

Name: Shashank N

3.23 - Practice Project - myPortfolio: Easy Way to Create Your Amazing Portfolio writeup

Step 1: Start by creating a new Angular application using the Angular CLI.

Step 2: Create the necessary components for the application: AppComponent, HomeComponent, SigninComponent, SignupComponent, ContactusComponent, and AboutUsComponent.

Step 3: Set up the routing configuration in the AppRoutingModuleModule file to map the different components to their corresponding routes.

Step 4: In the app.component.html file, define the structure of the main application layout. It includes a navigation bar (<nav>) and a container (<div class="container">) with a router outlet (<router-outlet>) to display the content based on the active route.

Step 5: Style the application using the provided CSS classes in the app.component.css file. This includes styling for the container, navbar, images, and router outlet.

Step 6: Define the HTML template and styles for each component:

Step 7: homepage.component.html: Displays a jumbotron with a welcome message and introduction.

Step 8: signin.component.html: Displays a form to enter email and password for login.

Step 9: signup.component.html: Displays a form to enter email and password for registration.

Step 10: contactus.component.html: Displays contact information such as address, email, and phone number.

Step 11: about-us.component.html: Displays information about the developer, including a profile picture and description.

Step 12: Implement the respective TypeScript files for each component (signin.component.ts, signup.component.ts, etc.), exporting the component class.

Step 13: In the app.component.ts file, define the AppComponent class with the necessary properties and methods. Set the title property to 'myprotfolio'.

Step 14: In the homepage.component.ts, signin.component.ts, signup.component.ts, contactus.component.ts, and about-us.component.ts files.

Step 15: Build and run the Angular application to see the portfolio website with navigation, content, and functionality