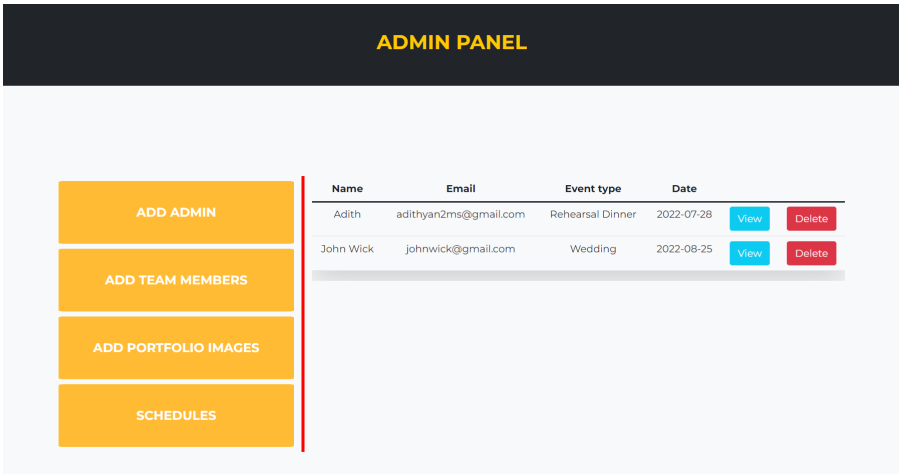


Figure 5.6: Admin Panel

5.1.7 Admin Handling Requested Events

Admin views all the event requests and accepts or rejects them based on certain circumtances like unavailability of members ,frank details etc.He also sets the event handling team before accepting the event.

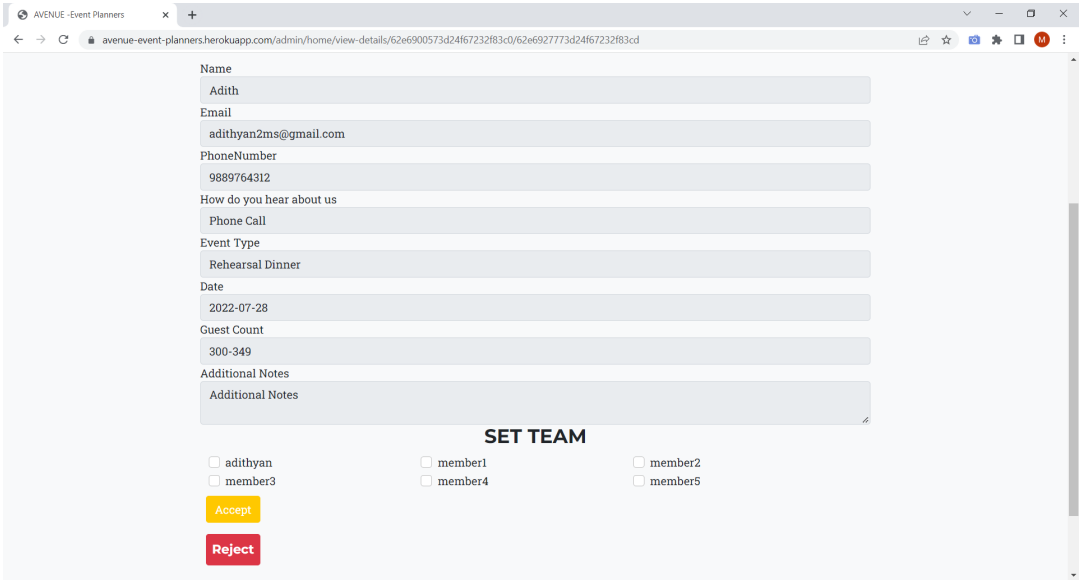


Figure 5.7: Admin handling event request

5.1.8 Scheduled Events

Events accepted by the admin are shown here containing the event details and the team members.

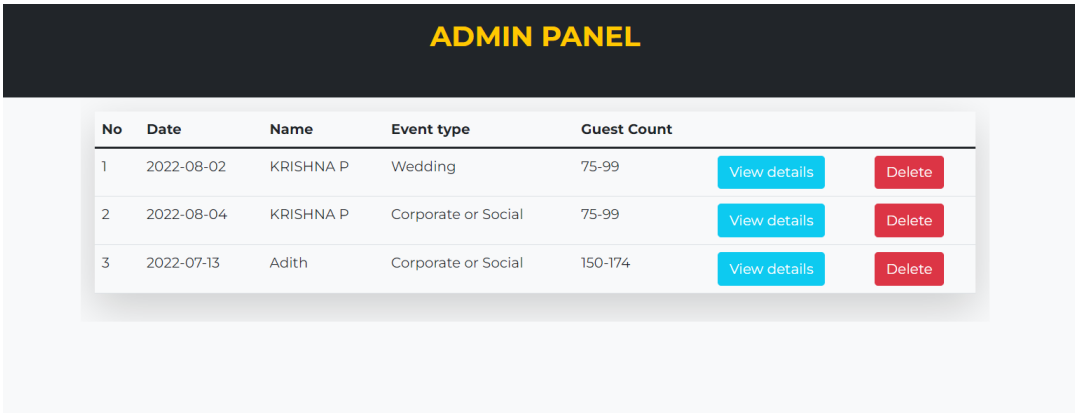


Figure 5.8: Scheduled Event Page

5.1.9 Adding Event Planners

Admin adds new event planners details. After adding the details the login creditionals for the member's account are mailed to the planner's mailid.

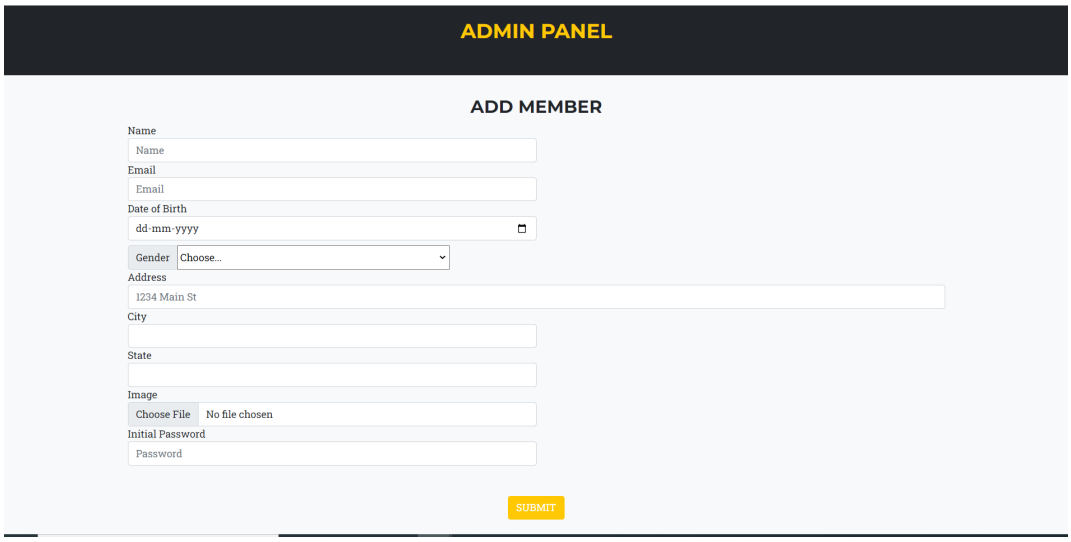


Figure 5.9: Adding Members Page

5.1.10 Adding Portfolio Images

Portfolio Images for the website added by the admins so that new event images can be updated occasionally.

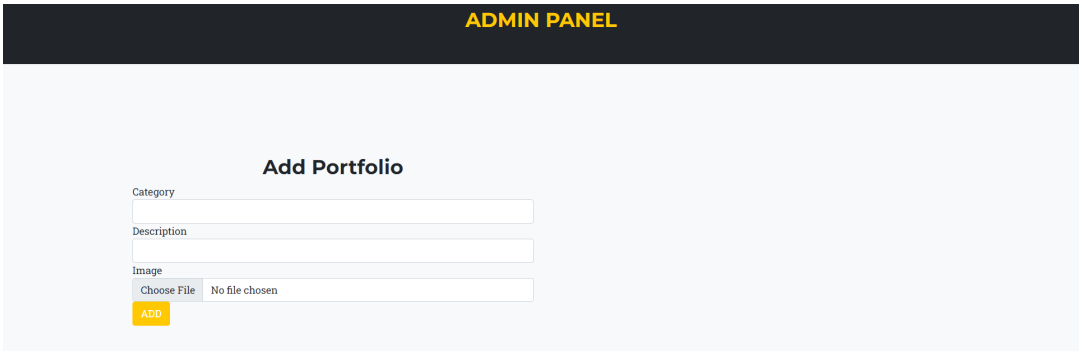


Figure 5.10: Add Portfolio images Page

5.1.11 Chat Page

Once the event has been scheduled the client can chat with the event planners using the chat interface.

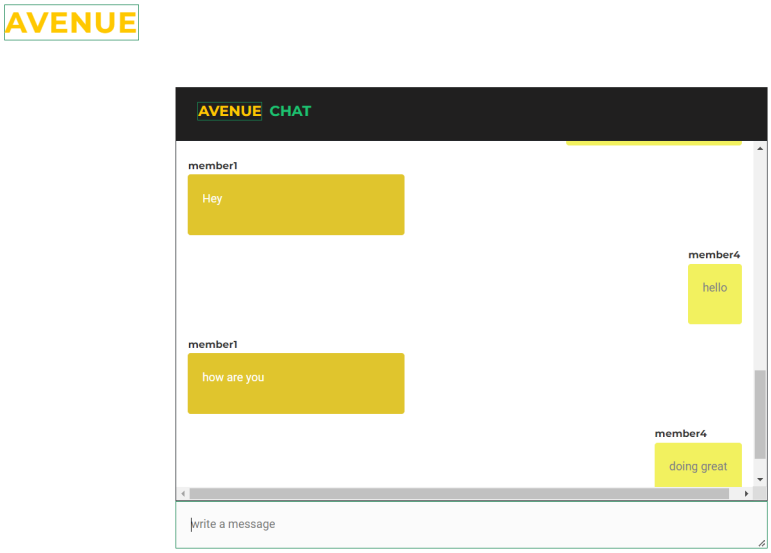


Figure 5.11: Chat Page

5.2 Discussion of Results

The main objective of the research project was to design and develop an online event management system. The project aim was automating the processes of booking and handling the events at Avenue Event Managem through the Design and Development of an online event management system.

A number of steps are summarized in terms of the four specific objectives. The achievement of these specific objectives is explained below;

The first specific objective was to review the current event managements system. This was done in chapter two where a number of literatures relating to the research problem were reviewed which helped in identifying the gaps in the related work that was already done by other researchers

The requirements of the proposed system i.e. user and system requirements were collected and analyzed using different methodologies as discussed in chapter three. The proposed system requirements and the system designs are also explained in the same chapter.

The third specific objective was to develop the online Event Management System. The System was developed basing on the designs presented in chapter 3, Software tools like NodeJS, MongoDB, HTML ,CSS and JavaScript were used in the development process.

All the activities mentioned above were done with the main aim of achieving what was proposed.

Chapter 6

CONCLUSION

This project set out to design and develop an online event management system that would help in carrying out online event scheduling / booking. For reliability, effectiveness, efficiency to be realized, the company will need to adopt the developed system. This online event management system could be implemented at several places requiring to manage different events.All the requirements and possibilities were taken care during the project development.

REFERENCES

[1] "Event management - A special kind of project management " (Peter J.A Reusch, Pascal Reusch,.) published in 2019.

[2] "Development of a mobile based birth and funeral event planning application in Bahrain " (Ehab Juma Adwan, Manal Zulfiqar, Jeehan Malik, Ayesha Arif. [2021])

[3] "Factors Affecting Customer Satisfaction with Ecommerce Websites - An Omani Perspective " (Shouvik Sanyala , Mohammed Wamique Hisamb. [2019])

[4] "Transforming a website from desktop to mobile a cross platform viewpoint " (Leeladevi B, Pinku Hazarika, Hemadri Pavan Kumar Nanyan [2020])

[5] "Twitter bootstrap and AngularJS: Frontend frameworks to expedite science gateway development " (Viknes Balasubramanee, Chathuri Wimalasena, Raminder Singh,Marlon Pierce. [2022])

[6] "Web Development and performance comparison of Web Development Technologies in Node.js and Python " (Sai Sri Nandan Challapalli, Prakarsh Kaushik, Shashikant Suman, Basu Dev Shivahare, Vimal Bibhu, Amar Deep Gupta. [2021])

[7] "Express supervision system based on NodeJS and MongoDB " (Li Liang, Ligu Zhu, Wenqian Shang, Dongyu Feng, Zida Xiao. [2019])

[8] "Study on Website Search Engine Optimization " (Zhou Hui, Qin Shingang, Liu Jinhua, Chen Jianli [2020])