





Workshop Angular 7

Na aula 1 aprendemos o que é o Angular e como ele trabalha explorando a sua arquitetura e entendendo os seus elementos.

Por fim, criamos nossa aplicação Angular, geramos módulos e componentes, mas como fazemos para acessarmos nossos componentes, seria possível configurar uma rota específica para cada um?

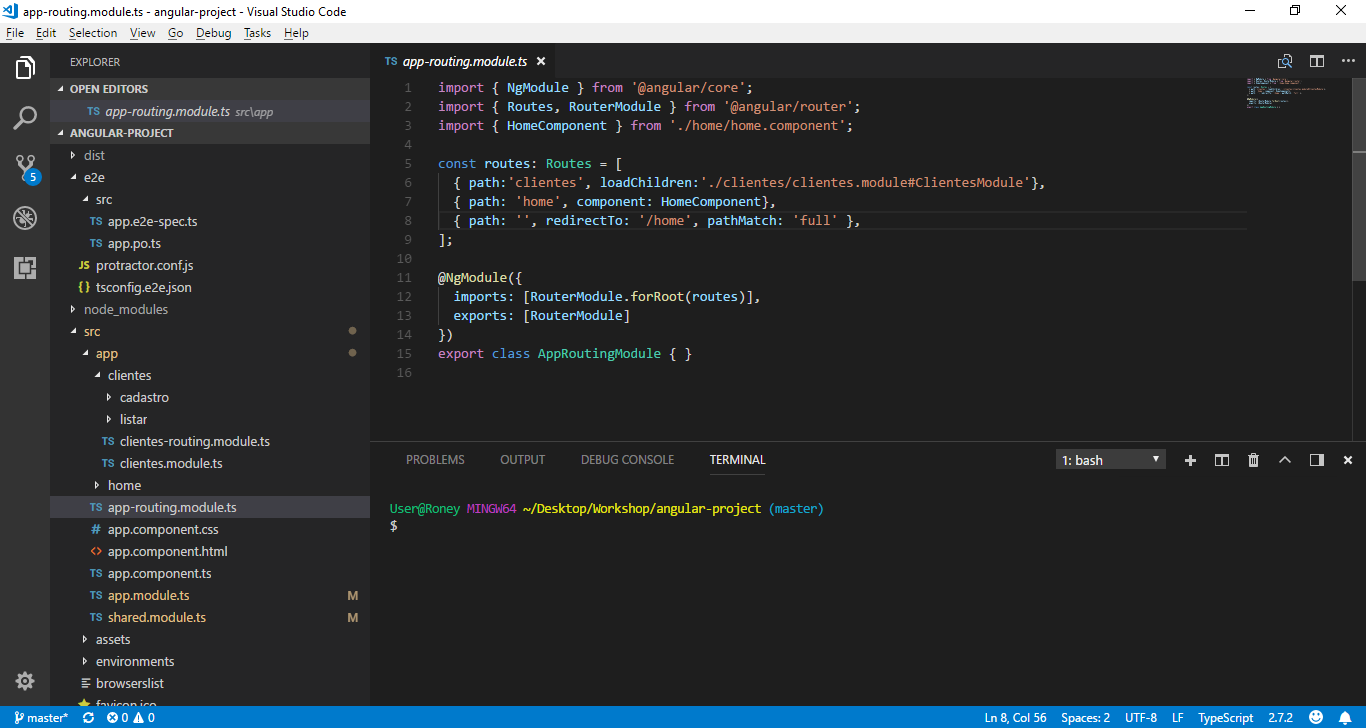
**Criando rotas**

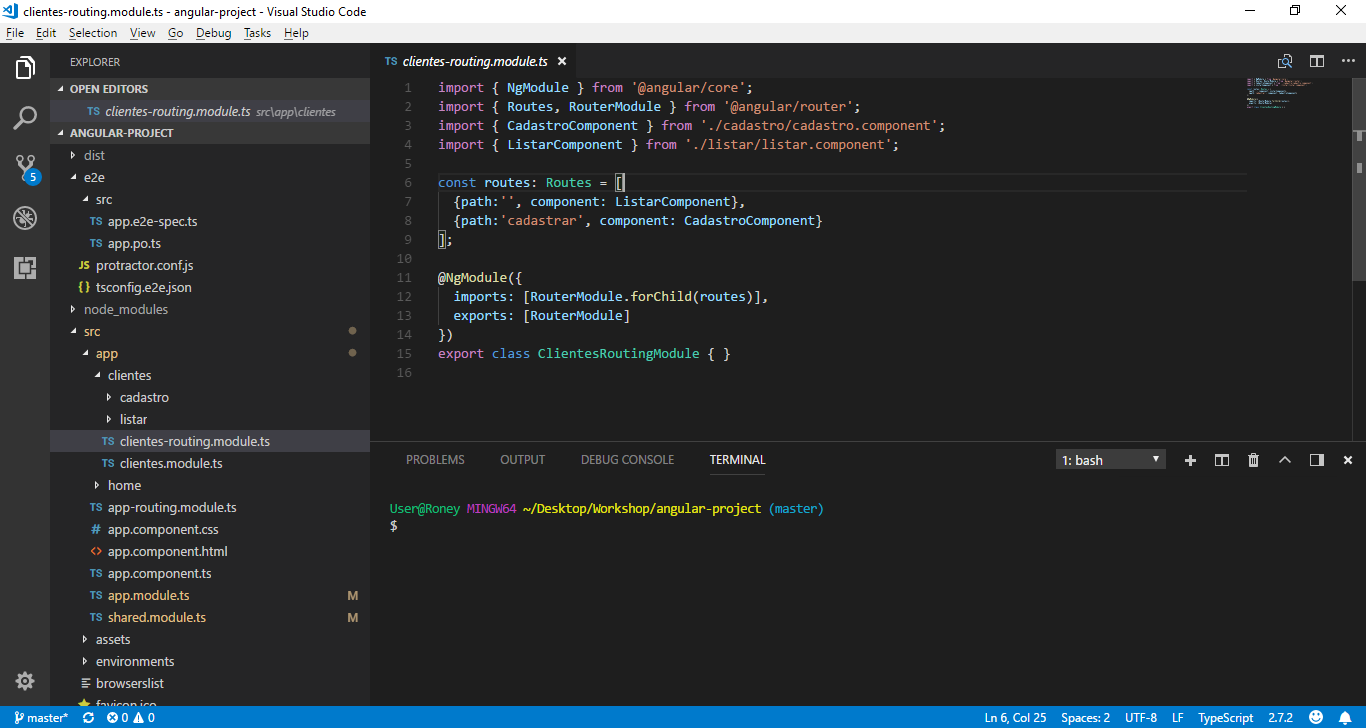
As rotas da aplicação, por padrão são configuradas em modulos denominados como routing.module.ts

Quando criamos nosso projeto, executamos o comando de novo projeto seguido do comando –routing gerando automaticamente o arquivo app.routing.module.ts.

Podemos destacar o modulo app.routing.module.ts como o roteador principal da aplicação sendo ele o responsável por dar as rotas iniciais dos outros módulos da aplicação e os routing.modules de cada um destes módulos são responsáveis por declarar as rotas específicas dos seus componentes.

**Declarando Rotas em app-routing.module.ts**



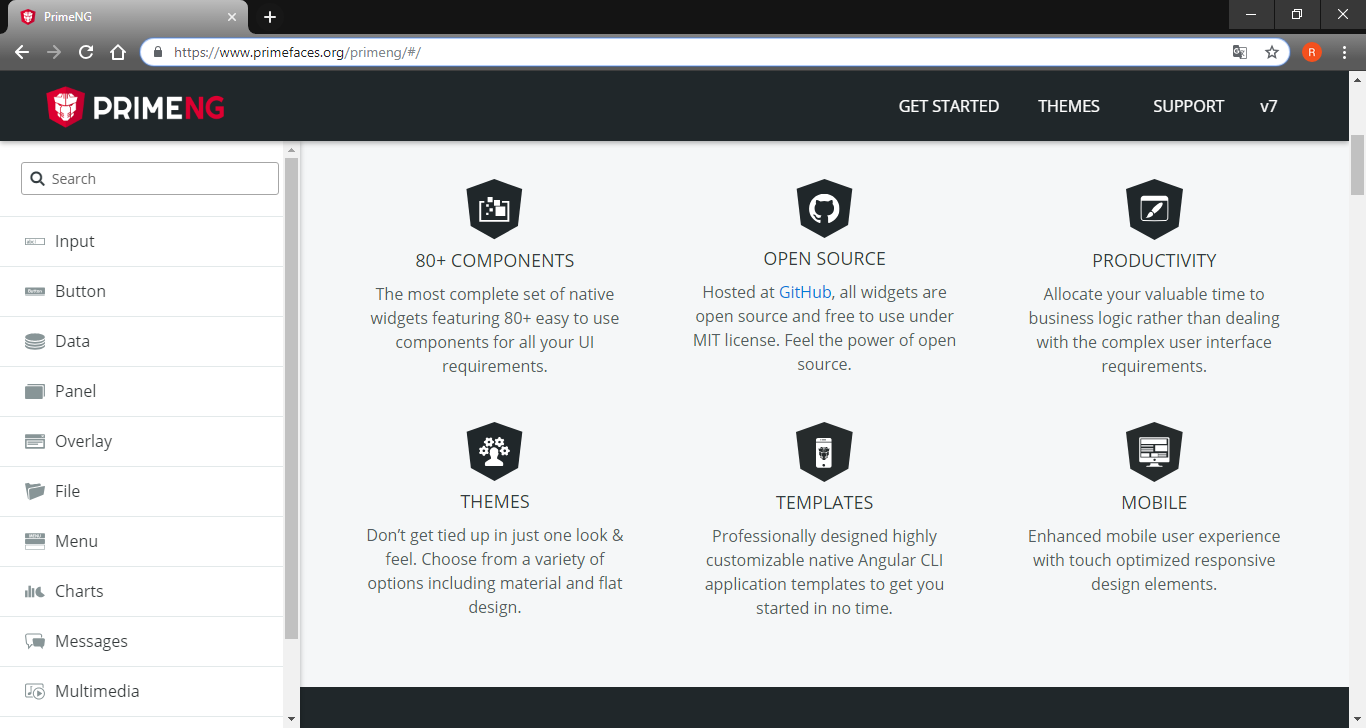
**Declarando Rotas em clientes-routing.module.ts**

**Trabalhando com Core.module.forRoot()**

**Realizando um CRUD**

Explorando o PrimeNG

PrimeNG é uma coleção de componentes de UI ricos para Angular. Todos os widgets são de código aberto e gratuitos para uso sob a licença MIT. O PrimeNG é desenvolvido pela [PrimeTek Informatics](http://www.primetek.com.tr/) , um fornecedor com anos de experiência no desenvolvimento de soluções de UI de código aberto.



Importando o PrimeNg em nosso Projeto

Primeiro passo

Primeiramente, é necessário baixar as dependências do PrimeNG onde as mesmas estão disponíveis no npm.

Para baixa-las basta executar os comandos abaixo:

npm install primeng --save

npm install primeicons –save

npm install @angular/animations --save

A dependência animations é utilizada em diversos componentes do framework, logo ao realizar o download da biblioteca, o modulo **BrowserAnimationsModule**

do Animations deve ser importado em nossa aplicação.

Segundo passo

Importarno as dependências do PrimeNG no package json e declarar nos assets os styles do PrimeNG.

Se caso em nosso package.json não ter sido gravadas as declarações das dependências do PrimeNG, devemos adiciona-las dentro de **dependencies**:

"dependencies": {

//...

"primeng": "^7.0.0",

"primeicons": "^1.0.0"

},

Por fim, para nossa aplicação reconhecer os estilos do PrimeNG, devemos adicionar os seguintes CSSs dentro de Styles do arquivo agular.json.

"styles": [

"node\_modules/primeicons/primeicons.css",

"node\_modules/primeng/resources/themes/nova-light/theme.css",

"node\_modules/primeng/resources/primeng.min.css",

//...

],

Exercício PrimeNg

**Criando um menu**

Com o primeng

Trabalhando com Guards

Segregando nível de acesso por Roles

Usando auth e jwt

Token de acesso

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c3IiOiJyYW1vcmluIiwibmFtZSI6IlJPTkVZIEFVR1VTVE8gQU1PUklNIiwiZnVuY2lvbmFsIjoiMDMzMjIwOTkzIiwiaWF0IjoxNTE2MjM5MDIyLCJleHAiOiIyMDE5LTEwLTAxIDAwOjAwOjAwIiwiYXV0aG9yaXRpZXMiOlsiUk9MRV9HUlVQT19DQURBU1RSQVIiLCJST0xFX0dSVVBPX0VESVRBUiIsIlJPTEVfR1JVUE9fQ09OU1VMVEFSIiwiUk9MRV9HUlVQT19FWENMVUlSIl19.xyrCM336Qx-Xw4i-zsSOg2YhAUsC-ftKHdTbpvbJjQU

https://www.youtube.com/watch?v=z4JUm0Bq9AM

<https://stackoverflow.com/questions/37364973/promise-vs-observable>

Usando o async await

Referencias

<https://www.devmedia.com.br/novidades-do-angular-7/40194>

<https://angular.io/cli>

<https://blog.angular.io/version-6-of-angular-now-available-cc56b0efa7a4>

Pré Requisitos

Git Bash

Visual Studio Code

Possíveis Problemas

Proxy e comunicação por IP