

Componente de Login – parte 1



Componente de Login – Parte 1

En esta lección, vamos a modificar el componente de login para agregar funcionalidad de autenticación en nuestra aplicación.

Paso 1: Modificación del Componente de Cabecera

Primero, ajustamos el componente del cabecero para agregar un enlace de "Login" en la barra de menú de la aplicación.

Código del archivo `cabecero.component.html`:

```
<nav class="navbar navbar-expand-lg">
  <div class="container">
    <!-- Enlace a la página principal (Izquierda) -->
    <a [routerLink]="/" class="navbar-brand">Control Clientes</a>
    <!-- Botón para colapsar la barra de navegación en pantallas pequeñas -->
    <button class="navbar-toggler" type="button"
      data-bs-toggle="collapse" data-bs-target="#navbarCollapse"
      aria-controls="navbarCollapse" aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <!-- Contenido de la barra de navegación -->
```

```
<div class="collapse navbar-collapse" id="navbarCollapse">
  <ul class="navbar-nav me-auto">
    <li class="nav-item px-2">
      <a [routerLink]="/" class="nav-link">Inicio</a>
    </li>
  </ul>
  <!-- Alineado a la derecha pero dentro de los márgenes -->
  <ul class="navbar-nav">
    <li class="nav-item px-2">
      <a [routerLink]="/" class="nav-link">
        <i class="bi bi-person-circle"></i> Bienvenido
      </a>
    </li>
    <li class="nav-item px-2">
      <a [routerLink]="/login" class="nav-link">
        <i class="bi bi-person"></i> Login
      </a>
    </li>
  </ul>
</div>
</div>
</nav>
```

Comentarios:

- Se ha agregado el elemento de Login en el menú, incluyendo un ícono de person de Bootstrap.

Paso 2: Modificación del Componente de Login

Ahora, vamos a crear la estructura básica del componente de login. En este paso, solo agregamos lo mínimo necesario para visualizar el formulario de inicio de sesión.

Código del archivo `login.component.ts`:

```
import { Component } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { Router } from '@angular/router';

@Component({
  selector: 'app-login',
  standalone: true,
  imports: [FormsModule],
  templateUrl: './login.component.html',
```

```

    styleUrls: ['./login.component.css'
})
export class LoginComponent {
  email: string | null = null;
  password: string | null = null;

  constructor(
    private router: Router
  ) { }

  login(){
  }
}

```

Comentarios:

- Hemos creado un componente `LoginComponent` como un componente standalone.
- Utilizamos `FormsModule` para manejar la entrada de datos del formulario.
- Las variables `email` y `password` almacenarán los datos ingresados en el formulario.
- El método `login()` se encargará de la autenticación una vez que se implemente.

Paso 3: Estructura del Formulario de Login

El siguiente paso es diseñar el formulario para que los usuarios puedan ingresar sus credenciales.

Código del archivo `login.component.html`:

```

<header id="main-header" class="py-2 bg-info text-white">
  <div class="container">
    <div class="row">
      <div class="col-md-6">
        <h1><i class="bi bi-person"></i> Login</h1>
      </div>
    </div>
  </div>
</header>

<section id="login">
  <div class="container my-5">
    <div class="row">
      <div class="col-md-6 mx-auto">
        <div class="card">

```

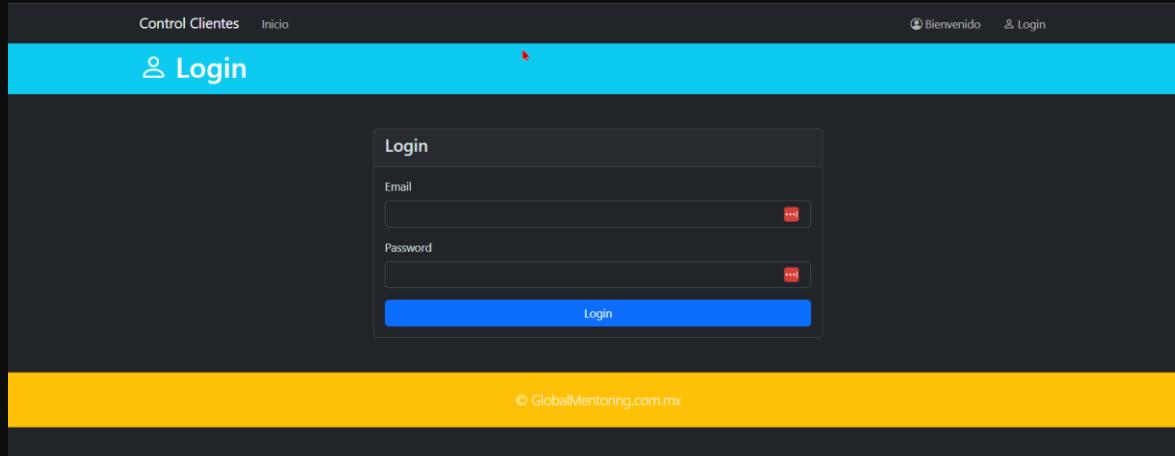
```
<div class="card-header">
  <h4>Login</h4>
</div>
<div class="card-body">
  <form (submit)="login()">
    <div class="mb-3">
      <label for="email" class="form-label">Email</label>
      <input
        type="email"
        class="form-control"
        name="email"
        [(ngModel)]="email"
        required
      />
    </div>
    <div class="mb-3">
      <label for="password" class="form-label">Password</label>
      <input
        type="password"
        class="form-control"
        name="password"
        [(ngModel)]="password"
        required
      />
    </div>
    <input
      type="submit"
      value="Login"
      class="btn btn-primary w-100"
    />
  </form>
</div>
</div>
</div>
</div>
</section>
```

Comentarios:

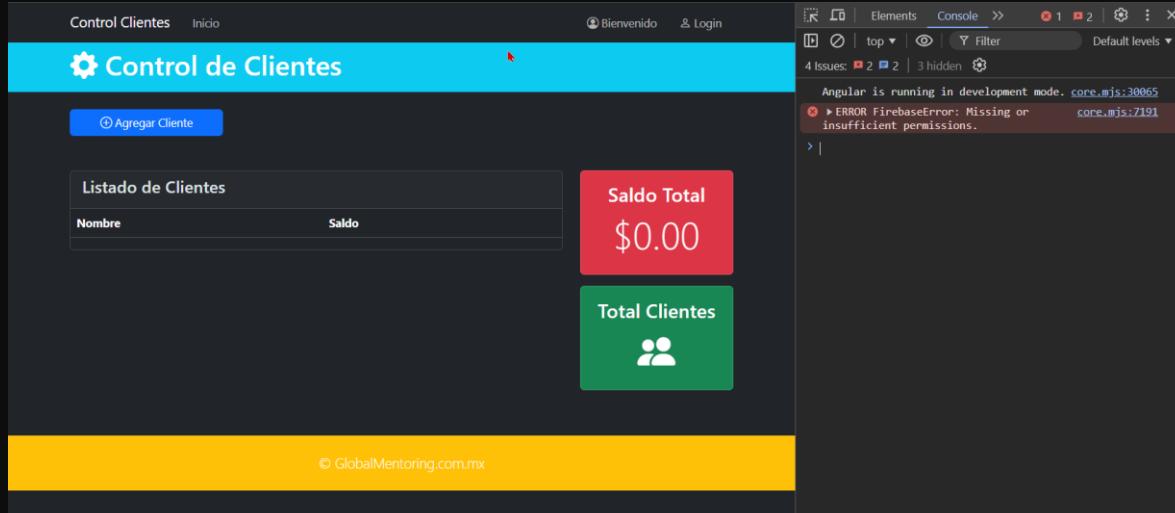
- El formulario utiliza Bootstrap para un diseño limpio y responsive.
- Incluye los campos básicos de **email** y **password** con validación requerida.
- El botón de "Login" es un botón tipo `submit` que llamará al método `login()` cuando el formulario sea enviado.

Resultado hasta el Momento

Al finalizar estos pasos, habremos agregado un enlace para el login en el cabecera y diseñado un formulario básico que se utilizará para la autenticación en nuestra aplicación.



Debido a que ya hemos activado las reglas de seguridad en Cloude Firestore, si intentan acceder al listado de clientes NO obtendrán de momento los datos, esto lo resolveremos más adelante. De momento va a marcar error al obtener el listado de clientes, esto es normal.



Más adelante al hacer login en nuestra aplicación podremos volver a visualizar el listado de clientes.

Recuerda que puedes descargar el código actualizado en la sección de recursos de esta lección.

¡Continúa así , el esfuerzo vale la pena!

Saludos 

Ing. Ubaldo Acosta

Fundador de GlobalMentoring.com.mx