



Front End – Semana 7

Carlos Andres Valero Velazco: 100317197

Williams Andrés Muñoz Arboleda, Código de estudiante: 100249876.

Daniel Alfredo Serrato Rios: 100225445

Tutor: Olarte John

POLITECNICO GRANCOLOMBIANO INSTITUCION UNIVERSITARIA

Facultad de Ingeniería, Diseño e Innovación

Colombia, 2023.

La finalidad de este documento es presentar el avance de la implementación de la página con una simulación de conexión a base de datos que fue desarrollado en la última versión de Angular .El objetivo es mantener el diseño de los mockups que fueron previamente propuesta en la semana 3, dentro del adjunto se presentara una breve explicación de como levantar el proyecto y una breve contextualización de las tecnologías utilizadas.

Repositorio: <https://github.com/Andres13025/Modico>

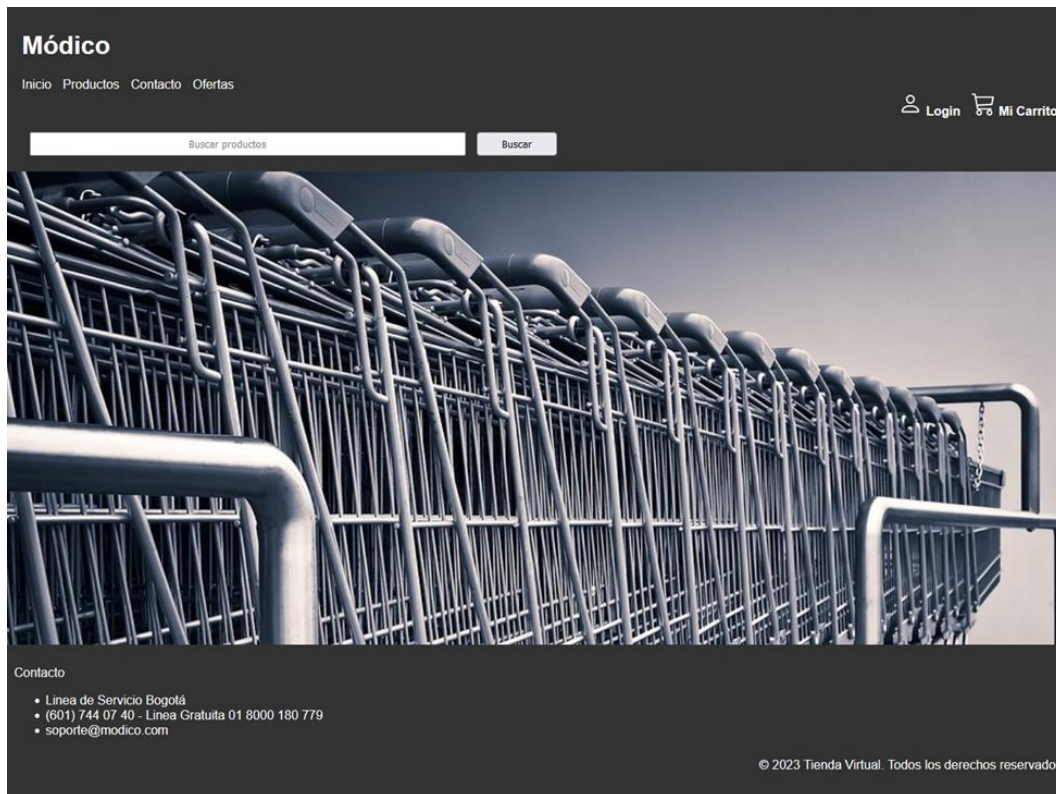
Dentro de esta etapa de desarrollo se decidió migrar el proyecto a un framework (Angular) debido a que se requiere simular la conexión a base de datos y se optó por usar la herramienta integrada de HttpClient que posee el framework, además de otras utilidades como la utilización de Bootstrap y Material.

Se mantuvo la utilización de Bootstrap

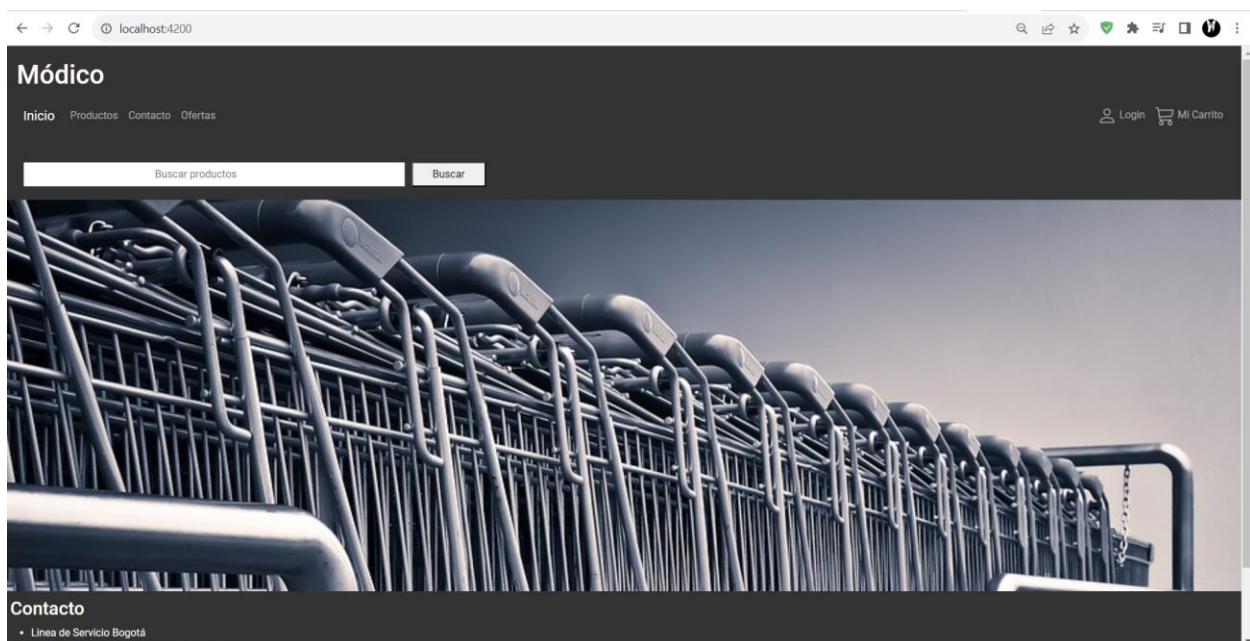
Requisitos:

- Instalar una versión 18.17 superior de Nodejs.

Implementación Home vista Semana 5:



Implementación Home vista Semana 7:

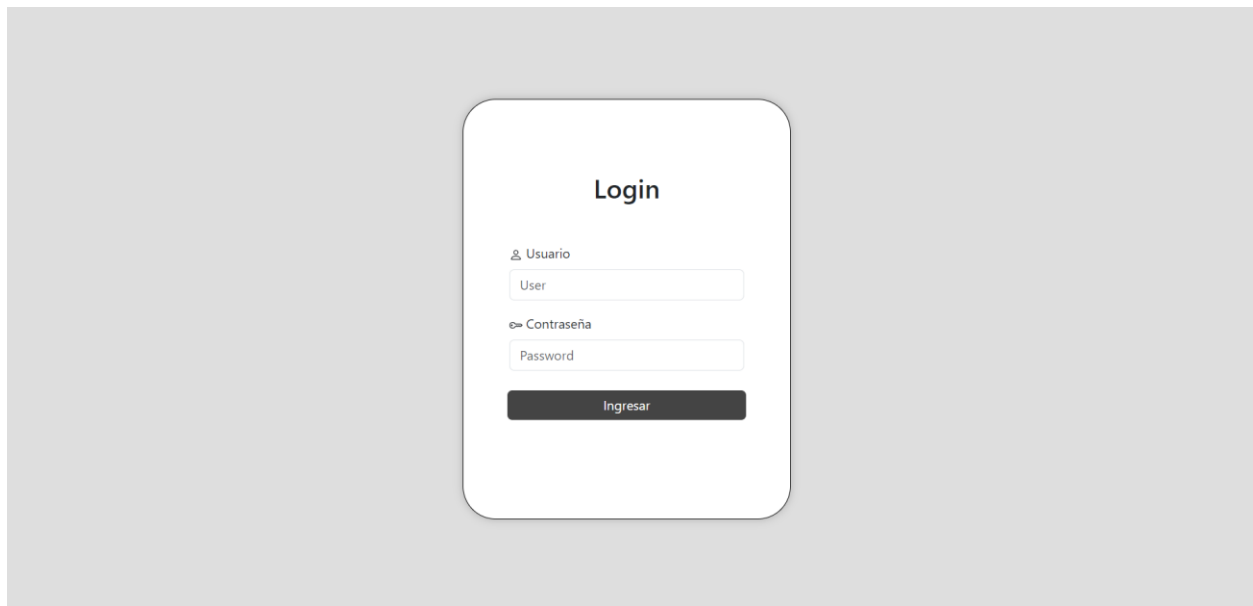


Cambios:

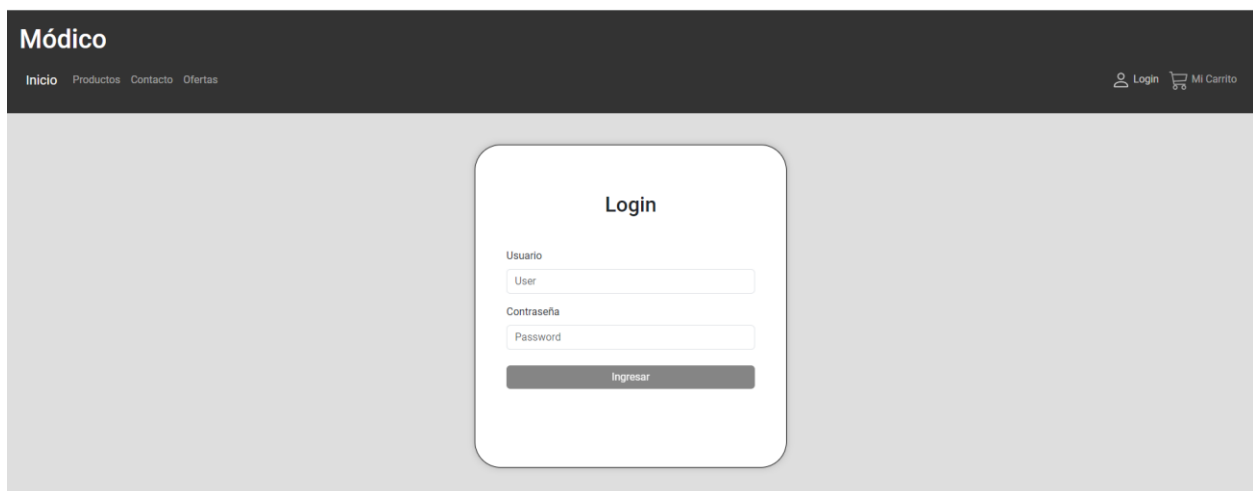
Se ajusto la barra de navegación para dejar al mismo nivel el Login

```
<nav class="navbar navbar-expand-md navbar-dark bg-dark justify-content-center toolbarbos">
  <a routerLink="/" style="margin-left: 10px;" class="navbar-brand d-flex mr-auto">Inicio</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#collapsingNavbar3">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="navbar-collapse collapse w-100" id="collapsingNavbar3">
    <ul class="navbar-nav w-100 justify-content-start">
      <li class="nav-item active">
        <a routerLink="/productos" class="nav-link">Productos</a>
      </li>
      <li class="nav-item active">
        <a routerLink="/" class="nav-link">Contacto</a>
      </li>
      <li class="nav-item">
        <a routerLink="/ofertas" class="nav-link" href="#">Ofertas</a>
      </li>
    </ul>
    <ul class="nav navbar-nav ml-auto w-100 justify-content-end">
      <li class="nav-item">
        <a routerLink="/login" class="nav-link"> <svg
          xmlns="http://www.w3.org/2000/svg"
          width="30" height="30" fill="currentColor"
          class="bi bi-person" viewBox="0 0 16 16">
            <path
              d="M8 8a3 3 0 1 0 0-6 3 3 0 0 0 0 6Zm2-3a2 2 0 1 1-4 0 2 2 0 0 1 4 0Zm4 8c0 1-1 1-1 1H3s-1 0-1 1 1-4 6-4 6 3 6 4Zm-1-.004c-.001-.246-.154-.986-.832-1.664C11.516 10.68 10.289 10 8 10c-2.29 0-3.516.68-4.168 1.332-.678.678-.83 1.418-.832 1.664h10Z"/>
            </svg> Login</a>
      </li>
      <li class="nav-item">
        <a routerLink="/" class="nav-link"> <svg
          xmlns="http://www.w3.org/2000/svg"
          width="30" height="30" fill="currentColor"
          class="bi bi-cart3" viewBox="0 0 16 16">
            <path
```

```
        d="M0 1.5A.5.5 0 0 1 .5 1H2a.5.5 0 0 1 .485.379L2.89
3H14.5a.5.5 0 0 1 .49.598l-1 5a.5.5 0 0 1-.465.401l-9.397.472L4.415 11H13a.5.5 0
0 1 0 1H4a.5.5 0 0 1-.491-.408L2.01 3.607 1.61 2H.5a.5.5 0 0 1-.5-.5zM3.102 4l.84
4.479 9.144-.459L13.89 4H3.102zM5 12a2 2 0 1 0 0 4 2 2 0 0 0 0-4zm7 0a2 2 0 1 0 0
4 2 2 0 0 0 0-4zm-7 1a1 1 0 1 1 0 2 1 1 0 0 1 0-2zm7 0a1 1 0 1 1 0 2 1 1 0 0 1 0-
2z"/>
        </svg>    Mi Carrito</a>
    </li>
    <!-- <li class="nav-item">
        <a class="nav-link" href="#">Right</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#">Right</a>
    </li> -->
</ul>
</div>
</nav>
```

Implementación Login vista Semana 5:

A screenshot of a login form titled "Login" centered on a light gray background. The form is a white rounded rectangle containing the following elements: a title "Login", a label "Usuario" with a person icon, a text input field with the placeholder "User", a label "Contraseña" with a key icon, a text input field with the placeholder "Password", and a dark gray button labeled "Ingresar".

Implementación Login vista Semana 7:

A screenshot of a login form titled "Login" centered on a light gray background, now part of a web application. At the top is a dark gray header with the text "Módico" on the left and navigation links "Inicio", "Productos", "Contacto", and "Ofertas" in the middle. On the right of the header are links for "Login" (with a person icon) and "Mi Carrito" (with a shopping cart icon). The login form itself is a white rounded rectangle with the following elements: a title "Login", a label "Usuario", a text input field with the placeholder "User", a label "Contraseña", a text input field with the placeholder "Password", and a dark gray button labeled "Ingresar".

Cambios:

Se agrego que los campos fueran requeridos y el botón se deshabilitara si no están llenos además de que en el componente se valida que el usuario exista dentro del Json especificado, adema de utilizar formularios reactivos para actualizar el estado de los inputs.

```
import { Component, OnInit } from '@angular/core';
import { FormBuilder, FormGroup, Validators } from '@angular/forms';
import { LoginServiceService } from '../service/login-service.service';
import { usuario } from '../models/user.model';
import { Router } from '@angular/router';

@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent implements OnInit {
  form: FormGroup;
  user: usuario[] = [];

  constructor(private formBuilder: FormBuilder,
    private loginService: LoginServiceService,
    private router: Router) {
    this.form = this.formBuilder.group({
      user: ['', Validators.required],
      password: ['', Validators.required]
    });
  }

  ngOnInit(): void {
    this.loginService.getData().subscribe(data => {
      this.user = data.usuarios;
    });
  }

  validate(): void {
    let currentpassword: string = this.form.controls['password'].value;
    let currentname: string = this.form.controls['user'].value;
    let x = this.user.filter(x => x.password == currentpassword && x.nombre ==
currentname);
    if(x.length > 0){
      this.router.navigate(['/admin']);
    }else{
      alert("Informacion Incorrecta");
    }
    console.log(this.user);
  }
}
```



```
}
import { HttpClient } from '@angular/common/http';
import { Injectable } from '@angular/core';
import { Observable, map } from 'rxjs';

@Injectable({
  providedIn: 'root'
})
export class LoginServiceService {

  private dataUrl = 'assets/user.json'; // Ruta al archivo JSON
  constructor(private http: HttpClient) { }

  getData(): Observable<any> {
    return this.http.get<any[]>(this.dataUrl);
  }

  searchProductByName(name: string): Observable<any[]> {
    return this.http.get<any[]>(this.dataUrl).pipe(
      map((data: any[]) => data.filter(product =>
product.name.toLowerCase().includes(name.toLowerCase()))))
    );
  }
}
```


Implementación Productos Vistas Semana 5:

Módico

MI Cuenta

Mi Carrito


Buscar productos



Lechuga

\$ 4.800


Agregar al Carrito



Agucate

\$ 6.000


Agregar al Carrito



Tomate

\$ 5.000


Agregar al Carrito



Sandia

\$ 10.000


Agregar al Carrito



Durazno

\$ 6.000


Agregar al Carrito



Arandanos

\$ 6.000


Agregar al Carrito



Pimientos

\$ 4.500


Agregar al Carrito



Fresas

\$ 4.500


Agregar al Carrito



Arandanos

\$ 6.000


Agregar al Carrito



Quinoa

\$ 12.000


Agregar al Carrito



Semillas de Chia

\$ 15.000


Agregar al Carrito



Almendras

\$ 9.500


Agregar al Carrito



Crema de Mani

\$ 20.000


Agregar al Carrito



Chocolate amargo

\$ 11.200


Agregar al Carrito



Crema de avellanas

\$ 25.600

Agregar al Carrito



Yogurt Griego

\$ 16.000


Agregar al Carrito

Implementación Productos Vistas Semana 7:

Módico


[Inicio](#)[Productos](#)[Contacto](#)[Ofertas](#)

[Login](#)[Mi Carrito](#)




Lechuga
\$ 4.800

[Agregar al Carrito](#)




Agucate
\$ 6.000

[Agregar al Carrito](#)




Tomate
\$ 5.000

[Agregar al Carrito](#)




Pimientos
\$ 4.500

[Agregar al Carrito](#)




Fresas
\$ 4.500

[Agregar al Carrito](#)




Arandanos
\$ 6.000

[Agregar al Carrito](#)




Sandia
\$ 10.000

[Agregar al Carrito](#)



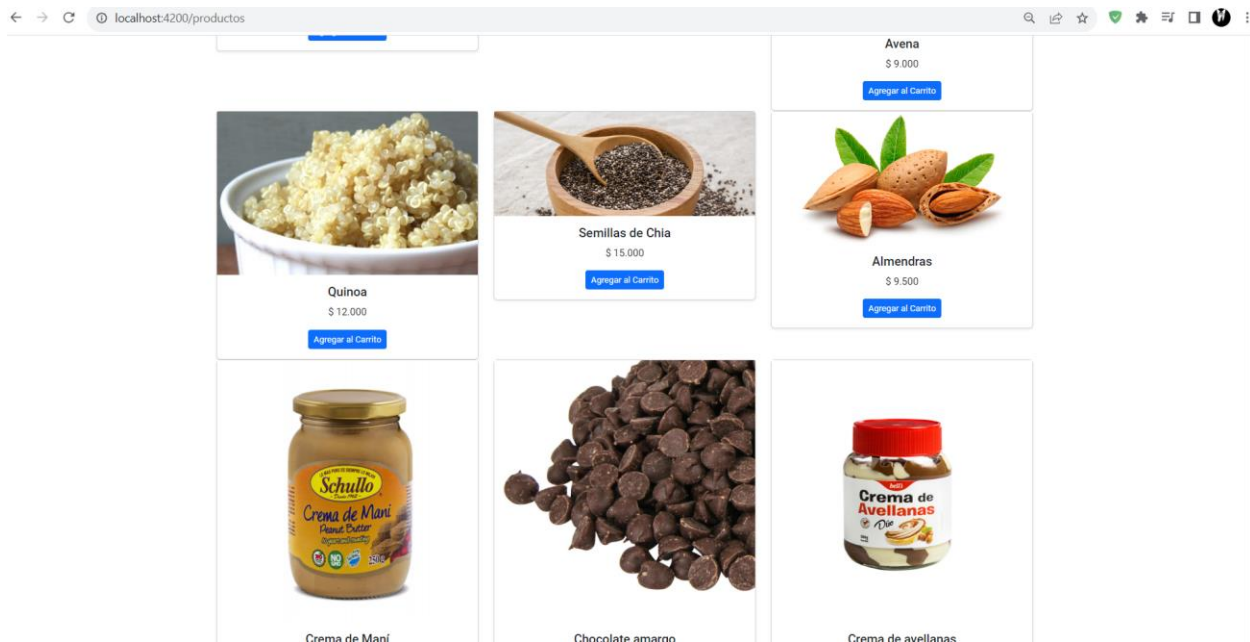
Durazno
\$ 6.000

[Agregar al Carrito](#)



Avena
\$ 9.000

[Agregar al Carrito](#)



Cambios:

Se agrego un servicio para consultar la información de los productos

```
import { HttpClient } from '@angular/common/http';
import { Component } from '@angular/core';
import { Observable, map } from 'rxjs';

@Component({
  selector: 'app-productos',
  templateUrl: './productos.component.html',
  styleUrls: ['./productos.component.css']
})

export class ProductosComponent {

  private dataUrl = 'assets/productos.json';

  productos: any[]=[];

  constructor(private http: HttpClient) { }

  getData(): Observable<any> {
    return this.http.get<any[]>(this.dataUrl);
  }
}
```

```
searchProductByName(name: string): Observable<any[]> {  
  return this.http.get<any[]>(this.dataUrl).pipe(  
    map((data: any[]) => data.filter(product =>  
product.name.toLowerCase().includes(name.toLowerCase())))  
  );  
}  
  
ngOnInit(): void {  
  this.getData().subscribe(data => {  
    this.productos = data.productos;  
    console.log(this.productos);  
  });  
}  
}
```

Implementación Vista Agregar Productos Semana 5

Módico

Productos

PRODUCTOS

Add Product

ID PRODUCTO

PRODUCT PRICE

AMOUNT

PRODUCT NAME

PRODUCT IMAGE

DISCOUNT



Guardar

Product ID	Product Name	Product Price	Product urlimage	Amount	Discount	Eliminar
0011	Fresas	1500	..assets/image/fresa.jpg	3	0	<div>borrar</div>
0011	Lechuga	1000	..assets/image/lechuga.jpg	1	10	<div>borrar</div>
0011	Sandia	5000	..assets/image/sandia.jpg	3	7	<div>borrar</div>

Implementación Vista Agregar Productos Semana 7

Módico

[Inicio](#) [Productos](#) [Contacto](#) [Ofertas](#)

 Login  Mi Carrito

PRODUCTOS

Add Product

ID PRODUCTO

PRODUCT PRICE




AMOUNT

PRODUCT NAME

PRODUCT IMAGE

DISCOUNT

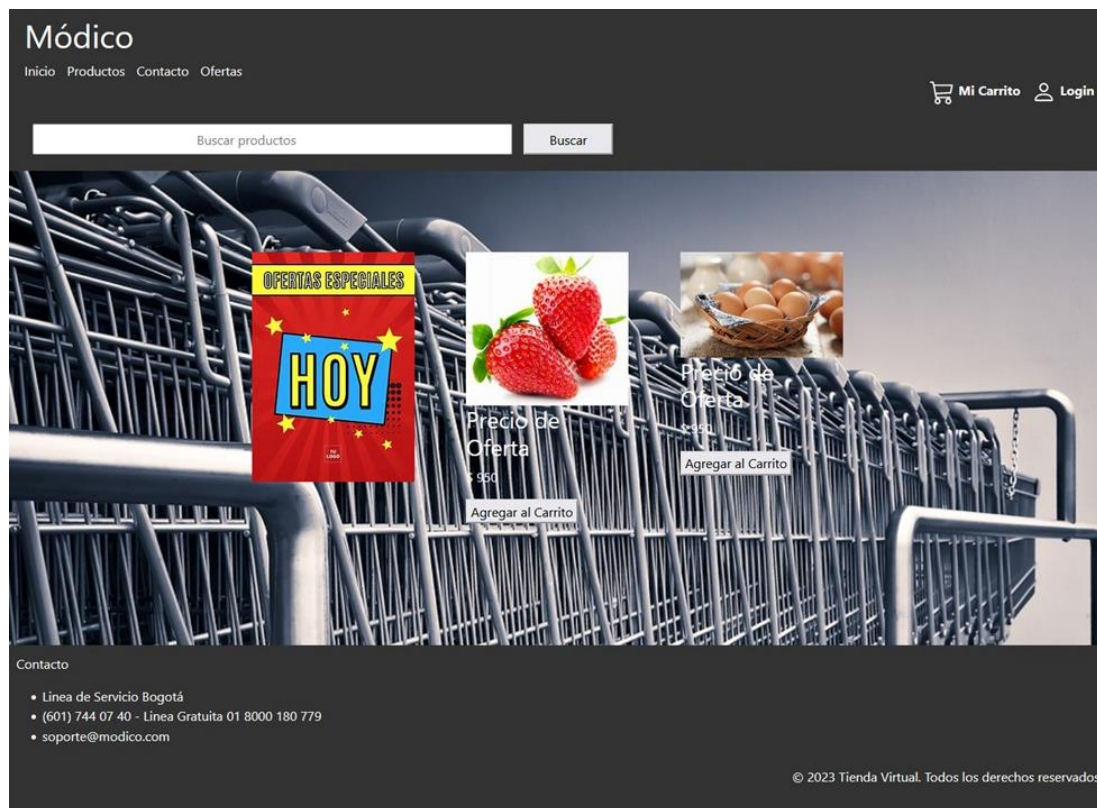
Guardar

Product ID	Product Name	Product Price	Amount	Product urlimage	Discount	Eliminar
001	Fresas	1500	3	https://	0	
002	Lechuga	1000	1	https://	50	
003	Sandia	3000	1	https://	50	

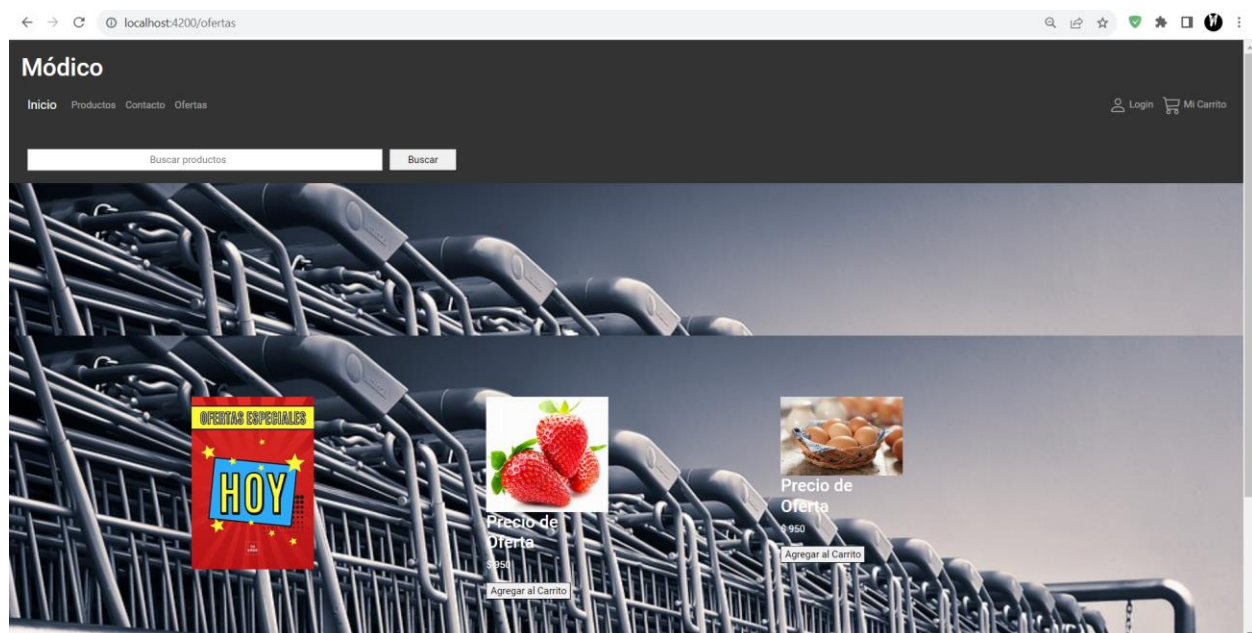
Cambios:

se migro la tabla a un componente de material y se pueden agregar ítems a la tabla además de borrarlos de manera reactiva.

Implementación Vista Ofertas Semana 5



Implementación Vista Ofertas Semana 7



Cambios:

Solo se migro el componente

Adicionales:

Las dependencias dentro del proyecto se manejan dentro modulo siguiente y su configuración de rutas:

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppComponent } from './app.component';
import { AppRoutingModuleModule } from './app-routing.module';
import { LoginComponent } from './login/login.component';
import { AdminComponent } from './admin/admin.component';
import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { MatTableModule } from '@angular/material/table';
import { MatIconModule } from '@angular/material/icon';
import { Route, RouterModule } from '@angular/router';
import { HomeComponent } from './home/home.component';
import { ProductosComponent } from './productos/productos.component';
import { FormsModule, ReactiveFormsModule } from '@angular/forms';
import { OfertasComponent } from './ofertas/ofertas.component';
import { HttpClientModule } from '@angular/common/http';

const routes: Route[] = [
  { path: '', component: HomeComponent },
  { path: "login", component: LoginComponent },
  { path: "admin", component: AdminComponent },
  { path: "productos", component: ProductosComponent },
  { path: "ofertas", component: OfertasComponent },
  { path: "**", redirectTo: "admin" }
];

@NgModule({
  declarations: [
    AppComponent,
    LoginComponent,
    AdminComponent,
    HomeComponent,
    ProductosComponent,
    OfertasComponent
  ],
  imports: [
    BrowserModule,
    BrowserAnimationsModule,
    MatTableModule,
    MatIconModule,
    FormsModule,
    ReactiveFormsModule,
    HttpClientModule,
    AppRoutingModuleModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule {}
```



```
    HttpClientModule,  
    RouterModule.forRoot(routes)  
  ],  
  providers: [],  
  bootstrap: [AppComponent]  
})  
export class AppModule { }
```

Bibliografía

Angular. (s/f). Angular.Io. Recuperado el 11 de octubre de 2023, de <https://angular.io/>

Angular ngfor example. (s/f). StackBlitz. Recuperado el 11 de octubre de 2023, de
<https://stackblitz.com/edit/angular-ngfor-example?file=app%2Fapp.component.ts>

GitHub: Let's build from here. (s/f).