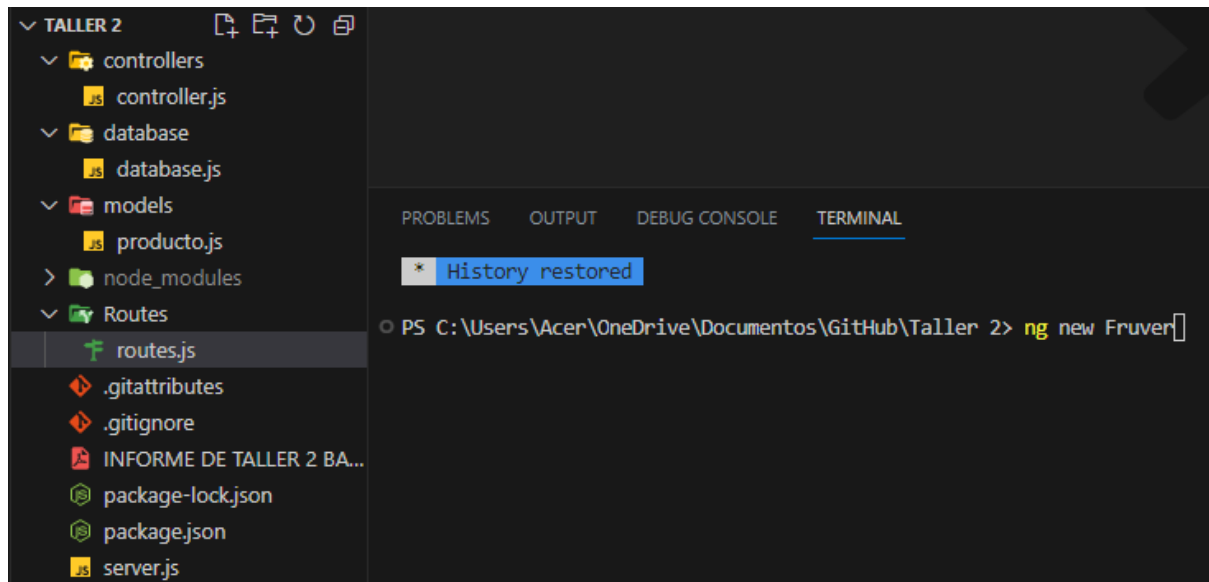


INFORME DE TALLER 3 FRONTEND

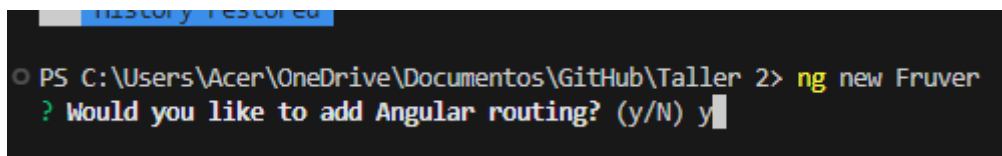
Para la ejecución de este taller se necesita la base de datos así como el backend del Taller 2 activos.

- Se ejecuta el siguiente comando dentro de la carpeta del backend del anterior taller



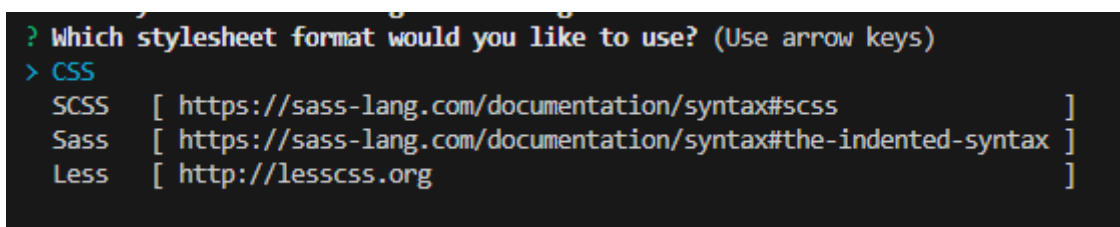
```
PS C:\Users\Acer\OneDrive\Documentos\GitHub\Taller 2> ng new Fruver
```

- Se acepta el routing



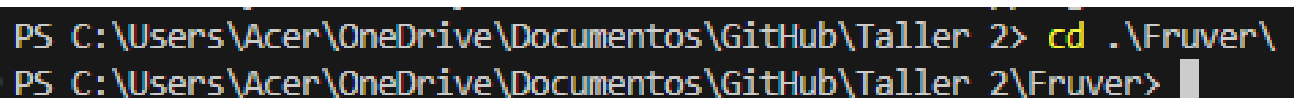
```
? Would you like to add Angular routing? (y/N) y
```

- Se elige CSS



```
? Which stylesheet format would you like to use? (Use arrow keys)  
> CSS  
SCSS [ https://sass-lang.com/documentation/syntax#scss ]  
Sass [ https://sass-lang.com/documentation/syntax#the-indented-syntax ]  
Less [ http://lesscss.org ]
```

- Se traslada al nuevo directorio de trabajo



```
PS C:\Users\Acer\OneDrive\Documentos\GitHub\Taller 2> cd ../Fruver/  
PS C:\Users\Acer\OneDrive\Documentos\GitHub\Taller 2\Fruver>
```

- Se genera el componente main dentro de la carpeta components

```
\Documentos\GitHub\Taller 2\Fruver> ng g c /components/main
```

- Se genera el componente details dentro de la carpeta components

```
\Taller 2\Fruver> ng g c /components/details
```

- Se genera el servicio prods dentro de la carpeta services

```
\Taller 2\Fruver> ng g s /services/prods
```

- Se bajan las siguientes imágenes dentro de la carpeta de assets





- Se trae las configuraciones de CSS y JavaScript de BootsTrap al index de la carpeta madre del frontend

```
Explorer (Ctrl+Shift+E) ... index.html U X
> OPEN EDITORS
TALLER 2
  > controllers
  > database
  > Fruver
  > .angular
  > .vscode
  > node_modules
  > src
    > app
    > assets
    > favicon.ico
    > index.html
    > main.ts
    > styles.css
    > .editorconfig
    > .gitignore

Fruver > src > index.html > html > body > script
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>Fruver</title>
6   <base href="/">
7   <meta name="viewport" content="width=device-width, initial-scale=1">
8   <link rel="icon" type="image/x-icon" href="favicon.ico">
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet"
10     integrity="sha384-9ndCyUaIbzA12FUVXJi0CjMCapSm075npJef0486qLnuZ2cdeRh002iuK6FUUVM" crossorigin="anonymous">
11 </head>
12 <body>
13   <app-root></app-root>
14   <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"
15     integrity="sha384-I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/VR+Vc4jQKc+hVqc2pM80Dewa9n" crossorigin="anonymous"></script>
16   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js"
17     integrity="sha384-fbbOQedDUMZ5KreZpsbe1LCZPVmfTnH7ois6mU1QK+m14rQ112bG8q41eYeH/Fs" crossorigin="anonymous"></script>
18 </body>
19 </html>
```

- Se importan los módulos de FormsModule y HttpClientModule, así como los componentes creados en el archivo app.module.ts

```
app.module.ts U X
Fruver > src > app > app.module.ts > AppModule
1  import { NgModule } from '@angular/core';
2  import { BrowserModule } from '@angular/platform-browser';
3  import { HttpClientModule } from '@angular/common/http';
4  import { FormsModule } from '@angular/forms';
5
6  import { AppRoutingModuleModule } from './app-routing.module';
7  import { AppComponent } from './app.component';
8  import { MainComponent } from './components/main/main.component';
9  import { DetailsComponent } from './components/details/details.component';
10
11  @NgModule({
12    declarations: [
13      AppComponent,
14      MainComponent,
15      DetailsComponent
16    ],
17    imports: [
18      BrowserModule,
19      AppRoutingModuleModule,
20      HttpClientModule,
21      FormsModule,
22    ],
23    providers: [],
24    bootstrap: [AppComponent]
25  })
26  export class AppModule {}
```

- Se Crea las rutas respectivas para cada componente en el archivo app-routing.module.ts

```
app-routing.module.ts U X
Fruver > src > app > app-routing.module.ts > ...
1  import { NgModule } from '@angular/core';
2  import { RouterModule, Routes } from '@angular/router';
3  import { MainComponent } from './components/main/main.component';
4  import { DetailsComponent } from './components/details/details.component';
5
6  const routes: Routes = [
7    {path: '**',redirectTo: '/productos',pathMatch: 'full'},
8    {path: 'productos',component: MainComponent},
9    {path: 'productos/:id',component: DetailsComponent}
10 ];
11
12 @NgModule({
13   imports: [RouterModule.forRoot(routes)],
14   exports: [RouterModule]
15 })
16 export class AppRoutingModuleModule {}
```

- Se modifica el servicio prods.services.ts

```
prods.service.ts U X
Fruver > src > app > services > prods.service.ts > ProdsService > obtenerProducto
1  import { HttpClient } from '@angular/common/http';
2  import { Injectable } from '@angular/core';
3
4  @Injectable({
5    providedIn: 'root'
6  })
7  export class ProdsService {
8
9    URL='http://localhost:3000'
10   constructor(private http:HttpClient) { }
11
12   obtenerProductos(){
13     return this.http.get<any>(this.URL+'/productos')
14   }
15
16   obtenerProducto(id:string){
17     return this.http.get<any>(this.URL+'/productos/'+id)
18   }
19 }
20
```

- Se modifica el archivo app.component.html

```
app.component.html U X
Fruver > src > app > app.component.html > router-outlet
Go to component
1  <nav class="navbar navbar-expand-lg bg-body-tertiary">
2    <div class="container-fluid">
3      
4      <div class="navbar-collapse" id="navbarSupportedContent">
5        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
6          <li class="nav-item">
7            <a class="nav-link active" aria-current="page" [routerLink]="['/productos']">Productos</a>
8          </li>
9        </ul>
10       <form class="d-flex" role="search">
11         <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
12         <button class="btn btn-outline-success" type="button">Search</button>
13       </form>
14       <ul class="navbar-nav mb-2 mb-lg-0 px-3">
15         <li class="nav-item">
16           <a class="nav-link" aria-current="page" >Registrance</a>
17         </li>
18       </ul>
19     </div>
20   </div>
21 </nav>
22
23 <router-outlet></router-outlet>
```

- Se modifica el archivo main.component.html y main.component.ts

```

main.component.html U X
Fruver > src > app > components > main > main.component.html > div..container.d-flex
Go to component
1 <div class="container d-flex">
2   <div class="card m-4" style="width: 18rem;" *ngFor="let prod of prods">
3     <img [src]="imgs[prods.indexOf(prod)]" class="card-img-top" [routerLink]="['/productos/',prod.id]">
4     <div class="card-body container">
5       <h4>Desde {{ObtainValue()}}000</h4>
6       <h5 class="card-title">{{prod.nombre}}</h5>
7       <p class="card-text">{{prod.detalle}}</p>
8       <a [routerLink]="['/productos/',prod.id]" class="btn btn-danger">Añadir al Carrito</a>
9     </div>
10  </div>
11 </div>

```

```

main.component.ts U X
Fruver > src > app > components > main > main.component.ts > MainComponent >
1 import { Component } from '@angular/core';
2 import { ProdsService } from 'src/app/services/prods.service';
3
4 @Component({
5   selector: 'app-main',
6   templateUrl: './main.component.html',
7   styleUrls: ['./main.component.css']
8 })
9 export class MainComponent {
10   producto={
11     id:0,
12     nombre:'',
13     detalle:''
14   }
15
16   prods:any[]=[this.producto]
17
18   imgs=[
19     'assets/Manzana.png',
20     'assets/pera.png',
21     'assets/lechuga.png',
22     'assets/espinaca.png',
23   ]
24
25   constructor(private prodServ:ProdsService){
26     this.prodServ.obtenerProductos().subscribe(res=>{
27       this.prods=res
28     })
29   }
30
31   ObtainValue(){
32     return Math.floor( Math.random() * (10 - 1) + 1);
33   }
34 }

```

- Se modifica el archivo details.component.html y details.component.ts

```

details.component.html U X
Fruver > src > app > components > details > details.component.html > div.container
Go to component
1 <div class="container">
2   <div class="row">
3     <div class="col-6">
4       <img [src]="imgs[parseInt()-1]" style="height: 70vh;">
5     </div>
6     <div class="col-6 my-auto">
7       <h1>{{prod.detalle}}</h1>
8       <a href="#" class="btn btn-danger">Seguir Comprando</a>
9     </div>
10  </div>
11 </div>

```

```

details.component.ts U X
Fruver > src > app > components > details > details.component.ts > DetailsComponent
1 import { Component } from '@angular/core';
2 import { ActivatedRoute } from '@angular/router';
3 import { ProdsService } from 'src/app/services/prods.service';
4
5 @Component({
6   selector: 'app-details',
7   templateUrl: './details.component.html',
8   styleUrls: ['./details.component.css']
9 })
10 export class DetailsComponent {
11
12   imgs=[
13     'assets/Manzana.png',
14     'assets/pera.png',
15     'assets/lechuga.png',
16     'assets/espinaca.png',
17   ]
18
19   id=''
20
21   prod:any
22
23   constructor(private prodServ:ProdsService,public R:ActivatedRoute){
24     this.id=R.snapshot.params['id']
25     this.prodServ.obtenerProducto(this.id).subscribe(res=>{
26       this.prod=res
27     })
28   }
29
30   parseInt(){
31     return parseInt(this.id)
32   }
33 }

```

- Se activa el servicio con el comando

```
\Documentos\GitHub\Taller 2\Fruver> ng serve
```

ANDRÉS MAURICIO MORA CUASQUER / 218034238

<https://github.com/ANDRYUS5000/Taller-2/tree/Taller3>