Phase-4

**Event Management App**

Description:

Build an Angular app where the admin can view and manage employees.

**app.component.html**

<nav class="navbar navbar-expand-sm bg-light">

  <div class="container-fluid">

    <a class="navbar-brand" routerLink=""><strong>Event Manager</strong></a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav"

      aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

      <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse" id="navbarNavAltMarkup">

      <div class="navbar-nav">

        <a class="nav-item nav-link" routerLink="add">

          Add Event

        </a>

        <a class="nav-item nav-link" routerLink="update">

          Update Event

        </a>

        <a class="nav-item nav-link" routerLink="">

          List Event

        </a>

        <a class="nav-item nav-link" routerLink="login">

          Login

        </a>

        <a class="nav-item nav-link" routerLink="register">

          Register

        </a>

      </div>

    </div>

  </div>

</nav>

<div class="container">

  <router-outlet></router-outlet>

  <footer>

    <aside class="footer-left">

      <strong>

        <p>Register Your Event with Us, to stay connected!</p>

      </strong>

      <strong>Address: </strong>

      <p>Event Manager, Opposite to National Public School, <br />

        Kengeri, Bengaluru, Karnataka, India - 560060</p>

    </aside>

    <aside class="footer-right">

      <strong>Reach us directly through the below map location!</strong><br/>

      <iframe

      src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3888.773034783152!2d77.48679681408944!3d12.922304019456188!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3bae3edeb452e2db%3A0xe38b8e47f689ba2d!2sNational%20Public%20School%20Kengeri!5e0!3m2!1sen!2sin!4v1665222955709!5m2!1sen!2sin"

      width="300" height="110" style="border:10;" allowfullscreen="" loading="faster"

      referrerpolicy="no-referrer-when-downgrade"></iframe>

    </aside>

    <aside class="footer-center">

      <strong>Contact Details: </strong><br />

      <strong>Mobile: </strong>

      <p>7019748596</p>

      <!-- <strong>Landline: </strong>

      <p>08642-260125</p> -->

      <strong>e-mail:</strong><br/>

      <i class="fa fa-envelope" style="font-size:20px">eventmanger2022@gmail.com</i><br/><br/>

      <p>All rights reserved by

      <i class="fa fa-copyright" style="font-size:16px">Eventmanger2022</i>

    </p>

    </aside>

  </footer>

</div>

**event.service.ts**

import { Injectable } from '@angular/core';

import {HttpClient} from '@angular/common/http'

@Injectable({

  providedIn: 'root'

})

export class EventService {

url = "http://localhost:3000/event"

rootUrl = "http://localhost:3000/"

  constructor(private http: HttpClient) { }

  getList()

  {

    return this.http.get(this.url);

  }

  saveEvent(data: any)

  {

    // console.warn("Service", data);

    return this.http.post(this.url,data)

  }

  deleteEvent(id: any){

    return this.http.delete(`${this.url}/${id}`)

  }

  getCurrentEvent(id: any)

  {

    return this.http.get(`${this.url}/${id}`)

  }

  updateEvent(id: any, data: any)

  {

    return this.http.put(`${this.url}/${id}`,data)

  }

  registerUser(data: any){

    return this.http.post(this.rootUrl+ "Users", data);

  }

  loginUser(data: any){

    return this.http.get(this.rootUrl+ "Users", data);

  }

}

**add-event.component.html**

<h1>Add New Event</h1>

<div \*ngIf="alert" class="alert alert-success alert-dismissible fade show" role="alert">

    <strong>Success!</strong> Your data has been entered successfully!

    <button (click)="closeAlert()" type="button" class="close" data-dismiss="alert" aria-label="Close">

      <span aria-hidden="true">&times;</span>

    </button>

  </div>

<form [formGroup]="addEvent" (ngSubmit)="collectEvent()">

    <div class="form-group">

      <label>Name</label>

      <input type="text" class="form-control" placeholder="Enter Name" formControlName = "name">

    </div>

    <div class="form-group">

        <label>email</label>

        <input type="email" class="form-control" placeholder="Enter Mail Address" formControlName="email">

    </div>

    <div class="form-group">

        <label>Address</label>

        <input type="text" class="form-control" placeholder="Enter Address" formControlName="address">

    </div><br/>

    <button type="submit" class="btn btn-primary">Submit</button>

  </form>

**add-event.component.ts**

import { Component, OnInit } from '@angular/core';

import {FormGroup, FormControl} from '@angular/forms';

import { EventService } from '../event.service';

@Component({

  selector: 'app-add-event',

  templateUrl: './add-event.component.html',

  styleUrls: ['./add-event.component.css']

})

export class AddEventComponent implements OnInit {

  alert:boolean = false

  addEvent = new FormGroup({

    name: new FormControl(''),

    email: new FormControl(''),

    address: new FormControl(''),

  })

  constructor(private event:EventService) { }

  ngOnInit(): void {

  }

  collectEvent()

  {

    this.event.saveEvent(this.addEvent.value).subscribe((result) =>{

      this.alert=true

      this.addEvent.reset({})

    })

  }

  closeAlert()

  {

    this.alert=false

  }

}

**list-event.component.ts**

import { Component, NgModule, OnInit } from '@angular/core';

import { EventService } from '../event.service';

@Component({

  selector: 'app-list-event',

  templateUrl: './list-event.component.html',

  styleUrls: ['./list-event.component.css']

})

export class ListEventComponent implements OnInit {

  constructor(private event:EventService) { }

  collection: any = {};

  ngOnInit(): void {

    this.event.getList().subscribe((result: any) => {

      console.warn(result)

      this.collection = result

    })

  }

  deleteEvent(item: any){

    this.collection.splice(item-0, 1)

    this.event.deleteEvent(item).subscribe((result) => {

      console.warn("result", result);

    })

  }

}

**login-event.component.ts**

import { Component, OnInit } from '@angular/core';

import {FormGroup, FormControl} from '@angular/forms';

import { EventService } from '../event.service';

@Component({

  selector: 'app-login-event',

  templateUrl: './login-event.component.html',

  styleUrls: ['./login-event.component.css']

})

export class LoginEventComponent implements OnInit {

  alert:boolean = false

  login = new FormGroup({

    email: new FormControl(''),

    password: new FormControl(''),

  })

  constructor(private event:EventService) { }

  ngOnInit(): void {

  }

  collection(){

    console.warn(this.login.value)

    this.event.loginUser(this.login.value).subscribe((result)=> {

      this.alert = true

      console.warn("result", result)

    } )

  }

  closeAlert(){

    this.alert = false

  }

}

**register-event.component.ts**

import { Component, OnInit } from '@angular/core';

import {FormGroup, FormControl} from '@angular/forms';

import { EventService } from '../event.service';

@Component({

  selector: 'app-register-event',

  templateUrl: './register-event.component.html',

  styleUrls: ['./register-event.component.css']

})

export class RegisterEventComponent implements OnInit {

  alert:boolean = false

  register = new FormGroup({

    name: new FormControl(''),

    email: new FormControl(''),

    password: new FormControl(''),

  })

  constructor(private event:EventService) { }

  ngOnInit(): void {

  }

  collection(){

    console.warn(this.register.value)

    this.event.registerUser(this.register.value).subscribe((result) => {

      this.alert = true

      console.warn("result", result)

    })

  }

  closeAlert() {

    this.alert = false

  }

}

**update-event.component.ts**

import { Component, OnInit } from '@angular/core';

import {FormGroup, FormControl} from '@angular/forms';

// import { EventService } from '../event.service';

import { ActivatedRoute } from '@angular/router';

import { EventService } from '../event.service';

@Component({

  selector: 'app-update-event',

  templateUrl: './update-event.component.html',

  styleUrls: ['./update-event.component.css']

})

export class UpdateEventComponent implements OnInit {

  alert:boolean = false;

  editEvent = new FormGroup({

    name: new FormControl(''),

    email: new FormControl(''),

    address: new FormControl(''),

  })

  constructor(private router: ActivatedRoute, private event:EventService) {

  }

  ngOnInit(): void {

    console.warn(this.router.snapshot.params['id']);

    this.event.getCurrentEvent(this.router.snapshot.params['id']).

    subscribe((result: any) =>

    {

      this.editEvent = new FormGroup({

        name: new FormControl(result['name']),

        email: new FormControl(result['email']),

        address: new FormControl(result['address']),

      })

    })

  }

  collection()

  {

    console.warn(this.editEvent.value);

    this.event.updateEvent(this.router.snapshot.params['id'], this.editEvent.value).subscribe((result) =>

      this.alert=true

    )

  }

  closeAlert() {

    this.alert=false

  }

}

**app-routing.module.ts**

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import {AddEventComponent} from './add-event/add-event.component';

import { ListEventComponent } from './list-event/list-event.component';

import { LoginEventComponent } from './login-event/login-event.component';

import { RegisterEventComponent } from './register-event/register-event.component';

import { UpdateEventComponent } from './update-event/update-event.component';

const routes: Routes = [

  {

    component: AddEventComponent,

    path: 'add'

  },

  {

    component: LoginEventComponent,

    path: 'login'

  },

  {

    component: UpdateEventComponent,

    path: 'update/:id'

  },

  {

    component: RegisterEventComponent,

    path: 'register'

  },

  {

    component: ListEventComponent,

    path: ''

  }

];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

**app.component.html**

<nav class="navbar navbar-expand-sm bg-light">

  <div class="container-fluid">

    <a class="navbar-brand" routerLink=""><strong>Event Manager</strong></a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav"

      aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

      <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse" id="navbarNavAltMarkup">

      <div class="navbar-nav">

        <a class="nav-item nav-link" routerLink="add">

          Add Event

        </a>

        <a class="nav-item nav-link" routerLink="update">

          Update Event

        </a>

        <a class="nav-item nav-link" routerLink="">

          List Event

        </a>

        <a class="nav-item nav-link" routerLink="login">

          Login

        </a>

        <a class="nav-item nav-link" routerLink="register">

          Register

        </a>

      </div>

    </div>

  </div>

</nav>

<div class="container">

  <router-outlet></router-outlet>

  <footer>

    <aside class="footer-left">

      <strong>

        <p>Register Your Event with Us, to stay connected!</p>

      </strong>

      <strong>Address: </strong>

      <p>Event Manager, Opposite to National Public School, <br />

        Kengeri, Bengaluru, Karnataka, India - 560060</p>

    </aside>

    <aside class="footer-right">

      <strong>Reach us directly through the below map location!</strong><br/>

      <iframe

      src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3888.773034783152!2d77.48679681408944!3d12.922304019456188!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3bae3edeb452e2db%3A0xe38b8e47f689ba2d!2sNational%20Public%20School%20Kengeri!5e0!3m2!1sen!2sin!4v1665222955709!5m2!1sen!2sin"

      width="300" height="110" style="border:10;" allowfullscreen="" loading="faster"

      referrerpolicy="no-referrer-when-downgrade"></iframe>

    </aside>

    <aside class="footer-center">

      <strong>Contact Details: </strong><br />

      <strong>Mobile: </strong>

      <p>7019748596</p>

      <!-- <strong>Landline: </strong>

      <p>08642-260125</p> -->

      <strong>e-mail:</strong><br/>

      <i class="fa fa-envelope" style="font-size:20px">eventmanger2022@gmail.com</i><br/><br/>

      <p>All rights reserved by

      <i class="fa fa-copyright" style="font-size:16px">Eventmanger2022</i>

    </p>

    </aside>

  </footer>

</div>

**app.component.ts**

import { Component } from '@angular/core';

@Component({

  selector: 'app-root',

  templateUrl: './app.component.html',

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'Event\_Manager';

}

**app.module.ts**

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

import { AddEventComponent } from './add-event/add-event.component';

import { UpdateEventComponent } from './update-event/update-event.component';

import { HttpClientModule } from '@angular/common/http';

import { ListEventComponent } from './list-event/list-event.component';

import { LoginEventComponent } from './login-event/login-event.component';

import { RegisterEventComponent } from './register-event/register-event.component';

import { ReactiveFormsModule } from '@angular/forms';

@NgModule({

  declarations: [

    AppComponent,

    AddEventComponent,

    UpdateEventComponent,

    ListEventComponent,

    LoginEventComponent,

    RegisterEventComponent

  ],

  imports: [

    BrowserModule,

    HttpClientModule,

    AppRoutingModule,

    ReactiveFormsModule

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule {}