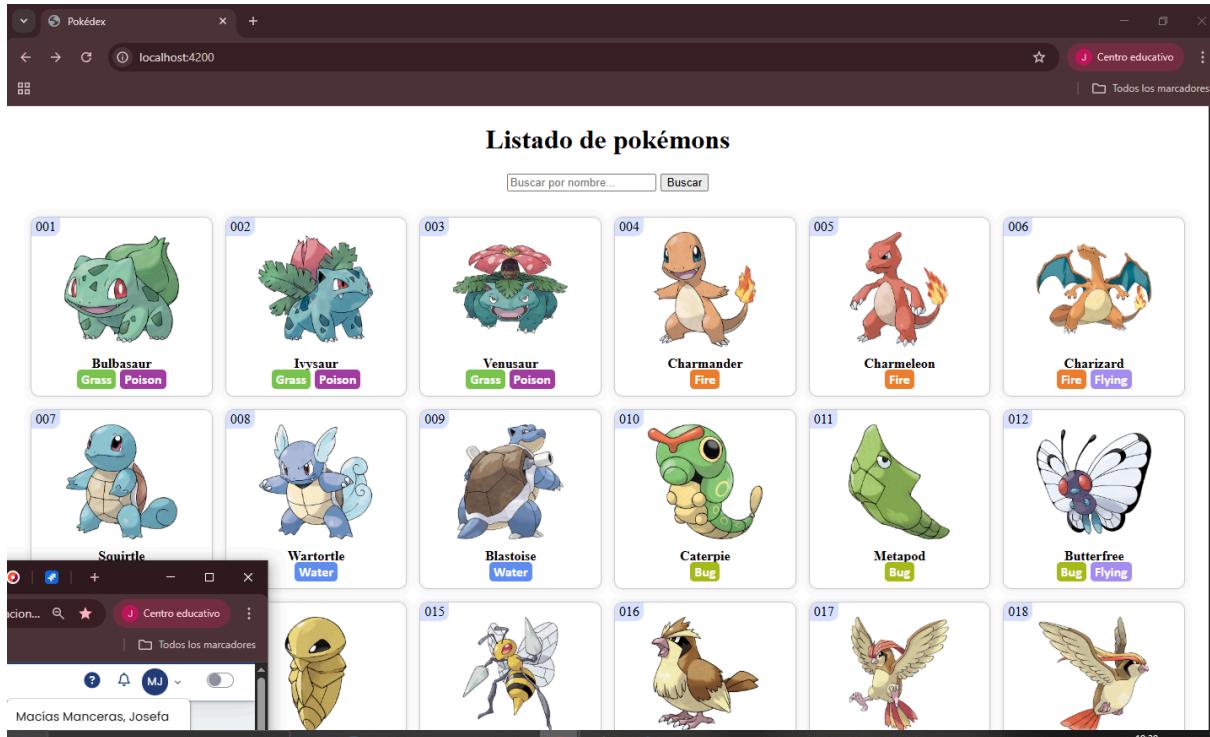


Resumen:

- Muestra los 151 pokémon con su imagen.
- Tiene un buscador por nombre.
- Al pinchar en el pokémon, se abre su ficha con más datos.
- Tiene un botón para darle a anterior/siguiente desde la ficha.

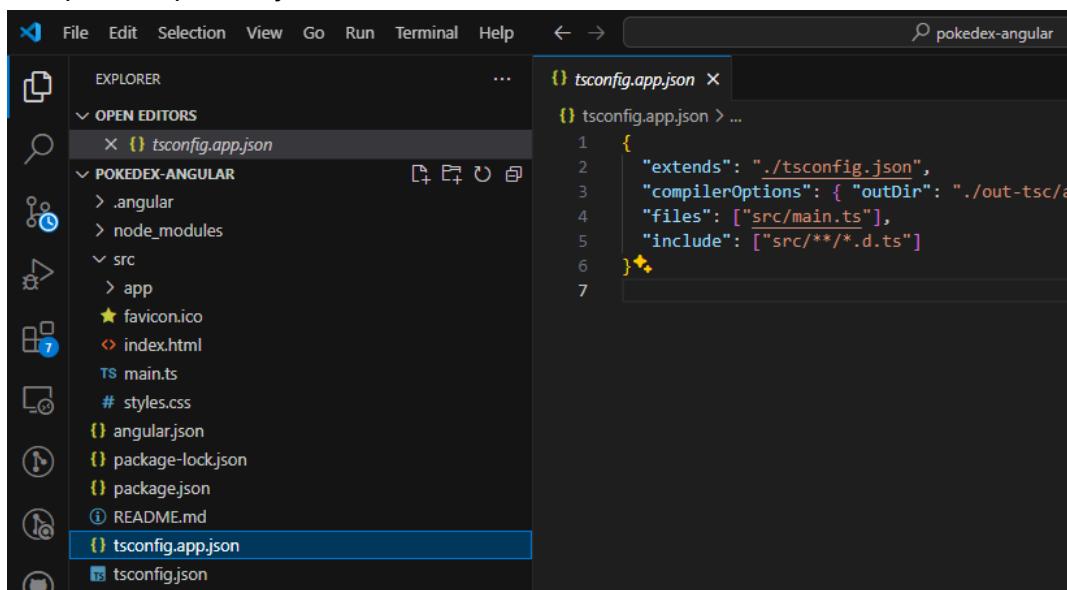


1. Creación del proyecto:

ng new pokedex --skip-tests:

Seleccione las opciones: Hojas de estilo CSS - No habilitar SSG/Prerendering.

Compruebo que se ejecute en Visual Studio Code.



2. Vista Listado de pokémon: Con el componente: ng g c pages/pokemon-list. Y se configura el enrutador y la salida:

**src/app/app.routes.ts:**

```
export const routes: Routes = [
  { path: '', component: PokemonListComponent }
];
```

```
ts app.routes.ts x
src > app > ts app.routes.ts > ...
1 import { Routes } from '@angular/router';
2 import { PokemonListComponent } from './pages/pokemon-list/pokemon-list.component';
3 import { PokemonComponent } from './pages/pokemon/pokemon.component';
4
5 export const routes: Routes = [
6   { path: '', component: PokemonListComponent },
7   { path: 'pokemon/:id', component: PokemonComponent },
8   { path: '**', redirectTo: '' }
9
10
```

**src/app/app.component.html:** <router-outlet></router-outlet>

Esto fuerza a que se renderice el nuevo componente.

```
ts app.component.ts x
src > app > ts app.component.ts > ...
1 import { Component } from '@angular/core';
2 import { RouterOutlet } from '@angular/router';
3
4 @Component({
5   selector: 'app-root',
6   standalone: true,
7   imports: [RouterOutlet],
8   templateUrl: './app.component.html',
9   styleUrls: ['./app.component.css']
10 })
11 export class AppComponent {}
```

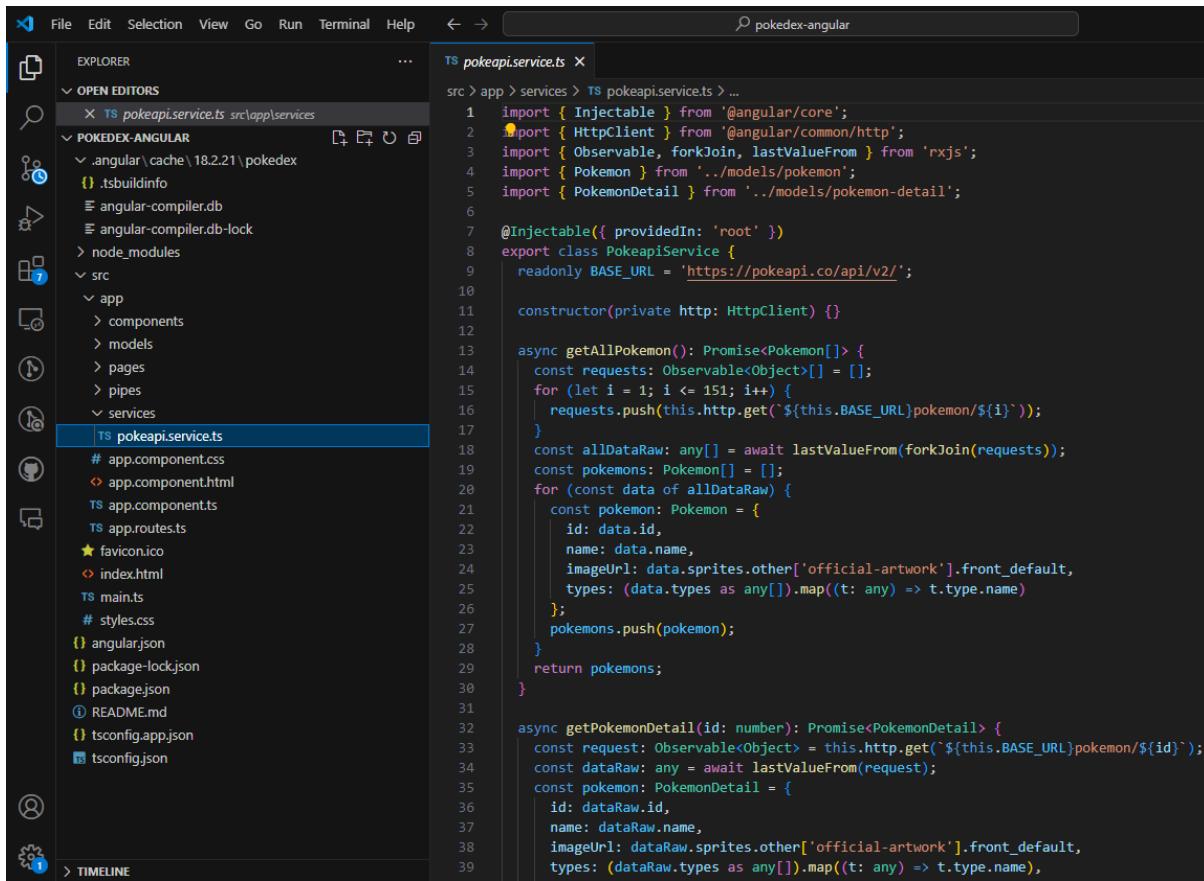
3. Modelos y servicio:

- **Modelo Pokemon:** ng g interface models/pokemon:

```
export interface Pokemon {
  id: number;
  name: string;
  imageUrl: string;
  types: string[];
}
```

- **Servicio PokeapiService:** ng g s services/pokeapi

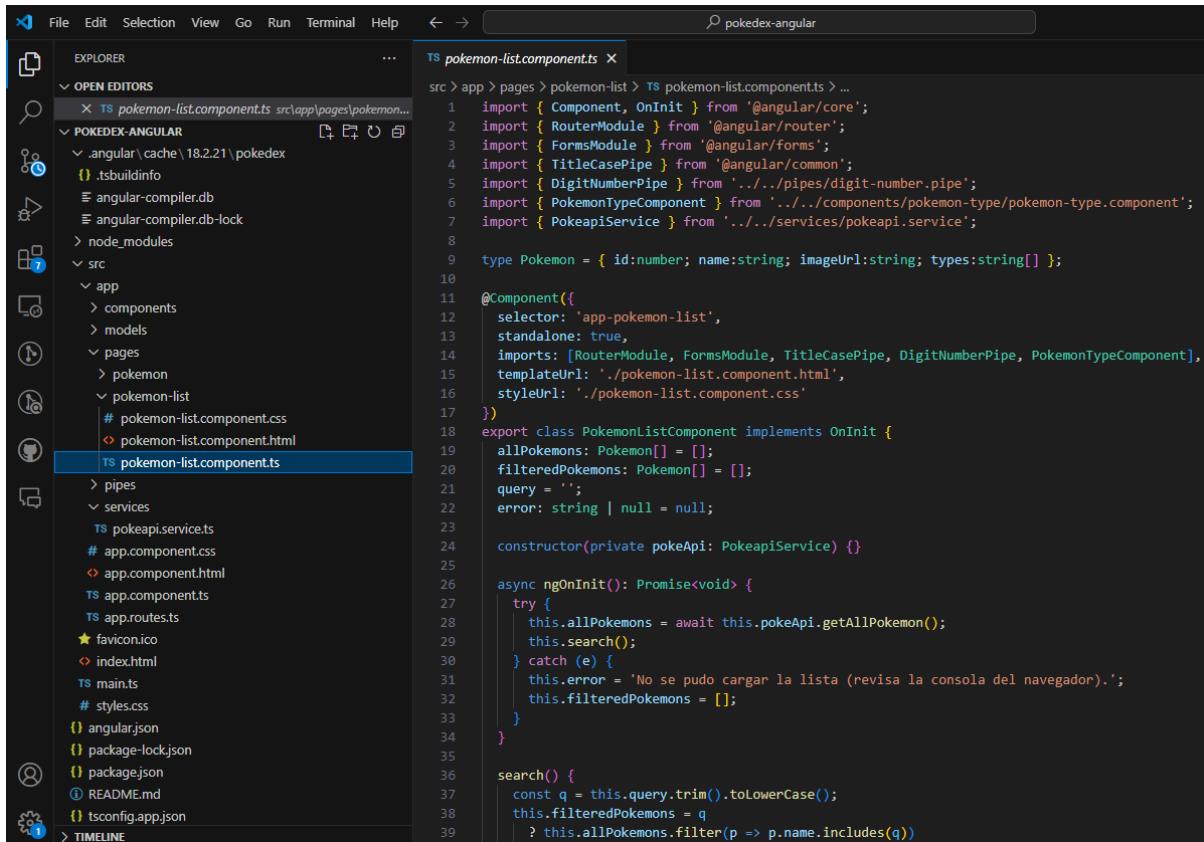
# Pokédex en Angular - Josefa Macias Manceras 2º DAM



```
src > app > services > TS pokeapi.service.ts > ...
1 import { Injectable } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3 import { Observable, forkJoin, lastValueFrom } from 'rxjs';
4 import { Pokemon } from '../models/pokemon';
5 import { PokemonDetail } from '../models/pokemon-detail';
6
7 @Injectable({ providedIn: 'root' })
8 export class PokeapiService {
9   readonly BASE_URL = 'https://pokeapi.co/api/v2/';
10
11   constructor(private http: HttpClient) {}
12
13   async getAllPokemon(): Promise<Pokemon[]> {
14     const requests: Observable<Object>[] = [];
15     for (let i = 1; i <= 151; i++) {
16       requests.push(this.http.get(`${this.BASE_URL}pokemon/${i}`));
17     }
18
19     const allDataRaw: any[] = await lastValueFrom(forkJoin(requests));
20     const pokemons: Pokemon[] = [];
21     for (const data of allDataRaw) {
22       const pokemon: Pokemon = {
23         id: data.id,
24         name: data.name,
25         imageUrl: data.sprites.other['official-artwork'].front_default,
26         types: (data.types as any[]).map(t: any) => t.type.name
27       };
28       pokemons.push(pokemon);
29     }
30
31     return pokemons;
32   }
33
34   async getPokemonDetail(id: number): Promise<PokemonDetail> {
35     const request: Observable<Object> = this.http.get(`${this.BASE_URL}pokemon/${id}`);
36     const dataRaw: any = await lastValueFrom(request);
37     const pokemon: PokemonDetail = {
38       id: dataRaw.id,
39       name: dataRaw.name,
        ...
```

## 4. Listado: lógica y vista:

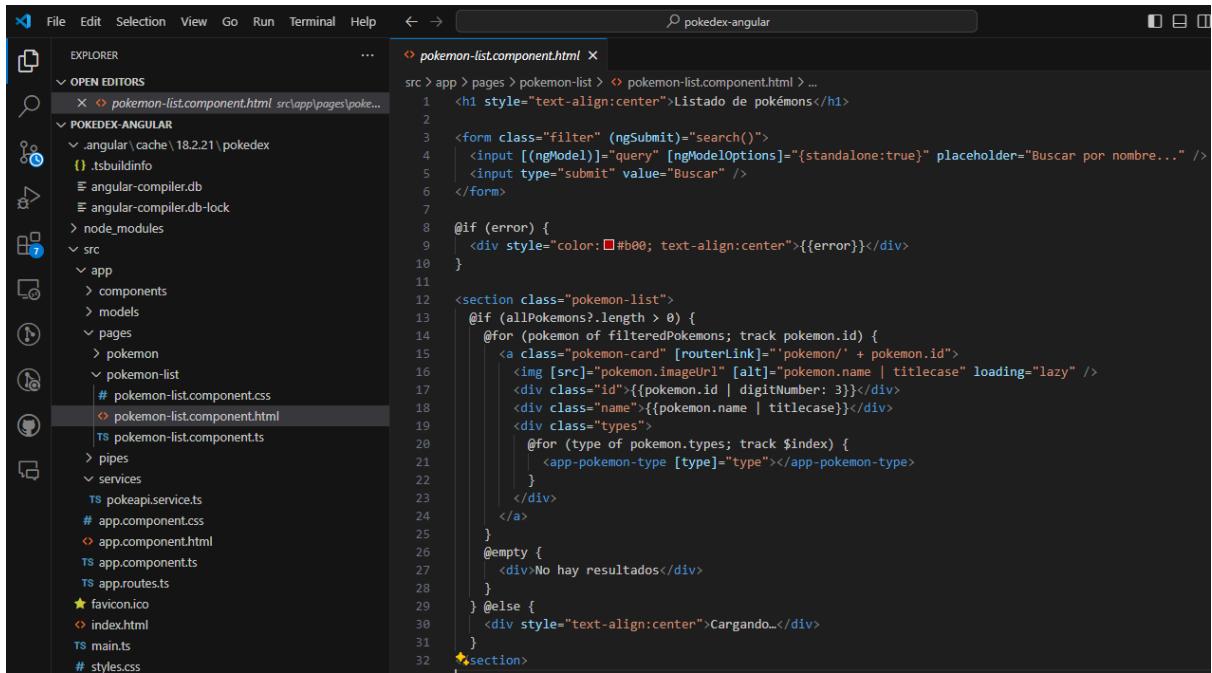
### 4.1 Lógica del componente:



```
src > app > pages > pokemon-list > TS pokemon-list.component.ts > ...
1 import { Component, OnInit } from '@angular/core';
2 import { RouterModule } from '@angular/router';
3 import { FormsModule } from '@angular/forms';
4 import { TitleCasePipe } from '@angular/common';
5 import { DigitNumberPipe } from './pipes/digit-number.pipe';
6 import { PokemonTypeComponent } from './components/pokemon-type/pokemon-type.component';
7 import { PokeapiService } from './services/pokeapi.service';
8
9 type Pokemon = { id:number; name:string; imageUrl:string; types:string[] };
10
11 @Component({
12   selector: 'app-pokemon-list',
13   standalone: true,
14   imports: [RouterModule, FormsModule, TitleCasePipe, DigitNumberPipe, PokemonTypeComponent],
15   templateUrl: './pokemon-list.component.html',
16   styleUrls: ['./pokemon-list.component.css'
17 })
18 export class PokemonListComponent implements OnInit {
19   allPokemons: Pokemon[] = [];
20   filteredPokemons: Pokemon[] = [];
21   query = '';
22   error: string | null = null;
23
24   constructor(private pokeApi: PokeapiService) {}
25
26   async ngOnInit(): Promise<void> {
27     try {
28       this.allPokemons = await this.pokeApi.getAllPokemon();
29       this.search();
30     } catch (e) {
31       this.error = 'No se pudo cargar la lista (revisa la consola del navegador).';
32       this.filteredPokemons = [];
33     }
34   }
35
36   search() {
37     const q = this.query.trim().toLowerCase();
38     this.filteredPokemons = q
39       ? this.allPokemons.filter(p => p.name.includes(q))
        ...
```

# Pokédex en Angular - Josefa Macias Manceras 2º DAM

## 4.2 HTML básico:

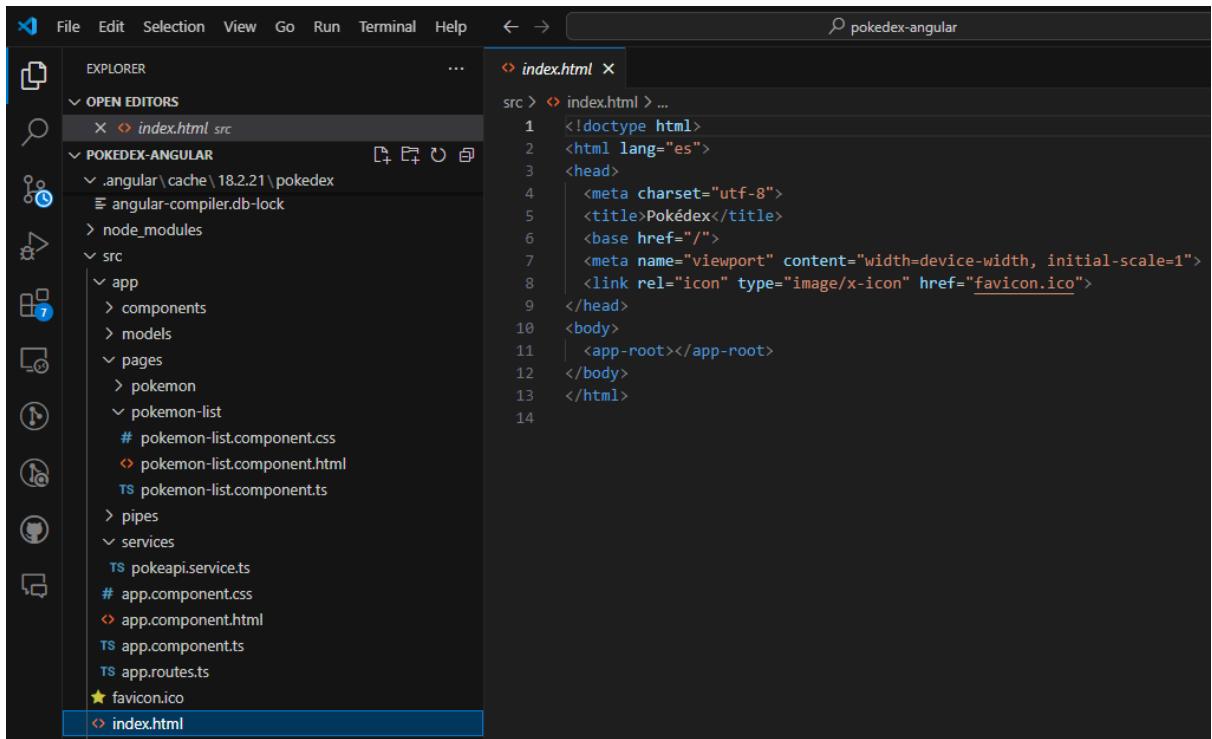


The screenshot shows the Visual Studio Code interface with the file `pokemon-list.component.html` open in the editor. The code implements a search functionality for a list of pokémons. It includes an input field for the search query, a submit button, and a section to display the results. If no results are found, it shows a message. If there's an error, it displays an error message. The code uses Angular's template syntax (`<ng-template>`, `<ngFor>`, `<ngIf>`) and services like `pokeapi.service.ts` to fetch data.

```
<h1 style="text-align:center">Listado de pokémons</h1>
<form class="filter" (ngSubmit)="search()">
  <input [(ngModel)]="query" [ngModelOptions]="{standalone:true}" placeholder="Buscar por nombre..." />
  <input type="submit" value="Buscar" />
</form>
<div style="color:#b00; text-align:center">{{error}}</div>
<section class="pokemon-list">
  <div class="name">{{pokemon.name | titlecase}}</div>
  <div class="types">
    <app-pokemon-type [type]="type"></app-pokemon-type>
  </div>
</div>
<div style="text-align:center">Cargando...</div>
</div>
```

## 5. Mejoras de presentación:

### 5.1 Título de la pestaña:



The screenshot shows the Visual Studio Code interface with the file `index.html` open in the editor. This is the main entry point of the Angular application. It defines the document structure with an `<app-root>` element where the application's components will be rendered.

```
<!doctype html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <title>Pokédex</title>
    <base href="/">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="icon" type="image/x-icon" href="favicon.ico">
  </head>
  <body>
    <app-root></app-root>
  </body>
</html>
```

### 5.2 Estilos del grid:

# Pokédex en Angular - Josefa Macias Manceras 2º DAM

```
# pokemon-list.component.css
src > app > pages > pokemon-list > # pokemon-list.component.css > h1
1  h1 { text-align: center; }
2  .filter {
3    display: flex;
4    gap: 5px;
5    width: 100%;
6    justify-content: center;
7    margin-bottom: 10px;
8  }
9  .pokemon-list {
10   display: grid;
11   grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
12   gap: 15px;
13   padding: 20px;
14 }
15 a { text-decoration: none; color: inherit; }
16 .pokemon-card {
17   --shadow-color: #rgba(0,0,0,0.1);
18   --border-radius: 10px;
19   position: relative;
20   border: 1px solid #ccc;
21   padding: 10px;
22   border-radius: var(--border-radius);
23   text-align: center;
24   box-shadow: 0 0px 8px var(--shadow-color);
25   cursor: pointer;
26 }
```

5.3 Nombre con mayúscula inicial: En el HTML:

```
<div class="name">{{ pokemon.name | titlecase }}</div>
```

5.4 Pipe para id con 3 dígitos: ng g pipe pipes/digit-number.

```
ts digit-number.pipe.ts
src > app > pipes > ts digit-number.pipe.ts > ...
1  import { Pipe, PipeTransform } from '@angular/core';
2  ...
3  @Pipe({
4    name: 'digitNumber',
5    standalone: true
6  })
7  export class DigitNumberPipe implements PipeTransform {
8    transform(value: number, digits: number): string {
9      return value.toString().padStart(digits, '0');
10   }
11 }
```

Uso en HTML: <div class="id">{{ pokemon.id | digitNumber:3 }}</div>

6. Componente de tipo (colores): ng g c components/pokemon-type:

## Pokédex en Angular - Josefa Macias Manceras 2º DAM

```
File Edit Selection View Go Run Terminal Help ← → pokedex-angular

EXPLORER ... TS pokemon-type.component.ts ×
OPEN EDITORS src\app\components\p...
POKEDEX-ANGULAR D+ E+ ⌂ ⌂
src app components\pokemon-type
  pokemon-type.component.html
TS pokemon-type.component.ts
  models
  pages
    pokémon
    pokémon-list
      # pokémon-list.component.css

src > app > components > pokemon-type > TS pokemon-type.component.ts > ...
1 import { Component, Input } from '@angular/core';
2 import { NgClass, TitleCasePipe } from '@angular/common';
3
4 @Component({
5   selector: 'app-pokemon-type',
6   standalone: true,
7   imports: [NgClass, TitleCasePipe],
8   templateUrl: './pokemon-type.component.html',
9   styleUrls: ['./pokemon-type.component.css'
10 })
11 export class PokemonTypeComponent {
12   @Input() type: string = '';
13 }
14
```

HTML del tipo: <span [ngClass]="type">{{ type | titlecase }}</span>

Uso en la lista:

```
@for (type of pokémon.types; track $index) {
  <app-pokemon-type [type]="type"></app-pokemon-type>
}
```

### 7. Vista “Detalle de pokémon”:

#### 7.1 Modelo y método del servicio:

ng g c pages/pokémon

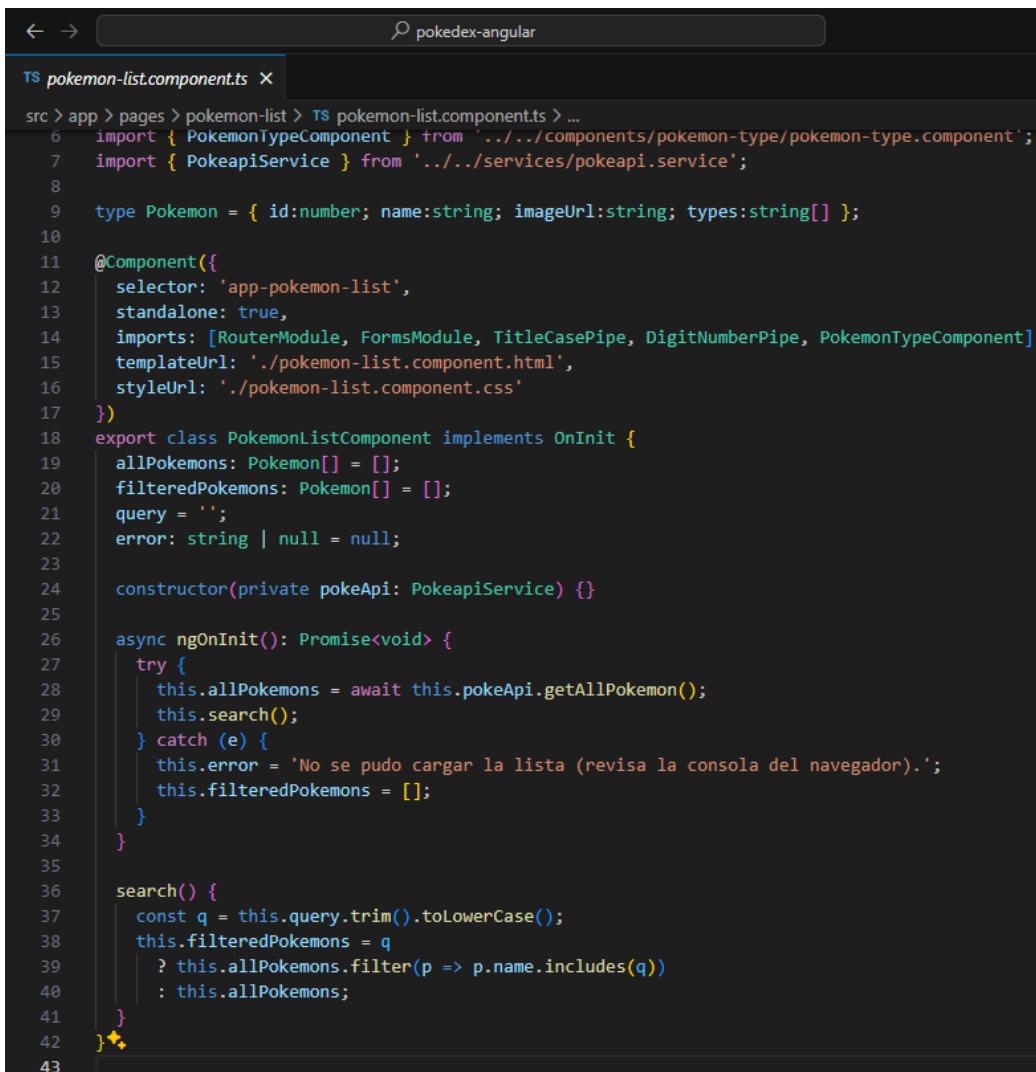
ng g interface models/pokemon-detail

```
File Edit Selection View Go Run Terminal Help ← → pokedex-angular

EXPLORER ... TS pokemon-detail.ts ×
OPEN EDITORS src\app\models\p...
POKEDEX-ANGULAR D+ E+ ⌂ ⌂
src app components\pokemon-type
  models
    pokemon-detail.ts
    # pokemon-detail.component.css

src > app > models > TS pokemon-detail.ts > ...
1 import { Pokemon } from './pokemon';
2
3 export interface PokemonDetail extends Pokemon {
4   height: number;
5   weight: number;
6 }
7
```

#### 7.2 HTML y lógica del detalle:

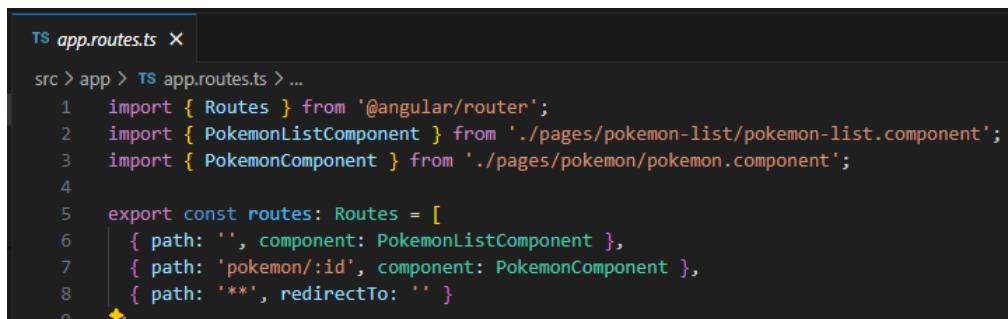


```

ts pokemon-list.component.ts ✘
src > app > pages > pokemon-list > ts pokemon-list.component.ts > ...
  6 import { PokemonTypeComponent } from '../../../../../components/pokemon-type/pokemon-type.component';
  7 import { PokeapiService } from '../../../../../services/pokeapi.service';
  8
  9 type Pokemon = { id:number; name:string; imageUrl:string; types:string[] };
 10
 11 @Component({
 12   selector: 'app-pokemon-list',
 13   standalone: true,
 14   imports: [RouterModule, FormsModule, TitleCasePipe, DigitNumberPipe, PokemonTypeComponent],
 15   templateUrl: './pokemon-list.component.html',
 16   styleUrls: ['./pokemon-list.component.css'
 17 })
 18 export class PokemonListComponent implements OnInit {
 19   allPokemons: Pokemon[] = [];
 20   filteredPokemons: Pokemon[] = [];
 21   query = '';
 22   error: string | null = null;
 23
 24   constructor(private pokeApi: PokeapiService) {}
 25
 26   async ngOnInit(): Promise<void> {
 27     try {
 28       this.allPokemons = await this.pokeApi.getAllPokemon();
 29       this.search();
 30     } catch (e) {
 31       this.error = 'No se pudo cargar la lista (revisa la consola del navegador).';
 32       this.filteredPokemons = [];
 33     }
 34   }
 35
 36   search() {
 37     const q = this.query.trim().toLowerCase();
 38     this.filteredPokemons =
 39       ? this.allPokemons.filter(p => p.name.includes(q))
 40       : this.allPokemons;
 41   }
 42 }
 43

```

#### 8. Rutas del detalle y enlace desde la lista:



```

ts app.routes.ts ✘
src > app > ts app.routes.ts > ...
  1 import { Routes } from '@angular/router';
  2 import { PokemonListComponent } from './pages/pokemon-list/pokemon-list.component';
  3 import { PokemonComponent } from './pages/pokemon/pokemon.component';
  4
  5 export const routes: Routes = [
  6   { path: '', component: PokemonListComponent },
  7   { path: 'pokemon/:id', component: PokemonComponent },
  8   { path: '**', redirectTo: '' }
  9

```

[Anterior](#) #151 Mew [Siguiente](#)



Psychic

Altura: 0.4 m

Peso: 4 Kg

#### 9. Buscador por nombre:

Imports: usar FormsModule en el componente de lista.

```
<form class="filter" (ngSubmit)="search()">
  <input [(ngModel)]="query" [ngModelOptions]="{standalone:true}" placeholder="Buscar por
nombre..." />
  <input type="submit" value="Buscar" />
</form>
```



#### Listado de pokémons



TS (lista):

# Pokédex en Angular - Josefa Macias Manceras 2º DAM

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "POKEDEX-ANGULAR".
  - src:** Contains "app" and "models".
  - app:** Contains "pages" and "models".
  - pages:** Contains "pokemon" and "pokemon-list".
    - "pokemon" contains "pokemon.component.css" and "pokemon.component.html".
    - "pokemon-list" contains "pokemon-list.component.css" and "pokemon-list.component.html".- Editor:** Displays the file `pokemon-list.component.ts`. The code is as follows:

```
src > app > pages > pokemon-list > pokemon-list.component.html > ...
1  <h1 style="text-align:center">Listado de pokémons</h1>
2
3  <form class="filter" (ngSubmit)="search()">
4    <input [(ngModel)]="query" [ngModelOptions]="{standalone:true}" placeholder="Buscar" type="text"/>
5    <input type="submit" value="Buscar" />
6  </form>
7
8  @if (error) {
9    <div style="color:#b00; text-align:center">{{error}}</div>
10 }
11
12 <section class="pokemon-list">
13   @if (allPokemons?.length > 0) {
14     @for (pokemon of filteredPokemons; track pokemon.id) {
15       <a class="pokemon-card" [routerLink]="'pokemon/' + pokemon.id">
16         <img [src]="pokemon.imageUrl" [alt]="pokemon.name | titlecase" />
17         <div class="id">{{pokemon.id | digitNumber: 3}}</div>
18     }
19   }
20 }
```

## 10. Botones Anterior/Siguiente y actualización al cambiar la URL

Métodos goPrevious/goNext

The screenshot shows the VS Code interface with the following details:

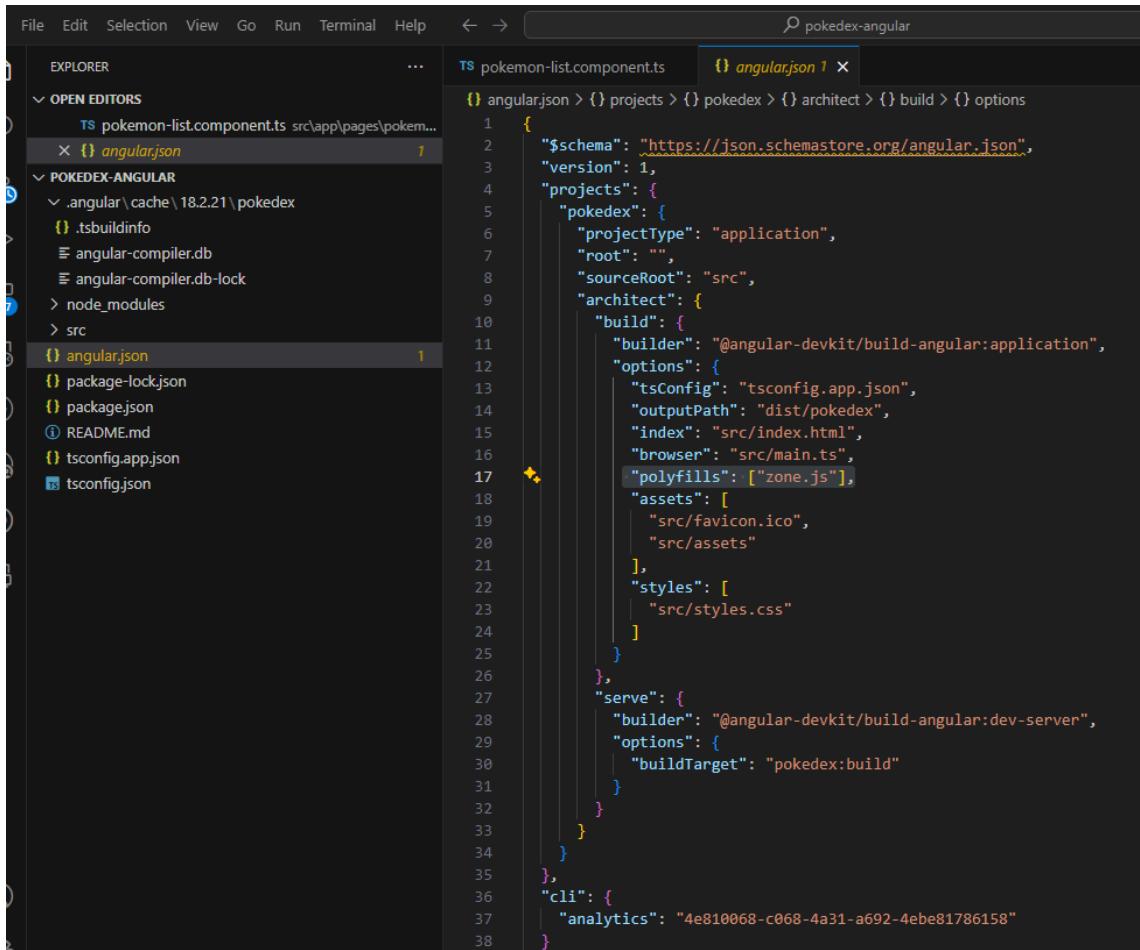
- File Explorer:** Shows the project structure under "POKEDEX-ANGULAR".
- Editor:** Displays the file `pokemon.component.ts`. The code is as follows:

```
src > app > pages > pokemon > pokemon.component.html > ...
1  @if (pokemon) {
2    <h1>
3      <button (click)="goPrevious()">Anterior</button>
4      #{{pokemon.id | digitNumber: 3}} {{pokemon.name | titlecase}}
5      <button (click)="goNext()">Siguiente</button>
6    </h1>
7
8    <section>
9      <img [src]="pokemon.imageUrl" alt="{{pokemon.name}}" />
10     <div class="info">
11       <div class="types">
12         @for (type of pokemon.types; track $index) {
13           <app-pokemon-type [type]="type"></app-pokemon-type>
14         }
15       </div>
16       <div>Altura: {{pokemon.height / 10}} m</div>
17       <div>Peso: {{pokemon.weight / 10}} Kg</div>
18     </div>
19   </section>
20 }
```



## 11. Ajustes extra:

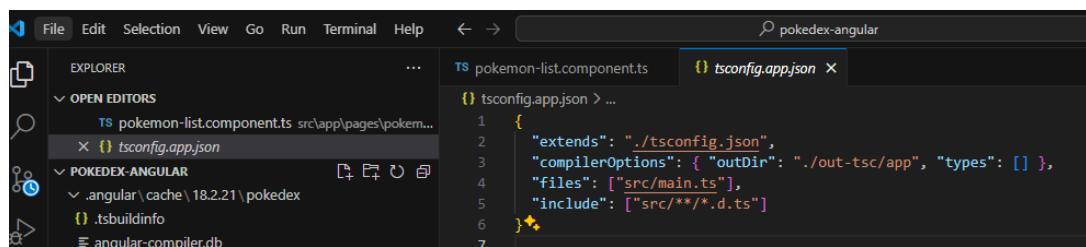
Me estaba dando problemas al cargar la página, así que creé tsconfig.app.json y añadí "tsConfig": "tsconfig.app.json" en angular.json y añadí zone.js



```

File Edit Selection View Go Run Terminal Help ⏪ ⏴ pokedex-angular
EXPLORER OPEN EDITORS
  TS pokemon-list.component.ts src\app\pages\pokem...
  X {} angular.json 1
  POKEDEX-ANGULAR
    .angular\cache\18.2.21\pokedex
      .tsbuildinfo
      angular-compiler.db
      angular-compiler.db-lock
    node_modules
    src
    {} angular.json 1
    package-lock.json
    package.json
    README.md
    tsconfig.app.json
    tsconfig.json
TS pokemon-list.component.ts angular.json 1
{
  "$schema": "https://json.schemastore.org/angular.json",
  "version": 1,
  "projects": {
    "pokedex": {
      "projectType": "application",
      "root": "",
      "sourceRoot": "src",
      "architect": {
        "build": {
          "builder": "@angular-devkit/build-angular:application",
          "options": {
            "tsConfig": "tsconfig.app.json",
            "outputPath": "dist/pokedex",
            "index": "src/index.html",
            "browser": "src/main.ts",
            "polyfills": ["zone.js"],
            "assets": [
              "src/favicon.ico",
              "src/assets"
            ],
            "styles": [
              "src/styles.css"
            ]
          },
          "serve": {
            "builder": "@angular-devkit/build-angular:dev-server",
            "options": {
              "buildTarget": "pokedex:build"
            }
          }
        },
        "cli": {
          "analytics": "4e810068-c068-4a31-a692-4ebe81786158"
        }
      }
    }
  }
}

```



```

File Edit Selection View Go Run Terminal Help ⏪ ⏴ pokedex-angular
EXPLORER OPEN EDITORS
  TS pokemon-list.component.ts src\app\pages\pokem...
  X {} tsconfig.app.json ...
  POKEDEX-ANGULAR
    .angular\cache\18.2.21\pokedex
      .tsbuildinfo
      angular-compiler.db
TS tsconfig.app.json ...
{
  "extends": "./tsconfig.json",
  "compilerOptions": {
    "outDir": "./out-tsc/app",
    "types": []
  },
  "files": [
    "src/main.ts"
  ],
  "include": [
    "src/**/*.d.ts"
  ]
}

```