Backend:

Pom.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>2.7.4</version>
              <relativePath/> <!-- lookup parent from repository -->
       </parent>
       <groupId>com.example</groupId>
       <artifactId>MAadharApp</artifactId>
       <version>0.0.1-SNAPSHOT</version>
       <name>MAadharApp</name>
       <description>Demo project for Spring Boot</description>
       <properties>
              <java.version>11</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>mysql</groupId>
                     <artifactId>mysql-connector-java</artifactId>
                     <scope>runtime</scope>
              </dependency>
              <dependency>
                     <groupId>org.springframework.boot</groupId>
                     <artifactId>spring-boot-starter-test</artifactId>
                     <scope>test</scope>
              </dependency>
              <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
    <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>4.2.1</version>
     </dependency>
      <dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.5</version>
    <scope>test</scope>
</dependency>
	</dependencies>

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

</project>
```

MAadharAppApplication.java:

```
package com.aadhar;

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.aadhar.bean")
@EnableJpaRepositories(basePackages = "com.aadhar.repository")
public class MAadharAppApplication {

	public static void main(String[] args) {
		SpringApplication.run(MAadharAppApplication.class, args);
		System.out.println("Server running on port number 9090");
	}

}
```

Login.Java:

```
package com.aadhar.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.Java:

```
package com.aadhar.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.java:

```
package com.aadhar.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.aadhar.bean.Login;
import com.aadhar.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.java:

```
package com.aadhar.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.aadhar.bean.User;
import com.aadhar.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);

	}

}
```

LoginService.java:

```
package com.aadhar.service;

import java.util.Optional;

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

import com.aadhar.bean.Login;
import com.aadhar.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.java:

```
package com.aadhar.service;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.aadhar.bean.User;
import com.aadhar.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;
		}
	}

}
```

LoginRepository.java:

```
package com.aadhar.repository;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;
import com.aadhar.bean.Login;
@Repository
public interface LoginRepository extends JpaRepository<Login, String>{

}
```

UserRepo.java:

```
package com.aadhar.repository;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;
import com.aadhar.bean.User;
@Repository
public interface UserRepo extends JpaRepository<User,Integer> {

}
```

Application.properties:

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/capstone
spring.datasource.username=root
spring.datasource.password=Kartheek99#
spring.jpa.hibernate.ddl-auto=update
server.port=9090
```

Testing:

MAadharAppApplicationTests.java:

```
package com.aadhar;

import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class MAadharAppApplicationTests {

	@Test
	void contextLoads() {
	}

}
```

MAadharAppApplicationTestsCode.java:

```
package com.aadhar;
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 MAadharAppApplicationTestsCode {
	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() {
                    System.setProperty("webdriver.chrome.driver", 
"C:\\\\Users\\\\durgasindhu\\\\Downloads\\\\chromedriver_win32/chromedriver.exe");
                     driver = new ChromeDriver();
             }
             @AfterMethod
             public void afterMethod() {
                    //driver.close();
                     driver = null;
             }
}
```

Test.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite">
  <test thread-count="5" name="Test">
    <classes>
      <class name="com.aadhar.MAadharAppApplicationTestsCode"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```