Criando um app Angular e consumindo uma API REST [3 de 3]

https://medium.com/@andrewchanm/criando-um-app-angular-7-e-consumindo-uma-api-rest-3-de-3-7d3b22aa09a6



Olá pessoal, sou o Andrew desenvolvedor .net, sempre pensei em compartilhar meu conhecimento, esse é minha primeira série de posts e escolhi Angular 7, então podem fazer críticas e sugestões, será muito bom o feedback.

Primeiramente peço desculpas pela demora em escrever a Parte 3, Para quem não viu a Parte 2 segue o link, nessa parte vamos desenvolver os componentes, o objetivo é que eles se comunique com a nossa API REST.

1. Componente de Lista de Veiculos

Abra e edite o arquivo "src/app/ veiculos /veiculos.component.ts" aqui nós vamos importar nossa servisse, que foi criada no artigo Parte 2 segue o **link**,

```
import { VeiculoService } from 'src/app/services/veiculo.service';
```

Este componente chama três métodos VeiculoService:

- getAll()
- deleteAll()
- findByDescricao(this.descricao)

components/veiculos /veiculos-list.component.ts

```
import { Component, OnInit } from '@angular/core';
import { VeiculoService } from 'src/app/services/veiculo.service';

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

    veiculos: any;
    currentVeiculo = null;
    currentIndex = -1;
    descricao = '';

    constructor(private veiculoService: VeiculoService) { }

    ngOnInit(): void {
        this.retrieveVeiculos();
    }
}
```

```
retrieveVeiculos(): void {
  this.veiculoService.getAll()
    .subscribe(
      data => {
        this.veiculos = data;
        console.log(data);
      },
      error => {
       console.log(error);
      });
}
refreshList(): void {
  this.retrieveVeiculos();
  this.currentVeiculo = null;
  this.currentIndex = -1;
setActiveVeiculo(veiculo, index): void {
  this.currentVeiculo = veiculo;
  this.currentIndex = index;
removeAllVeiculos(): void {
  this.veiculoService.deleteAll()
    .subscribe(
      response => {
        console.log(response);
        this.retrieveVeiculos();
      },
      error => {
        console.log(error);
      });
}
searchDescricao(): void {
  this.veiculoService.findByDescricao(this.descricao)
    .subscribe(
      data => {
        this.veiculos = data;
        console.log(data);
      },
      error => {
       console.log(error);
      });
}
```

components/tutorials-list/tutorials-list.component.html

```
<input type="text" class="form-control"</pre>
placeholder="Search by descricao" [(ngModel)]="descricao"/>
                                     <div class="input-group-append">
                                             <button class="btn btn-outline-</pre>
secondary" type="button" (click)="searchDescricao()">Search</button>
                                     </div>
                              </div>
                      </div>
                      <div class="col-md-6">
                              <h4>Lista de Veículos</h4>
                              <select multiple class="form-control"</pre>
id="listVeiculos">
                                <option *ngFor="let veiculo of veiculos; let i =</pre>
index" [class.active] = " i == currentIndex" (click) = "setActiveVeiculo (veiculo,
i) ">{{ veiculo.descricao }}</option>
                              </select>
                              <button class="m-1 btn btn-sm btn-danger"</pre>
(click) = "removeAllVeiculos()">Elimina Todos</button>
                      </div>
                      <div class="col-md-6">
                              <div *ngIf="currentVeiculo">
                                     <h4>Veiculo</h4>
                                     <div><label><strong>Chapa:</strong></label>{{
currentVeiculo.chapa }}</div>
                                     <div><label><strong>Marca:</strong></label>{{
currentVeiculo.marca }}</div>
                                     <div><label><strong>Ano:</strong></label>{{
currentVeiculo.ano }}</div>
       <div><label><strong>Descrição:</strong></label>{{ currentVeiculo.descricao}
<div><label><strong>Vendido:</strong></label>{{ currentVeiculo.vendido}
} < /div>
       <div><label><strong>Cadastrado:</strong></label>{{ currentVeiculo.created}
}}</div>
       <div><label><strong>Atualizado:</strong></label>{{ currentVeiculo.updated}
}}</div>
                                     <a class="m-1 btn btn-sm btn-warning"
routerLink="/veiculos/{{ currentVeiculo.id }}">Edit</a>
                              </div>
                              <div *ngIf="!currentVeiculo">
                                     <br />
                                     Por favor click em um Veículo...
                              </div>
                      </div>
              </div>
       </div>
</div>
```

2. Componente de inclusão de Produto

Este componente tem um formulário para submeter um novo. O método VEiculoService.create() é chamado.

Abra e edite o arquivo "src/app/veiculo-novo/veiculo-novo.component.ts". Aqui nós vamos importa algumas coisas:

components/add-tutorial/add-tutorial.component.ts

```
import { Component, OnInit } from '@angular/core';
import { VeiculoService } from 'src/app/services/veiculo.service';
@Component({
 selector: 'app-veiculo-novo',
  templateUrl: './veiculo-novo.component.html',
 styleUrls: ['./veiculo-novo.component.css']
})
export class VeiculoNovoComponent implements OnInit {
    chapa: '';
    marca: '';
    ano: '';
    descricao: '';
   vendido: false;
    // message = '';
    veiculo = {
       chapa: '',
        marca: '',
       ano: '',
       descricao: '',
       vendido: false
    };
      submitted = false;
      constructor(private veiculoService: VeiculoService) { }
      ngOnInit(): void {
      saveVeiculo(): void {
        const data = {
              chapa: this.chapa,
              marca: this.marca,
              ano: this.ano,
              descricao: this.descricao,
              vendido: this.vendido
        };
        this.veiculoService.create(data)
          .subscribe(
            response => {
              console.log(response);
              // this.message = 'O novo veiculo foi cadastrado!';
              this.submitted = true;
            },
            error => {
              console.log(error);
            });
      }
      newVeiculo(): void {
          this.veiculo = {
              chapa: '',
              marca: '',
              ano: '',
              descricao: '',
              vendido: false
          };
```

```
this.submitted = false;
}
```

components/add-tutorial/add-tutorial.component.html

```
<div class="submit-form">
    <div *ngIf="!submitted">
        <div class="card">
            <div class="card-header">
                 <div class="clearfix">
                     <h5 class="card-title aw-titulo-card">Novo Veículo</h5>
                    <a class="btn btn-link aw-link-card" href="#">Voltar para
pesquisa</a>
                </div>
            </div>
            <div class="card-body">
                <div class="form-group row">
                     <label for="chapa" class="col-sm-2 col-form-</pre>
label">Chapa</label>
                     <div class="col-sm-2">
                         <input type="text" class="form-control" required</pre>
[(ngModel)]="chapa" name="chapa">
                    </div>
                </div>
                <div class="form-group row">
                     <label for="marca" class="col-sm-2 col-form-</pre>
label">Marca</label>
                     <div class="col-sm-2">
                         <select class="custom-select my-1 mr-sm-2"</pre>
[(ngModel)]="marca" name="marca">
                             <option selected>Choose...
                             <option value="CHEVROLET">Chevrolet</option>
                             <option value="FORD">Ford</option>
                             <option value="HONDA">Honda</option>
                             <option value="VOLKSWAGEN">VolksWagen</option>
                         </select>
                     </div>
                </div>
                <div class="form-group row">
                     <label for="ano" class="col-sm-2 col-form-label">Ano</label>
                     <div class="col-sm-2">
                         <input type="text" class="form-control" [(ngModel)]="ano"</pre>
name="ano">
                     </div>
                </div>
                <div class="form-group row">
                     <label for="descricao" class="col-sm-2 col-form-</pre>
label">Descrição</label>
                    <div class="col-sm-6">
                         <input type="text" class="form-control"</pre>
[(ngModel)]="descricao" name="descricao">
                     </div>
                </div>
                <div class="form-group row">
                     <label for="vendido" class="col-sm-2 col-form-</pre>
label">Vendido</label>
                     <div class="col-sm-2">
                         <select class="form-control" [(ngModel)]="vendido"</pre>
name="vendido">
                             <option value="false">Não</option>
```

```
<option value="true">Sim</option>
                         </select>
                    </div>
                </div>
                <div class="form-group row">
                    <div class="col-sm-2">
                         <button (click)="saveVeiculo()" type="submit" class="btn</pre>
btn-primary">Salvar</button>
                                     <!-- <p>{{ message }} -->
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div *ngIf="submitted">
        <h4>You submitted successfully!</h4>
        <button class="btn btn-success" (click)="newVeiculo()">Add</button>
    </div>
</div>
```

3. Componente de edição de Produto

O componente de edição é muito parecido com o do Novo, com a diferença que recebe o parâmetro **id** da rota e carrega os dados no formulário.

Abra e edite o arquivo "src/app/produto-editar/produto-editar.component.ts".

```
import { Component, OnInit } from '@angular/core';
import { VeiculoService } from 'src/app/services/veiculo.service';
import { ActivatedRoute, Router } from '@angular/router';
@Component({
  selector: 'app-veiculo-editar',
  templateUrl: './veiculo-editar.component.html',
  styleUrls: ['./veiculo-editar.component.css']
})
export class VeiculoEditarComponent implements OnInit {
  currentVeiculo = null;
 message = '';
  constructor (
    private veiculoService: VeiculoService,
    private route: ActivatedRoute,
    private router: Router) { }
  ngOnInit(): void {
    this.message = '';
    this.getVeiculo(this.route.snapshot.paramMap.get('id'));
  getVeiculo(id): void {
    this.veiculoService.get(id)
      .subscribe(
        data => {
          this.currentVeiculo = data;
          console.log(data);
        },
        error => {
          console.log(error);
        });
```

```
updateVeiculo(): void {
  this.veiculoService.update(this.currentVeiculo.id, this.currentVeiculo)
    .subscribe(
      response => {
        console.log(response);
        this.message = 'O veiculo foi atualizado com sucesso!';
      error => {
        console.log(error);
      });
}
updateVendido(status): void {
  const data = {
            vendido: status
  };
  this.veiculoService.sale(this.currentVeiculo.id, data)
    .subscribe(
      response => {
        this.currentVeiculo.vendido = status;
        console.log(response);
        this.message = 'O veiculo foi atualizado com sucesso!';
      },
      error => {
        console.log(error);
      });
deleteVeiculo(): void {
  this.veiculoService.delete(this.currentVeiculo.id)
    .subscribe(
      response => {
        console.log(response);
        this.router.navigate(['/veiculos']);
      },
      error => {
        console.log(error);
      });
}
```

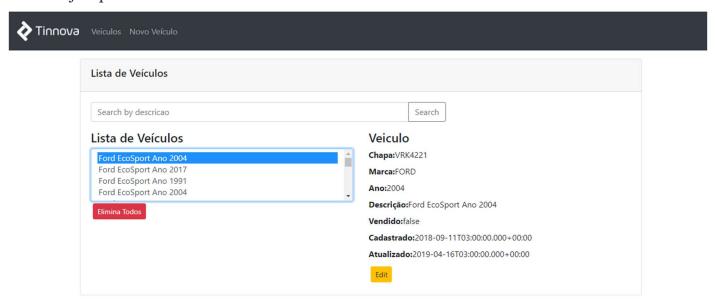
Em Seguida, Abra e edite o arquivo "src/app/produto-editar/produto-editar.component.html".

```
<div class="submit-form">
    <div *ngIf="currentVeiculo" class="edit-form">
        <div class="card">
            <div class="card-header">
                <div class="clearfix">
                    <h5 class="card-title aw-titulo-card">Veículo</h5>
                    <a class="btn btn-link aw-link-card" href="#">Voltar para
pesquisa</a>
                </div>
            </div>
            <div class="card-body">
                <form>
                                     <div class="form-group row">
                                             <label for="chapa" class="col-sm-2 col-</pre>
form-label">Chapa</label>
                                             <div class="col-sm-2">
```

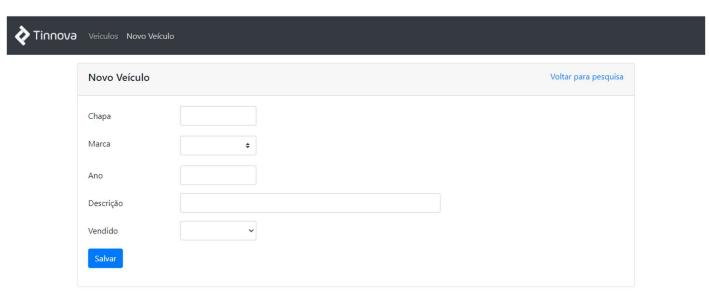
```
<input type="text" class="form-</pre>
control" required [(ngModel)]="currentVeiculo.chapa" name="chapa">
                                              </div>
                                       </div>
                                       <div class="form-group row">
                                              <label for="marca" class="col-sm-2 col-</pre>
form-label">Marca</label>
                                               <div class="col-sm-2">
                                                      <select class="custom-select</pre>
my-1 mr-sm-2" [(ngModel)]="currentVeiculo.marca" name="marca">
                                                               <option</pre>
selected>Choose...
                                                              <option</pre>
value="CHEVROLET">Chevrolet
                                                              <option</pre>
value="FORD">Ford</option>
                                                              <option</pre>
value="HONDA">Honda
                                                              <option</pre>
value="VOLKSWAGEN">VolksWagen</option>
                                                      </select>
                                              </div>
                                       </div>
                                       <div class="form-group row">
                                               <label for="ano" class="col-sm-2 col-</pre>
form-label">Ano</label>
                                               <div class="col-sm-2">
                                                      <input type="text" class="form-</pre>
control" [(ngModel)]="currentVeiculo.ano" name="ano">
                                              </div>
                                       </div>
                                       <div class="form-group row">
                                              <label for="descricao" class="col-sm-2</pre>
col-form-label">Descrição</label>
                                              <div class="col-sm-6">
                                                      <input type="text" class="form-</pre>
control" [(ngModel)]="currentVeiculo.descricao" name="descricao">
                                              </div>
                                       </div>
                                       <div class="form-group row">
                                              <label for="vendido" class="col-sm-2</pre>
col-form-label">Vendido</label>
                          {{ currentVeiculo.vendido ? "Sim" : "Não" }}
                                               <!-- <div class="col-sm-2">
                                                       <select class="form-control"</pre>
[(ngModel)]="currentVeiculo.vendido" name="vendido">
                                                              <option</pre>
value="false">Não</option>
                                                              <option</pre>
value="true">Sim</option>
                                                      </select>
                                              </div> -->
                                       </div>
                               </form>
                               <button *ngIf="currentVeiculo.vendido" class="badge</pre>
badge-primary mr-2" (click) = "updateVendido (false) ">Cancelar Venda < /button>
                               <button *ngIf="!currentVeiculo.vendido" class="badge</pre>
badge-primary mr-2"(click) = "updateVendido(true)"> Vender < / button>
                               <button class="badge badge-danger mr-2"</pre>
(click) = "deleteVeiculo()" > Delete < / button >
```

4. Considerações Finais

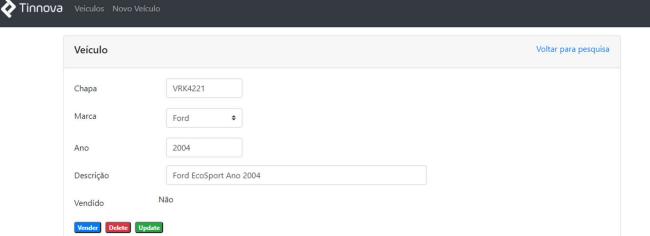
Conseguimos finalizar o objetivo inicial de criar um app Angular comunicando com uma API REST, um completo CRUD, entendemos como as rotas passam os parâmetros, no final a aplicação ficou show! veja aqui:



Listagem de Veículos



Cadastra um Novo Veículo



Edita um Novo Veículo

Referências

Entendendo RxJS Observable com Angular

Após passar por um projeto com Angular 2(ou somente Angular, para os mais íntimos) posso dizer que: É um framework com...

medium.com

Documentação Oficial Angular Schematics Material Angular Service O que é REST?