CAPSTONE PROJECT REPORT
ON

MATURITY ASSESSMENT TOOL

UNDERTAKEN AT

Building a better
working world

EY-GDS

Submitted by Team-B

Giridharan B. (College: SRM University, Roll Number: RA1811032020033)
Antriksh Dwivedi (College: SRM University, Roll Number: RA1811032020022)
Khushi Kapoor (College: SRM University, Roll Number: RA1811004010337)
Md Aryan Khan (College: SRM University, Roll Number: RA1811005010211)
Parth Bhargava (College: SRM University, Roll Number: RA1811005010070)
Qazi Mamoon(College: SRM University, Roll Number: RA1811005010129)
Prashant Kumar (College: SRM University, Roll Number: RA1811004020093)
Souvik Pan (College: SRM University, Roll Number: RA1811005010312)

Under the supervision of

Mr Anish Jeyanth [Project Mentor. Email: *Anish.Jeyanth@gds.ey.com*]
Mr Ramakrishna [Course Trainer. Email: *dotnettrainer2k22@gmail.com*]

ABSTRACT

The issue of calculating the ability of an organization to analyze its employees has been a long dilemma. To compare the abilities with the competing organizations, to find out the most scoring employee/organization, to find out the most consistent employee/organization or to track and analyze the abilities of all the organizations comes hand in hand with huge calculational troubles.

This ability of the organization to master its project is termed Maturity, and the calculation of this ability can be termed Maturity Assessment.

This project focuses on creating a Maturity-Assessment-Tool Website.

Through a Maturity Assessment Tool, we can get the maturity score of a particular project member by answering the questions given in a survey/s under the given project and post evaluation of the survey questions, the Maturity of the Project Member can be easily assessed.

INTRODUCTION

The process of calculating the Maturity of a Project Member becomes an enormous issue as the number of surveys, projects and questions increases. Which is a problem that is bound to happen for the big organizations that have to compare their abilities with the competing organizations, improve their position amongst the most scoring organizations, maintain consistency and track and analyze the abilities of all the competitor organizations.

A Maturity Assessment Tool measures the respondent's maturity in one or more areas through a series of questions. A user will get to log in or create a new profile if they don't happen to have one already. If the user is an admin, they get to create a project and create surveys under it by adding the questions to the surveys. Then the admin gets to all the project members that are assigned to the project. If the user is a project member, they would get to see all the projects that are assigned to them and they get to access the surveys and answer the provided questions. Upon answering the questions and getting the results, the Maturity Assessment Tool will present the scores on the screen hence fulfilling the requirements.

PROJECT SCOPE

Problem Statement

To design a perfectly functional website in Angular which is connected to the Database and has an ASP.NET Web API that connects the Angular Front-end to the Back-end Database on SQL.

Objective

Finding the maturity score of a project member can be achieved by answering a set of the given question under the surveys of a project. The objective of this project is to design a tool that could assess the maturity of the given project member.

Required Hardware

A Functional Windows PC or Laptop
8 GB RAM
512 GB SSD or 1 TB HDD
200 GB Space which would all cumulate to cost around ₹ 50,000.

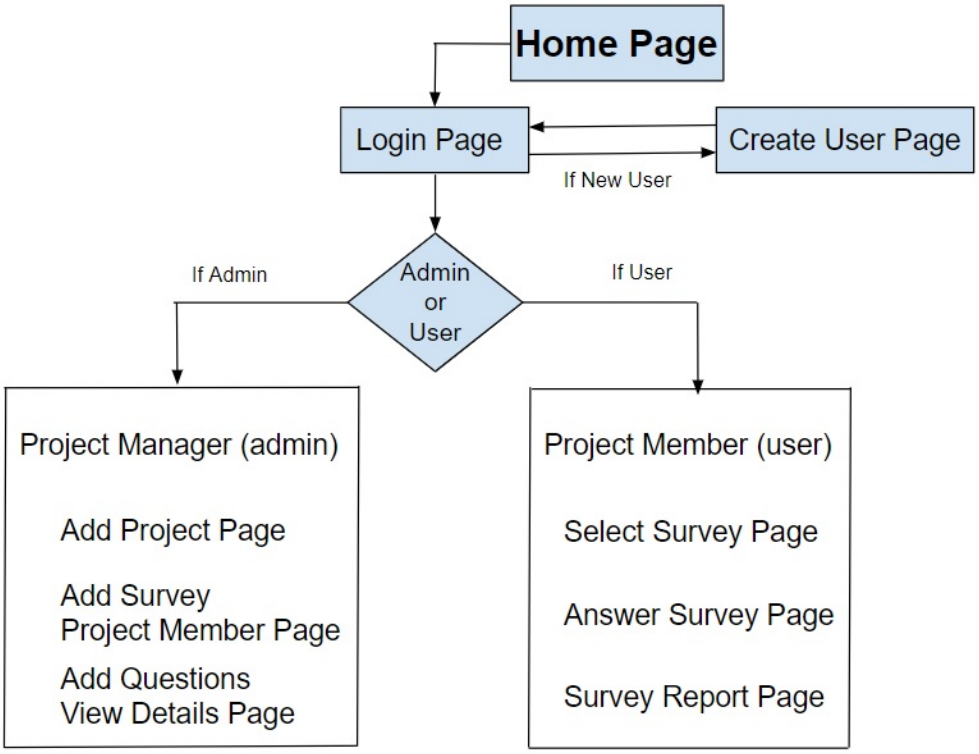
Required Software

Microsoft SQL Server
Visual Studio 2019
Visual Studio Code
A Web Browser (used Chrome Version 100.0.4896.88 (Official Build) (64-bit))
Postman (testing API)

Deliverables

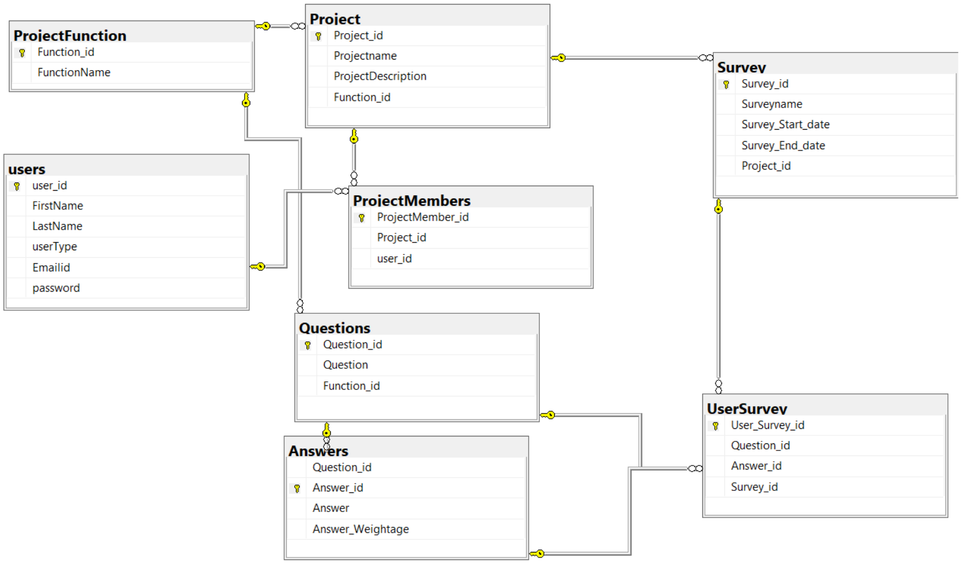
A perfectly functional product that streamlines the process of assessing the maturity of any project will be delivered.

FLOW CHART



The project will have two users: Admin and Team Member
The Admin logs into the account or creates a new account if necessary. The admin then creates projects and corresponding surveys, then selects the team members of the project and creates a questionnaire with weightage. The Team Member logs into their account and selects the project they are assigned to, selects the corresponding survey and answers the questions. Once the questions are answered the Maturity Assessment Tool calculates the maturity score of the project member.

E-R DIAGRAM



The user table will have the details of the registered user. Any newly created user will be stored in this table. The project table will have details of the project that will be created on the front end. The project members table will have the database of all the members that are assigned to a project by getting the foreign key. The questions table will contain all the predefined questions. The answer table will contain predefined answers corresponding to the questions by getting the foreign key from the question table. When a user creates a survey, the details will go inside the survey table. Since the user survey and project table have a many-to-many relationship, the survey table will act as a middle table to connect them both. The user survey table will have the details of the questions and answers belonging to that particular survey.

There will be two users in the project: an administrator and a team member.
If necessary, the Admin logs into the account or creates a new one. The administrator then establishes projects and surveys, selects project team members, and develops a questionnaire with weighting. The Team Member goes into their account and chooses the project to which they have been assigned, as well as the survey to which they have been assigned. After the questions are answered, the Maturity Assessment Tool generates the project member's maturity score.

SCREENSHOTS OF THE PROJECT

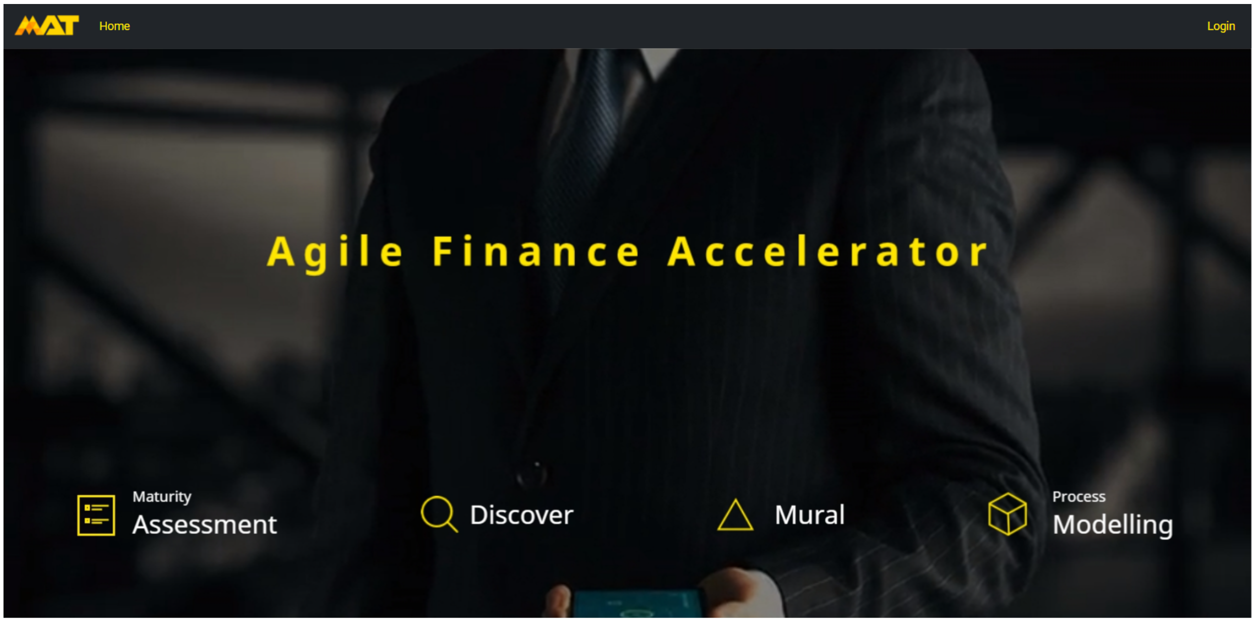


Image 1.1 - Project Home Page

This is the Home Page of the project Maturity Assessment Tool. When the server starts and the files are executed, this page shows up primarily. At the top right corner, we can locate the login button in the navbar. Clicking that will redirect the user to the Login Page.

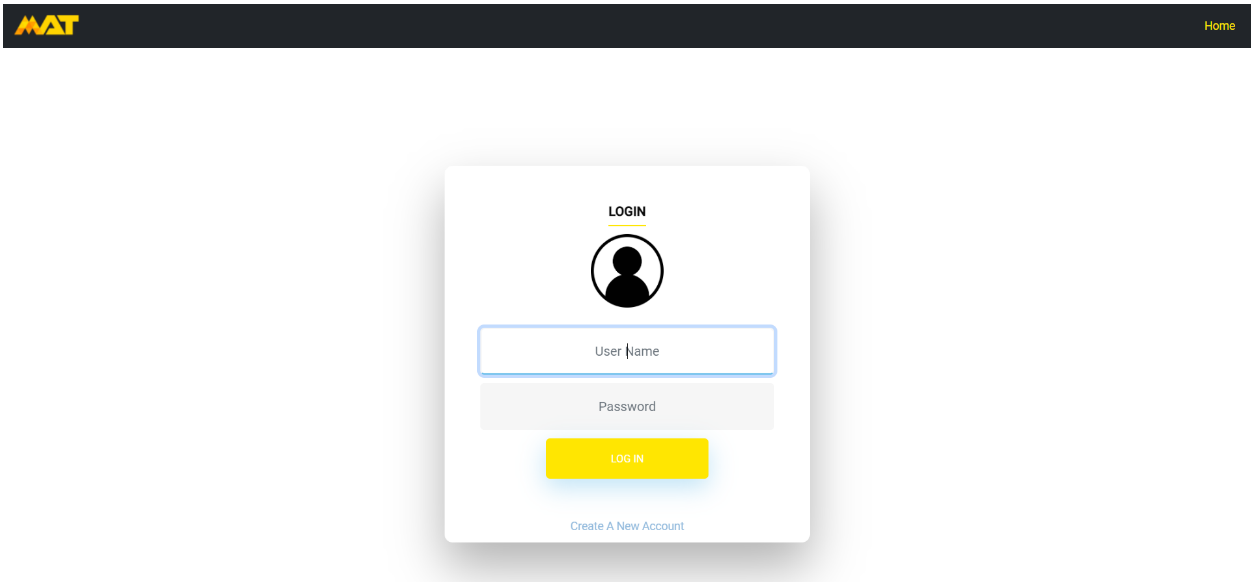


Image 1.2 - Login Page

This is the Login Page. This page shows up when the user clicks on the Login button in the navbar of the Home Page. The user has to give their Email ID and correct password to log into their account. If the user does not have an account already, they can create it by clicking the “Create A New Account” at the bottom. Any unregistered Email and/or wrong password will not let the user in. It will prompt a “Wrong Username or Password” message.

Upon clicking the Login button, the user will be redirected to the login account.

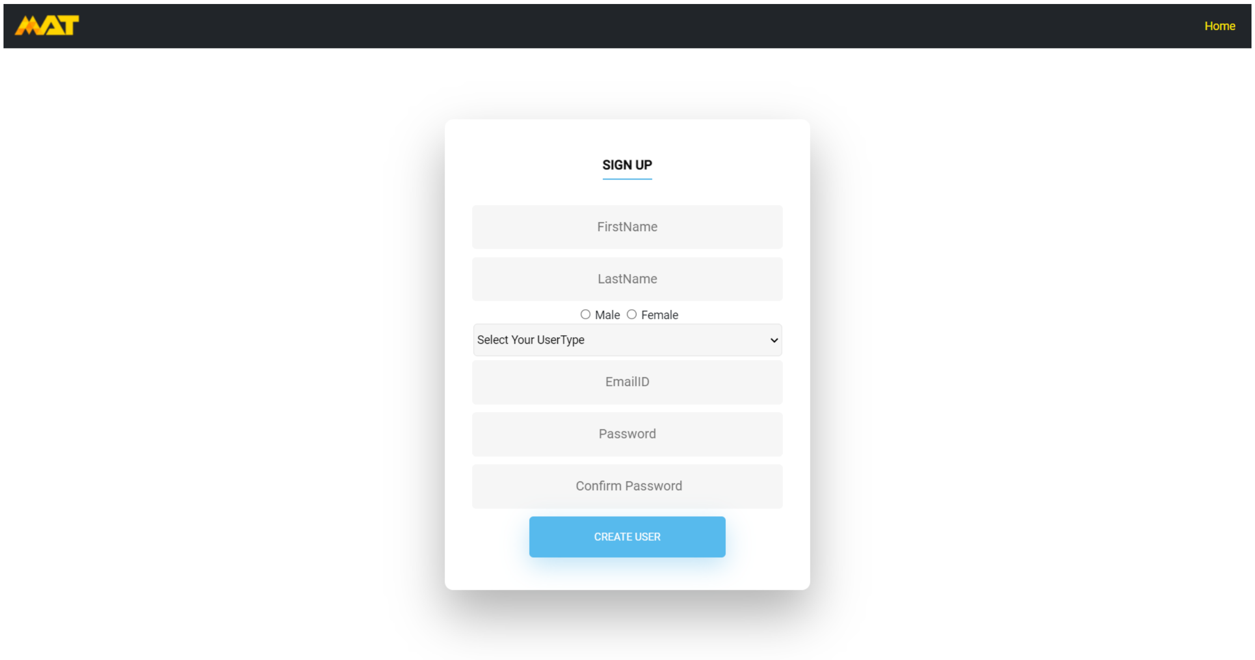


Image 1.3 - Sign-Up Page

This page shows up if the user clicks on “Create A New Account” from the Login Page. The user has to fill up the details. This is a validated form. All the criteria are to be filled/answered, any unanswered criterion will prompt to fill the details correctly in the corresponding criterion. First Name and Last Name are given a minimum length and no numbers or special characters will be accepted. Either of the gender checkboxes is to be checked. Under the dropdown “Select Your User Type” the user has to select if he will be serving as a Project Manager(Admin) or a Project Member. Then the user will have to give a correct email, which is validated to accept data in general email format. Finally, the Password is to be at least 8 characters with at least one uppercase letter, one lowercase letter, one number and one special character.

Upon clicking the Create User button, the user will be redirected to the Login Page, the user will, then, have to log into their account with the correct username and password.

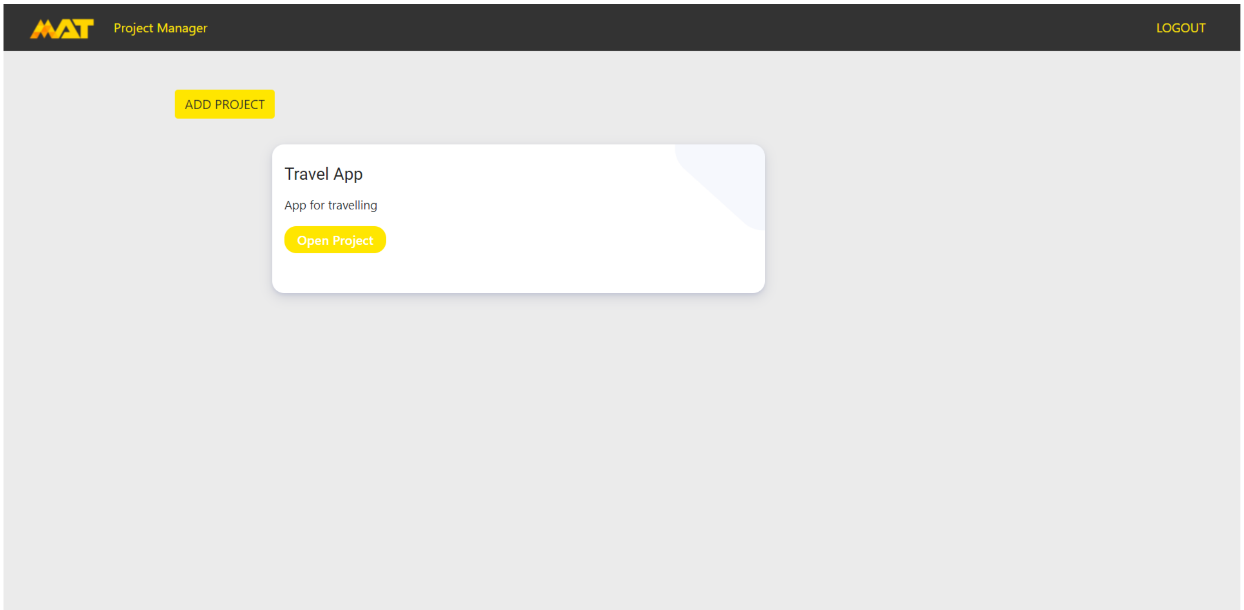


Image 1.4 - Project Home Page

This page shows up when the logged-in user is a Project Manager(Admin). The Project Manager(Admin) can Open the created projects or even make new project boxes.

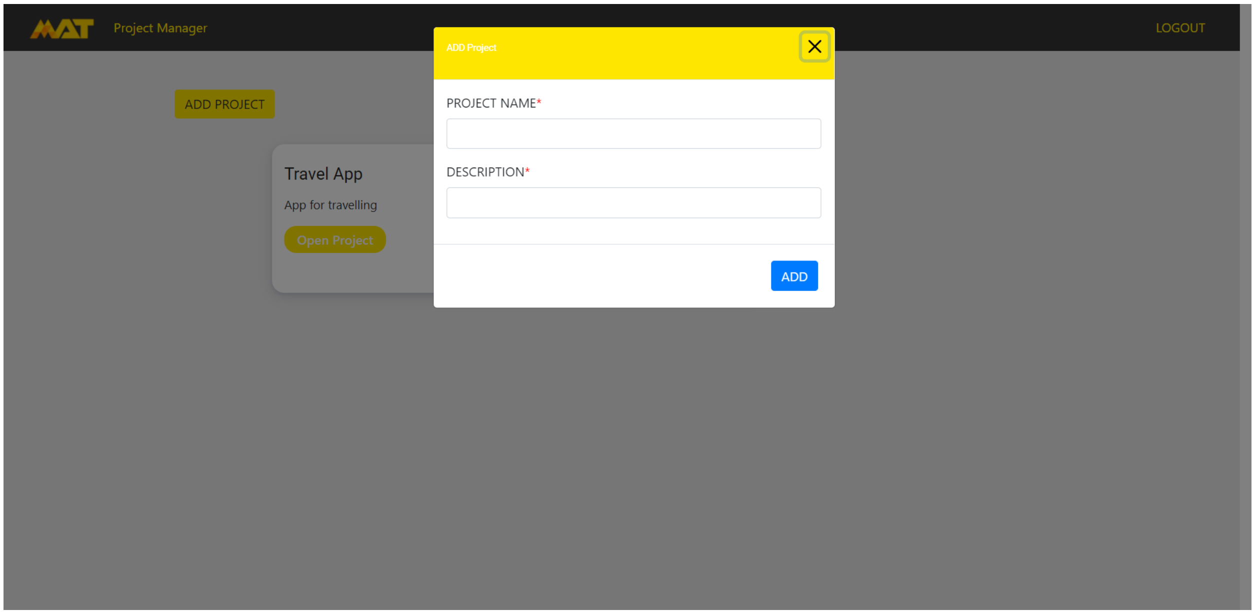
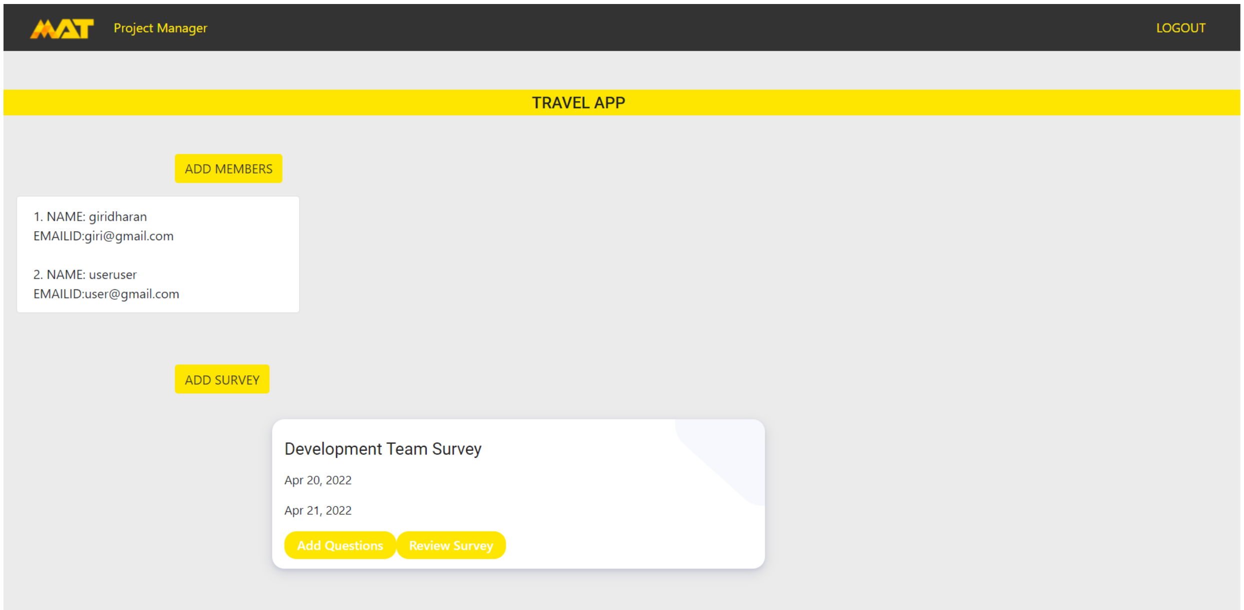


Image 1.5 - Add Project Popup

This Pop-up shows up when the Project Manager(Admin) clicks on “Add Project”. Here the Project Manager(Admin) can Add the project name and a small description.



1. NAME: giridharan
EMAILID:giri@gmail.com

2. NAME: useruser
EMAILID:user@gmail.com

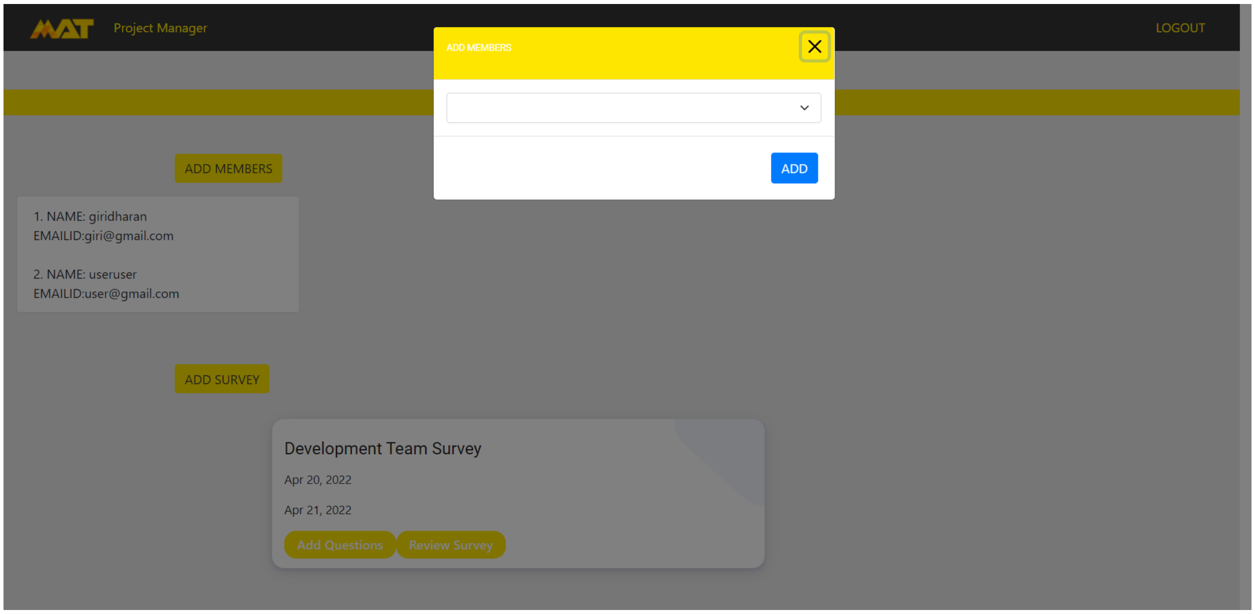
Development Team Survey

Apr 20, 2022

Apr 21, 2022

Image 1.6 - Add Survey and Add Members Page

This is the Project Page, this shows up when the Project Manager(Admin) opens a project. The Project Manager(Admin) can add project members by clicking on “Add Members” and can add surveys by clicking the “Add Survey” button.



1. NAME: giridharan
EMAILID:giri@gmail.com

2. NAME: useruser
EMAILID:user@gmail.com

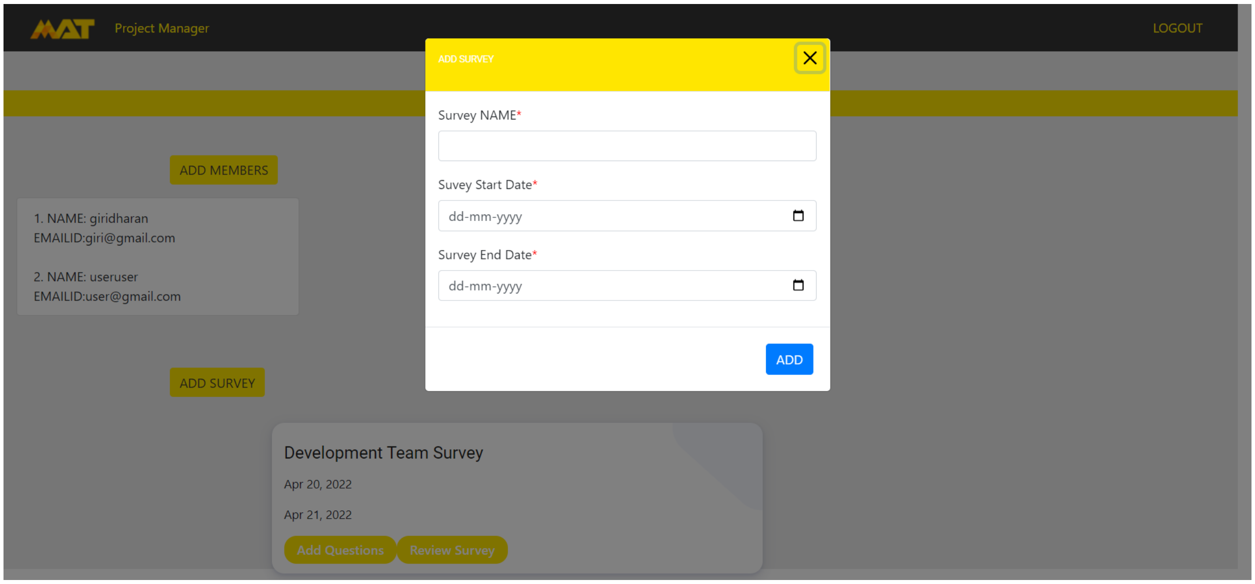
Development Team Survey

Apr 20, 2022

Apr 21, 2022

Image 1.7 - Add Members Popup

This Pop-up shows up when the Project Manager(Admin) clicks on “Add Members”. The Project Manager(Admin) can use the dropdown to choose the members for the project.



Survey NAME*

Suvey Start Date*

dd-mm-yyyy

Survey End Date*

dd-mm-yyyy

1. NAME: giridharan
EMAILID:giri@gmail.com

2. NAME: useruser
EMAILID:user@gmail.com

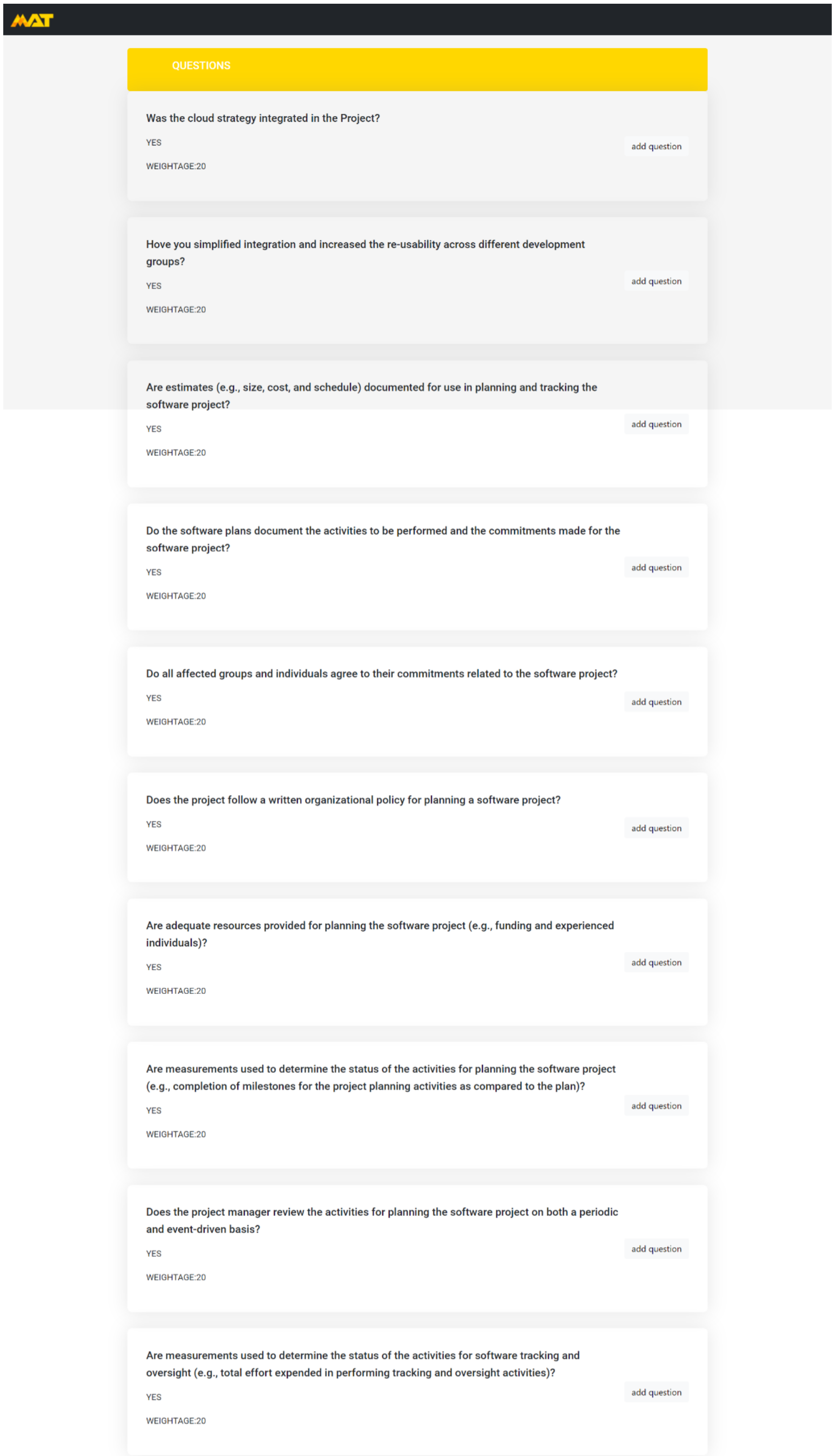
Development Team Survey

Apr 20, 2022

Apr 21, 2022

Image 1.8 - Add Survey Popup

This Pop-up when the Project Manager(Admin) clicks on the “Add Survey” button. The Project Manager(Admin) can add a survey name and set a survey start and end date.



QUESTIONS

Was the cloud strategy integrated in the Project?

YES

WEIGHTAGE:20

add question

Hove you simplified integration and increased the re-usability across different development groups?

YES

WEIGHTAGE:20

add question

Are estimates (e.g., size, cost, and schedule) documented for use in planning and tracking the software project?

YES

WEIGHTAGE:20

add question

Do the software plans document the activities to be performed and the commitments made for the software project?

YES

WEIGHTAGE:20

add question

Do all affected groups and individuals agree to their commitments related to the software project?

YES

WEIGHTAGE:20

add question

Does the project follow a written organizational policy for planning a software project?

YES

WEIGHTAGE:20

add question

Are adequate resources provided for planning the software project (e.g., funding and experienced individuals)?

YES

WEIGHTAGE:20

add question

Are measurements used to determine the status of the activities for planning the software project (e.g., completion of milestones for the project planning activities as compared to the plan)?

YES

WEIGHTAGE:20

add question

Does the project manager review the activities for planning the software project on both a periodic and event-driven basis?

YES

WEIGHTAGE:20

add question

Are measurements used to determine the status of the activities for software tracking and oversight (e.g., total effort expended in performing tracking and oversight activities)?

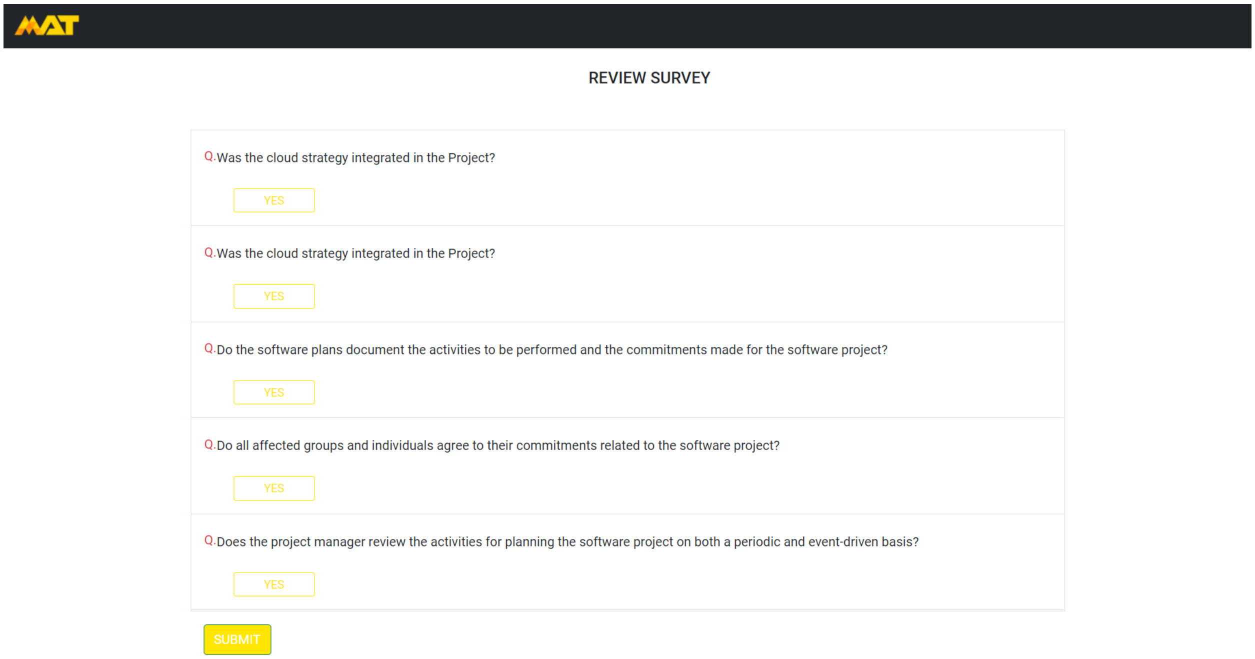
YES

WEIGHTAGE:20

add question

Image 1.9 - Add Questions Page

The Project Manager(Admin) can add questions by choosing out of these questions



REVIEW SURVEY

Q.Was the cloud strategy integrated in the Project?

Q.Was the cloud strategy integrated in the Project?

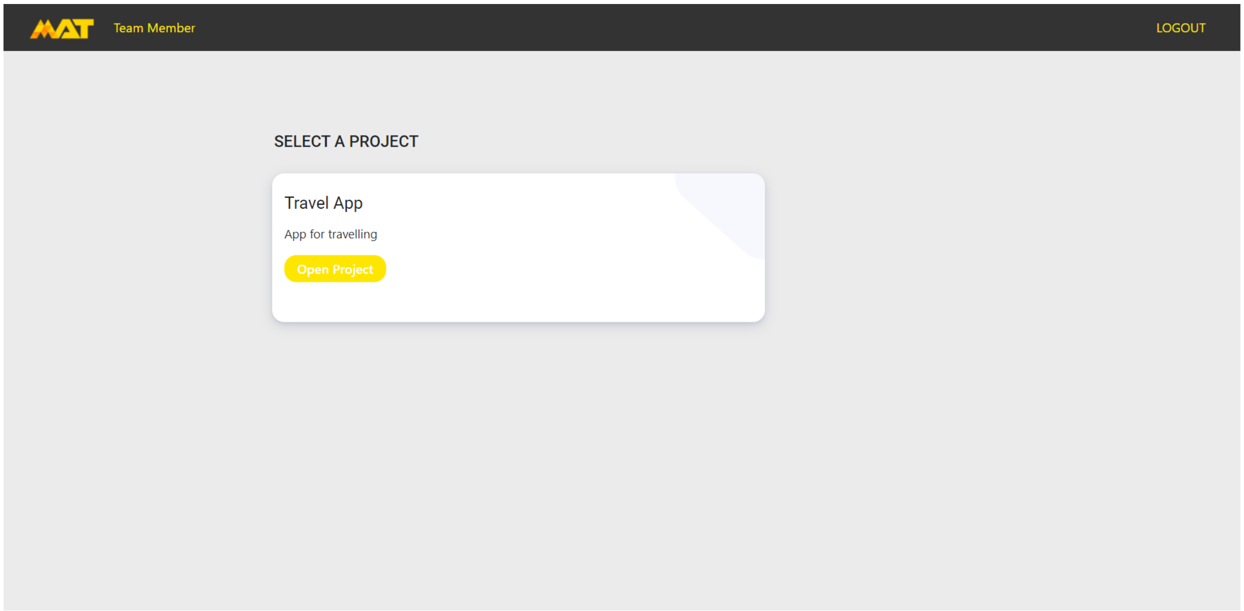
Q.Do the software plans document the activities to be performed and the commitments made for the software project?

Q.Do all affected groups and individuals agree to their commitments related to the software project?

Q.Does the project manager review the activities for planning the software project on both a periodic and event-driven basis?

Image 1.10 - Review Survey Page

This page shows up when the Project Manager(Admin) clicks on the “Review Survey” button on the Project Page. Here the Project Manager(Admin) will see what the Team Members will see once they get to answer the survey questions.



SELECT A PROJECT

Travel App

App for travelling

Image 1.11 - Select Project Page

This page shows up when the logged-in user is a Project Member. The Project Member can select the projects assigned to them by their Project Manager. The Project Member can click on the “Open Project” button to see and attend the surveys.

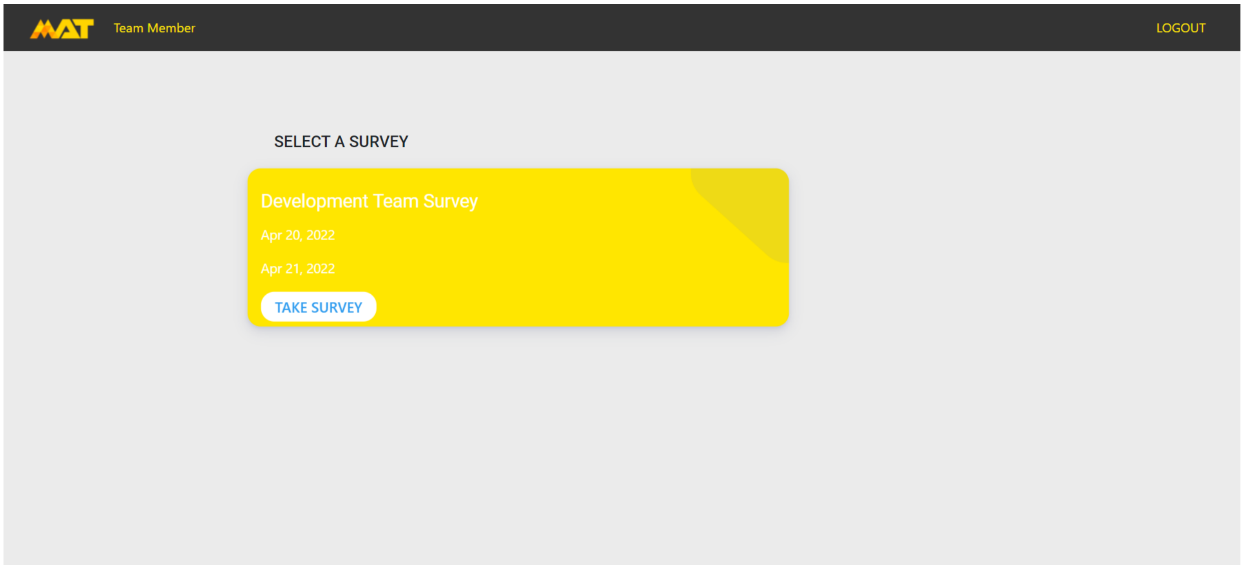
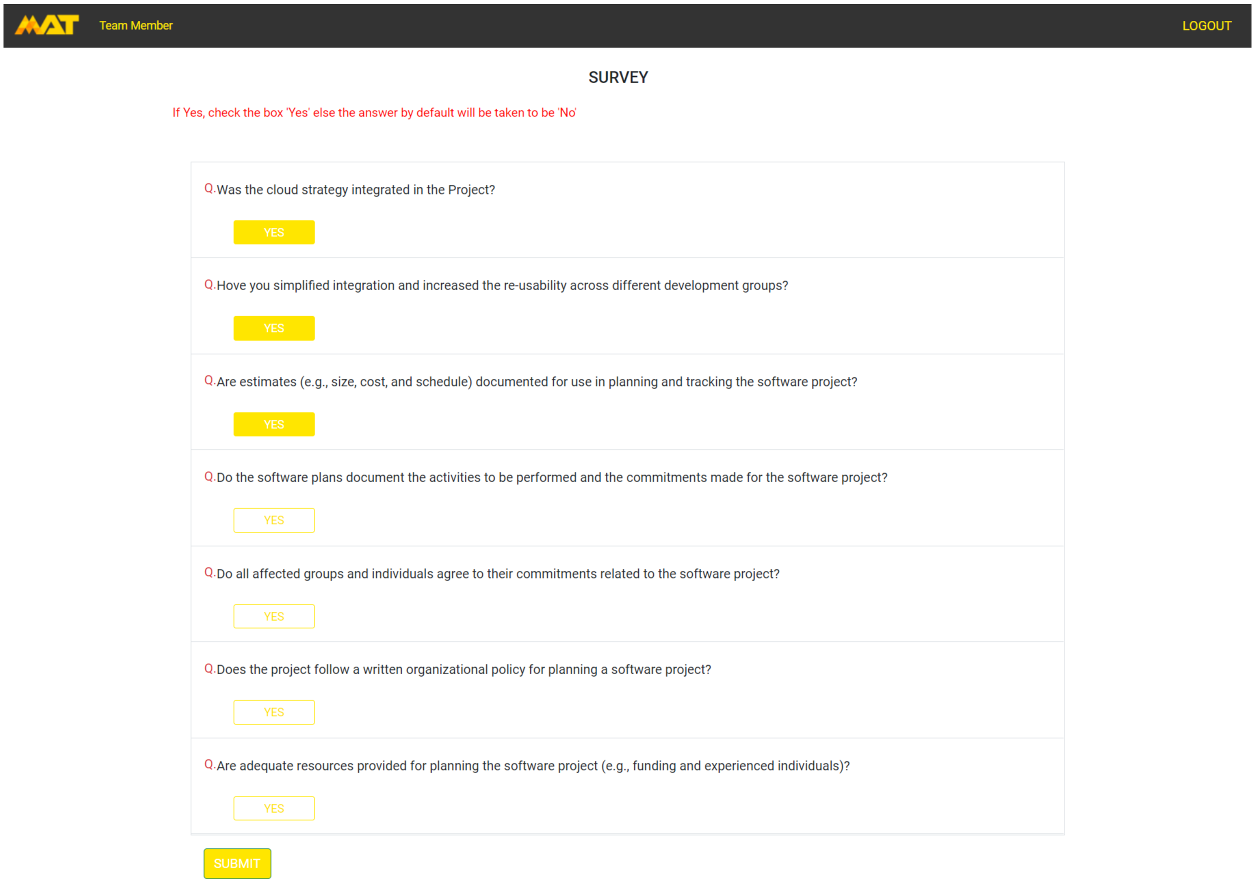


Image 1.12 - Select Survey Page

This page shows up when the Project Member clicks on “Open Project”. Here the user gets to see the surveys that are added to this particular project. The Project Member can click on the “Take Survey” button to enter the survey and answer the questions. The surveys have a start date and end date, set by the Project Manager, the Project Member will have to attend the survey in the given timeframe.



SURVEY

If Yes, check the box 'Yes' else the answer by default will be taken to be 'No'

Q.Was the cloud strategy integrated in the Project?

Q.Hove you simplified integration and increased the re-usability across different development groups?

Q.Are estimates (e.g., size, cost, and schedule) documented for use in planning and tracking the software project?

Q.Do the software plans document the activities to be performed and the commitments made for the software project?

Q.Do all affected groups and individuals agree to their commitments related to the software project?

Q.Does the project follow a written organizational policy for planning a software project?

Q.Are adequate resources provided for planning the software project (e.g., funding and experienced individuals)?

Image 1.13 - Questions Page

This is the Survey Questions Page, the Project Member will have to click Yes if the requisite asked in the question matches with their project plan. If not, then the Project Member has to leave the question unanswered, that particular answer will automatically be taken as “No”.

Upon answering all the questions, the Project Member has to click on the “Submit” button and will be redirected to the Report Page.

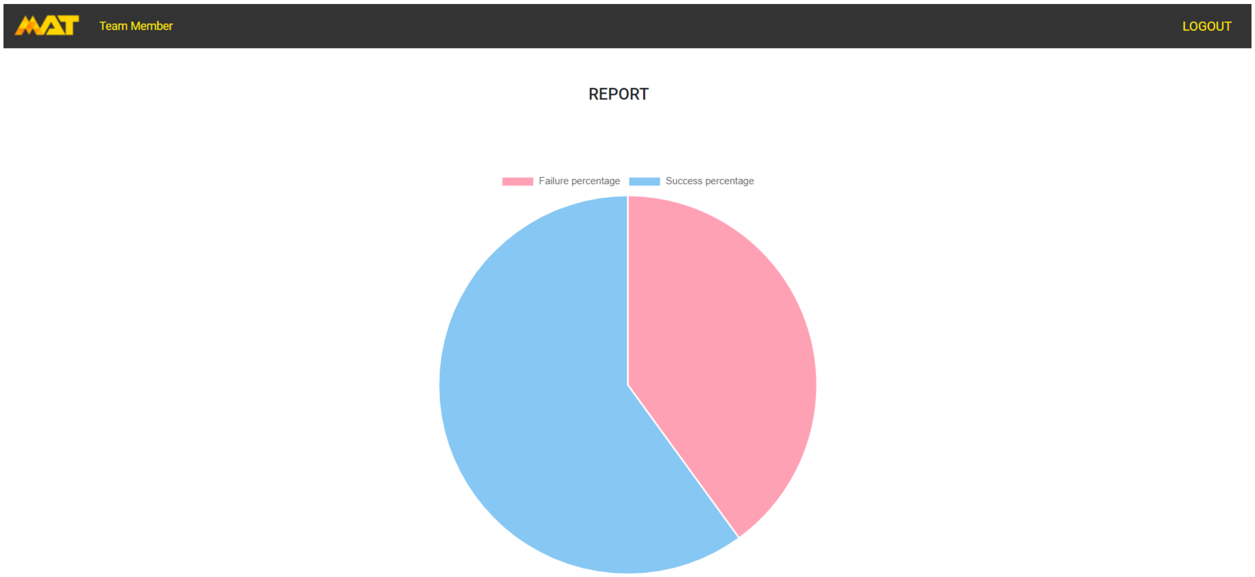
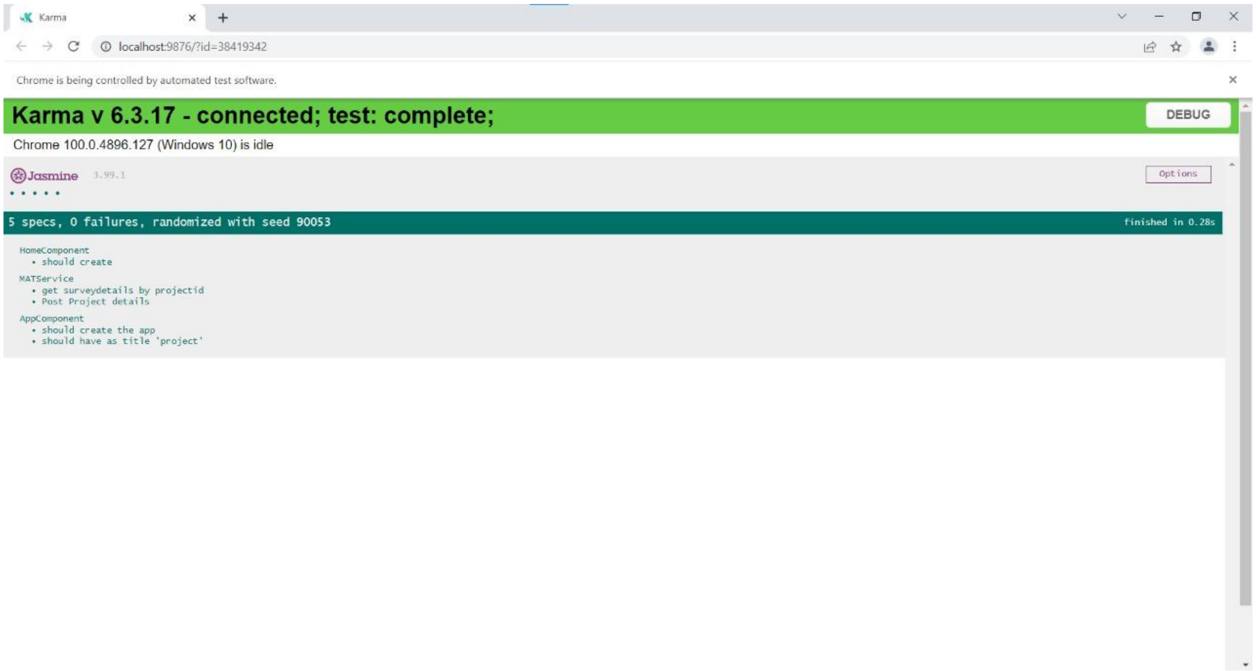


Image 1.14 - Report Page

This is the Report Page. This page shows up when the Project Member clicks on the “Submit” button on the Questions Page. This will show the percentage of maturity in a Pie Chart Format. The portion in red will indicate the percentage that is incorrect and the portion in blue will indicate the percentage which is acceptable for the project survey.



Karma
localhost:9876/?id=38419342
Chrome is being controlled by automated test software.
DEBUG
Jasmine 3.99.1
Options

```
Karma v 6.3.17 - connected; test: complete;
Chrome 100.0.4896.127 (Windows 10) is idle

5 specs, 0 failures, randomized with seed 90053                finished in 0.28s

HomeComponent
  • should create
MATService
  • get surveydetails by projectid
  • Post Project details
AppComponent
  • should create the app
  • should have as title 'project'
```

Image 1.15 - Unit Testing

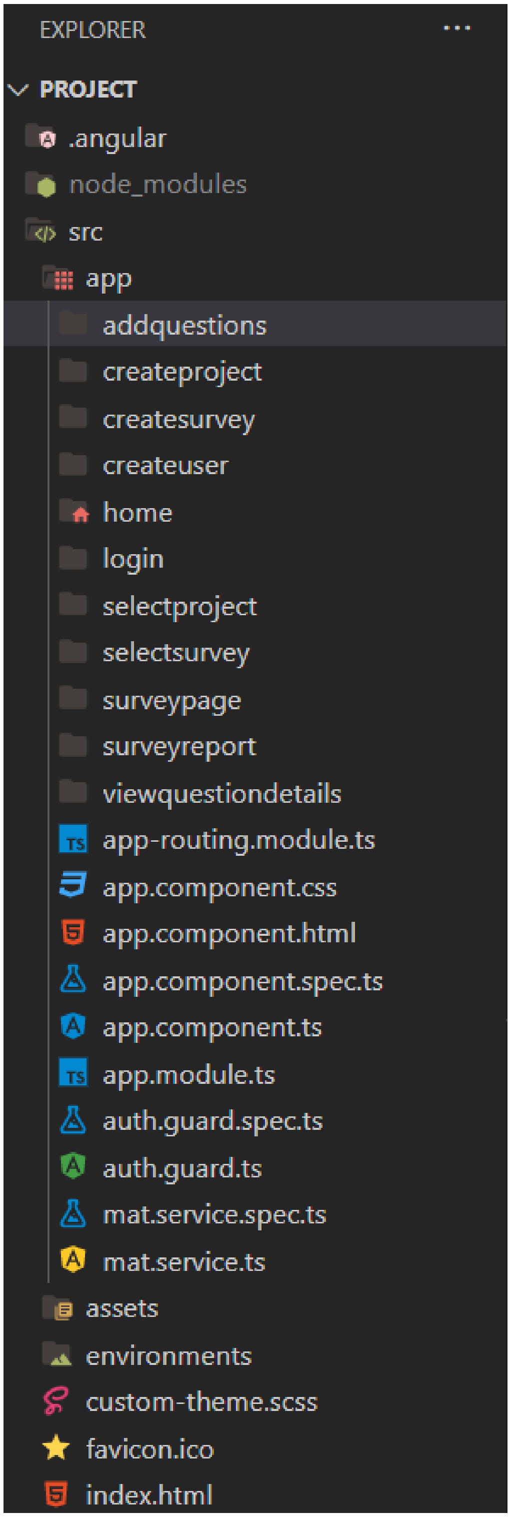
This is the screenshot after checking and testing the unit terms of the project. Formally, Unit Testing is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. This testing methodology is done during the development process by the software developers and sometimes QA staff.

GITHUB REPOSITORY LINK

https://github.com/Eminem-ant/MaturityAssessmentToolRepo

- Download the Project from GitHub
- Select the file in the Angular Folder
- Open the project in Visual Studio Code and Run the commands:
    - ★ ***npm install*** **and** ***ng serve --open***
- Run the SQL commands present in the .txt file in the Database folder
- Fill up the required database entries
- Open the Project_MAT in Visual Studio
- Change the connection string to the newly created database's connection string.
- Run the following command in the NuGet package manager console:
    - ★ **Scaffold-DbContext "*(new connection string)*" Microsoft.EntityFrameworkCore.Sqlserver -outputDir Models -Force**

ANGULAR PART PART



EXPLORER ···

PROJECT
- .angular
- node_modules
- src
  - app
    - addquestions
    - createproject
    - createsurvey
    - createuser
    - home
    - login
    - selectproject
    - selectsurvey
    - surveypage
    - surveyreport
    - viewquestiondetails
    - app-routing.module.ts
    - app.component.css
    - app.component.html
    - app.component.spec.ts
    - app.component.ts
    - app.module.ts
    - auth.guard.spec.ts
    - auth.guard.ts
    - mat.service.spec.ts
    - mat.service.ts
  - assets
  - environments
  - custom-theme.scss
  - favicon.ico
  - index.html

Add questions

Component.css

```css
body{
  background:#f5f5f5;
  }

/* ===== Career ===== */
.career-form {
background-color: #4e63d7;
border-radius: 5px;
padding: 0 16px;
}

.career-form .form-control {
background-color: rgba(255, 255, 255, 0.2);
border: 0;
padding: 12px 15px;
color: #fff;
}

.career-form .form-control::-webkit-input-placeholder {
/* Chrome/Opera/Safari */
color: #fff;
}

.career-form .form-control::-moz-placeholder {
/* Firefox 19+ */
color: #fff;
}

.career-form .form-control:-ms-input-placeholder {
/* IE 10+ */
color: #fff;
}

.career-form .form-control:-moz-placeholder {
/* Firefox 18- */
```

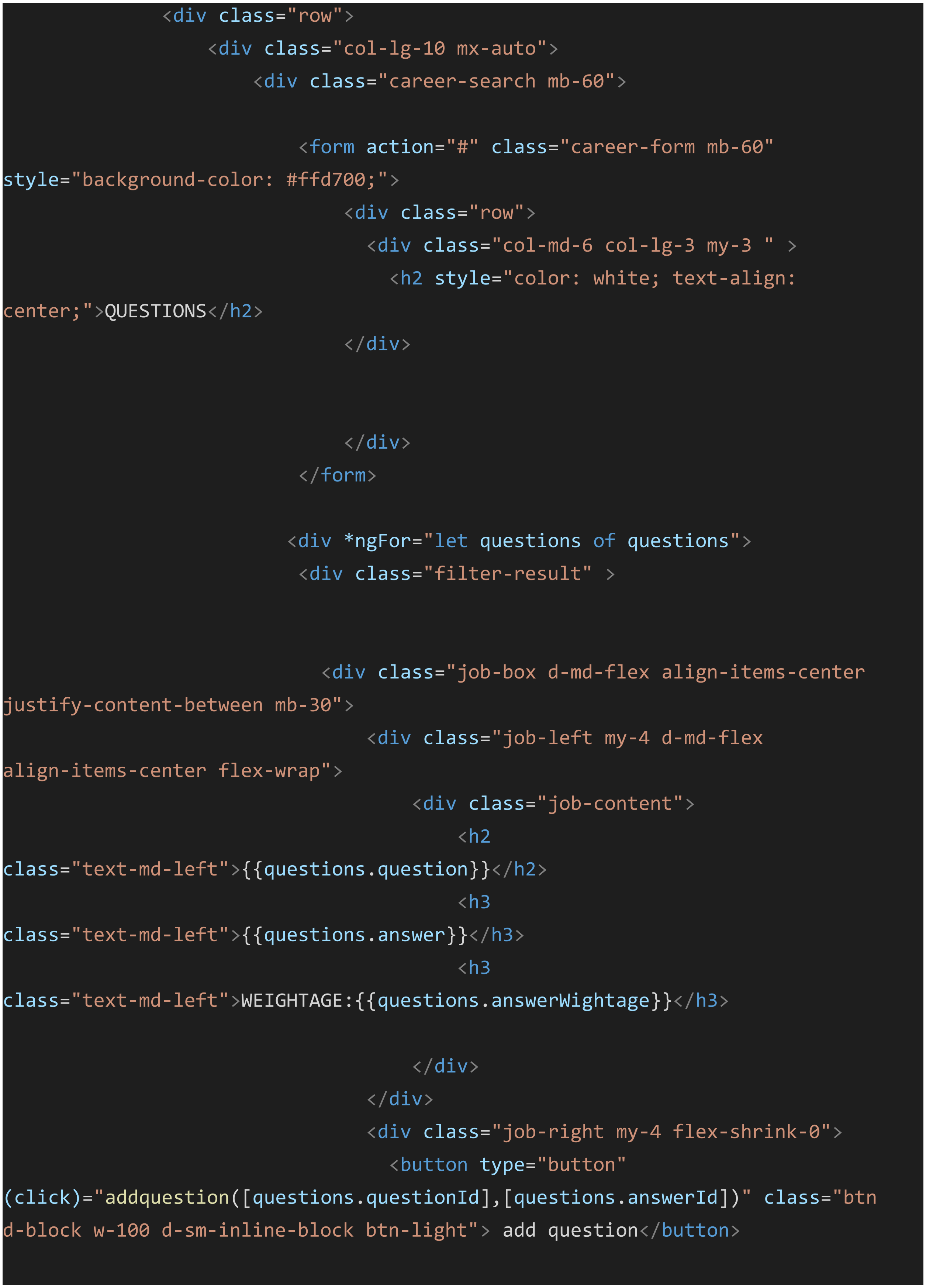
```
color: #fff;
}

.career-form .custom-select {
background-color: rgba(255, 255, 255, 0.2);
border: 0;
padding: 12px 15px;
color: #fff;
width: 100%;
border-radius: 5px;
text-align: left;
height: auto;
background-image: none;
}

.career-form .custom-select:focus {
-webkit-box-shadow: none;
        box-shadow: none;
}

.career-form .select-container {
position: relative;
}

.career-form .select-container:before {
position: absolute;
right: 15px;
top: calc(50% - 14px);
font-size: 18px;
color: #ffffff;
content: '\F2F9';
font-family: "Material-Design-Iconic-Font";
}

.filter-result .job-box {
-webkit-box-shadow: 0 0 35px 0 rgba(130, 130, 130, 0.2);
        box-shadow: 0 0 35px 0 rgba(130, 130, 130, 0.2);
border-radius: 8px;
padding: 10px 35px;
```

```css
}

ul {
list-style: none;
}

.list-disk li {
list-style: none;
margin-bottom: 12px;
}

.list-disk li:last-child {
margin-bottom: 0;
}

.job-box .img-holder {
height: 65px;
width: 65px;
background-color: #4e63d7;
background-image: -webkit-gradient(linear, left top, right top, from(rgba(78, 99, 215, 0.9)), to(#5a85dd));
background-image: linear-gradient(to right, rgba(78, 99, 215, 0.9) 0%, #5a85dd 100%);
font-family: "Open Sans", sans-serif;
color: #fff;
font-size: 22px;
font-weight: 700;
display: -webkit-box;
display: -ms-flexbox;
display: flex;
-webkit-box-pack: center;
    -ms-flex-pack: center;
        justify-content: center;
-webkit-box-align: center;
    -ms-flex-align: center;
        align-items: center;
border-radius: 65px;
}
```

```
.career-title {
background-color: #4e63d7;
color: #fff;
padding: 15px;
text-align: center;
border-radius: 10px 10px 0 0;
background-image: -webkit-gradient(linear, left top, right top, from(rgba(78, 99,
215, 0.9)), to(#5a85dd));
background-image: linear-gradient(to right, rgba(78, 99, 215, 0.9) 0%, #5a85dd
100%);
}

.job-overview {
-webkit-box-shadow: 0 0 35px 0 rgba(130, 130, 130, 0.2);
        box-shadow: 0 0 35px 0 rgba(130, 130, 130, 0.2);
border-radius: 10px;
}

@media (min-width: 992px) {
.job-overview {
  position: -webkit-sticky;
  position: sticky;
  top: 70px;
}
}

.job-overview .job-detail ul {
margin-bottom: 28px;
}

.job-overview .job-detail ul li {
opacity: 0.75;
font-weight: 600;
margin-bottom: 15px;
}

.job-overview .job-detail ul li i {
font-size: 20px;
position: relative;
top: 1px;
}

.job-overview .overview-bottom,
.job-overview .overview-top {
padding: 35px;
}

.job-content ul li {
font-weight: 600;
opacity: 0.75;
border-bottom: 1px solid #ccc;
padding: 10px 5px;
}

@media (min-width: 768px) {
.job-content ul li {
  border-bottom: 0;
  padding: 0;
}
}

.job-content ul li i {
font-size: 20px;
position: relative;
top: 1px;
}
.mb-30 {
  margin-bottom: 30px;
}
```

Component.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>

</head>
<body>
  <nav class="navbar navbar-expand-lg  bg-dark" style="background-color:
#333333;">
    <div class="container-fluid">
        <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
     </div>
  </nav>
<br>
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/material-design-iconic-font/2.2.0/css/material-design-iconic-font.min.css"
integrity="sha256-3sPp8BkKUE7QyPSl6VfBByBroQbKxKG7tsusY2mhbVY="
crossorigin="anonymous" />

  <div class="container">
```

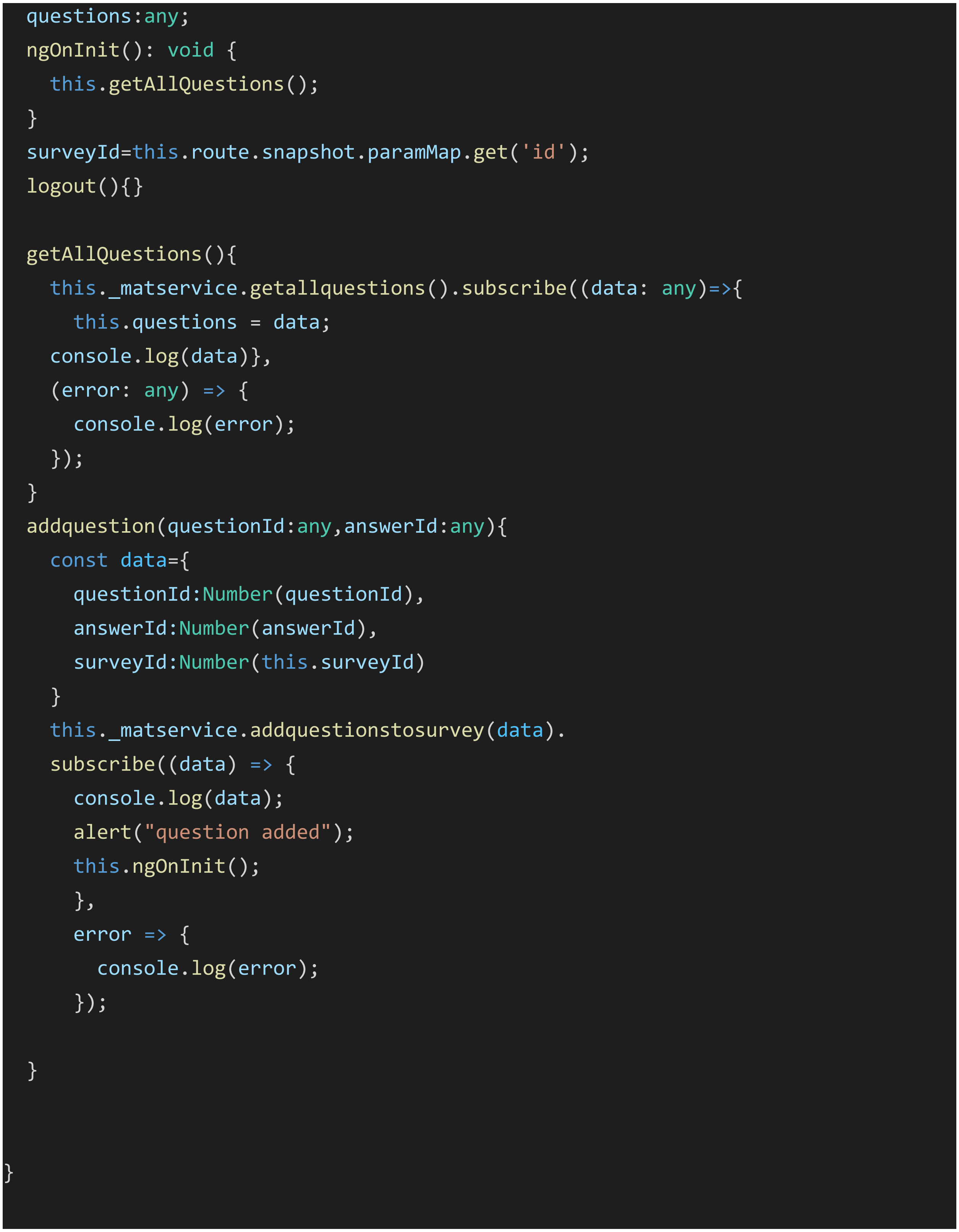


```
              <div class="row">
                  <div class="col-lg-10 mx-auto">
                      <div class="career-search mb-60">

                          <form action="#" class="career-form mb-60"
style="background-color: #ffd700;">
                              <div class="row">
                                <div class="col-md-6 col-lg-3 my-3 " >
                                  <h2 style="color: white; text-align:
center;">QUESTIONS</h2>
                              </div>


                              </div>
                          </form>

                         <div *ngFor="let questions of questions">
                          <div class="filter-result" >


                            <div class="job-box d-md-flex align-items-center
justify-content-between mb-30">
                                <div class="job-left my-4 d-md-flex
align-items-center flex-wrap">
                                    <div class="job-content">
                                        <h2
class="text-md-left">{{questions.question}}</h2>
                                        <h3
class="text-md-left">{{questions.answer}}</h3>
                                        <h3
class="text-md-left">WEIGHTAGE:{{questions.answerWightage}}</h3>

                                    </div>
                                </div>
                                <div class="job-right my-4 flex-shrink-0">
                                  <button type="button"
(click)="addquestion([questions.questionId],[questions.answerId])" class="btn
d-block w-100 d-sm-inline-block btn-light"> add question</button>
```

```
                                </div>
                            </div>
                        </div>
                         </div>


                      </div>



                  </div>
              </div>

          </div>
          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.
min.js"
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroaf
W" crossorigin="anonymous"></script>

</body>
</html>
```

Component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { MATService } from '../mat.service';

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

  constructor(private _matservice:MATService , private route:
ActivatedRoute,private router: Router,private jwtHelper: JwtHelperService)  {}
```

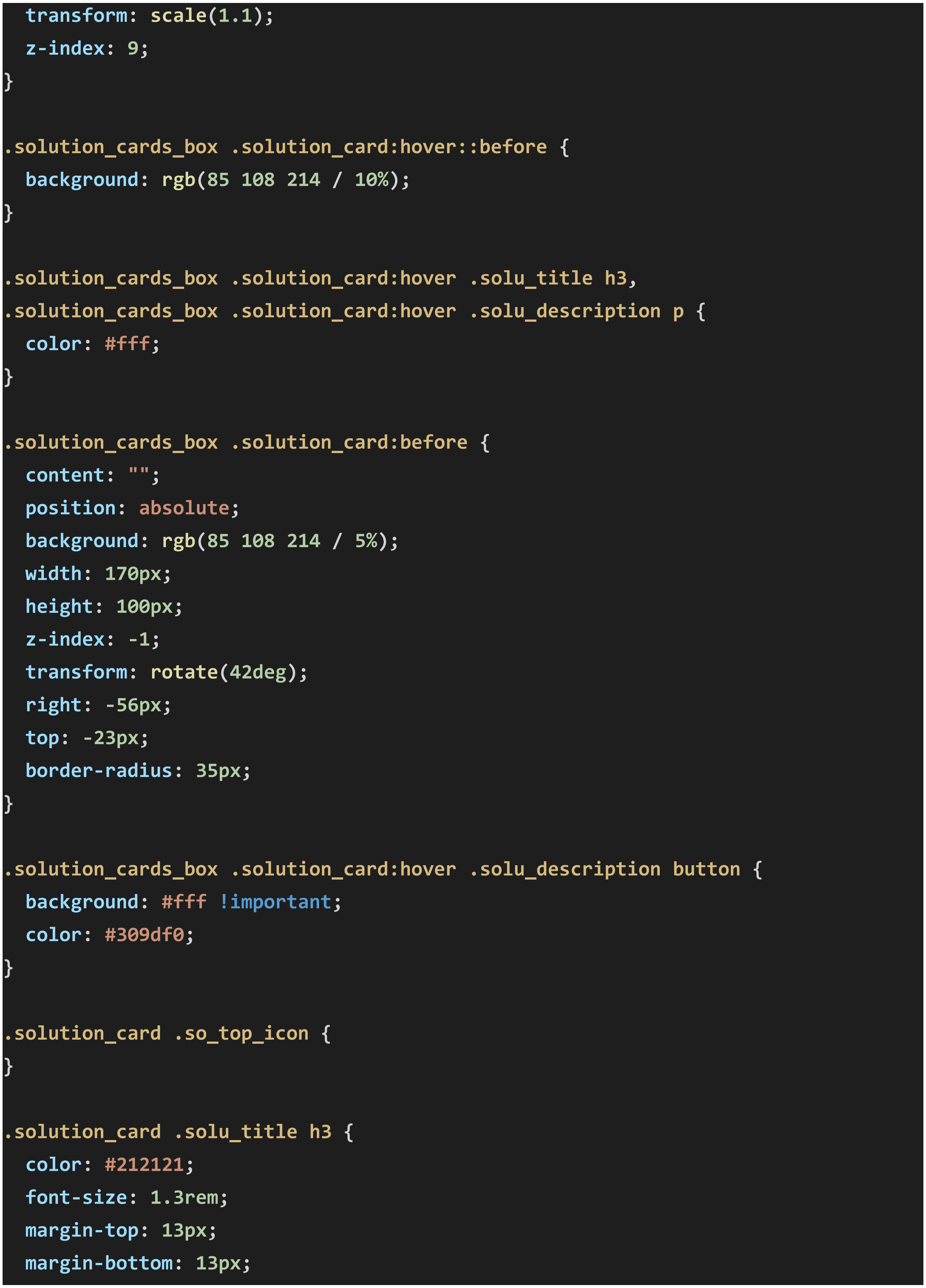


```
  questions:any;
  ngOnInit(): void {
    this.getAllQuestions();
  }
  surveyId=this.route.snapshot.paramMap.get('id');
  logout(){}

  getAllQuestions(){
    this._matservice.getallquestions().subscribe((data: any)=>{
      this.questions = data;
    console.log(data)},
    (error: any) => {
      console.log(error);
    });
  }
  addquestion(questionId:any,answerId:any){
    const data={
      questionId:Number(questionId),
      answerId:Number(answerId),
      surveyId:Number(this.surveyId)
    }
    this._matservice.addquestionstosurvey(data).
    subscribe((data) => {
      console.log(data);
      alert("question added");
      this.ngOnInit();
      },
      error => {
        console.log(error);
      });

  }


}
```

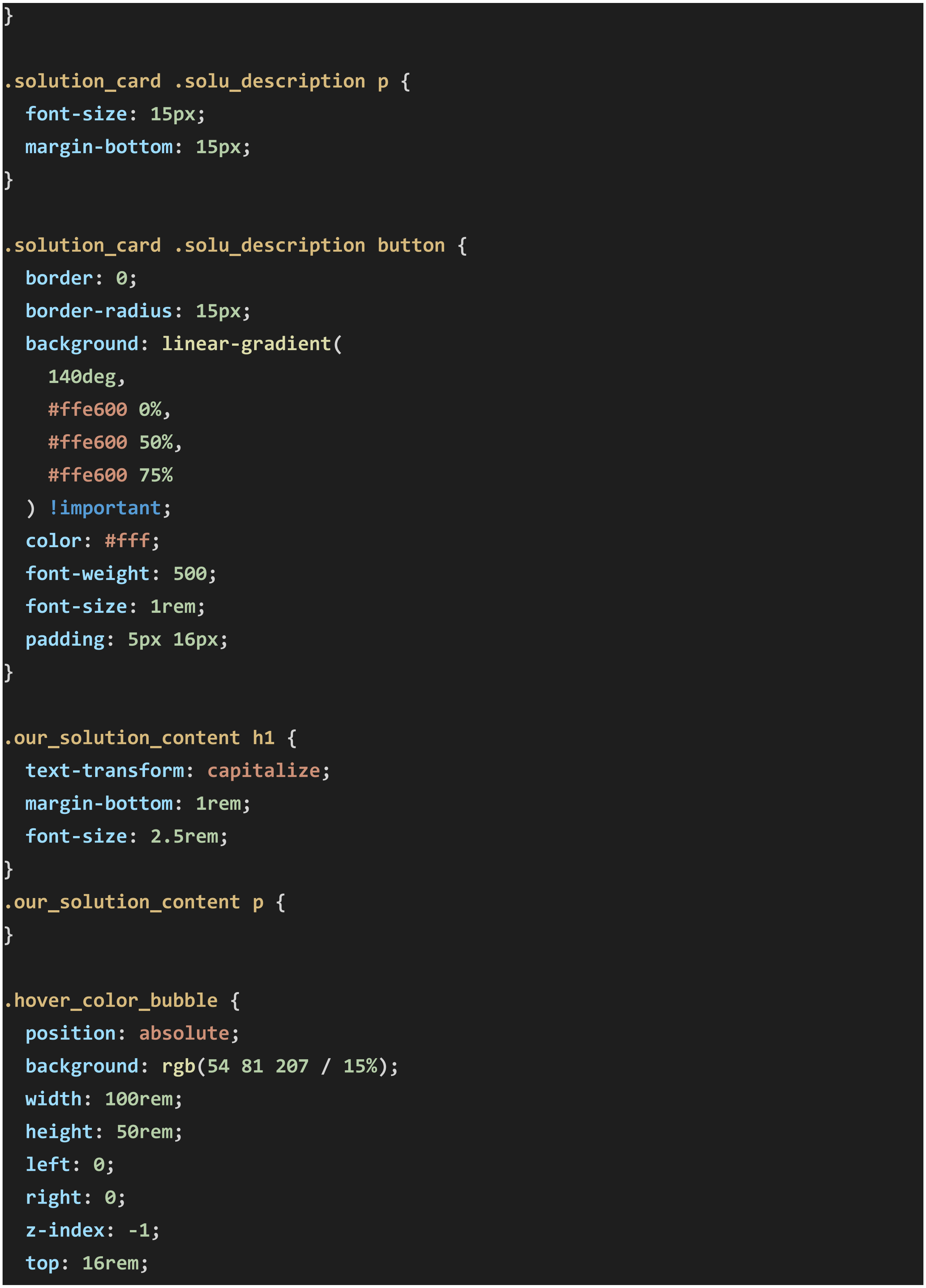
Create Project

Component.css

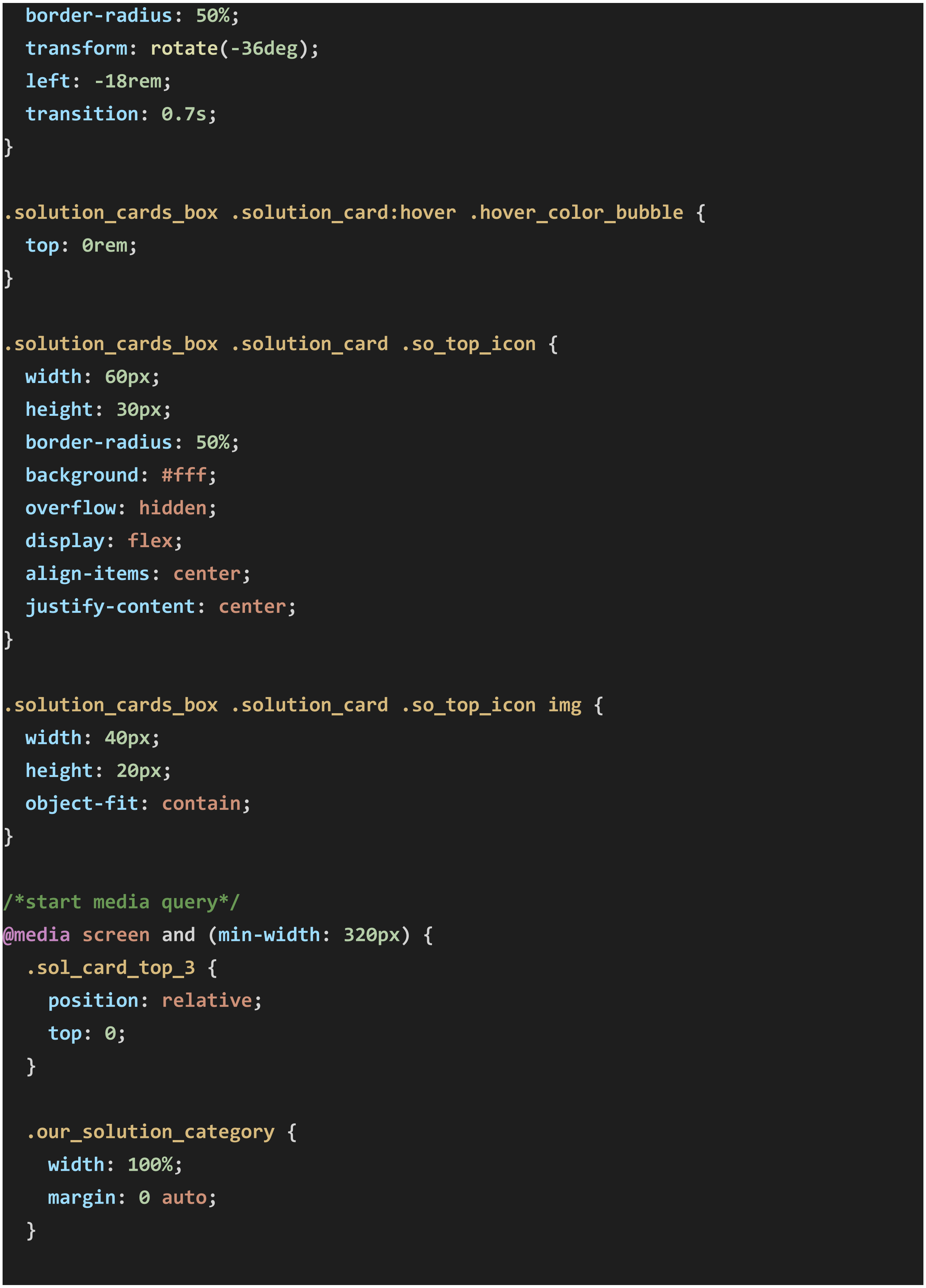
```
body{
  background-color: #ebebeb;
}
.section_our_solution .row {
  align-items: center;
}

.our_solution_category {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
}
.our_solution_category .solution_cards_box {
  display: flex;
  flex-direction: column;
  justify-content: center;
}
.solution_cards_box .solution_card {
  flex: 0 50%;
  background: #fff;
  box-shadow: 0 2px 4px 0 rgba(136, 144, 195, 0.2),
    0 5px 15px 0 rgba(37, 44, 97, 0.15);
  border-radius: 15px;
  margin: 8px;
  padding: 10px 15px;
  position: relative;
  z-index: 1;
  overflow: hidden;
  min-height: 185px;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover {
  background: #ffe600;
  color: #fff;
```



```css
  transform: scale(1.1);
  z-index: 9;
}

.solution_cards_box .solution_card:hover::before {
  background: rgb(85 108 214 / 10%);
}

.solution_cards_box .solution_card:hover .solu_title h3,
.solution_cards_box .solution_card:hover .solu_description p {
  color: #fff;
}

.solution_cards_box .solution_card:before {
  content: "";
  position: absolute;
  background: rgb(85 108 214 / 5%);
  width: 170px;
  height: 100px;
  z-index: -1;
  transform: rotate(42deg);
  right: -56px;
  top: -23px;
  border-radius: 35px;
}

.solution_cards_box .solution_card:hover .solu_description button {
  background: #fff !important;
  color: #309df0;
}

.solution_card .so_top_icon {
}

.solution_card .solu_title h3 {
  color: #212121;
  font-size: 1.3rem;
  margin-top: 13px;
  margin-bottom: 13px;
```



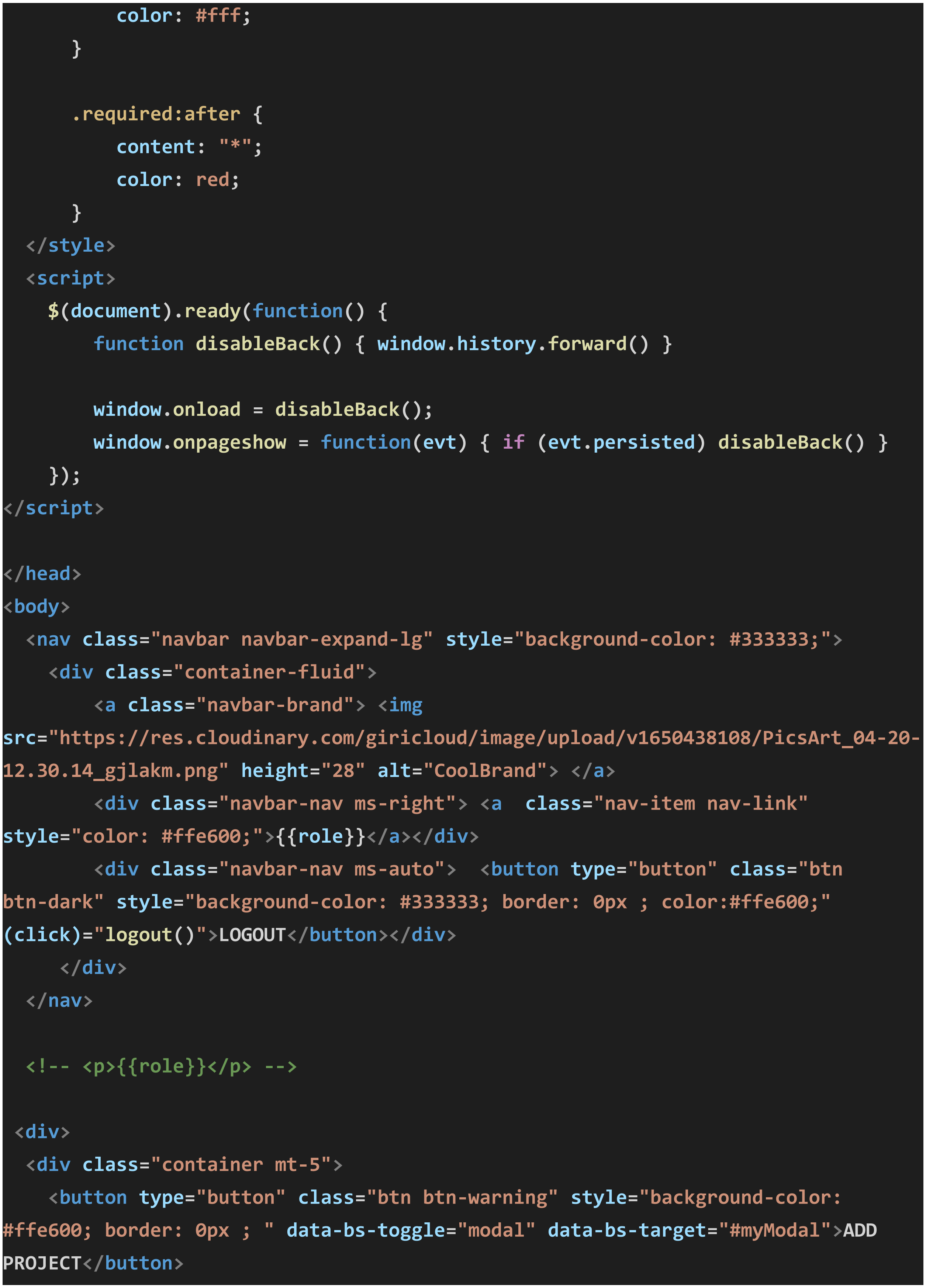
```
}

.solution_card .solu_description p {
  font-size: 15px;
  margin-bottom: 15px;
}

.solution_card .solu_description button {
  border: 0;
  border-radius: 15px;
  background: linear-gradient(
    140deg,
    #ffe600 0%,
    #ffe600 50%,
    #ffe600 75%
  ) !important;
  color: #fff;
  font-weight: 500;
  font-size: 1rem;
  padding: 5px 16px;
}

.our_solution_content h1 {
  text-transform: capitalize;
  margin-bottom: 1rem;
  font-size: 2.5rem;
}
.our_solution_content p {
}

.hover_color_bubble {
  position: absolute;
  background: rgb(54 81 207 / 15%);
  width: 100rem;
  height: 50rem;
  left: 0;
  right: 0;
  z-index: -1;
  top: 16rem;
```



```css
  border-radius: 50%;
  transform: rotate(-36deg);
  left: -18rem;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover .hover_color_bubble {
  top: 0rem;
}

.solution_cards_box .solution_card .so_top_icon {
  width: 60px;
  height: 30px;
  border-radius: 50%;
  background: #fff;
  overflow: hidden;
  display: flex;
  align-items: center;
  justify-content: center;
}

.solution_cards_box .solution_card .so_top_icon img {
  width: 40px;
  height: 20px;
  object-fit: contain;
}

/*start media query*/
@media screen and (min-width: 320px) {
  .sol_card_top_3 {
    position: relative;
    top: 0;
  }

  .our_solution_category {
    width: 100%;
    margin: 0 auto;
  }
  .our_solution_category .solution_cards_box {
    flex: auto;
  }
}
@media only screen and (min-width: 768px) {
  .our_solution_category .solution_cards_box {
    flex: 1;
  }
}
@media only screen and (min-width: 1024px) {
  .sol_card_top_3 {
    position: relative;
    top: -3rem;
  }
  .our_solution_category {
    width: 50%;
    margin: 0 auto;
  }
}
```

Component.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.c
ss" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP
1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
```



```
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>
  <script>
    $(document).ready(function() {
        function disableBack() { window.history.forward() }

        window.onload = disableBack();
        window.onpageshow = function(evt) { if (evt.persisted) disableBack() }
    });
</script>

</head>
<body>
  <nav class="navbar navbar-expand-lg" style="background-color: #333333;">
    <div class="container-fluid">
        <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
        <div class="navbar-nav ms-right"> <a  class="nav-item nav-link"
style="color: #ffe600;">{{role}}</a></div>
        <div class="navbar-nav ms-auto">  <button type="button" class="btn
btn-dark" style="background-color: #333333; border: 0px ; color:#ffe600;"
(click)="logout()">LOGOUT</button></div>
     </div>
  </nav>

  <!-- <p>{{role}}</p> -->

 <div>
  <div class="container mt-5">
    <button type="button" class="btn btn-warning" style="background-color:
#ffe600; border: 0px ; " data-bs-toggle="modal" data-bs-target="#myModal">ADD
PROJECT</button>
```

```
    <div class="modal" id="myModal">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header" style="background-color: #ffe600;">
                    <h5 class="modal-title">ADD Project</h5>
                    <button type="button" class="btn-close"
data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <form>
                        <div class="mb-3">
                            <label class="form-label required">PROJECT
NAME</label>
                            <input type="text" class="form-control"
name="projectname" id="projectname" [(ngModel)]="createproject.projectname" >
                        </div>
                        <div class="mb-3">
                            <label class="form-label
required">DESCRIPTION</label>
                            <input type="email" class="form-control"
name="projectDescription" id="projectDescription"
[(ngModel)]="createproject.projectDescription" >
                        </div>
                    </form>
                </div>
                <div class="modal-footer">
                    <button type="submit" aria-label="Close"
(click)="addproject()" data-mdb-dismiss="modal" class="btn
btn-primary">ADD</button>
                </div>
            </div>
        </div>
    </div>
</div>
 </div>
<br>
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/material-design-iconic-font/2.2.0/cs
s/material-design-iconic-font.min.css"
```

```
integrity="sha256-3sPp8BkKUE7QyPSl6VfBByBroQbKxKG7tsusY2mhbVY="
crossorigin="anonymous" />

  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

  <div class="section_our_solution">
    <div class="row">
      <div *ngFor="let item of projectlists">
      <div class="col-lg-10 col-md-10 col-sm-10">
        <div class="our_solution_category">
          <div class="solution_cards_box">
            <div class="solution_card">

              <div class="solu_title">
                <h3>{{item.projectname}}</h3>
              </div>
              <div class="solu_description">
                <p>
                  {{item.projectDescription}}
                </p>
                <button type="button"
routerLink="/CreatesurveyComponent/{{item.projectId}}"class="read_more_btn">Open
Project</button>
                <!-- <a routerLink="/CreatesurveyComponent/{{item.projectId}}"
class="read_more_btn">Open Project</a> -->
              </div>
            </div>

      </div>
    </div>
  </div>
  </div>
  </div>
  </div>

          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.
min.js"
```

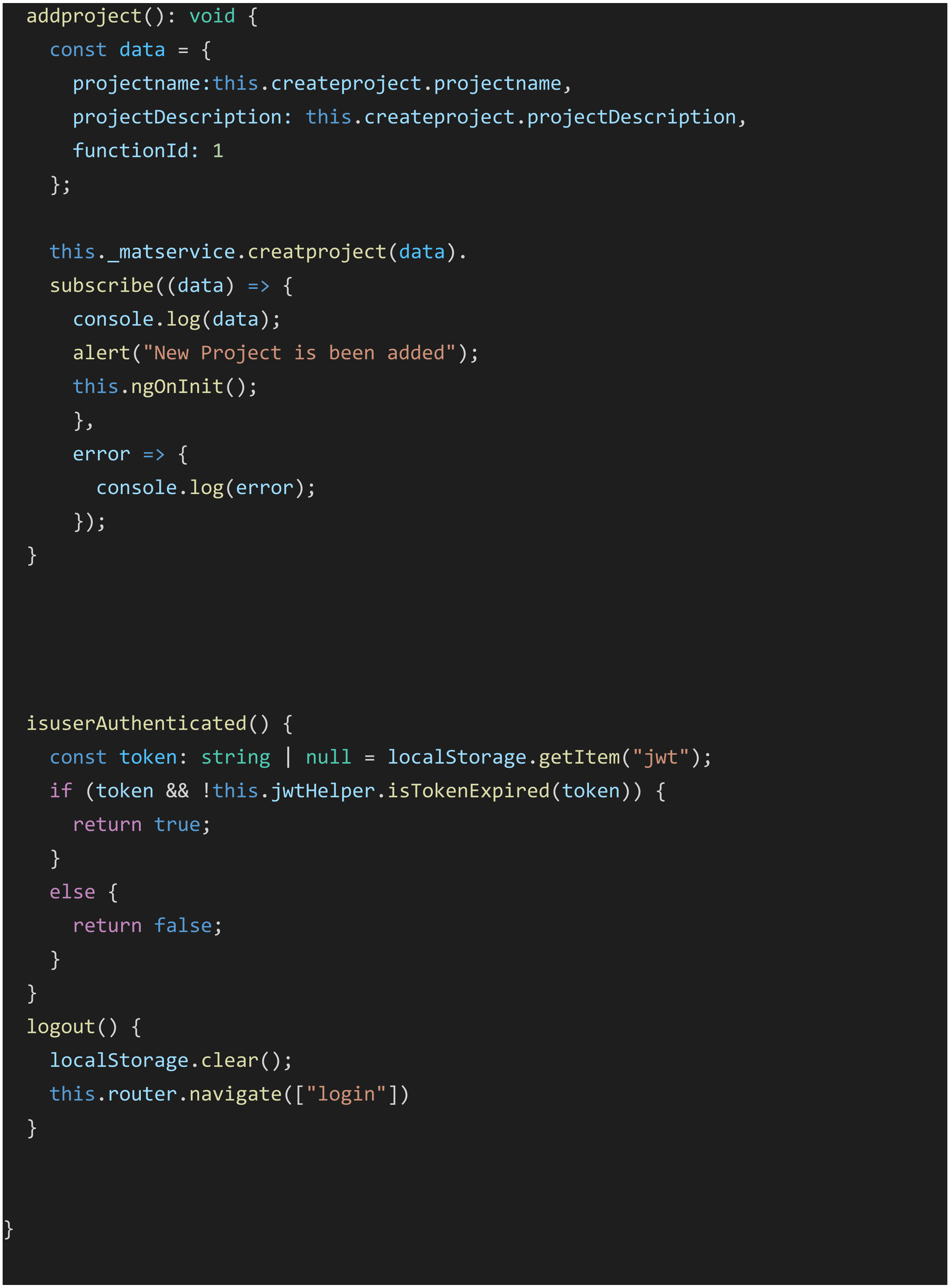
```
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroaf
W" crossorigin="anonymous"></script>


</body>
</html>
```

Component.spec.ts

```
import { ComponentFixture, TestBed } from '@angular/core/testing';

import { CreateprojectComponent } from './createproject.component';
import { projectdetails } from './projectdetails';

// describe('CreateprojectComponent', () => {
//   let component: CreateprojectComponent;
//   let fixture: ComponentFixture<CreateprojectComponent>;

//   beforeEach(async () => {
//     await TestBed.configureTestingModule({
//       declarations: [ CreateprojectComponent ]
//     })
//     .compileComponents();
//   });

//   beforeEach(() => {
//     fixture = TestBed.createComponent(CreateprojectComponent);
//     component = fixture.componentInstance;
//     fixture.detectChanges();
//   });
//   it("testing function", async ()=>{
//     component.projectlist();
//   });



// });
```

Component.ts

```
import { Component, OnInit } from '@angular/core';
import { Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { MATService } from '../mat.service';
import{projectdetails}from './projectdetails'



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

  constructor(private _matservice:MATService ,private router: Router,private
jwtHelper: JwtHelperService)  {}
  role=localStorage.getItem("usertype");
  projectlists:projectdetails[]=[];
  createproject:projectdetails={
    projectId: 0,
    projectname: '',
    projectDescription: '',
    functionId: 1
  };


  ngOnInit(): void {
  this.projectlist();

  }
 projectlist(){
  this._matservice.getAllprojects().subscribe ( (data: any)=>{
    this.projectlists = data;
  console.log(data)})
  }
```

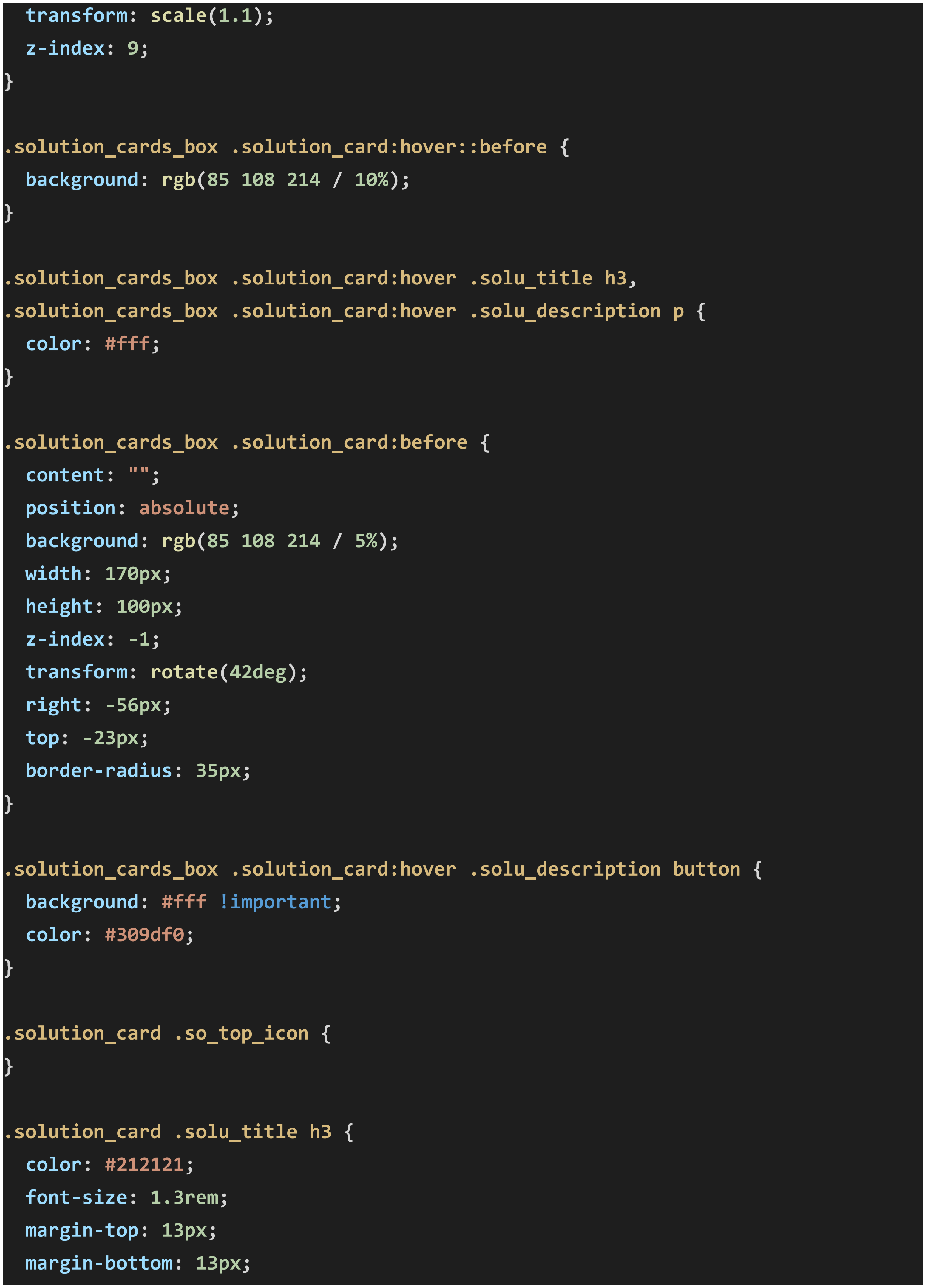


```
  addproject(): void {
    const data = {
      projectname:this.createproject.projectname,
      projectDescription: this.createproject.projectDescription,
      functionId: 1
    };

    this._matservice.creatproject(data).
    subscribe((data) => {
      console.log(data);
      alert("New Project is been added");
      this.ngOnInit();
      },
      error => {
        console.log(error);
      });
  }




  isuserAuthenticated() {
    const token: string | null = localStorage.getItem("jwt");
    if (token && !this.jwtHelper.isTokenExpired(token)) {
      return true;
    }
    else {
      return false;
    }
  }
  logout() {
    localStorage.clear();
    this.router.navigate(["login"])
  }


}
```

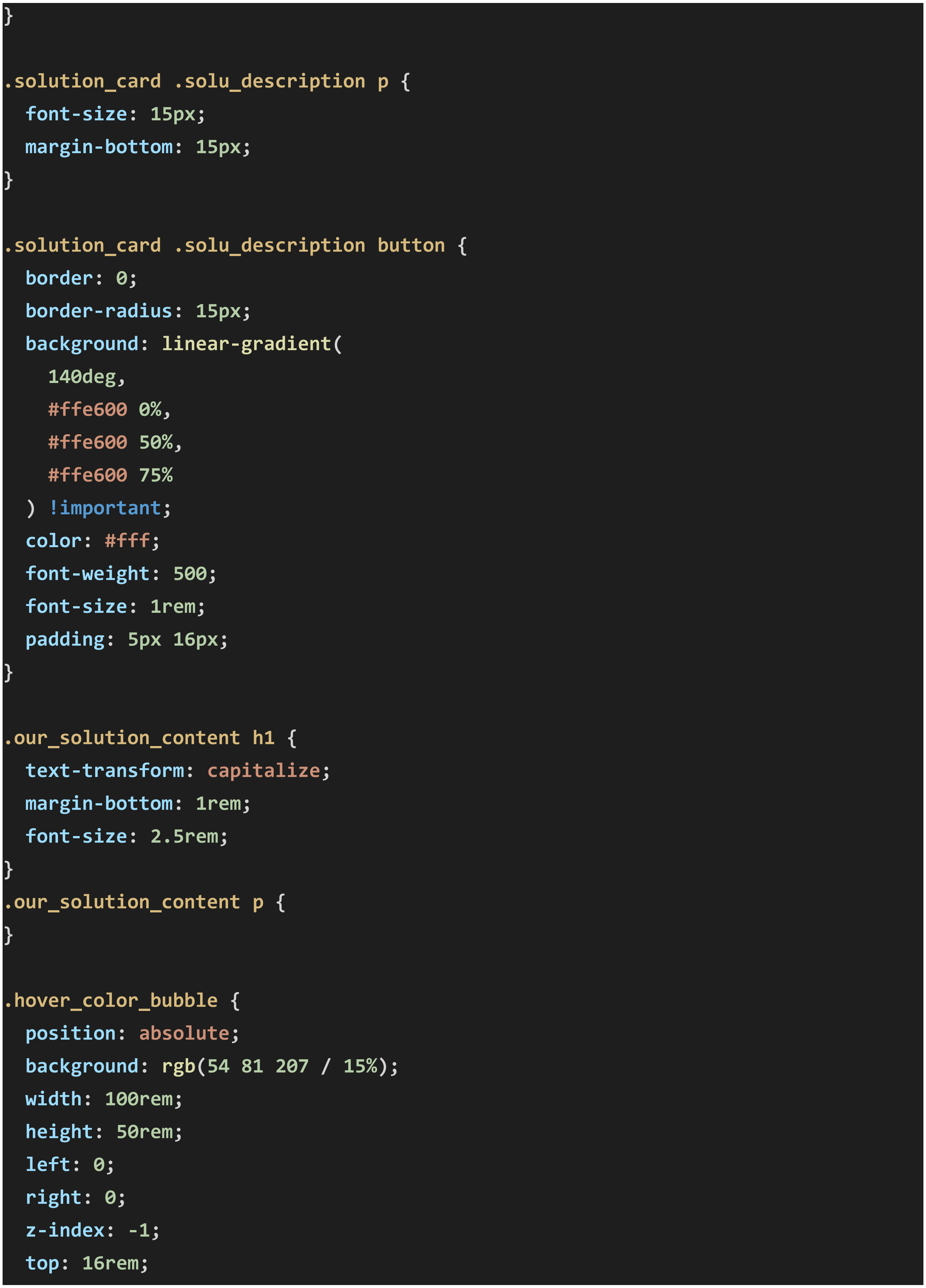
Create Survey

Component.css

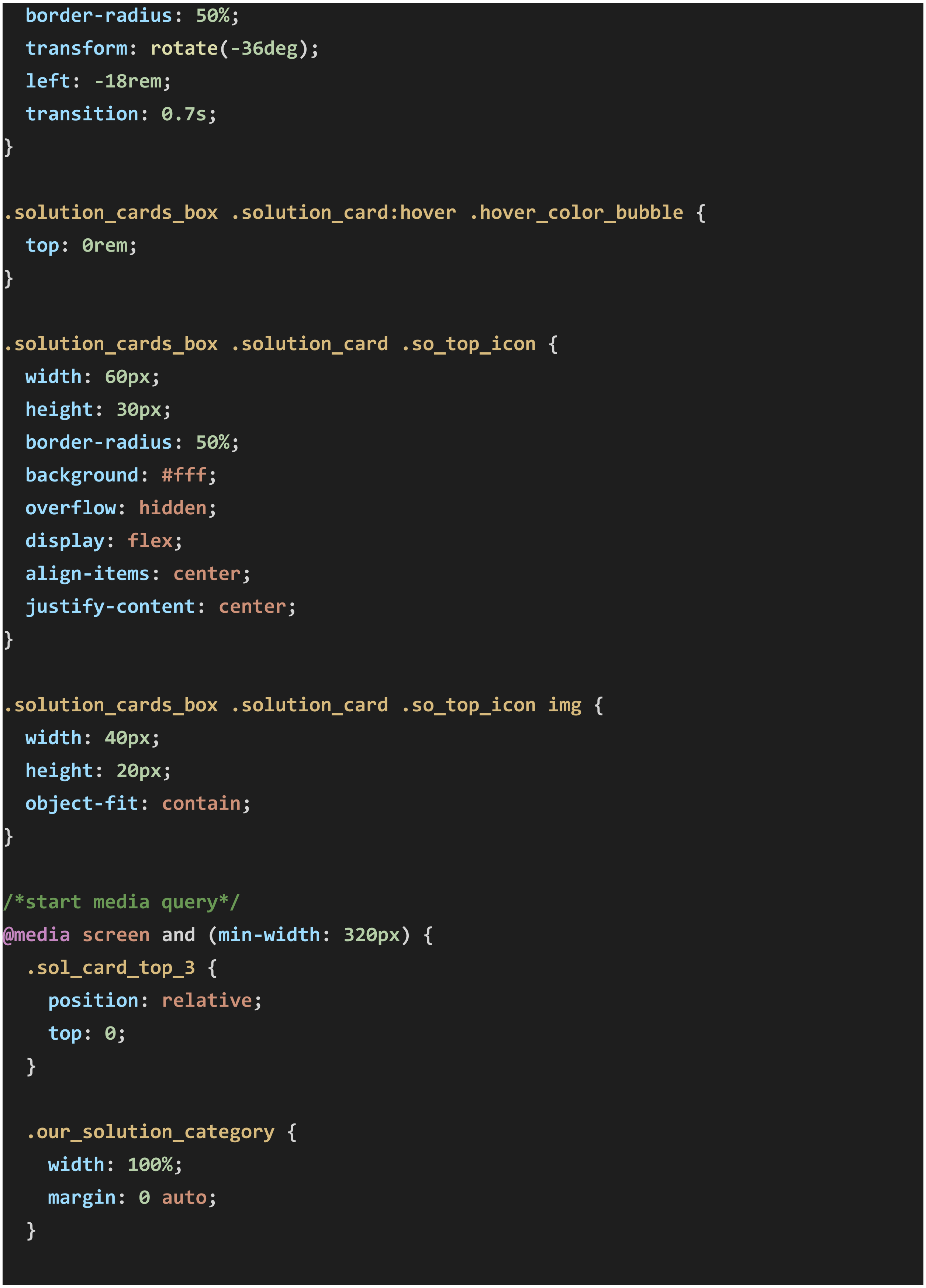
```css
body{
  background-color: #ebebeb;
}
.section_our_solution .row {
  align-items: center;
}

.our_solution_category {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
}
.our_solution_category .solution_cards_box {
  display: flex;
  flex-direction: column;
  justify-content: center;
}
.solution_cards_box .solution_card {
  flex: 0 50%;
  background: #fff;
  box-shadow: 0 2px 4px 0 rgba(136, 144, 195, 0.2),
    0 5px 15px 0 rgba(37, 44, 97, 0.15);
  border-radius: 15px;
  margin: 8px;
  padding: 10px 15px;
  position: relative;
  z-index: 1;
  overflow: hidden;
  min-height: 185px;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover {
  background: #ffe600;
  color: #fff;
```



```
  transform: scale(1.1);
  z-index: 9;
}

.solution_cards_box .solution_card:hover::before {
  background: rgb(85 108 214 / 10%);
}

.solution_cards_box .solution_card:hover .solu_title h3,
.solution_cards_box .solution_card:hover .solu_description p {
  color: #fff;
}

.solution_cards_box .solution_card:before {
  content: "";
  position: absolute;
  background: rgb(85 108 214 / 5%);
  width: 170px;
  height: 100px;
  z-index: -1;
  transform: rotate(42deg);
  right: -56px;
  top: -23px;
  border-radius: 35px;
}

.solution_cards_box .solution_card:hover .solu_description button {
  background: #fff !important;
  color: #309df0;
}

.solution_card .so_top_icon {
}

.solution_card .solu_title h3 {
  color: #212121;
  font-size: 1.3rem;
  margin-top: 13px;
  margin-bottom: 13px;
```



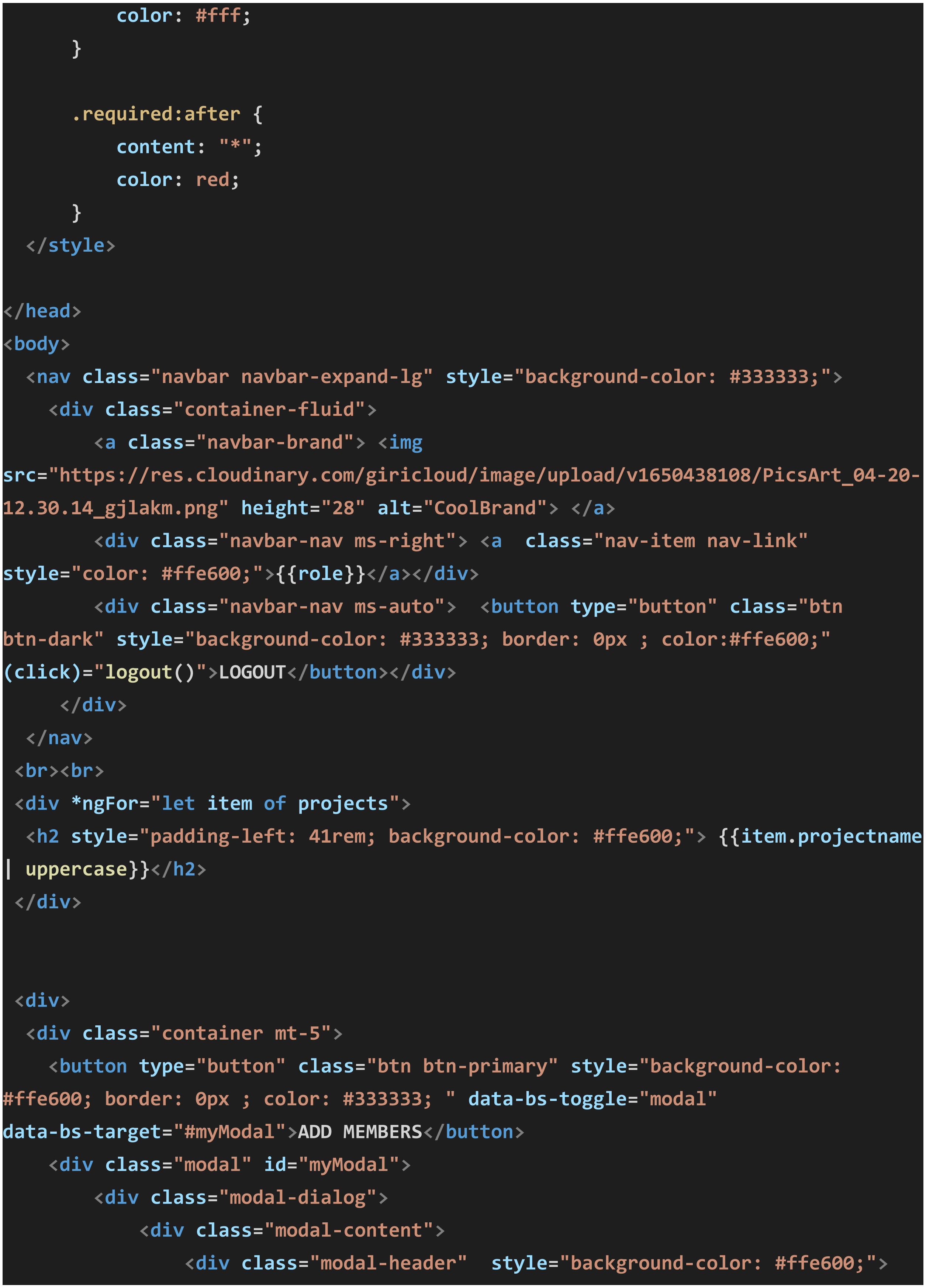
```css
}

.solution_card .solu_description p {
  font-size: 15px;
  margin-bottom: 15px;
}

.solution_card .solu_description button {
  border: 0;
  border-radius: 15px;
  background: linear-gradient(
    140deg,
    #ffe600 0%,
    #ffe600 50%,
    #ffe600 75%
  ) !important;
  color: #fff;
  font-weight: 500;
  font-size: 1rem;
  padding: 5px 16px;
}

.our_solution_content h1 {
  text-transform: capitalize;
  margin-bottom: 1rem;
  font-size: 2.5rem;
}
.our_solution_content p {
}

.hover_color_bubble {
  position: absolute;
  background: rgb(54 81 207 / 15%);
  width: 100rem;
  height: 50rem;
  left: 0;
  right: 0;
  z-index: -1;
  top: 16rem;
```



```
  border-radius: 50%;
  transform: rotate(-36deg);
  left: -18rem;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover .hover_color_bubble {
  top: 0rem;
}

.solution_cards_box .solution_card .so_top_icon {
  width: 60px;
  height: 30px;
  border-radius: 50%;
  background: #fff;
  overflow: hidden;
  display: flex;
  align-items: center;
  justify-content: center;
}

.solution_cards_box .solution_card .so_top_icon img {
  width: 40px;
  height: 20px;
  object-fit: contain;
}

/*start media query*/
@media screen and (min-width: 320px) {
  .sol_card_top_3 {
    position: relative;
    top: 0;
  }

  .our_solution_category {
    width: 100%;
    margin: 0 auto;
  }
  .our_solution_category .solution_cards_box {
    flex: auto;
  }
}
@media only screen and (min-width: 768px) {
  .our_solution_category .solution_cards_box {
    flex: 1;
  }
}
@media only screen and (min-width: 1024px) {
  .sol_card_top_3 {
    position: relative;
    top: -3rem;
  }
  .our_solution_category {
    width: 50%;
    margin: 0 auto;
  }
}
```

Component.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
```



```
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>

</head>
<body>
  <nav class="navbar navbar-expand-lg" style="background-color: #333333;">
    <div class="container-fluid">
        <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
        <div class="navbar-nav ms-right"> <a  class="nav-item nav-link"
style="color: #ffe600;">{{role}}</a></div>
        <div class="navbar-nav ms-auto">  <button type="button" class="btn
btn-dark" style="background-color: #333333; border: 0px ; color:#ffe600;"
(click)="logout()">LOGOUT</button></div>
     </div>
  </nav>
 <br><br>
 <div *ngFor="let item of projects">
  <h2 style="padding-left: 41rem; background-color: #ffe600;"> {{item.projectname
| uppercase}}</h2>
 </div>


 <div>
  <div class="container mt-5">
    <button type="button" class="btn btn-primary" style="background-color:
#ffe600; border: 0px ; color: #333333; " data-bs-toggle="modal"
data-bs-target="#myModal">ADD MEMBERS</button>
    <div class="modal" id="myModal">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header"  style="background-color: #ffe600;">
```

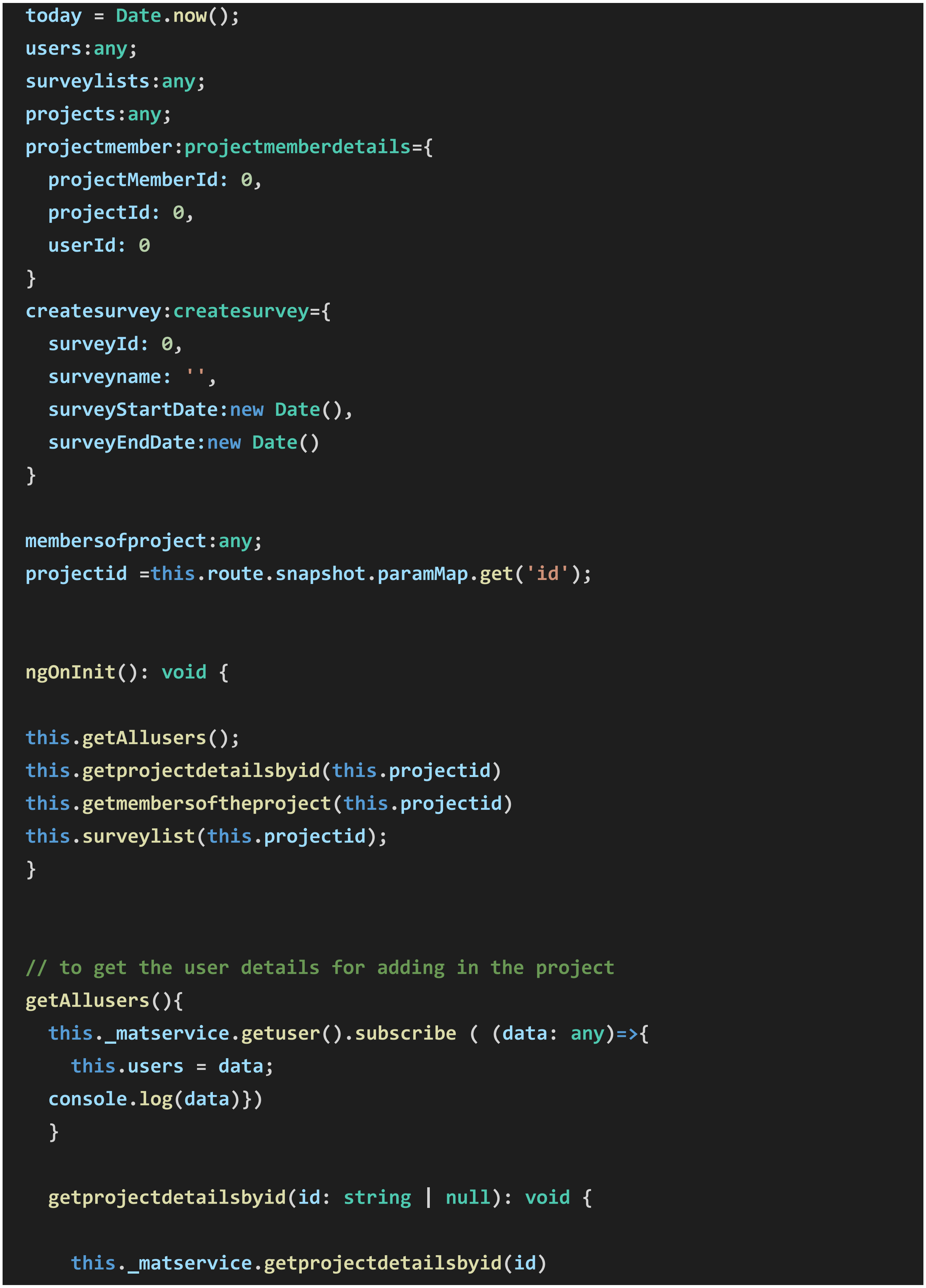
```
                    <h5 class="modal-title">ADD MEMBERS</h5>
                    <button type="button" class="btn-close"
data-bs-dismiss="modal"></button>
                </div>
                <div class="modal-body">
                    <form>
                      <select class="form-select" aria-label="Default select
example" id="userId"  name="userId" [(ngModel)]="projectmember.userId">
                          <option *ngFor="let user of users"
value="{{user.userId}}" >
                            {{user.firstName | uppercase}}
                          </option>

                      </select>
                    </form>
                </div>
                <div class="modal-footer">
                    <button type="submit" aria-label="Close"
(click)="addmember()" data-mdb-dismiss="modal"  class="btn
btn-primary">ADD</button>
                </div>
            </div>
        </div>
    </div>
</div>
 </div>

<!-- PROJECT MEMBER LIST -->
<ol class="list-group list-group-numbered w-25 p-3">
  <li class="list-group-item" *ngFor="let member of membersofproject" >NAME:
{{member.firstName}}{{member.lastName}} EMAILID:{{member.emailid}}</li>

</ol>



<!-- add survey -->

<div>
```

```
 <div class="container mt-5">
  <button type="button" class="btn btn-primary" style="background-color:
#ffe600; border: 0px ; color:#333333 " data-bs-toggle="modal"
data-bs-target="#myModall">ADD SURVEY</button>
  <div class="modal" id="myModall">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header"  style="background-color: #ffe600;">
          <h5 class="modal-title">ADD SURVEY</h5>
          <button type="button" class="btn-close"
data-bs-dismiss="modal"></button>
        </div>
        <div class="modal-body">
         <form>
          <div class="mb-3">
            <label class="form-label required">Survey  NAME</label>
            <input type="text" class="form-control"
name="projectname" id="projectname" [(ngModel)]="createsurvey.surveyname" >
          </div>
          <div class="mb-3">
            <label class="form-label required">Suvey Start
Date</label>
            <input type="date" class="form-control"
name="projectDescription" id="projectDescription" [min]="today"
[(ngModel)]="createsurvey.surveyStartDate">
          </div>
          <div class="mb-3">
           <label class="form-label required">Survey End Date</label>
           <input type="date" class="form-control"
name="projectDescription" id="projectDescription" [min]="today"
[(ngModel)]="createsurvey.surveyEndDate" >
         </div>
        </form>
        </div>
        <div class="modal-footer">
          <button type="submit" aria-label="Close"
(click)="addsurvey()" data-mdb-dismiss="modal"  class="btn
btn-primary">ADD</button>
        </div>
```

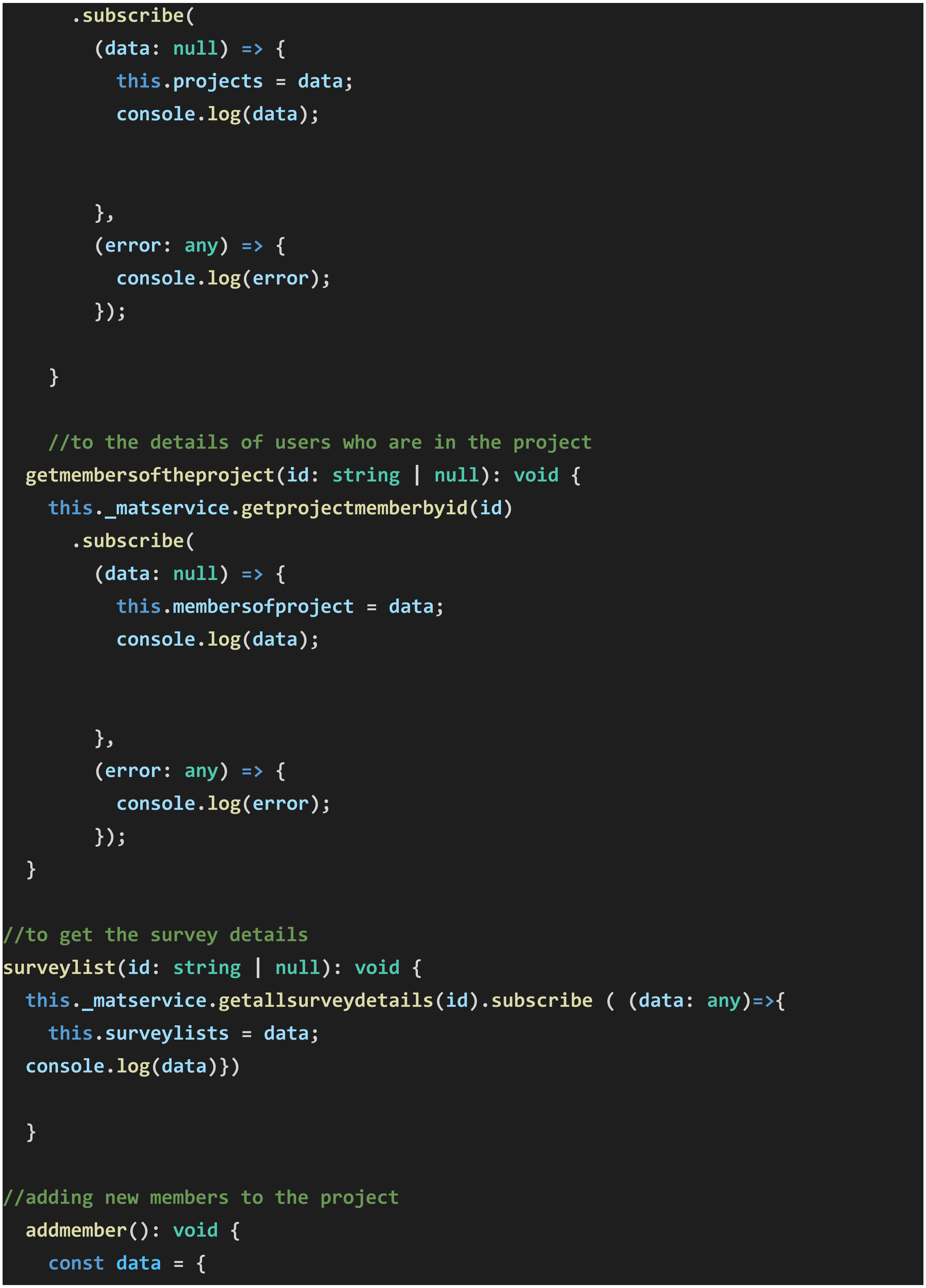
```
            </div>
        </div>
    </div>
</div>
 </div>


 <!-- surveylist -->
<br>
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

<div class="section_our_solution">
  <div class="row">
    <div *ngFor="let item of surveylists">
    <div class="col-lg-10 col-md-10 col-sm-10">
      <div class="our_solution_category">
        <div class="solution_cards_box">
          <div class="solution_card">

            <div class="solu_title">
              <h3>{{item.surveyname}}</h3>
            </div>
            <div class="solu_description">
              <p>
                {{item.surveyStartDate | date}}
              </p>
              <p>
                {{item.surveyEndDate | date}}
              </p>
              <button type="button"
routerLink="/AddquestionsComponent/{{item.surveyId}}" class="read_more_btn">Add
Questions</button>
              <button type="button"
routerLink="/ViewquestiondetailsComponent/{{item.surveyId}}"
class="read_more_btn">Review Survey</button>
              <!-- <a routerLink="/CreatesurveyComponent/{{item.projectId}}"
class="read_more_btn">Open Project</a> -->
            </div>
          </div>

    </div>
  </div>
</div>
</div>
</div>
</div>

          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroafW" crossorigin="anonymous"></script>

</body>
</html>
```

Component.ts

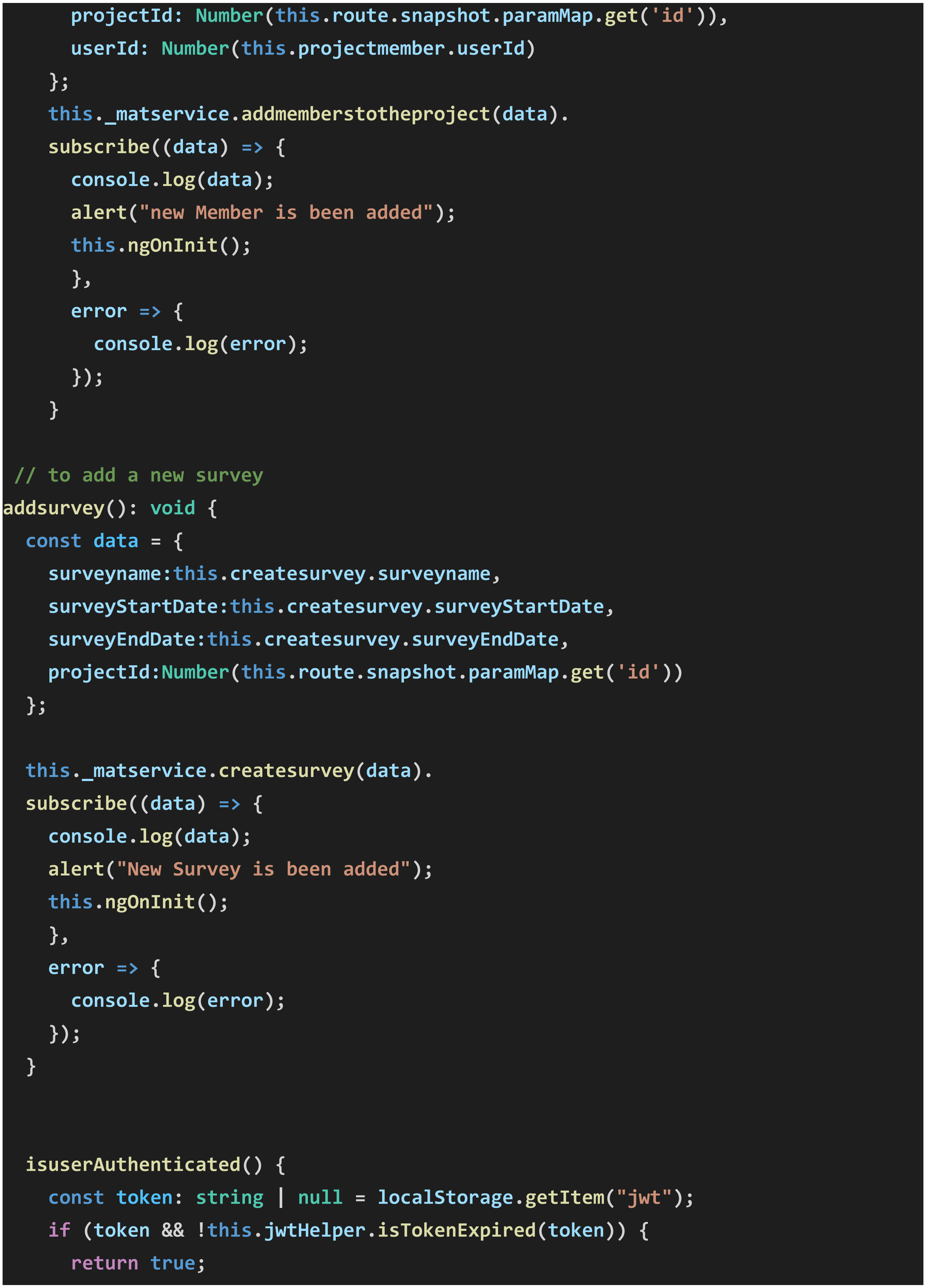
```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { MATService } from '../mat.service';
import { createsurvey } from './createsurvey';
import { projectmemberdetails } from './projectmemberdetails';
import{projectdetails} from 'src/app/createproject/projectdetails'

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

  constructor(private _matservice:MATService , private route: ActivatedRoute,private router: Router,private jwtHelper: JwtHelperService)  {}
  role=localStorage.getItem("usertype");
```



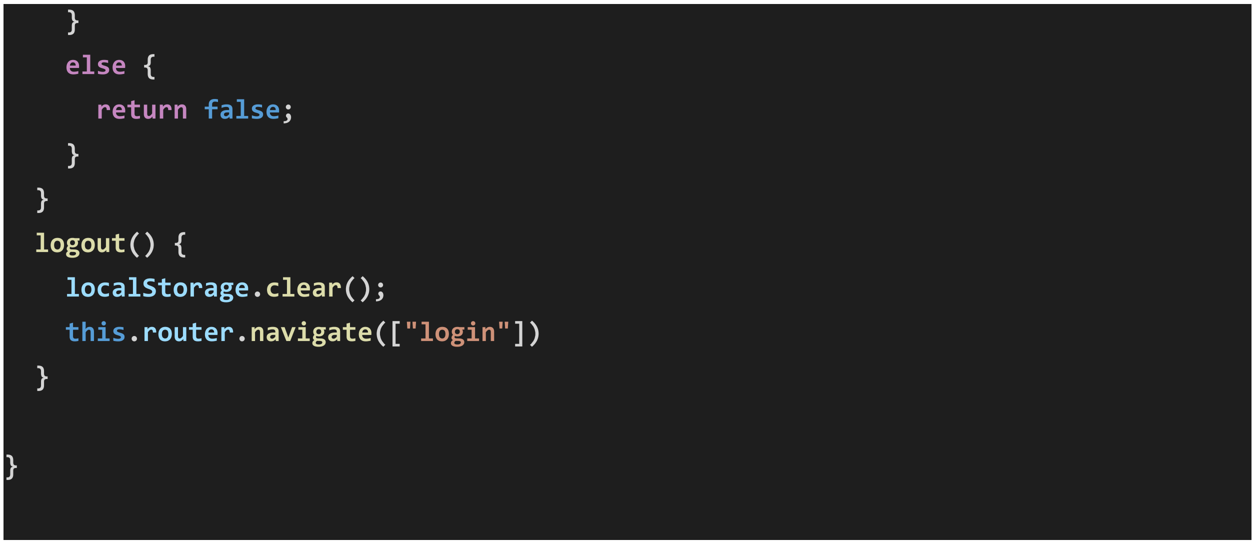
```
today = Date.now();
users:any;
surveylists:any;
projects:any;
projectmember:projectmemberdetails={
  projectMemberId: 0,
  projectId: 0,
  userId: 0
}
createsurvey:createsurvey={
  surveyId: 0,
  surveyname: '',
  surveyStartDate:new Date(),
  surveyEndDate:new Date()
}

membersofproject:any;
projectid =this.route.snapshot.paramMap.get('id');


ngOnInit(): void {

this.getAllusers();
this.getprojectdetailsbyid(this.projectid)
this.getmembersoftheproject(this.projectid)
this.surveylist(this.projectid);
}


// to get the user details for adding in the project
getAllusers(){
  this._matservice.getuser().subscribe ( (data: any)=>{
    this.users = data;
  console.log(data)})
  }

  getprojectdetailsbyid(id: string | null): void {

    this._matservice.getprojectdetailsbyid(id)
```



```
   .subscribe(
    (data: null) => {
     this.projects = data;
     console.log(data);


    },
    (error: any) => {
     console.log(error);
    });

  }

  //to the details of users who are in the project
 getmembersoftheproject(id: string | null): void {
  this._matservice.getprojectmemberbyid(id)
   .subscribe(
    (data: null) => {
     this.membersofproject = data;
     console.log(data);


    },
    (error: any) => {
     console.log(error);
    });
 }

//to get the survey details
surveylist(id: string | null): void {
 this._matservice.getallsurveydetails(id).subscribe ( (data: any)=>{
  this.surveylists = data;
 console.log(data)})

 }

//adding new members to the project
 addmember(): void {
  const data = {
```



```
      projectId: Number(this.route.snapshot.paramMap.get('id')),
      userId: Number(this.projectmember.userId)
    };
    this._matservice.addmemberstotheproject(data).
    subscribe((data) => {
      console.log(data);
      alert("new Member is been added");
      this.ngOnInit();
      },
      error => {
        console.log(error);
      });
    }

 // to add a new survey
addsurvey(): void {
  const data = {
    surveyname:this.createsurvey.surveyname,
    surveyStartDate:this.createsurvey.surveyStartDate,
    surveyEndDate:this.createsurvey.surveyEndDate,
    projectId:Number(this.route.snapshot.paramMap.get('id'))
  };

  this._matservice.createsurvey(data).
  subscribe((data) => {
    console.log(data);
    alert("New Survey is been added");
    this.ngOnInit();
    },
    error => {
      console.log(error);
    });
  }


  isuserAuthenticated() {
    const token: string | null = localStorage.getItem("jwt");
    if (token && !this.jwtHelper.isTokenExpired(token)) {
      return true;
```

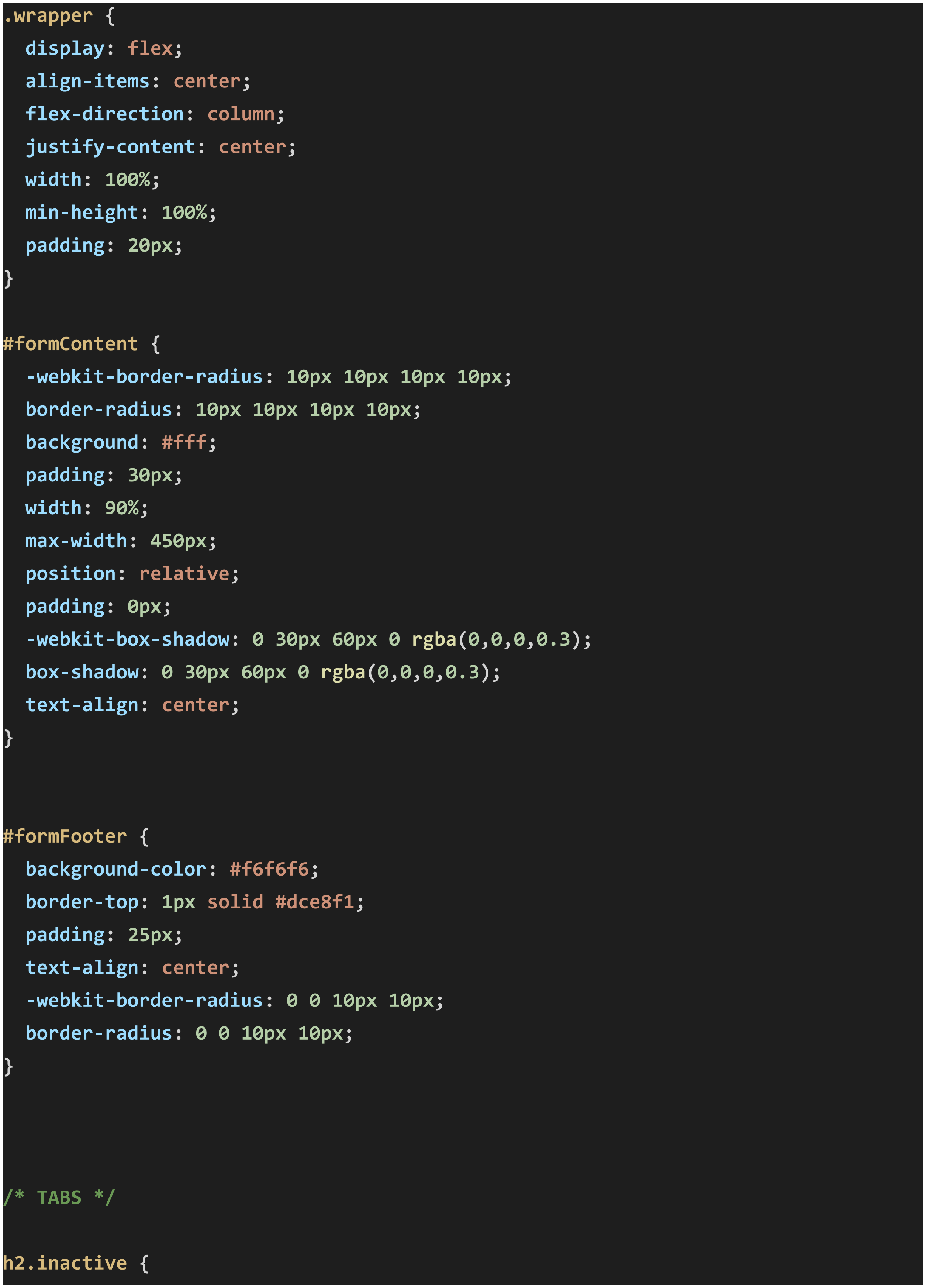


```
    }
    else {
      return false;
    }
  }
  logout() {
    localStorage.clear();
    this.router.navigate(["login"])
  }

}
```

Create User

Component.css

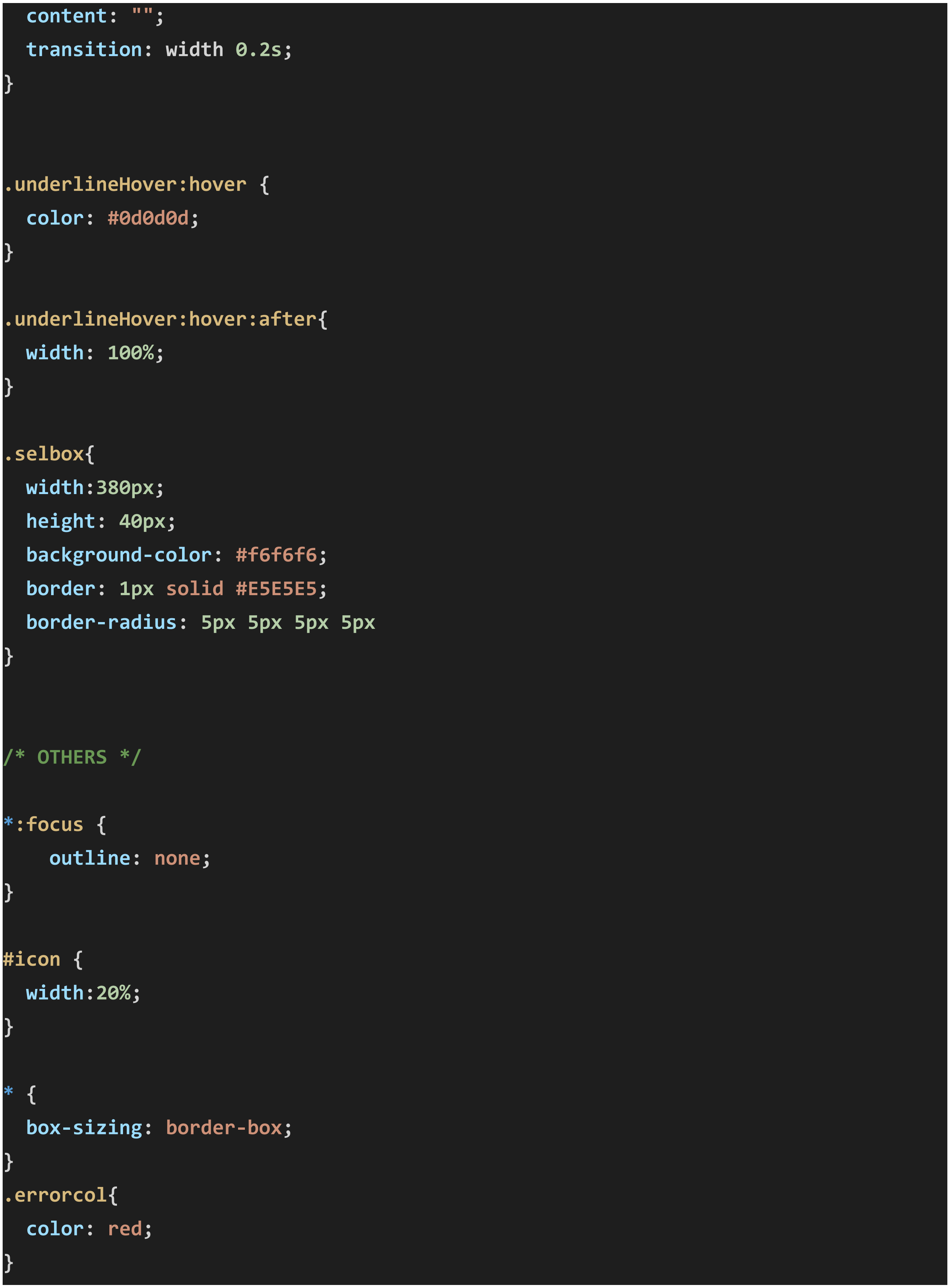
```css
@import url('https://fonts.googleapis.com/css?family=Poppins');

/* BASIC */

html {
  background-color: #56baed;
}

body {
  font-family: "Poppins", sans-serif;
  height: 100vh;
  background-color: #ebebeb;
}

a {
  color: #92badd;
  display:inline-block;
  text-decoration: none;
  font-weight: 400;
}

h2 {
  text-align: center;
  font-size: 16px;
  font-weight: 600;
  text-transform: uppercase;
  display:inline-block;
  margin: 40px 8px 10px 8px;
  color: #cccccc;
}



/* STRUCTURE */
```



```css
.wrapper {
  display: flex;
  align-items: center;
  flex-direction: column;
  justify-content: center;
  width: 100%;
  min-height: 100%;
  padding: 20px;
}

#formContent {
  -webkit-border-radius: 10px 10px 10px 10px;
  border-radius: 10px 10px 10px 10px;
  background: #fff;
  padding: 30px;
  width: 90%;
  max-width: 450px;
  position: relative;
  padding: 0px;
  -webkit-box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
  box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
  text-align: center;
}


#formFooter {
  background-color: #f6f6f6;
  border-top: 1px solid #dce8f1;
  padding: 25px;
  text-align: center;
  -webkit-border-radius: 0 0 10px 10px;
  border-radius: 0 0 10px 10px;
}



/* TABS */

h2.inactive {
  color: #cccccc;
}

h2.active {
  color: #0d0d0d;
  border-bottom: 2px solid #5fbae9;
}



/* FORM TYPOGRAPHY*/

input[type=button], input[type=submit], input[type=reset]  {
  background-color: #56baed;
  border: none;
  color: white;
  padding: 15px 80px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  text-transform: uppercase;
  font-size: 13px;
  -webkit-box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
  box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
  -webkit-border-radius: 5px 5px 5px 5px;
  border-radius: 5px 5px 5px 5px;
  margin: 5px 20px 40px 20px;
  -webkit-transition: all 0.3s ease-in-out;
  -moz-transition: all 0.3s ease-in-out;
  -ms-transition: all 0.3s ease-in-out;
  -o-transition: all 0.3s ease-in-out;
  transition: all 0.3s ease-in-out;
}

input[type=button]:hover, input[type=submit]:hover, input[type=reset]:hover  {
  background-color: #39ace7;
}

input[type=button]:active, input[type=submit]:active, input[type=reset]:active  {
```

```css
  -moz-transform: scale(0.95);
  -webkit-transform: scale(0.95);
  -o-transform: scale(0.95);
  -ms-transform: scale(0.95);
  transform: scale(0.95);
}

input[type=text],input[type=password],input[type=email] {
  background-color: #f6f6f6;
  border: none;
  color: #0d0d0d;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  margin: 5px;
  width: 85%;
  border: 2px solid #f6f6f6;
  -webkit-transition: all 0.5s ease-in-out;
  -moz-transition: all 0.5s ease-in-out;
  -ms-transition: all 0.5s ease-in-out;
  -o-transition: all 0.5s ease-in-out;
  transition: all 0.5s ease-in-out;
  -webkit-border-radius: 5px 5px 5px 5px;
  border-radius: 5px 5px 5px 5px;
}
input[type=radio]{
  background-color: #f6f6f6;
  border: none;
  color: #0d0d0d;
  padding: 15px 32px;
  text-align: left;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  margin: 5px;

  border: 2px solid #f6f6f6;
  -webkit-transition: all 0.5s ease-in-out;
  -moz-transition: all 0.5s ease-in-out;
  -ms-transition: all 0.5s ease-in-out;
  -o-transition: all 0.5s ease-in-out;
  transition: all 0.5s ease-in-out;
  -webkit-border-radius: 5px 5px 5px 5px;
  border-radius: 5px 5px 5px 5px;
}

input[type=text]:focus
,input[type=password]:focus,input[type=email]:focus,input[type=radio]:focus{
  background-color: #fff;
  border-bottom: 2px solid #5fbae9;
}

input[type=text]:placeholder,input[type=password]:placeholder,input[type=email]:p
laceholder {
  color: #cccccc;
}



/* ANIMATIONS */

/* Simple CSS3 Fade-in-down Animation */
.fadeInDown {
  -webkit-animation-name: fadeInDown;
  animation-name: fadeInDown;
  -webkit-animation-duration: 1s;
  animation-duration: 1s;
  -webkit-animation-fill-mode: both;
  animation-fill-mode: both;
}

@-webkit-keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}

@keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}

/* Simple CSS3 Fade-in Animation */
@-webkit-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@-moz-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@keyframes fadeIn { from { opacity:0; } to { opacity:1; } }

.fadeIn {
  opacity:0;
  -webkit-animation:fadeIn ease-in 1;
  -moz-animation:fadeIn ease-in 1;
  animation:fadeIn ease-in 1;

  -webkit-animation-fill-mode:forwards;
  -moz-animation-fill-mode:forwards;
  animation-fill-mode:forwards;

  -webkit-animation-duration:1s;
  -moz-animation-duration:1s;
  animation-duration:1s;
```

```
}

.fadeIn.first {
  -webkit-animation-delay: 0.4s;
  -moz-animation-delay: 0.4s;
  animation-delay: 0.4s;
}

.fadeIn.second {
  -webkit-animation-delay: 0.6s;
  -moz-animation-delay: 0.6s;
  animation-delay: 0.6s;
}

.fadeIn.third {
  -webkit-animation-delay: 0.8s;
  -moz-animation-delay: 0.8s;
  animation-delay: 0.8s;
}

.fadeIn.fourth {
  -webkit-animation-delay: 1s;
  -moz-animation-delay: 1s;
  animation-delay: 1s;
}
.fadeIn.five{
  -webkit-animation-delay: 1.2s;
  -moz-animation-delay: 1.2s;
  animation-delay: 1.2s;
}

/* Simple CSS3 Fade-in Animation */
.underlineHover:after {
  display: block;
  left: 0;
  bottom: -10px;
  width: 0;
  height: 2px;
  background-color: #56baed;
```

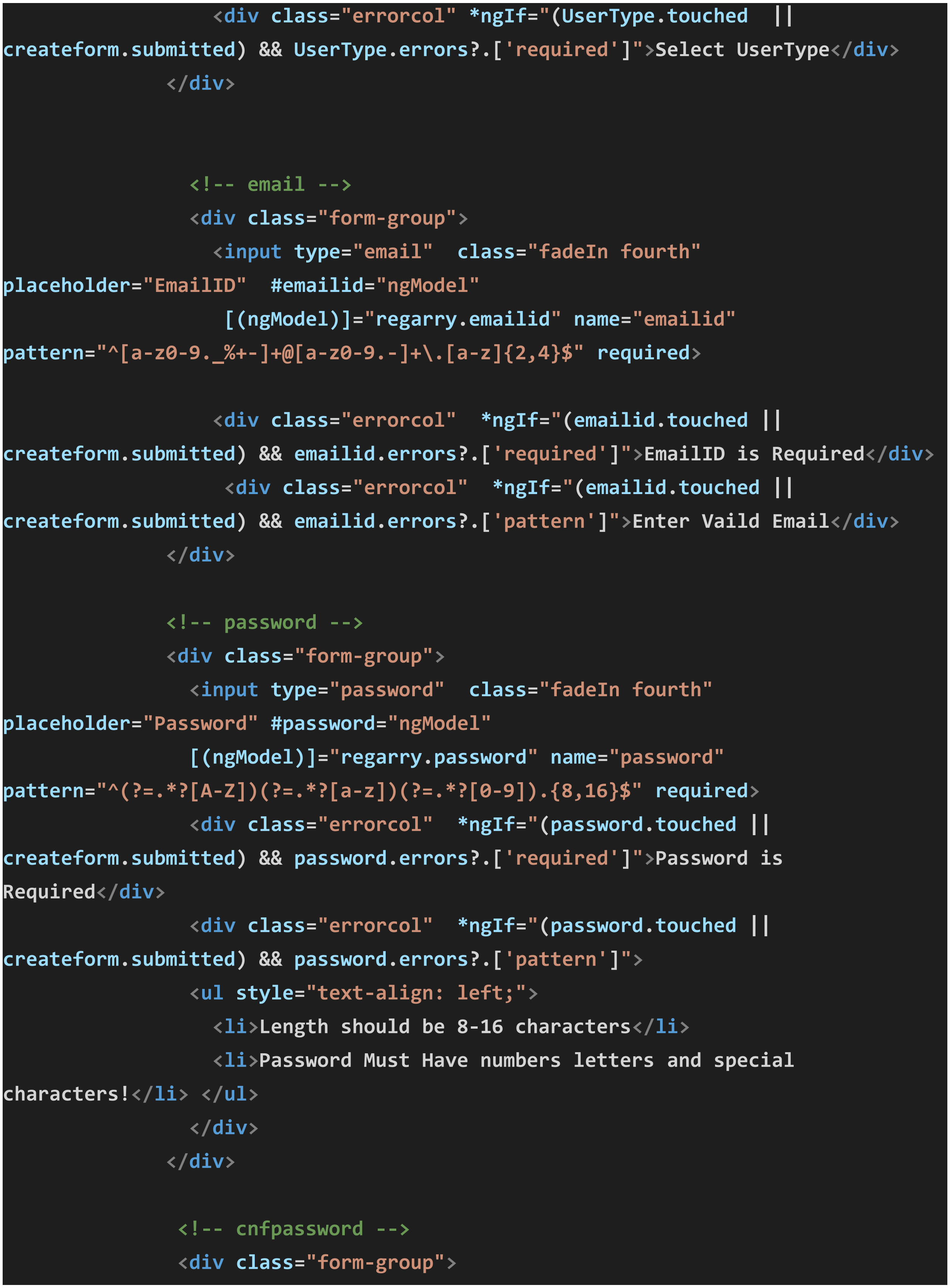


```
  content: "";
  transition: width 0.2s;
}


.underlineHover:hover {
  color: #0d0d0d;
}

.underlineHover:hover:after{
  width: 100%;
}

.selbox{
  width:380px;
  height: 40px;
  background-color: #f6f6f6;
  border: 1px solid #E5E5E5;
  border-radius: 5px 5px 5px 5px
}


/* OTHERS */

*:focus {
    outline: none;
}

#icon {
  width:20%;
}

* {
  box-sizing: border-box;
}
.errorcol{
  color: red;
}
```

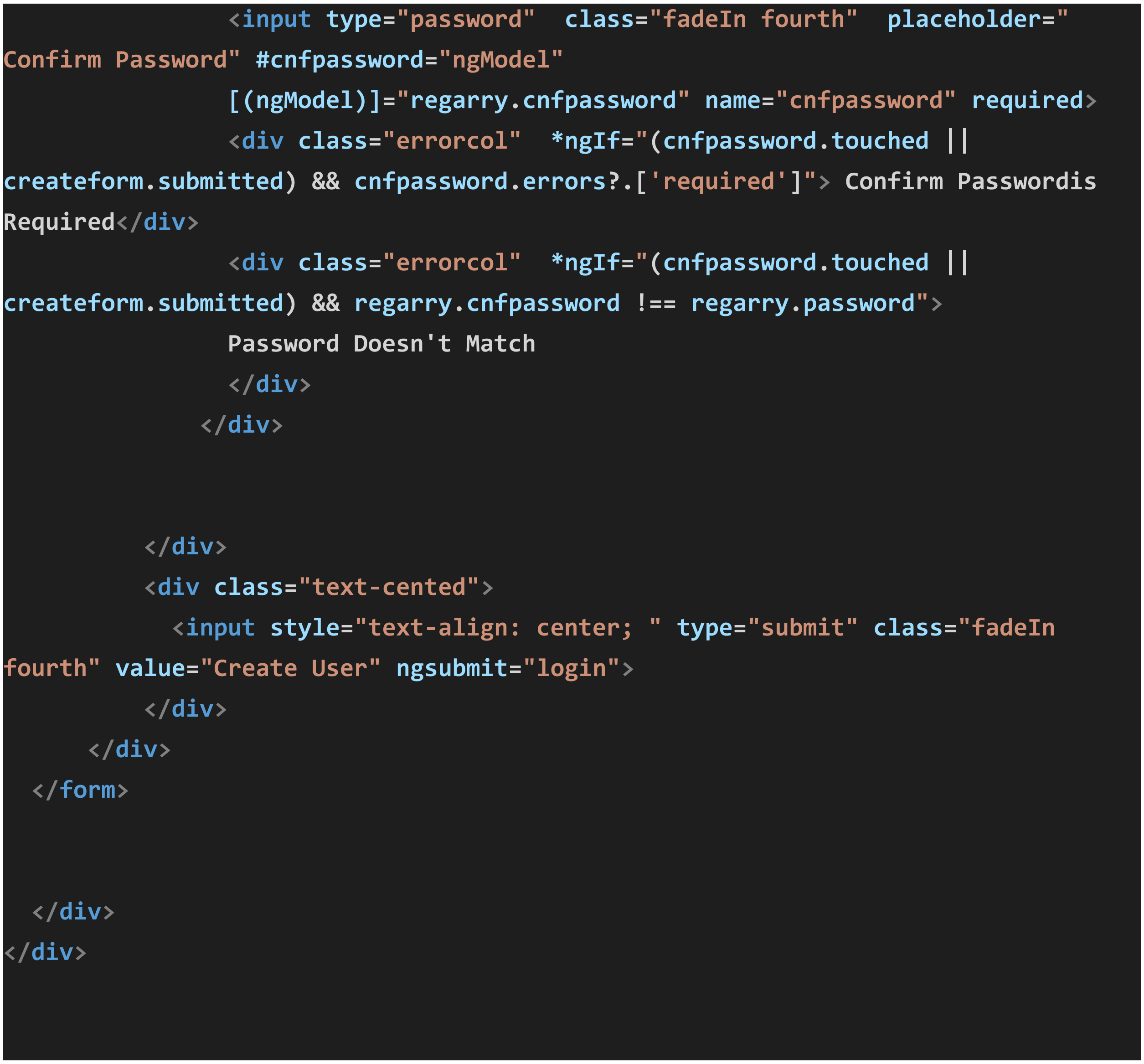
Component.html

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark" style="background-color:
#333333;">
  <div class="container-fluid">
      <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
      <div class="navbar-nav ms-right"> <a href="Home" class="nav-item nav-link"
style="color:#ffe600;">Home</a></div>
    </div>
</nav>

<div class="wrapper fadeInDown">
  <div id="formContent">
    <!-- Tabs Titles -->
    <h2 class="active"> Sign Up </h2>

    <!-- Icon -->


    <!-- Create Form -->
    <form #createform="ngForm" (ngSubmit)="createform.form.valid && addemp()"
noValidate>
      <div class="panel panel-primary">
          <div class="panel-heading">
              <h3 class="panel-title"></h3>
          </div>
          <div class="panel-body">
            <!-- firstName -->
              <div class="form-group">
                <input type="text"  class="fadeIn first" id="firstName"
placeholder="FirstName" #firstName="ngModel"
                [(ngModel)]="regarry.firstName" name="firstName"  minlength="4"
required >
               <div class="errorcol" *ngIf="(firstName.touched ||
createform.submitted) && firstName.errors?.['required']">FirstName is
Required</div>
```

```
               <div class="errorcol"  *ngIf="(firstName.touched  ||
createform.submitted) && firstName.errors?.['minlength']">Required 4
characters</div>
              </div>

              <!-- lastName -->
              <div class="form-group">
                <input type="text"  class="fadeIn second" placeholder="LastName"
#lastName="ngModel"
                [(ngModel)]="regarry.lastName" name="lastName" minlength="4"
required >
                <div class="errorcol"  *ngIf="(lastName.touched ||
createform.submitted) && lastName.errors?.['required']">LastName is
Required</div>
                <div class="errorcol"  *ngIf="(lastName.touched  ||
createform.submitted) && lastName.errors?.['minlength']">Required 4
characters</div>
                </div>



             <!-- Gender -->

              <div class="form-group fadeIn third ">
                    <input type="radio" name="gender" value="male" required>Male
                    <input type="radio" name="gender" value="female"
required>Female
              </div>

              <!-- Dropdown box-->
              <div class="form-group ">
                  <select id="UserType" name="UserType" class="selbox fadeIn
fourth"    #UserType="ngModel"  [(ngModel)]="regarry.userType"
[(ngModel)]='ngSelect' required>
                    <option value="1">Select Your UserType</option>
                    <option value="Project Manager">Project Manager</option>
                    <option value="Team Member">Team Member</option>
                  </select>
```

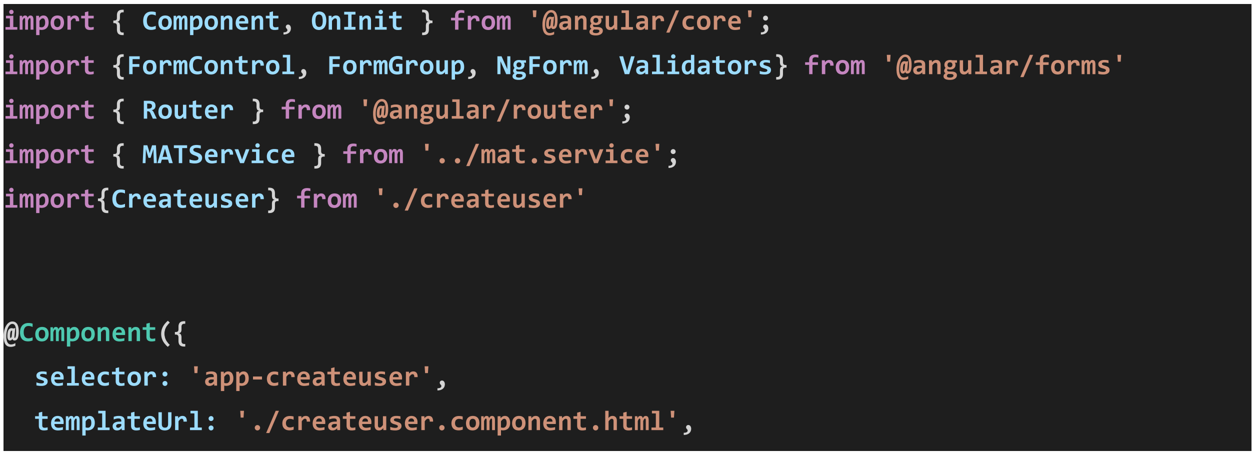


```
                  <div class="errorcol" *ngIf="(UserType.touched  ||
createform.submitted) && UserType.errors?.['required']">Select UserType</div>
              </div>


                <!-- email -->
                <div class="form-group">
                  <input type="email"  class="fadeIn fourth"
placeholder="EmailID"  #emailid="ngModel"
                   [(ngModel)]="regarry.emailid" name="emailid"
pattern="^[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$" required>

                  <div class="errorcol"  *ngIf="(emailid.touched ||
createform.submitted) && emailid.errors?.['required']">EmailID is Required</div>
                   <div class="errorcol"  *ngIf="(emailid.touched ||
createform.submitted) && emailid.errors?.['pattern']">Enter Vaild Email</div>
              </div>

              <!-- password -->
              <div class="form-group">
                <input type="password"  class="fadeIn fourth"
placeholder="Password" #password="ngModel"
                [(ngModel)]="regarry.password" name="password"
pattern="^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9]).{8,16}$" required>
                <div class="errorcol"  *ngIf="(password.touched ||
createform.submitted) && password.errors?.['required']">Password is
Required</div>
                <div class="errorcol"  *ngIf="(password.touched ||
createform.submitted) && password.errors?.['pattern']">
                <ul style="text-align: left;">
                  <li>Length should be 8-16 characters</li>
                  <li>Password Must Have numbers letters and special
characters!</li> </ul>
                </div>
              </div>

               <!-- cnfpassword -->
               <div class="form-group">
```

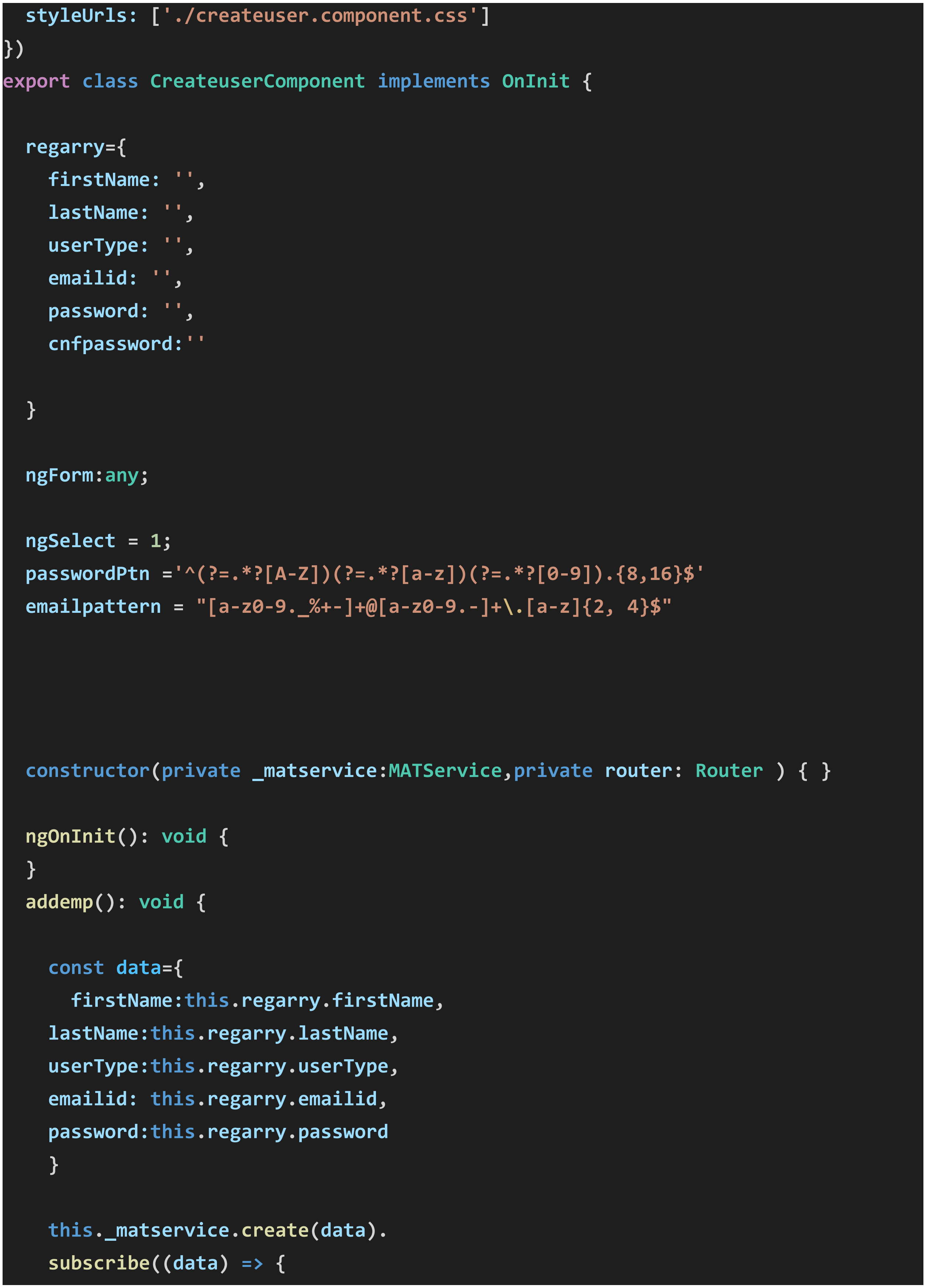


```
                <input type="password"  class="fadeIn fourth"  placeholder="
Confirm Password" #cnfpassword="ngModel"
                [(ngModel)]="regarry.cnfpassword" name="cnfpassword" required>
                <div class="errorcol"  *ngIf="(cnfpassword.touched ||
createform.submitted) && cnfpassword.errors?.['required']"> Confirm Passwordis
Required</div>
                <div class="errorcol"  *ngIf="(cnfpassword.touched ||
createform.submitted) && regarry.cnfpassword !== regarry.password">
                Password Doesn't Match
                </div>
              </div>


          </div>
          <div class="text-cented">
            <input style="text-align: center; " type="submit" class="fadeIn
fourth" value="Create User" ngsubmit="login">
          </div>
      </div>
  </form>


  </div>
</div>
```

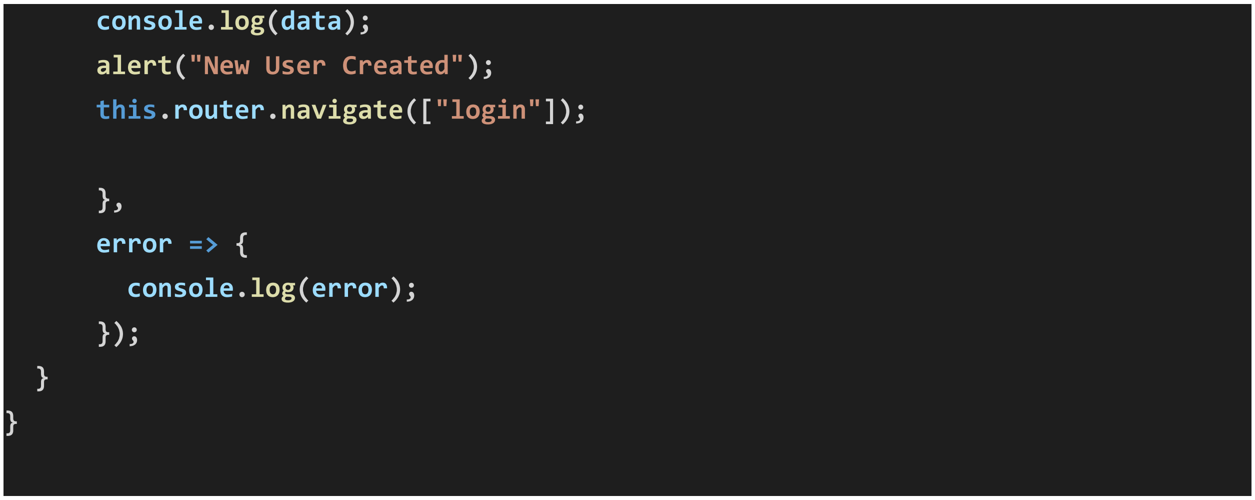
Component.ts



```
import { Component, OnInit } from '@angular/core';
import {FormControl, FormGroup, NgForm, Validators} from '@angular/forms'
import { Router } from '@angular/router';
import { MATService } from '../mat.service';
import{Createuser} from './createuser'


@Component({
  selector: 'app-createuser',
  templateUrl: './createuser.component.html',
```



```
  styleUrls: ['./createuser.component.css']
})
export class CreateuserComponent implements OnInit {

  regarry={
    firstName: '',
    lastName: '',
    userType: '',
    emailid: '',
    password: '',
    cnfpassword:''

  }

  ngForm:any;

  ngSelect = 1;
  passwordPtn ='^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9]).{8,16}$'
  emailpattern = "[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2, 4}$"




  constructor(private _matservice:MATService,private router: Router ) { }

  ngOnInit(): void {
  }
  addemp(): void {

    const data={
      firstName:this.regarry.firstName,
    lastName:this.regarry.lastName,
    userType:this.regarry.userType,
    emailid: this.regarry.emailid,
    password:this.regarry.password
    }

    this._matservice.create(data).
    subscribe((data) => {
```



```
      console.log(data);
      alert("New User Created");
      this.router.navigate(["login"]);

      },
      error => {
        console.log(error);
      });
  }
}
```

Home

Component.css

```
.carousel-item {
  height: 92.7vh;
  min-height: 350px;
  background: no-repeat center center scroll;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  background-size: cover;
}
```

Component.html

```

<nav class="navbar navbar-expand-lg  bg-dark" style="background-color: #333333;">
  <div class="container-fluid">
      <a class="navbar-brand"> <img src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
      <div class="navbar-nav ms-right"> <a href="Home" class="nav-item nav-link" style="color: #ffe600;">Home</a></div>
      <div class="navbar-nav ms-auto"> <a href="login" class="nav-item nav-link" style="color: #ffe600;">Login</a></div>
   </div>
</nav>
<div id="carouselExampleSlidesOnly" class="carousel slide" data-bs-ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      <img src="https://res.cloudinary.com/giricloud/image/upload/v1650447535/PicsArt_04-20-03.04.51_vh8dmk.jpg" class="d-block w-100" alt="...">
    </div>
    <div class="carousel-item">
```

```
      <img
src="https://securecdn.pymnts.com/wp-content/uploads/2018/12/gartner-corporate-fi
nance-tech-adoption.jpg" class="d-block w-100" alt="...">
    </div>
    <div class="carousel-item">
      <img
src="https://securecdn.pymnts.com/wp-content/uploads/2017/04/enterprise-technolog
y-corporate-adoption-leadership-sulture-sage-capital-one-napcp-sas-survey-1000x60
0.jpg" class="d-block w-100" alt="...">
    </div>
  </div>
</div>
```

Component.spec.ts

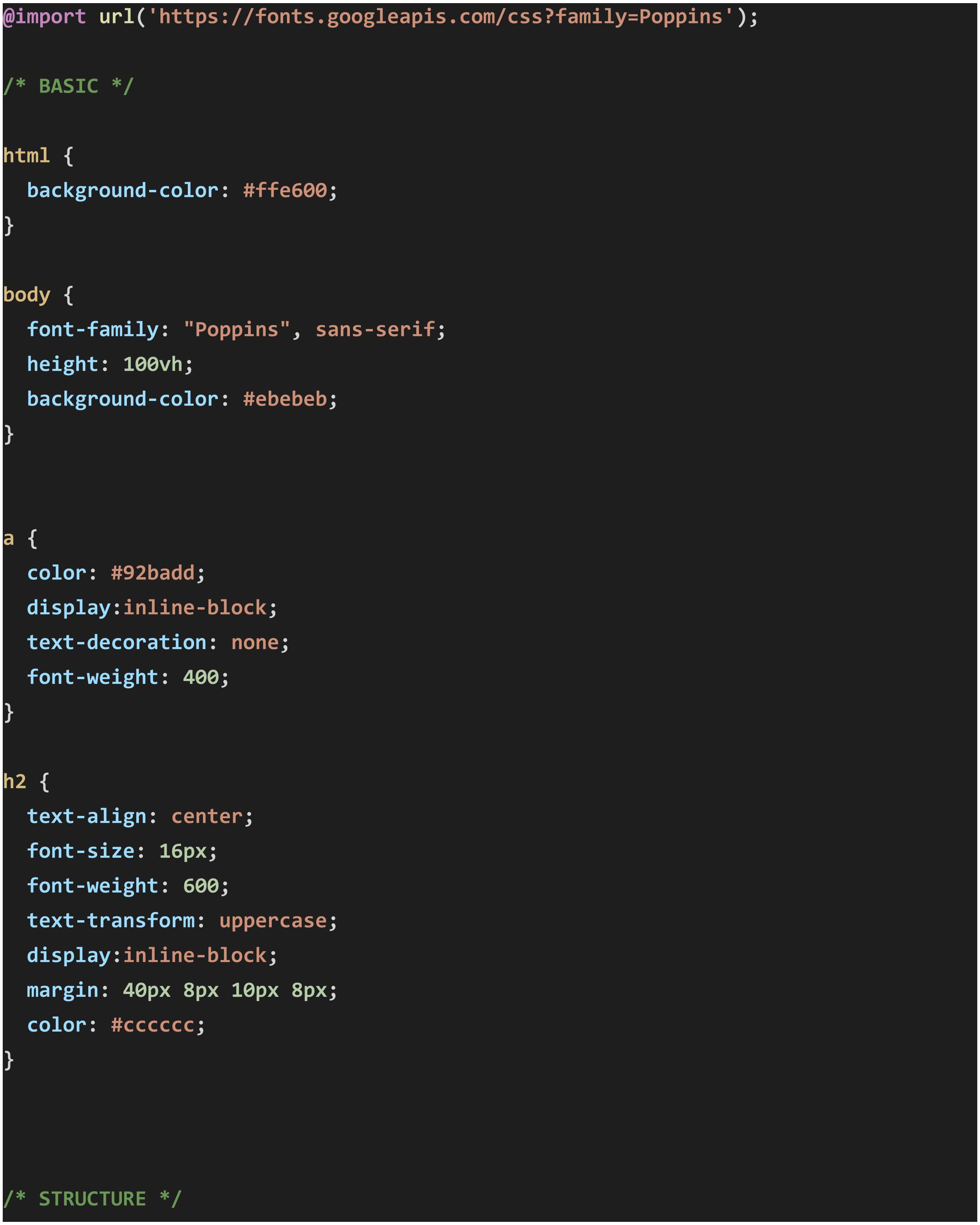
```
import { ComponentFixture, TestBed } from '@angular/core/testing';

import { HomeComponent } from './home.component';

describe('HomeComponent', () => {
  let component: HomeComponent;
  let fixture: ComponentFixture<HomeComponent>;

  beforeEach(async () => {
    await TestBed.configureTestingModule({
      declarations: [ HomeComponent ]
    })
    .compileComponents();
  });

  beforeEach(() => {
    fixture = TestBed.createComponent(HomeComponent);
    component = fixture.componentInstance;
    fixture.detectChanges();
  });

  it('should create', () => {
    expect(component).toBeTruthy();
  });
});
```

Component.ts

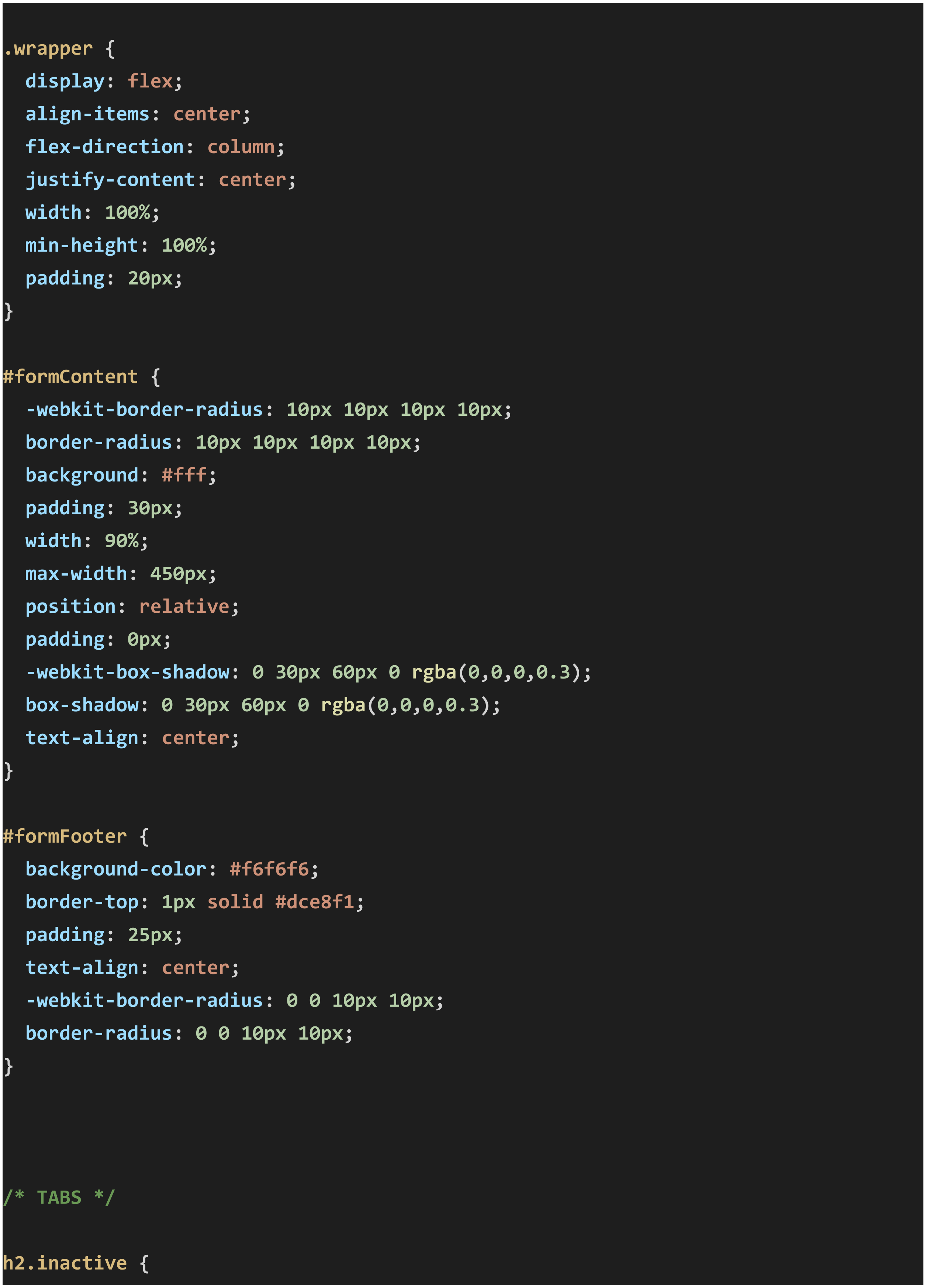
```
import { Component, OnInit } from '@angular/core';

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

  constructor() { }

  ngOnInit(): void {
  }

}
```

Login

Component.css

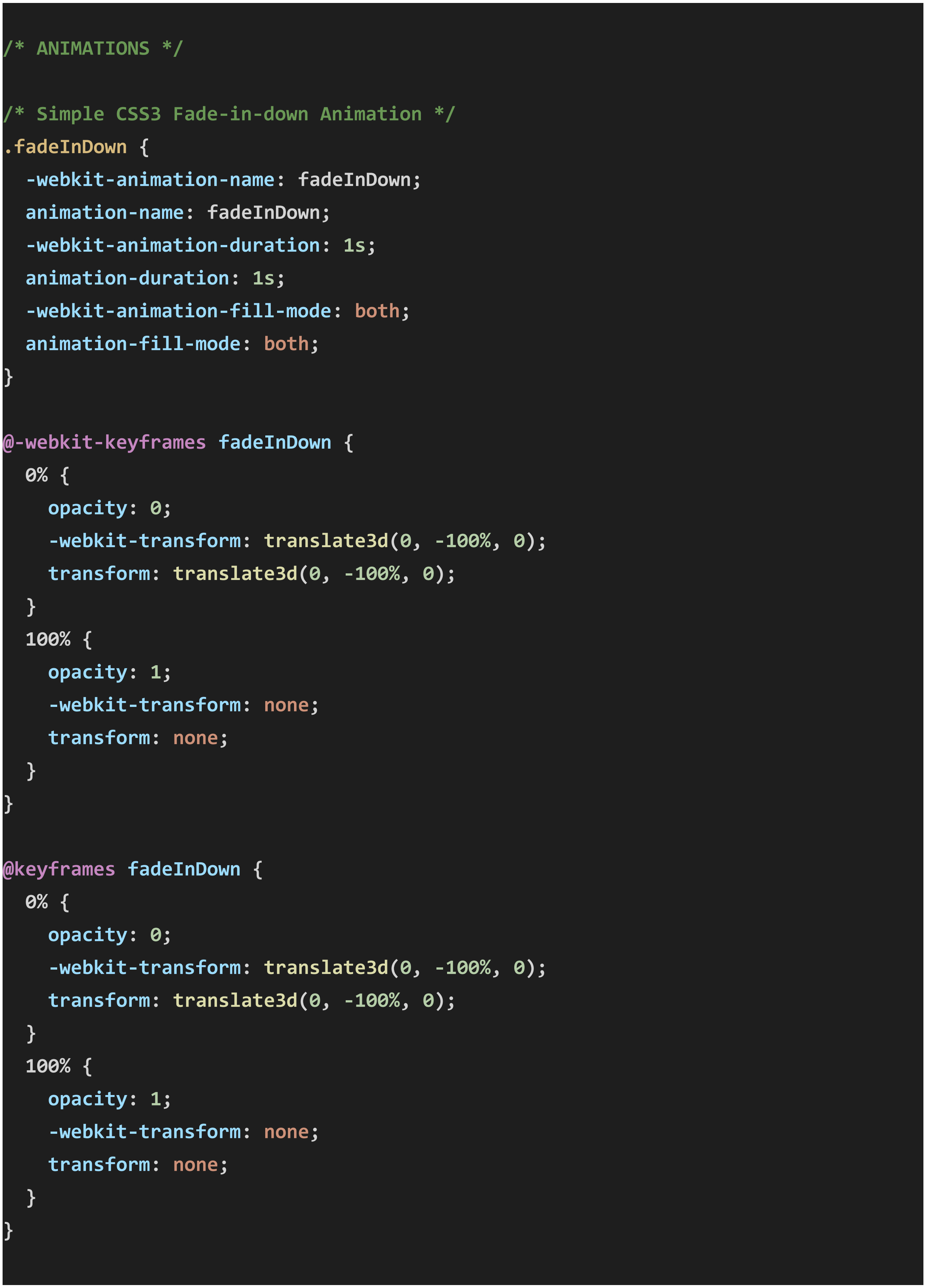


```css
@import url('https://fonts.googleapis.com/css?family=Poppins');

/* BASIC */

html {
  background-color: #ffe600;
}

body {
  font-family: "Poppins", sans-serif;
  height: 100vh;
  background-color: #ebebeb;
}


a {
  color: #92badd;
  display:inline-block;
  text-decoration: none;
  font-weight: 400;
}

h2 {
  text-align: center;
  font-size: 16px;
  font-weight: 600;
  text-transform: uppercase;
  display:inline-block;
  margin: 40px 8px 10px 8px;
  color: #cccccc;
}



/* STRUCTURE */
```

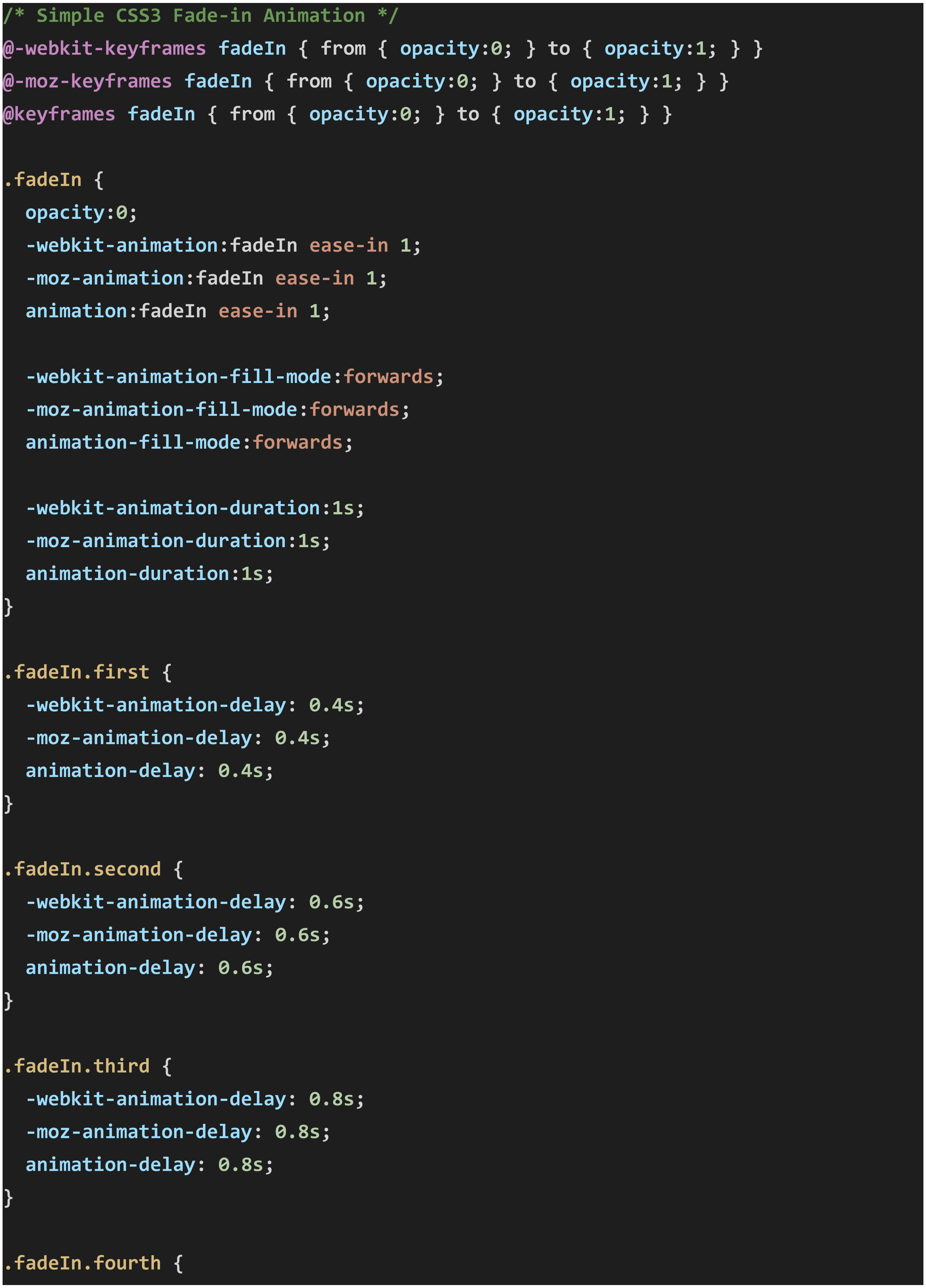


```
.wrapper {
  display: flex;
  align-items: center;
  flex-direction: column;
  justify-content: center;
  width: 100%;
  min-height: 100%;
  padding: 20px;
}

#formContent {
  -webkit-border-radius: 10px 10px 10px 10px;
  border-radius: 10px 10px 10px 10px;
  background: #fff;
  padding: 30px;
  width: 90%;
  max-width: 450px;
  position: relative;
  padding: 0px;
  -webkit-box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
  box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
  text-align: center;
}

#formFooter {
  background-color: #f6f6f6;
  border-top: 1px solid #dce8f1;
  padding: 25px;
  text-align: center;
  -webkit-border-radius: 0 0 10px 10px;
  border-radius: 0 0 10px 10px;
}



/* TABS */

h2.inactive {
```

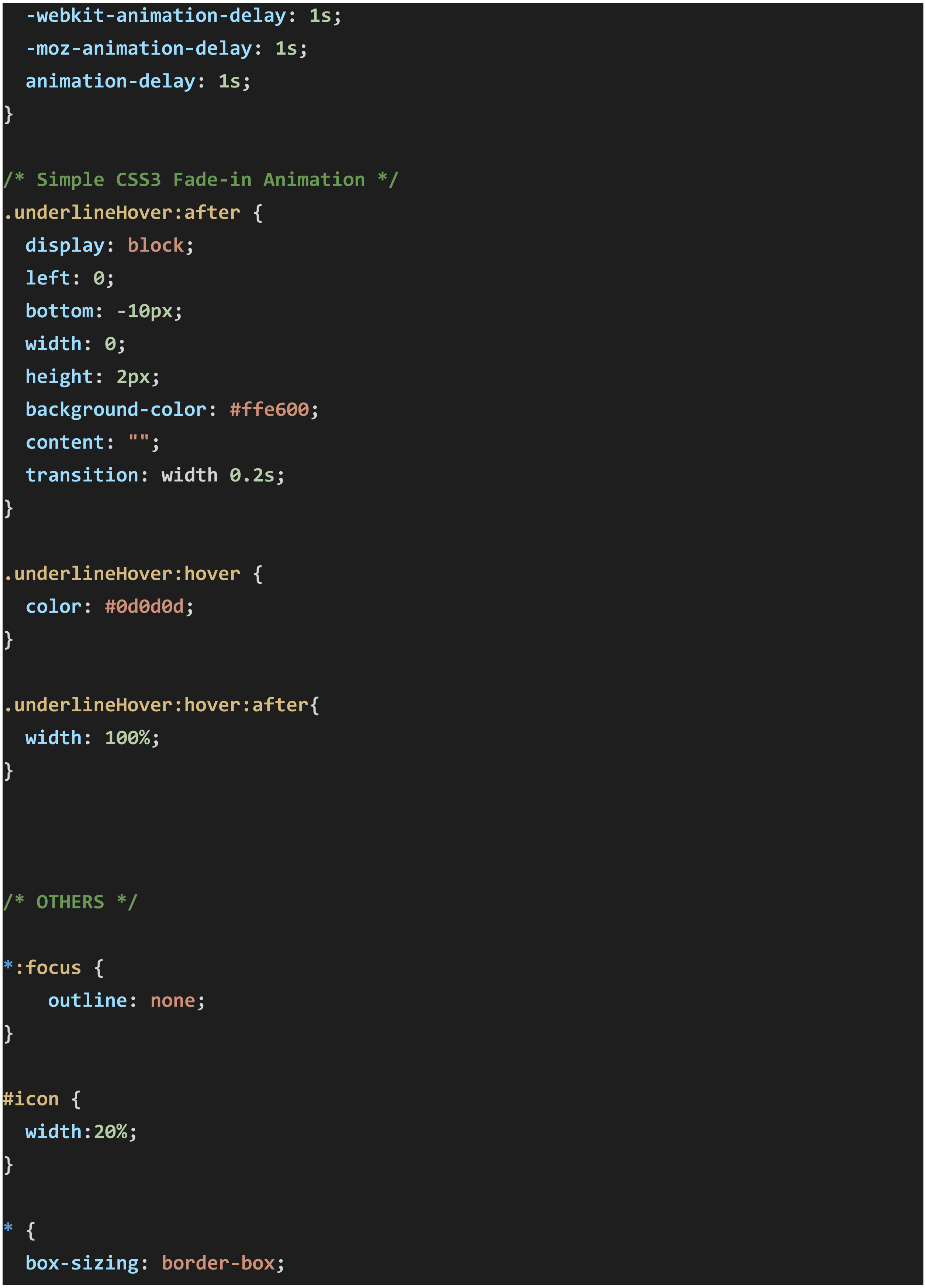
```css
  color: #cccccc;
}

h2.active {
  color: #0d0d0d;
  border-bottom: 2px solid #ffe600;
}



/* FORM TYPOGRAPHY*/

input[type=button], input[type=submit], input[type=reset]  {
  background-color: #ffe600;
  border: none;
  color: white;
  padding: 15px 80px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  text-transform: uppercase;
  font-size: 13px;
  -webkit-box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
  box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
  -webkit-border-radius: 5px 5px 5px 5px;
  border-radius: 5px 5px 5px 5px;
  margin: 5px 20px 40px 20px;
  -webkit-transition: all 0.3s ease-in-out;
  -moz-transition: all 0.3s ease-in-out;
  -ms-transition: all 0.3s ease-in-out;
  -o-transition: all 0.3s ease-in-out;
  transition: all 0.3s ease-in-out;
}

input[type=button]:hover, input[type=submit]:hover, input[type=reset]:hover  {
  background-color: #ffe600;
}

input[type=button]:active, input[type=submit]:active, input[type=reset]:active  {
  -moz-transform: scale(0.95);
  -webkit-transform: scale(0.95);
  -o-transform: scale(0.95);
  -ms-transform: scale(0.95);
  transform: scale(0.95);
}

input[type=text],input[type=password],input[type=email] {
  background-color: #f6f6f6;
  border: none;
  color: #0d0d0d;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  margin: 5px;
  width: 85%;
  border: 2px solid #f6f6f6;
  -webkit-transition: all 0.5s ease-in-out;
  -moz-transition: all 0.5s ease-in-out;
  -ms-transition: all 0.5s ease-in-out;
  -o-transition: all 0.5s ease-in-out;
  transition: all 0.5s ease-in-out;
  -webkit-border-radius: 5px 5px 5px 5px;
  border-radius: 5px 5px 5px 5px;
}

input[type=text]:focus,input[type=password]:focus,input[type=email]:focus{
  background-color: #fff;
  border-bottom: 2px solid #5fbae9;
}

input[type=text]:placeholder,input[type=password]:placeholder,input[type=email]:placeholder{
  color: #cccccc;
}
```



```
/* ANIMATIONS */

/* Simple CSS3 Fade-in-down Animation */
.fadeInDown {
  -webkit-animation-name: fadeInDown;
  animation-name: fadeInDown;
  -webkit-animation-duration: 1s;
  animation-duration: 1s;
  -webkit-animation-fill-mode: both;
  animation-fill-mode: both;
}

@-webkit-keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}

@keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}
```



```
/* Simple CSS3 Fade-in Animation */
@-webkit-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@-moz-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@keyframes fadeIn { from { opacity:0; } to { opacity:1; } }

.fadeIn {
  opacity:0;
  -webkit-animation:fadeIn ease-in 1;
  -moz-animation:fadeIn ease-in 1;
  animation:fadeIn ease-in 1;

  -webkit-animation-fill-mode:forwards;
  -moz-animation-fill-mode:forwards;
  animation-fill-mode:forwards;

  -webkit-animation-duration:1s;
  -moz-animation-duration:1s;
  animation-duration:1s;
}

.fadeIn.first {
  -webkit-animation-delay: 0.4s;
  -moz-animation-delay: 0.4s;
  animation-delay: 0.4s;
}

.fadeIn.second {
  -webkit-animation-delay: 0.6s;
  -moz-animation-delay: 0.6s;
  animation-delay: 0.6s;
}

.fadeIn.third {
  -webkit-animation-delay: 0.8s;
  -moz-animation-delay: 0.8s;
  animation-delay: 0.8s;
}

.fadeIn.fourth {
```



```css
  -webkit-animation-delay: 1s;
  -moz-animation-delay: 1s;
  animation-delay: 1s;
}

/* Simple CSS3 Fade-in Animation */
.underlineHover:after {
  display: block;
  left: 0;
  bottom: -10px;
  width: 0;
  height: 2px;
  background-color: #ffe600;
  content: "";
  transition: width 0.2s;
}

.underlineHover:hover {
  color: #0d0d0d;
}

.underlineHover:hover:after{
  width: 100%;
}



/* OTHERS */

*:focus {
    outline: none;
}

#icon {
  width:20%;
}

* {
  box-sizing: border-box;
```

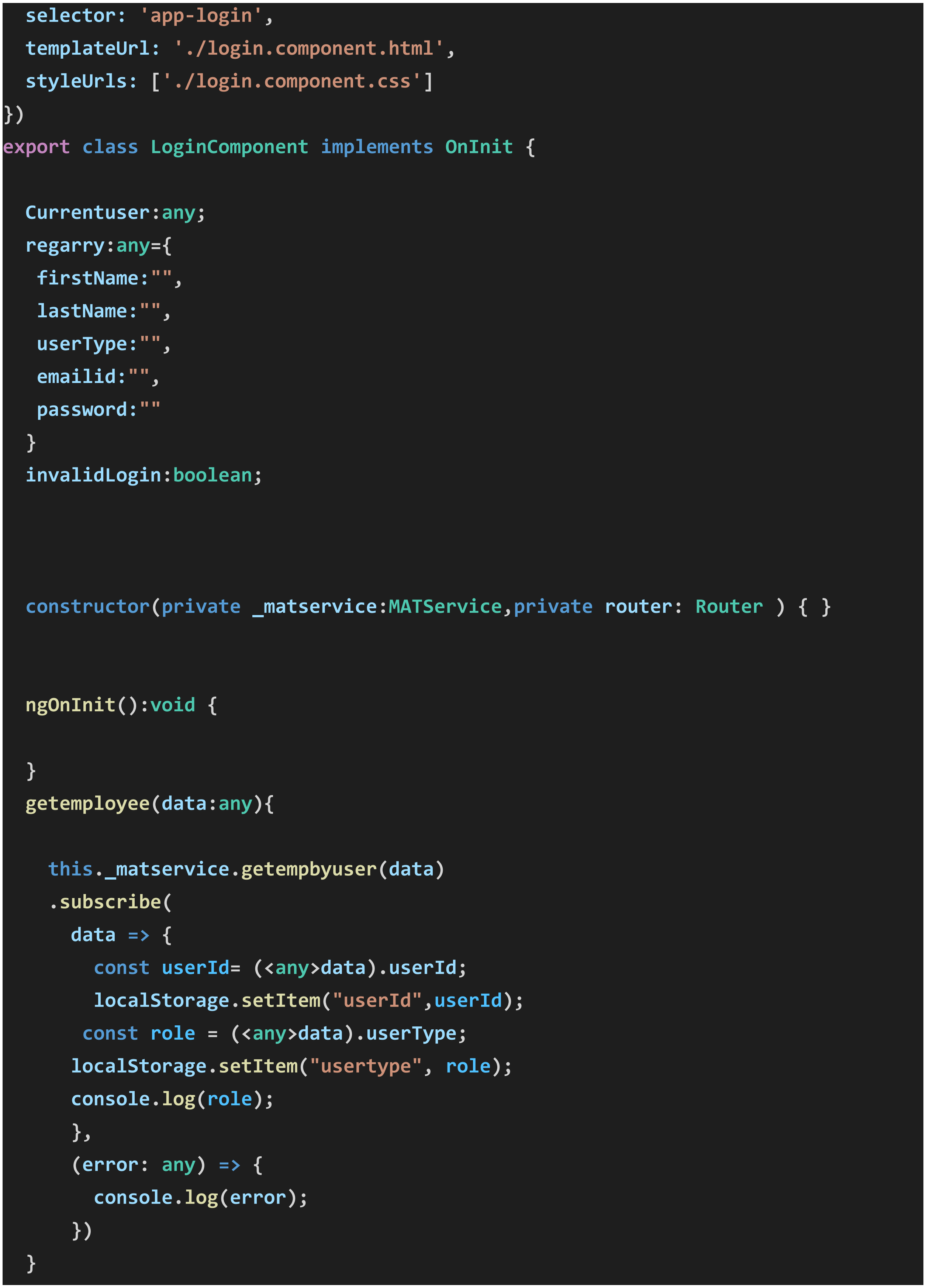
```
}
.navbar-fixed-top {
  height: 25px; /* Whatever you want. */
}
```

Component.html

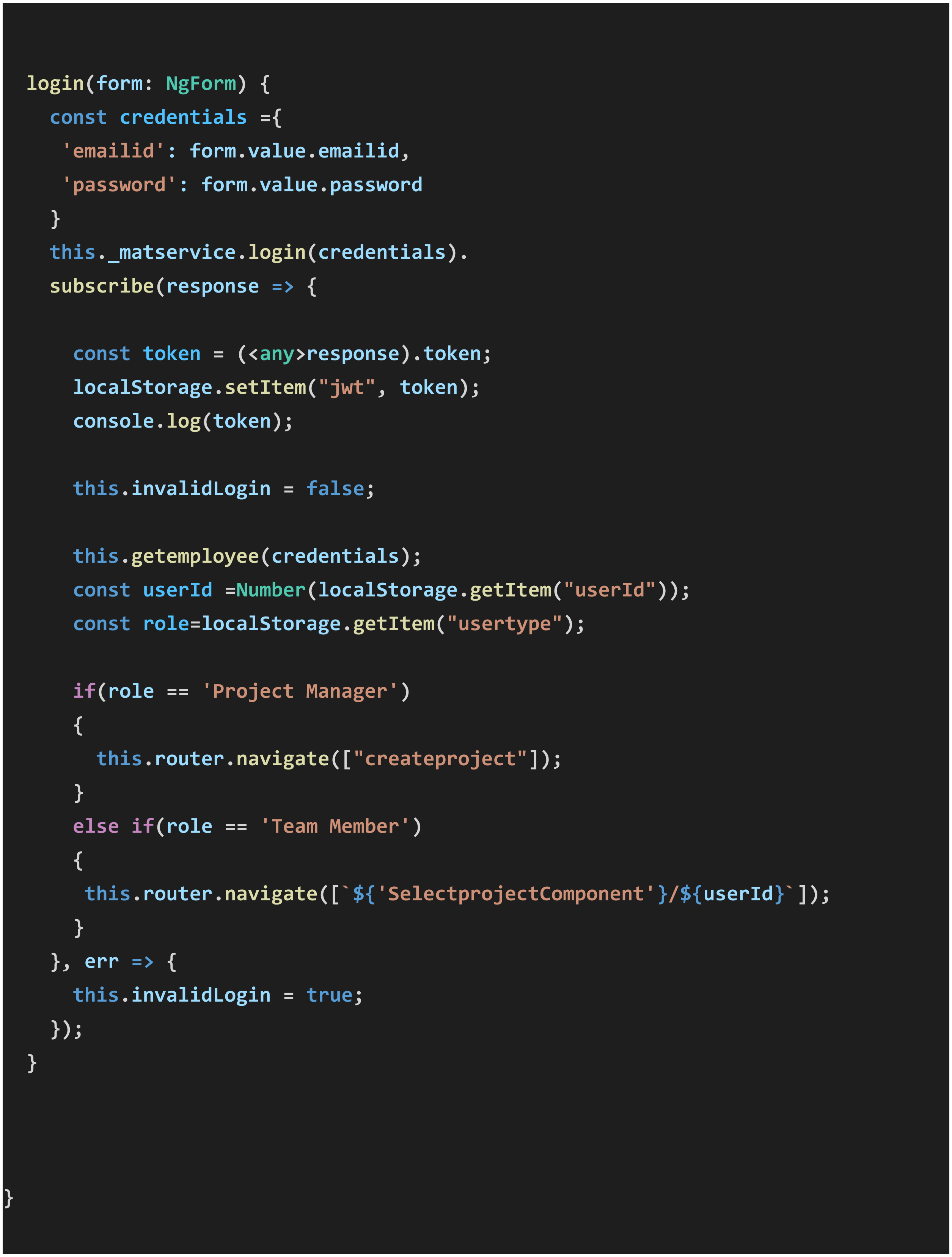
```



<nav class="navbar navbar-expand-lg  bg-dark" style="background-color: #333333;">
  <div class="container-fluid">
      <a href="home" class="navbar-brand"> <img src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
      <div class="navbar-nav ms-right"> <a href="Home" class="nav-item nav-link" style="color: #ffe600;">Home</a></div>
    </div>
</nav>

<div class="wrapper fadeInDown">
  <div id="formContent">
    <!-- Tabs Titles -->
    <h2 class="active"> Login</h2>


    <!-- Icon -->
    <div class="fadeIn first">
      <img src="http://cdn.onlinewebfonts.com/svg/img_311846.png" id="icon" alt="User Icon" />
    </div>

    <form class="form-signin" #loginForm="ngForm" (ngSubmit)="login(loginForm)">
      <div class="container-fluid">
        <!-- <h2 class="form-signin-heading">Login</h2> -->
        <div *ngIf="invalidLogin" class="alert alert-danger">Invalid username or password.</div>
        <br/>
        <input type="email" id="emailid" name="emailid" ngModel 
class="form-control" placeholder="User Name" required autofocus>
        <br/>

        <input type="password" id="password" name="password" ngModel 
class="form-control" placeholder="Password" required>
        <br/>
        <!-- <button class="btn btn-lg btn-primary btn-block" type="submit">Sign 
in</button> -->
      </div>

      <div class="text-cented">
        <input style="text-align: center;  background-color: #ffe600;" 
type="submit" class="fadeIn fourth" value="Log In">
      </div>
      <a href="/createuser" class="nav-link underlineHover">Create A  New 
Account</a>
    </form>




  </div>
</div>
```

Component.ts

```
import { Component, OnInit } from '@angular/core';
import {FormControl, FormGroup, NgForm, Validators} from '@angular/forms'
import { Router } from '@angular/router';
import { MATService } from '../mat.service';
import { Createuser } from '../createuser/createuser';
import{login} from './login'

import { ToastrService } from 'ngx-toastr';

@Component({
```



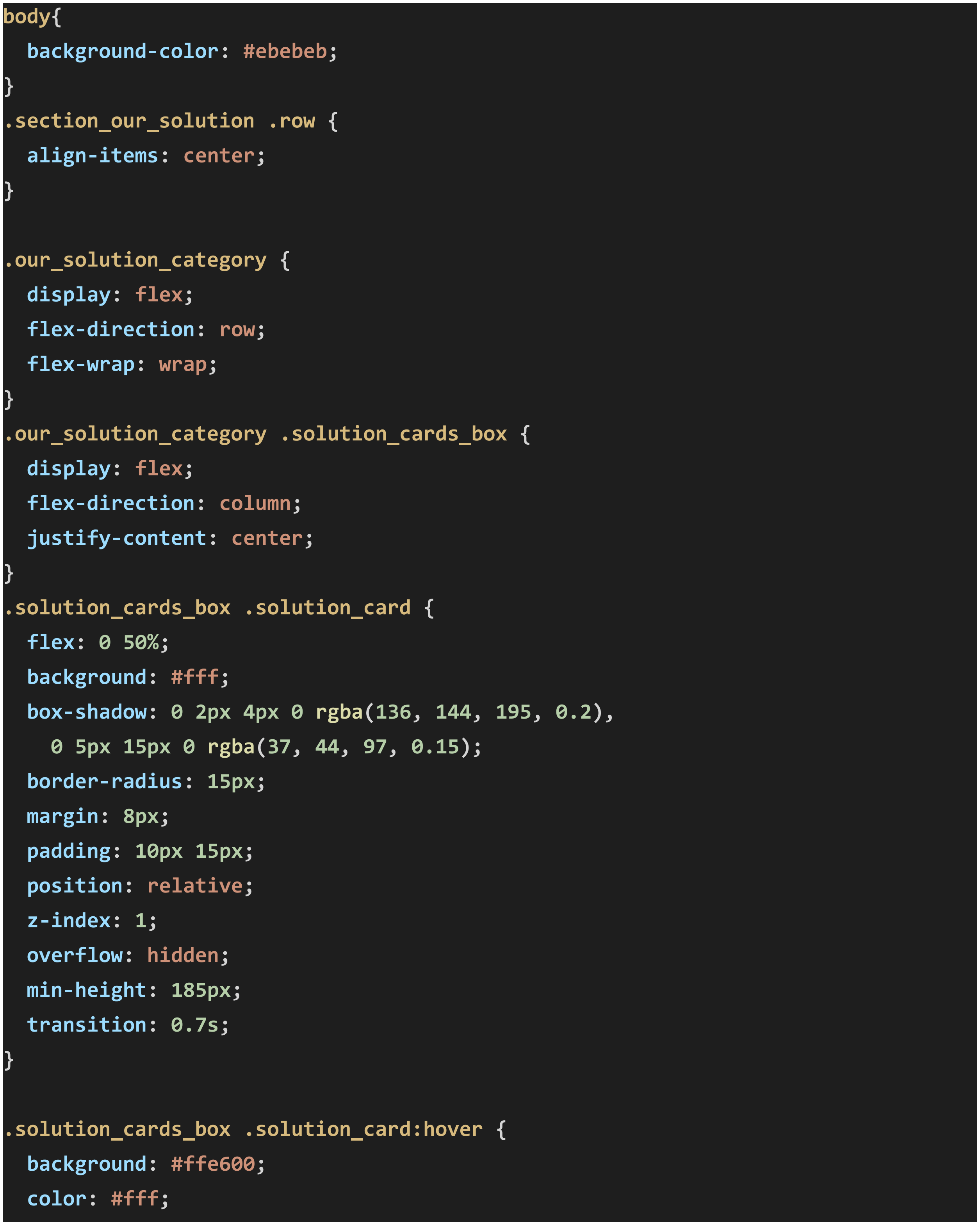
```
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent implements OnInit {

  Currentuser:any;
  regarry:any={
   firstName:"",
   lastName:"",
   userType:"",
   emailid:"",
   password:""
  }
  invalidLogin:boolean;



  constructor(private _matservice:MATService,private router: Router ) { }


  ngOnInit():void {

  }
  getemployee(data:any){

    this._matservice.getempbyuser(data)
    .subscribe(
      data => {
        const userId= (<any>data).userId;
        localStorage.setItem("userId",userId);
       const role = (<any>data).userType;
      localStorage.setItem("usertype", role);
      console.log(role);
      },
      (error: any) => {
        console.log(error);
      })
  }
```



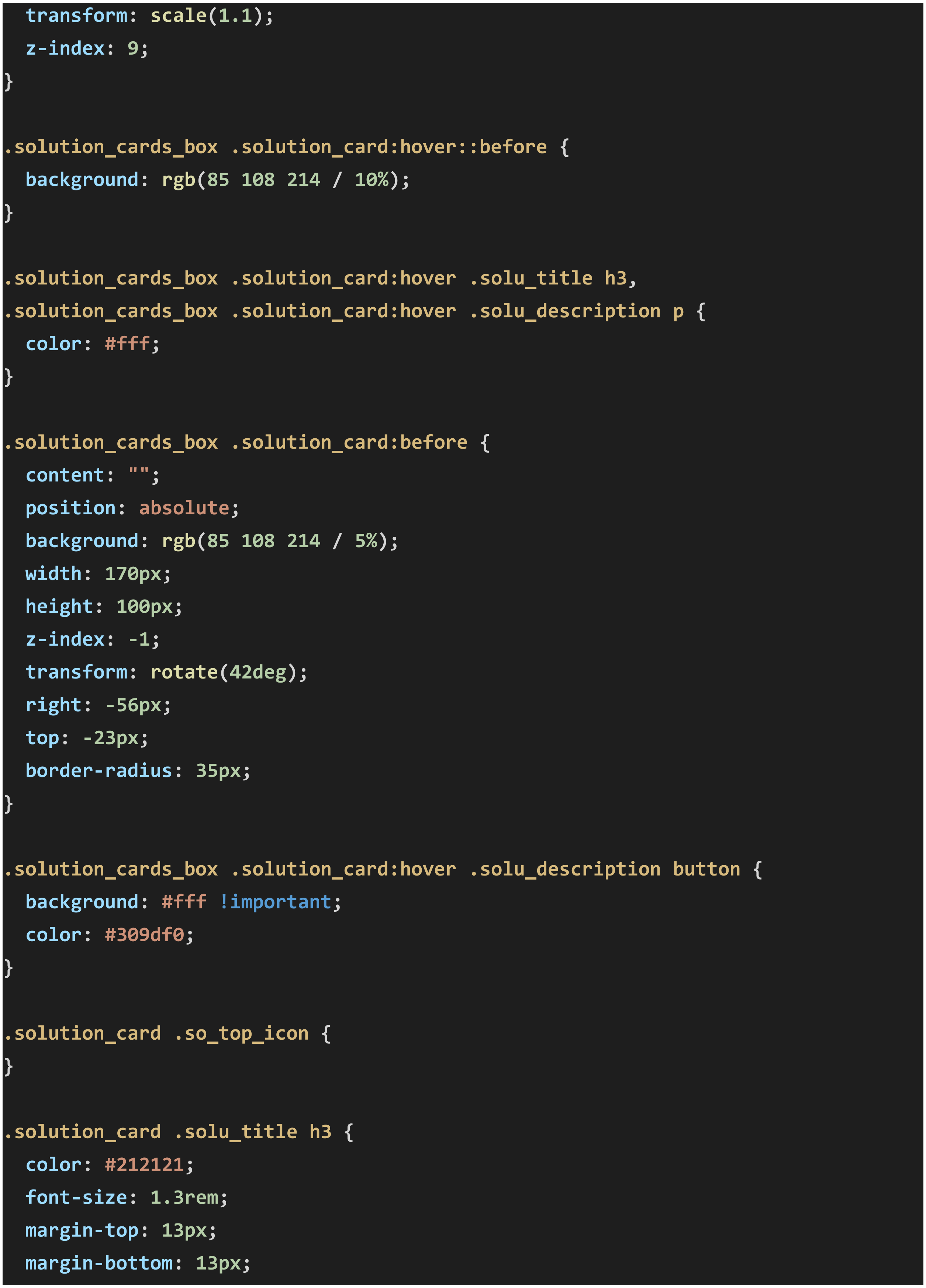
```
  login(form: NgForm) {
    const credentials ={
     'emailid': form.value.emailid,
     'password': form.value.password
    }
    this._matservice.login(credentials).
    subscribe(response => {

      const token = (<any>response).token;
      localStorage.setItem("jwt", token);
      console.log(token);

      this.invalidLogin = false;

      this.getemployee(credentials);
      const userId =Number(localStorage.getItem("userId"));
      const role=localStorage.getItem("usertype");

      if(role == 'Project Manager')
      {
        this.router.navigate(["createproject"]);
      }
      else if(role == 'Team Member')
      {
       this.router.navigate([`${'SelectprojectComponent'}/${userId}`]);
      }
    }, err => {
      this.invalidLogin = true;
    });
  }



}
```

Select Project

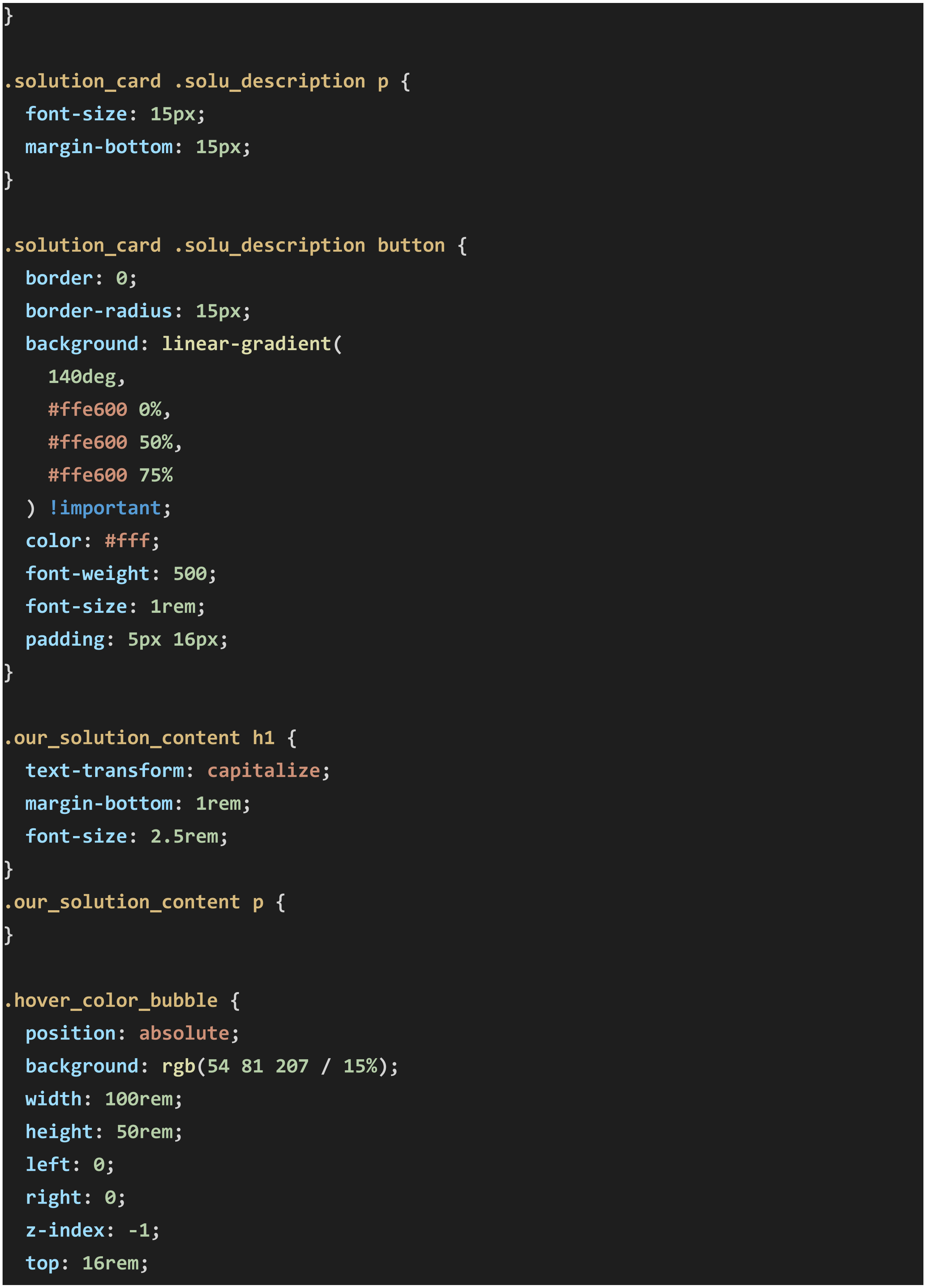
Component.css



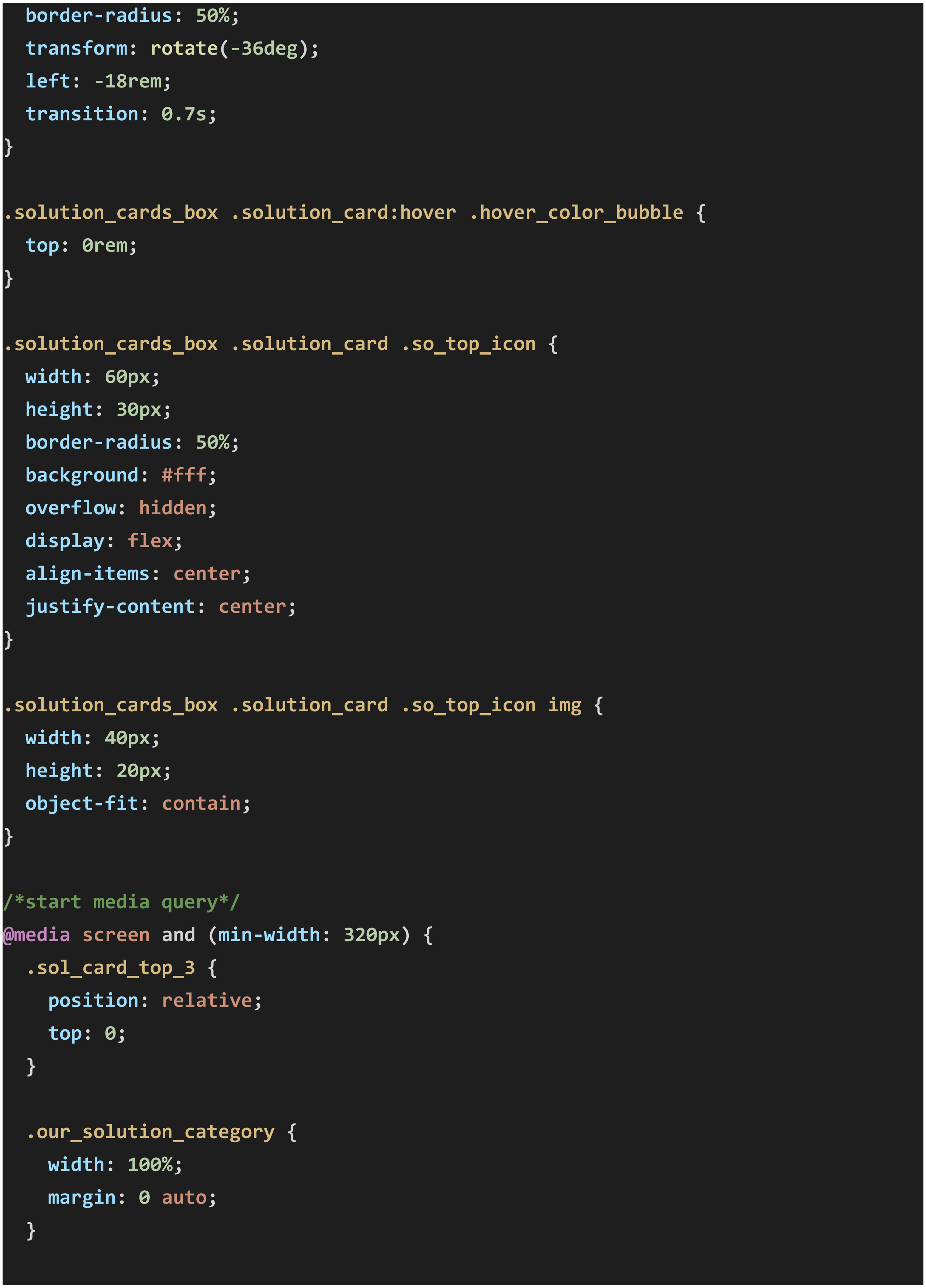
```css
body{
  background-color: #ebebeb;
}
.section_our_solution .row {
  align-items: center;
}

.our_solution_category {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
}
.our_solution_category .solution_cards_box {
  display: flex;
  flex-direction: column;
  justify-content: center;
}
.solution_cards_box .solution_card {
  flex: 0 50%;
  background: #fff;
  box-shadow: 0 2px 4px 0 rgba(136, 144, 195, 0.2),
    0 5px 15px 0 rgba(37, 44, 97, 0.15);
  border-radius: 15px;
  margin: 8px;
  padding: 10px 15px;
  position: relative;
  z-index: 1;
  overflow: hidden;
  min-height: 185px;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover {
  background: #ffe600;
  color: #fff;
```



```css
  transform: scale(1.1);
  z-index: 9;
}

.solution_cards_box .solution_card:hover::before {
  background: rgb(85 108 214 / 10%);
}

.solution_cards_box .solution_card:hover .solu_title h3,
.solution_cards_box .solution_card:hover .solu_description p {
  color: #fff;
}

.solution_cards_box .solution_card:before {
  content: "";
  position: absolute;
  background: rgb(85 108 214 / 5%);
  width: 170px;
  height: 100px;
  z-index: -1;
  transform: rotate(42deg);
  right: -56px;
  top: -23px;
  border-radius: 35px;
}

.solution_cards_box .solution_card:hover .solu_description button {
  background: #fff !important;
  color: #309df0;
}

.solution_card .so_top_icon {
}

.solution_card .solu_title h3 {
  color: #212121;
  font-size: 1.3rem;
  margin-top: 13px;
  margin-bottom: 13px;
```



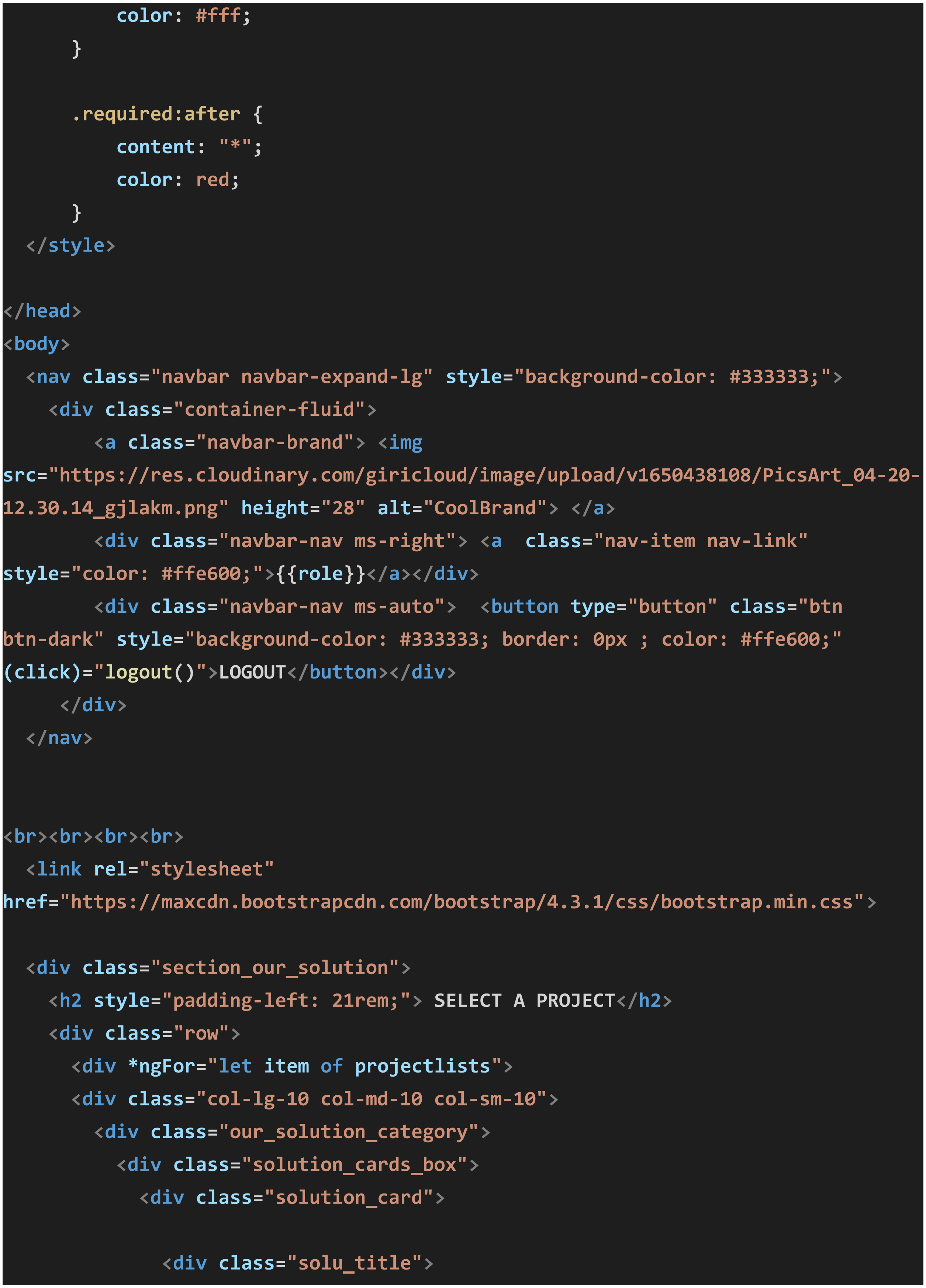
```css
}

.solution_card .solu_description p {
  font-size: 15px;
  margin-bottom: 15px;
}

.solution_card .solu_description button {
  border: 0;
  border-radius: 15px;
  background: linear-gradient(
    140deg,
    #ffe600 0%,
    #ffe600 50%,
    #ffe600 75%
  ) !important;
  color: #fff;
  font-weight: 500;
  font-size: 1rem;
  padding: 5px 16px;
}

.our_solution_content h1 {
  text-transform: capitalize;
  margin-bottom: 1rem;
  font-size: 2.5rem;
}
.our_solution_content p {
}

.hover_color_bubble {
  position: absolute;
  background: rgb(54 81 207 / 15%);
  width: 100rem;
  height: 50rem;
  left: 0;
  right: 0;
  z-index: -1;
  top: 16rem;
```



```css
  border-radius: 50%;
  transform: rotate(-36deg);
  left: -18rem;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover .hover_color_bubble {
  top: 0rem;
}

.solution_cards_box .solution_card .so_top_icon {
  width: 60px;
  height: 30px;
  border-radius: 50%;
  background: #fff;
  overflow: hidden;
  display: flex;
  align-items: center;
  justify-content: center;
}

.solution_cards_box .solution_card .so_top_icon img {
  width: 40px;
  height: 20px;
  object-fit: contain;
}

/*start media query*/
@media screen and (min-width: 320px) {
  .sol_card_top_3 {
    position: relative;
    top: 0;
  }

  .our_solution_category {
    width: 100%;
    margin: 0 auto;
  }
  .our_solution_category .solution_cards_box {
    flex: auto;
  }
}
@media only screen and (min-width: 768px) {
  .our_solution_category .solution_cards_box {
    flex: 1;
  }
}
@media only screen and (min-width: 1024px) {
  .sol_card_top_3 {
    position: relative;
    top: -3rem;
  }
  .our_solution_category {
    width: 50%;
    margin: 0 auto;
  }
}
```

Component.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.c
ss" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP
1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
```

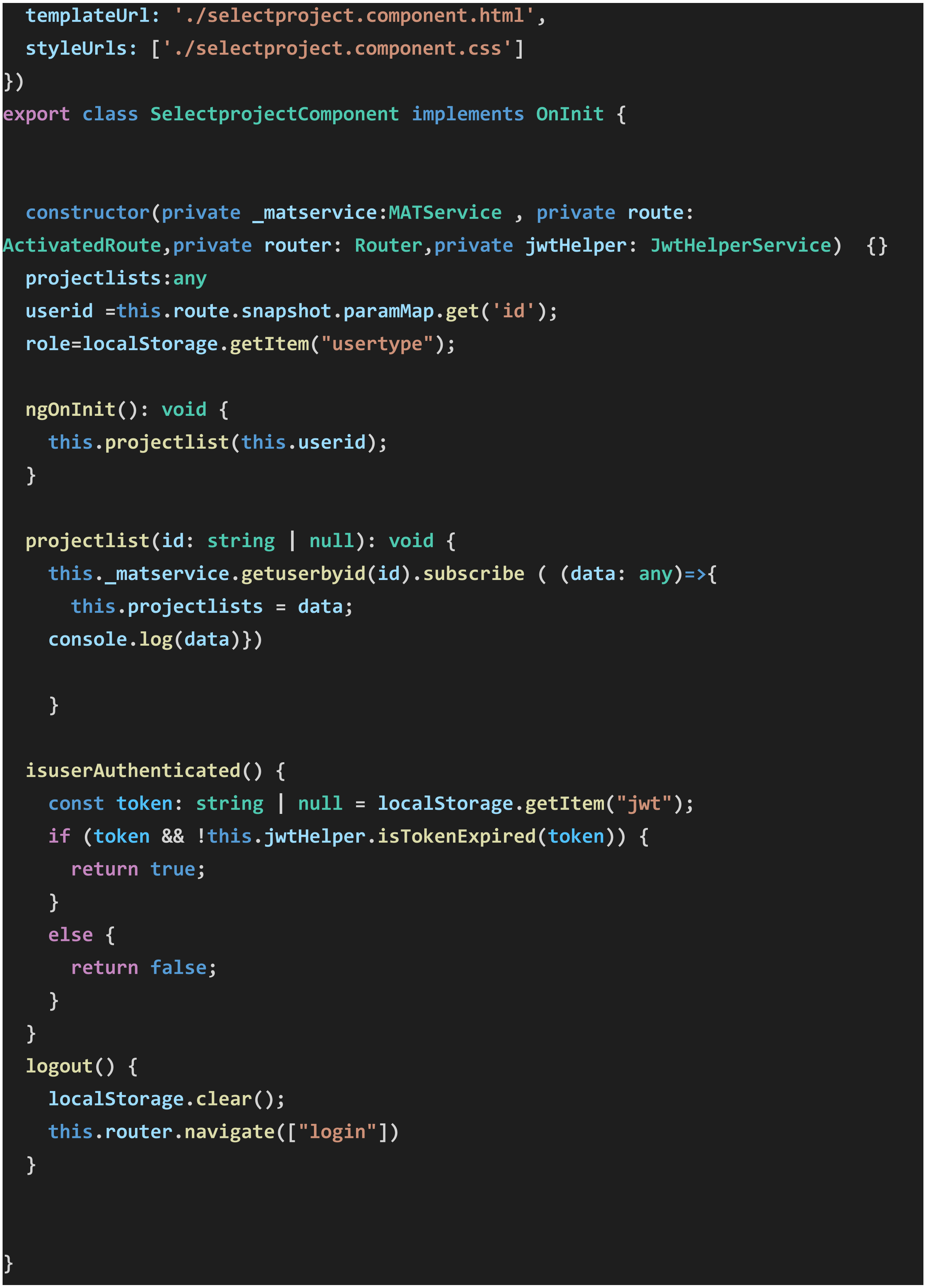


```
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>

</head>
<body>
  <nav class="navbar navbar-expand-lg" style="background-color: #333333;">
    <div class="container-fluid">
        <a class="navbar-brand"> <img src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
        <div class="navbar-nav ms-right"> <a  class="nav-item nav-link" style="color: #ffe600;">{{role}}</a></div>
        <div class="navbar-nav ms-auto">  <button type="button" class="btn btn-dark" style="background-color: #333333; border: 0px ; color: #ffe600;" (click)="logout()">LOGOUT</button></div>
     </div>
  </nav>


<br><br><br><br>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

  <div class="section_our_solution">
    <h2 style="padding-left: 21rem;"> SELECT A PROJECT</h2>
    <div class="row">
      <div *ngFor="let item of projectlists">
      <div class="col-lg-10 col-md-10 col-sm-10">
        <div class="our_solution_category">
          <div class="solution_cards_box">
            <div class="solution_card">

              <div class="solu_title">
```

```
                <h3>{{item.projectname}}</h3>
              </div>
              <div class="solu_description">
                <p>
                  {{item.projectDescription}}
                </p>
                <button type="button"
routerLink="/SelectsurveyComponent/{{item.projectId}}"class="read_more_btn">Open
Project</button>
                <!-- <a routerLink="/CreatesurveyComponent/{{item.projectId}}"
class="read_more_btn">Open Project</a> -->
              </div>
            </div>

      </div>
    </div>
  </div>
  </div>
  </div>
  </div>
          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.
min.js"
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroaf
W" crossorigin="anonymous"></script>

</body>
</html>
```

Component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { MATService } from '../mat.service';

@Component({
  selector: 'app-selectproject',
```

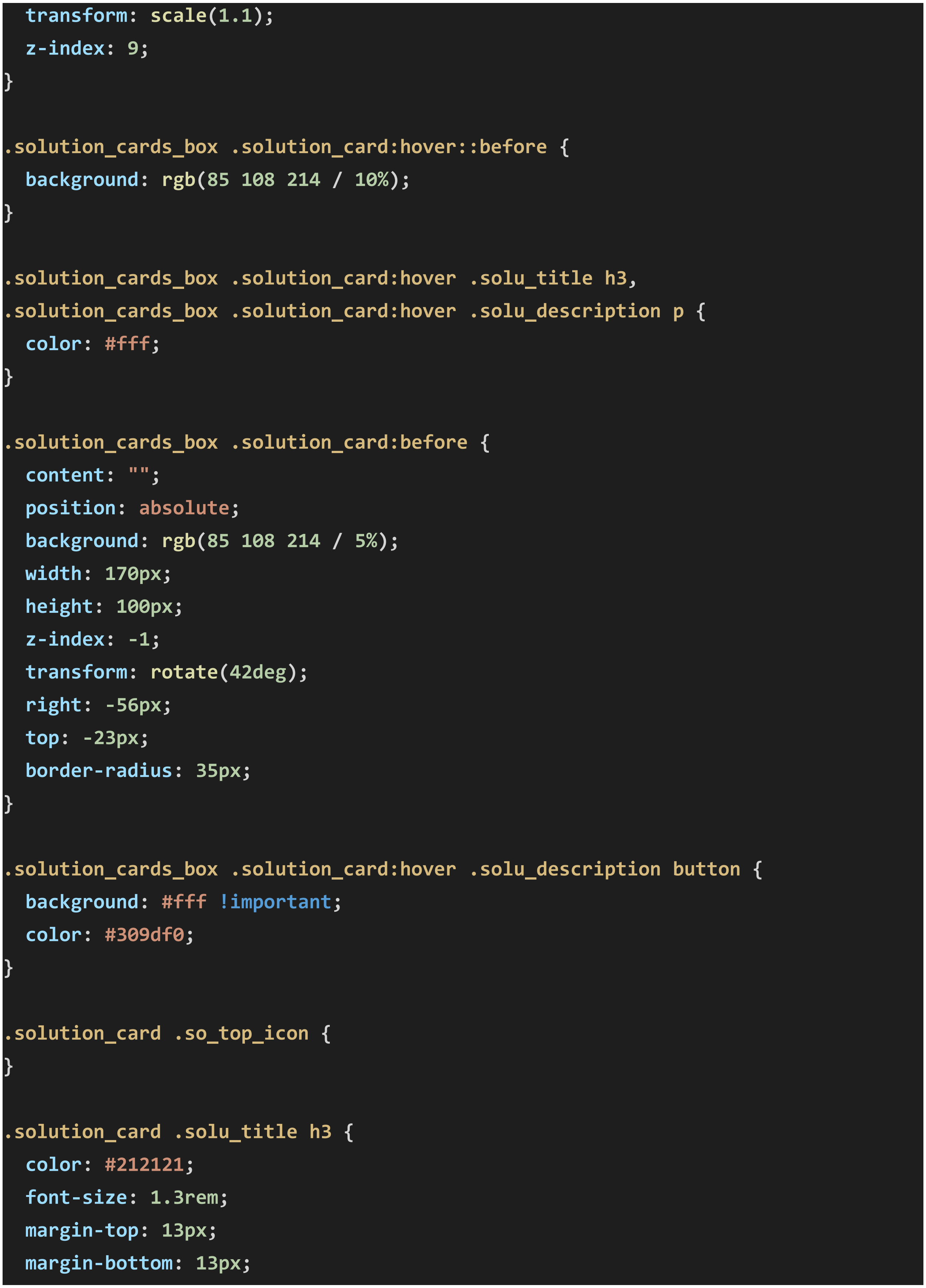


```
  templateUrl: './selectproject.component.html',
  styleUrls: ['./selectproject.component.css']
})
export class SelectprojectComponent implements OnInit {


  constructor(private _matservice:MATService , private route:
ActivatedRoute,private router: Router,private jwtHelper: JwtHelperService)  {}
  projectlists:any
  userid =this.route.snapshot.paramMap.get('id');
  role=localStorage.getItem("usertype");

  ngOnInit(): void {
    this.projectlist(this.userid);
  }

  projectlist(id: string | null): void {
    this._matservice.getuserbyid(id).subscribe ( (data: any)=>{
      this.projectlists = data;
    console.log(data)})

    }

  isuserAuthenticated() {
    const token: string | null = localStorage.getItem("jwt");
    if (token && !this.jwtHelper.isTokenExpired(token)) {
      return true;
    }
    else {
      return false;
    }
  }
  logout() {
    localStorage.clear();
    this.router.navigate(["login"])
  }



}
```

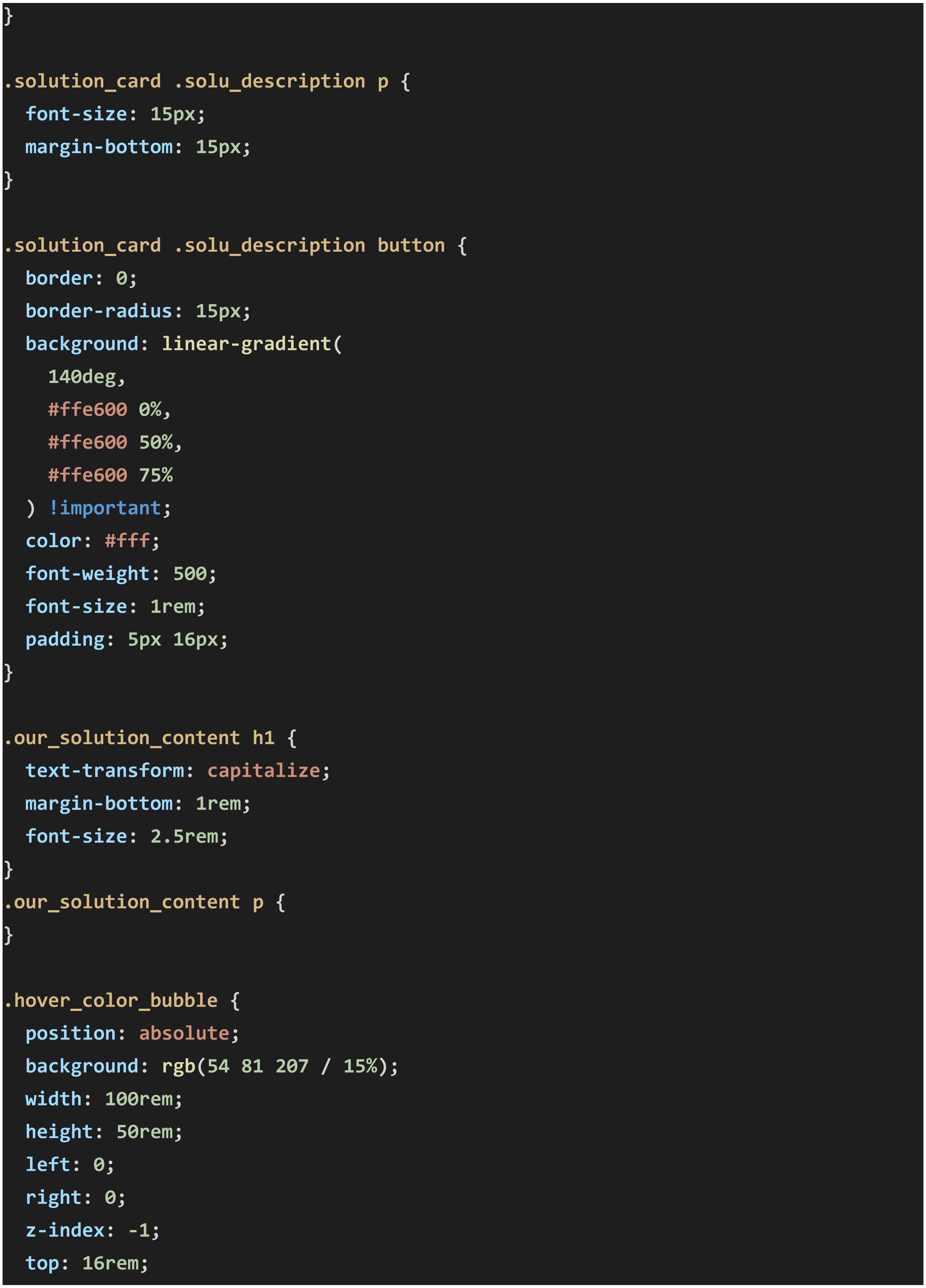
Select Survey

Component.css

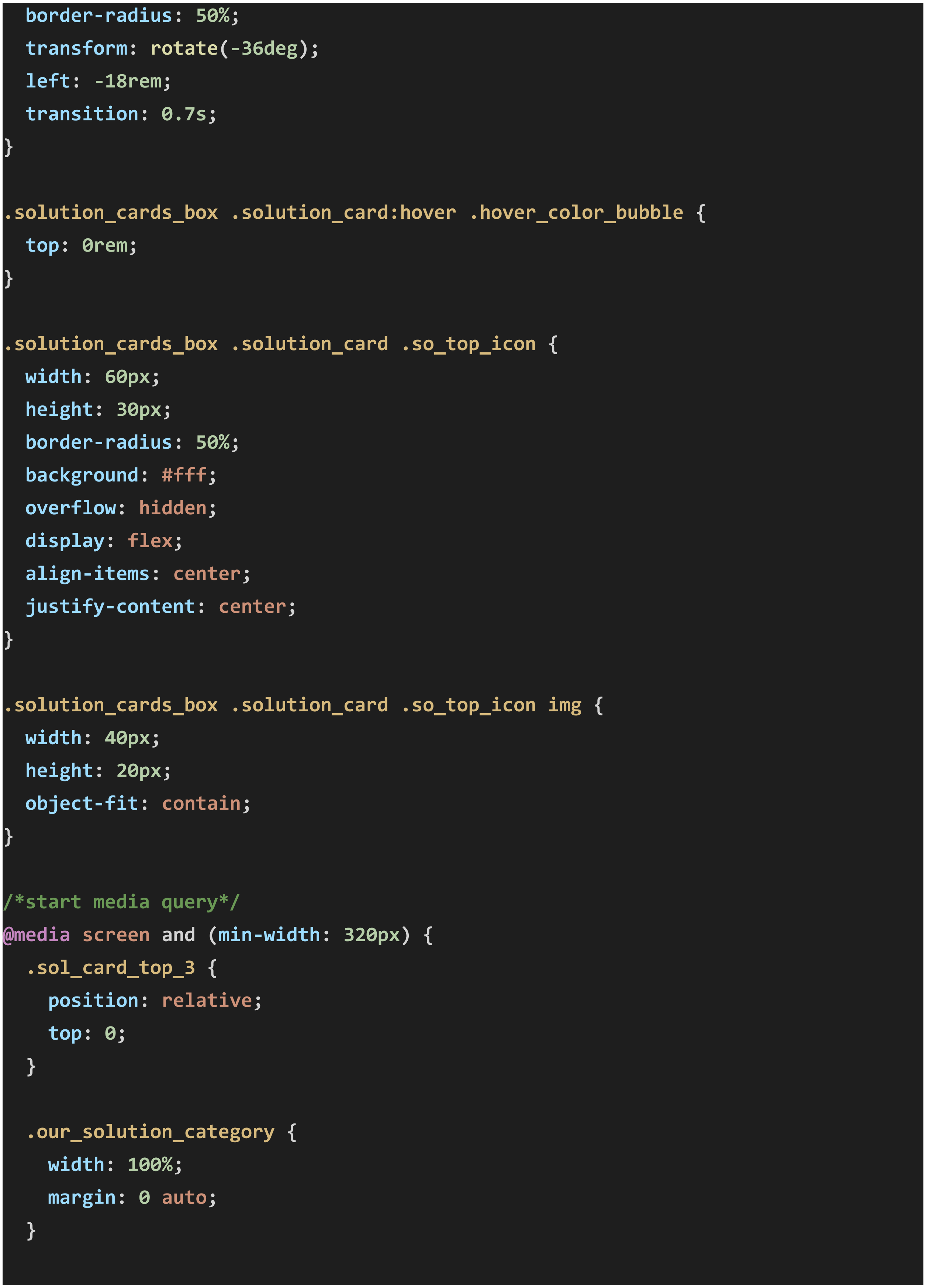
```
body{
  background-color: #ebebeb;
}
.section_our_solution .row {
  align-items: center;
}

.our_solution_category {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
}
.our_solution_category .solution_cards_box {
  display: flex;
  flex-direction: column;
  justify-content: center;
}
.solution_cards_box .solution_card {
  flex: 0 50%;
  background: #fff;
  box-shadow: 0 2px 4px 0 rgba(136, 144, 195, 0.2),
    0 5px 15px 0 rgba(37, 44, 97, 0.15);
  border-radius: 15px;
  margin: 8px;
  padding: 10px 15px;
  position: relative;
  z-index: 1;
  overflow: hidden;
  min-height: 178px;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover {
  background: #ffe600;
  color: #fff;
```



```css
  transform: scale(1.1);
  z-index: 9;
}

.solution_cards_box .solution_card:hover::before {
  background: rgb(85 108 214 / 10%);
}

.solution_cards_box .solution_card:hover .solu_title h3,
.solution_cards_box .solution_card:hover .solu_description p {
  color: #fff;
}

.solution_cards_box .solution_card:before {
  content: "";
  position: absolute;
  background: rgb(85 108 214 / 5%);
  width: 170px;
  height: 100px;
  z-index: -1;
  transform: rotate(42deg);
  right: -56px;
  top: -23px;
  border-radius: 35px;
}

.solution_cards_box .solution_card:hover .solu_description button {
  background: #fff !important;
  color: #309df0;
}

.solution_card .so_top_icon {
}

.solution_card .solu_title h3 {
  color: #212121;
  font-size: 1.3rem;
  margin-top: 13px;
  margin-bottom: 13px;
```



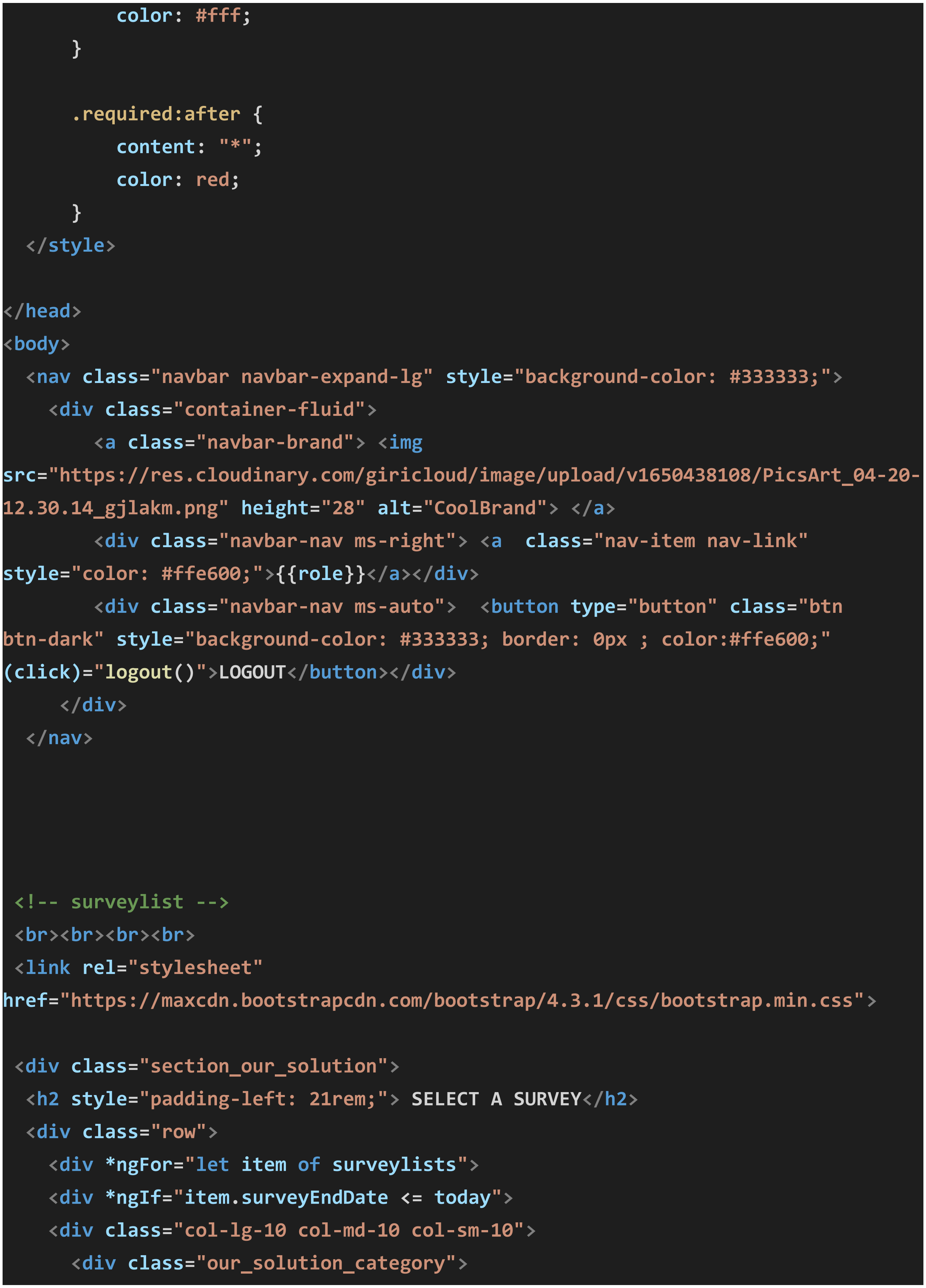
```
}

.solution_card .solu_description p {
  font-size: 15px;
  margin-bottom: 15px;
}

.solution_card .solu_description button {
  border: 0;
  border-radius: 15px;
  background: linear-gradient(
    140deg,
    #ffe600 0%,
    #ffe600 50%,
    #ffe600 75%
  ) !important;
  color: #fff;
  font-weight: 500;
  font-size: 1rem;
  padding: 5px 16px;
}

.our_solution_content h1 {
  text-transform: capitalize;
  margin-bottom: 1rem;
  font-size: 2.5rem;
}
.our_solution_content p {
}

.hover_color_bubble {
  position: absolute;
  background: rgb(54 81 207 / 15%);
  width: 100rem;
  height: 50rem;
  left: 0;
  right: 0;
  z-index: -1;
  top: 16rem;
```



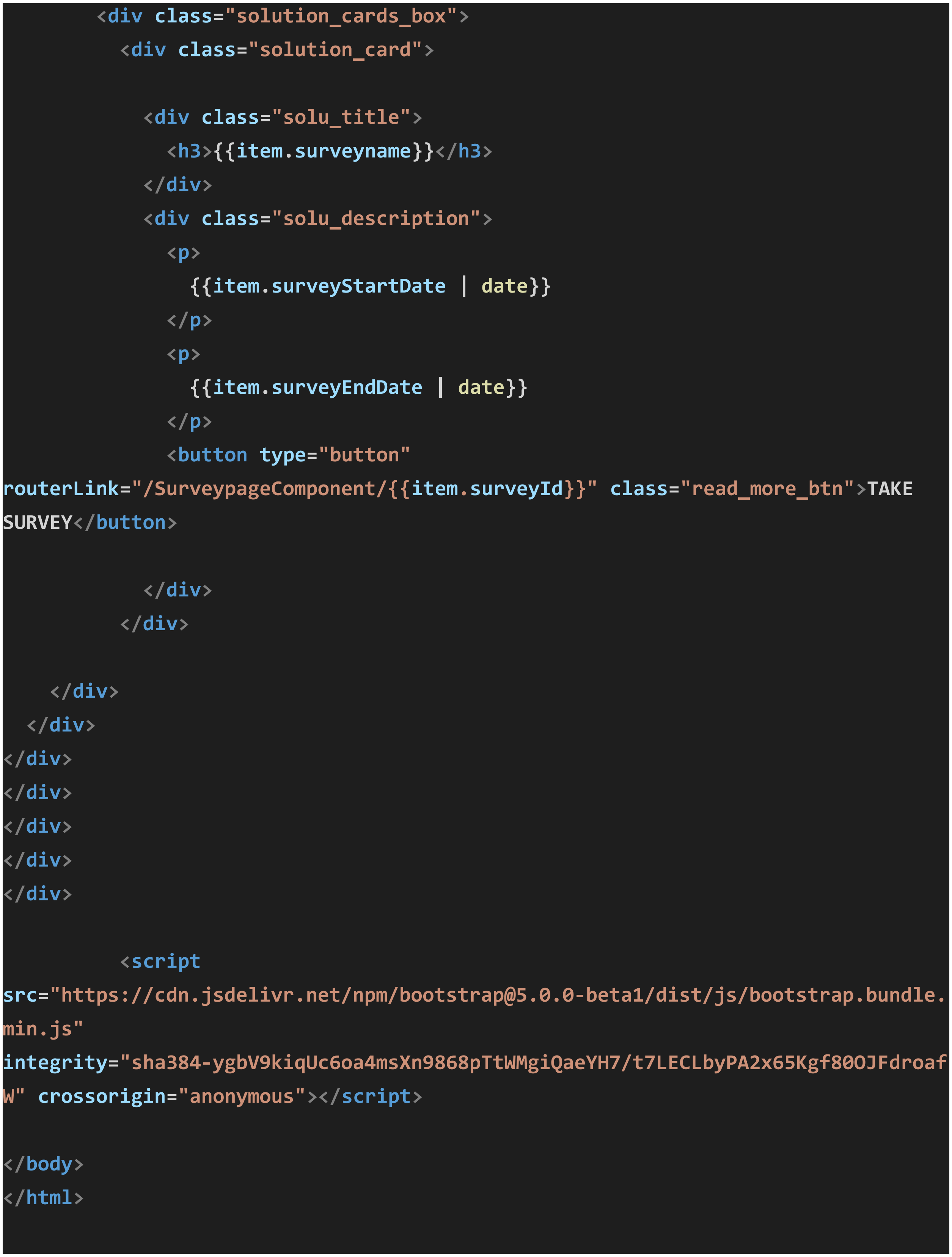
```css
  border-radius: 50%;
  transform: rotate(-36deg);
  left: -18rem;
  transition: 0.7s;
}

.solution_cards_box .solution_card:hover .hover_color_bubble {
  top: 0rem;
}

.solution_cards_box .solution_card .so_top_icon {
  width: 60px;
  height: 30px;
  border-radius: 50%;
  background: #fff;
  overflow: hidden;
  display: flex;
  align-items: center;
  justify-content: center;
}

.solution_cards_box .solution_card .so_top_icon img {
  width: 40px;
  height: 20px;
  object-fit: contain;
}

/*start media query*/
@media screen and (min-width: 320px) {
  .sol_card_top_3 {
    position: relative;
    top: 0;
  }

  .our_solution_category {
    width: 100%;
    margin: 0 auto;
  }
  .our_solution_category .solution_cards_box {
    flex: auto;
  }
}
@media only screen and (min-width: 768px) {
  .our_solution_category .solution_cards_box {
    flex: 1;
  }
}
@media only screen and (min-width: 1024px) {
  .sol_card_top_3 {
    position: relative;
    top: -3rem;
  }
  .our_solution_category {
    width: 50%;
    margin: 0 auto;
  }
}
```

Component.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.c
ss" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP
1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
```

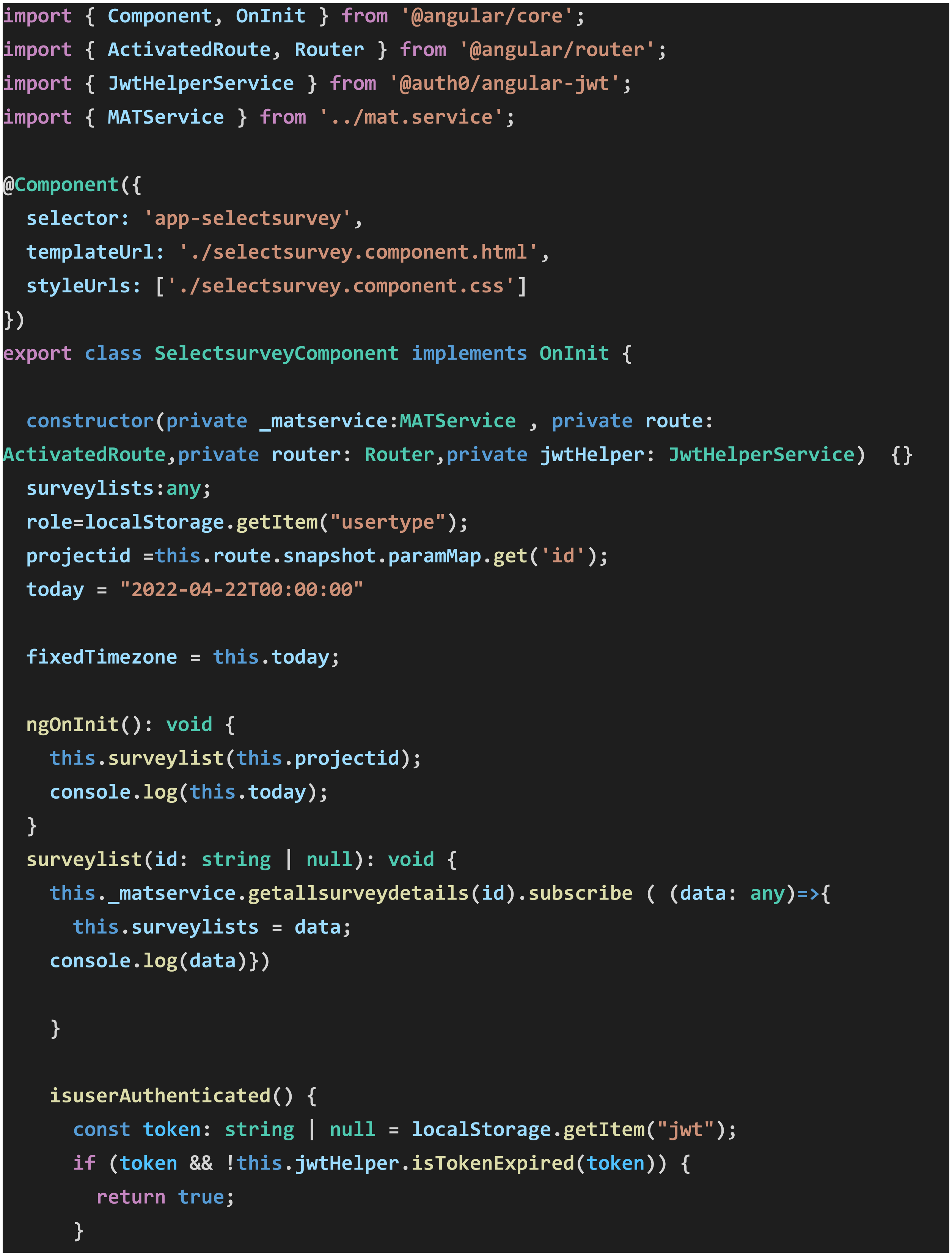


```
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>

</head>
<body>
  <nav class="navbar navbar-expand-lg" style="background-color: #333333;">
    <div class="container-fluid">
        <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
        <div class="navbar-nav ms-right"> <a  class="nav-item nav-link"
style="color: #ffe600;">{{role}}</a></div>
        <div class="navbar-nav ms-auto">  <button type="button" class="btn
btn-dark" style="background-color: #333333; border: 0px ; color:#ffe600;"
(click)="logout()">LOGOUT</button></div>
     </div>
  </nav>




 <!-- surveylist -->
 <br><br><br><br>
 <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

 <div class="section_our_solution">
  <h2 style="padding-left: 21rem;"> SELECT A SURVEY</h2>
  <div class="row">
    <div *ngFor="let item of surveylists">
    <div *ngIf="item.surveyEndDate <= today">
    <div class="col-lg-10 col-md-10 col-sm-10">
      <div class="our_solution_category">
```

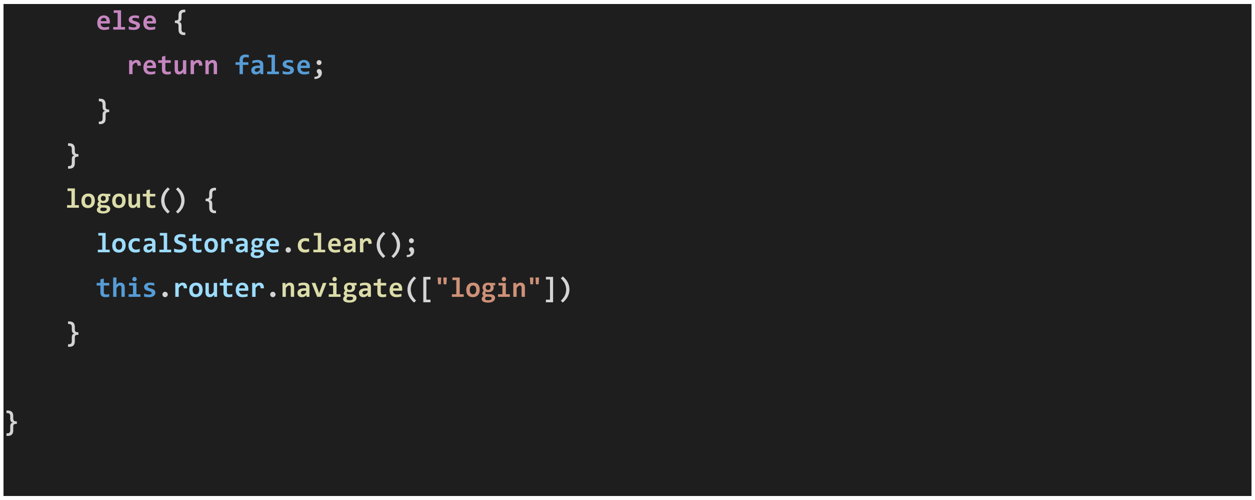


```
        <div class="solution_cards_box">
          <div class="solution_card">

            <div class="solu_title">
              <h3>{{item.surveyname}}</h3>
            </div>
            <div class="solu_description">
              <p>
                {{item.surveyStartDate | date}}
              </p>
              <p>
                {{item.surveyEndDate | date}}
              </p>
              <button type="button"
routerLink="/SurveypageComponent/{{item.surveyId}}" class="read_more_btn">TAKE
SURVEY</button>

            </div>
          </div>

    </div>
  </div>
</div>
</div>
</div>
</div>
</div>

          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.
min.js"
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroaf
W" crossorigin="anonymous"></script>

</body>
</html>
```

Component.ts



```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { MATService } from '../mat.service';

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

  constructor(private _matservice:MATService , private route:
ActivatedRoute,private router: Router,private jwtHelper: JwtHelperService)  {}
  surveylists:any;
  role=localStorage.getItem("usertype");
  projectid =this.route.snapshot.paramMap.get('id');
  today = "2022-04-22T00:00:00"

  fixedTimezone = this.today;

  ngOnInit(): void {
    this.surveylist(this.projectid);
    console.log(this.today);
  }
  surveylist(id: string | null): void {
    this._matservice.getallsurveydetails(id).subscribe ( (data: any)=>{
      this.surveylists = data;
    console.log(data)})

    }

    isuserAuthenticated() {
      const token: string | null = localStorage.getItem("jwt");
      if (token && !this.jwtHelper.isTokenExpired(token)) {
        return true;
      }
```



```
      else {
        return false;
      }
    }
    logout() {
      localStorage.clear();
      this.router.navigate(["login"])
    }

}
```

Survey Page

Component.css

```css
body {
  background-color: #eee
}

label.radio {
  cursor: pointer
}

label.radio input {
  position: absolute;
  top: 0;
  left: 0;
  visibility: hidden;
  pointer-events: none
}

label.radio span {
  padding: 4px 0px;
  border: 1px solid #ffe600;
  display: inline-block;
  color: #ffe600;
  width: 100px;
  text-align: center;
  border-radius: 3px;
  margin-top: 7px;
  text-transform: uppercase
}

label.radio input:checked+span {
  border-color:#ffe600;
  background-color: #ffe600;
  color: #fff
}

.ans {
  margin-left: 36px !important
}

.btn:focus {
  outline: 0 !important;
  box-shadow: none !important
}

.btn:active {
  outline: 0 !important;
  box-shadow: none !important
}
```

Component.html

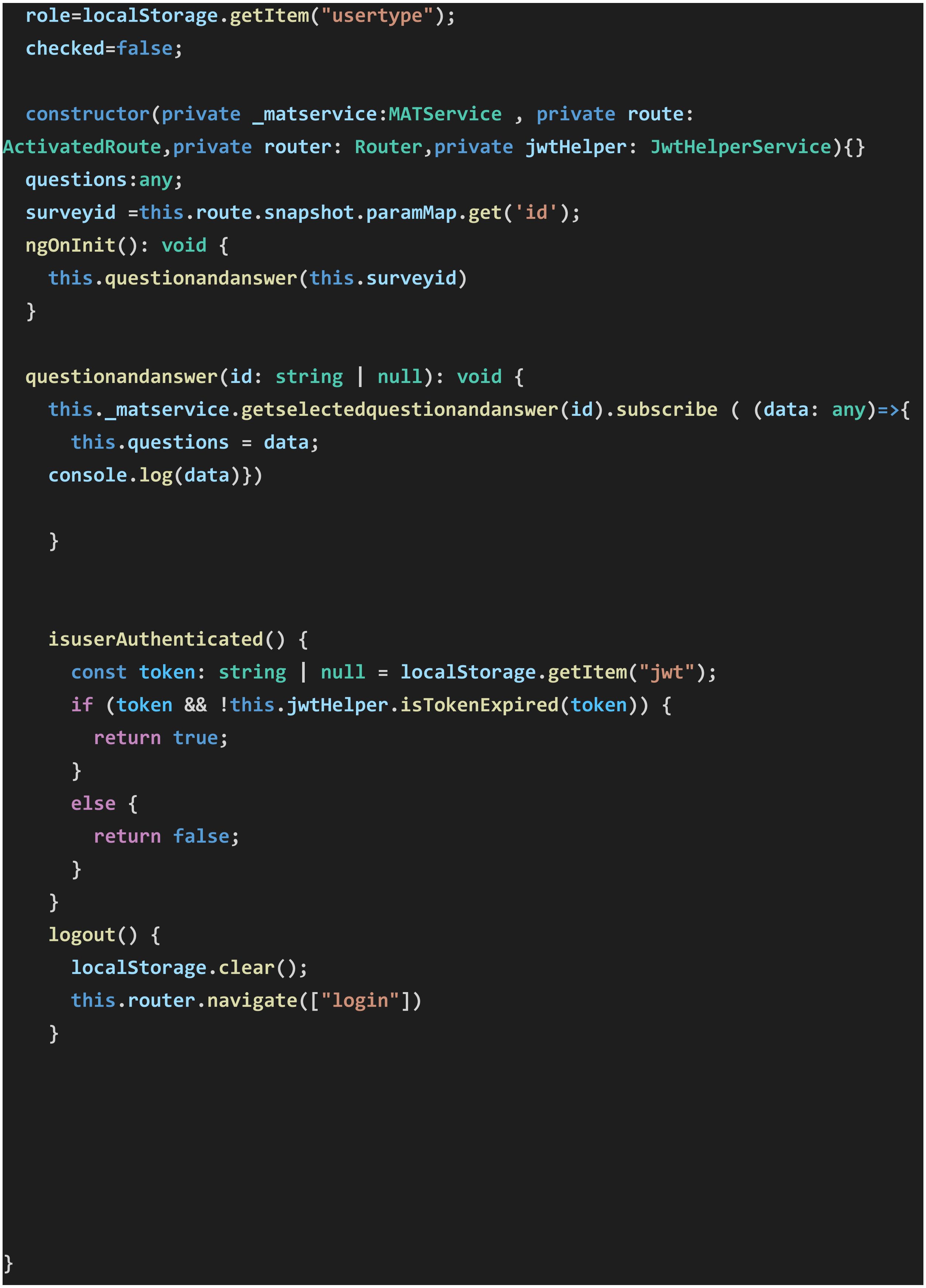
```
<!-- <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.c
ss" rel="stylesheet"
integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP
1" crossorigin="anonymous">

  <title>Bootstrap 5 Modal</title>
  <style>
      .modal-header {
          background: #F7941E;
          color: #fff;
      }

      .required:after {
          content: "*";
          color: red;
      }
  </style>

</head>
<body>










  <div class="container mb-5">

    <div class="row">
      <h1 style="text-align: center;">SURVEY</h1>
      <h3 style="text-align: left; color: red;">Answer According to your work
</h3>
      <div  *ngFor="let question of questions">
        <div class="col-12">
          <p class="fw-bold">{{question.question}}</p><br>
          <div  class="form-check"> <input class="form-check-input"
type="checkbox" (change)="checkCheckBoxvalue($event)">
            <label class="form-check-label" for="defaultCheck1">
              {{question.answer}}
            </label>
                  </div><br>
      </div>
      </div>



        <div class="col-12">
            <div class="d-flex justify-content-center"> <button type="button"
routerLink="/SurveyreportComponent/{{counter}}"class="btn btn-primary px-4 py-2
fw-bold" data-bs-toggle="modal" data-bs-target="#myModall"> Submit</button>
</div>
        </div>
    </div>
  </div>
```

```
          <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.
min.js"
integrity="sha384-ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf80OJFdroaf
W" crossorigin="anonymous"></script>

</body>
</html> -->
<nav class="navbar navbar-expand-lg" style="background-color: #333333;">
  <div class="container-fluid">
      <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
      <div class="navbar-nav ms-right"> <a  class="nav-item nav-link"
style="color: #ffe600;">{{role}}</a></div>
      <div class="navbar-nav ms-auto">  <button type="button" class="btn
btn-dark" style="background-color: #333333; border: 0px ; color:#ffe600;"
(click)="logout()">LOGOUT</button></div>
   </div>
</nav>
<br>
<h2 style="padding-left: 45rem;"> SURVEY</h2>
<h4 style="padding-left: 13rem; color: red;">If Yes, check the box 'Yes' else the
answer by default will be taken to be 'No'</h4>
<div class="container mt-5">
  <div class="d-flex justify-content-center row">
      <div class="col-md-10 col-lg-10">
          <div class="border">


              <div  *ngFor="let question of questions">
              <div class="question bg-white p-3 border-bottom">
                  <div class="d-flex flex-row align-items-center question-title">
                      <h3 class="text-danger">Q.</h3>
                      <h3 class="mt-1 ml-2">{{question.question}}</h3>
                  </div>
                  <div class="ans ml-2">
                      <label class="radio"> <input type="checkbox" name="brazil"
(change)="checkCheckBoxvalue($event)"> <span>{{question.answer}}</span>
```

```
                      </label>
                  </div>
              </div>
            </div>

          </div>
          <div class="d-flex flex-row justify-content-between align-items-center p-3 bg-white"><button class="btn btn-primary border-success align-items-center btn-success" style="background-color: #ffe600;" routerLink="/SurveyreportComponent/{{counter}}" type="button">SUBMIT<i class="fa fa-angle-right ml-2"></i></button></div>
      </div>

  </div>
</div>
```

Component.ts

```
import { Component, OnInit, ViewChild } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { MATService } from '../mat.service';
import { ChartData, ChartOptions } from 'chart.js';
import { JwtHelperService } from '@auth0/angular-jwt';




@Component({
  selector: 'app-surveypage',
  templateUrl: './surveypage.component.html',
  styleUrls: ['./surveypage.component.css']
})
export class SurveypageComponent implements OnInit {
  counter=0;
  checkCheckBoxvalue(event:any){
    this.counter++;
    console.log(this.counter)
  }
```



```
  role=localStorage.getItem("usertype");
  checked=false;

  constructor(private _matservice:MATService , private route:
ActivatedRoute,private router: Router,private jwtHelper: JwtHelperService){}
  questions:any;
  surveyid =this.route.snapshot.paramMap.get('id');
  ngOnInit(): void {
    this.questionandanswer(this.surveyid)
  }

  questionandanswer(id: string | null): void {
    this._matservice.getselectedquestionandanswer(id).subscribe ( (data: any)=>{
      this.questions = data;
    console.log(data)})

    }



    isuserAuthenticated() {
      const token: string | null = localStorage.getItem("jwt");
      if (token && !this.jwtHelper.isTokenExpired(token)) {
        return true;
      }
      else {
        return false;
      }
    }
    logout() {
      localStorage.clear();
      this.router.navigate(["login"])
    }






}
```

Survey Report

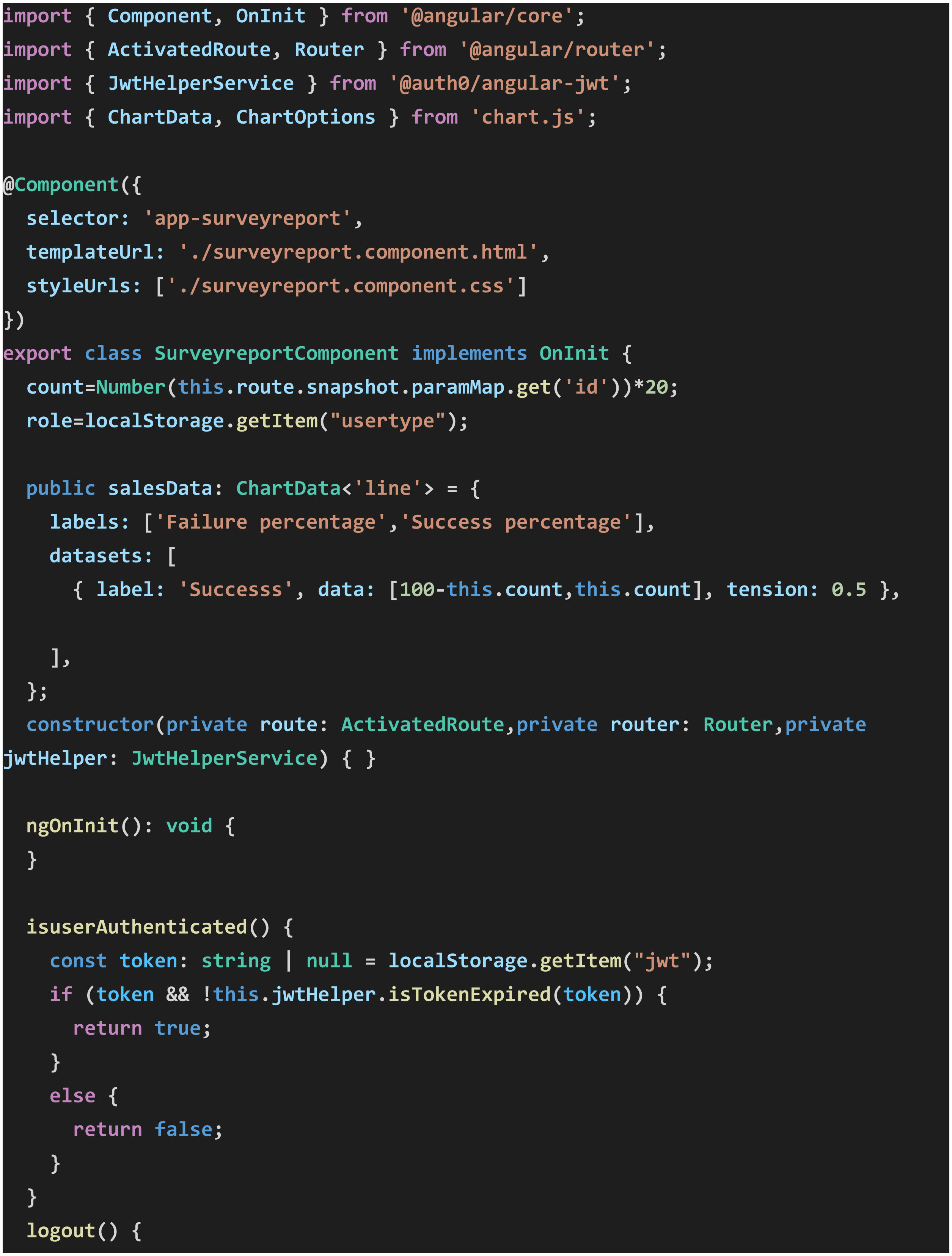
Component.css

```
.canvas-container {
  display: flex;
  align-items: center;
  justify-content: center;
  width:750px ;
  height:400px;
}
```

Component.html

```
<nav class="navbar navbar-expand-lg" style="background-color: #333333;">
  <div class="container-fluid">
      <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>
      <div class="navbar-nav ms-right"> <a  class="nav-item nav-link"
style="color: #ffe600;">{{role}}</a></div>
      <div class="navbar-nav ms-auto">  <button type="button" class="btn
btn-dark" style="background-color: #333333; border: 0px ; color:#ffe600;"
(click)="logout()">LOGOUT</button></div>
   </div>
</nav>
<br><br>
<h2 style="padding-left: 45rem;"> REPORT</h2>
<br><br><br>

<div  style="width: 500px; height: 200px;margin: auto; ">
<canvas
 baseChart
[data]="salesData"
[type]="'pie'">
</canvas>
</div>
```

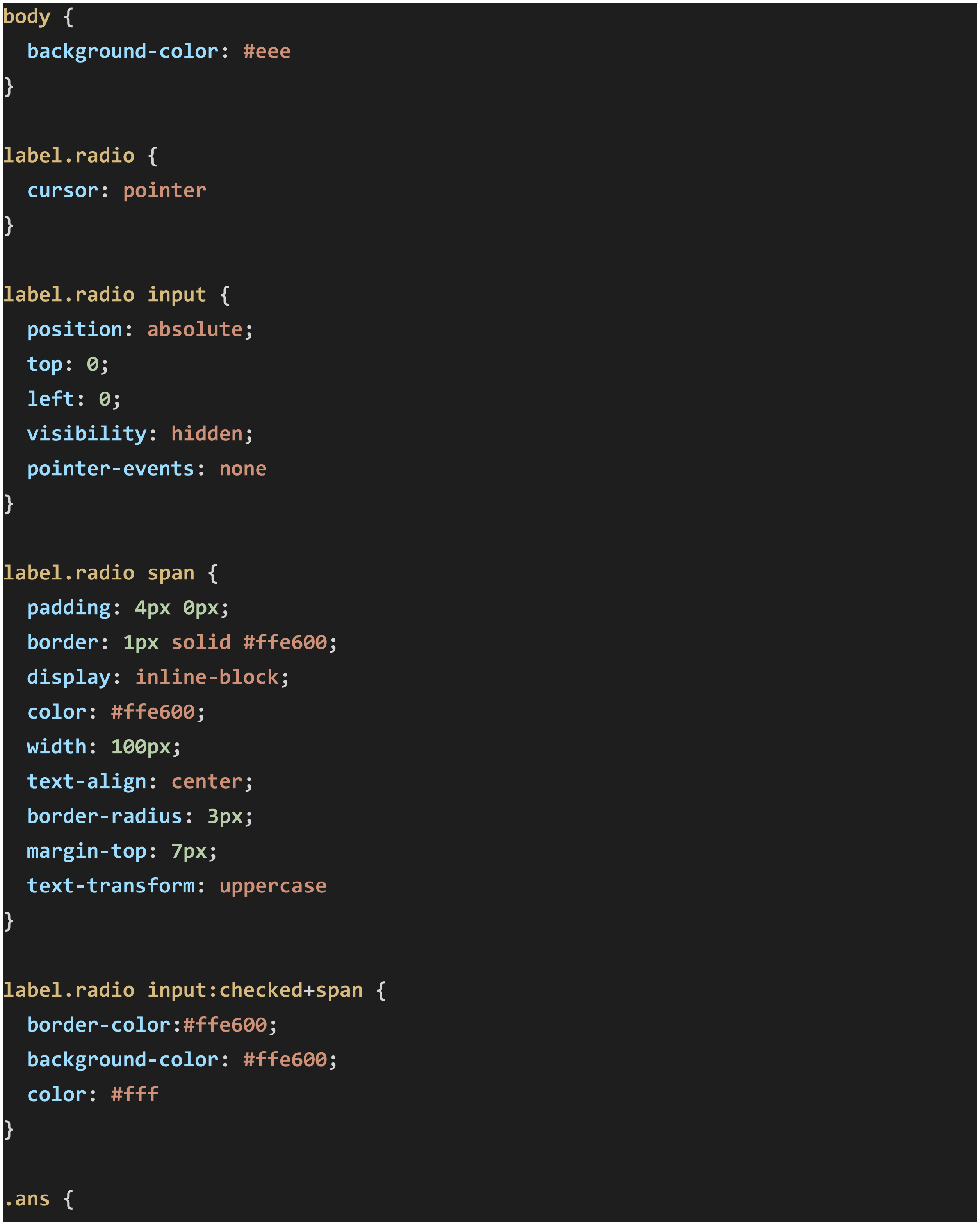
Component.ts



```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { JwtHelperService } from '@auth0/angular-jwt';
import { ChartData, ChartOptions } from 'chart.js';

@Component({
  selector: 'app-surveyreport',
  templateUrl: './surveyreport.component.html',
  styleUrls: ['./surveyreport.component.css']
})
export class SurveyreportComponent implements OnInit {
  count=Number(this.route.snapshot.paramMap.get('id'))*20;
  role=localStorage.getItem("usertype");

  public salesData: ChartData<'line'> = {
    labels: ['Failure percentage','Success percentage'],
    datasets: [
      { label: 'Successs', data: [100-this.count,this.count], tension: 0.5 },

    ],
  };
  constructor(private route: ActivatedRoute,private router: Router,private
jwtHelper: JwtHelperService) { }

  ngOnInit(): void {
  }

  isuserAuthenticated() {
    const token: string | null = localStorage.getItem("jwt");
    if (token && !this.jwtHelper.isTokenExpired(token)) {
      return true;
    }
    else {
      return false;
    }
  }
  logout() {
    localStorage.clear();
    this.router.navigate(["login"])
  }

}
```

View Question Details

Component.css



```
body {
  background-color: #eee
}

label.radio {
  cursor: pointer
}

label.radio input {
  position: absolute;
  top: 0;
  left: 0;
  visibility: hidden;
  pointer-events: none
}

label.radio span {
  padding: 4px 0px;
  border: 1px solid #ffe600;
  display: inline-block;
  color: #ffe600;
  width: 100px;
  text-align: center;
  border-radius: 3px;
  margin-top: 7px;
  text-transform: uppercase
}

label.radio input:checked+span {
  border-color:#ffe600;
  background-color: #ffe600;
  color: #fff
}

.ans {
  margin-left: 36px !important
}

.btn:focus {
  outline: 0 !important;
  box-shadow: none !important
}

.btn:active {
  outline: 0 !important;
  box-shadow: none !important
}
```

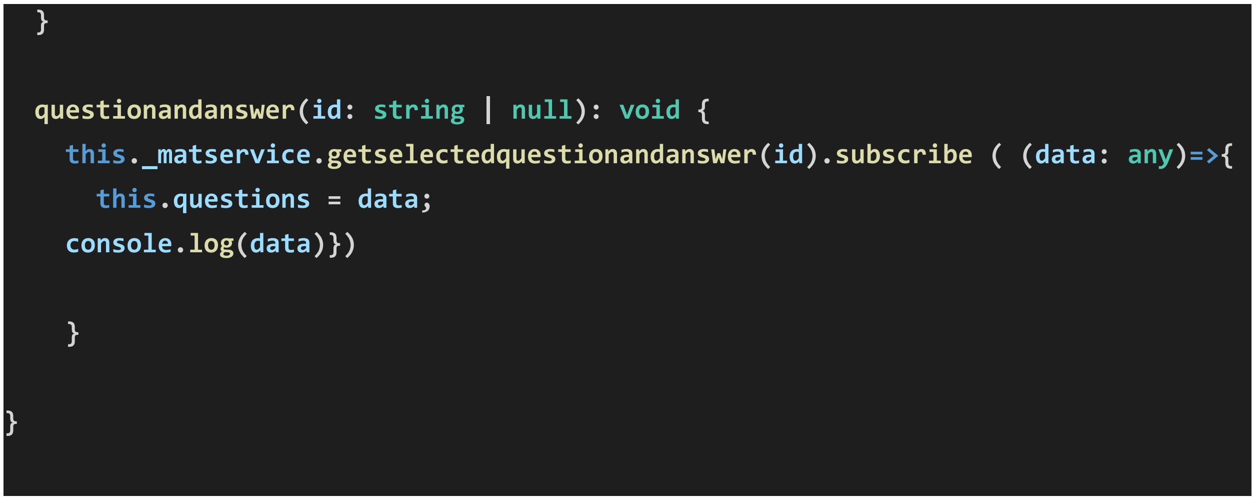
Component.html

```
<nav class="navbar navbar-expand-lg  bg-dark">
  <div class="container-fluid">
      <a class="navbar-brand"> <img
src="https://res.cloudinary.com/giricloud/image/upload/v1650438108/PicsArt_04-20-
12.30.14_gjlakm.png" height="28" alt="CoolBrand"> </a>

   </div>
</nav>
<br>
<h2 style="padding-left: 45rem;">REVIEW SURVEY</h2>
<div class="container mt-5">
  <div class="d-flex justify-content-center row">
      <div class="col-md-10 col-lg-10">
          <div class="border">

              <div  *ngFor="let question of questions">
              <div class="question bg-white p-3 border-bottom">
                  <div class="d-flex flex-row align-items-center question-title">
                      <h3 class="text-danger">Q.</h3>
                      <h3 class="mt-1 ml-2">{{question.question}}</h3>
                  </div>
                  <div class="ans ml-2">
```

```
                      <label class="radio"> <input type="checkbox" name="brazil">
<span>{{question.answer}}</span>
                      </label>
                  </div>
              </div>
            </div>

          </div>
          <div class="d-flex flex-row justify-content-between align-items-center
p-3 bg-white"><button class="btn btn-primary border-success align-items-center
btn-success" style="background-color: #ffe600;"  type="button">SUBMIT<i class="fa
fa-angle-right ml-2"></i></button></div>
      </div>

  </div>
</div>
```

Component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { MATService } from '../mat.service';

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

  constructor(private _matservice:MATService , private route:
ActivatedRoute,private router: Router)  {}
  questions:any;
  surveydetails:any
  surveyid =this.route.snapshot.paramMap.get('id');

  ngOnInit(): void {
    this.questionandanswer(this.surveyid)
```

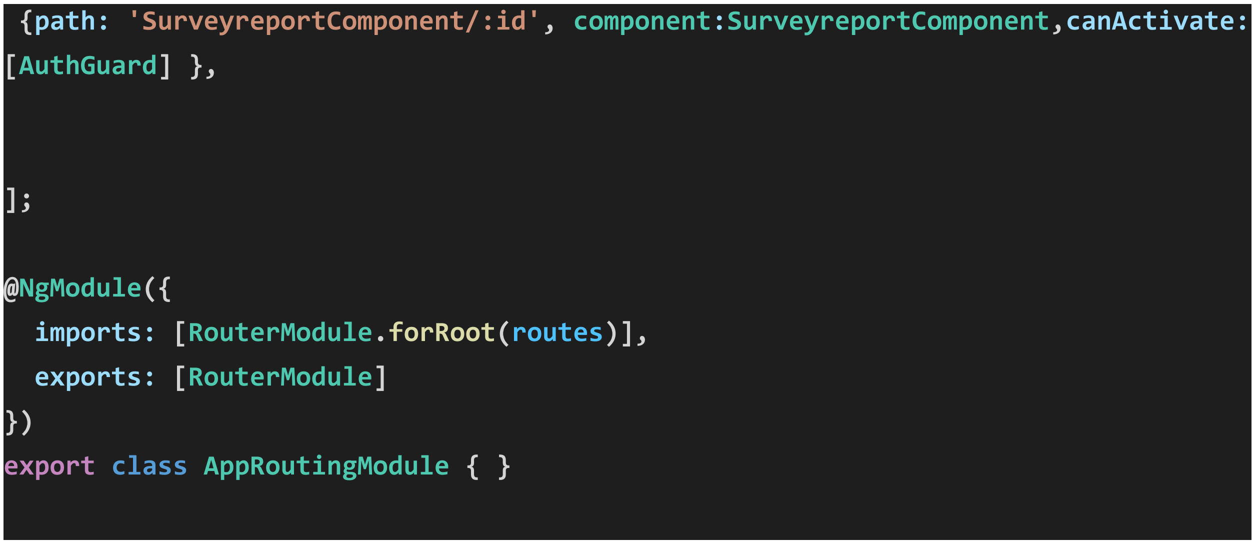


```
  }

  questionandanswer(id: string | null): void {
    this._matservice.getselectedquestionandanswer(id).subscribe ( (data: any)=>{
      this.questions = data;
    console.log(data)})

    }

}
```

App Routing

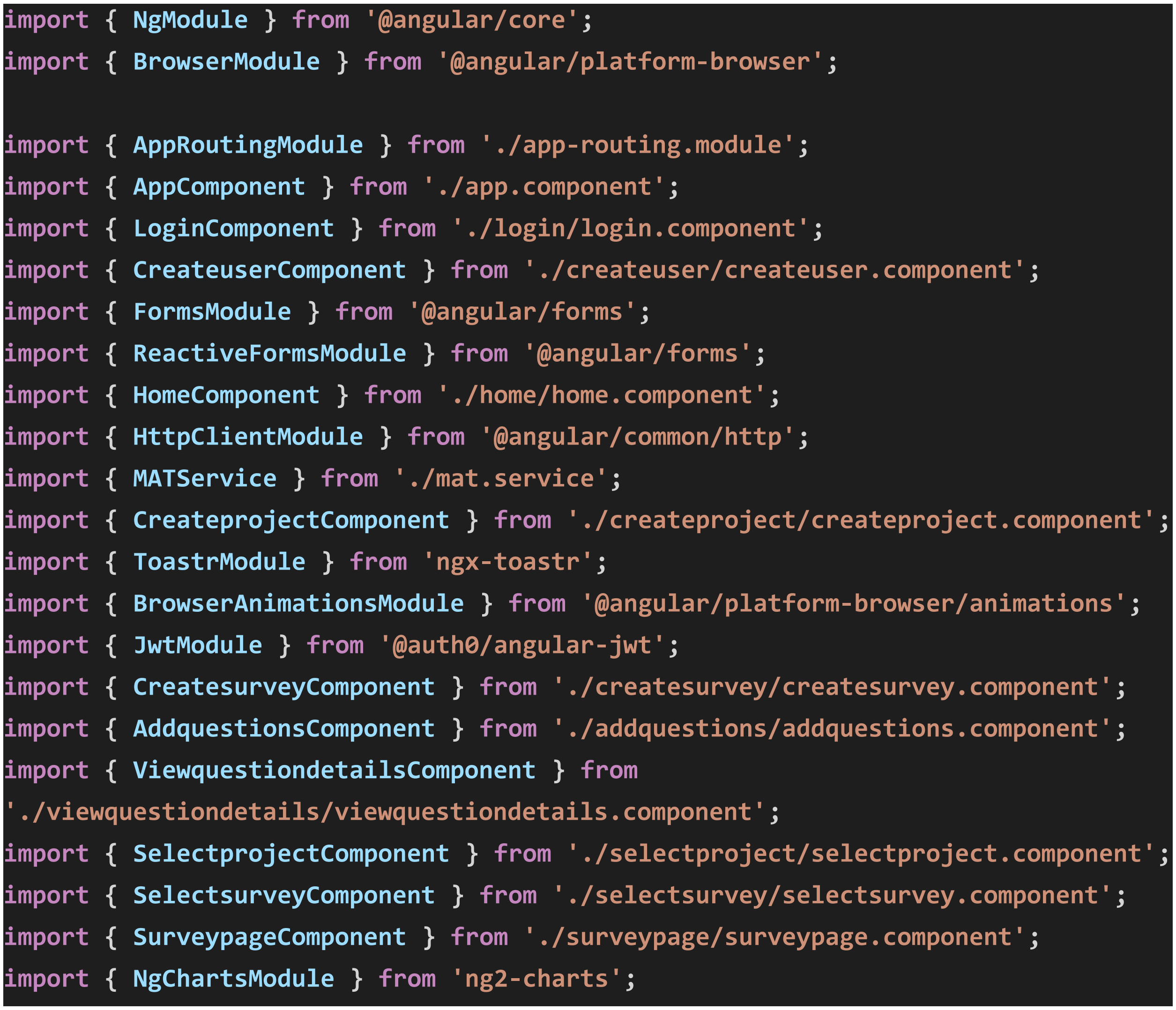
App-routing.module.ts

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { CreateuserComponent } from './createuser/createuser.component';
import { LoginComponent } from './login/login.component';
import { HomeComponent } from './home/home.component';
import { CreateprojectComponent } from './createproject/createproject.component';
import { CreatesurveyComponent } from './createsurvey/createsurvey.component';
import { AddquestionsComponent } from './addquestions/addquestions.component';
import { ViewquestiondetailsComponent } from
'./viewquestiondetails/viewquestiondetails.component';
import { SelectprojectComponent } from './selectproject/selectproject.component';
import { SelectsurveyComponent } from './selectsurvey/selectsurvey.component';
import { SurveypageComponent } from './surveypage/surveypage.component';
import { SurveyreportComponent } from './surveyreport/surveyreport.component';
import { AuthGuard } from './auth.guard';


const routes: Routes = [
{ path: '', redirectTo: 'Home', pathMatch: 'full' },
 {path: 'Home', component:HomeComponent},
 {path: 'login', component:LoginComponent},
 {path: 'createuser', component:CreateuserComponent},
 {path: 'createproject', component:CreateprojectComponent,canActivate:
[AuthGuard] },
 {path: 'CreatesurveyComponent/:id', component:CreatesurveyComponent
,canActivate: [AuthGuard] },
 {path: 'AddquestionsComponent/:id', component:AddquestionsComponent,canActivate:
[AuthGuard] },
 {path: 'ViewquestiondetailsComponent/:id',
component:ViewquestiondetailsComponent},
 {path: 'SelectprojectComponent/:id',
component:SelectprojectComponent,canActivate: [AuthGuard] },
 {path: 'SelectsurveyComponent/:id', component:SelectsurveyComponent,canActivate:
[AuthGuard] },
 {path: 'SurveypageComponent/:id', component:SurveypageComponent,canActivate:
[AuthGuard] },
```

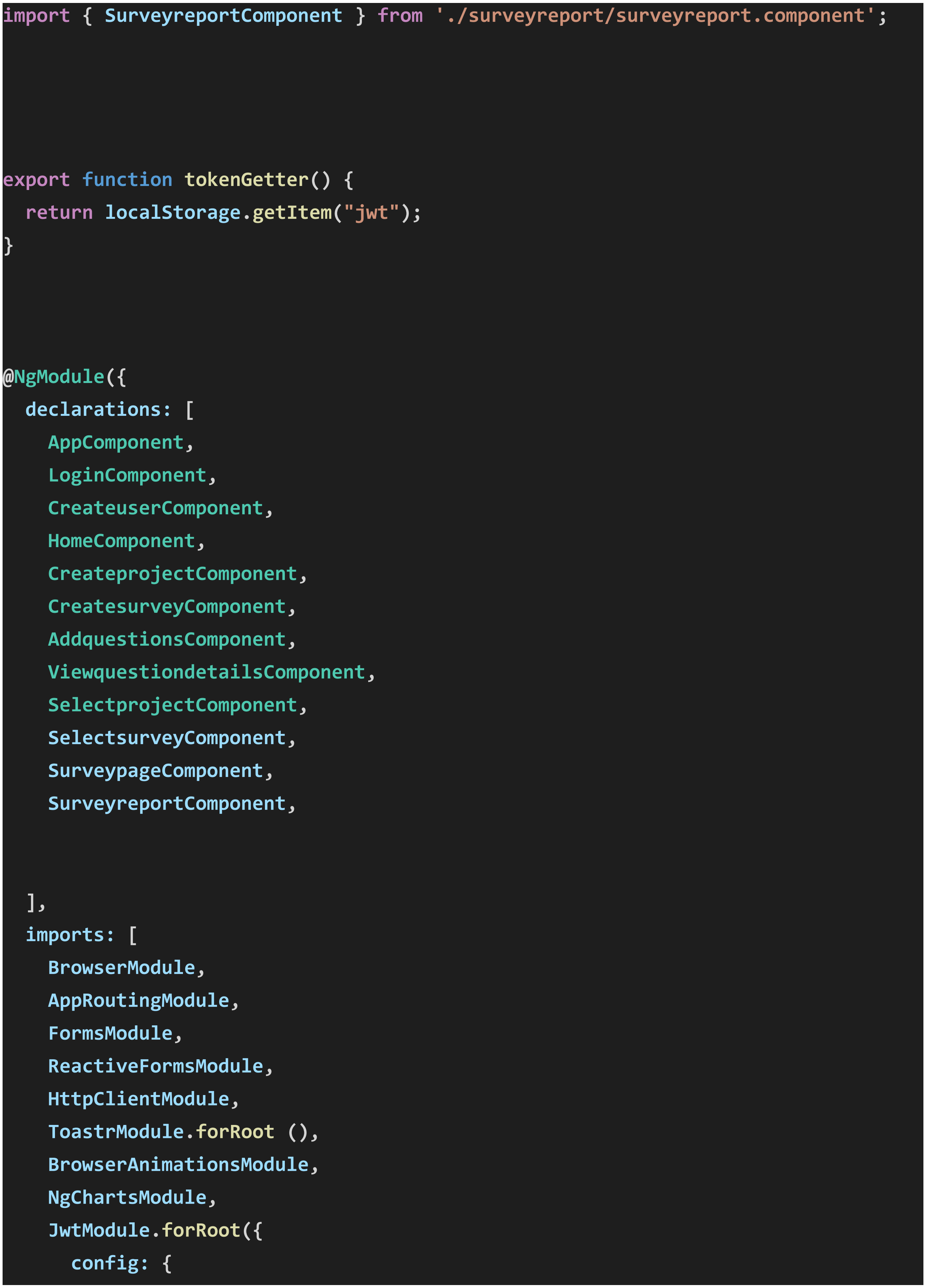


```
 {path: 'SurveyreportComponent/:id', component:SurveyreportComponent,canActivate:
[AuthGuard] },



];

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

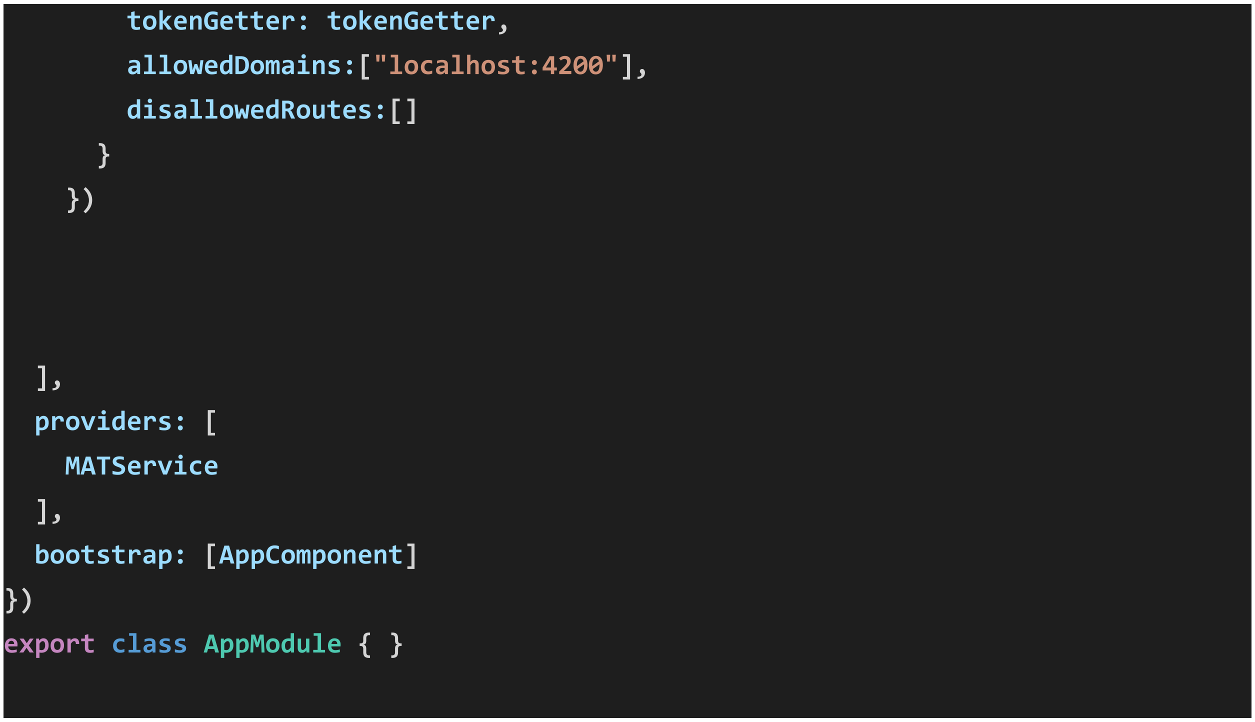
App.module.ts



```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { LoginComponent } from './login/login.component';
import { CreateuserComponent } from './createuser/createuser.component';
import { FormsModule } from '@angular/forms';
import { ReactiveFormsModule } from '@angular/forms';
import { HomeComponent } from './home/home.component';
import { HttpClientModule } from '@angular/common/http';
import { MATService } from './mat.service';
import { CreateprojectComponent } from './createproject/createproject.component';
import { ToastrModule } from 'ngx-toastr';
import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { JwtModule } from '@auth0/angular-jwt';
import { CreatesurveyComponent } from './createsurvey/createsurvey.component';
import { AddquestionsComponent } from './addquestions/addquestions.component';
import { ViewquestiondetailsComponent } from
'./viewquestiondetails/viewquestiondetails.component';
import { SelectprojectComponent } from './selectproject/selectproject.component';
import { SelectsurveyComponent } from './selectsurvey/selectsurvey.component';
import { SurveypageComponent } from './surveypage/surveypage.component';
import { NgChartsModule } from 'ng2-charts';
```



```
import { SurveyreportComponent } from './surveyreport/surveyreport.component';




export function tokenGetter() {
  return localStorage.getItem("jwt");
}



@NgModule({
  declarations: [
    AppComponent,
    LoginComponent,
    CreateuserComponent,
    HomeComponent,
    CreateprojectComponent,
    CreatesurveyComponent,
    AddquestionsComponent,
    ViewquestiondetailsComponent,
    SelectprojectComponent,
    SelectsurveyComponent,
    SurveypageComponent,
    SurveyreportComponent,


  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    FormsModule,
    ReactiveFormsModule,
    HttpClientModule,
    ToastrModule.forRoot (),
    BrowserAnimationsModule,
    NgChartsModule,
    JwtModule.forRoot({
      config: {
```

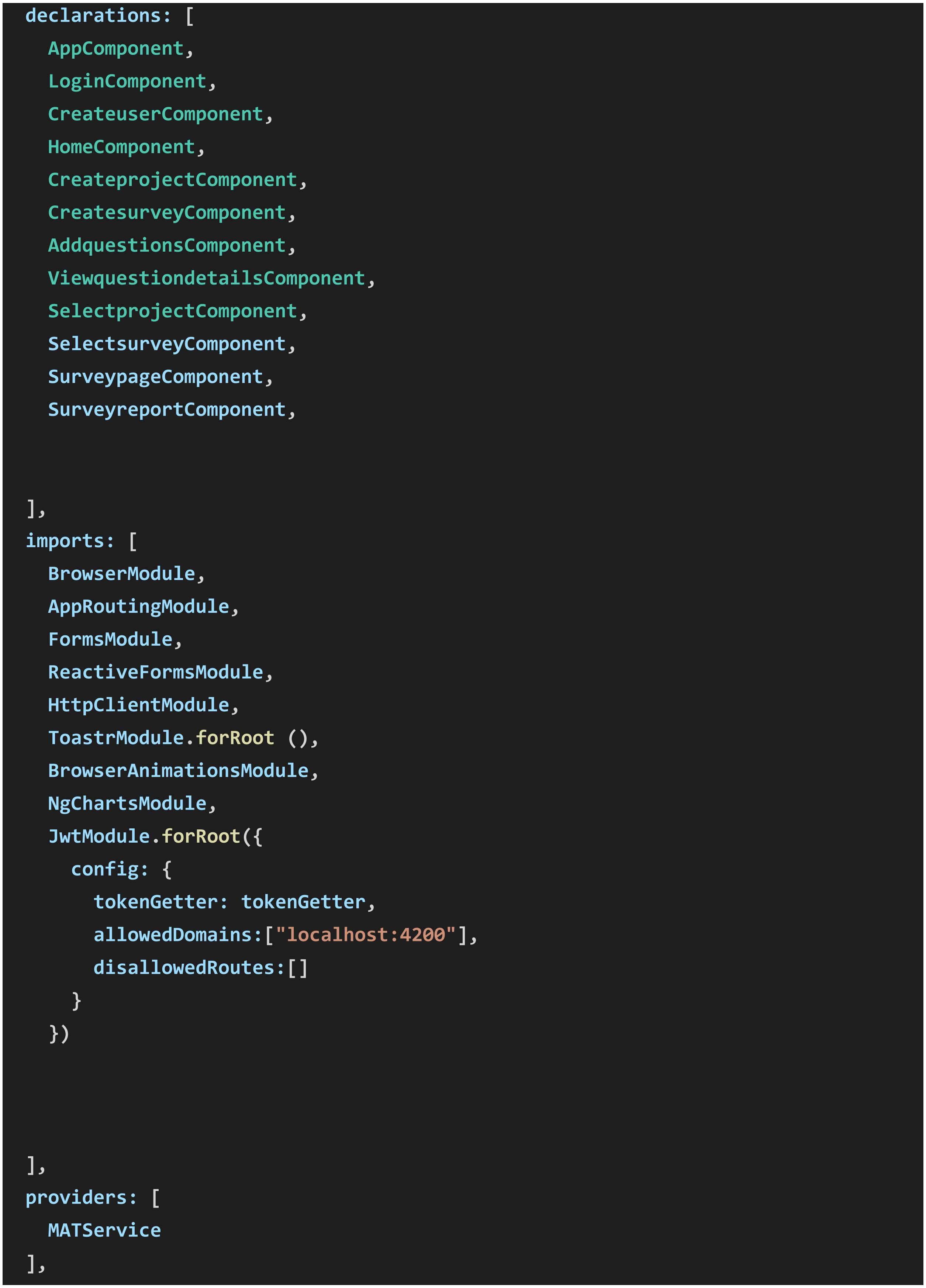


```
        tokenGetter: tokenGetter,
        allowedDomains:["localhost:4200"],
        disallowedRoutes:[]
      }
    })



  ],
  providers: [
    MATService
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

App Component

app.component.css

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { LoginComponent } from './login/login.component';
import { CreateuserComponent } from './createuser/createuser.component';
import { FormsModule } from '@angular/forms';
import { ReactiveFormsModule } from '@angular/forms';
import { HomeComponent } from './home/home.component';
import { HttpClientModule } from '@angular/common/http';
import { MATService } from './mat.service';
import { CreateprojectComponent } from './createproject/createproject.component';
import { ToastrModule } from 'ngx-toastr';
import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { JwtModule } from '@auth0/angular-jwt';
import { CreatesurveyComponent } from './createsurvey/createsurvey.component';
import { AddquestionsComponent } from './addquestions/addquestions.component';
import { ViewquestiondetailsComponent } from
'./viewquestiondetails/viewquestiondetails.component';
import { SelectprojectComponent } from './selectproject/selectproject.component';
import { SelectsurveyComponent } from './selectsurvey/selectsurvey.component';
import { SurveypageComponent } from './surveypage/surveypage.component';
import { NgChartsModule } from 'ng2-charts';
import { SurveyreportComponent } from './surveyreport/surveyreport.component';




export function tokenGetter() {
  return localStorage.getItem("jwt");
}




@NgModule({
```



```
  declarations: [
    AppComponent,
    LoginComponent,
    CreateuserComponent,
    HomeComponent,
    CreateprojectComponent,
    CreatesurveyComponent,
    AddquestionsComponent,
    ViewquestiondetailsComponent,
    SelectprojectComponent,
    SelectsurveyComponent,
    SurveypageComponent,
    SurveyreportComponent,


  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    FormsModule,
    ReactiveFormsModule,
    HttpClientModule,
    ToastrModule.forRoot (),
    BrowserAnimationsModule,
    NgChartsModule,
    JwtModule.forRoot({
      config: {
        tokenGetter: tokenGetter,
        allowedDomains:["localhost:4200"],
        disallowedRoutes:[]
      }
    })



  ],
  providers: [
    MATService
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

app.component.html

```
<router-outlet></router-outlet>
```

app.component.spec.ts

```
import { TestBed } from '@angular/core/testing';
import { RouterTestingModule } from '@angular/router/testing';
import { AppComponent } from './app.component';

describe('AppComponent', () => {
  beforeEach(async () => {
    await TestBed.configureTestingModule({
      imports: [
        RouterTestingModule
      ],
      declarations: [
        AppComponent
      ],
    }).compileComponents();
  });
  it('should create the app', () => {
    const fixture = TestBed.createComponent(AppComponent);
    const app = fixture.componentInstance;
    expect(app).toBeTruthy();
  });

  it(`should have as title 'project'`, () => {
    const fixture = TestBed.createComponent(AppComponent);
    const app = fixture.componentInstance;
    expect(app.title).toEqual('project');
  });
});
```

app.component.ts

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

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

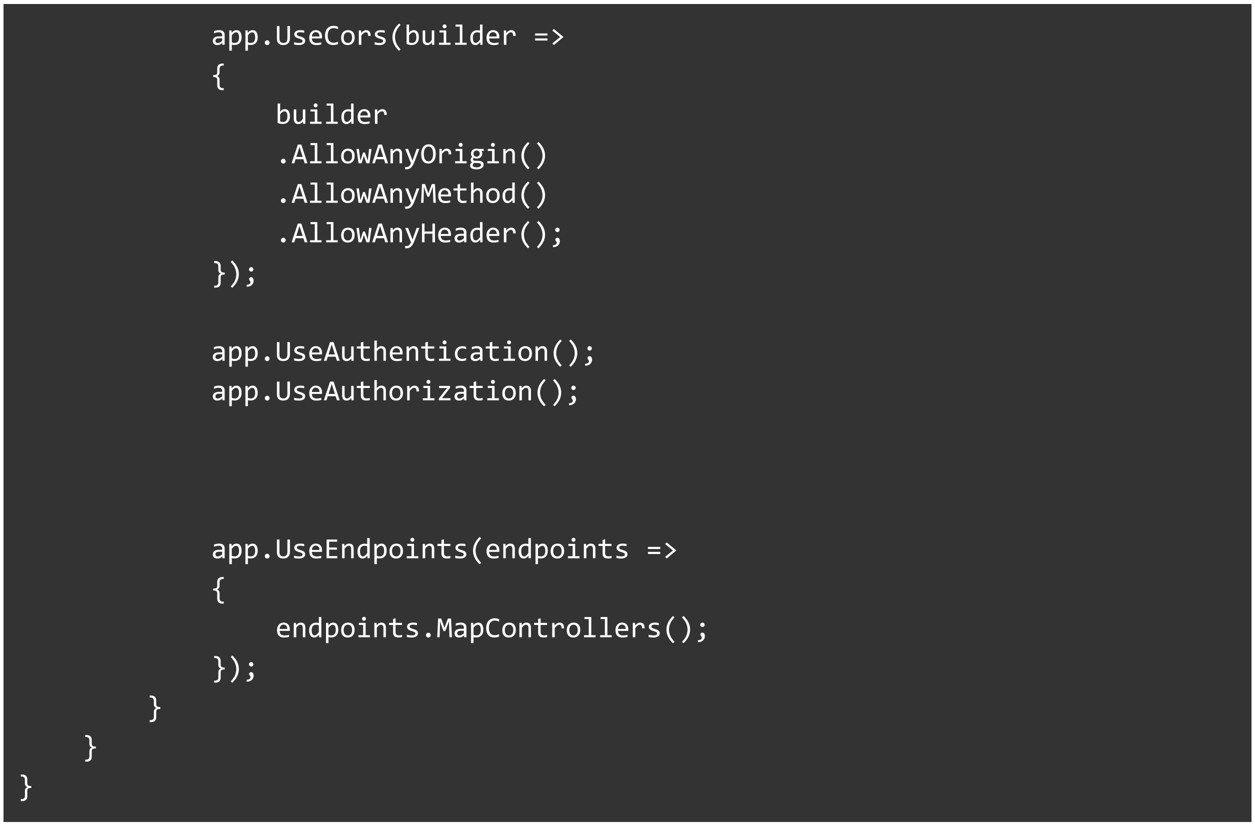
API C-SHARP CODE

Program.cs

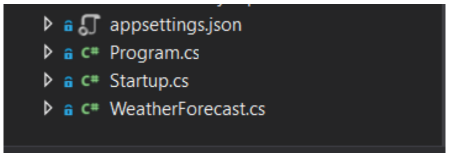
```csharp
using Microsoft.AspNetCore.Hosting;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.Hosting;
using Microsoft.Extensions.Logging;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace PROJECT_MAT
{
    public class Program
    {
        public static void Main(string[] args)
        {
            CreateHostBuilder(args).Build().Run();
        }

        public static IHostBuilder CreateHostBuilder(string[] args) =>
            Host.CreateDefaultBuilder(args)
                .ConfigureWebHostDefaults(webBuilder =>
                {
                    webBuilder.UseStartup<Startup>();
                });
    }
}
```

Status.cs

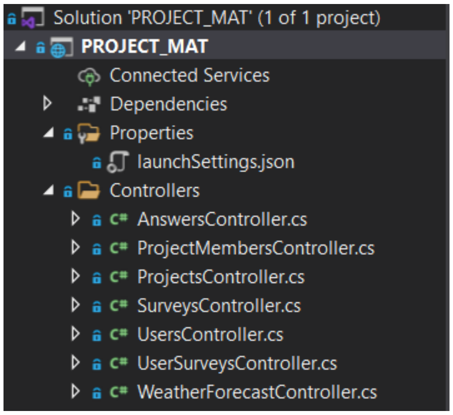
```
using Microsoft.AspNetCore.Authentication.JwtBearer;
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;
using Microsoft.Extensions.Logging;
using PROJECT_MAT.Models;
using Microsoft.IdentityModel.Tokens;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Text;

namespace PROJECT_MAT
{
    public class Startup
    {
        public Startup(IConfiguration configuration)
        {
            Configuration = configuration;
        }

        public IConfiguration Configuration { get; }

        // This method gets called by the runtime. Use this method to add
services to the container.
        public void ConfigureServices(IServiceCollection services)
        {

            services.AddControllers();
            services.AddTransient<IUserRepository, UserRepo>();
            services.AddTransient<IprojectRepository, ProjectRepo>();
            services.AddTransient<IProjectMemberRepository,
projectmemberrepo>();
            services.AddTransient<ISurveyRepository, surveyrepo>();
            services.AddTransient<IUserSurveyRepository,UserSurveyRepo>();
            services.AddCors(options =>
            {
                options.AddDefaultPolicy(
                     builder => builder.AllowAnyOrigin());
            });

            var key =
Encoding.UTF8.GetBytes(Configuration["ApplicationSettings:JWT_Secret"].ToSt
ring());

            services.AddAuthentication(x =>
            {
                x.DefaultAuthenticateScheme =
JwtBearerDefaults.AuthenticationScheme;
                x.DefaultChallengeScheme =
JwtBearerDefaults.AuthenticationScheme;
                x.DefaultScheme = JwtBearerDefaults.AuthenticationScheme;
            }).AddJwtBearer(x => {
                x.RequireHttpsMetadata = false;
                x.SaveToken = false;
                x.TokenValidationParameters = new
Microsoft.IdentityModel.Tokens.TokenValidationParameters
                {
                    ValidateIssuerSigningKey = true,
                    IssuerSigningKey = new SymmetricSecurityKey(key),
                    ValidateIssuer = false,
                    ValidateAudience = false,
                    ClockSkew = TimeSpan.Zero
                };
            });

        }

        // This method gets called by the runtime. Use this method to
configure the HTTP request pipeline.
        public void Configure(IApplicationBuilder app, IWebHostEnvironment
env)
        {
            if (env.IsDevelopment())
            {
                app.UseDeveloperExceptionPage();
            }

            app.UseRouting();
```



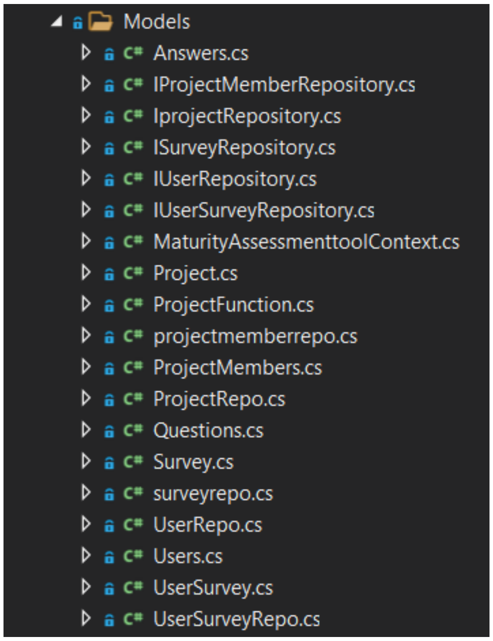
```
            app.UseCors(builder =>
            {
                builder
                .AllowAnyOrigin()
                .AllowAnyMethod()
                .AllowAnyHeader();
            });

            app.UseAuthentication();
            app.UseAuthorization();



            app.UseEndpoints(endpoints =>
            {
                endpoints.MapControllers();
            });
        }
    }
}
```



appsettings.json
Program.cs
Startup.cs
WeatherForecast.cs



Solution 'PROJECT_MAT' (1 of 1 project)
PROJECT_MAT
Connected Services
Dependencies
Properties
launchSettings.json
Controllers
AnswersController.cs
ProjectMembersController.cs
ProjectsController.cs
SurveysController.cs
UsersController.cs
UserSurveysController.cs
WeatherForecastController.cs



Models
Answers.cs
IProjectMemberRepository.cs
IprojectRepository.cs
ISurveyRepository.cs
IUserRepository.cs
IUserSurveyRepository.cs
MaturityAssessmenttoolContext.cs
Project.cs
ProjectFunction.cs
projectmemberrepo.cs
ProjectMembers.cs
ProjectRepo.cs
Questions.cs
Survey.cs
surveyrepo.cs
UserRepo.cs
Users.cs
UserSurvey.cs
UserSurveyRepo.cs

SQL CODE

Table Creation Querys

```
create database MaturityAssessmenttool

CREATE TABLE users
( user_id int PRIMARY KEY identity,
FirstName varchar(25) ,
LastName varchar(25) ,
userType varchar(25) ,
Emailid varchar(25) ,
password varchar(20)
  );

create table ProjectFunction(
Function_id int PRIMARY KEY identity,
FunctionName varchar(25)
)

create table Project(
Project_id int PRIMARY KEY identity,
Projectname varchar(25),
ProjectDescription varchar(50),
Function_id int FOREIGN KEY REFERENCES ProjectFunction(Function_id)
)

create table ProjectMembers(
ProjectMember_id int PRIMARY KEY identity,
Project_id int FOREIGN KEY REFERENCES Project(Project_id),
user_id  int FOREIGN KEY REFERENCES users(user_id )
)

create table Survey(
Survey_id int PRIMARY KEY identity,
Surveyname varchar(25),
Survey_Start_date date,
Survey_End_date date,
Project_id int FOREIGN KEY REFERENCES Project(Project_id)
)

create table Questions(
Question_id int PRIMARY KEY identity,
Question varchar(255),
Function_id int FOREIGN KEY REFERENCES ProjectFunction(Function_id)
)


create table Answers(
Question_id int FOREIGN KEY REFERENCES Questions(Question_id),
Answer_id int PRIMARY KEY identity,
Answer varchar(25),
Answer_Weightage int
)

 create table UserSurvey(
User_Survey_id int PRIMARY KEY identity,
Question_id int FOREIGN KEY REFERENCES Questions(Question_id),
Answer_id int FOREIGN KEY REFERENCES Answers(Answer_id),
Survey_id int FOREIGN KEY REFERENCES Survey(Survey_id),
)
```