

Angular Frontend Project Manual

This manual provides step-by-step instructions for setting up an Angular project named "frontend" using Angular CLI. It also includes instructions for creating two components: "sign-up" and "log-in".

Prerequisites

Before proceeding with the setup, make sure you have the following prerequisites installed on your system:

- Node.js
- Angular CLI

Project Setup

To set up the Angular project named "frontend" using Angular CLI, follow these steps:

1. Open a terminal or command prompt.
2. Navigate to the desired directory where you want to create your project.
3. Run the following command to create a new Angular project:

```
ng new frontend
```

4. Change to the project directory:

```
cd frontend
```

5. Start the development server:

```
ng serve
```

This will compile the project and start a development server at

```
http://localhost:4200
```

.

Creating Components

Sign Up Component

1. Create the `sign-up` component by running the following command:

```
ng generate component sign-up
```

This will generate the required files for the component under the `src/app/sign-up` directory.

2. Open the `sign-up.component.html` file located at `src/app/sign-up` and replace its content with the following code:

```
<div class="container d-flex justify-content-center align-items-center h-100">
  <div class="row custom-row border border-dark rounded p-5">
    <div class="mb-3">
      <label for="exampleFormControlInput1" class="form-label">Email address</label>
      <input type="email" class="form-control" id="exampleFormControlInput1" placeholder="name@example.com">
    </div>
    <div class="mb-3">
      <label for="inputPassword5" class="form-label">Password</label>
      <input type="password" id="inputPassword5" class="form-control" aria-labelledby="passwordHelpBlock">
    </div>

    <button type="button" class="btn btn-danger">Danger</button>
    <p class="mt-3">Don't have an Account? <a href="sign-up">SignUp</a>
  </p>
</div>
</div>
```

Log In Component

1. Create the `log-in` component by running the following command:

```
ng generate component log-in
```

This will generate the required files for the component under the `src/app/log-in` directory.

2. Open the `log-in.component.html` file located at `src/app/log-in` and replace its content with the following code:

```
<div class="container d-flex justify-content-center align-items-center h-100">
  <div class="row custom-row border border-dark rounded p-5">
    <div class="mb-3">
      <label for="exampleFormControlInput1" class="form-label">Email address</label>
      <input type="email" class="form-control" id="exampleFormControlInput1" placeholder="name@example.com">
    </div>
    <div class="mb-3">
      <label for="inputPassword5" class="form-label">Password</label>
      <input type="password" id="inputPassword5" class="form-control" aria-labelledby="passwordHelpBlock">
    </div>
    <div class="row">
      <div class="col mb-3">
        <label for="exampleFormControlInput1" class="form-label">First Name</label>
```

```

        <input type="text" class="form-control" placeholder="First
name" aria-label="First name">
      </div>
      <div class="col mb-3">
        <label for="exampleFormControlInput1" class="form-
label">Last Name</label>
        <input type="text" class="form-control" placeholder="Last
name" aria-label="Last name">
      </div>
    </div>
    <button type="button" class="btn btn-danger">Danger</button>
    <p class="mt-3">Already have an Account? <a href="log-in">Login</a>
  </p>
</div>
</div>

```

Adding Bootstrap

1. Open the `index.html` file located at `src` folder.
2. Add the following code inside the `<head>` section of the file to include the Bootstrap CSS stylesheet:

```

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.c
s" rel="stylesheet"
integrity="sha384-
9ndCyuaIbzAi2FUVXJi0CjmCapSm07SnpJef0486qhlLnuZ2cdeRh002iuk6FUUVM"
crossorigin="anonymous">

```

3. Add the following code before the closing `</body>` tag to include the Bootstrap JavaScript bundle:

```

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.m
in.js"
integrity="sha384-
geWF76RCwLtnZ8qwwOWPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz"
crossorigin="anonymous"></script>

```

Setting up Routes

1. Open the `app-routing.module.ts` file located at `src/app` and replace its content with the following code:

```

import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';
import { SignUpComponent } from './sign-up/sign-up.component';
import { LogInComponent } from './log-in/log-in.component';

const routes: Routes = [
  { path: 'sign-up', component: SignUpComponent },
  { path: 'log-in', component: LogInComponent },

```

```
];

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

2. Open the `app.component.html` file located at `src/app` and replace its content with the following code:

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

That's it! You have successfully set up the Angular project named "frontend" with two components, "sign-up" and "log-in". The routes for these components have also been configured. You can now continue building your Angular application using these components.

Adding Routes in components

In both `sign-up.component.ts` and `log-in.component.ts` import following library

```
import { Router } from '@angular/router';
```

Also make constructor in export class like following

```
constructor(private router: Router) { }
```