Registration.html

<div class="container">

<!-- Form -->

<div class="row">

<div class="col-6">

<section class="h-100 bg-dark">

<div class="container h-100">

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

<div class="col">

<div class="card card-registration">

<div class="row g-0">

<div class="">

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

<h3 class="mb-5 text-uppercase">Citizen Registration form</h3>

<div class="row">

<div class="col-md-6 mb-4">

<div class="form-outline">

<input type="text" id="form3Example1m" class="form-control form-control-lg" />

<label class="form-label" for="form3Example1m">First name</label>

</div>

</div>

<div class="col-md-6 mb-4">

<div class="form-outline">

<input type="text" id="form3Example1n" class="form-control form-control-lg" />

<label class="form-label" for="form3Example1n">Last name</label>

</div>

</div>

</div>

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

<input type="text" id="form3Example8" class="form-control form-control-lg" />

<label class="form-label" for="form3Example8">Address</label>

</div>

<div class="d-md-flex justify-content-start align-items-center mb-4 py-2">

<h6 class="mb-0 me-4">Gender: </h6>

<div class="form-check form-check-inline mb-0 me-4">

<input class="form-check-input" type="radio" name="inlineRadioOptions" id="femaleGender"

value="option1" />

<label class="form-check-label" for="femaleGender">Female</label>

</div>

<div class="form-check form-check-inline mb-0 me-4">

<input class="form-check-input" type="radio" name="inlineRadioOptions" id="maleGender"

value="option2" />

<label class="form-check-label" for="maleGender">Male</label>

</div>

<div class="form-check form-check-inline mb-0">

<input class="form-check-input" type="radio" name="inlineRadioOptions" id="otherGender"

value="option3" />

<label class="form-check-label" for="otherGender">Other</label>

</div>

</div>

<div class="row">

<div class="col-md-6 mb-4">

<select class="select">

<option value="1">State</option>

<option value="2">Option 1</option>

<option value="3">Option 2</option>

<option value="4">Option 3</option>

</select>

</div>

<div class="col-md-6 mb-4">

<select class="select">

<option value="1">City</option>

<option value="2">Option 1</option>

<option value="3">Option 2</option>

<option value="4">Option 3</option>

</select>

</div>

</div>

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

<input type="date" id="form3Example9" class="form-control form-control-lg" />

<label class="form-label" for="form3Example9">DOB</label>

</div>

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

<input type="number" id="form3Example90" class="form-control form-control-lg" />

<label class="form-label" for="form3Example90">Contact No.</label>

</div>

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

<input type="text" id="form3Example97" class="form-control form-control-lg" />

<label class="form-label" for="form3Example97">Email ID</label>

</div>

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

<input type="password" id="form3Example97" class="form-control form-control-lg" />

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

</div>

<div class="d-flex justify-content-end pt-3">

<button type="button" class="btn btn-light btn-lg">Reset all</button>

<button type="button" class="btn btn-primary btn-lg ms-2">Register</button>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

</div>

<div class="col-6">

<img src="https://www.thenewsminute.com/sites/default/files/styles/w702xh400/public/Aadhaar\_Women\_Rep\_PTI\_COVID\_Vaccine\_1200\_Nadia\_160821.jpg?itok=MPqu2xbk" alt="">

</div>

<!-- Form end -->

</div>

</div>

Signin.html

<div class="container">

<!-- Form -->

<div class="row">

<div class="col-6">

<form class="form-example card mt-5 pl-5 pt-5 pb-5" [formGroup]="signupForm" (ngSubmit)="onSubmit()">

<h1>Login</h1>

<div class="form-group">

<label for="phone\_num" class="text-light">MobileNumber:</label>

<input

type="text"

class="form-control phone\_num"

id="phone\_num"

placeholder="Mobile Number"

name="phone\_num"

style="width: 20vw"

/>

</div>

<div class="form-group">

<label for="password" class="text-light">Password:</label>

<input

type="password"

class="form-control password"

id="password"

placeholder="Password"

name="password"

style="width: 20vw"

/>

</div>

<div>

<button type="submit" class="btn btn-primary btn-customized">

Login

</button>

</div>

<!-- End input fields -->

</form>

</div>

<div class="col-6">

<img src="https://www.thenewsminute.com/sites/default/files/styles/w702xh400/public/Aadhaar\_Women\_Rep\_PTI\_COVID\_Vaccine\_1200\_Nadia\_160821.jpg?itok=MPqu2xbk" alt="">

</div>

<!-- Form end -->

</div>

</div>

Adminlogin.html

<div class="container">

<!-- Form -->

<div class="row">

<div class="col-6">

<form class="form-example card mt-5 pl-5 pt-5 pb-5" (ngSubmit)="onSubmit()">

<h1>Admin Login</h1>

<div class="form-group">

<label for="phone\_num" class="text-light">Username:</label>

<input

type="text"

class="form-control uName"

id="uName"

placeholder="Username"

name="uName"

style="width: 20vw"

/>

</div>

<div class="form-group">

<label for="password" class="text-light">Password:</label>

<input

type="password"

[(ngModel)]="admin.pass"

class="form-control password"

id="pass"

placeholder="Password"

name="pass"

style="width: 20vw"

/>

</div>

<div>

<button type="submit" class="btn btn-primary btn-customized">

Login

</button>

</div>

<!-- End input fields -->

</form>

</div>

<div class="col-6">

<img width="500rem" src="https://img.freepik.com/free-vector/follow-me-social-business-theme-design\_24877-50426.jpg?w=740&t=st=1665652644~exp=1665653244~hmac=c6269792f6b083040ab891f5db687a9b5c42c11f8f8b512b520e157fea8c1b4c" alt="">

</div>

<!-- Form end -->

</div>

</div>

Admindashboard.component.html

<div id="layoutSidenav">

<div id="layoutSidenav\_content">

<main>

<div class="container-fluid px-4">

<div><p class="h2 mt-4 mb-4">Dashboard</p></div>

<div class="row">

<div class="col-xl-3 col-md-6">

<div class="card bg-primary text-white mb-4">

<div class="card-body">New Card Requests</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">View Details</a><span class="badge badge-light ms-2">0</span>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

<div class="col-xl-3 col-md-6">

<div class="card bg-warning text-white mb-4">

<div class="card-body">Correction Requests</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">View Details</a><span class="badge badge-danger ms-2">1</span>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

<div class="col-xl-3 col-md-6">

<div class="card bg-success text-white mb-4">

<div class="card-body">Approved Requests</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">View Details</a><span class="badge badge-danger ms-2">3</span>

<div class="small text-white"><i class="fa fa-angle-right"></i></div>

</div>

</div>

</div>

<div class="col-xl-3 col-md-6">

<div class="card bg-danger text-white mb-4">

<div class="card-body">Delete Requests</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">View Details</a><span class="badge badge-light ms-2">0</span>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

</div>

<div class="card mb-4">

<div class="card-header">

<i class="fas fa-table me-1"></i>

Issued citizen Data

</div>

<div class="card-body">

<table id="datatablesSimple" class="table table-hover">

</tbody>

</table>

</div>

</div>

</div>

</main>

<footer class="py-4 bg-light mt-auto">

<div class="container-fluid px-4">

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

<div class="text-muted">Copyright &copy; Varniraj Service PVT. LTD 2022</div>

<div>

<a href="#">Privacy Policy</a>

&middot;

<a href="#">Terms &amp; Conditions</a>

</div>

</div>

</div>

</footer>

</div>

</div>

Dashboard.html

<div class="container">

<h2>Your Card</h2>

<div>

<div class="card" style="width: 638px;" >

<div class="card-header">

<i class="fa-solid fa-id-card"></i>

Preview

</div>

<div class="card-body"><img src="https://newsd.in/wp-content/uploads/2019/01/aadhar-card-sample-picture.jpg" alt="" width="100%" height="100%"></div>

</div>

<button class="btn btn-primary ml-5" (click)="gotoUpdateAadhar()">

Update aadhar details

</button>

</div>

</div>

<div id="layoutSidenav">

<div id="layoutSidenav\_content">

<main>

<div class="container-fluid px-4">

<div><p class="h2 mt-4 mb-4">User Dashboard</p></div>

<div class="row">

<div class="col-xl-3 col-md-6">

<div class="card bg-primary text-white mb-4">

<div class="card-body">New Card Requests</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">Apply</a>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

<div class="col-xl-3 col-md-6">

<div class="card bg-warning text-white mb-4">

<div class="card-body">Correction Request</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">Send</a>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

<div class="col-xl-3 col-md-6">

<div class="card bg-danger text-white mb-4">

<div class="card-body">Delete Request</div>

<div class="card-footer d-flex align-items-center">

<a class="small text-white stretched-link" href="#">Send</a>

<div class="small text-white"><i class="fas fa-angle-right"></i></div>

</div>

</div>

</div>

</div>

</div>

</main>

<footer class="py-4 bg-light mt-auto">

<div class="container-fluid px-4">

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

<div class="text-muted">Copyright &copy; Varniraj Service PVT. LTD 2022</div>

<div>

<a href="#">Privacy Policy</a>

&middot;

<a href="#">Terms &amp; Conditions</a>

</div>

</div>

</div>

</footer>

</div>

</div>

ServletInitializer.java

package com.admin;

import org.springframework.boot.builder.SpringApplicationBuilder;

import org.springframework.boot.web.servlet.support.SpringBootServletInitializer;

public class ServletInitializer extends SpringBootServletInitializer {

@Override

protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {

return application.sources(MAadharAdminApplication.class);

}

}

AdminController.java

package com.admin.controller;

import java.util.List;

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

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.RequestParam;

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

import com.admin.model.CardDetails;

import com.admin.model.Credentials;

import com.admin.model.UserDetails;

import com.admin.model.UserRequests;

import com.admin.service.AdminServices;

@RestController

@RequestMapping("/admin")

public class AdminController {

@Autowired

AdminServices service;

@PostMapping("/signin")

public boolean signin(@RequestBody Credentials cred) {

Credentials c=service.read(cred.getuName());

if(c.getuName().matches(cred.getuName()) && c.getPass().matches(cred.getPass())) {

return true;

}

else {

return false;

}

}

@GetMapping("/allData")

public List<CardDetails> viewAllData(){

return service.getAllCards();

}

@GetMapping("/pending")

public List<CardDetails> viewPendingCardReq(){

return service.PendingReq();

}

@GetMapping("/approved")

public List<CardDetails> viewApprovedCards(){

return service.ApprovedReq();

}

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

public boolean deleteCard(@PathVariable long id) {

return service.deleteCard(id);

}

@PostMapping("/add")

public boolean addData(@RequestBody CardDetails card) {

service.addCard(card);

return true;

}

@PostMapping("/updateCard/{id}")

public void updateState(@RequestBody CardDetails newData) {

service.updateCardDetails(newData);

}

@PostMapping("/applyCard")

public Long applyCard(@RequestBody UserDetails data) {

CardDetails application = new CardDetails();

application.setName(data.getName());

application.setAddress(data.getAddress());

application.setDob(data.getDob());

application.setEmail(data.getEmail());

application.setGender(data.getGender());

application.setMobile(data.getMobileNo());

application.setState("Pending");

service.addCard(application);

return service.find(data.getName());

}

@PostMapping("/submitRequest")

public boolean submitRequest(@RequestBody UserDetails ud,String reqName) {

UserRequests req=new UserRequests();

req.setName(ud.getName());

req.setEmail(ud.getEmail());

req.setAddress(ud.getAddress());

req.setDob(ud.getDob());

req.setGender(ud.getGender());

req.setMobileNo(ud.getMobileNo());

req.setReqName(reqName);

req.setReqState("Pending");

req.setPass(ud.getPass());

return service.submitRequest(req);

}

@GetMapping("/pendingReq")

public List<UserRequests> viewPending(){

return service.viewPending();

}

}

CardDetails.java

package com.admin.model;

import java.sql.Date;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import org.hibernate.annotations.ColumnDefault;

@Entity

public class CardDetails {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private long citizenId;

private String name;

private Date dob;

private String address;

private String email;

private long mobile;

private String gender;

@ColumnDefault("Pending")

private String state;

@Column(nullable = true)

private Date issueDate;

@Column(nullable = true)

private long aCardId;

public long getCitizenId() {

return citizenId;

}

public void setCitizenId(long citizenId) {

this.citizenId = citizenId;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public Date getDob() {

return dob;

}

public void setDob(Date dob) {

this.dob = dob;

}

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 long getMobile() {

return mobile;

}

public void setMobile(long mobile) {

this.mobile = mobile;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public String getState() {

return state;

}

public void setState(String state) {

this.state = state;

}

public Date getIssueDate() {

return issueDate;

}

public void setIssueDate(Date issueDate) {

this.issueDate = issueDate;

}

public long getaCardId() {

return aCardId;

}

public void setaCardId(long aCardId) {

this.aCardId = aCardId;

}

public CardDetails(long citizenId, String name, Date dob, String address, String email, long mobile, String gender,

String state, Date issueDate, long aCardId) {

super();

this.citizenId = citizenId;

this.name = name;

this.dob = dob;

this.address = address;

this.email = email;

this.mobile = mobile;

this.gender = gender;

this.state = state;

this.issueDate = issueDate;

this.aCardId = aCardId;

}

public CardDetails() {

super();

}

@Override

public String toString() {

return "CardDetails [citizenId=" + citizenId + ", name=" + name + ", dob=" + dob + ", address=" + address

+ ", email=" + email + ", mobile=" + mobile + ", gender=" + gender + ", state=" + state + ", issueDate="

+ issueDate + ", aCardId=" + aCardId + "]";

}

}

Credentials.java

package com.admin.model;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Credentials {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private long aId;

@Column

private String uName;

@Column

private String pass;

public String getuName() {

return uName;

}

public void setuName(String uName) {

this.uName = uName;

}

public String getPass() {

return pass;

}

public void setPass(String pass) {

this.pass = pass;

}

public Credentials(String uName, String pass) {

super();

this.uName = uName;

this.pass = pass;

}

public Credentials() {

super();

}

@Override

public String toString() {

return "Credentials [uName=" + uName + ", pass=" + pass + "]";

}

UserDetails.java

**package** com.admin.model;

**import** java.sql.Date;

**public** **class** UserDetails {

**private** String email;

**private** String pass;

**private** String name;

**private** Date dob;

**private** String address;

**private** **long** mobileNo;

**private** String gender;

**public** String getEmail() {

**return** email;

}

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

**this**.email = email;

}

**public** String getPass() {

**return** pass;

}

**public** **void** setPass(String pass) {

**this**.pass = pass;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** Date getDob() {

**return** dob;

}

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

**this**.dob = dob;

}

**public** String getAddress() {

**return** address;

}

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

**this**.address = address;

}

**public** **long** getMobileNo() {

**return** mobileNo;

}

**public** **void** setMobileNo(**long** mobileNo) {

**this**.mobileNo = mobileNo;

}

**public** String getGender() {

**return** gender;

}

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

**this**.gender = gender;

}

**public** UserDetails( String email, String pass, String name, Date dob, String address, **long** mobileNo,

String gender) {

**super**();

**this**.email = email;

**this**.pass = pass;

**this**.name = name;

**this**.dob = dob;

**this**.address = address;

**this**.mobileNo = mobileNo;

**this**.gender = gender;

}

**public** UserDetails() {

**super**();

}

@Override

**public** String toString() {

**return** "UserCredentials [ email=" + email + ", pass=" + pass + ", name=" + name + ", dob=" + dob

+ ", address=" + address + ", mobileNo=" + mobileNo + ", gender=" + gender + "]";

}

}

UserRequests.java

package com.admin.model;

import java.sql.Date;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class UserRequests {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private long reqId;

private String reqName;

private String reqState;

private String email;

private String pass;

private String name;

private Date dob;

private String address;

private long mobileNo;

private String gender;

public long getReqId() {

return reqId;

}

public void setReqId(long reqId) {

this.reqId = reqId;

}

public String getReqName() {

return reqName;

}

public void setReqName(String reqName) {

this.reqName = reqName;

}

public String getReqState() {

return reqState;

}

public void setReqState(String reqState) {

this.reqState = reqState;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPass() {

return pass;

}

public void setPass(String pass) {

this.pass = pass;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public Date getDob() {

return dob;

}

public void setDob(Date dob) {

this.dob = dob;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public long getMobileNo() {

return mobileNo;

}

public void setMobileNo(long mobileNo) {

this.mobileNo = mobileNo;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public UserRequests(long reqId, String reqName, String reqState, String email, String pass, String name, Date dob,

String address, long mobileNo, String gender) {

super();

this.reqId = reqId;

this.reqName = reqName;

this.reqState = reqState;

this.email = email;

this.pass = pass;

this.name = name;

this.dob = dob;

this.address = address;

this.mobileNo = mobileNo;

this.gender = gender;

}

public UserRequests() {

super();

}

@Override

public String toString() {

return "UserRequests [reqId=" + reqId + ", reqName=" + reqName + ", reqState=" + reqState + ", email=" + email

+ ", pass=" + pass + ", name=" + name + ", dob=" + dob + ", address=" + address + ", mobileNo="

+ mobileNo + ", gender=" + gender + "]";

}

}

AdminRepository.java

package com.admin.repository;

import org.springframework.data.repository.CrudRepository;

import org.springframework.stereotype.Repository;

import com.admin.model.Credentials;

@Repository

public interface AdminRepository extends CrudRepository<Credentials, Long> {

Credentials findByuName(String uName);

}

CardRepository.java

package com.admin.repository;

import java.util.List;

import org.springframework.data.repository.CrudRepository;

import com.admin.model.CardDetails;

public interface CardRepository extends CrudRepository<CardDetails,Long> {

List<CardDetails> findByState(String string);

CardDetails findByName(String name);

}

RequestRepository.java

package com.admin.repository;

import java.util.List;

import org.springframework.data.repository.CrudRepository;

import com.admin.model.UserRequests;

public interface RequestRepository extends CrudRepository<UserRequests, Long>

List<UserRequests> findByReqState(String string);

}

AdminServices.java

package com.admin.service;

import java.util.List;

import java.util.Optional;

import org.hibernate.engine.internal.Nullability;

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

import org.springframework.stereotype.Service;

import com.admin.model.CardDetails;

import com.admin.model.Credentials;

import com.admin.model.UserRequests;

import com.admin.repository.AdminRepository;

import com.admin.repository.CardRepository;

import com.admin.repository.RequestRepository;

@Service

public class AdminServices {

@Autowired

AdminRepository Repo;

@Autowired

CardRepository cardRepo;

@Autowired

RequestRepository reqRepo;

public void create(Credentials cred) {

Repo.save(cred);

}

public List<Credentials> readAll(){

return (List<Credentials>) Repo.findAll();

}

public Credentials read(String uName) {

return Repo.findByuName(uName);

}

public List<CardDetails> getAllCards(){

return (List<CardDetails>) cardRepo.findAll();

}

public void updateCardDetails(CardDetails newCardData) {

Optional<CardDetails> oldData=cardRepo.findById(newCardData.getCitizenId());

cardRepo.save(newCardData);

}

public boolean deleteCard(long citizenId) {

cardRepo.deleteById(citizenId);

return true;

}

public boolean addCard(CardDetails data) {

cardRepo.save(data);

return true;

}

public List<CardDetails> PendingReq(){

return cardRepo.findByState("Pending");

}

public List<CardDetails> ApprovedReq(){

return cardRepo.findByState("Approved");

}

public long find(String name) {

return cardRepo.findByName(name).getCitizenId();

}

public Iterable<UserRequests> allUserRequests() {

return reqRepo.findAll();

}

public boolean submitRequest(UserRequests req) {

reqRepo.save(req);

return true;

}

public List<UserRequests> viewPending(){

return reqRepo.findByReqState("Pending");

}

}

Application.properties

spring.jpa.database-platform=org.hibernate.dialect.MySQL5InnoDBDialect

spring.jpa.hibernate.ddl-auto=update

spring.datasource.driverClassName=com.mysql.cj.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/aadhar

spring.datasource.username=root

[spring.datasource.password=Simplilearn](mailto:spring.datasource.password=Jyotidevadiga@371)

UserConfig.java

package com.user;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import org.springframework.web.client.RestTemplate;

@Configuration

public class UserConfig {

@Bean

public RestTemplate getRestTemplate() {

RestTemplate restTemplate=new RestTemplate();

return restTemplate;

}

}

UserController.java

package com.user.controller;

import java.util.List;

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

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

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.RequestParam;

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

import com.user.model.UserCredentials;

import com.user.services.UserServices;

@RestController

@RequestMapping("/AadharApp")

public class UserController {

@Autowired

UserServices services;

@PostMapping("/login")

public boolean login(@RequestParam String email,@RequestParam String pass) {

return services.login(email, pass);

}

@PostMapping("/register")

public boolean register(@RequestBody UserCredentials uCred) {

return services.register(uCred);

}

@PostMapping("/applyCard")

public long applyCard(@RequestBody UserCredentials userCred) {

return services.applyAadhar(userCred);

}

@PostMapping("/sendRequest")

public boolean sendRequest(@RequestBody UserCredentials userCred,@RequestParam String reqName) {

return services.sendRequest(userCred, reqName);

}

}

UserCredentials.java

package com.user.model;

import java.sql.Date;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class UserCredentials {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private long uId;

private String email;

private String pass;

private String name;

private Date dob;

private String address;

private long mobileNo;

private String gender;

public long getuId() {

return uId;

}

public void setuId(long uId) {

this.uId = uId;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPass() {

return pass;

}

public void setPass(String pass) {

this.pass = pass;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public Date getDob() {

return dob;

}

public void setDob(Date dob) {

this.dob = dob;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public long getMobileNo() {

return mobileNo;

}

public void setMobileNo(long mobileNo) {

this.mobileNo = mobileNo;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public UserCredentials(long uId, String email, String pass, String name, Date dob, String address, long mobileNo,

String gender) {

super();

this.uId = uId;

this.email = email;

this.pass = pass;

this.name = name;

this.dob = dob;

this.address = address;

this.mobileNo = mobileNo;

this.gender = gender;

}

public UserCredentials() {

super();

}

@Override

public String toString() {

return "UserCredentials [uId=" + uId + ", email=" + email + ", pass=" + pass + ", name=" + name + ", dob=" + dob

+ ", address=" + address + ", mobileNo=" + mobileNo + ", gender=" + gender + "]";

}

}

UserServices.java

package com.user.services;

import java.net.URL;

import java.util.Arrays;

import java.util.HashMap;

import java.util.Map;

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

import org.springframework.http.HttpEntity;

import org.springframework.http.HttpHeaders;

import org.springframework.http.HttpMethod;

import org.springframework.http.MediaType;

import org.springframework.stereotype.Service;

import org.springframework.web.client.RestTemplate;

import com.user.model.UserCredentials;

import com.user.repository.UserRepository;

@Service

public class UserServices {

@Autowired

UserRepository Repo;

@Autowired

RestTemplate restTemplate;

public boolean login(String email, String pass) {

UserCredentials uCred= Repo.findByEmail(email);

if(uCred==null)

{

return false;

}

if(uCred.getEmail().matches(email)&&uCred.getPass().matches(pass))

{

return true;

}

else

return false;

}

public boolean register(UserCredentials uCred) {

Repo.save(uCred);

return true;

}

public Iterable<UserCredentials> getAll(){

return Repo.findAll();

}

public Long applyAadhar(UserCredentials userCred) {

String url="http://localhost:8080/admin/applyCard";

HttpHeaders headers = new HttpHeaders();

headers.setAccept(Arrays.asList(MediaType.APPLICATION\_JSON));

HttpEntity<UserCredentials> entity = new HttpEntity<UserCredentials>(userCred,headers);

return restTemplate.exchange(

url, HttpMethod.POST, entity, Long.class).getBody();

}

public boolean sendRequest(UserCredentials userCred,String reqName) {

String url="http://localhost:8080/admin/submitRequest";

HttpHeaders headers = new HttpHeaders();

headers.setAccept(Arrays.asList(MediaType.APPLICATION\_JSON));

HttpEntity<UserCredentials> entity = new HttpEntity<UserCredentials>(userCred,headers);

Map<String, String> map = new HashMap<>();

map.put("reqName", reqName);

return restTemplate.exchange(url, HttpMethod.POST, entity, Boolean.class, map).getBody();

// return restTemplate.exchange(

// url, HttpMethod.POST, entity, Boolean.class).getBody();

}

}