**Sourcecode**

1. **ComplaintRedressalApplication.java**

package com.simplilearn.finalphase2;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.EnableAutoConfiguration;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.boot.autoconfigure.jdbc.DataSourceAutoConfiguration;

@SpringBootApplication

//@SpringBootApplication(exclude = {DataSourceAutoConfiguration.class })

//@EnableAutoConfiguration(exclude = {DataSourceAutoConfiguration.class })

public class ComplaintRedressalApplication {

public static void main(String[] args) {

SpringApplication.run(ComplaintRedressalApplication.class, args);

}

}

1. **Application.properties**

#configure database

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/complaint\_redressal\_db1

spring.datasource.username=root

spring.datasource.password=root12345

#spring.datasource.platform= mysql

spring.jpa.hibernate.ddl-auto = update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5InnoDBDialect

spring.main.allow-circular-references=true

#above line will allow the circular references to different beans

#spring.autoconfigure.exclude=org.springframework.boot.autoconfigure.jdbc.DataSourceAutoConfiguration

spring.jpa.generate-ddl=true

#Hibernate ddl auto (create, create-drop, validate, update)

spring.jpa.properties.hibernate.format\_sql=true

1. **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>2.7.9</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<groupId>com.simplilearn</groupId>

<artifactId>Backend1</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>Backend1</name>

<description>A capstone project for Complaint Redressal System</description>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</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>com.h2database</groupId>

<artifactId>h2</artifactId>

<scope>runtime</scope>

</dependency>

<!--<dependency>

<groupId>javax.xml.bind</groupId>

<artifactId>jaxb-api</artifactId>

</dependency>-->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt</artifactId>

<version>0.9.1</version>

</dependency>

<!--<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-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>

1. **User.java**

package com.simplilearn.finalphase2.entity;

import java.util.Set;

import javax.persistence.CascadeType;

import javax.persistence.Entity;

import javax.persistence.FetchType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.JoinTable;

import javax.persistence.ManyToMany;

import javax.persistence.Table;

@Entity

@Table(name="users")

public class User {

@Id

private String userName;

private String fullName;

private String userpassword;

// many uses may have many different roles

/\*

\* By itself create a third table named USER\_ROLE that will have user and its

\* associated role details this third table will have user\_id and associated

\* role\_id

\*/

@ManyToMany(fetch = FetchType.EAGER, cascade = CascadeType.ALL)

@JoinTable(name = "USER\_ROLE", joinColumns = { @JoinColumn(name = "USER\_ID") }, inverseJoinColumns = {

@JoinColumn(name = "ROLE\_ID") })

private Set<Role> role;

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getFullName() {

return fullName;

}

public void setFullName(String fullName) {

this.fullName = fullName;

}

public String getUserpassword() {

return userpassword;

}

public void setUserpassword(String userpassword) {

this.userpassword = userpassword;

}

public Set<Role> getRole() {

return role;

}

public void setRole(Set<Role> role) {

this.role = role;

}

}

1. **App.component.html**

<div class="bg">

<div>

<app-header></app-header>

</div>

<div >

<router-outlet></router-outlet>

</div>

<footer class="bg-dark text-center text-white">

<!-- Copyright -->

<div class="text-center p-3" style="background-color: rgb(169,169,169)">

© 2023 Copyright: ABC Telecom || Developed By: Rashmi Rathi || Simplilearn.

<!-- <a class="text-white" href="https://mdbootstrap.com/">MDBootstrap.com</a> -->

</div>

<!-- Copyright -->

</footer>

</div>

1. **style.css**

html, body { height: 100%; }

body { margin: 0 ; font-family: Roboto, "Helvetica Neue", sans-serif; }

.bg{

background-image: url(/assets/telecom-bg1.jpg);

height: 650px;

width: 1149px;

align-items: center;

/\* background-repeat: no-repeat; \*/

}

1. **index.html**

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>ComplaintRedressal</title>

  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

  <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500&display=swap" rel="stylesheet">

  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

</head>

<body class="mat-typography">

  <app-root></app-root>

</body>

</html>

1. **login.component.html**

<div class="container mt-5">

<div clas="card" style="padding: 50px;padding-right: 300px; padding-left: 300px;background-color:#F0FFFF">

<form #loginForm="ngForm" (ngSubmit)="login(loginForm)">

<div class="mb-2 text-center" style="text-align: center;color:red;font-style: italic;font-size: larger;">

{{msg}}

</div>

<input type="text" ngModel name="userName" id="userName" placeholder="Enter username"

style="padding: 15px;" class="form-control mb-4 ml-5 ml-5">

<input type="password" ngModel name="userPassword" id="userPassword" placeholder="Enter username" style="padding: 15px;"

class="form-control mb-4 ml-5 mr-5">

<div class=" ml-5 mr-5" style="padding: 15px;background-color: #F0FFFF">

<input type="submit" value="Login" class="btn btn-outline-primary form-control rounded-2 ml-10 mr-10 mb-2 " />

<input type="reset" value="Reset" class="btn btn-outline-warning form-control rounded-2 ml-10 mr-10" />

</div>

</form>

</div>

<!-- outside card -->

<div class="text-center">

<button mat-raised-button type="button" color="primary" class="mt-2 mb-5" (click)="registerUser()" >Create customer account</button>

</div>

</div>

1. [**home.component.html**](http://home.component.html)
2. <div class="container">
3. <p class="mt-3">
4. ABC Telecom is India's foremost and truly-integrated telecommunications service provider, with a corporate customer base of over 40,000 Indian and multinational corporations, including small and medium enterprises, and close to 300 of the finest enterprise and carrier companies globally.
5. </p>
7. <p class="mt-2">
8. ABC Telecom touches 90 per cent of the country's population, supported by an OFC network of over 190,000 km. The Company also has a significant tower infrastructure that caters to other telecommunications operators.
9. </p>
10. <p class="mt-2" >
11. Spread across India's top 29 cities, ABC Telecom' data solutions for Homes and Small Enterprises cater to buildings connected to the Reliance network. Reliance Communications also owns the world’s largest private under-sea cable system under its subsidiary Global Cloud Xchange, which offers a comprehensive portfolio of solutions customized for Carriers, Enterprises and New Media Companies globally.
12. </p>
13. <p class="mt-2">
14. GCX's under-sea cable system spans more than 67,000 route km, which, seamlessly integrated with Reliance Communications' 200,000 route km of domestic optic fiber backbone, provides a robust Global Service Delivery Platform for Enterprises. GCX is uniquely equipped to support businesses through the deployment of Next-Generation Enterprise solutions across its Cloud Delivery Networks. Since its inception in December 2002, Reliance Communications has been revolutionizing the way the world communicates, impacting not just lifestyles, but the very lives of its customers-globally
15. </p>
16. For further information, please visit: <a href="#">www.abctele.co.in</a>
17. <div class="row mt-2">
18. <div class="column">
19. <a><img src="/assets/telecom-1.jpg" alt="Snow" style="width:100%;height: 200px;"></a>
20. <p class="text-center"><b>Career</b></p>
21. </div>
22. <div class="column">
23. <a><img src="/assets/telecom-2.jpg" alt="Forest" style="width:100%;height: 200px;"></a>
24. <p class="text-center"><b>Our Offices</b></p>
25. </div>
26. <div class="column">
27. <a><img src="/assets/telecom-3.jpg" alt="Mountains" style="width:100%;height: 200px;"></a>
28. <p class="text-center"><b>Our Vision</b></p>
29. </div>
30. </div>
31. </div>

**10.** **allcomplaints.component.html**

<div class="row">

    <div class="col-md-12">

            <div class="card-header"><h1>Complaint Details:</h1></div>

        <div class="card-body" style="background-color: aliceblue;">

            <table class="table table-info">

                <tr style="text-align: center;">

                    <th style="padding: 5px;">ID</th>

                    <th style="padding: 5px;">Full Name</th>

                    <th style="padding: 5px;">Heading</th>

                    <th style="padding: 5px;">Details</th>

                    <!-- <th style="padding: 5px;">Director</th>

                    <th style="padding: 5px;">Language</th> -->

                    <th style="padding: 10px;">Address</th>

                    <th style="padding: 5px;">Pincode</th>

                    <th style="padding: 5px;">complaintStatus</th>

                </tr>

                <tr style="text-align: center;" \*ngFor="let c of complaintdetails">

                    <td style="padding: 5px;">{{c.complaintId}}</td>

                    <td style="padding: 5px;">{{c.fullname}}</td>

                    <td style="padding: 5px;">{{c.heading}}</td>

                    <td style="padding: 5px;">{{c.details}}</td>

                    <td style="padding: 5px;">{{c.address}}</td>

                    <!-- <td style="padding: 5px;">{{mt.director}}</td>

                    <td style="padding: 5px;">{{mt.language}}</td> -->

                    <td style="padding: 5px;">{{c.pincode}}</td>

                    <td style="padding: 5px;">{{c.complaintStatus}}</td>

                </tr>

            </table>

        </div>

    </div>

</div>

**11. customer-home.component.html**

<div class="container">

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

        <h1>Welcome!!</h1>

        <h2>Any complaints?</h2>

        <a routerLink="/complaintReg" style="color: red;"><h2>Raise your complaints here...</h2></a>

    </div>

</div>

**12. complaint-register.component.html**

<div class="container mt-5">

    <div class="card p-5" style="background-color:#012169;">

        <div class="row">

            <div class="col-12">

        <form #comaplaintForm="ngForm" (ngSubmit)="registerComplaint(comaplaintForm)" >

            <mat-form-field appearance="outline" class="w-100">

                <mat-label>Full Name: </mat-label>

                <input matInput placeholder="fullname" [(ngModel)]="complaintDetails.fullname" name="fullname" style="color: whitesmoke;">

            </mat-form-field>

            <!--Enter complaint category-->

            <!-- <mat-form-field appearance="fill">

                <mat-label>Category</mat-label>

                <mat-select [formControl]="Category" multiple>

                  <mat-select-trigger>

                    {{category.value?.[0] || ''}}

                    <span \*ngIf="(category.value?.length || 0) > 1" class="example-additional-selection">

                      (+{{(category.value?.length || 0) - 1}} {{category.value?.length === 2 ? 'other' : 'others'}})

                    </span>

                  </mat-select-trigger>

                  <mat-option \*ngFor="let topping of toppingList" [value]="topping">{{category}}</mat-option>

                </mat-select>

              </mat-form-field>

               -->

            <mat-form-field appearance="outline" class="w-100">

                <mat-label>Complaint Title: </mat-label>

                <input matInput placeholder="Complaint Title" [(ngModel)]="complaintDetails.heading" name="heading" style="color: whitesmoke;">

            </mat-form-field>

            <mat-form-field appearance="outline" class="w-100">

                <mat-label>Complaint Details: </mat-label>

                <textarea  type="text" matInput placeholder="Complaint Details" [(ngModel)]="complaintDetails.details" name="details" style="color: whitesmoke;"></textarea>

            </mat-form-field>

            <mat-form-field appearance="outline" class="w-100">

                <mat-label>Address: </mat-label>

                <input  matInput placeholder="Address" [(ngModel)]="complaintDetails.address" name="address" style="color: whitesmoke;" required>

            </mat-form-field>

            <mat-form-field appearance="outline" class="w-100">

                <mat-label>Pincode: </mat-label>

                <input  matInput placeholder="Pincode" [(ngModel)]="complaintDetails.pincode" name="pincode" style="color: whitesmoke;" required>

            </mat-form-field>

            <div align="centre">

                <button type="submit" mat-raised-button color="accent">Register Complaint</button>

             </div>

        </form>

    </div>

</div>

    </div>

</div>

**13. engineer-home.component.html**

<div class="container">

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

        <h1>Welcome Ankita!!</h1>

        <h2>Check Complaints Assigned to you...</h2>

    </div>

</div>

**14. header.component.html**

<mat-toolbar color="primary">

    <mat-toolbar-row >

        <span ><img src="/assets/telecom-icon2.jpg" style="width: 80px; height: 60px; cursor: pointer;" class="img-fluid rounded-lg" routerLink=""></span>

      <span class="ml-3"routerLink="" style="cursor: pointer;color: black;" >ABC Telecom</span>

      <span class="example-spacer"></span>

      <button mat-raised-button color="accent" \*ngIf="! isLoggedIn()" routerLink="/login">Login</button>

      <button mat-raised-button color="warn" class="ml-2" \*ngIf="isLoggedIn()" (click)="logout()">Logout</button>

    </mat-toolbar-row>

      <mat-toolbar-row color="primary" >

        <button mat-raised-button color="accent" \*ngIf="isCustomer()" routerLink="/customerHome">Dashboard</button>

        <button mat-raised-button color="accent" \*ngIf="isCustomer()" routerLink="/complaintReg" class="ml-2">Register Complaint</button>

        <button mat-raised-button color="accent" \*ngIf="isCustomer()" routerLink="/mycomplaints" class="ml-2">Previous complaints</button>

        <button mat-raised-button color="accent" \*ngIf="isAdmin()" routerLink="/adminHome">Dashboard</button>

        <button mat-raised-button color="accent" \*ngIf="isAdmin()" routerLink="/allcomplaints" class="ml-2" >All complaints</button>

        <button mat-raised-button color="accent" \*ngIf="isAdmin()" routerLink="/userList" class="ml-2">All Users</button>

        <button mat-raised-button color="accent" \*ngIf="isManager()" routerLink="/managerHome">Dashboard</button>

        <button mat-raised-button color="accent" \*ngIf="isManager()" routerLink="/allComplaintsManager" class="ml-2" >All complaints</button>

        <button mat-raised-button color="accent" \*ngIf="isEngineer()" routerLink="/complaintEngg" class="ml-2">Assigned Complaints</button>

      </mat-toolbar-row>

      </mat-toolbar>

**15. manager-home.html**

<div class="container">

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

        <h1>Welcome Manager!!</h1>

        <h2>Assign complaints </h2>

    </div>

</div>