

### Contextualização:

Este projeto inicialmente foi desenvolvido tomando como base o projeto de estudo roubank, onde foi pego este projeto como inicial e foi-se desenvolvendo as funcionalidades específicas para o projeto esse eu já li.

- Criar o repositório no GitHub com a estrutura do Gitflow, ou seja, branches main e develop.


O projeto encontra-se em:

<https://github.com/elizangelacristinha/angular-esseEuJaLi> .

- Usar componentes de algum framework CSS (Bootstrap, Materialize ou outro).

No referido projeto utilizou-se o framework Materialize.

- Apresentar as telas com layout responsivo usando ou não algum framework CSS.



NOVO   LISTAR   VISUALIZAR   MARCAR LEITURA   RANKING   PONTOS E TROFÉUS

# LOGIN

ENTRE COM OS SEUS DADOS

Nome de Usuário


login1

Senha

\*\*\*

ENVIAR

© COPYRIGHT - TODOS OS DIREITOS RESERVADOS



NOVO   LISTAR   VISUALIZAR   MARCAR LEITURA   RANKING   PONTOS E TROFÉUS

SEJA BEM VINDO(A)... MILENE (LOGIN1)

## LISTA DE LIVROS

ESTA É A LISTA DOS LIVROS MAIS LIDOS DO MUNDO. SELECIONE PARA LER O RESUMO E MARCAR SE JÁ O LEU E GANHE PONTOS!


FILTRE PELOS TIPOS DE OPERAÇÃO

☐ Título   ☐ Autor

ID	TÍTULO	AUTOR	
1	O Senhor dos Anéis	J. R. R. Tolkien	<div>VISUALIZAR</div>
2	Miguel de Cervantes	Don Quixote	<div>VISUALIZAR</div>
3	Charles Dickens	Um Conto de Duas Cidades	<div>VISUALIZAR</div>

VOLTAR AO MENU PRINCIPAL

© COPYRIGHT - TODOS OS DIREITOS RESERVADOS



NOVO   LISTAR   VISUALIZAR   MARCAR LEITURA   RANKING   PONTOS E TROFÉUS

# ENTRE COM OS DADOS DO USUÁRIO

Nome Completo \*

Nome de Usuário \*

login1

Senha \*

\*\*\*

Repita a senha \*


Por favor, nos informe a sua senha com pelo menos 6 caracteres!

ENVIAR

## USUÁRIOS CADASTRADOS

Nome Completo	Nome de usuário	CPF	
Mikene (Login1)	login1		 
Vicente (Login3)	login2		 
Antônio (Login3)	login3		 

© COPYRIGHT - TODOS OS DIREITOS RESERVADOS



NOVO   LISTAR   VISUALIZAR   MARCAR LEITURA   RANKING   PONTOS E TROFÉUS

SEJA BEM VINDO(A)... MILENE (LOGIN1)

1 - O SENHOR DOS ANÉIS

AUTOR

J. R. R. Tolkien


DESCRIÇÃO

A história se passa na Terra Média, nos últimos dias da Terceira Era do Mundo. Ela é dividida em três grandes partes e conta as lutas travadas pela posse do Anel. Nos primórdios do tempo, os deuses forjaram Anéis do Poder e os deram às diversas raças que povoavam o mundo. Nove anéis foram dados aos reis dos homens, sete anéis aos reis dos anões e três anéis aos reis dos elfos. Entretanto, Sauron, um deus maligno, aprendeu o segredo da criação dos anéis e forjou O Anel, tão poderoso que era capaz de dominar todos os outros. De posse do Anel, ele espalhou o Mal sobre o mundo, escravizando os reis dos homens e transformando-os nos Cavaleiros Negros, servos do Anel. Numa batalha titânica entre as forças do Bem e do Mal que marcou o final da Segunda Era, o Anel foi arrancado de Sauron e as forças do Mal foram derrotadas. Sauron perdeu sua forma corporal e seu espírito ficou preso na Terra de Mordor. Poder e a derrota de Sauron, é contada no livro "O Silmarillion". FONTE: <https://paginadonardo.wordpress.com/2020/01/27/resumo-o-senhor-dos-anéis-j-r-r-tolkien/>

ESSE EU JÁ LI

Meus Pontos: 12

© COPYRIGHT - TODOS OS DIREITOS RESERVADOS



NOVO   LISTAR   VISUALIZAR   MARCAR LEITURA   RANKING   PONTOS E TROFÉUS

SEJA BEM VINDO(A)...

RANKING DE PONTOS

ESTA É A LISTA DE PONTOS DOS USUÁRIOS

FILTRE PELOS TIPOS

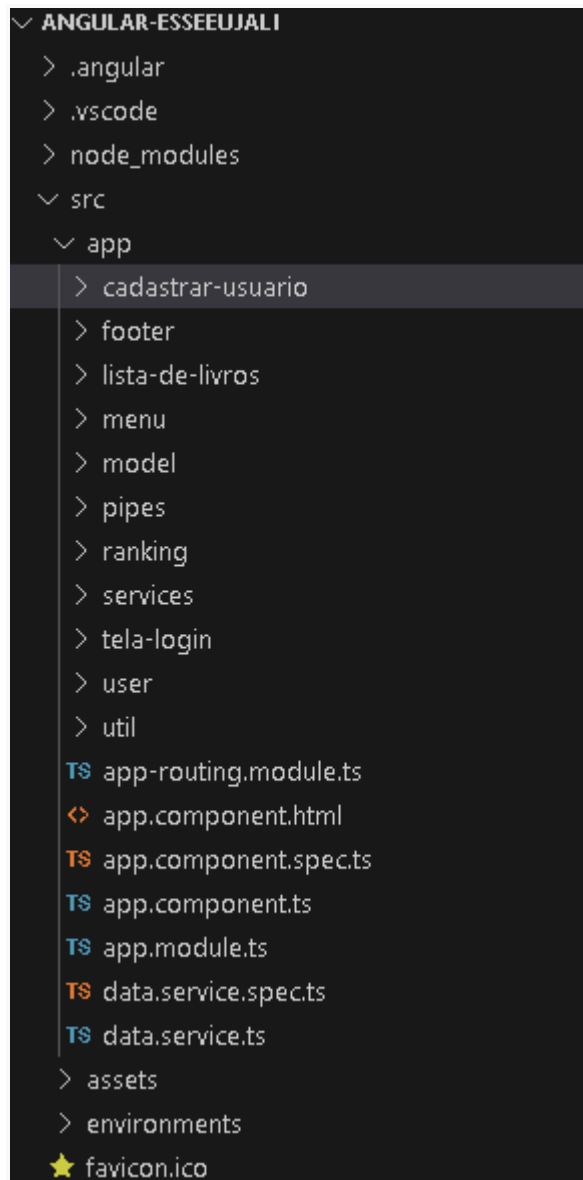
☐ Pontos   ☐ Usuário

Pontos	Usuário
12	login1
0	login2
0	login3

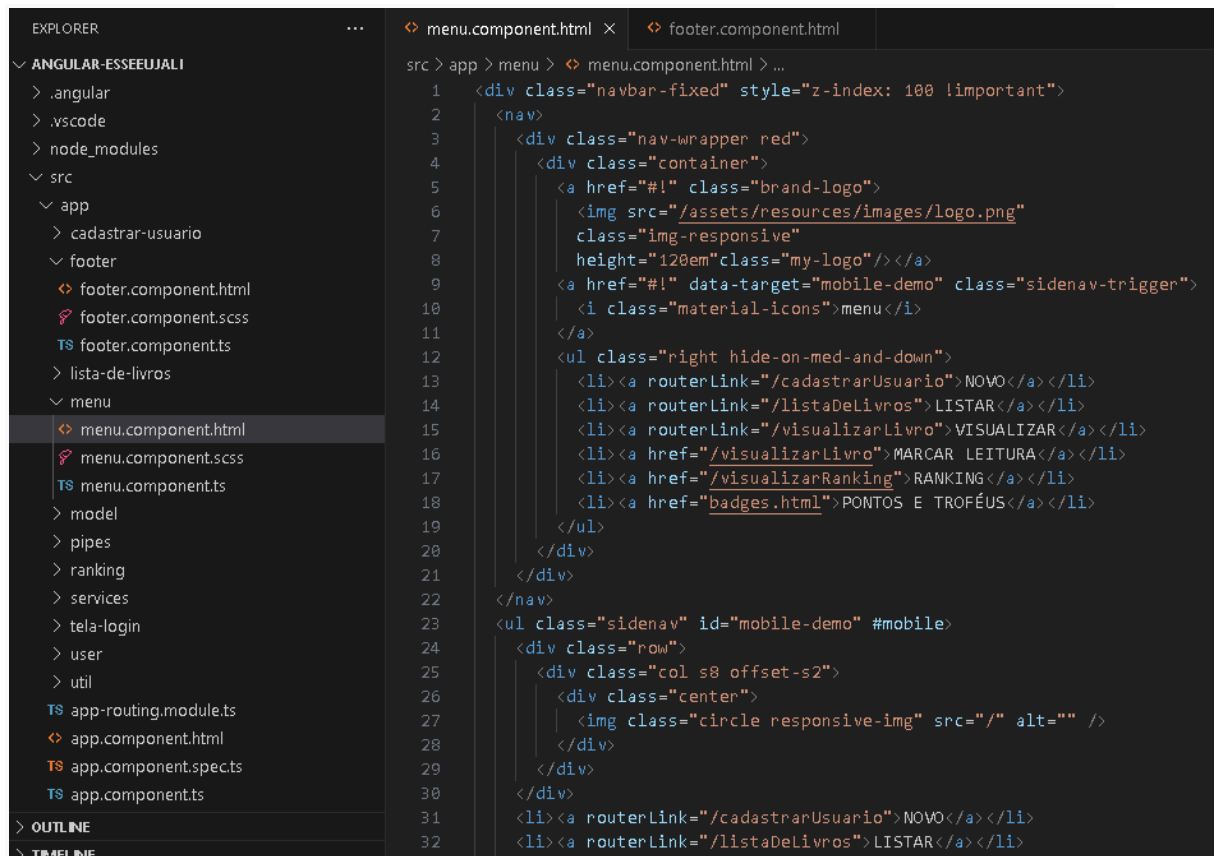
VOLTAR AO MENU PRINCIPAL

© COPYRIGHT - TODOS OS DIREITOS RESERVADOS

- Construir páginas web com o conceito de componentes.



- Criar o layout da aplicação com componentes, ou seja, o cabeçalho e rodapé precisam ser componentes.



```
src > app > menu > menu.component.html > ...
1 <div class="navbar-fixed" style="z-index: 100 !important">
2   <nav>
3     <div class="nav-wrapper red">
4       <div class="container">
5         <a href="#" class="brand-logo">
6            </a>
9         <a href="#" data-target="mobile-demo" class="sidenav-trigger">
10          <i class="material-icons">menu</i>
11        </a>
12        <ul class="right hide-on-med-and-down">
13          <li><a routerLink="/cadastrarUsuario">NOVO</a></li>
14          <li><a routerLink="/listaDeLivros">LISTAR</a></li>
15          <li><a routerLink="/visualizarLivro">VISUALIZAR</a></li>
16          <li><a href="/visualizarLivro">MARCAR LEITURA</a></li>
17          <li><a href="/visualizarRanking">RANKING</a></li>
18          <li><a href="badges.html">PONTOS E TROFÉUS</a></li>
19        </ul>
20      </div>
21    </div>
22  </nav>
23  <ul class="sidenav" id="mobile-demo" #mobile>
24    <div class="row">
25      <div class="col s8 offset-s2">
26        <div class="center">
27          
28        </div>
29      </div>
30    </div>
31    <li><a routerLink="/cadastrarUsuario">NOVO</a></li>
32    <li><a routerLink="/listaDeLivros">LISTAR</a></li>
```

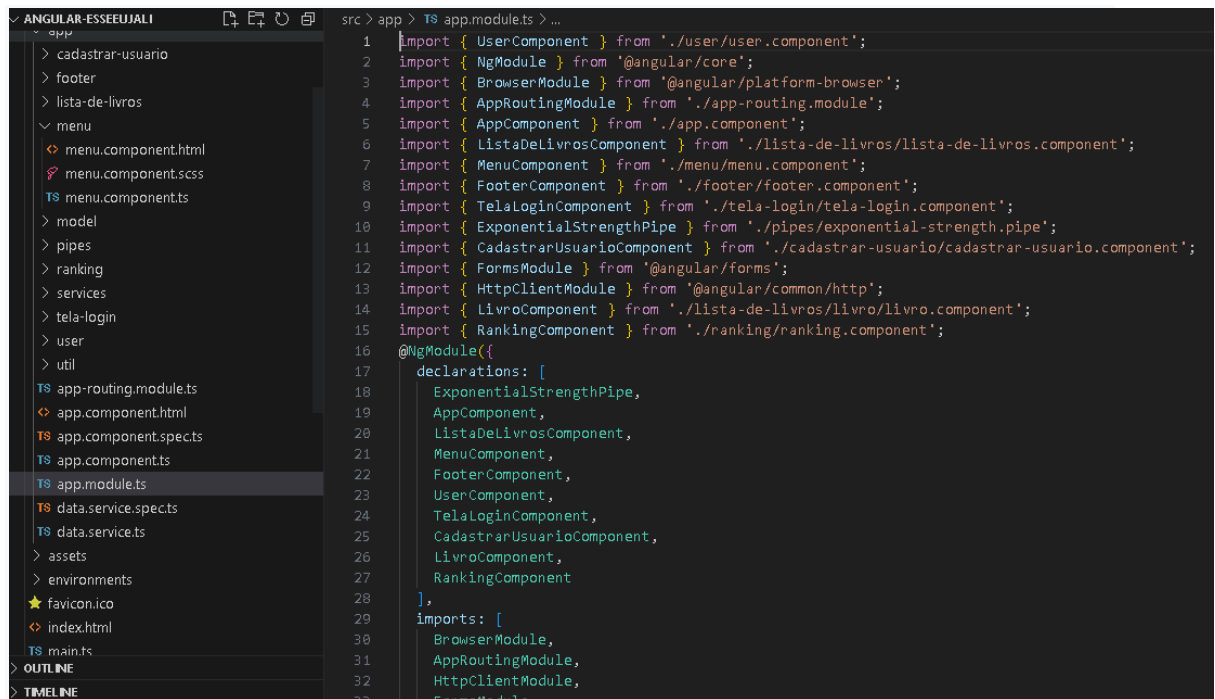


```
src > app > footer > footer.component.html > footer.page-footer.red > div.footer-copyright > div.container.my-font
1 <footer class="page-footer red">
2   <div class="container red">
3     <div class="row">
4       <div class="col s12 center">
5         <img />
6       </div>
7     </div>
8   </div>
9   <div class="footer-copyright">
10     <div class="container my-font">
11       &copy; Copyright - Todos os direitos reservados
12     </div>
13   </div>
```

- Usar pelo menos dois tipos de data-binding (Interpolation, Property Binding, Event Binding e Two Way Data Binding).



- Mapear componentes à rotas no módulo de rotas.

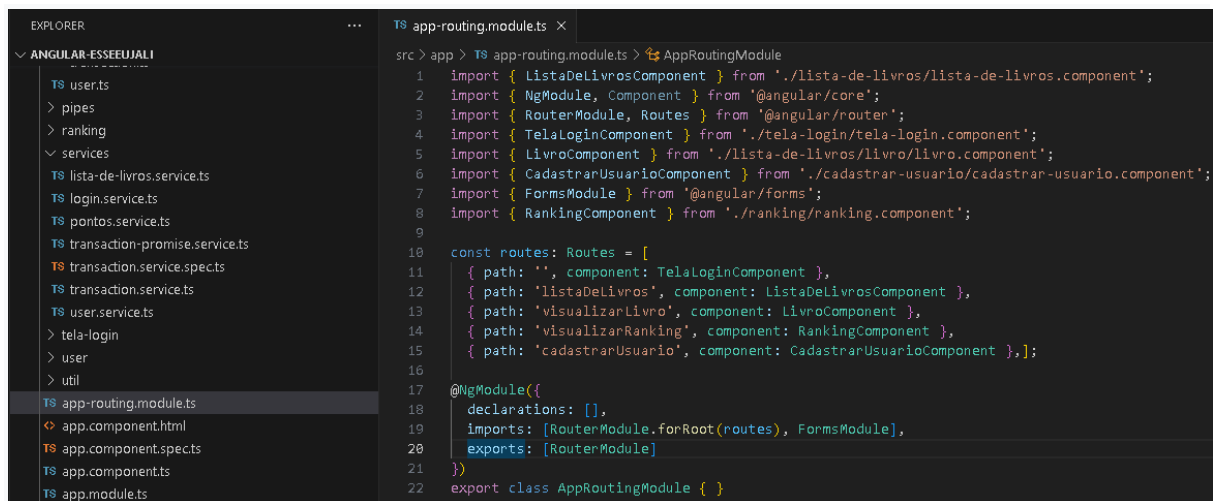


```

src > app > TS app.module.ts > ...
1  import { UserComponent } from './user/user.component';
2  import { NgModule } from '@angular/core';
3  import { BrowserModule } from '@angular/platform-browser';
4  import { AppRoutingModule } from './app-routing.module';
5  import { AppComponent } from './app.component';
6  import { ListaDelivrosComponent } from './lista-de-livros/lista-de-livros.component';
7  import { MenuComponent } from './menu/menu.component';
8  import { FooterComponent } from './footer/footer.component';
9  import { TelaLoginComponent } from './tela-login/tela-login.component';
10 import { ExponentialStrengthPipe } from './pipes/exponential-strength.pipe';
11 import { CadastrarUsuarioComponent } from './cadastrar-usuario/cadastrar-usuario.component';
12 import { FormsModule } from '@angular/forms';
13 import { HttpClientModule } from '@angular/common/http';
14 import { LivroComponent } from './lista-de-livros/livro/livro.component';
15 import { RankingComponent } from './ranking/ranking.component';
16 @NgModule({
17   declarations: [
18     ExponentialStrengthPipe,
19     AppComponent,
20     ListaDelivrosComponent,
21     MenuComponent,
22     FooterComponent,
23     UserComponent,
24     TelaLoginComponent,
25     CadastrarUsuarioComponent,
26     LivroComponent,
27     RankingComponent
28   ],
29   imports: [
30     BrowserModule,
31     AppRoutingModule,
32     HttpClientModule,
33     FormsModule

```

- Criar navegação entre páginas por meio de rotas.



```

src > app > TS app-routing.module.ts > TS AppRoutingModule
1  import { ListaDelivrosComponent } from './lista-de-livros/lista-de-livros.component';
2  import { NgModule, Component } from '@angular/core';
3  import { RouterModule, Routes } from '@angular/router';
4  import { TelaLoginComponent } from './tela-login/tela-login.component';
5  import { LivroComponent } from './lista-de-livros/livro/livro.component';
6  import { CadastrarUsuarioComponent } from './cadastrar-usuario/cadastrar-usuario.component';
7  import { FormsModule } from '@angular/forms';
8  import { RankingComponent } from './ranking/ranking.component';
9
10 const routes: Routes = [
11   { path: '', component: TelaLoginComponent },
12   { path: 'listaDelivros', component: ListaDelivrosComponent },
13   { path: 'visualizarLivro', component: LivroComponent },
14   { path: 'visualizarRanking', component: RankingComponent },
15   { path: 'cadastrarUsuario', component: CadastrarUsuarioComponent },];
16
17 @NgModule({
18   declarations: [],
19   imports: [RouterModule.forRoot(routes), FormsModule],
20   exports: [RouterModule]
21 })
22 export class AppRoutingModule { }

```

- Passar dados entre componentes que representam diferentes telas via parâmetros de rotas.

Não foi utilizado.

- Validar campos do formulário com REGEX e apresentar os erros.

```

src > app > cadastrar-usuario > < cadastrar-usuario.component.html > <div.container.con
  40 [(ngModel)] = this.user.userName
  41 #name="ngModel"
  42 required/>
  43 <label for="input-username">Nome de Usuário *</label>
  44 <span
  45 class="helper-text"
  46 data-error="O campo é inválido"
  47 data-success="Ok!"
  48 ></span>
  49 </div>
  50
  51 <div class="input-field col s12 l6">
  52   <input
  53     id="input-password"
  54     name="input-password"
  55     type="password"
  56     class="validate"
  57     [(ngModel)]="this.user.password"
  58     #name="ngModel"
  59     pattern="^[a-zA-Z0-9_@#\\&\\$]{6,18}$"
  60     required
  61   />
  62   <label for="input-password">Senha *</label>
  63 </div>

```

- Desabilitar o botão de submit enquanto o formulário está inválido.

Inserir lógica.

- Fazer requisições a API com tratamento da resposta com Promises ou Observables.

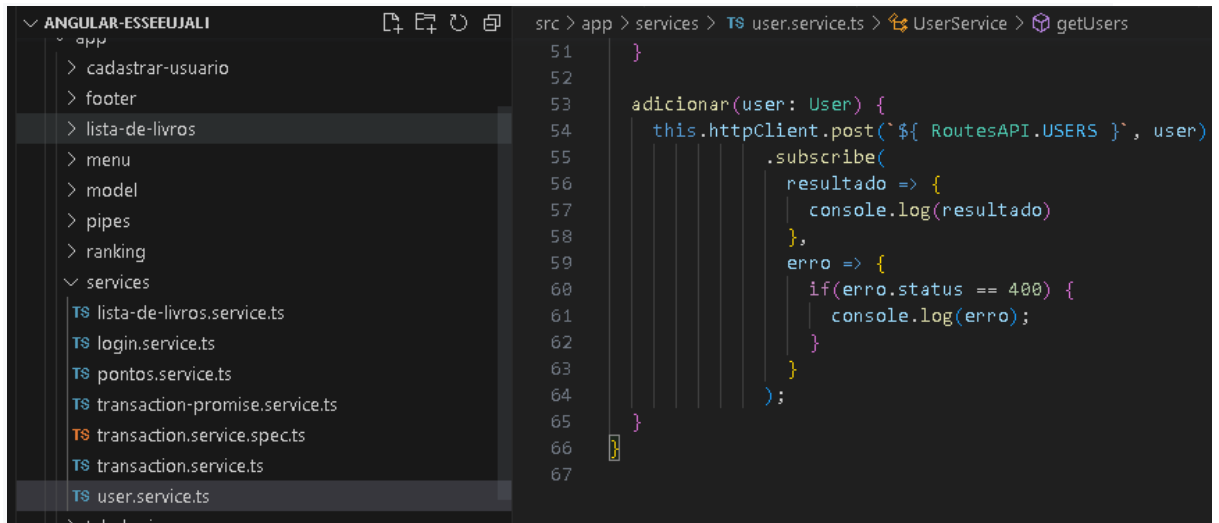
```

src > app > services > TS user.service.ts > UserService > getUsers
  30 );
  31 }
  32
  33 patch(user: User): Observable<User> {
  34   return this.httpClient.patch<User>(
  35     `${RoutesAPI.USERS}/${user.id}`,
  36     user,
  37     this.httpOptions
  38   );
  39 }
  40
  41 update(user: User): Observable<User> {
  42   return this.httpClient.put<User>(
  43     `${RoutesAPI.USERS}/${user.id}`,
  44     user,
  45     this.httpOptions
  46   );
  47 }
  48
  49 getUsers(): Observable<User[]> {
  50   return this.httpClient.get<User[]>(RoutesAPI.USERS);
  51 }
  52
  53 adicionar(user: User) {
  54   this.httpClient.post(`${RoutesAPI.USERS}`, user)
  55     .subscribe(
  56       resultado => {
  57         console.log(resultado)
  58       },
  59       erro => {
  60         if(erro.status == 400) {
  61           console.log(erro);
  62         }
  63       }
  64     );
  65 }

```

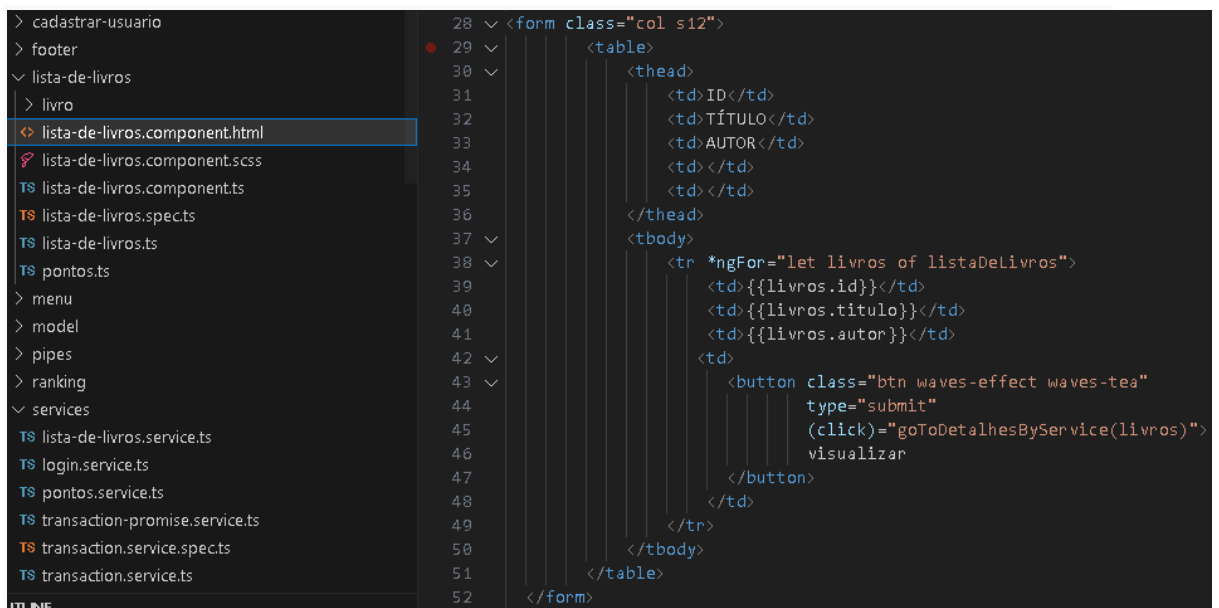


- Cadastrar uma entidade no JSON Server



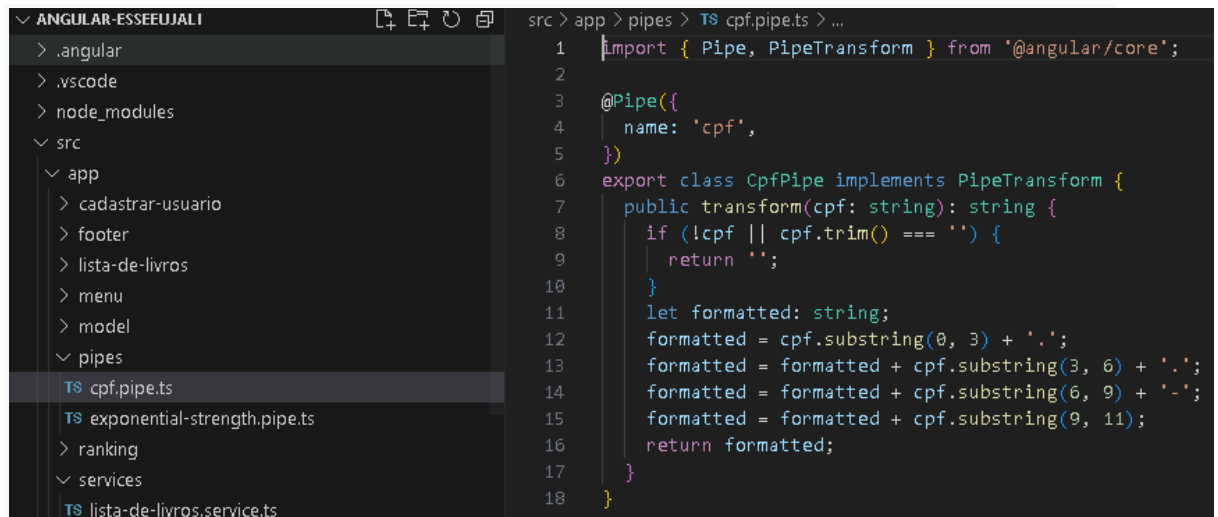
```
51 }
52
53 adicionar(user: User) {
54   this.httpClient.post(`${ RoutesAPI.USERS }`, user)
55     .subscribe(
56       resultado => {
57         console.log(resultado)
58       },
59       erro => {
60         if(erro.status == 400) {
61           console.log(erro);
62         }
63       }
64     );
65 }
66
67
```

- Apresentar uma lista de dados com a diretiva estrutural ngFor.



```
28 <form class="col s12">
29   <table>
30     <thead>
31       <tr>
32         <th>ID</th>
33         <th>TÍTULO</th>
34         <th>AUTOR</th>
35       </tr>
36     </thead>
37     <tbody>
38       <tr *ngFor="let livros of listaDeLivros">
39         <td>{{livros.id}}</td>
40         <td>{{livros.titulo}}</td>
41         <td>{{livros.autor}}</td>
42         <td>
43           <button class="btn waves-effect waves-tea"
44             type="submit"
45             (click)="goToDetalhesByService(livros)">
46             visualizar
47           </button>
48         </td>
49       </tr>
50     </tbody>
51   </table>
52 </form>
```

- Formatar a apresentação de dados com Pipes.



```
src > app > pipes > Ts cpf.pipe.ts > ...
1  import { Pipe, PipeTransform } from '@angular/core';
2
3  @Pipe({
4    name: 'cpf',
5  })
6  export class CpfPipe implements PipeTransform {
7    public transform(cpf: string): string {
8      if (!cpf || cpf.trim() === '') {
9        return '';
10      }
11      let formatted: string;
12      formatted = cpf.substring(0, 3) + '.';
13      formatted = formatted + cpf.substring(3, 6) + '.';
14      formatted = formatted + cpf.substring(6, 9) + '-';
15      formatted = formatted + cpf.substring(9, 11);
16      return formatted;
17    }
18  }
```

- Build e deploy da aplicação.