

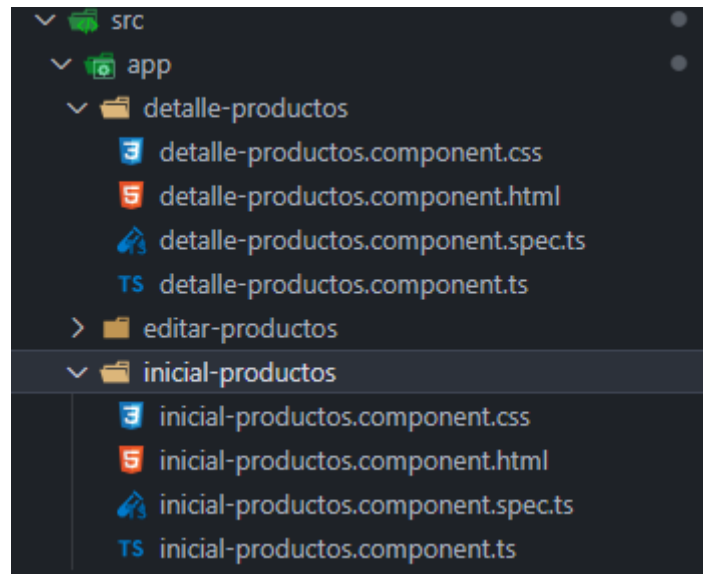
FRONTEND FRUVER CHAZATAR

PRESENTADO POR
JAIME FERNANDO CHAZATAR CARANGUAY

PRESENTADO A
MG. VICENTE AUX REVELO

DIPLOMADO DE NUEVAS TECNOLOGÍAS PARA EL DESARROLLO DE SOFTWARE
UNIVERSIDAD DE NARIÑO
JULIO DE 2023

1. Para el desarrollo de nuestro Frontend utilizamos el **framework Angular**. En primer lugar creamos los componentes respectivos para las dos interfaces que vamos a desarrollar usando el comando *ng generate component*



2. Agregamos las rutas de los 2 nuevos componentes que creamos dentro del archivo *app-routing.modules* y definimos el *component: InicialProductosComponent* como componente en ruta predeterminado

```
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { ListaProductosComponent } from '../lista-productos/lista-productos.component';
4 import { EditarProductosComponent } from '../editar-productos/editar-productos.component';
5 import { InicialProductosComponent } from '../inicial-productos/inicial-productos.component';
6 import { DetalleProductosComponent } from '../detalle-productos/detalle-productos.component';
7
8 const routes: Routes = [
9
10   { path: 'productos/lista', component: ListaProductosComponent },
11   { path: 'productos/editar/:idProducto', component: EditarProductosComponent },
12   { path: 'productos', component: InicialProductosComponent },
13   { path: 'productos/detalle/:idProducto', component: DetalleProductosComponent },
14   { path: 'productos/agregar', component: EditarProductosComponent },
15   { path: '', redirectTo: '/productos', pathMatch: 'full' },
16
17 ];
18
19 @NgModule({
20   imports: [RouterModule.forRoot(routes)],
21   exports: [RouterModule]
22 })
23 export class AppRoutingModule { }
24
```

TERMINAL

```
✓ Compiled successfully.
✓ Browser application bundle generation complete.

5 unchanged chunks

Build at: 2023-07-19T15:08:24.638Z - Hash: bd6d335c26533e64 - Time: 319ms

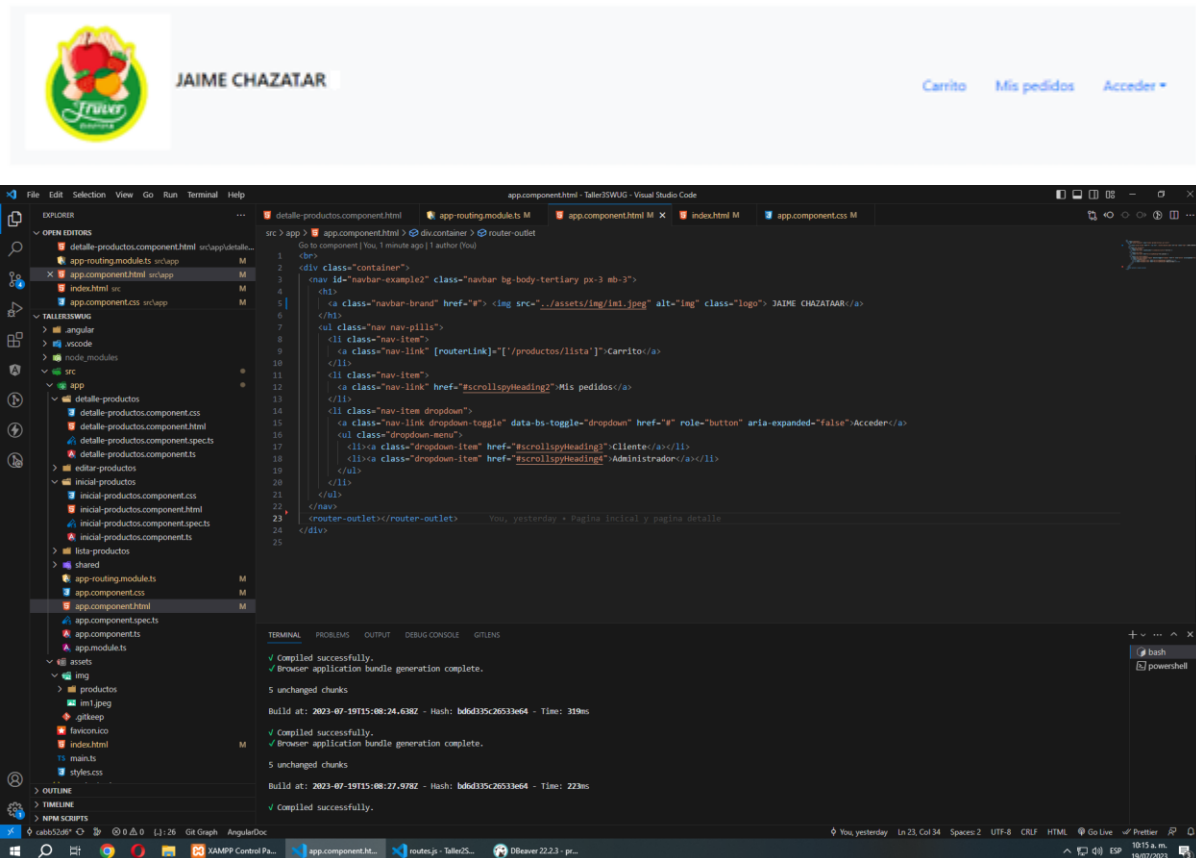
✓ Compiled successfully.
✓ Browser application bundle generation complete.

5 unchanged chunks

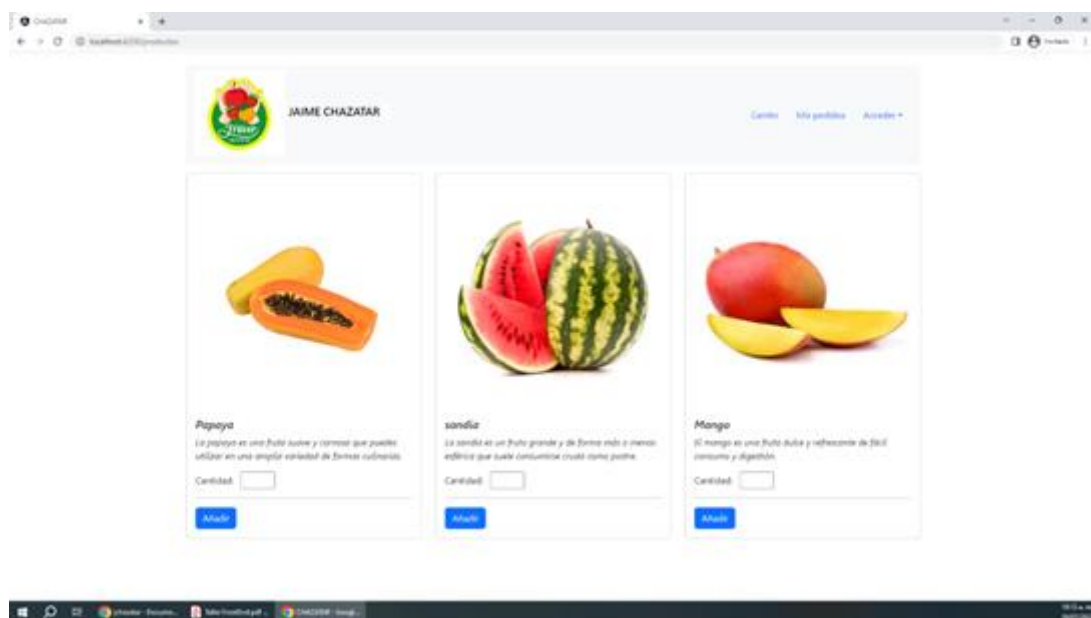
Build at: 2023-07-19T15:08:27.978Z - Hash: bd6d335c26533e64 - Time: 223ms

✓ Compiled successfully.
```

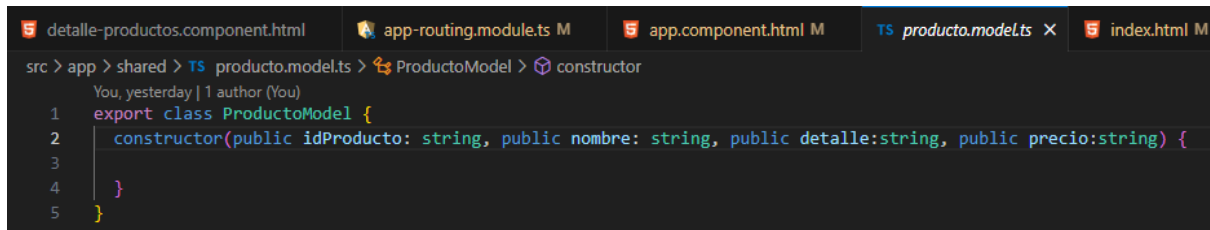
3. Ahora procedemos a crear la barra de navegación en el archivo **app.component** de manera que este elemento pueda aparecer en todas las páginas a las que nos redirijamos y agregamos un **[routerLink]** al logo para poder regresar a la página inicial cuando lo necesitemos



4. Diseñamos la interfaz inicial que contendrá los productos almacenados en nuestra base de datos anteriormente creada.

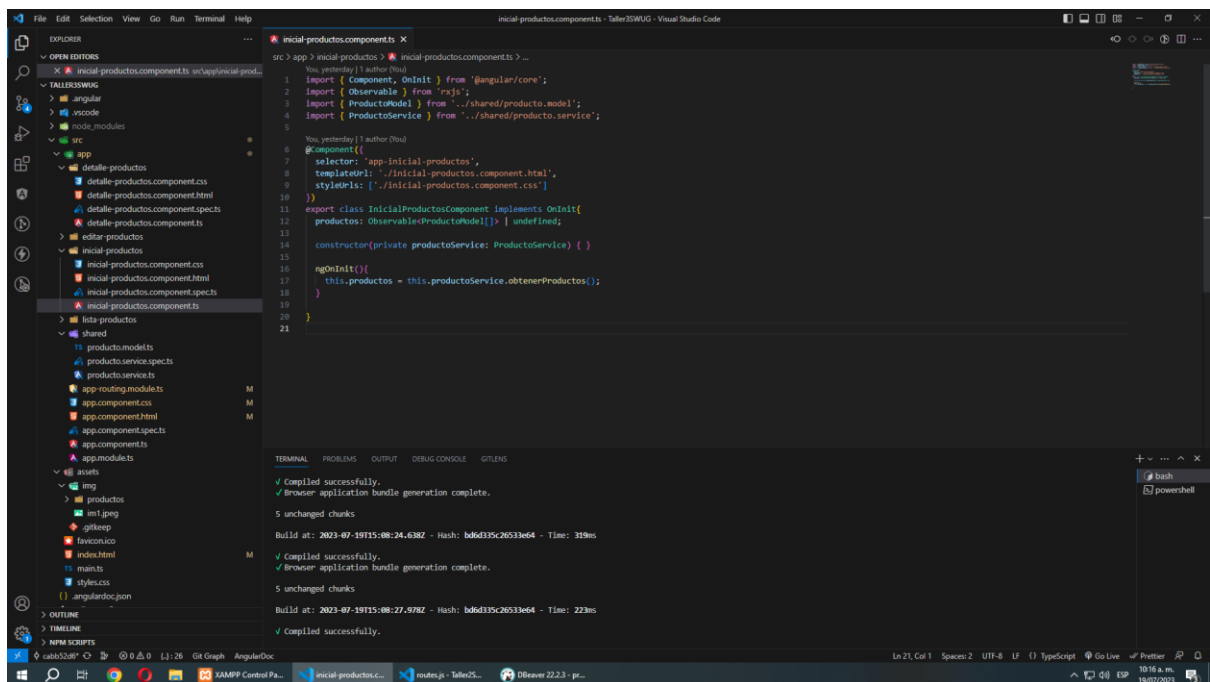


Para ello utilizamos un modelo constructor con el cual traeremos todos los datos que necesitamos de los diferentes productos



```
src > app > shared > TS product.model.ts > ProductoModel > constructor
You, yesterday | 1 author (You)
1 export class ProductoModel {
2   constructor(public idProducto: string, public nombre: string, public detalle:string, public precio:string) {
3
4   }
5 }
```

5. Desde *inicial-productos.components.ts* hacemos un llamado al modelo y hacemos uso del servicio requerido.

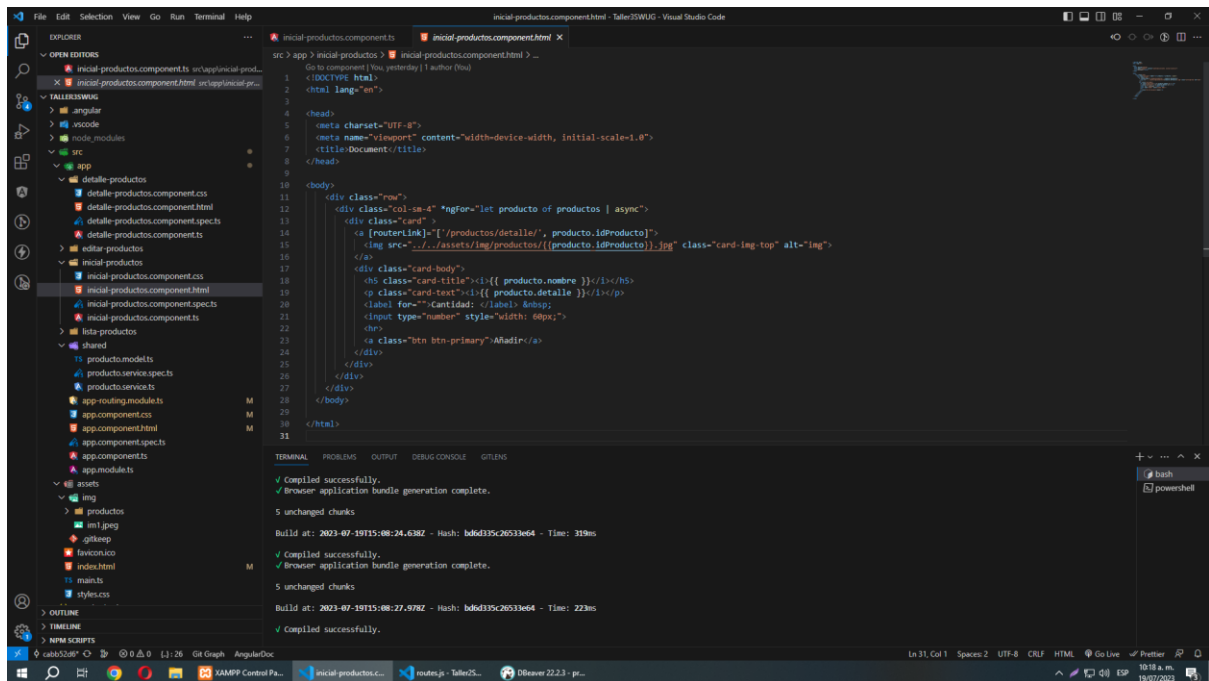


```
src > app > inicial-productos > inicial-productos.components.ts > ...
You, yesterday | 1 author (You)
1 import { Component, OnInit } from '@angular/core';
2 import { Observable } from 'rxjs';
3 import { ProductoModel } from '../shared/producto.model';
4 import { ProductoService } from '../shared/producto.service';
5
6 You, yesterday | 1 author (You)
7 @Component({
8   selector: 'app-inicial-productos',
9   templateUrl: './inicial-productos.component.html',
10   styleUrls: ['./inicial-productos.component.css']
11 })
12 export class InicialProductosComponent implements OnInit {
13   products: Observable<ProductoModel[]> | undefined;
14
15   constructor(private productoService: ProductoService) {}
16
17   ngOnInit() {
18     this.products = this.productoService.obtenerProductos();
19   }
20
21 }
```

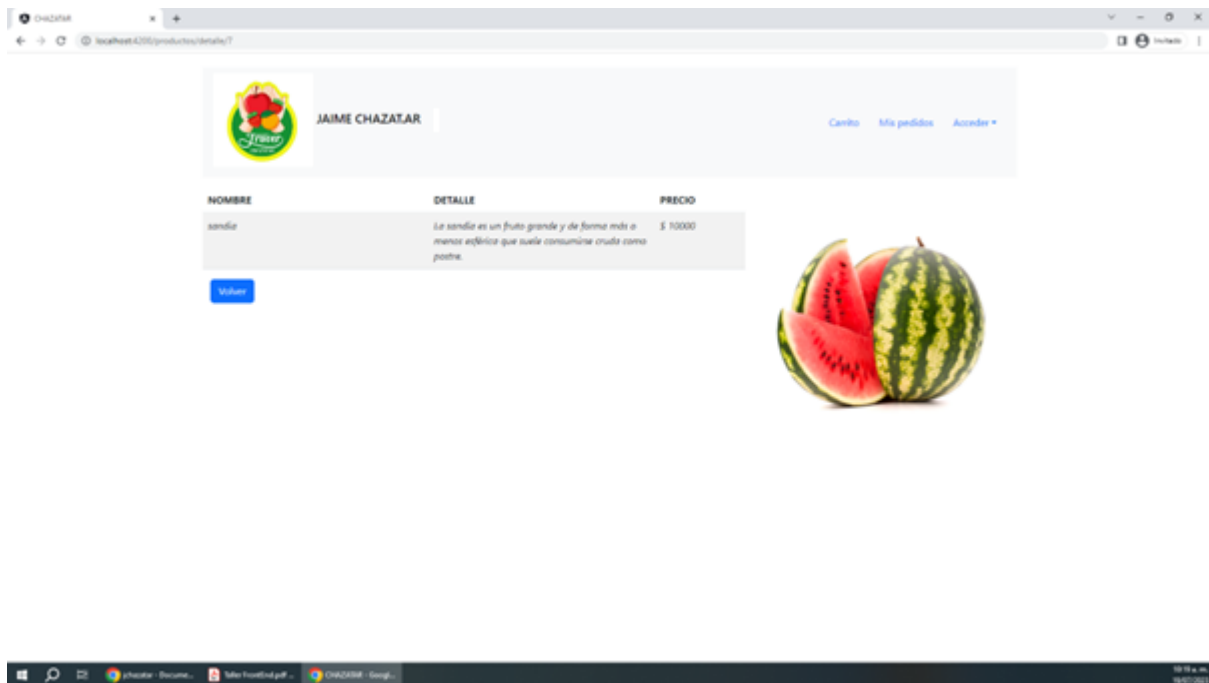
```
✓ Compiled successfully.
✓ Browser application bundle generation complete.
5 unchanged chunks
Build at: 2023-07-19T15:08:24-0302 - Hash: b6d335c26533e44 - Time: 319ms
✓ Compiled successfully.
✓ Browser application bundle generation complete.
5 unchanged chunks
Build at: 2023-07-19T15:08:27-0782 - Hash: b6d335c26533e44 - Time: 223ms
✓ Compiled successfully.
```

6. Posteriormente crearemos la interfaz con las diferentes *card* de los productos que a su vez se desarrollan usando un `*ngFor` para cada producto llamado de la base de datos. Agregamos también un `[routerLink]` a cada imagen de los productos que serán el enlace por el cual nos redirigiremos a los detalles de cada uno.

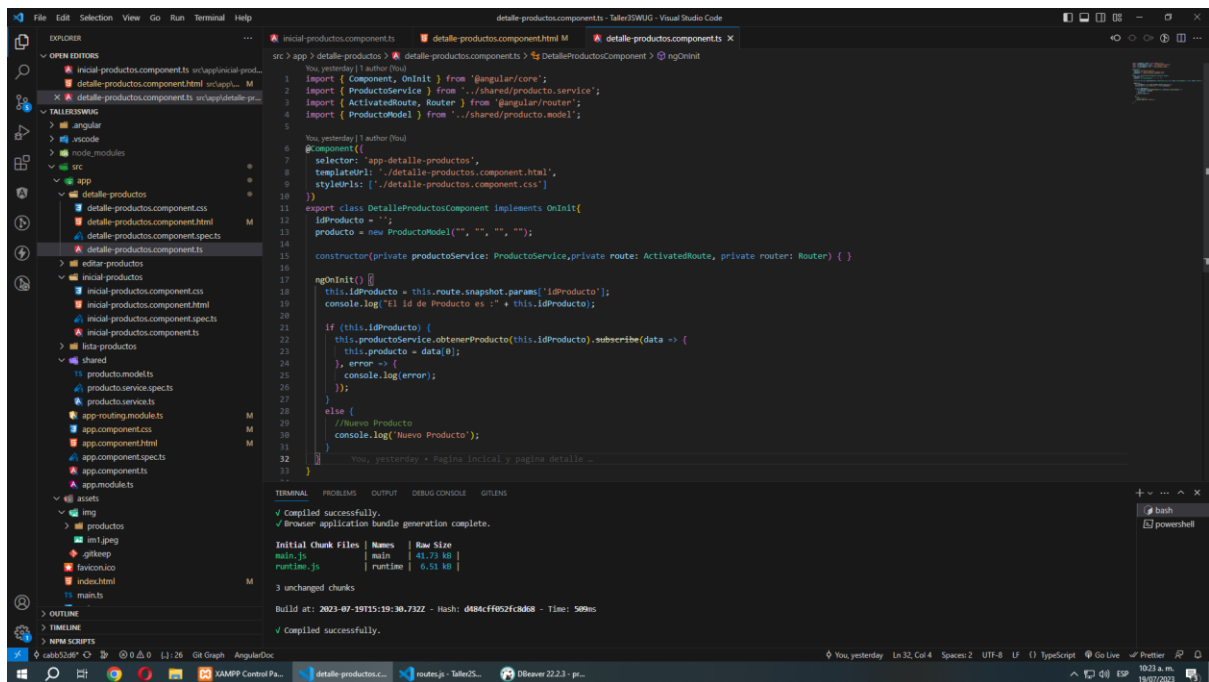
En el `[routerLink]` debemos recuperar el id de cada producto y enviarlo junto a la ruta para que cuando vayamos a mostrar el detalle lo haga solamente del producto que seleccionamos, el `[routerLink]` tendrá la ruta a *component*:
DetalleProductosComponent que explicaremos más adelante



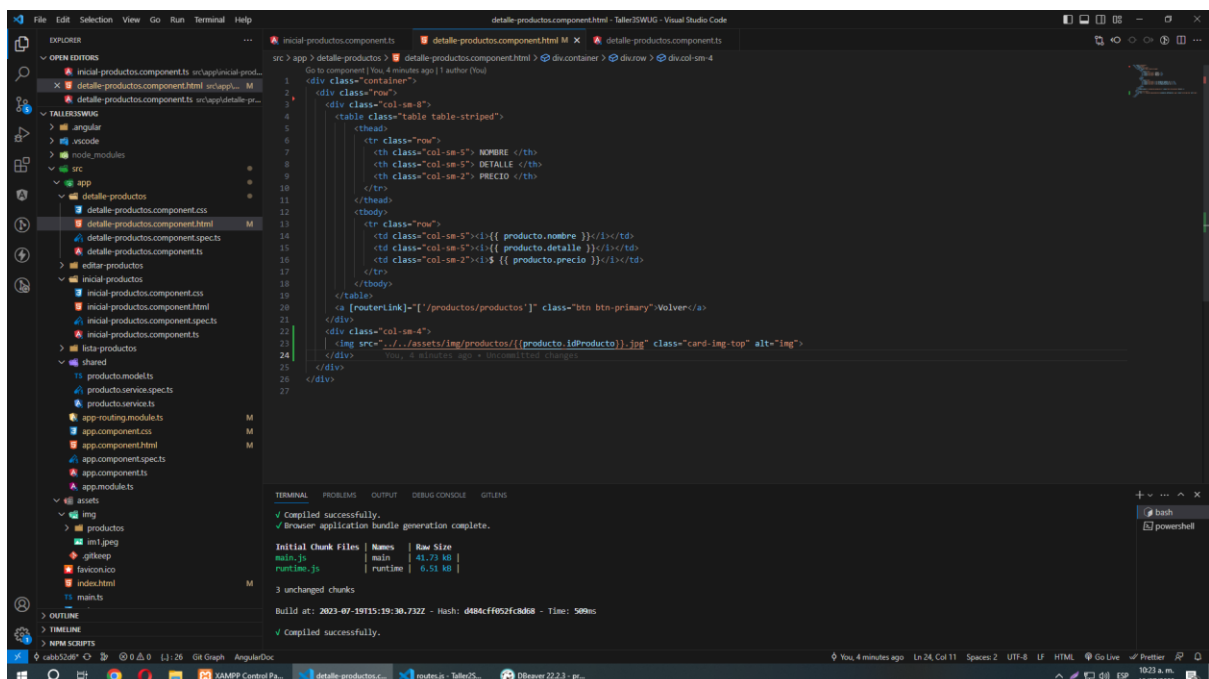
7. Desarrollamos la interfaz detalle para los productos de nuestra base de datos.



8. Incorporaremos en el archivo *detalle-productos.component.ts* el uso del servicio *obtenerProducto* al cual le será enviado el *idProducto* que obtenemos cuando se da click a la imagen del producto, esto retornara el modelo del producto con los datos de ese *idProducto* para mostrar el detalle de cada producto especificado.



9. En el archivo *detalle-productos.component.html* hacemos la estructura html que tendrá la página del detalle de productos en la cual llamaremos los datos que necesitamos de nuestra base de datos y que estarán almacenados en la variable *producto*, esta tendrá el modelo con los datos del producto seleccionado e identificado con el *idProducto*,



Anexo - Algunas propuestas de logo diseñadas antes de la selección final

