**Movie.component.ts**

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

import { moviedataformat, usersdata } from './data.model';

import { MovieserviceService } from '../Services/movieservice.service';

@Component({

  selector: 'app-movieproject',

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

  styleUrls: ['./movieproject.component.css']

})

export class MovieprojectComponent implements OnInit {

  medicinesdata :moviedataformat[]=[]

  searcht:any

  image=true

  payment=false

  login=true

  invalid=false

  takedetailsvar=false

  cartvar=false

  userdata :any=[]

  constructor(private service:MovieserviceService) { }

  ngOnInit(): void {

    this.getmedicines()

    this.getusers()

  }

  onchecklogin(logindata:{email:string,password:string})

  {

    for(const i in this.userdata){

      if(this.userdata[i].email==logindata.email && this.userdata[i].password==logindata.password)

      {

        this.login=false

        this.service.userinfo=this.userdata[i]

        this.service.id=this.userdata[i].id

      }

      else{

        this.invalid=true

      }

    }

  }

  getmedicines(){

    this.service.getmedicines()

    .subscribe((data)=>{

      this.medicinesdata=data

    })

  }

  getusers(){

    this.service.getusersfromdb()

    .subscribe((data )=>{

      this.userdata=data

    })

  }

  registeruser(registerdata:usersdata){

    console.log(registerdata)

    this.takedetailsvar=false

    this.login=true

    this.service.reguser(registerdata)

    .subscribe(()=>{

      this.getusers()

      this.invalid=false

    })

  }

  onregister(){

    this.takedetailsvar=true

    this.login=false

  }

  oncart(){

    this.cartvar=true

  }

}

**Movie.component.html**

<div class="home">

    <div class="w3-bar w3 w3-red">

      <nav class="navbar navbar-expand-sm bg-yellow navbar-dark w3-large py-sm-5">

              <div class="topnav-centered">

               <h1 class="active"  style="font-size:70px;">Movie 🎬🎥</h1>

              </div>

      </nav>

      </div>

</div>

  <section class="vh-100" \*ngIf="login" >

  <div class="container py-5 h-100">

    <div

      class="row d-flex justify-content-center align-items-center h-100">

      <div class="col-12 col-md-8 col-lg-6 col-xl-5">

        <div class="card shadow-2-strong" style="border-radius: 1rem;">

          <div class="card-body p-5 text-center">

            <h3 class="mb-5">User Login</h3>

            <h3 class="mb-5" \*ngIf="invalid">

              <p style="color:red">Invalid</p>

            </h3>

            <h3 class="mb-5" \*ngIf="invalid">

              <p style="color: green"> Didn't Register Before?<br>Please Do Register</p>

            </h3>

            <form #f="ngForm">

              <div class="form-outline mb-4">

                <input name="email" type="email" id="typeEmailX-2"

                  class="form-control form-control-lg" ngModel required />

                  <label

                  class="form-label" for="typeEmailX-2">Email</label>

              </div>

              <div class="form-outline mb-4">

                <input name="password" type="password" id="typePasswordX-2"

                  class="form-control form-control-lg" ngModel required />

                  <label

                  class="form-label" for="typePasswordX-2">Password</label>

              </div>

              <button class="btn btn-primary btn-lg btn-block" type="submit"

              [disabled]="!f.valid"

              (click)="onchecklogin(f.value)">Login</button>

              <br>

              <p>

                Don't have an account? <button class="btn btn-primary btn-sm " (click)="onregister()">Register

                  here</button>

              </p>

              <hr class="my-4">

              </form>

          </div>

        </div>

      </div>

    </div>

  </div>

</section>

  <div \*ngIf="!payment && !login && !takedetailsvar && !cartvar">

    <br><br><br> <br><br><br><br><br><br>

    <div class="search">

      <input type="text" class="search"  size="50" name="searcht" placeholder="Search"

        [(ngModel)]="searcht">

    </div>

    <div>

    <app-moviesingleproduct \*ngFor="let i of medicinesdata | filter :searcht" [x]="i"></app-moviesingleproduct>

    </div>

  </div>

  <br><br><br><br><br>

  <div class="con"\*ngIf="!payment && !login && !takedetailsvar && !cartvar">

    <div class="row">

        <div class="col">

            <button class="btn btn-outline-primary btn-lg" (click)="oncart()" style="size:0cqmax;" >

              Cart 🛒

            </button>

        </div>

    </div>

  </div>

  <app-moviecart \*ngIf="cartvar"></app-moviecart>

<div class="container my-5 py-5" \*ngIf="takedetailsvar">

  <div class="row d-flex justify-content-center py-5">

      <div class="col-md-7">

  <div class="card">

      <div class="card-body p-4">

<form #f="ngForm">

  <div class="row mb-4">

    <div class="col">

      <div class="form-outline">

          <label class="form-label" for="form6Example1"> Name</label>

        <input type="text" name="name" id="form6Example1" class="form-control" size="8" ngModel required/>

      </div>

    </div>

    <div class="col">

      <div class="form-outline">

          <label class="form-label" for="form6Example2">Age</label>

        <input type="text" name="age" id="form6Example2" class="form-control"ngModel required />

      </div>

    </div>

  </div>

  <div class="form-outline mb-4">

    <label class="form-label" for="form6Example6">Phone</label>

  <input type="number" name="phone" id="form6Example6" class="form-control" ngModel required/>

</div> <div class="form-outline mb-4">

  <label class="form-label" for="form6Example6">Email</label>

<input type="email" name="email" id="form6Example6" class="form-control" ngModel required/>

</div>

  <div class="form-outline mb-4">

      <label class="form-label" for="form6Example6">Password</label>

    <input type="password" name="password" id="form6Example6" class="form-control" ngModel required/>

  </div>

  <button type="submit" class="btn btn-primary btn-block mb-4"(click)="registeruser(f.value)"

  [disabled]="!f.valid">Register</button>

</form>

</div>

</div>

</div>

</div>

</div>

**Movie.component.css**

.center {

    margin:0 auto;

    display:block;

    left: 50%;

    float: none;

    top: 50%;

    transform: translate(12%, -50%);

  }

  .column{

    float: left;

    width:25%;

    padding: 0 10px;

  }

  .row{

    margin :0 -5px;

  }

  .row:after{

    content: "";

    clear: both;

  }

  @media screen and (max-width:600px) {

    .column{

      width: 100%;

      display: block;

      margin-bottom: 20px;

    }

  }

  .card{

    padding: 16px;

  }

  .con{

    margin: 0;

    position: absolute;

    top: 15%;

    right:0%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

  .search{

    height: 50px;

    border-radius: 15px;

    border-color: aquamarine;

    width: 700px;

    margin:0 auto;

    display:block;

    left: 50%;

    float: none;

    top: 50%;

    transform: translate(2%, -50%);

  }

  .check{

    display:grid;

    width: 30%;

    margin: 10px;

    padding: 10px;

    border: 1px solid #ccc;

    box-shadow: 0 2px 2px rgba(0,0,0,0.2);

   }

**MovieCart.component.ts**

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

import { moviedataformat } from '../data.model';

import { MovieserviceService } from 'src/app/Services/movieservice.service';

@Component({

  selector: 'app-moviecart',

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

  styleUrls: ['./moviecart.component.css']

})

export class MoviecartComponent implements OnInit {

  x:moviedataformat[]=[]

  payment=false

  totalamount=0

  constructor(private cartservice:MovieserviceService) { }

  ngOnInit(): void {

    this.loaditems()

  }

  addone(additem:moviedataformat)

  {

    this.cartservice.items.push(additem)

    this.totalamount=0

    this.loaditems()

  }

  loaditems(){

    this.x= this.cartservice.items

    for(const i in this.x){

      this.totalamount= this.totalamount+this.x[i].price

    }

    console.log(this.totalamount)

  }

  delete(i)

  {

     delete this.cartservice.items[i]

     this.totalamount=0

     for(const i in this.cartservice.items){

      if(this.cartservice.items[i]){

      }

     }

     console.log(this.cartservice.items)

     this.loaditems()

  }

  onpayment()

  {

    this.payment=true

  }

}

**MovieCart.component.html**

<h3 \*ngIf="!payment" style="left: 50%; display: block;">Your cart 🛒</h3>

    <table \*ngIf="!payment" style="width: 80rem;

    height: 8rem;">

      <tr>

      <th>

        <h1 style="color: blue;">Name</h1>

      </th>

      <th>

        <h1 style="color: blue;">Price</h1>

      </th>

      <th>

        <h1 style="color: blue;">Date</h1>

      </th>

      <th>

        <h1 style="color: blue;">Time</h1>

      </th>

      <th>

        <h1 style="color: blue;">Add one more</h1>

      </th>

      <th>

      <h1 style="color: blue;">Delete</h1>

      </th>

      </tr>

      <tr \*ngFor="let single of x,let i=index">

        <td \*ngIf="single">

          <h1 >{{single.name}}</h1>

        </td>

        <td \*ngIf="single">

          <h1>{{single.price}}</h1>

        </td>

        <td \*ngIf="single">

            <h1>{{single.showdate}}</h1>

          </td>

          <td \*ngIf="single">

            <h1>{{single.time}}</h1>

          </td>

        <td \*ngIf="single">

          <button class="btn btn-outline-success m-2" (click)="addone(single)">

            Add

         </button>

        </td>

        <td \*ngIf="single">

          <button class="btn btn-outline-danger"(click)="delete(i)">

            Delete

         </button>

        </td>

      </tr>

    </table>

    <div \*ngIf="!payment">

    <h3  style="color: green;"><br><br>Total amount {{totalamount}}</h3>

    <br><br><button class="btn btn-warning btn-lg mx-5"(click)="onpayment()">

      Checkout

   </button>

</div>

   <app-moviepayment \*ngIf="payment"></app-moviepayment>

**Moviesingleproduct.component.ts**

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

import { MovieserviceService } from 'src/app/Services/movieservice.service';

import { MovieprojectComponent } from '../movieproject.component';

import { moviedataformat } from '../data.model';

@Component({

  selector: 'app-moviesingleproduct',

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

  styleUrls: ['./moviesingleproduct.component.css']

})

export class MoviesingleproductComponent implements OnInit {

  @Input()

  x;

  id

  constructor(private main:MovieprojectComponent,private cart:MovieserviceService) { }

  ngOnInit(): void {

  }

  onpayment(){

    this.main.payment=true

    }

    addcart(x:moviedataformat){

      this.cart.addtocart(x)

      window.alert("Item added to cart")

    }

}

**Moviesingleproduct.component.html**

<div class="column" \*ngIf="x.status=='true'">

    <div class="card" style="width: 20rem; border-color: red;">

        <img src="{{x.img}}" style="height: 200px; width: 200px;" alt="...">

        <div class="card-body">

          <p style="height: 100px; width: 150px; font-size:larger;"> <b>{{x.name}}</b></p>

          <p class="card-text">

            <b>Price :</b> {{x.price}} per ticket

            <br>

           <b>Language :</b>{{x.language}}

            <br>

            <b>Show Date : </b>{{x.showdate}}

            <br>

            <b>Time :</b>{{x.time}}

            <br>

            <b>Description :</b>{{x.description}}

          </p>

         <button class="btn btn-outline-warning  mx-5" (click)="addcart(x)">

          add to cart

       </button></div>

      </div><br><br><br><br>

        </div>

**Moviesingleproduct.component.css**

#sectio{

    left: 50%;

  }

  .container{

    left:0

  }

  .center {

    margin:0 auto;

    display:block;

    left: 50%;

    float: none;

    top:30%;

    right: 50%;

    transform: translate(12%, -50%);

  }

  .vertical-center {

    margin: 0;

    position: absolute;

    top: 40%;

    left: 40%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

  .column{

    float: left;

    width:25%;

    padding: 0 10px;

  }

  .row{

    margin :0 -5px;

  }

  .row:after{

    content: "";

    clear: both;

  }

  @media screen and (max-width:600px) {

    .column{

      width: 100%;

      display: block;

      margin-bottom: 20px;

    }

  }

  /\*

  .card{

    padding: 16px;

    width: 30%;

  } \*/

  .card{

     display:grid;

     width: 30%;

     margin: 10px;

     padding: 10px;

     border: 1px solid #ccc;

     box-shadow: 0 2px 2px rgba(0,0,0,0.2);

    }

    .a{

      text-align: center;

      text-size-adjust: 50px;

      background-color: blueviolet;

    }

**Moviepayment.component.ts**

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

@Component({

  selector: 'app-moviepayment',

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

  styleUrls: ['./moviepayment.component.css']

})

export class MoviepaymentComponent implements OnInit {

  confirm=false

  constructor() { }

  ngOnInit(): void {

  }

  confirmationpage(){

    this.confirm=true

  }

}

**Moviepayment.component.html**

<section class="gradient-custom" \*ngIf="!confirm">

    <div class="container my-5 py-5">

      <div class="row d-flex justify-content-center py-5">

        <div class="col-md-7 col-lg-5 col-xl-4">

          <div class="card" style="border-radius: 15px;">

            <div class="card-body p-4">

              <form #f="ngForm">

                <div class="d-flex justify-content-between align-items-center mb-3">

                  <div class="form-outline">

                    <input type="text" id="typeText" name="cardnumber" class="form-control form-control-lg" siez="17"

                      placeholder="1234 5678 9012 3457" minlength="8" maxlength="19" ngModel required />

                    <label class="form-label" for="typeText">Card Number</label>

                  </div>

                  <img src="https://img.icons8.com/color/48/000000/visa.png" alt="visa" width="64px" />

                </div>

                <div class="d-flex justify-content-between align-items-center mb-4">

                  <div class="form-outline">

                    <input type="text" name="cardholdername" id="typeName" class="form-control form-control-lg" siez="17"

                      placeholder="Cardholder's Name" ngModel required/>

                    <label class="form-label" for="typeName">Cardholder's Name</label>

                  </div>

                </div>

                <div class="d-flex justify-content-between align-items-center pb-2">

                  <div class="form-outline">

                    <input type="text" name="expiration" id="typeExp" class="form-control form-control-lg" placeholder="MM/YYYY"

                      size="7" id="exp" minlength="7" maxlength="7" ngModel required />

                    <label class="form-label" for="typeExp">Expiration</label>

                  </div>

                  <div class="form-outline">

                    <input type="password" name="cvv" id="typeText2" class="form-control form-control-lg"

                      placeholder="&#9679;&#9679;&#9679;" size="1" minlength="3" maxlength="3" ngModel required />

                    <label class="form-label" for="typeText2">Cvv</label><br>

                  </div>

                </div>

                  <br><button type="button" class="btn btn-info btn-lg btn-rounded"

                   (click)="confirmationpage()"

                   [disabled]="!f.valid">

                    Make Payment

                  </button>

              </form>

            </div>

          </div>

        </div>

      </div>

    </div>

  </section>

  <app-movieconfirmationpage \*ngIf="confirm"></app-movieconfirmationpage>

**Moviehome.component.ts**

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

@Component({

  selector: 'app-moviehome',

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

  styleUrls: ['./moviehome.component.css']

})

export class MoviehomeComponent implements OnInit {

  home=true

  buypage=false

  adminpage=false

  cart=false

  constructor() { }

  ngOnInit(): void {

  }

  onbuypage(){

    this.buypage=true

    this.home=false

  }

  onadminpage(){

    this.adminpage=true

    this.home=false

  }

  oncart(){

    this.cart=true

  }

}

**Moviehome.component.html**

<div class="home" \*ngIf="home">

    <div class="w3-bar w3 w3-red">

      <nav class="navbar navbar-expand-sm bg-yellow navbar-dark w3-large py-sm-5">

              <div class="topnav-centered">

               <h1 class="active" style="font-size:70px;">Movie 🎬🎥</h1>

              </div>

      </nav>

      </div>

      <h6>Developed By<br>Seelam Sindhuja<br>Emp Id:10840</h6>

  <br><br><br><img

  style=" width: 700px;position:center; margin-left: auto; margin-right: auto;display: block;"

  src="data:image/jpeg;base64,">

  <br><br><br>

  </div>

  <div class="container" \*ngIf="home">

    <div class="vertical-center">

        <button  class="btn w3-border-black w3-hover-black btn-lg  m-3" (click)="onbuypage()" style="size:18px;">Buy Tickets</button>

        <button type="button" class="btn w3-border-black w3-hover-black btn-lg  m-3" (click)="onadminpage()">Admin login</button>

      </div>

</div>

<app-movieproject \*ngIf="buypage"></app-movieproject>

<app-movieadmin \*ngIf="adminpage"></app-movieadmin>

**Moviehome.component.css**

.vertical-center {

    margin: 0;

    position: absolute;

    top: 85%;

    left: 40%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

.con{

margin: 0;

position: absolute;

top: 15%;

right:0%;

-ms-transform: translateY(-50%);

transform: translateY(-50%);

}

**Movieconfirmationpage.component.ts**

.vertical-center {

    margin: 0;

    position: absolute;

    top: 85%;

    left: 40%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

.con{

margin: 0;

position: absolute;

top: 15%;

right:0%;

-ms-transform: translateY(-50%);

transform: translateY(-50%);

}

**Movieconfirmationpage.component.html**

<div class="container mt-5 mb-5">

    <div class="row d-flex justify-content-center">

        <div class="col-md-8">

            <div class="card">

                    <div class="invoice p-5">

                        <h5>{{newuserdata.name}}

                            <br>Your Ticket Confirmed!✔</h5>

                        <span class="font-weight-bold d-block mt-4">Hello, {{newuserdata.name}}</span>

                        <span>You Ticket has been confirmed! Enjoy the Movie and have fun!!!</span>

                            <div class="product border-bottom table-responsive">

                                <table class="table table-borderless">

                                <tbody>

                                    <tr \*ngFor="let i of list">

                                        <td width="20%">

                                        <img src="{{i.img}}" width="120">

                                    </td>

                                    <td width="60%">

                                        <span class="font-weight-bold">{{i.name}}</span>

                                        <div class="product-qty">

                                            <span class="d-block">Date :{{i.showdate}}</span>

                                            <span class="d-block">Time :{{i.time}}</span>

                                        </div>

                                    </td>

                                    <td width="20%">

                                        <div class="text-right">

                                            <span class="font-weight-bold">{{i.price}}</span>

                                        </div>

                                    </td>

                                    </tr>

                                </tbody>

                                </table>

                            </div>

                            <p>We will be sending tickets confirmation email to {{newuserdata.email}}</p>

                            <p class="font-weight-bold mb-0">Thanks for booking with us!</p>

                    </div>

    </div>

        </div>

    </div>

</div>

**Movieadmin.component.ts**

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

@Component({

  selector: 'app-movieadmin',

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

  styleUrls: ['./movieadmin.component.css']

})

export class MovieadminComponent implements OnInit {

    checkstatus=0

  errorcheckstatus=0

  portal=false

  constructor() { }

  ngOnInit(): void {

  }

  checkadmin(admindata:{name:string,password:string}){

    if(admindata.name=='s' && admindata.password=='1'){

      this.checkstatus=1

      this.portal=true

    }

    else{

      this.errorcheckstatus=-1

    }

  }

}

**Movieadmin.component.html**

<div class="home" \*ngIf=" checkstatus==0">

    <div class="w3-bar w3 w3-red">

      <nav class="navbar navbar-expand-sm bg-yellow navbar-dark w3-large py-sm-5">

              <div class="topnav-centered">

               <h1 class="active">Movie Admin Portal</h1>

              </div>

      </nav>

      </div>

</div>

<app-movieadminportal \*ngIf="portal"></app-movieadminportal>

<section class="vh-100"  \*ngIf=" checkstatus==0">

        <div class="container py-5 h-100">

          <div class="row d-flex justify-content-center align-items-center h-100">

            <div class="col-12 col-md-8 col-lg-6 col-xl-5">

              <div class="card shadow-2-strong" style="border-radius: 1rem;">

                <div class="card-body p-5">

                  <h3 class="mb-5 text-center">Admin Sign in</h3>

                  <h2 \*ngIf="errorcheckstatus==-1" style="color: red;">Invalid!!! Login Again..</h2>

                  <form #f="ngForm">

                  <div class="form-outline mb-4">

                    <label class="form-label" for="typeEmailX-2">Email</label>

                    <input type="text"

                    name="name"

                    id="typeEmailX-2" class="form-control form-control-lg" ngModel required/>

                  </div>

                  <div class="form-outline mb-4">

                    <label class="form-label" for="typePasswordX-2">Password</label>

                    <input type="password"

                    name="password"

                    id="typePasswordX-2" class="form-control form-control-lg" ngModel required />

                  </div>

                  <button class="btn btn-primary btn-lg btn-block"

                  type="submit"

                  [disabled]="!f.valid"

                  (click)="checkadmin(f.value)"

                  >Login</button>

                  </form>

                </div>

              </div>

            </div>

          </div>

        </div>

      </section>

**Movieadminportal.component.ts**

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

import { moviedataformat } from '../../data.model';

import { MovieserviceService } from 'src/app/Services/movieservice.service';

@Component({

  selector: 'app-movieadminportal',

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

  styleUrls: ['./movieadminportal.component.css']

})

export class MovieadminportalComponent implements OnInit {

  genres

  addform=false

  display=true

  update=false

  x=false

  post:moviedataformat

  medicinesdata:moviedataformat[]=[]

  constructor(private service:MovieserviceService) { }

  ngOnInit(): void {

    this.load()

  }

  load(){

    this.service.getmedicines()

   .subscribe((data)=>{

    this.medicinesdata=data

    console.log(this.medicinesdata)

   })

  }

  remove(id:number){

    this.service.deletemedicine(id)

    .subscribe(()=>{

      this.load()

    })

  }

  changemedstatus(id :number){

    this.service.changestatus(id)

    .subscribe(()=>{

      this.load()

    })

  }

  edit(existupdatedata:moviedataformat){

    this.update=true

    this.post=existupdatedata

  }

  updatepost(data){

    this.service.editmedicines(data)

    .subscribe(()=>{

      this.update=false

      this.load()

    })

  }

  onaddform()

  {

    this.addform=true

  }

  add(data:moviedataformat)

  {

    this.service.addmed(data)

    .subscribe(()=>{

      this.addform=false

      this.load()

    })

  }

}

**Movieadminportal.component.html**

<div \*ngIf="addform" class="center" m-4 style="width:400px">

      <br><br><br><br><br><br><button class="btn btn-warning btn-lg mx-5 " (click)="addform=false">

          Back

    </button>

    </div>

    <div \*ngIf="addform" class="container my-5 py-5">

      <div class="row d-flex justify-content-center py-5">

          <div class="col-md-7">

      <div class="card">

          <div class="card-body p-4">

    <form #f="ngForm">

      <div class="row mb-4">

        <div class="col">

          <div class="form-outline">

              <label class="form-label" for="form6Example1">Name</label>

            <input type="text" name="name" class="form-control" size="8" ngModel required/>

          </div>

        </div>

        <div class="col">

          <div class="form-outline">

              <label class="form-label" for="form6Example2">Price</label>

            <input type="text" name="price" id="form6Example2" class="form-control"ngModel required />

          </div>

        </div>

      </div>

          <div class="form-outline mb-4">

            <label class="form-label" for="form6Example6">Language</label>

          <input type="text" name="language"  id="form6Example6" class="form-control" ngModel required/>

        </div>

      <div class="form-outline mb-4">

          <label class="form-label" for="form6Example4">Genre </label>

        <input type="text" name="genre"  id="form6Example4" class="form-control" ngModel required />

      </div>

      <div class="form-outline mb-4">

        <label class="form-label" for="form6Example7">Description</label>

      <textarea class="form-control" name="description" id="form6Example7" rows="4" ngModel required></textarea>

    </div>

    <div class="form-outline mb-4">

        <label class="form-label" for="form6Example4">Image</label>

      <input type="text" name="img" id="form6Example4"  class="form-control" ngModel required />

    </div>

  <div class="form-outline mb-4">

        <label class="form-label" for="form6Example4">Date </label>

      <input type="text" name="showdate" id="form6Example4"  class="form-control" ngModel required />

    </div>

    <div class="form-outline mb-4">

        <label class="form-label" for="form6Example4">Time </label>

      <input type="text" name="time" id="form6Example4"  class="form-control" ngModel required />

    </div>

  <div class="form-outline mb-4">

        <label class="form-label" for="form6Example4">Status</label>

      <input type="text" placeholder="Enter true/false" name="status" id="form6Example4"  class="form-control" ngModel required />

    </div>

      <button type="submit" class="btn btn-primary btn-block mb-4"

      [disabled]="!f.valid"

  (click)="add(f.value)">Add </button>

    </form>

    </div>

    </div>

    </div>

    </div>

    </div>

  <div \*ngIf="update" class="center" m-4 style="width:400px">

    <br><br><button class="btn btn-warning btn-lg mx-5" (click)="update=false">

        Back

  </button>

  </div>

  <div \*ngIf="update" class="container my-5 py-5">

    <div class="row d-flex justify-content-center py-5">

        <div class="col-md-7">

    <div class="card">

        <div class="card-body p-4">

  <form #f="ngForm">

    <div class="row mb-4">

      <div class="col">

        <div class="form-outline">

        <input type="hidden" name="id" [(ngModel)]="post.id" class="form-control" size="8" ngModel required/>

      </div>

        <div class="form-outline">

            <label class="form-label" for="form6Example1">Name</label>

          <input type="text" name="name" [(ngModel)]="post.name" class="form-control" size="8" ngModel required/>

        </div>

      </div>

      <div class="col">

        <div class="form-outline">

            <label class="form-label" for="form6Example2">Price</label>

          <input type="text" name="price"[(ngModel)]="post.price" id="form6Example2" class="form-control"ngModel required />

        </div>

      </div>

    </div>

        <div class="form-outline mb-4">

          <label class="form-label" for="form6Example6">Language</label>

        <input type="text" name="language" [(ngModel)]="post.language" id="form6Example6" class="form-control" ngModel required/>

      </div>

    <div class="form-outline mb-4">

        <label class="form-label" for="form6Example4">Genre</label>

      <input type="text" name="genre" [(ngModel)]="post.genre" id="form6Example4" class="form-control" ngModel required />

    </div>

    <div class="form-outline mb-4">

        <label class="form-label" for="form6Example7">Description</label>

      <textarea class="form-control" name="description"[(ngModel)]="post.description" id="form6Example7" rows="4" ngModel required></textarea>

    </div>

    <div class="form-outline mb-4">

      <label class="form-label" for="form6Example4">Image</label>

    <input type="text" name="img" id="form6Example4" [(ngModel)]="post.img" class="form-control" ngModel required />

  </div>

<div class="form-outline mb-4">

      <label class="form-label" for="form6Example4">Date</label>

    <input type="text" name="showdate" id="form6Example4" [(ngModel)]="post.date" class="form-control" ngModel required />

  </div>

  <div class="form-outline mb-4">

    <label class="form-label" for="form6Example4">Time </label>

  <input type="text" name="time" id="form6Example4" [(ngModel)]="post.time" class="form-control" ngModel required />

</div>

<div class="form-outline mb-4">

      <label class="form-label" for="form6Example4">Status</label>

    <input type="text" name="status" id="form6Example4" [(ngModel)]="post.status" class="form-control" ngModel required />

  </div>

    <button type="submit" class="btn btn-primary btn-block mb-4"

(click)="updatepost(f.value)">Update </button>

  </form>

  </div>

  </div>

  </div>

  </div>

  </div>

**Movieadminportal.component.css**

#sectio{

    left: 50%;

}

.container{

    left:0

}

.center {

    margin:0 auto;

    display:block;

    left: 50%;

    float: none;

    top:30%;

    right: 50%;

    transform: translate(12%, -50%);

  }

  .vertical-center {

    margin: 0;

    position: absolute;

    top: 40%;

    left: 40%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

  .column{

    float: left;

    width:25%;

    padding: 0 10px;

  }

  .row{

    margin :0 -5px;

  }

  .row:after{

    content: "";

    clear: both;

  }

  @media screen and (max-width:600px) {

    .column{

      width: 100%;

      display: block;

      margin-bottom: 20px;

    }

  }

  .card{

    padding: 16px;

  }

  .a{

    text-decoration: none;

    size: 8px;

    width: 20px;

  }

**Genres.component.ts**

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

import { genres } from '../data.model';

import { MovieserviceService } from 'src/app/Services/movieservice.service';

@Component({

  selector: 'app-genres',

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

  styleUrls: ['./genres.component.css']

})

export class GenresComponent implements OnInit {

  x

  post

  display=true

  update=false

  add=false

  constructor(private service :MovieserviceService) { }

  ngOnInit(): void {

    this.getgenresfromdb()

  }

  getgenresfromdb(){

    this.service.getgenres()

    .subscribe((data)=>{

this.x=data

console.log(data)

    })

  }

  addoreditgenres(body){

      this.service.addgenres(body)

      .subscribe(()=>{

        this.getgenresfromdb()

        this.update=false

        this.add=false

      })

  }

  deletegenre(id:number){

    this.service.deletegenres(id)

    .subscribe(()=>{

      this.getgenresfromdb()

    })

  }

  onupdate(single:genres){

    this.post=single

    this.update=true

  }

}

**Genres.component.html**

<div class="center" \*ngIf="display && !add &&!update" m-4 style="width:400px">

    <br><br><br><br><br><br><br>

    <button class="btn btn-warning btn-lg mx-5" (click)="add=true">

     Add item

  </button>

</div>

<table \*ngIf="display && !add &&!update" style="width: 80rem;

    height: 8rem;">

      <tr>

      <th>

        <h1 style="color: blue;">Name</h1>

      </th>

      <th>

        <h1 style="color: blue;">Update</h1>

      </th>

      <th>

      <h1 style="color: blue;">Delete</h1>

      </th>

      </tr>

      <tr \*ngFor="let single of x,let i=index">

        <td \*ngIf="single">

          <h1 >{{single.name}}</h1>

        </td>

        <td \*ngIf="single">

          <button class="btn btn-outline-success m-2" (click)="onupdate(single)">

            Update

         </button>

        </td>

        <td \*ngIf="single">

          <button class="btn btn-outline-danger"(click)="deletegenre(single.id)">

            Delete

         </button>

        </td>

      </tr>

    </table>

    <div \*ngIf="add" class="center" m-4 style="width:400px">

        <br><br><br><br><br><br><button class="btn btn-warning btn-lg mx-5 " (click)="add=false">

            Back

      </button>

      </div>

      <div \*ngIf="add" class="container my-5 py-5">

        <div class="row d-flex justify-content-center py-5">

            <div class="col-md-7">

        <div class="card">

            <div class="card-body p-4">

      <form #f="ngForm">

        <div class="row mb-4">

          <div class="col">

            <div class="form-outline">

                <label class="form-label" for="form6Example1">Name</label>

              <input type="text" name="name" class="form-control" size="8" ngModel required/>

            </div>

          </div>

        </div>

        <button type="submit" class="btn btn-primary btn-block mb-4"

        [disabled]="!f.valid"

    (click)="addoreditgenres(f.value)">Add </button>

      </form>

      </div>

      </div>

      </div>

      </div>

      </div>

    <div \*ngIf="update" class="center" m-4 style="width:400px">

      <br><br><button class="btn btn-warning btn-lg mx-5" (click)="update=false">

          Back

    </button>

    </div>

    <div \*ngIf="update" class="container my-5 py-5">

      <div class="row d-flex justify-content-center py-5">

          <div class="col-md-7">

      <div class="card">

          <div class="card-body p-4">

    <form #f="ngForm">

      <div class="row mb-4">

        <div class="col">

          <div class="form-outline">

          <input type="hidden" name="id" [(ngModel)]="post.id" class="form-control" size="8" ngModel required/>

        </div>

          <div class="form-outline">

              <label class="form-label" for="form6Example1">Name</label>

            <input type="text" name="name" [(ngModel)]="post.name" class="form-control" size="8" ngModel required/>

          </div>

     </div>

     </div>

      <button type="submit" class="btn btn-primary btn-block mb-4"

  (click)="addoreditgenres(f.value)">Update </button>

    </form>

    </div>

    </div>

    </div>

    </div>

    </div>

**Genres.component.css**

#sectio{

    left: 50%;

}

.container{

    left:0

}

.center {

    margin:0 auto;

    display:block;

    left: 50%;

    float: none;

    top:30%;

    right: 50%;

    transform: translate(12%, -50%);

  }

  .vertical-center {

    margin: 0;

    position: absolute;

    top: 40%;

    left: 40%;

    -ms-transform: translateY(-50%);

    transform: translateY(-50%);

  }

  .column{

    float: left;

    width:25%;

    padding: 0 10px;

  }

  .row{

    margin :0 -5px;

  }

  .row:after{

    content: "";

    clear: both;

  }

  @media screen and (max-width:600px) {

    .column{

      width: 100%;

      display: block;

      margin-bottom: 20px;

    }

  }

  .card{

    padding: 16px;

  }

  .a{

    text-decoration: none;

    size: 8px;

    width: 20px;

  }

**Backend**

**Controller**

package com.example.movieproject.controller;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.CrossOrigin;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import com.example.movieproject.entity.\*;

import com.example.movieproject.repository.\*;

@RestController

@CrossOrigin

@RequestMapping("")

public class HomeController {

@Autowired

GenresRepository gr;

@Autowired

MovieRepository mr;

@Autowired

UsersRepository ur;

@GetMapping("/movies")

public Iterable<movie> movies() {

Iterable<movie> movies = mr.findAll();

return movies;

}

@GetMapping("/movieusers")

public Iterable<users> user() {

Iterable<users> us = ur.findAll();

return us;

}

@PostMapping("/movieregisteruser")

public void userpost(@RequestBody users u) {

ur.save(u);

}

@GetMapping("/deletemovie/{id}")

public void deletemed(@PathVariable("id") int id) {

mr.deleteById(id);

}

@PostMapping("/editmovies")

public void editmed(@RequestBody movie m) {

mr.save(m);

}

@PostMapping("/addmovie")

public void addmed(@RequestBody movie m) {

mr.save(m);

}

@GetMapping("/changemoviestatus/{id}")

public void changestatus(@PathVariable("id") int id) {

movie m=mr.findById(id).get();

if(m.getStatus().equals("true"))

{

m.setStatus("false");

}

else {

m.setStatus("true");

}

mr.save(m);

}

@GetMapping("/genres")

public Iterable<genres> genres() {

Iterable<genres> ge = gr.findAll();

return ge;

}

@PostMapping("/addoreditgenres")

public void addgenres(@RequestBody genres m) {

gr.save(m);

}

@GetMapping("/deletegenres/{id}")

public void deletegenre(@PathVariable("id") int id) {

gr.deleteById(id);

}

}

Entities

Users.java

**package** com.example.movieproject.entity;

**import** jakarta.persistence.\*;

@Entity

**public** **class** genres {

@Id

@GeneratedValue(strategy = GenerationType.***IDENTITY***)

**int** id;

String name;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

@Override

**public** String toString() {

**return** "genres [id=" + id + ", name=" + name + "]";

}

}

Movie.java

package com.example.movieproject.entity;

import java.sql.Date;

import jakarta.persistence.\*;

@Entity

public class movie {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

int id;

String name;

int price;

String language;

String genre;

String description;

String showdate;

String time;

String status;

String img;

public String getImg() {

return img;

}

public void setImg(String img) {

this.img = img;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getPrice() {

return price;

}

public void setPrice(int price) {

this.price = price;

}

public String getLanguage() {

return language;

}

public void setLanguage(String language) {

this.language = language;

}

public String getGenre() {

return genre;

}

public void setGenre(String genre) {

this.genre = genre;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public String getShowdate() {

return showdate;

}

public void setShowdate(String showdate) {

this.showdate = showdate;

}

public String getTime() {

return time;

}

public void setTime(String time) {

this.time = time;

}

@Override

public String toString() {

return "movie [id=" + id + ", name=" + name + ", price=" + price + ", language=" + language + ", genre=" + genre

+ ", description=" + description + ", showdate=" + showdate + ", time=" + time + ", status=" + status

+ ", img=" + img + "]";

}

}

Genres.java

**package** com.example.movieproject.entity;

**import** jakarta.persistence.\*;

@Entity

**public** **class** users {

@Id

@GeneratedValue(strategy = GenerationType.***IDENTITY***)

**int** id;

String name;

String email;

**int** age;

String password;

String phone;

@Override

**public** String toString() {

**return** "users [id=" + id + ", name=" + name + ", email=" + email + ", age=" + age + ", password=" + password

+ ", phone=" + phone + "]";

}

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** String getPhone() {

**return** phone;

}

**public** **void** setPhone(String phone) {

**this**.phone = phone;

}

}

Repositories

Genres.java

package com.example.movieproject.repository;

import org.springframework.data.repository.CrudRepository;

import com.example.movieproject.entity.genres;

public interface GenresRepository extends CrudRepository<genres, Integer> {

}

Movie.java

package com.example.movieproject.repository;

import org.springframework.data.repository.CrudRepository;

import com.example.movieproject.entity.movie;

public interface MovieRepository extends CrudRepository<movie, Integer> {

}

Users.java

package com.example.movieproject.repository;

import org.springframework.data.repository.CrudRepository;

import com.example.movieproject.entity.users;

public interface UsersRepository extends CrudRepository<users, Integer> {

}