Code:

Login:

**package** com.simplilearn.bean;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

@Entity

**public** **class** Login {

@Id

**private** String emailid;

**private** String password;

@Column(name = "typeofuser")

**private** String typeOfUser;

**public** String getEmailid() {

**return** emailid;

}

**public** **void** setEmailid(String emailid) {

**this**.emailid = emailid;

}

**public** String getPassword() {

**return** password;

}

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

**this**.password = password;

}

**public** String getTypeOfUser() {

**return** typeOfUser;

}

**public** **void** setTypeOfUser(String typeOfUser) {

**this**.typeOfUser = typeOfUser;

}

@Override

**public** String toString() {

**return** "Login [emailid=" + emailid + ", password=" + password + ", typeOfUser=" + typeOfUser + "]";

}

}

User:

**package** com.simplilearn.bean;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

@Entity

**public** **class** User {

@Id

@Column(name="id")

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

**private** **int** id;

**private** String dob;

**private** String name;

**private** String address;

**private** String email;

**private** String mobile;

**private** String gender;

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

**return** id;

}

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

**this**.id = id;

}

**public** String getDob() {

**return** dob;

}

**public** **void** setDob(String dob) {

**this**.dob = dob;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** String getAddress() {

**return** address;

}

**public** **void** setAddress(String address) {

**this**.address = address;

}

**public** String getEmail() {

**return** email;

}

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

**this**.email = email;

}

**public** String getMobile() {

**return** mobile;

}

**public** **void** setMobile(String mobile) {

**this**.mobile = mobile;

}

**public** String getGender() {

**return** gender;

}

**public** **void** setGender(String gender) {

**this**.gender = gender;

}

}

LoginController:

**package** com.simplilearn.controller;

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

**import** org.springframework.http.MediaType;

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

**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.simplilearn.bean.Login;

**import** com.simplilearn.service.LoginService;

@RestController

@RequestMapping("login")

@CrossOrigin

**public** **class** LoginController {

@Autowired

LoginService loginService;

@PostMapping(value = "signIn",consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String signIn(@RequestBody Login login) {

System.***out***.println("I cam here");

**return** loginService.signIn(login);

}

@PostMapping(value = "signUp",consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String signUp(@RequestBody Login login) {

System.***out***.println(login);

**return** loginService.signUp(login);

}

}

UserController:

**package** com.simplilearn.controller;

**import** java.util.List;

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

**import** org.springframework.http.HttpStatus;

**import** org.springframework.http.ResponseEntity;

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

**import** org.springframework.web.bind.annotation.DeleteMapping;

**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.PutMapping;

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

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

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

**import** com.simplilearn.bean.User;

**import** com.simplilearn.service.UserService;

@CrossOrigin

@RestController

@RequestMapping("/api/user")

**public** **class** UserController {

@Autowired

**private** UserService service;

@PostMapping("/")

**public** ResponseEntity<User> adduser(@RequestBody User u){

User user= service.addUser(u);

**if**(user!=**null**)

**return** **new** ResponseEntity<User>(user,HttpStatus.***CREATED***);

**else**

**return** **new** ResponseEntity<User>(user,HttpStatus.***INTERNAL\_SERVER\_ERROR***);

}

@GetMapping("/")

**public** List<User> getAllUser(){

**return** service.getAllUser();

}

@GetMapping("/{id}")

**public** ResponseEntity<User>getUserById (@PathVariable **int** id){

User user= service.getUserById(id);

**if**(user!=**null**)

**return** **new** ResponseEntity<User>(user, HttpStatus.***FOUND***);

**else**

**return** **new** ResponseEntity<User>(user, HttpStatus.***NOT\_FOUND***);

}

@PutMapping("/{id}")

**public** ResponseEntity<Object> updateUser(@PathVariable **int** id,@RequestBody User newUser){

User user= service.updateUser(id, newUser);

**if** (user!=**null**)

**return** **new** ResponseEntity<Object>(user,HttpStatus.***OK***);

**else**

**return** **new** ResponseEntity<Object>("No User Available to Update",HttpStatus.***NOT\_FOUND***);

}

@DeleteMapping("/{id}")

**public** ResponseEntity<String>deleteUser(@PathVariable **int** id){

**boolean** result = service.deleteUser(id);

**if**(result)

**return** **new** ResponseEntity<String>("Object Deleted",HttpStatus.***OK***);

**else**

**return** **new** ResponseEntity<String>("NO user Found", HttpStatus.***NOT\_FOUND***);

}

}

LoginRepository:

**package** com.simplilearn.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.simplilearn.bean.Login;

@Repository

**public** **interface** LoginRepository **extends** JpaRepository<Login, String>{

}

UserRepo:

**package** com.simplilearn.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.simplilearn.bean.User;

@Repository

**public** **interface** UserRepo **extends** JpaRepository<User,Integer> {

}

LoginService:

**package** com.simplilearn.service;

**import** java.util.Optional;

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

**import** org.springframework.stereotype.Service;

**import** com.simplilearn.bean.Login;

**import** com.simplilearn.repository.LoginRepository;

@Service

**public** **class** LoginService {

@Autowired

**public** LoginRepository loginRepository;

**public** String signIn(Login login) {

Optional<Login> result = loginRepository.findById(login.getEmailid());

**if**(result.isPresent()) {

Login ll = result.get();

**if**(ll.getPassword().equals(login.getPassword())) {

**if**(login.getTypeOfUser().equals(ll.getTypeOfUser()) && login.getTypeOfUser().equals("admin")) {

**return** "Admin sucessfully login";

}**else** **if**(login.getTypeOfUser().equals(ll.getTypeOfUser()) && login.getTypeOfUser().equals("user")){

**return** "User successfully login";

}**else** {

**return** "Invalid details";

}

}**else** {

**return** "InValid password";

}

}**else** {

**return** "InValid emailId";

}

}

**public** String signUp(Login login) {

Optional<Login> result = loginRepository.findById(login.getEmailid());

**if**(result.isPresent()) {

**return** "Email Id alreay exists";

}**else** {

**if**(login.getTypeOfUser().equals("admin")) {

**return** "You can't create admin account";

}**else** {

loginRepository.save(login);

**return** "Account created successfully";

}

}

}

}

UserService:

**package** com.simplilearn.service;

**import** java.util.List;

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

**import** org.springframework.stereotype.Service;

**import** com.simplilearn.bean.User;

**import** com.simplilearn.repository.UserRepo;

@Service

**public** **class** UserService {

@Autowired

**private** UserRepo repo;

**public** User addUser(User u) {

**return** repo.save(u);

}

**public** List<User> getAllUser(){

**return** repo.findAll();

}

**public** User getUserById(**int** id) {

**if**(repo.findById(id).isPresent()) {

**return** repo.findById(id).get();

}

**else** {

**return** **null**;

}

}

**public** User updateUser(**int** id, User newUser) {

**if**(repo.findById(id).isPresent()) {

User olduser= repo.findById(id).get();

olduser.setName(newUser.getName());

olduser.setDob(newUser.getDob());

olduser.setAddress(newUser.getAddress());

olduser.setEmail(newUser.getEmail());

olduser.setMobile(newUser.getMobile());

olduser.setGender(newUser.getGender());

**return** repo.save(olduser);

}

**else** {

**return** **null**;

}

}

**public** **boolean** deleteUser(**int** id) {

**if**(repo.findById(id).isPresent()) {

repo.deleteById(id);

**return** **true**;

}

**else** {

**return** **false**;

}

}

}

MyAppApplicationTestscode:

**package** com;

**import** org.testng.annotations.AfterMethod;

**import** org.testng.annotations.BeforeMethod;

**import** org.testng.annotations.Test;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver;

**public** **class** MyAppApplicationTestscode {

WebDriver driver;

@Test(priority = 0)

**public** **void** testing() {

**long** start = System.*currentTimeMillis*();

driver.manage().window().maximize();

driver.get("http://localhost:4200/home");

**long** finish = System.*currentTimeMillis*();

**long** totalTime = finish - start;

System.***out***.println("Total Time for main page load = " + (totalTime \* 0.001) + " Seconds");

}

@Test(priority = 1)

**public** **void** signup() {

driver.get("http://localhost:4200/signUp");

driver.findElement(By.*xpath*("//\*[@id=\"userName\"]")).sendKeys("u1");

driver.findElement(By.*xpath*("//\*[@id=\"userPassword\"]")).sendKeys("u1");

driver.findElement(By.*xpath*("/html/body/app-root/app-signup/div/div/form/input[3]")).click();

System.***out***.println("sucessfully signUp ");

}

@Test(priority = 2)

**public** **void** signin() {

driver.get("http://localhost:4200/login");

driver.findElement(By.*xpath*("//\*[@id=\"userName\"]")).sendKeys("admin");

driver.findElement(By.*xpath*("//\*[@id=\"userPassword\"]")).sendKeys("admin");

driver.findElement(By.*xpath*("/html/body/app-root/app-login/div/div/form/input[3]")).click();

System.***out***.println("sucessfully logged in");

}

@Test(priority = 4)

**public** **void** adminhome() {

**long** start = System.*currentTimeMillis*();

driver.manage().window().maximize();

driver.get("http://localhost:4200/adminHome");

**long** finish = System.*currentTimeMillis*();

**long** totalTime = finish - start;

System.***out***.println("Total Time for admin home page load = " + (totalTime \* 0.001) + " Seconds");

}

@Test(priority = 5)

**public** **void** userhome() {

**long** start = System.*currentTimeMillis*();

driver.manage().window().maximize();

driver.get("http://localhost:4200/userHome");

**long** finish = System.*currentTimeMillis*();

**long** totalTime = finish - start;

System.***out***.println("Total Time for user home page load = " + (totalTime \* 0.001) + " Seconds");

}

@Test(priority = 6)

**public** **void** loadusers() {

**long** start = System.*currentTimeMillis*();

driver.manage().window().maximize();

driver.get("http://localhost:4200/allusers");

**long** finish = System.*currentTimeMillis*();

**long** totalTime = finish - start;

System.***out***.println("Total Time for allusers page load = " + (totalTime \* 0.001) + " Seconds");

}

@Test(priority = 7)

**public** **void** approve() {

**long** start = System.*currentTimeMillis*();

driver.manage().window().maximize();

driver.get("http://localhost:4200/approve");

**long** finish = System.*currentTimeMillis*();

**long** totalTime = finish - start;

System.***out***.println("Total Time for approve page load = " + (totalTime \* 0.001) + " Seconds");

}

@BeforeMethod

**public** **void** beforeMethod() {

driver = **new** ChromeDriver();

}

@AfterMethod

**public** **void** afterMethod() {

// driver.close();

driver = **null**;

}

}

admindashboard.component.html:

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

    <h1 class="text-center" style="font-family: inherit;">Welcome to myAadhaar Aadhar</h1><br><br>

    <div class="col text-center">

      <div class="col text-center">

        <div class="card text-white bg-success mx-auto" style="max-width: 18rem;">

            <div class="card-body">

              <h5 class="card-title"><a routerLink="/allusers" style="color:white ;">List of All the Users</a></h5>

              <p class="card-text">View all the Users</p>

            </div>

      </div>

      </div><br>

      <div class="col text-center">

        <div class="card text-white bg-info mx-auto" style="max-width: 18rem;">

            <div class="card-body">

              <h5 class="card-title"><a routerLink="/approve" style="color:white ;">Approve New Users</a></h5>

              <p class="card-text">Here we can approve and review modifications</p>

            </div>

      </div>

      </div>

    </div>

  </div>

allusers.component.html:

<div class="container mt-5">

    <table class="table">

        <thead class="thead-light">

          <tr>

            <th scope="col">Id</th>

            <th scope="col">Name</th>

            <th scope="col">DOB</th>

            <th scope="col">Address</th>

            <th scope="col">Email id</th>

            <th scope="col">Mobile Number </th>

            <th scope="col">Gender</th>

          </tr>

        </thead>

        <tbody>

            <tr \*ngFor="let u of users">

                <td>{{u.id}}</td>

                <td>{{u.name}}</td>

                <td>{{u.dob}}</td>

                <td>{{u.address}}</td>

                <td>{{u.email}}</td>

                <td>{{u.mobile}}</td>

                <td>{{u.gender}}</td>

            </tr>

        </tbody>

      </table>

</div>

editusers.component:

  <!-- Modal -->

  <div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">

    <div class="modal-dialog" role="document">

      <div class="modal-content">

        <div class="modal-header">

          <h5 class="modal-title" id="exampleModalLabel" style="color: green;">Approved successfully</h5>

          <button type="button" class="close" data-dismiss="modal" aria-label="Close">

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

          </button>

        </div>

        <div class="modal-body">

           <h2>Your Request Processed Sucessfully</h2>

        </div>

        <div class="modal-footer">

          <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>

        </div>

      </div>

    </div>

  </div>

  <div class="container mt-5">

    <table class="table">

        <thead class="thead-light">

          <tr>

            <th scope="col">Id</th>

            <th scope="col">Name</th>

            <th scope="col">DOB</th>

            <th scope="col">Address</th>

            <th scope="col">Email id</th>

            <th scope="col">Mobile Number </th>

            <th scope="col">Gender</th>

            <th scope="col">Action</th>

          </tr>

        </thead>

        <tbody>

            <tr \*ngFor="let u of users">

                <td>{{u.id}}</td>

                <td>{{u.name}}</td>

                <td>{{u.dob}}</td>

                <td>{{u.address}}</td>

                <td>{{u.email}}</td>

                <td>{{u.mobile}}</td>

                <td>{{u.gender}}</td>

                <td>

                <div class="row">

                  <div class="col"><button  type="button" class="btn btn-success" data-toggle="modal" data-target="#exampleModal">Approve</button></div>

                  <div class="col"><button type="button" (click)='BookDelete(u.id)' class="btn btn-danger">Delete</button></div>

             </div>

            </td>

            </tr>

        </tbody>

      </table>

</div>

footer.component.html:

  <style media="screen">

    footer{

        position:absolute;

        bottom: 0;

        left: 0;

        right: 0;

        background: linear-gradient(119.54deg,#000046 0%,#534f89 100%);

        height: auto;

        width: 100vw;

        color: #fff;

        padding: top 5px;;

    }

    .footer-content{

        display: flex;

        align-items: center;

        justify-content: center;

        flex-direction: column;

        text-align: center;

    }

    .footer-content p{

        max-width: 500px;

        line-height: 20px;

        font-size: 14px;

        color: #cacdd2;

    }

    @media (max-width:500px) {

    .footer-menu ul{

      display: flex;

      margin-top: 10px;

      margin-bottom: 20px;

    }

    }

        </style>

        <footer>

            <div class="footer-content">

                <p> copyright &copy; <a href="#"> Developers</a>  2022 Unique Identification Authority of India All Rights Reserved</p>

            </div>

        </footer>

header.component.html:

<nav class="navbar navbar-expand-lg navbar-dark" style="background: linear-gradient(119.54deg,#000046 0%,#1CB5E0 100%);">

    <div class="container-fluid">

      <a class="navbar-brand" href="#">M-Aadhaar</a>

      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

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

      </button>

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

        <ul class="navbar-nav me-auto mb-2 mb-lg-0">

          <li class="nav-item">

            <a class="nav-link active" aria-current="page" routerLink="/home">Home</a>

          </li>

          <li class="nav-item">

            <a class="nav-link active" aria-current="page" routerLink="/adminHome"

            \*ngIf="isAdmin()"

            >Admin Dashboard</a>

          </li>

          <li class="nav-item">

            <a class="nav-link active" aria-current="page" routerLink="/userHome"

            \*ngIf="isUser()"

            >User Dashboard</a>

          </li>

        </ul>

       </div>

       <button \*ngIf="!isLoggedIn()" type="button" class="btn btn-success" routerLink="/login">Login</button>

       <button \*ngIf="isLoggedIn()" type="button" class="btn btn-danger" (click)="logout()" >Logout</button>

    </div>

  </nav>

home.component:

<div class="container"><br>

    <div class="text-center"><img src="assets/images/aadhar-card.jpg" class="img-fluid" alt="..."></div><br>

    <h3 style="text-align: center;">Aadhaar</h3>

    <div class="aad-is-dv1" style="text-align: center;">

       <span class="li-text">A 12-digit unique identity for every Indian individual, including children and infants</span>

       <span class="li-text">Enables identification for every resident Indian</span>

       <span class="li-text">Establishes uniqueness of every individual on the basis of demographic and biometric information</span>

       <span class="li-text">It is a voluntary service that every resident can avail irrespective of present documentation </span>

       <span class="li-text">Each individual will be given a single unique Aadhaar ID number</span>

    </div>

 </div>

login.component:

<div class="container mt-5">

    <div class="card shadow p-3" style="padding: 20px;width: 30%;margin: auto;">

        <form [formGroup]="loginRef" (ngSubmit)="signIn()">

            <input type="text"

            ngModel

            name="userName"

            id="userName"

            placeholder="Enter your username"

            class="form-control mb-3"

            formControlName="emailid"

            >

            <input type="password"

            ngModel

            name="userPassword"

            id="userPassword"

            placeholder="Enter your password"

            class="form-control mb-3"

            formControlName="password"

            >

            <div class="row text-center">

                <div class="col"><input type="radio" name="typeOfUser" value="admin" formControlName="typeOfUser"/><h5 style="font-family: sans-serif;">admin</h5>

                </div>

                <div class="col"><input type="radio" name="typeOfUser" value="user" formControlName="typeOfUser"/><h5 style="font-family: sans-serif;">user</h5>

                </div>

            </div><br>

            <input type="submit"

             value="signin"

            class="btn btn-success form-control rounded-0">

            <br><br>

            <input type="reset" value="reset"

            class="btn btn-primary form-control rounded-0">

        </form>

        <br>

        <span style="color:red">{{msg}}</span><br/>

        <a class="text-center" routerLink="/signUp"><h4 style="font-family: sans-serif;">SignUp</h4></a>

    </div>

</div>

signup.component:

<div class="container mt-5">

    <h2 class="text-center">Create your Account </h2>

    <div class="card shadow p-3" style="padding: 20px;width: 30%;margin: auto;">

        <form [formGroup]="loginRef" (ngSubmit)="signUp()">

            <input type="text"

            ngModel

            name="userName"

            id="userName"

            placeholder="Enter your username"

            class="form-control mb-3"

            formControlName="emailid"

            >

            <input type="password"

            ngModel

            name="userPassword"

            id="userPassword"

            placeholder="Enter your password"

            class="form-control mb-3"

            formControlName="password"

            >

            <div class="row text-center">

                <div class="col"><input type="radio" name="typeOfUser" value="admin" formControlName="typeOfUser"/><h5 style="font-family: sans-serif;">admin</h5>

                </div>

                <div class="col"><input type="radio" name="typeOfUser" value="user" formControlName="typeOfUser"/><h5 style="font-family: sans-serif;">user</h5>

                </div>

            </div>

            <input type="submit"

            value="signup"

            class="btn btn-success form-control rounded-0">

            <br><br>

            <input type="reset" value="reset"

            class="btn btn-primary form-control rounded-0">

        </form>

        <br>

        <span style="color:red" class="text-center">{{msg}}</span>

        <a class="text-center" routerLink="/login"><h4>Login</h4></a>

    </div>

</div>

userdashboard.component:

<style>

  .custom {

    width: 200px !important;

  }

</style>

<div class="fluid-container text-center p-5">

  <h1>Welcome to M-Aadhaar page</h1>

  <br><br><br>

  <div class="row">

    <div class="col">

      <img src="assets/images/images.jpg" style="width: 60%;" alt="...">

    </div>

    <div class="col mt-5">

      <div class="row">

        <div class="col">

          <div class="d-grid  col-4 ">

            <button type="button" class="btn btn-success btn-lg rounded custom" data-toggle="modal"

              data-target="#exampleModal1">New Aadhaar</button><br><br>

            <button type="button" class="btn btn-primary  btn-lg rounded custom" data-toggle="modal"

              data-target="#exampleModal2">Update Aadhaar</button><br><br>

          </div>

        </div>

        <div class="row" style="margin-right: 30%;">

          <div class="col">

            <div class="d-grid col-4">

              <button type="button" class="btn btn-secondary btn-lg rounded custom" data-toggle="modal"

                data-target="#exampleModal3">Duplicate Aadhaar</button><br><br>

              <button type="button" class="btn btn-danger btn-lg rounded custom" data-toggle="modal"

                data-target="#exampleModal4">Delete Aadhaar</button><br><br>

            </div>

          </div>

        </div>

      </div>

    </div>

  </div>

</div>

<!-- Modal for Update -->

<div class="modal fade" id="exampleModal2" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"

  aria-hidden="true">

  <div class="modal-dialog" role="document">

    <div class="modal-content">

      <div class="modal-header">

        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>

        <button type="button" class="close" data-dismiss="modal" aria-label="Close">

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

        </button>

      </div>

      <div style="width: 500px; margin: auto;" class="modal-body">

        <div class="submit-form">

          <div \*ngIf="!isBookAdded">

            <div class="form-group">

              <label for="name">Name</label>

              <input class="form-control" id="name" required [(ngModel)]="user.name" name="email" />

            </div>

            <div class="form-group">

              <label for="dob">Date of birth</label>

              <input class="form-control" id="dob" required [(ngModel)]="user.dob" name="dob" />

            </div>

            <div class="form-group">

              <label for="address">Address</label>

              <input class="form-control" id="address" required [(ngModel)]="user.address" name="address" />

            </div>

            <div class="form-group">

              <label for="email">Email</label>

              <input class="form-control" id="email" required [(ngModel)]="user.email" name="email" />

            </div>

            <div class="form-group">

              <label for="mobile">Mobile</label>

              <input class="form-control" id="mobile" required [(ngModel)]="user.mobile" name="mobile" />

            </div>

            <div class="form-group">

              <label for="gender">Gender</label>

              <input class="form-control" id="gender" required [(ngModel)]="user.gender" name="gender" />

            </div>

            <button (click)="addUser()" class="btn btn-success">Update</button>

          </div>

          <div \*ngIf="isBookAdded">

            <h3>Data Updated SuccessFully</h3>

          </div>

        </div>

      </div>

    </div>

  </div>

</div>

<!-- Modal for creation -->

<div class="modal fade" id="exampleModal1" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"

  aria-hidden="true">

  <div class="modal-dialog" role="document">

    <div class="modal-content">

      <div class="modal-header">

        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>

        <button type="button" class="close" data-dismiss="modal" aria-label="Close">

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

        </button>

      </div>

      <div style="width: 500px; margin: auto;" class="modal-body">

        <div class="submit-form">

          <div \*ngIf="!isBookAdded">

            <div class="form-group">

              <label for="name">Name</label>

              <input class="form-control" id="name" required [(ngModel)]="user.name" name="email" />

            </div>

            <div class="form-group">

              <label for="dob">Date of birth</label>

              <input class="form-control" id="dob" required [(ngModel)]="user.dob" name="dob" />

            </div>

            <div class="form-group">

              <label for="address">Address</label>

              <input class="form-control" id="address" required [(ngModel)]="user.address" name="address" />

            </div>

            <div class="form-group">

              <label for="email">Email</label>

              <input class="form-control" id="email" required [(ngModel)]="user.email" name="email" />

            </div>

            <div class="form-group">

              <label for="mobile">Mobile</label>

              <input class="form-control" id="mobile" required [(ngModel)]="user.mobile" name="mobile" />

            </div>

            <div class="form-group">

              <label for="gender">Gender</label>

              <input class="form-control" id="gender" required [(ngModel)]="user.gender" name="gender" />

            </div>

            <button (click)="addUser()" class="btn btn-success">Create</button>

          </div>

          <div \*ngIf="isBookAdded">

            <h3>Data added SuccessFully</h3>

          </div>

        </div>

      </div>

    </div>

  </div>

</div>

<!-- Modal for duplicate -->

<div class="modal fade" id="exampleModal3" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"

  aria-hidden="true">

  <div class="modal-dialog" role="document">

    <div class="modal-content">

      <div class="modal-header">

        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>

        <button type="button" class="close" data-dismiss="modal" aria-label="Close">

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

        </button>

      </div>

      <div class="modal-body">

        <h4 style="color:green ;">Your Aadhhar card has been successfully duplicated. It will be delivered to a

          registered address in 30 working days.

        </h4><br>

        <h4 style="color:rgb(126, 126, 246);">Thank you for using m-Aadhaar services!</h4>

      </div>

      <div class="modal-footer">

        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>

      </div>

    </div>

  </div>

</div>

<!-- Modal for delete -->

<div class="modal fade" id="exampleModal4" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"

  aria-hidden="true">

  <div class="modal-dialog" role="document">

    <div class="modal-content">

      <div class="modal-header">

        <h5 class="modal-title" id="exampleModalLabel">Delete Aadhaar Card</h5>

        <button type="button" class="close" data-dismiss="modal" aria-label="Close">

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

        </button>

      </div>

      <div class="modal-body">

        <h4 style="color:green ;">Your request was placed successfully</h4><br>

        <h4 style="color: red;">The Aadhaar card gets deleted once the admin gets noticed</h4>

      </div>

      <div class="modal-footer">

        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>

      </div>

    </div>

  </div>

</div>