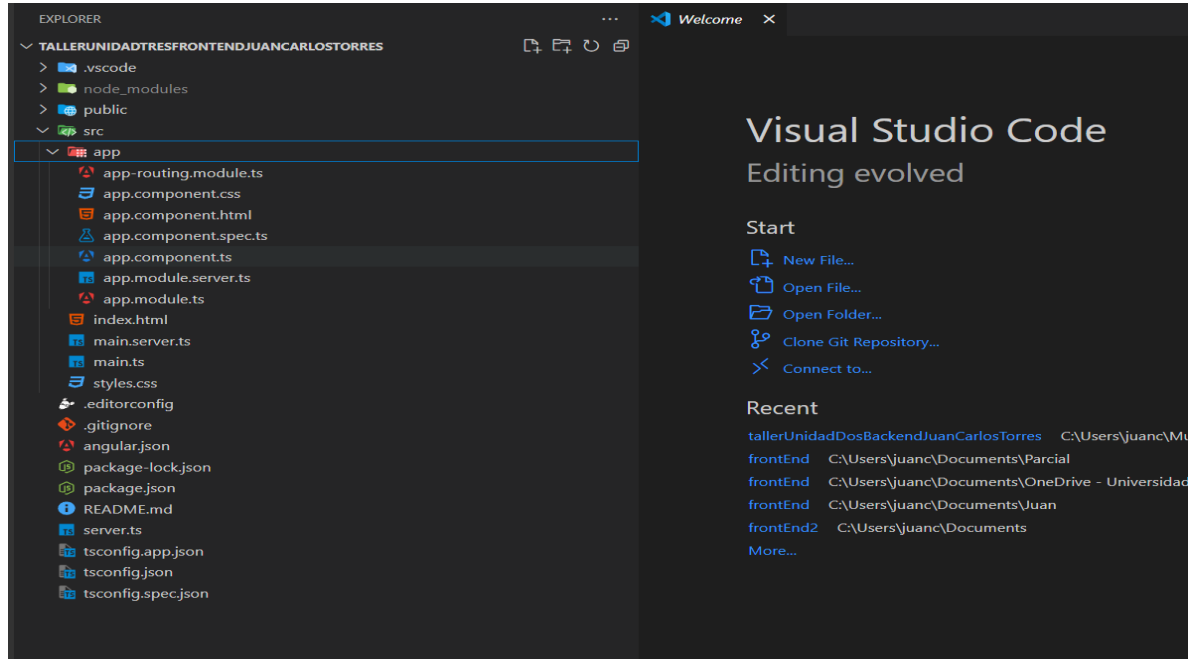


Comienzo a crear mi proyecto frontEnd el cual interactuara con mi proyecto backEnd

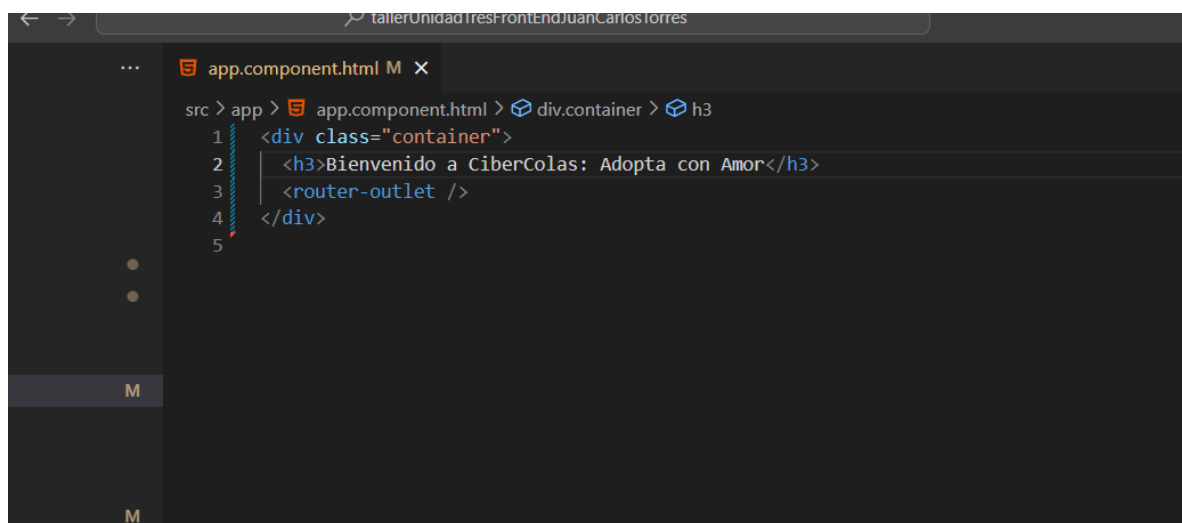
```
C:\Users\juanc\Documents\Diplomado2024>ng new tallerUnidadTresFrontEndJuanCarlosTorres --no-standalone
? Which stylesheet format would you like to use? (Use arrow keys)
> CSS [ https://developer.mozilla.org/docs/Web/CSS ]
Sass (SCSS) [ https://sass-lang.com/documentation/syntax#scss ]
Sass (Indented) [ https://sass-lang.com/documentation/syntax#the-indented-syntax ]
Less [ http://lesscss.org ]
```

```
C:\Users\juanc\Documents\Diplomado2024>ng new tallerUnidadTresFrontEndJuanCarlosTorres --no-standalone
? Which stylesheet format would you like to use? CSS [ https://developer.mozilla.org/docs/Web/CSS ]
? Do you want to enable Server-Side Rendering (SSR) and Static Site Generation (SSG/Prerendering)? (y/n)
```

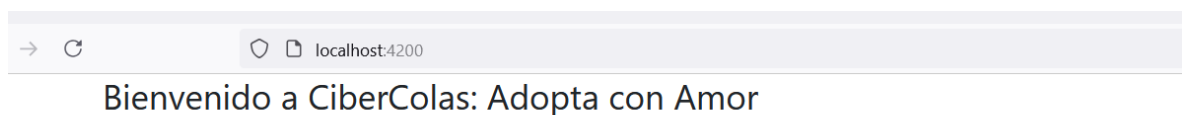
```
C:\Users\juanc\Documents\Diplomado2024>ng new tallerUnidadTresFrontEndJuanCarlosTorres --no-standalone
? Which stylesheet format would you like to use? CSS [ https://developer.mozilla.org/docs/Web/CSS ]
? Do you want to enable Server-Side Rendering (SSR) and Static Site Generation (SSG/Prerendering)? yes
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/angular.json (3243 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/package.json (1380 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/README.md (1128 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/tsconfig.json (1045 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/.editorconfig (290 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/.gitignore (629 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/tsconfig.app.json (504 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/tsconfig.spec.json (449 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/server.ts (1809 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/.vscode/extensions.json (134 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/.vscode/launch.json (490 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/.vscode/tasks.json (980 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/main.ts (256 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/index.html (339 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/styles.css (81 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/main.server.ts (71 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app-routing.module.ts (255 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.module.ts (467 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.component.html (20239 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.component.spec.ts (1180 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.component.ts (251 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.component.css (0 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/src/app/app.module.server.ts (332 bytes)
CREATE tallerUnidadTresFrontEndJuanCarlosTorres/public/favicon.ico (15086 bytes)
\ Installing packages (npm)...
```



```
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng add @ng-bootstrap/ng-bootstrap
✔ trap
✔ Determining Package Manager
  > Using package manager: npm
✔ Searching for compatible package version
  > Found compatible package version: @ng-bootstrap/ng-bootstrap@17.0.1.
✔ Loading package information from registry
✔ Confirming installation
✔ Installing package
UPDATE package.json (1521 bytes)
✔ Packages installed successfully.
UPDATE src/app/app.module.ts (540 bytes)
UPDATE angular.json (3435 bytes)
UPDATE src/main.ts (301 bytes)
UPDATE tsconfig.app.json (532 bytes)
UPDATE tsconfig.spec.json (477 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>
```



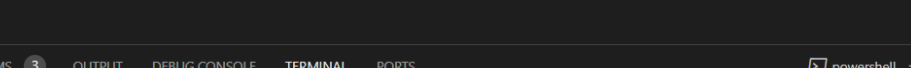
```
src > app > app.component.html > div.container > h3
1 <div class="container">
2   <h3>Bienvenido a CiberColas: Adopta con Amor</h3>
3   <router-outlet />
4 </div>
5
```



localhost:4200

Bienvenido a CiberColas: Adopta con Amor

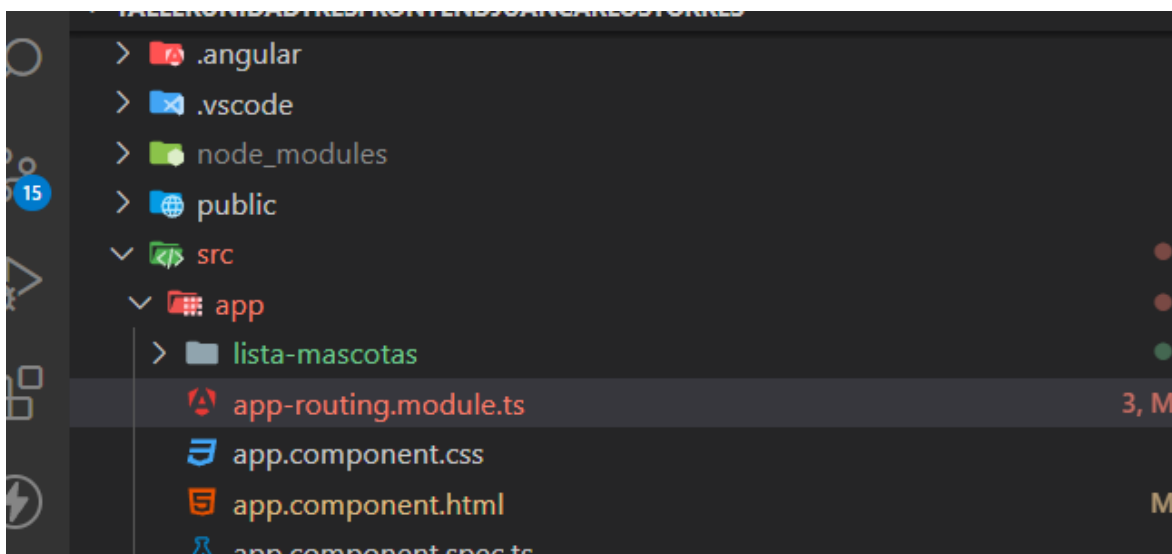
Vamos a generar un nuevo componente, crear una nueva carpeta llamada lista-mascotas



```

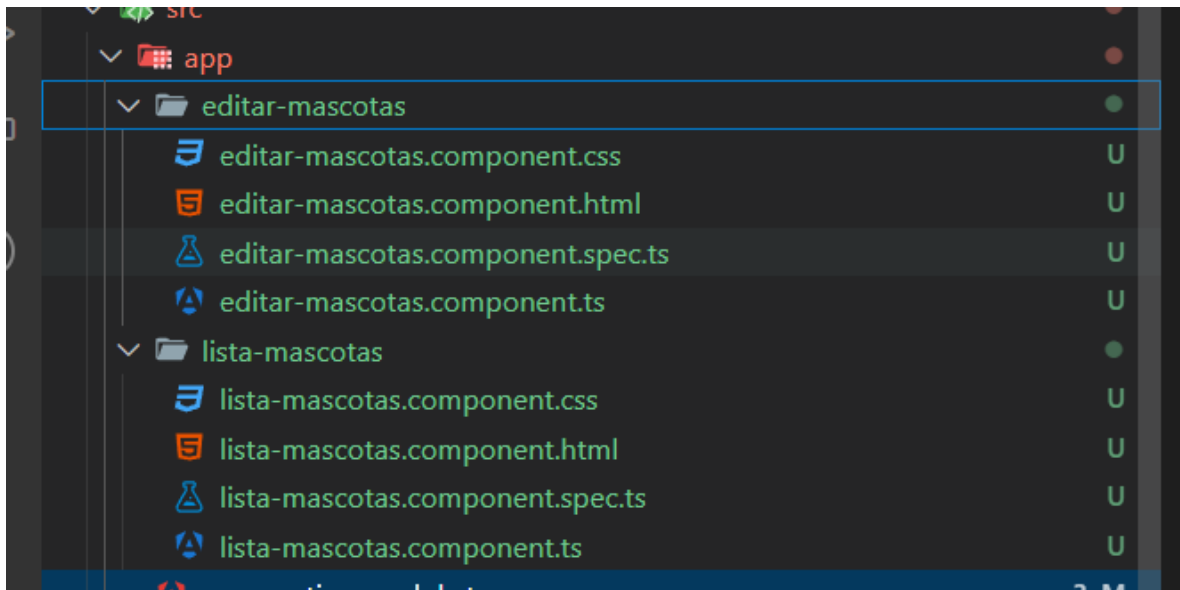
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng generate component lista-mascotas
CREATE src/app/lista-mascotas/lista-mascotas.component.html (30 bytes)
CREATE src/app/lista-mascotas/lista-mascotas.component.spec.ts (670 bytes)
CREATE src/app/lista-mascotas/lista-mascotas.component.ts (240 bytes)
CREATE src/app/lista-mascotas/lista-mascotas.component.css (0 bytes)
UPDATE src/app/app.module.ts (654 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>

```



Vamos a generar un nuevo componente, crear una nueva carpeta llamada editar-mascotas

```
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng generate component editar-mascotas
CREATE src/app/editar-mascotas/editar-mascotas.component.html (31 bytes)
CREATE src/app/editar-mascotas/editar-mascotas.component.spec.ts (677 bytes)
CREATE src/app/editar-mascotas/editar-mascotas.component.ts (244 bytes)
CREATE src/app/editar-mascotas/editar-mascotas.component.css (0 bytes)
UPDATE src/app/app.module.ts (772 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>
```



Vamos a configurar el enrutamiento de la aplicación con tres rutas específicas

```
src > app > app-routing.module.ts > ...
1  import { NgModule } from '@angular/core';
2  import { RouterModule, Routes } from '@angular/router';
3  import { ListaMascotasComponent } from '../lista-mascotas/lista-mascotas.component';
4  import { EditarMascotasComponent } from '../editar-mascotas/editar-mascotas.component';
5
6  const routes: Routes = [
7    {path: 'mascotas', component: ListaMascotasComponent},
8    {path: 'mascotas/editar/:idMascota', component: EditarMascotasComponent},
9    {path: 'mascotas/agregar', component: EditarMascotasComponent},
10   { path: '**', redirectTo: '/mascotas', pathMatch: 'full' },
11 ];
12
13 @NgModule({
14   imports: [RouterModule.forRoot(routes)],
15   exports: [RouterModule]
16 })
17 export class AppRoutingModule { }
18
```

Aquí definimos un modelo de datos (MascotaModel) con tres propiedades (id, nombre, edad, clase de animal, peso y color).

```
src > app > shared > ts mascota.model.ts > MascotaModel
1  export class MascotaModel {
2      constructor(
3          public id: string,
4          public nombre: string,
5          public edad: string,
6          public claseAnimal: string,
7          public peso: string,
8          public color: string
9      ) {}
10 }
```

Se crea un nuevo servicio llamado MascotaService dentro de la carpeta shared. Este servicio se usa para centralizar la lógica de negocio.

```
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng generate service shared/mascota
CREATE src/app/shared/mascota.service.spec.ts (378 bytes)
CREATE src/app/shared/mascota.service.ts (145 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>
```

```
ts mascota.model.ts
ts mascota.service.ts
ts app-routing.module.ts
```

Este servicio (MascotaService) gestiona operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para las mascotas.

```
// src/app/shared/mascota.service.ts

import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { Observable } from 'rxjs';
import { MascotaModel } from '../mascota.model';

@Injectable({
  providedIn: 'root',
})
export class MascotaService {
  BASE_URL = 'http://localhost:4000'; // Asegúrate de que esta URL apunte a tu backend

  constructor(private http: HttpClient) {}

  // Obtener todas las mascotas
  obtenerMascotas(): Observable<MascotaModel[]> {
    return this.http.get<MascotaModel[]>(`${this.BASE_URL}/mascotas/buscarTodo`);
  }

  // Obtener una mascota por ID
  obtenerMascota(idMascota: string): Observable<MascotaModel> {
    return this.http.get<MascotaModel>(`${this.BASE_URL}/mascotas/buscar/${idMascota}`);
  }

  // Crear una nueva mascota
  agregarMascota(mascota: MascotaModel): Observable<string> {
    return this.http.post<string>(`${this.BASE_URL}/mascotas/crear`, mascota);
  }

  // Actualizar una mascota existente
  actualizarMascota(mascota: MascotaModel): Observable<string> {
    return this.http.put<string>(`${this.BASE_URL}/mascotas/actualizar/${mascota.id}`, mascota);
  }
}
```

```
    // Eliminar una mascota por ID
    borrarMascota(idMascota: string): Observable<string> {
      return this.http.delete<string>(`${this.BASE_URL}/mascotas/eliminar/${idMascota}`);
    }
  }
}
```

Vamos a diseñar la parte que permite tener lista de mascotas en una tabla con opciones para agregar, editar y eliminar. Resumen de sus características:

```
<div class="container">
  <h3>Lista de Mascotas</h3>
  <a class="btn btn-primary" [routerLink]="['/mascotas/agregar']">Nueva Mascota</a>
  <br><br>

  <!-- Tabla de Mascotas -->
  <table class="table table-striped">
    <thead>
      <tr>
        <th>ID</th>
        <th>Nombre</th>
        <th>Edad</th>
        <th>Clase de Animal</th>
        <th>Peso</th>
        <th>Color</th>
        <th>Acciones</th>
      </tr>
    </thead>
    <tbody>
      <tr *ngFor="let mascota of mascotas">
        <td>{{ mascota.id }}</td>
        <td>{{ mascota.nombre }}</td>
        <td>{{ mascota.edad }}</td>
        <td>{{ mascota.claseAnimal }}</td>
        <td>{{ mascota.peso }}</td>
        <td>{{ mascota.color }}</td>
        <td>
          <a class="btn btn-info" [routerLink]="['/mascotas/editar', mascota.id]">Editar</a>
          <button class="btn btn-danger" (click)="borrarMascota(mascota.id)">Eliminar</button>
          <button class="btn btn-success" (click)="solicitarMascota(mascota.id)">Solicitar Mascota</button>
        </td>
      </tr>
    </tbody>
  </table>
</div>
```

Levantamos el servicio del frontEnd para ver si esta funcionando

PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> npm run start

```
> taller-unidad-tres-front-end-juan-carlos-torres@0.0.0 start
> ng serve
```

Browser bundles

Initial chunk files	Names	Raw size
styles.css	styles	277.63 kB
polyfills.js	polyfills	92.96 kB
main.js	main	12.27 kB
Initial total		382.87 kB

Server bundles

Initial chunk files	Names	Raw size
polyfills.server.mjs	polyfills.server	575.68 kB
main.server.mjs	main.server	12.77 kB
render-utils.server.mjs	render-utils.server	472 bytes

Hasta el momento así tenemos el frotEnd

Bienvenido a CiberColas: Adopta con Amor

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	Lazi	2	Perro	2 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota

Vamos a crear un formulario que permite ingresar y editar los detalles de una mascota, asegurándose de que todos los campos sean válidos antes de permitir el envío

```
<!-- src/app/editar-mascotas/editar-mascotas.component.html -->

<form (ngSubmit)="onSubmit()" #mascotaForm="ngForm">
  <div class="form-group">
    <label for="nombre">Nombre</label>
    <input type="text" id="nombre" class="form-control" required [(ngModel)]="mascota.nombre" name="nombre">
  </div>

  <div class="form-group">
    <label for="edad">Edad</label>
    <input type="number" id="edad" class="form-control" required [(ngModel)]="mascota.edad" name="edad">
  </div>

  <div class="form-group">
    <label for="claseAnimal">Clase de Animal</label>
    <input type="text" id="claseAnimal" class="form-control" required [(ngModel)]="mascota.claseAnimal" name="claseAnimal">
  </div>

  <div class="form-group">
    <label for="peso">Peso</label>
    <input type="text" id="peso" class="form-control" required [(ngModel)]="mascota.peso" name="peso">
  </div>

  <div class="form-group">
    <label for="color">Color</label>
    <input type="text" id="color" class="form-control" required [(ngModel)]="mascota.color" name="color">
  </div>

  <button type="submit" class="btn btn-primary" [disabled]="!mascotaForm.form.valid">Enviar</button>
</form>
```



```

1 // src/app/editar-mascotas/editar-mascotas.component.ts
2
3 import { Component, OnInit } from '@angular/core';
4 import { MascotaModel } from '../shared/mascota.model';
5 import { MascotaService } from '../shared/mascota.service';
6 import { ActivatedRoute, Router } from '@angular/router';
7
8 @Component({
9   selector: 'app-editar-mascotas',
10  templateUrl: './editar-mascotas.component.html',
11  styleUrls: ['./editar-mascotas.component.css'],
12 })
13 export class EditarMascotasComponent implements OnInit {
14   idMascota: string = '';
15   // Inicializar el objeto con todas las propiedades requeridas
16   mascota: MascotaModel = new MascotaModel('', '', '', '', '', '');
17
18   constructor(private mascotaService: MascotaService, private route: ActivatedRoute, private router: Router) {}
19
20   ngOnInit(): void {
21     this.idMascota = this.route.snapshot.params['idMascota'];
22     if (this.idMascota) {
23       this.obtenerMascota();
24     }
25   }
26
27   obtenerMascota(): void {
28     this.mascotaService.obtenerMascota(this.idMascota).subscribe({
29       next: (data) => {
30         this.mascota = data;
31       },
32       error: (err) => {
33         console.error(`Error al obtener la mascota: ${err}`);
34       },
35     });
36   }
37

```

```

38   onSubmit(): void {
39       console.log('Datos enviados al backend:', this.mascota); // Verificar qué datos se están enviando
40       if (
41           this.mascota.nombre &&
42           this.mascota.edad &&
43           this.mascota.claseAnimal &&
44           this.mascota.peso &&
45           this.mascota.color
46       ) {
47           if (this.idMascota) {
48               this.mascotaService.actualizarMascota(this.mascota).subscribe({
49                   next: () => {
50                       console.log('Mascota actualizada exitosamente.');
51                       this.router.navigate(['/mascotas']);
52                   },
53                   error: (err) => {
54                       console.error(`Error al actualizar la mascota: ${err}`);
55                   },
56               });
57           } else {
58               this.mascotaService.agregarMascota(this.mascota).subscribe({
59                   next: () => {
60                       console.log('Mascota agregada exitosamente.');
61                       this.router.navigate(['/mascotas']);
62                   },
63                   error: (err) => {
64                       console.error(`Error al agregar la mascota: ${err}`);
65                   },
66               });
67           }
68       } else {
69           alert('Por favor, completa todos los campos antes de enviar.');
```

```

7   import { ListaMascotasComponent } from './lista-mascotas/lista-mascotas.component';
8   import { EditarMascotasComponent } from './editar-mascotas/editar-mascotas.component';
9   import { FormsModule } from '@angular/forms';
10
11   @NgModule({
12       declarations: [
13           AppComponent,
14           ListaMascotasComponent,
15           EditarMascotasComponent
16       ],
17       imports: [
18           BrowserModule,
19           AppRoutingModule,
20           NgbModule,
21           FormsModule
22       ],
23       providers: [
24           provideClientHydration()
25       ],
26       bootstrap: [AppComponent]
27   })

```

Ya añade mascota

Bienvenido a CiberColas: Adopta con Amor

Nombre

Ciro

Edad

5

Clase de Animal

Perro

Peso

3 kilo

Color

Cafe

Enviar

Bienvenido a CiberColas: Adopta con Amor

Lista de Mascotas

Nueva Mascota

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	Lazi	2	Perro	2 kilo	Azul	<div>EditarEliminarSolicitar Mascota</div>
13	Kloi	2	Perro	8 kilo	Castaño	<div>EditarEliminarSolicitar Mascota</div>
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	<div>EditarEliminarSolicitar Mascota</div>
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	<div>EditarEliminarSolicitar Mascota</div>
17	Ciro	5	Perro	3 kilo	Cafe	<div>EditarEliminarSolicitar Mascota</div>

Ya edita mascota

Bienvenido a CiberColas: Adopta con Amor

Nombre

lobe

Edad

10

Clase de Animal

Perro

Peso

6 kilo

Color

Azul

Enviar

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	lobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
17	Ciro	5	Perro	3 kilo	Cafe	Editar Eliminar Solicitar Mascota

Ya elimina, vamos a eliminar la ultima fila

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	lobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
17	Ciro	5	Perro	3 kilo	Cafe	Editar Eliminar Solicitar Mascota

Ya se elimino

Bienvenido a CiberColas: Adopta con Amor

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	lobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota

Ahora vamos a configurar la tabla de solicitudes de mascota, de tal manera que se permita solicitar y gestionar

```
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng generate component crear-solicitud
CREATE src/app/crear-solicitud/crear-solicitud.component.html (31 bytes)
CREATE src/app/crear-solicitud/crear-solicitud.component.spec.ts (677 bytes)
CREATE src/app/crear-solicitud/crear-solicitud.component.ts (244 bytes)
CREATE src/app/crear-solicitud/crear-solicitud.component.css (0 bytes)
UPDATE src/app/app.module.ts (1043 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>
```

Crearemos un formulario que se utiliza para crear o editar solicitudes para una mascota. Los datos que se recopilan incluyen la cédula, nombre, apellido y dirección del solicitante.

```
src > app > crear-solicitud > crear-solicitud.component.html > div.container
1 <!-- src/app/crear-solicitud/crear-solicitud.component.html -->
2 <div class="container">
3   <h3>{{ isEditMode ? 'Editar' : 'Crear' }} Solicitud para Mascota ID: {{ solicitud.idMascota }}</h3>
4   <form (ngSubmit)="onSubmit()" #solicitudForm="ngForm">
5     <div class="form-group">
6       <label for="cedula">Cédula</label>
7       <input type="text" id="cedula" class="form-control" required [(ngModel)]="solicitud.cedula" name="cedula">
8     </div>
9
10    <div class="form-group">
11      <label for="nombre">Nombre</label>
12      <input type="text" id="nombre" class="form-control" required [(ngModel)]="solicitud.nombre" name="nombre">
13    </div>
14
15    <div class="form-group">
16      <label for="apellido">Apellido</label>
17      <input type="text" id="apellido" class="form-control" required [(ngModel)]="solicitud.apellido" name="apel
18    </div>
19
20    <div class="form-group">
21      <label for="direccion">Dirección</label>
22      <input type="text" id="direccion" class="form-control" required [(ngModel)]="solicitud.direccion" name="di
23    </div>
24
25    <button type="submit" class="btn btn-primary" [disabled]="!solicitudForm.form.valid">{{ isEditMode ? 'Actual
26  </form>
27 </div>
28
```

Codificamos este componente que gestiona la creación y edición de solicitudes de adopción de mascotas en función de si se tiene un idSolicitud (modo de edición) o un idMascota (modo de creación)

```
src / app / crear-solicitud / crear-solicitud.component.ts / CreateSolicitudComponent / OnSubmit
1 // src/app/crear-solicitud/crear-solicitud.component.ts
2
3 import { Component, OnInit } from '@angular/core';
4 import { ActivatedRoute, Router } from '@angular/router';
5 import { SolicitudService } from '../shared/solicitud.service';
6 import { SolicitudModel } from '../shared/solicitud.model';
7
8 @Component({
9   selector: 'app-crear-solicitud',
10  templateUrl: './crear-solicitud.component.html',
11  styleUrls: ['./crear-solicitud.component.css'],
12 })
13 export class CrearSolicitudComponent implements OnInit {
14   idMascota: string = '';
15   idSolicitud: string = ''; // ID de la solicitud para el modo de edición
16   isEditMode: boolean = false; // Indica si es modo de edición
17
18   solicitud: SolicitudModel = new SolicitudModel('', '', '', {
19     cedula: '',
20     nombre: '',
21     apellido: '',
22     direccion: ''
23   });
24
25   constructor(
26     private route: ActivatedRoute,
27     private solicitudService: SolicitudService,
28     private router: Router
29   ) {}
30
31   ngOnInit(): void {
32     // Obtener el ID de la solicitud de la ruta para determinar si es modo de edición
33     this.idSolicitud = this.route.snapshot.params['idSolicitud'];
```

```
31   ngOnInit(): void {
32     // Obtener el ID de la solicitud de la ruta para determinar si es modo de edición
33     this.idSolicitud = this.route.snapshot.params['idSolicitud'];
34     this.idMascota = this.route.snapshot.params['idMascota']; // Obtener ID de la mascota (opcional)
35
36     // Comprobar si estamos en modo de edición o creación
37     if (this.idSolicitud) {
38       this.isEditMode = true;
39       this.obtenerSolicitud(this.idSolicitud); // Llamar al método para obtener la solicitud y cargar los datos en
40     } else if (this.idMascota) {
41       // Si estamos en modo de creación, asignar el ID de la mascota
42       this.solicitud.idMascota = this.idMascota;
43     }
44   }
45
46   // Método para obtener la solicitud y cargar sus datos en el formulario
47   obtenerSolicitud(idSolicitud: string): void {
48     this.solicitudService.obtenerSolicitud(idSolicitud).subscribe({
49       next: (data) => {
50         // Asignar los datos obtenidos al modelo `solicitud`
51         this.solicitud = data;
52         // Si el objeto `persona` existe, asignar los datos correctamente
53         if (data.persona) {
54           this.solicitud.cedula = data.persona.cedula;
55           this.solicitud.nombre = data.persona.nombre;
56           this.solicitud.apellido = data.persona.apellido;
57           this.solicitud.direccion = data.persona.direccion;
58         }
59       },
60       error: (err) => {
61         console.error('Error al obtener la solicitud: ${err}');
```

```

// Método para manejar la creación o actualización de la solicitud
onSubmit(): void {
  if (this.isEditMode) {
    // Si estamos en modo de edición, llamar al servicio para actualizar la solicitud
    this.solicitudService.actualizarSolicitud(this.solicitud).subscribe({
      next: () => {
        console.log('Solicitud actualizada exitosamente.');
```

this.router.navigate(['/mascotas']); // Redirigir a la lista de mascotas
 },
 error: (err) => {
 console.error('Error al actualizar la solicitud: \${err}');
 },
 });
 } else {
 // Si estamos en modo de creación, llamar al servicio para crear una nueva solicitud
 this.solicitudService.crearSolicitud(this.solicitud).subscribe({
 next: () => {
 console.log('Solicitud creada exitosamente.');

this.router.navigate(['/mascotas']); // Redirigir a la lista de mascotas
 },
 error: (err) => {
 console.error('Error al crear la solicitud: \${err}');
 },
 });
 }
}

Vamos a crear el componente solicitud.model.ts

```

src > app > shared > ts solicitud.model.ts > SolicitModel
1 // src/app/shared/solicitud.model.ts
2
3 export class SolicitModel {
4   constructor(
5     public idSolicitud: string,
6     public cedulaPersona: string,
7     public idMascota: string,
8     public persona?: {
9       cedula: string;
10      nombre: string;
11      apellido: string;
12      direccion: string;
13    }, // Agregar la propiedad `persona`
14     public mascota?: {
15       id: string;
16       nombre: string;
17       edad: number;
18       claseAnimal: string;
19       peso: string;
20       color: string;
21     } // Agregar la propiedad `mascota`
22   ) {}
23
24   // O agregar las propiedades directamente en el modelo si se necesitan sin estructura adicional
25   public cedula?: string;
26   public nombre?: string;
27   public apellido?: string;
28   public direccion?: string;
29 }
30
31

```

Se crea un nuevo servicio llamado `SolicitudService` dentro de la carpeta `shared`.

```
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres> ng generate service shared/solicitud
CREATE src/app/shared/solicitud.service.spec.ts (388 bytes)
CREATE src/app/shared/solicitud.service.ts (147 bytes)
PS C:\Users\juanc\Documents\Diplomado2024\tallerUnidadTresFrontEndJuanCarlosTorres>
```

Vamos a codificar en `solicitud.service.ts`, donde creamos métodos para gestionar las solicitudes de adopción, conectándose al backend para **obtener**, **crear**, **actualizar** y **eliminar** solicitudes mediante peticiones.

```
src > app > shared > solicitud.service.ts > ...
1 // src/app/shared/solicitud.service.ts
2
3 import { Injectable } from '@angular/core';
4 import { HttpClient } from '@angular/common/http';
5 import { Observable } from 'rxjs';
6 import { SolicitudModel } from './solicitud.model';
7
8 @Injectable({
9   providedIn: 'root',
10 })
11 export class SolicitudService {
12   BASE_URL = 'http://localhost:4000'; // URL de tu backend
13
14   constructor(private http: HttpClient) {}
15
16   // Obtener la lista completa de solicitudes
17   obtenerSolicitudes(): Observable<SolicitudModel[]> {
18     return this.http.get<SolicitudModel[]>(`${this.BASE_URL}/solicitudes/buscarTodo`);
19   }
20
21   // Crear una nueva solicitud
22   crearSolicitud(solicitud: SolicitudModel): Observable<string> {
23     return this.http.post<string>(`${this.BASE_URL}/solicitudes/crear`, solicitud);
24   }
25
26   // Eliminar una solicitud por ID
27   eliminarSolicitud(idSolicitud: string): Observable<string> {
```



```

src > app > shared > solicitud.service.ts > ...
11 export class SolicitudService {
17   obtenerSolicitudes(): Observable<SolicitudModel[]> {
19   }
20
21   // Crear una nueva solicitud
22   crearSolicitud(solicitud: SolicitudModel): Observable<string> {
23     return this.http.post<string>(`${this.BASE_URL}/solicitudes/crear`, solicitud);
24   }
25
26   // Eliminar una solicitud por ID
27   eliminarSolicitud(idSolicitud: string): Observable<string> {
28     return this.http.delete<string>(`${this.BASE_URL}/solicitudes/eliminar/${idSolicitud}`);
29   }
30
31   // Obtener una solicitud por ID
32   obtenerSolicitud(idSolicitud: string): Observable<SolicitudModel> {
33     return this.http.get<SolicitudModel>(`${this.BASE_URL}/solicitudes/buscar/${idSolicitud}`);
34   }
35
36   // **Añadir este método para actualizar una solicitud**
37   actualizarSolicitud(solicitud: SolicitudModel): Observable<string> {
38     // Llamada PUT al backend con el ID de la solicitud y el objeto solicitud actualizado
39     return this.http.put<string>(`${this.BASE_URL}/solicitudes/actualizar/${solicitud.idSolicitud}`, solicitud);
40   }
41 }
42

```

Modificamos lista-mascotas.component.html para que mas abajo se visualice la tabla de solicitud con sus botone editar y eliminar

```

src > app > lista-mascotas > lista-mascotas.component.html > div.container
1 <!-- src/app/lista-mascotas/lista-mascotas.component.html -->
2
3 <div class="container">
4   <h3>Lista de Mascotas</h3>
5   <a class="btn btn-primary" [routerLink]="['/mascotas/agregar']">Nueva Mascota</a>
6   <br><br>
7
8   <!-- Tabla de Mascotas -->
9   <table class="table table-striped">
10     <thead>
11       <tr>
12         <th>ID</th>
13         <th>Nombre</th>
14         <th>Edad</th>
15         <th>Clase de Animal</th>
16         <th>Peso</th>
17         <th>Color</th>
18         <th>Acciones</th>
19       </tr>
20     </thead>
21     <tbody>
22       <tr *ngFor="let mascota of mascotas">
23         <td>{{ mascota.id }}</td>
24         <td>{{ mascota.nombre }}</td>
25         <td>{{ mascota.edad }}</td>
26         <td>{{ mascota.claseAnimal }}</td>
27         <td>{{ mascota.peso }}</td>
28         <td>{{ mascota.color }}</td>
29         <td>
30           <a class="btn btn-info" [routerLink]="['/mascotas/editar', mascota.id]">Editar</a>
31           <button class="btn btn-danger" (click)="borrarMascota(mascota.id)">Eliminar</button>
32           <button class="btn btn-success" (click)="solicitarMascota(mascota.id)">Solicitar Mascota</button>
33         </td>

```

```

        <td>{{ mascota.claseAnimal }}</td>
        <td>{{ mascota.peso }}</td>
        <td>{{ mascota.color }}</td>
        <td>
            <a class="btn btn-info" [routerLink]="['/mascotas/editar', mascota.id]">Editar</a>
            <button class="btn btn-danger" (click)="borrarMascota(mascota.id)">Eliminar</button>
            <button class="btn btn-success" (click)="solicitarMascota(mascota.id)">Solicitar Mascota</button>
        </td>
    </tr>
</tbody>
</table>

<br><br>

<!-- Tabla para Mostrar las Solicitudes -->
<h3>Lista de Solicitudes</h3>
<table class="table table-striped">
    <thead>
        <tr>
            <th>ID de Solicitud</th>
            <th>Cédula</th>
            <th>Nombre</th>
            <th>Apellido</th>
            <th>Dirección</th>
            <th>ID Mascota</th>
            <th>Nombre de la Mascota</th>
            <th>Acciones</th> <!-- Nueva columna para acciones -->
        </tr>
    </thead>

```

```

43     <thead>
44         <tr>
45             <th>ID Mascota</th>
46             <th>Nombre de la Mascota</th>
47             <th>Acciones</th> <!-- Nueva columna para acciones -->
48         </tr>
49     </thead>
50     <tbody>
51         <tr *ngFor="let solicitud of solicitudes">
52             <td>{{ solicitud.idSolicitud }}</td>
53             <td>{{ solicitud.persona?.cedula }}</td> <!-- Operador seguro para evitar errores -->
54             <td>{{ solicitud.persona?.nombre }}</td>
55             <td>{{ solicitud.persona?.apellido }}</td>
56             <td>{{ solicitud.persona?.direccion }}</td>
57             <td>{{ solicitud.idMascota }}</td>
58             <td>{{ solicitud.mascota?.nombre }}</td>
59             <td>
60                 <!-- Botón Editar Solicitud -->
61                 <button class="btn btn-info" (click)="editarSolicitud(solicitud.idSolicitud)">Editar Solicitud (ID: {{
62                     solicitud.idSolicitud
63                 }})</button>
64                 <!-- Botón Eliminar Solicitud -->
65                 <button class="btn btn-danger" (click)="eliminarSolicitud(solicitud.idSolicitud)">Eliminar Solicitud</button>
66             </td>
67         </tr>
68     </tbody>
69 </table>
70 </div>

```

Modificamos el archivo app-routing-module.ts

```
src > app > app-routing.module.ts > AppRoutingModule
1 // src/app/app-routing.module.ts
2
3 import { NgModule } from '@angular/core';
4 import { RouterModule, Routes } from '@angular/router';
5 import { ListaMascotasComponent } from '../lista-mascotas/lista-mascotas.component';
6 import { EditarMascotasComponent } from '../editar-mascotas/editar-mascotas.component';
7 import { CrearSolicitudComponent } from '../crear-solicitud/crear-solicitud.component';
8
9 const routes: Routes = [
10   { path: 'mascotas', component: ListaMascotasComponent },
11   { path: 'mascotas/editar/:idMascota', component: EditarMascotasComponent },
12   { path: 'mascotas/agregar', component: EditarMascotasComponent },
13   { path: 'solicitudes/crear/:idMascota', component: CrearSolicitudComponent }, // Ruta para crear solicitudes
14   { path: 'solicitudes/editar/:idSolicitud', component: CrearSolicitudComponent }, // Ruta para editar solicitudes
15   { path: '**', redirectTo: '/mascotas', pathMatch: 'full' },
16 ];
17
18 @NgModule({
19   imports: [RouterModule.forRoot(routes)],
20   exports: [RouterModule],
21 })
22 export class AppRoutingModule {}
23
```

Modificamos también lista-mascota.component.ts

Se verifica que realmente esta realizando solicitud

Bienvenido a CiberColas: Adopta con Amor

Crear Solicitud para Mascota ID: 16

Cédula
1098848

Nombre
Marcos

Apellido
Salazar

Dirección
Buenos Aires

Enviar Solicitud

Como podemos ver si esta realizando solicitudes, la solicitud N14 fue la que hice

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	lobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota

Lista de Solicitudes

ID de Solicitud	Cédula	Nombre	Apellido	Dirección	ID Mascota	Nombre de la Mascota	Acciones
11	10067553	Darwin	Castillo	Vargas	12	lobe	Editar Solicitud (ID: 11) Eliminar Solicitud
12	985647	Franco	Cortes	Nueva independencia	13	Kloi	Editar Solicitud (ID: 12) Eliminar Solicitud
13	898779	Yeison	Alfredo	Unión Victoria	13	Kloi	Editar Solicitud (ID: 13) Eliminar Solicitud
14	1098848	Marcos	Salazar	Buenos Aires	16	Kiro	Editar Solicitud (ID: 14) Eliminar Solicitud

Ahora vamos editar solicitud, se edito la solicitud N14

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	lobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota

Lista de Solicitudes

ID de Solicitud	Cédula	Nombre	Apellido	Dirección	ID Mascota	Nombre de la Mascota	Acciones
11	10067553	Darwin	Castillo	Vargas	12	lobe	Editar Solicitud (ID: 11) Eliminar Solicitud
12	985647	Franco	Cortes	Nueva independencia	13	Kloi	Editar Solicitud (ID: 12) Eliminar Solicitud
13	898779	Yeison	Alfredo	Unión Victoria	13	Kloi	Editar Solicitud (ID: 13) Eliminar Solicitud
14	10988488	Luis	Quiñones	Morrito	16	Kiro	Editar Solicitud (ID: 14) Eliminar Solicitud

Ahora vamos a eliminar solicitud numero 14

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	Iobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota

localhost:4200 dice

¿Estás seguro de que deseas eliminar esta solicitud?

[Aceptar](#)

[Cancelar](#)

Lista de Solicitudes

ID de Solicitud	Cédula	Nombre	Apellido	Dirección	ID Mascota	Nombre de la Mascota	Acciones
11	10067553	Darwin	Castillo	Vargas	12	Iobe	Editar Solicitud (ID: 11) Eliminar Solicitud
12	985647	Franco	Cortes	Nueva independencia	13	Kloi	Editar Solicitud (ID: 12) Eliminar Solicitud
13	898779	Yeison	Alfredo	Unión Victoria	13	Kloi	Editar Solicitud (ID: 13) Eliminar Solicitud
14	10988488	Luis	Quiñones	Morrito	16	Kiro	Editar Solicitud (ID: 14) Eliminar Solicitud

Ya se elimino

Bienvenido a CiberColas: Adopta con Amor

Lista de Mascotas

[Nueva Mascota](#)

ID	Nombre	Edad	Clase de Animal	Peso	Color	Acciones
12	Iobe	10	Perro	6 kilo	Azul	Editar Eliminar Solicitar Mascota
13	Kloi	2	Perro	8 kilo	Castaño	Editar Eliminar Solicitar Mascota
15	Laika	1	Gato	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota
16	Kiro	1	Loro	2 kilo	Blanco con rayas negra	Editar Eliminar Solicitar Mascota

Lista de Solicitudes

ID de Solicitud	Cédula	Nombre	Apellido	Dirección	ID Mascota	Nombre de la Mascota	Acciones
11	10067553	Darwin	Castillo	Vargas	12	Iobe	Editar Solicitud (ID: 11) Eliminar Solicitud
12	985647	Franco	Cortes	Nueva independencia	13	Kloi	Editar Solicitud (ID: 12) Eliminar Solicitud
13	898779	Yeison	Alfredo	Unión Victoria	13	Kloi	Editar Solicitud (ID: 13) Eliminar Solicitud

Bueno como podemos ver ya se tiene el frotEnd para los 2 módulos, gestión de mascota y gestión de solicitud. Nota si se quiere eliminar una mascota que se esta solicitada por un cliente no se puede borrar la mascota de la tabla, ya que el diseño de la tabla lo hice entidad relación, para el proyecto diseñe 3 tablas. Persona, mascotas y solicitudes. Personas y mascotas están ligadas en la tabla solicitud es por esa razón que no se puede eliminar una mascota mientras se tenga una solicitud, si se borra la solicitud se podrá eliminar la mascota.

No se pudo realizar una interfaz amigable para el usuario final por el corto tiempo.

