mAadhar Application source code

Front-end sample code

Add-addhar.component.html

```
<div class="container">
    <h2>Enter Your details for Aadhar card</h2>
    <form [formGroup]="aadharRef" (ngSubmit)="storeAadhar()">
        <label>Full Name</label><br/>
        <input type="text" formControlName="cname" class="form-control"><br/>
        <label>Date of Birth</label><br/>
        <input type="text" formControlName="dob" class="form-control"><br/>
        <label>Address</label><br/>
        <input type="text" formControlName="address" class="form-control"><br/>
        <label>Gender</label><br/>
        <input type="text" formControlName="gender" class="form-control"><br/>
        <label>Mobile number</label><br/>
        <input type="text" formControlName="number" class="form-control"><br/>
        <label>Email Id</label><br/>
        <input type="text" formControlName="emailid" class="form-control"><br/>
        <input type="submit" value="Submit" class="btn btn-primary"/>
        <input type="reset" value="reset" class="btn btn-secondary"/><br/>
    </form><br/>
    <span style="color: red">{{storeMsg}}</span>
</div>
```

Add-aadhar.component.ts

```
import { Component, OnInit } from '@angular/core';
import { FormControl, FormGroup } from '@angular/forms';
import { AadharService } from '../aadhar.service';

@Component({
  selector: 'app-add-aadhar',
  templateUrl: './add-aadhar.component.html',
  styleUrls: ['./add-aadhar.component.css']
})
export class AddAadharComponent implements OnInit {

  aadharRef = new FormGroup({
    cname:new FormControl(),
    dob:new FormControl(),
    address:new FormControl(),
    gender:new FormControl(),
    emailid:new FormControl(),
    number:new FormControl()
  })
  storeMsg : string=""
  constructor(public ps:AadharService) { }

  ngOnInit(): void {
  }

  storeAadhar(){
    let aadhar = this.aadharRef.value;
    this.ps.storeAadhar(aadhar).subscribe({
      next:(result:any)=>this.storeMsg=result,
      error:(error:any)=>console.log(error),
      complete:()=>console.log("completed")
    })
    this.aadharRef.reset();
  }

}
```

Login
Login.component.html

```
<div id="listofservice">
    <h3>Available Service</h3>
    <ul class="list-group">
        <li class="list-group-item">Aadhar Apply</li>
        <li class="list-group-item">Update Aadhar</li>
        <li class="list-group-item">Aadhar Status</li>
        <li class="list-group-item">Deletion of Aadhar</li>
    </ul>
</div>
<div id="container">
    <h2>Login Here</h2>
    <form [formGroup]="loginRef" (ngSubmit)="signIn()">
        <label class="form-label">EmailId</label>
        <input type="email" formControlName="emailid" class="form-
control"/><br/>
        <label class="form-label">Password</label>
        <input type="password" formControlName="password" class="form-
control"/><br/>
        <label>Type Of User</label>
        <input type="radio" name="typeOfUser" value="admin"
formControlName="typeOfUser" class="form-check-input"/>Admin
        <input type="radio" name="typeOfUser" value="user"
formControlName="typeOfUser" class="form-check-input"/>User<br/>

        <input type="submit" value="Sign In" class="btn btn-primary"/>
        <input type="reset" value="Reset" class="btn btn-secondary"/>
        <a routerLink="/signUp" class="btn btn-success">New User</a>
    </form>
    <br/>
    <span style="color:red">{{ msg }}</span>
</div>
```

Login.component.ts

```
import { Component, OnInit } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';
import { Router } from '@angular/router';
import { LoginService } from '../login.service';

@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent implements OnInit {
  loginRef = new FormGroup({
    emailid: new FormControl(),
    password: new FormControl(),
    typeOfUser: new FormControl('user') // Default user type
  });
  msg: string = '';

  constructor(public ls: LoginService, public router: Router) {}

  ngOnInit(): void {}

  signIn() {
    let login = this.loginRef.value;
    console.log(login); // Check if the form values are being captured
correctly

    this.ls.signIn(login).subscribe({
      next: (result: any) => {
        console.log(result); // Log the result to debug
        if (result === 'Admin Successfully login') {
          sessionStorage.setItem('userDetails', login.emailid);
          this.router.navigate(['/adminHome']);
        } else if (result === 'User successfully login') {
          sessionStorage.setItem('userDetails', login.emailid);
          this.router.navigate(['/userHome']);
        } else {
          this.msg = result; // Make sure the response from the server matches
your expectations
        }
      },
      error: (error: any) => {
        console.error(error); // Log the error for debugging
        this.msg = 'An error occurred.';
      },
      complete: () => console.log('completed')
    });
  }
}
```

Signup

Sinup.component.html

```
<div
  style="
    display: flex;
    flex-direction: column;
    align-items: center;
    justify-content: center;
    height: 100vh;
  "
>
  <h3>Create Account</h3>
  <form [formGroup]="loginRef" (ngSubmit)="signUp()" style="width: 300px">
    <label>EmailId</label><br />
    <input type="email" formControlName="emailid" /><br />
    <label>Password</label><br />
    <input type="password" formControlName="password" /><br />
    <label>TypeOfUser</label><br />
    <input
      type="radio"
      name="typeOfUser"
      value="admin"
      formControlName="typeOfUser"
    />admin
    <input
      type="radio"
      name="typeOfUser"
      value="user"
      formControlName="typeOfUser"
    />user<br />
    <input type="submit" value="signUp" />
    <input type="reset" value="reset" />
  </form>
  <br />
  <span style="color: red">{{ msg }}</span
  ><br />
  <a routerLink="/login">login</a>
</div>
```

Signup.component.ts

```
import { Component, OnInit } from '@angular/core';
import { FormControl, FormGroup } from '@angular/forms';
import { LoginService } from '../login.service';

@Component({
  selector: 'app-signup',
  templateUrl: './signup.component.html',
  styleUrls: ['./signup.component.css']
})
export class SignupComponent implements OnInit {
  loginRef = new FormGroup({
    emailid:new FormControl(),
    password:new FormControl(),
    typeOfUser:new FormControl()
  });
  msg:string=""

  constructor(public ls:LoginService) { }

  ngOnInit(): void {
  }
  signUp() {
    let login = this.loginRef.value;
    this.ls.signUp(login).subscribe({
      next:(result:any)=>this.msg=result,
      error:(error:any)=>console.log(error),
      complete:()=>console.log("completed")
    })
  }

}
```

Add.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { HttpClientModule } from '@angular/common/http';
import { FormsModule, ReactiveFormsModule } from '@angular/forms';
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { LoginComponent } from './login/login.component';
import { SignupComponent } from './signup/signup.component';
import { AdmindashboardComponent } from
'./admindashboard/admindashboard.component';
import { UserdashboardComponent } from
'./userdashboard/userdashboard.component';
```

```
import { AddAadharComponent } from './add-aadhar/add-aadhar.component';
import { AdminAadharViewComponent } from './admin-aadhar-view/admin-aadhar-view.component';
import { UpdateAadharComponent } from './update-aadhar/update-aadhar.component';
import { DeleteAadharComponent } from './delete-aadhar/delete-aadhar.component';

@NgModule({
  declarations: [
    AppComponent,
    LoginComponent,
    SignupComponent,
    AdmindashboardComponent,
    UserdashboardComponent,
    AddAadharComponent,
    AdminAadharViewComponent,
    UpdateAadharComponent,
    DeleteAadharComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    HttpClientModule,
    FormsModule,
    ReactiveFormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

add.component.ts

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'frontend-app';
}
```

App.routing.module.ts

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { AddAadharComponent } from './add-aadhar/add-aadhar.component';
import { AdminAadharViewComponent } from './admin-aadhar-view/admin-aadhar-view.component';
import { AdmindashboardComponent } from './admindashboard/admindashboard.component';
import { DeleteAadharComponent } from './delete-aadhar/delete-aadhar.component';
import { LoginComponent } from './login/login.component';
import { SignupComponent } from './signup/signup.component';
import { UpdateAadharComponent } from './update-aadhar/update-aadhar.component';
import { UserdashboardComponent } from './userdashboard/userdashboard.component';

const routes: Routes = [
  { path: 'login', component: LoginComponent },
  { path: 'adminHome', component: AdmindashboardComponent },
  { path: 'userHome', component: UserdashboardComponent },
  { path: 'applyforaadhar', component: AddAadharComponent },
  { path: 'signUp', component: SignupComponent },
  { path: 'updateAadhar', component: UpdateAadharComponent },
  { path: 'deleteAadhar', component: DeleteAadharComponent },
  { path: 'findAllAadhar', component: AdminAadharViewComponent },
  { path: '', redirectTo: 'login', pathMatch: 'full' },
  // Add more routes as needed...
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {}
```

backend code sample

MAadharAppApplication.java

```
package com.ecommerce.main;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.autoconfigure.domain.EntityScan;
import org.springframework.data.jpa.repository.config.EnableJpaRepositories;

@SpringBootApplication(scanBasePackages = "com")
@EntityScan(basePackages = "com.mAadhar.bean")
@EnableJpaRepositories(basePackages = "com.mAadhar.repository")
public class MAadharAppApplication {

	public static void main(String[] args) {
        SpringApplication.run(MAadharAppApplication.class, args);
        System.out.println("Server running on port number 9091");
    }


}
```

Aadhar.java

```
package com.mAadhar.bean;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;

@Entity
public class Aadhar {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int cid;
    private String cname;
    private String dob;
    private String address;
    private String emailid;
    private String number;
    private String gender;
    public int getCid() {
        return cid;
    }
    public void setCid(int cid) {
        this.cid = cid;
    }
    public String getCname() {
        return cname;
    }
    public void setCname(String cname) {
        this.cname = cname;
    }
    public String getDob() {
        return dob;
    }
    public void setDob(String dob) {
        this.dob = dob;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public String getEmailid() {
        return emailid;
    }
    public void setEmailid(String emailid) {
        this.emailid = emailid;
    }
    public String getNumber() {
        return (String) number;
    }
    public void setNumber(String number) {
        this.number = (String) number;
    }
    public String getGender() {
        return gender;
    }
    public void setGender(String gender) {
        this.gender = gender;
    }
    @Override
    public String toString() {
        return "Aadhar [cid=" + cid + ", cname=" + cname + ", dob=" + dob + ",
address=" + address + ", emailid="
                + emailid + ", number=" + number + ", gender=" + gender + "]";
    }


}
```

Login.java

```
package com.mAadhar.bean;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;
import jakarta.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 setEmailis(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;
	}


}
```

AadharController.java

```
package com.mAadhar.controller;

import java.util.List;

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.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PatchMapping;
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.mAadhar.bean.Aadhar;
import com.mAadhar.service.AadharService;



@RestController
@RequestMapping("aadhar")
@CrossOrigin(origins="*")
public class AadharController {
	@Autowired
	AadharService aadharService;
	@PostMapping(value = "storeAadhar", consumes =
MediaType.APPLICATION_JSON_VALUE)
	public String storeAadhar(@RequestBody Aadhar aadhar) {
		return aadharService.storeAadhar(aadhar);
	}
	@PatchMapping(value = "updateAadhar", consumes =
MediaType.APPLICATION_JSON_VALUE)
	public String updateAadhar(@RequestBody Aadhar aadhar) {
		return aadharService.updateAadhar(aadhar);
	}
	@GetMapping(value = "findAllAadhar", produces =
MediaType.APPLICATION_JSON_VALUE)
	public List<Aadhar> getAllAadhar(){
		return aadharService.getAllAadhar();
	}
	@GetMapping(value = "findAllAadhar/{cid}")
	public String findAadharById(@PathVariable("cid") int cid) {
		return aadharService.findAadharById(cid);
	}
	@DeleteMapping(value = "deleteAadhar/{cid}")
	public String deleteAadharUsingId(@PathVariable("cid") int cid) {
		return aadharService.deleteAadhar(cid);
	}

}
```

LoginController.java

```
package com.mAadhar.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.mAadhar.bean.Login;
import com.mAadhar.service.LoginService;




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.mAadhar.bean.Login;
import com.mAadhar.service.LoginService;



@RestController
@RequestMapping("login")
@CrossOrigin(origins="*")
public class LoginController {
        @Autowired
        LoginService loginService;
        @PostMapping(value = "signIn", consumes =
MediaType.APPLICATION_JSON_VALUE)
        public String signIn(@RequestBody Login login) {
                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);
        }
}
```

AadharRepository.java

```
package com.mAadhar.repository;

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

import com.mAadhar.bean.Aadhar;

public interface AadharRepository extends JpaRepository<Aadhar, Integer>{

}
```

LoginRepository.java

```
package com.mAadhar.repository;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.mAadhar.bean.Login;
```

```
@Repository
public interface LoginRepository extends JpaRepository<Login, String>{

}
```

AadharService.java

```
package com.mAadhar.service;

import java.util.List;
import java.util.Optional;


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

import com.mAadhar.bean.Aadhar;
import com.mAadhar.repository.AadharRepository;

@Service
public class AadharService {
    @Autowired
    AadharRepository aadharRepository;

    public String storeAadhar(Aadhar aadhar) {
        aadharRepository.save(aadhar);
        return "Aadhar Details Submitted Succesfully";
    }
    public List<Aadhar> getAllAadhar(){
        return aadharRepository.findAll();
    }
    public String findAadharById(int cid) {
        Optional<Aadhar> result = aadharRepository.findById(cid);
        if(result.isPresent()) {
            Aadhar a = result.get();
            return a.toString();
        }else {
            return "Aadhar Details not found";
        }
    }
    public String deleteAadhar(int cid) {
        Optional<Aadhar> result = aadharRepository.findById(cid);
        if(result.isPresent()) {
            Aadhar a = result.get();
            aadharRepository.delete(a);
            return "Aadhar Deleted Succesfully";
        }else {
            return "Aadhar details didn't match with record";
        }
    }
    public String updateAadhar(Aadhar aadhar) {
        Optional<Aadhar> result = aadharRepository.findById(aadhar.getCid());
        if(result.isPresent()) {
            Aadhar a = result.get();
            a.setCname(aadhar.getCname());
            a.setAddress(aadhar.getAddress());
            a.setDob(aadhar.getDob());
            a.setEmailid(aadhar.getEmailid());
            a.setNumber(aadhar.getNumber());
            aadharRepository.saveAndFlush(a);
            return "Aadhar Details Updated Succesfully";
        }else {
            return "Aadhar details not present";
        }
    }

}
```

AadharService.java

```
package com.mAadhar.service;

import java.util.List;
import java.util.Optional;


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

import com.mAadhar.bean.Aadhar;
import com.mAadhar.repository.AadharRepository;

@Service
public class AadharService {
    @Autowired
    AadharRepository aadharRepository;

    public String storeAadhar(Aadhar aadhar) {
            aadharRepository.save(aadhar);
            return "Aadhar Details Submitted Succesfully";
    }
    public List<Aadhar> getAllAadhar(){
            return aadharRepository.findAll();
    }
    public String findAadharById(int cid) {
            Optional<Aadhar> result = aadharRepository.findById(cid);
            if(result.isPresent()) {
                    Aadhar a = result.get();
                    return a.toString();
            }else {
                    return "Aadhar Details not found";
            }
    }
    public String deleteAadhar(int cid) {
            Optional<Aadhar> result = aadharRepository.findById(cid);
            if(result.isPresent()) {
                    Aadhar a = result.get();
                    aadharRepository.delete(a);
                    return "Aadhar Deleted Succesfully";
            }else {
                    return "Aadhar details didn't match with record";
            }
    }
    public String updateAadhar(Aadhar aadhar) {
            Optional<Aadhar> result = aadharRepository.findById(aadhar.getCid());
            if(result.isPresent()) {
                    Aadhar a = result.get();
                    a.setCname(aadhar.getCname());
                    a.setAddress(aadhar.getAddress());
                    a.setDob(aadhar.getDob());
                    a.setEmailid(aadhar.getEmailid());
                    a.setNumber(aadhar.getNumber());
                    aadharRepository.saveAndFlush(a);
                    return "Aadhar Details Updated Succesfully";
            }else {
                    return "Aadhar details not present";
            }
    }

}
```

Dockerfile

```
FROM openjdk:8
COPY target/aadhar-backend.jar .
CMD ["java","-jar","aadhar-backend.jar"]
```

Application.properties

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/capstone_project
spring.datasource.username=root
spring.datasource.password=123456
spring.jpa.hibernate.ddl-auto=update
server.port=9091
```

pom.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
	xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
	<modelVersion>4.0.0</modelVersion>
	<parent>
		<groupId>org.springframework.boot</groupId>
		<artifactId>spring-boot-starter-parent</artifactId>
		<version>3.1.2</version>
		<relativePath/> <!-- lookup parent from repository -->
	</parent>
	<groupId>com.ecommerce</groupId>
	<artifactId>mAadhar</artifactId>
	<version>0.0.1-SNAPSHOT</version>
	<name>mAdhar</name>
	<description>Demo project for Spring Boot</description>
	<properties>
		<java.version>17</java.version>
	</properties>
	<dependencies>
		<dependency>
			<groupId>org.springframework.boot</groupId>
			<artifactId>spring-boot-starter-data-jpa</artifactId>
		</dependency>
		<dependency>
			<groupId>org.springframework.boot</groupId>
			<artifactId>spring-boot-starter-web</artifactId>
		</dependency>

		
		<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-devtools</artifactId>
<scope>runtime</scope>
<optional>true</optional>
</dependency>
<dependency>
			<groupId>com.mysql</groupId>
			<artifactId>mysql-connector-j</artifactId>
			<scope>runtime</scope>
		</dependency>
		<dependency>
			<groupId>org.springframework.boot</groupId>
			<artifactId>spring-boot-starter-test</artifactId>
			<scope>test</scope>
		</dependency>
	</dependencies>
```

```
        <build>
                <plugins>
                        <plugin>
                                <groupId>org.springframework.boot</groupId>
                                <artifactId>spring-boot-maven-plugin</artifactId>
                        </plugin>
                </plugins>
        </build>

</project>
```