239 inicio de proyecto

Debemos de ejecutar

Ionic start fotosgram tabs

Seguimos las reglas

Recordar que debemos de tener instalado angular ionic

Y otras depencias que se encuentran en la lista

[**Instalaciones recomendadas - Curso de ionic**](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057#instalaciones-recomendadas---curso-de-ionic)

[**Instalaciones Necesarias**](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057#instalaciones-necesarias)

* [Google Chrome](https://www.google.com/chrome/)
* [Visual Studio Code](https://code.visualstudio.com/)
* [Postman](https://www.postman.com/downloads/)
* [Mongo Compass](https://www.mongodb.com/try/download/compass)
* [Git](https://git-scm.com/)

git config --global user.name "Tu nombre"

git config --global user.email "Tu correo"

* [Node](https://nodejs.org/es/)
* [Android Studio](https://developer.android.com/studio/?gclid=Cj0KCQjwhb36BRCfARIsAKcXh6GRXJN_hJrabNpOE94384hWx1uh4qPgqVQBiZJMkDEcNUgTQf3UwZoaAr-ZEALw_wcB&gclsrc=aw.ds)
* XCode - Sólo para usuarios de OSX
* [AngularCLI](https://cli.angular.io/)
* [ionic framework](https://ionicframework.com/)

[**Extensiones de VSCode**](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057#extensiones-de-vscode)

* [Activitus Bar](https://marketplace.visualstudio.com/items?itemName=Gruntfuggly.activitusbar)
* [ionic Snippets](https://marketplace.visualstudio.com/items?itemName=fivethree.vscode-ionic-snippets)
* [TypeScript Importer](https://marketplace.visualstudio.com/items?itemName=pmneo.tsimporter)

[**Configuración del Bracket Pair Colorizer 2**](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057#configuraci%C3%B3n-del-bracket-pair-colorizer-2)

[Bracket Pair Colorizer 2](https://marketplace.visualstudio.com/items?itemName=CoenraadS.bracket-pair-colorizer-2)

"bracket-pair-colorizer-2.colors": [

"#fafafa",

"#9F51B6",

"#F7C244",

"#F07850",

"#9CDD29",

"#C497D4"

],

[**Tema que estoy usando en VSCode:**](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057#tema-que-estoy-usando-en-vscode)

* [Monokai Night](https://marketplace.visualstudio.com/items?itemName=fabiospampinato.vscode-monokai-night)
* [Iconos](https://marketplace.visualstudio.com/items?itemName=PKief.material-icon-theme)

[**ValeProgramacion**](https://gist.github.com/ValeProgramacion)**commented [on Mar 24, 2020](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057?permalink_comment_id=3225608" \l "gistcomment-3225608)**

Profesor, en el curso de ionic de udemy la lista aparentemente es mas grande...puede ser?

Principio del formulario

Final del formulario

[**mm0ra**](https://gist.github.com/mm0ra)**commented [on Apr 26, 2020](https://gist.github.com/Klerith/1a2801452e4abbb30ff29a1da0cbc057?permalink_comment_id=3270637" \l "gistcomment-3270637)**

Cuales son las extensiones requeridas de visual studio code que menciona en el curso de UDEMY?

Principio del formulario

Final del formulario

Cuales son las extensiones requeridas de visual studio code que menciona en el curso de UDEMY?

AB Html Formatter

Activitus Bar

Angular Snippets

Angular Language Service

ionic Snippets

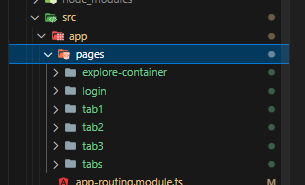
Paste JSON as Code

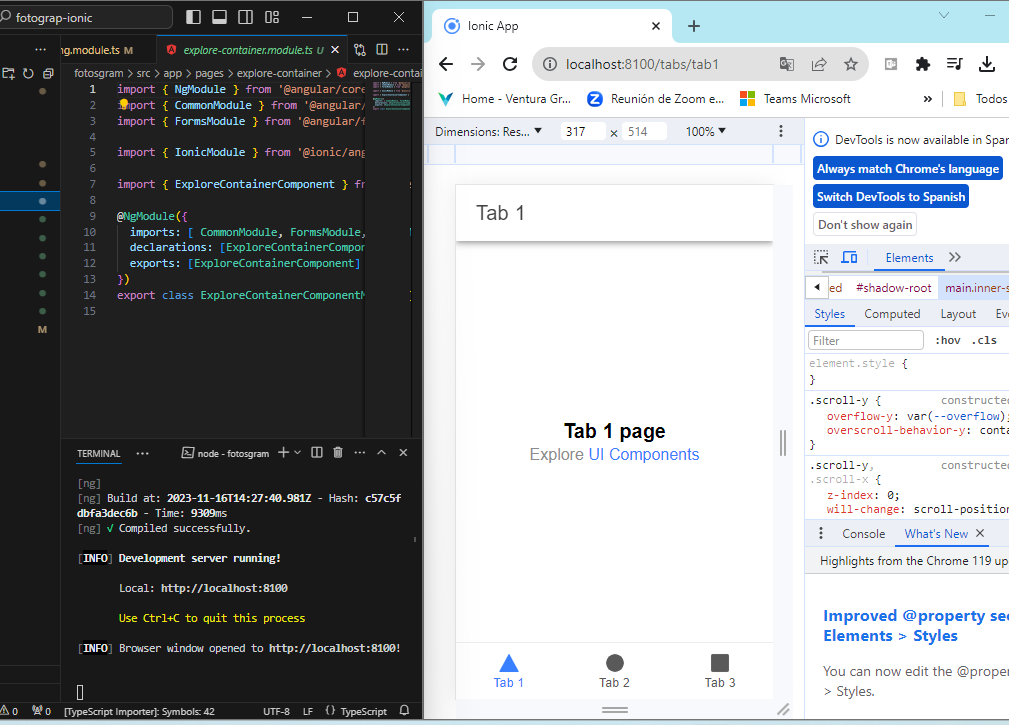
TypeScript Importer

Listo empezamos creando una carpeta page y una pagina login

Ionic g page pages/login

Luego agregamos tadas paginas que creo ionic por defecto y la pegamos en esta carpeta



Luego lanzamos ionic serve para visualizar la app

240 Obtener el listado de nuestros Posts

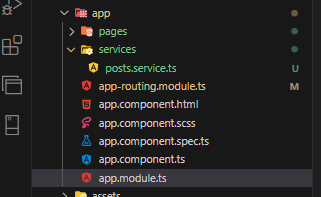
Primero generamos un servicio

Ionic g Service services/posts

Los archivos de prueba que se crean se deben borrar si no se van a usar

Posts.service.spec.ts

En este app.module.ts general se importan los componentes que se requieran



import { HttpClientModule } from '@angular/common/http';

y se agrega en el import ese modulo para usarlo

@NgModule({

  declarations: [AppComponent],

  imports: [BrowserModule, IonicModule.forRoot(), AppRoutingModule, HttpClientModule],

  providers: [{ provide: RouteReuseStrategy, useClass: IonicRouteStrategy }],

  bootstrap: [AppComponent],

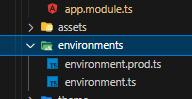
})

export class AppModule {}

después de haber importado ahora podemos importar y utilizar estos components http cliente

en el servicio de post para aquellas peticiones

nota en los env agrego un valor para luego ser utilizado en alguna clase necesaria



export const environment = {

  production: true,

  url: 'http://localhost:3000'

};

En ambos

Luego en servicio lo llamo y agrego las rutas



import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { environment } from 'src/environments/environment';

const URL = environment.url;

@Injectable({

  providedIn: 'root'

})

export class PostsService {

//para el paginado

  paginaPosts = 0;

  constructor( private http: HttpClient) {}

  // metodo donde empiezo a utilizar la ruta

    getPosts(){

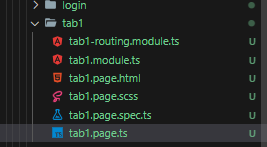
      this.paginaPosts ++;

      return this.http.get(`${ URL}/posts/?pagina=${ this.paginaPosts}`);

    }

}

Y este servicio lo utilizo en la pagina que quiera



import { Component, OnInit } from '@angular/core';

import { PostsService } from '../../services/posts.service';

@Component({

  selector: 'app-tab1',

  templateUrl: 'tab1.page.html',

  styleUrls: ['tab1.page.scss']

})

// implementar el OnInit

export class Tab1Page implements OnInit {

  constructor(private PostsService: PostsService) {}

  ngOnInit() {

this.PostsService.getPosts()

.subscribe( resp =>{

  console.log(resp);

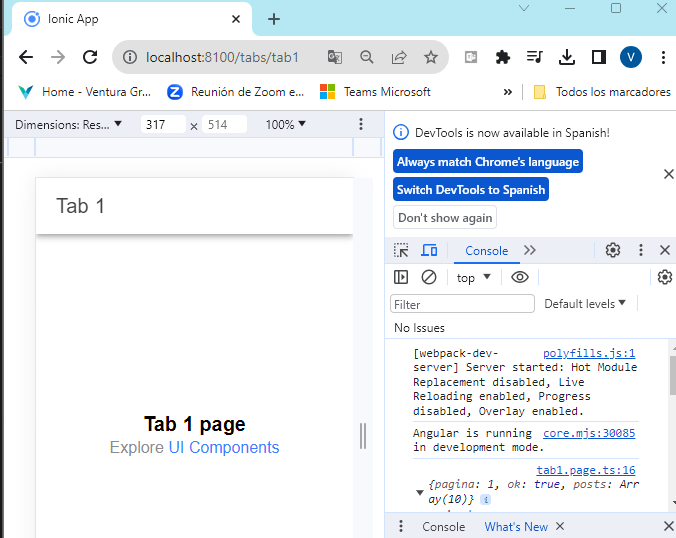
});

  }

}

Listo carga los datos

Si no funciona el cors instalar extencion cors para brobar



241 Backend – Cors resolver el problema de peticiones CROSS Domain

En la api index.ts agregamos las sentencias del cors, ya están en el archivo del backend

243 Interfaces necesarias para el tipado de nuestros datos

Crear una carpeta de interfaces y dentro de ella el archivo interfaces.ts

Este es la estructura del usuario

export interface RespuestaPosts {

ok: boolean;

pagina: number;

posts: Post[];

}

 //? opcionales los datos

export interface Post {

imgs?: string[];

\_id?: string;

mensaje?: string;

coords?: string;

usuario?: Usuario;

created?: string;

}

export interface Usuario {

avatar?: string;

\_id?: string;

nombre?: string;

email?: string;

}

Esta interfaces la pasamos en el return del servicio

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { environment } from 'src/environments/environment';

import { RespuestaPosts } from '../interfaces/interfaces';

const URL = environment.url;

@Injectable({

  providedIn: 'root'

})

export class PostsService {

//para el paginado

  paginaPosts = 0;

  constructor( private http: HttpClient) {}

  // metodo donde empiezo a utilizar la ruta

    getPosts(){

      this.paginaPosts ++;

      return this.http.get<RespuestaPosts>(`${ URL}/posts/?pagina=${ this.paginaPosts}`);

    }

}

244 Componente Post, Posts y Components Moudule

Le quite los nombres y cambie los iconos

<ion-tabs>

  <ion-tab-bar slot="bottom">

    <ion-tab-button tab="tab1" href="/tabs/tab1">

      <ion-icon aria-hidden="true" name="home"></ion-icon>

    </ion-tab-button>

    <ion-tab-button tab="tab2" href="/tabs/tab2">

      <ion-icon aria-hidden="true" name="add-circle-outline"></ion-icon>

    </ion-tab-button>

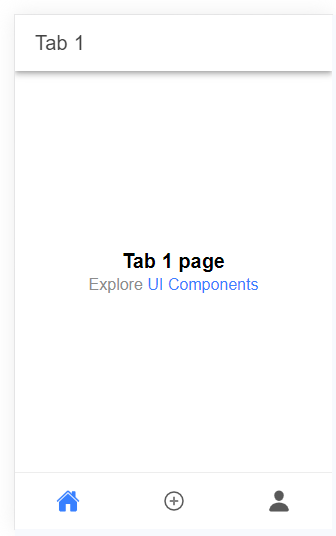
    <ion-tab-button tab="tab3" href="/tabs/tab3">

      <ion-icon aria-hidden="true" name="person"></ion-icon>

    </ion-tab-button>

  </ion-tab-bar>

</ion-tabs>.



Paso al tab1.page.ts

import { Component, OnInit } from '@angular/core';

import { PostsService } from '../../services/posts.service';

import { Post } from 'src/app/interfaces/interfaces';

@Component({

  selector: 'app-tab1',

  templateUrl: 'tab1.page.html',

  styleUrls: ['tab1.page.scss']

})

// implementar el OnInit

export class Tab1Page implements OnInit {

  // arreglo de post

  posts: Post[] = [];

  constructor(private PostsService: PostsService) {}

  ngOnInit() {

this.PostsService.getPosts()

.subscribe( resp =>{

  console.log(resp);

  //cargo los post nuevos

  this.posts.push( ...resp.posts);

});

  }

}

Modificar tap1.html

<ion-header [translucent]="true">

  <ion-toolbar>

    <ion-title>

      Home

    </ion-title>

  </ion-toolbar>

</ion-header>

<ion-content [fullscreen]="true">

 <!-- Referncia -->

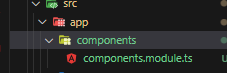
 <!-- Posts -->

 <!-- infinte Scroll -->

</ion-content>

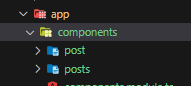
Crear componentes.module.ts

ionic g m components



ionic g c components/posts

ionic g c components/post



Modificar el components.module.ts para exportar estos dos componentes

import { NgModule } from '@angular/core';

import { CommonModule } from '@angular/common';

import { PostsComponent } from './posts/posts.component';

import { PostComponent } from './post/post.component';

import { IonicModule } from '@ionic/angular';

@NgModule({

  //los declaramos

  declarations: [

    PostsComponent,

    PostComponent

  ],

 // exportamos solo los posts

  exports: [

    PostsComponent

  ],

//importamos ionic module

  imports: [

    CommonModule,

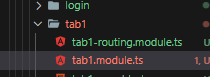
    IonicModule

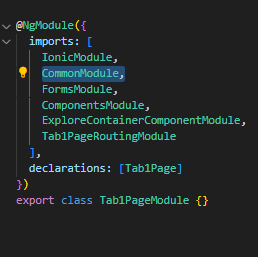
  ]

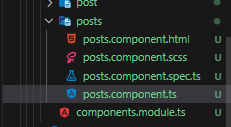
})

export class ComponentsModule { }

importar estos componentes en la modulos de las paginas que se van a utilizar







import { Component, Input, OnInit } from '@angular/core';

import { Post } from 'src/app/interfaces/interfaces';

@Component({

  selector: 'app-posts',

  templateUrl: './posts.component.html',

  styleUrls: ['./posts.component.scss'],

})

export class PostsComponent  implements OnInit {

//array de input posts

  @Input() posts: Post[]= [];

  constructor() { }

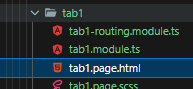
  ngOnInit() {

    console.log(this.posts);

  }

}

Ya podemos usar este componente en el



<ion-header [translucent]="true">

  <ion-toolbar>

    <ion-title>

      Home

    </ion-title>

  </ion-toolbar>

</ion-header>

<ion-content [fullscreen]="true">

 <!-- Referncia -->

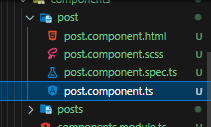
 <!-- Posts -->

 <app-posts [posts]="posts"></app-posts>

 <!-- infinte Scroll -->

</ion-content>

En el post debemos de cargar la información que pasara por los posts



import { Component, Input, OnInit } from '@angular/core';

import { Post } from 'src/app/interfaces/interfaces';

@Component({

  selector: 'app-post',

  templateUrl: './post.component.html',

  styleUrls: ['./post.component.scss'],

})

export class PostComponent  implements OnInit {

//input de post

  @Input() post: Post = {};

  constructor() { }

  ngOnInit() {}

}

En su 

Le mandamos el mensaje

<p>

  post: {{ post.mensaje}}

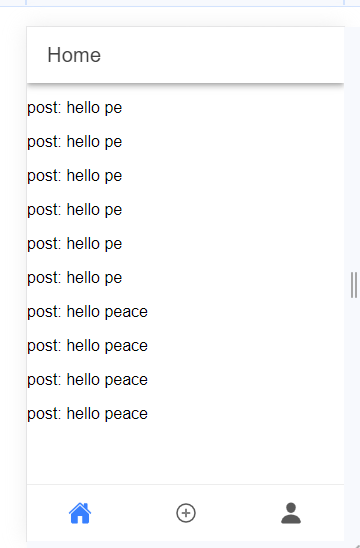
</p>

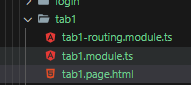
Y ahora en el posts html cargo todo los post



<app-post \*ngFor="let post of posts"

[post]="post">v</app-post>



Personalizar un poco el- tab1html 

<ion-header [translucent]="true">

  <ion-toolbar>

    <ion-buttons slot="start">

      <ion-icon slot="icon-only" name="bonfire"></ion-icon>

    </ion-buttons>

    <ion-title>

      FotosGram

    </ion-title>

  </ion-toolbar>

</ion-header>

<ion-content [fullscreen]="true">

 <!-- Referncia -->

 <!-- Posts -->

 <app-posts [posts]="posts"></app-posts>

 <!-- infinte Scroll -->

</ion-content>



245 postscomponet diseño multicolumna



<ion-grid>

  <ion-row>

    <ion-col size="12" size-lg="3" size-md="4" size-sm="6" size-xs="12"

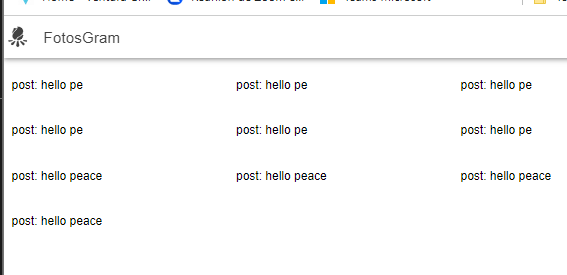
    \*ngFor="let post of posts">

    <app-post [post]="post"></app-post>

  </ion-col>

  </ion-row>

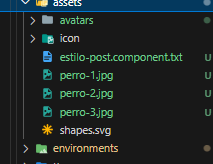
</ion-grid>



246 PostComponent HTML de un post ya coge forma



Por el momento utilizaremos unas imágenes y avatars por defecto gaurdadas en el acces



Trae un scss en la carpeta que toca pegarlo en el scss del post para aplicar la clase post



.post {

    margin-top: 10px;

    -webkit-box-shadow: 0 2px 10px -5px rgba(0,0,0,0.5);

    box-shadow: 0 2px 10px -5px rgba(0,0,0,0.5);

}

.image-slide {

    width: 100%;

    height: 250px;

    // background-color: red;

    /\* Full height \*/

    // height: 100%;

    /\* Imagen centrada y colocada de forma elegante \*/

    background-position: center;

    background-repeat: no-repeat;

    background-size: cover;

}

En el html al utilizar el slides sale un error por versión como estoy usando la 7 debo de actualizar o instalar esto

1 el swiper con "npm instal swiper@latest"

2 components.module.ts

import { NgModule, CUSTOM\_ELEMENTS\_SCHEMA } from '@angular/core';

import { CommonModule } from '@angular/common';

import { PostsComponent } from './posts/posts.component';

import { PostComponent } from './post/post.component';

import { IonicModule } from '@ionic/angular';

@NgModule({

  //los declaramos

  declarations: [

    PostsComponent,

    PostComponent

  ],

  // para el uso de los swiper slide en version 7 ionic

  schemas: [CUSTOM\_ELEMENTS\_SCHEMA],

 // exportamos solo los posts

  exports: [

    PostsComponent,

  ],

//importamos ionic module

  imports: [

    CommonModule,

    IonicModule,

  ]

})

export class ComponentsModule { }

3 post.component.ts agrego el swiper

import { Component, Input, OnInit } from '@angular/core';

import { Post } from 'src/app/interfaces/interfaces';

import {register} from 'swiper/element/bundle'

//para utilizar el slides

register();

@Component({

  selector: 'app-post',

  templateUrl: './post.component.html',

  styleUrls: ['./post.component.scss'],

})

export class PostComponent  implements OnInit {

//input de post

  @Input() post: Post = {};

  constructor() { }

  ngOnInit() {}

}

4 post.component.html

<div class="post">

<ion-item >

  <ion-avatar slot="start">

<img src="/assets/avatars/av-1.png" alt="">

  </ion-avatar>

  <ion-label>

    <h3>Cristian</h3>

    <h5>Stiy92</h5>

  </ion-label>

</ion-item>

<swiper-container>

  <swiper-slide>

   <img src="/assets/perro-1.jpg">

  </swiper-slide>

  <swiper-slide>

    <img src="/assets/perro-2.jpg">

   </swiper-slide>

   <swiper-slide>

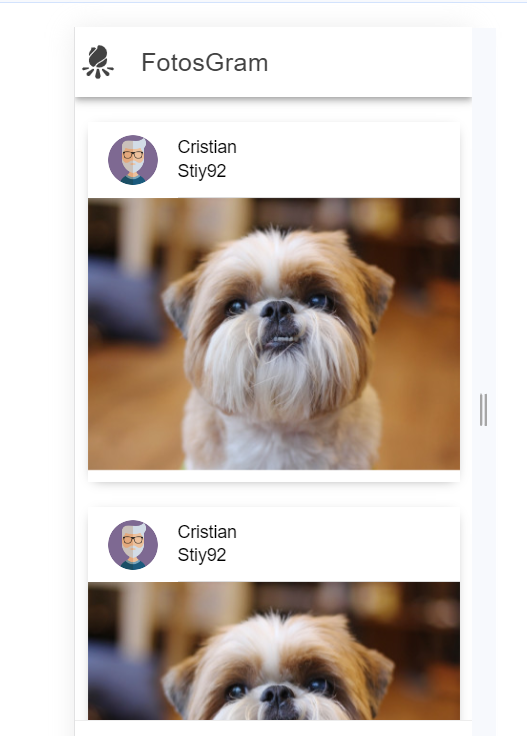
    <img src="/assets/perro-3.jpg">

   </swiper-slide>

  </swiper-container>

</div>

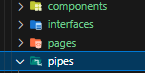
Listo



247 DomSanitizer Pipe – colocar imágenes en el background de forma segura

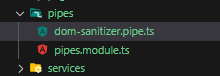
1 generar moudule pipes

Ionic g m pipes



Crear pipes / domsanatizer

Ionic g pipe pipes/domSanitizer



import { Pipe, PipeTransform } from '@angular/core';

import { DomSanitizer } from '@angular/platform-browser';

@Pipe({

  name: 'domSanitizer'

})

export class DomSanitizerPipe implements PipeTransform {

  constructor( private domSanitizer: DomSanitizer){}

  transform(img: string): any {

    const domImg = `background-image: url('${img}')`;

//ageurar el estilo de las imagenes cargadas

    return this.domSanitizer.bypassSecurityTrustStyle( domImg);

  }

}

En el pipe module lo declaramos e exportamos

import { NgModule } from '@angular/core';

import { CommonModule } from '@angular/common';

import { DomSanitizerPipe } from './dom-sanitizer.pipe';

@NgModule({

  declarations: [

    DomSanitizerPipe

  ],

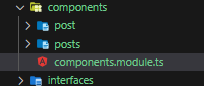
  exports: [

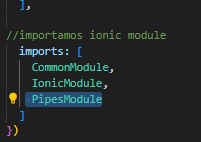
    DomSanitizerPipe

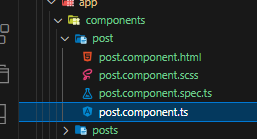
  ]

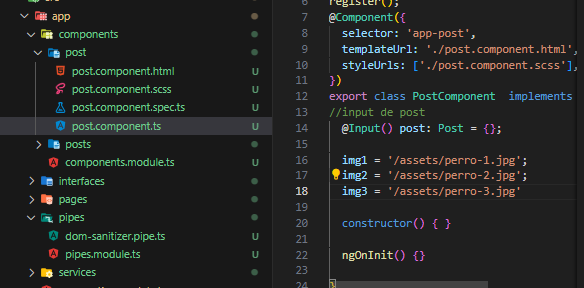
})

export class PipesModule { }

lo importo en el 



Ejemplo poniendolas en el 



El el html los agrego en el swiper slide



<div class="post">

<ion-item >

  <ion-avatar slot="start">

<img src="/assets/avatars/av-1.png" alt="">

  </ion-avatar>

  <ion-label>

    <h3>Cristian</h3>

    <h5>Stiy92</h5>

  </ion-label>

</ion-item>

<swiper-container>

  <swiper-slide class="image-slide" [style]=" img1 | domSanitizer">

  </swiper-slide>

  <swiper-slide class="image-slide" [style]=" img2 | domSanitizer">

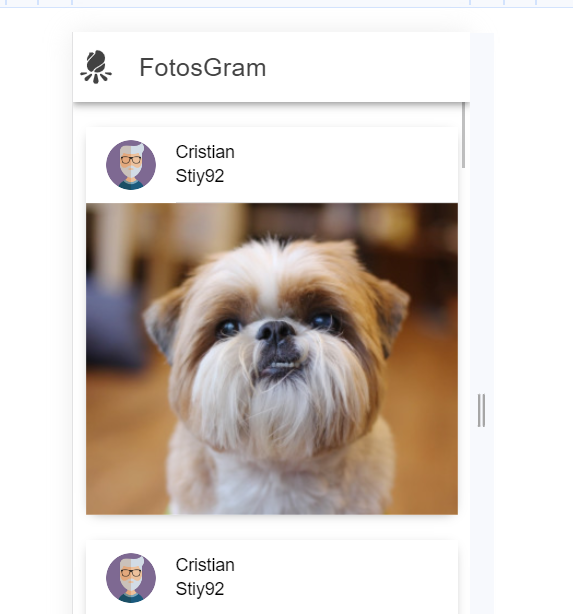
  </swiper-slide>

    <swiper-slide class="image-slide" [style]=" img3 | domSanitizer">

    </swiper-slide>

  </swiper-container>

</div>



248 infinite-scroll De nuestros POSTs