

Chapter 1

INTRODUCTION

EventZee is a full stack web application that helps users know about all the ongoing and upcoming events. It lets the users see the details about a particular event such as the venue, its timings, entry fees and other details.

The front end of the application is built using JavaScript, HTML5 and CSS3. HTML5 is used to give structure to the web pages of our application. CSS3 is used to give style to the web pages and JavaScript is used to add logic and DOM manipulation to the web pages. JQuery is a JavaScript library used for DOM manipulation.

The back end of the application that is built using Firebase Real time database

Chapter 2

PROJECT REQUIREMENTS

2.1 Hardware

- Processor : Intel i5 2.4GHz, 64bit processor
- Ram : 4GB RAM
- Hard Disk : 50GB
- Networking technology : Ethernet / Wireless Ethernet

2.2 Software

- Operating System : WINDOWS / MAC / LINUX
- Programming Languages : HTML5, CSS3, JavaScript
- Libraries : Font-Awesome, JQuery, aos
- Database : Firebase Realtime database
- Web browser : Google Chrome, IE8+, Mozilla Firefox

Chapter 3

LITERATURE SURVEY

3.1 Current market

Locating events and viewing the type of the event, prizes offered and images from home helps the user pick an event he would want to attend depending on his interests, without actually visiting the place. This way the user is also able to events near him that he never knew were happening.

Online event viewing and booking portals are flourishing in India as people prefer to save time and pick the event based on what kind of events he is interested in, instead of going to every place and exploring it one by one. These portals allow users to select the class of events he wants to look for maybe either cultural, technical or theatrical events. They then select a particular event and view the description and various details of these particular events. The user can decide to attend the event based on the type of event and prizes offered.

3.2 Front-end Technology

3.2.1 JavaScript

JavaScript, often abbreviated as JS, is a high-level, dynamic, weakly typed, prototype-based, multi-paradigm, and interpreted programming language. JavaScript on the front-end is used for DOM manipulation and AJAX.

3.2.2 HTML5

HTML5 is the latest version of Hypertext Mark-up Language that is used to define the structure of web pages. Hypertext Mark-up Language is the standard mark-up language for creating web pages and web applications.

3.2.3 CSS3

CSS or Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a mark-up language. CSS3 is the latest evolution of the Cascading Style Sheets language and aims at extending CSS2.1.

3.2.4 JQuery

JQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. It is free, open-source software using the permissive MIT License.

3.3 Database technology

3.3.1 Firebase Realtime database

Firebase provides a real-time database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. The company provides client libraries that enable integration with Android, iOS, JavaScript, Java, Objective-C.

Chapter 4

SYSTEM DESIGN

4.1 Database Design

Table used in this project is:

- **USER_MESSAGES:** Information about the users accessing the application which include
 - Company Name
 - Email ID
 - Message
 - Name
 - Phone

4.2 Server design

The server program of our application is built using JavaScript upon the Firebase realtime database . The server program holds event handles for all the routes in the application to which requests can be sent. The program then processes these requests and responds accordingly. The server has the ability to connect with the firebase database server using the mysql package.

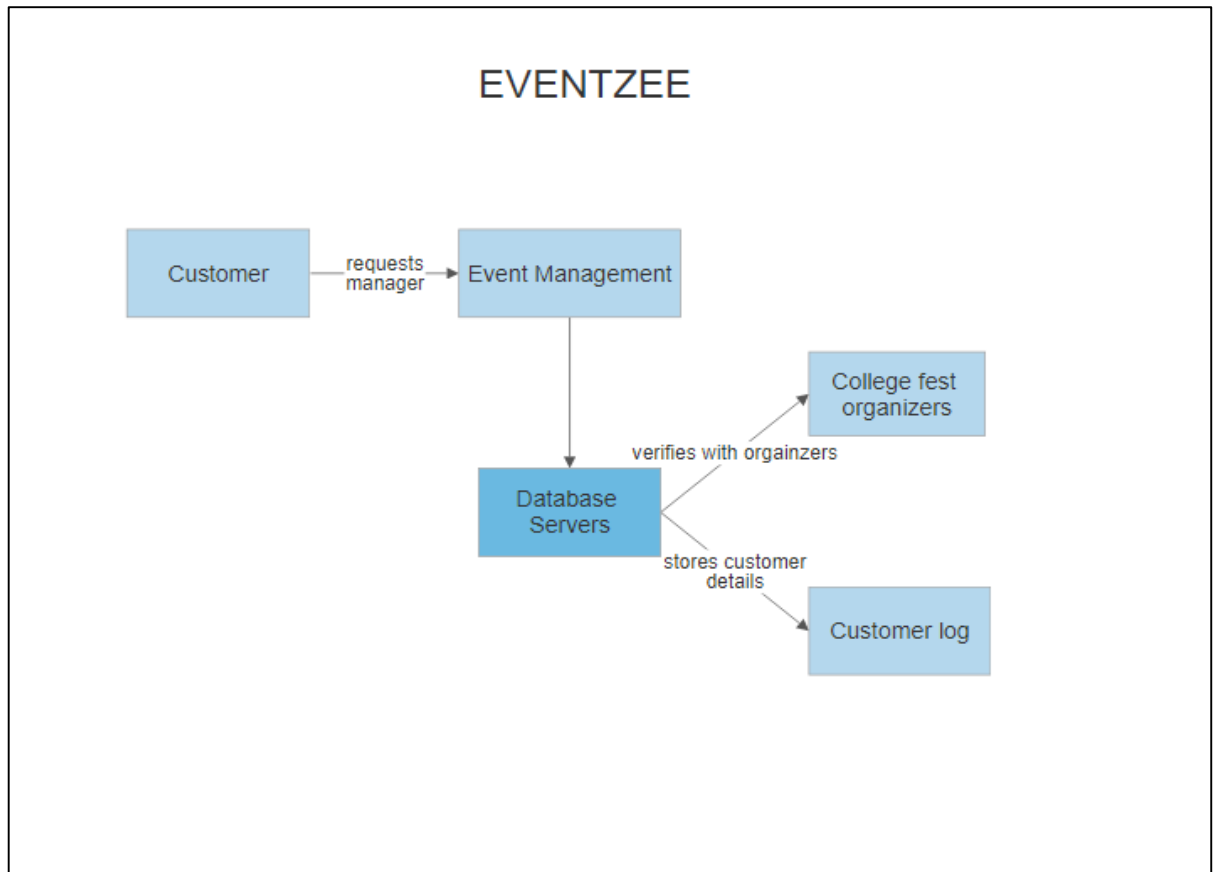


Fig 4.1 Dataflow Diagram

Chapter 5

IMPLEMENTATION

5.1 Main Page

5.1.1 Index

```
<!DOCTYPE html>
<html>
<link rel="icon" href="img/logo.png" type="image/png" sizes="16x16">
<head>
  <title>Web Mini project</title>

  <link rel="stylesheet" type="text/css" href="try.css">
  <link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">
  <script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>

  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
  awesome/4.7.0/css/font-awesome.min.css">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://fonts.googleapis.com/css?family=Quicksand" rel="stylesheet">
  <link href="https://fonts.googleapis.com/css?family=Varela" rel="stylesheet">
<body>

<script>
  AOS.init();
</script>

<header>
  <nav>
    <div class="topnav" id="tt">
```

```

    <a href="#home" class="active">Home</a>
    <a href="#portion5">Major Events</a>
    <a href="EventList.html">Events List</a>
    <a href="contact.html">Contact Us</a>
</div>
</nav>
</header>

<div class="portion" id="home">

    <div class="vd">
        <video id="vdid" preload="auto" autoplay="true" loop="loop" muted="muted">
            <source src="img/vid.mp4" type="video/mp4" style="overflow-x: hidden;">
            video not possible
        </video>
    </div>

    <h2></h2>

</div>
</div>
    <div class="button">
        <button class="collap">Details</button>
        <div class="content">
            <p>
                <div class="button3">
                    <p style="padding-bottom: .2em;"><b>Organisers :</b></p>
                    <a href="#portion6">Hrudhai</a>
                    <a href="tel:xxxxxxxx">xxxxxxxx</a>
                </div>
                <div class="button2">
                    <p style="padding-bottom: 0.2em;"><b>Detials : </b></p>
                    <a>1st Prize : Rs 4500</a>
                    <a>2nd Prize : Rs 2500</a>

```



```
</div>
</div>
</div>
</script>
</body>
</html>
```

5.2 Events list

5.2.1 Technical Events

```
<!DOCTYPE html>
<html>
<link rel="icon" href="img/logo.png" type="image/png" sizes="16x16">
<head>
  <title>Technical Events</title>
  <link rel="stylesheet" type="text/css" href="tryeve.css">
  <link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">
  <script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>

  <link href="https://fonts.googleapis.com/css?family=Quicksand" rel="stylesheet">
  <link href="https://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">
  <div class="info" style="text-align: left;font-size: 1.5em; background-color: rgba(189,
197, 255, 0.16)">
    <h1>Technical Events</h1>
  </div>
  <div class="portion1 info">
```

5.3 Contact

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Contact EVENTZEE</title>
```

```
<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-  
awesome.min.css" rel="stylesheet" integrity="sha384-  
wvfXpqpZZVQGK6TAh5PVlGOfQNHSoD2xbE+QkPxCAFINEEvoEH3SIl0sibVcOQV  
nN" crossorigin="anonymous">  
<link rel="stylesheet"  
href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/3.5.2/animate.css" />  
<link rel="stylesheet" href="style.css">  
</head>  
<body>  
  <div class="container">  
    <h1 class="brand"><span>EVENTZEE</span> Contact Form</h1>  
    <div class="wrapper">  
      <div class="company-info">  
        <h3>EVENTZEE</h3>  
        <ul>  
          <li><i class="fa fa-road"></i> Hosur Rd, Konappana Agrahara, Electronic City,  
Bengaluru, Karnataka</li>  
          <li><i class="fa fa-phone"></i> 080661 86611</li>  
          <li><i class="fa fa-envelope"></i> contact@eventzee.com</li>  
        </ul>  
      </div>  
      <div class="contact">  
        <h3>Email Us</h3>  
        <div class="alert">Your message has been sent</div>  
        <form id="contactForm">  
          <p>  
            <label>Name</label>  
            <input type="text" name="name" id="name" required>  
          </p>  
          <p>  
            <label>Company</label>  
            <input type="text" name="company" id="company">  
          </p>  
          <p>  
            <label>Email Address</label>
```

```
<input type="email" name="email" id="email" required>
</p>
<p>
  <label>Phone Number</label>
  <input type="text" name="phone" id="phone">
</p>
<p class="full">
  <label>Message</label>
  <textarea name="message" rows="5" id="message"></textarea>
</p>
<p class="full">
  <button type="submit">Submit</button>
</p>
</form>
</div>
</div>
</div>

<script src="https://www.gstatic.com/firebasejs/4.3.0/firebase.js"></script>
<script src="https://www.gstatic.com/firebasejs/6.4.0/firebase-database.js"></script>
<script src="main.js"></script>
</body>
</html>
```

5.4 Server

```
var config = {
  apiKey: "AIzaSyBcKUy9py-r3o1t5EzciQaUUyVSdCc9mOQ",
  authDomain: "webminiproject-7e2a7.firebaseio.com",
  databaseURL: "https://webminiproject-7e2a7.firebaseio.com",
  projectId: "webminiproject-7e2a7",
  storageBucket: "webminiproject-7e2a7.appspot.com",
  messagingSenderId: "175885322876",
  appId: "1:175885322876:web:b2b54fe03113b1034257f9",
  measurementId: "G-YG8N62EFZ4"
};
```

```
firebase.initializeApp(config);

// Reference messages collection
var messagesRef = firebase.database().ref('messages');

// Listen for form submit
document.getElementById('contactForm').addEventListener('submit', submitForm);

// Submit form
function submitForm(e){
    e.preventDefault();

    // Clear form
    document.getElementById('contactForm').reset();
}

// Function to get get form values
function getInputVal(id){
    return document.getElementById(id).value;
}

// Save message to firebase
function saveMessage(name, company, email, phone, message){
    var newMessageRef = messagesRef.push();
    newMessageRef.set({
        name: name,
        company: company,
        email: email,
        phone: phone,
        message: message
    });

    // Listen for form submit
    document.getElementById('contactForm').addEventListener('submit', submitForm);

    // Submit form
```

```
function submitForm(e){
    e.preventDefault();
        <p class="full">
            <button type="submit">Submit</button>
        </p>
    </form>
</div>
</div>
</div>

<script src="https://www.gstatic.com/firebasejs/4.3.0/firebase.js"></script>
<script src="https://www.gstatic.com/firebasejs/6.4.0/firebase-database.js"></script>
<script src="main.js"></script>
</body>
</html>
```

Chapter 6

TESTING

6.1 Unit Testing

The three units, namely, database unit, views unit and backend Node.js unit are tested individually before integrating them into one single web application.

- The database is tested through a number of DDL (Data Definition Language) and DML (Data Manipulation Language) commands in order to discover inconsistencies that arise regarding prime key constraints and other referential integrity constants.
- The views unit is primarily tested for its frontend functionality that is, for the user interface. It is ensured that the input forms function as they are required. The html pages are tested individually to avoid any discrepancies in the layout, format and style of the user interface. After linking the JavaScript page to the CSS files to obtain the required style for the web application and the JavaScript files to perform logic on the front end, tests are performed again in order to ensure no inconsistency while linking.
- The firebase connection is individually tested for proper database connectivity and to discover any errors that might arise while using the JavaScript mysql package. Queries are executed using different Statement objects with test data to ensure the right connection is established with the database and the queries yield required output.

After it is ensured that the individual unit work fine and generate the required output, they are integrated into a web application. The data that is received as input through the JavaScript is received by the server program where the request handlers perform the respective actions. The route handlers in the program have an established connection to the database where it does suitable operations on the database using the data as per the functionality selected by the user.

6.2 Integration Testing

After the three individual units have been integrated into a single web application, it is essential to check that the application works as a whole. The integration testing is started by testing that all the web pages can be accessed. Smooth transitions from one web page to the correct redirected page is also checked.

The forms are tested with input test data and verified that the input is received and processed by the server program that has established a connection with the database. The test data received as input is used to modify the database through the JavaScript - mysql API and SQL commands. This procedure is also tested to ensure correct access, insertion and update of the tables corresponding to the action performed by the user.

Chapter 7

RESULTS

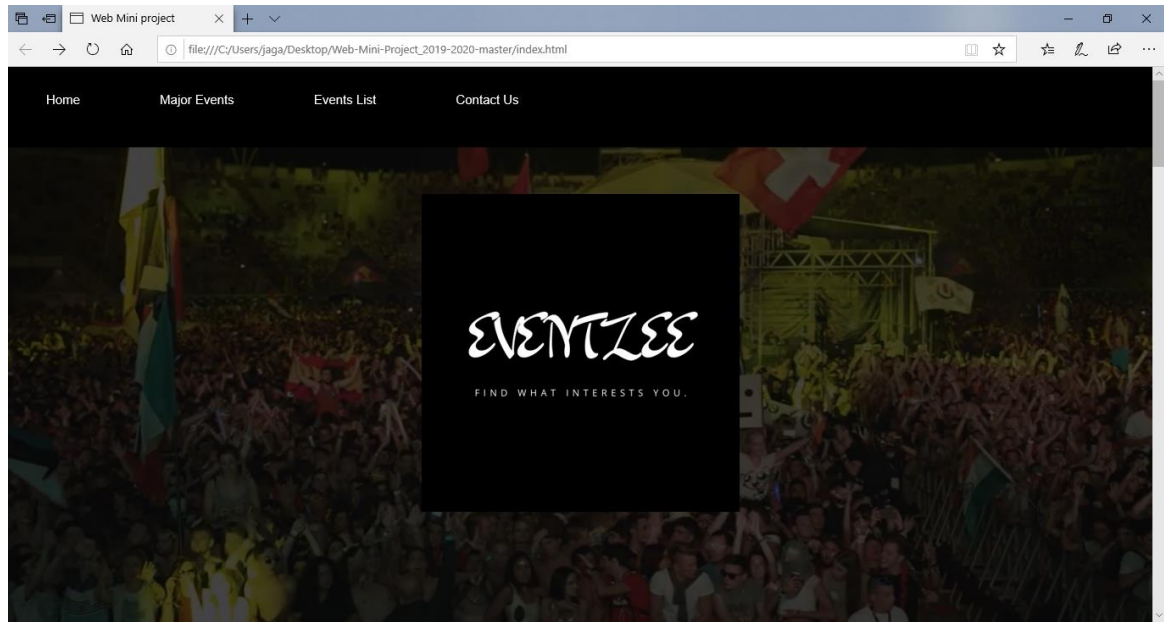


Fig 7.1 Landing page

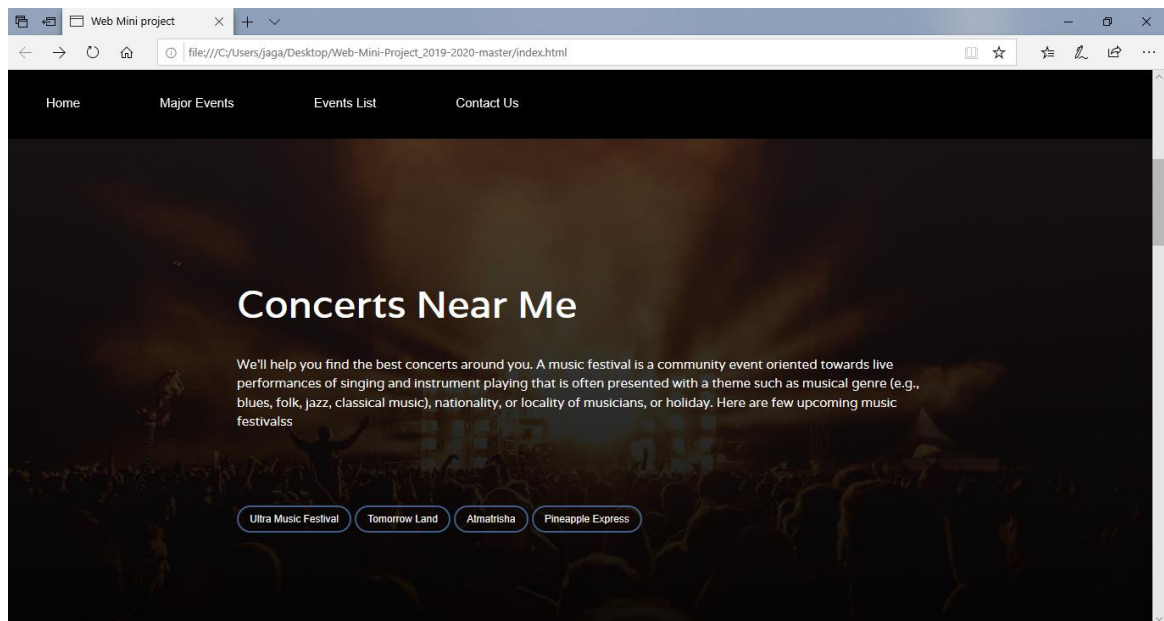


Fig 7.2 Concerts page

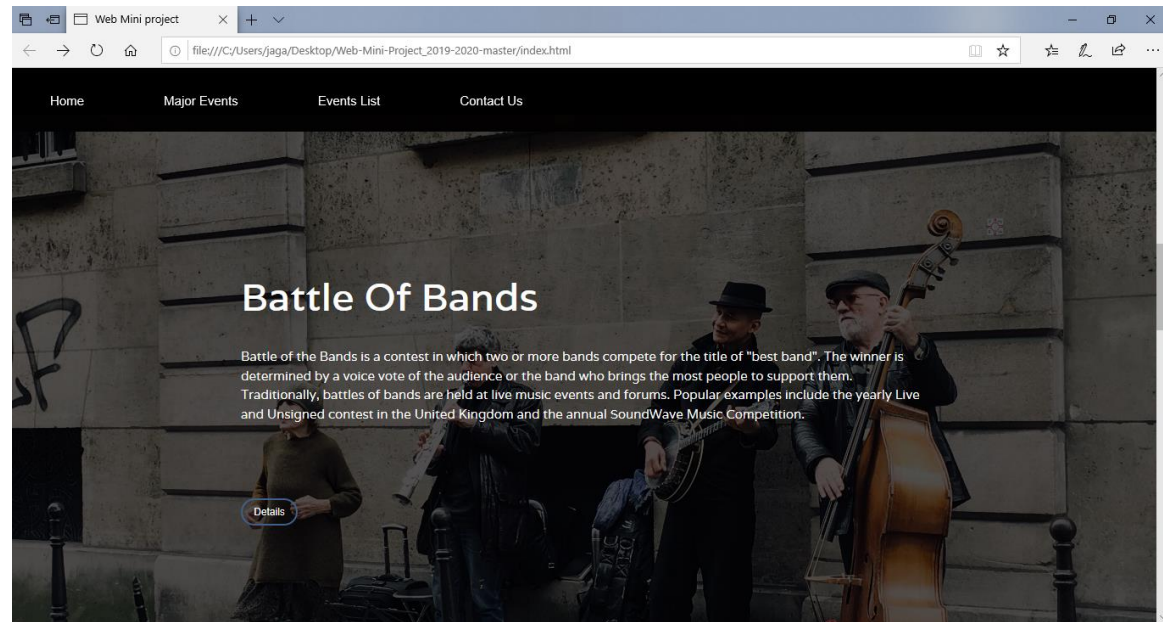


Fig 7.3 Battle of Bands Page

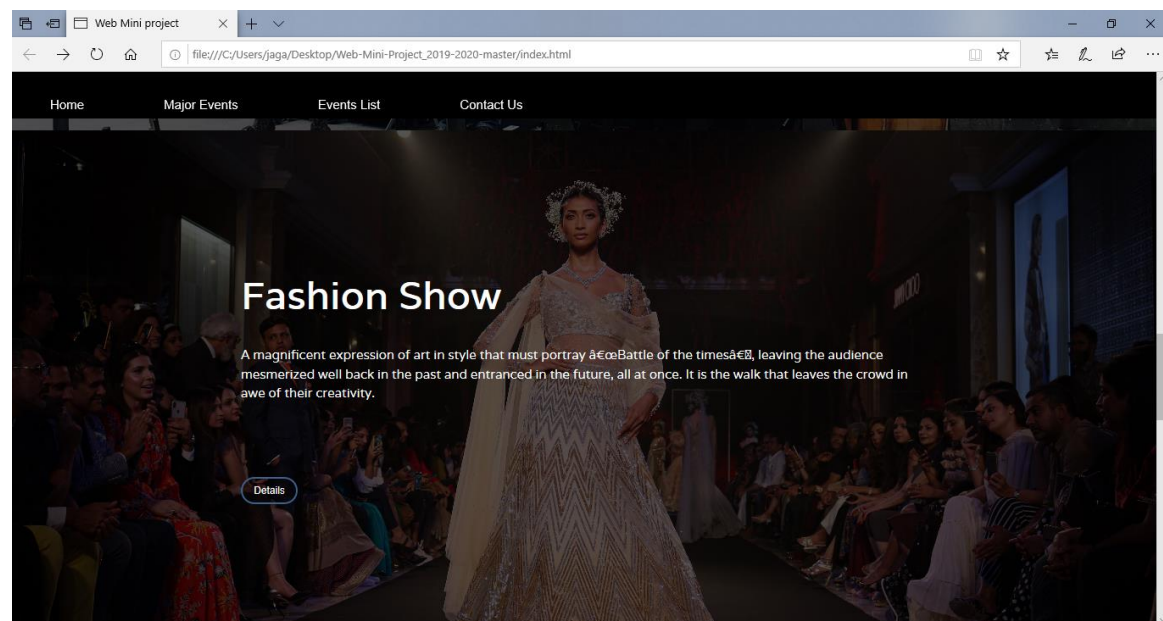


Fig 7.4 Fashion Show page

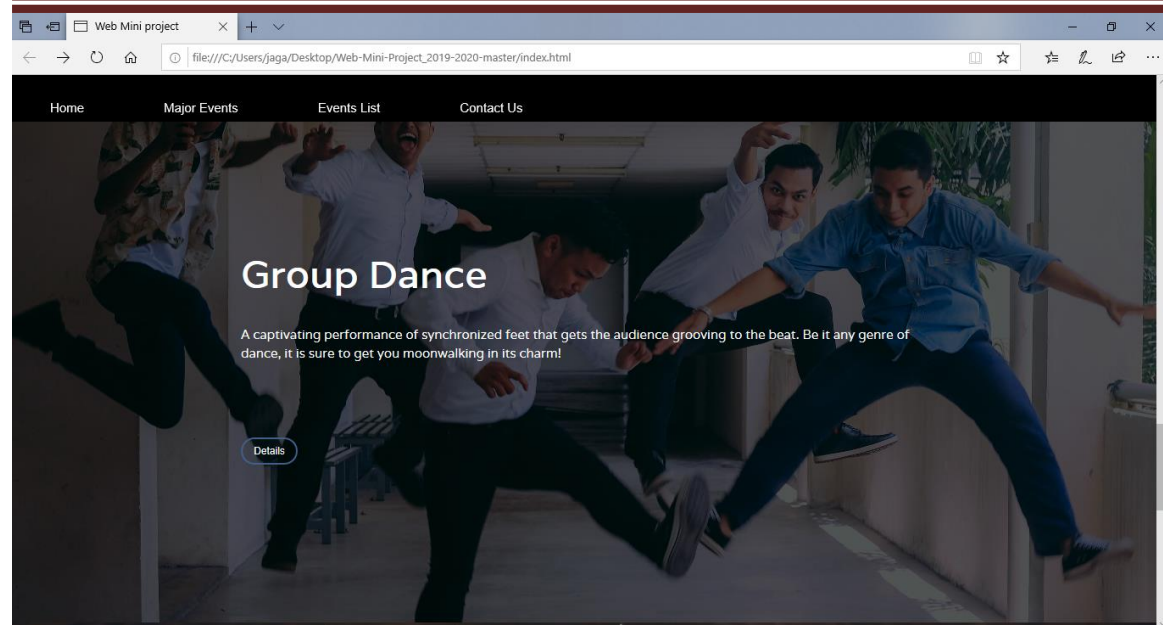


Fig 7.5 Group dance page

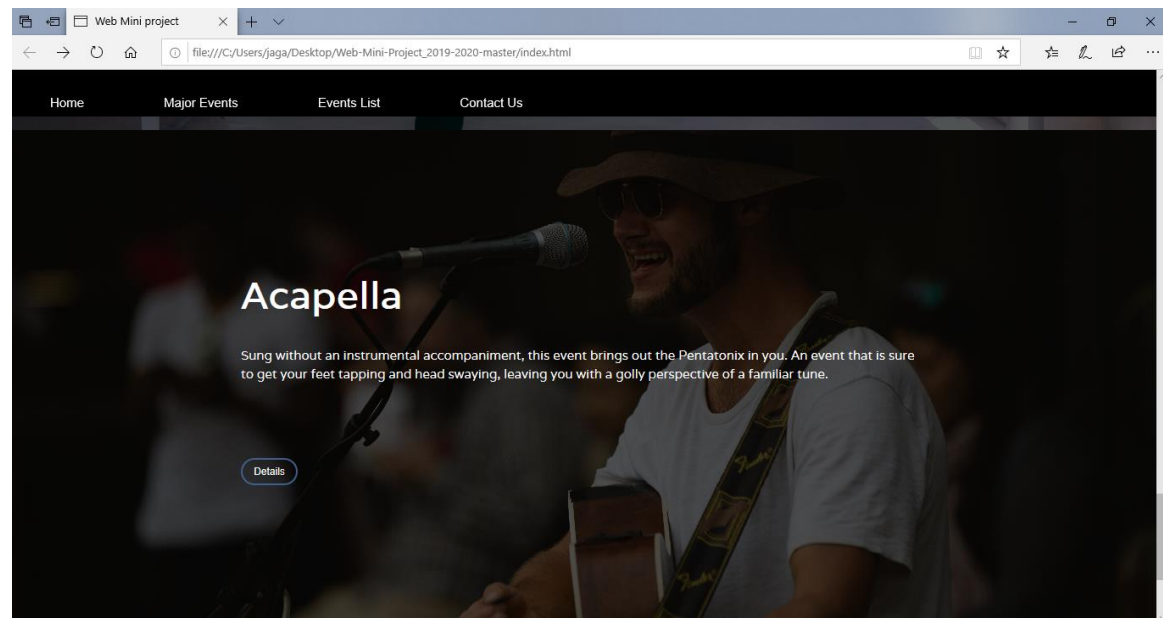


Fig 7.6 Acapella Page

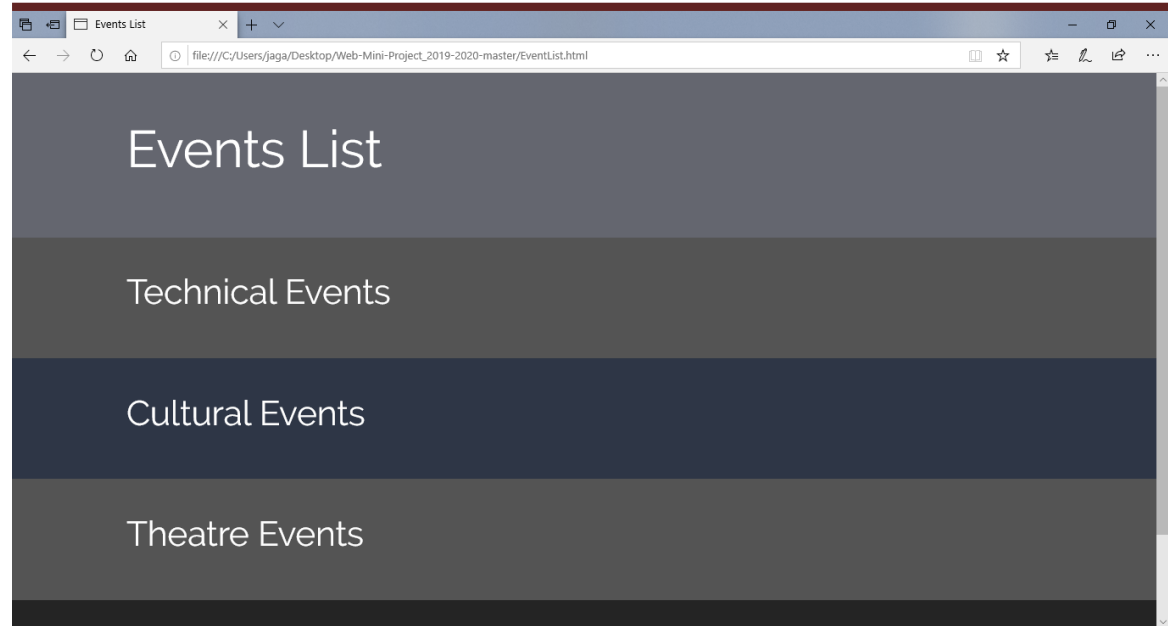


Fig 7.7 Event list

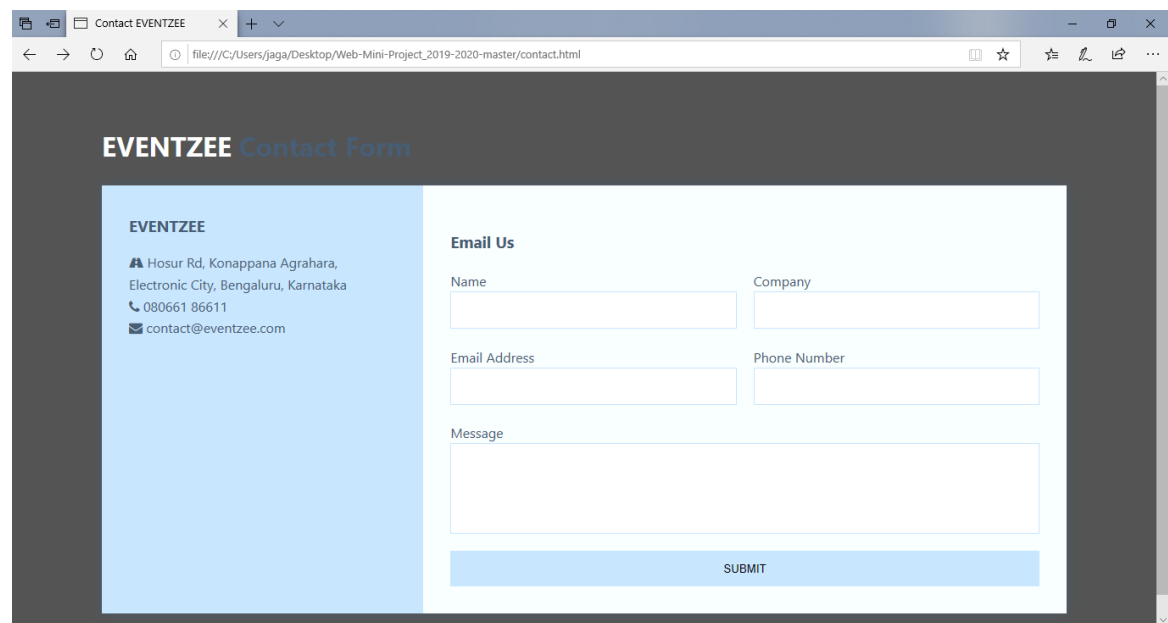


Fig 7.10 Contact us

Chapter 8

CONCLUSION

Our application currently supports the following functionalities. The user will be able to view all the ongoing and upcoming events. Once the user has assessed the events, he/she can click on one of the events to view the venue of the event, type of event, organizer's details and also a brief description of the event. User can contact the administrator or the organizer of the events by dropping a message in the website.

In the future we plan on building a system that connects our application with an Organization or Institute's cultural team so that the users can have a look upon the cultural activities happening at these different institutes organized by a team or by individual to allow more and more events to be discovered. We also plan on implementing the functionality of displaying nearby events of events of a particular location.

Chapter 9

REFERENCE

- [1] <https://developer.mozilla.org/en-US/>
- [2] <https://www.firebase.google.com>
- [3] <https://www.wikipedia.org/>
- [4] <https://www.tutorialspoint.com/mysql/>