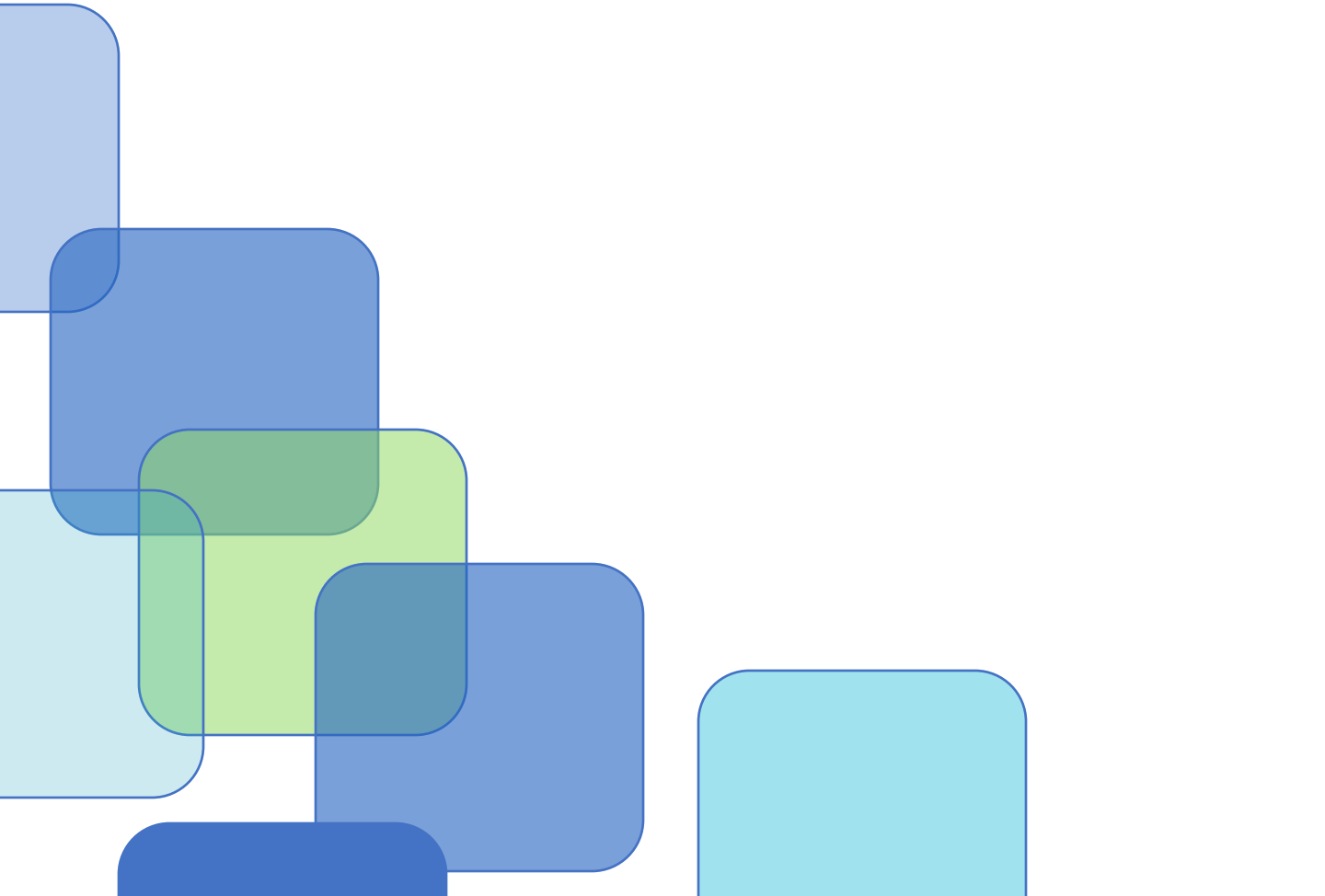




AE-3. Angular

Ana Núñez
Javier Núñez



Para la realización de la actividad hemos creado un Github donde cada integrante ha subido su trabajo. Hemos elegido el trabajo de Javier Núñez para hacer la explicación del funcionamiento del trabajo. En este repositorio hemos subido un video donde se ve el funcionamiento de la web. Este es el enlace:

[Javier-Nunez-Blas/AE-3.-ANGULAR \(github.com\)](https://github.com/Javier-Nunez-Blas/AE-3.-ANGULAR)

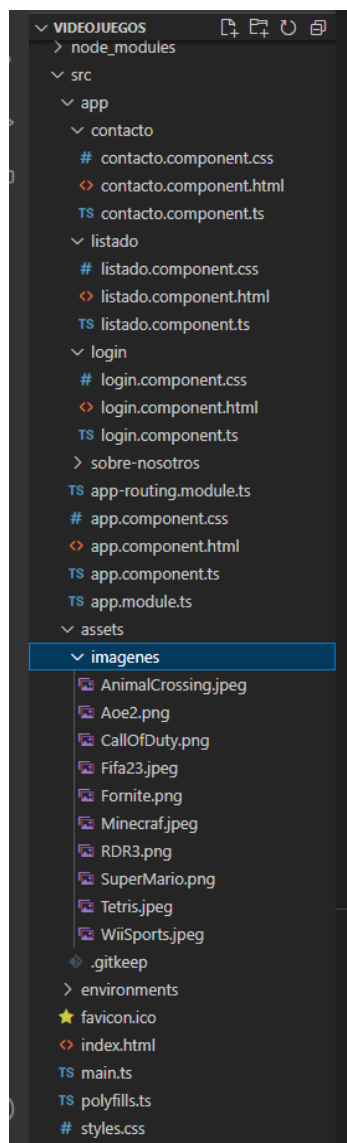
Los usuarios y contraseñas para acceder a la pagina web son:

Usuario: javier Contraseña: javier

Usuario: ana Contraseña: ana

(en minúscula)

Así quedaría el árbol de carpetas con los componentes-



Requerimiento 1 :

- Los videojuegos tendrán un identificador, un título, una compañía, una imagen y una valoración media.

listado.component.ts : Clase videojuego con la lista de videojuegos con todos su atributos.

```
// Añadimos los videojuegos con todos sus atributos a la clase videojuego
videojuegos: Videojuego[] = [
  new Videojuego(1, "AOE2 Definitive Edition", "Nintendo", "Aoe2.png", 9),
  new Videojuego(2, "Fifa 2023", "Konami", "Fifa23.jpeg", 7),
  new Videojuego(3, "Call of duty Modern Warfare 3", "Microsoft", "CallOfDuty.png", 7),
  new Videojuego(4, "Animal crossing", "Microsoft", "AnimalCrossing.jpeg", 5),
  new Videojuego(5, "Fornite", "Sega", "Fornite.png", 6),
  new Videojuego(6, "Red Dead Redemption 2", "Rockstar Studios", "RDR3.png", 9),
  new Videojuego(7, "Super Mario Bros", "Nintendo", "SuperMario.png", 7),
  new Videojuego(8, "Wii Sports", "Nintendo", "WiiSports.jpeg", 5),
  new Videojuego(9, "Tetris", "EA Mobile", "Tetris.jpeg", 3),
  new Videojuego(10, "Minecraft", "CD Projekt Red", "Minecraf.jpeg", 5)
]
```

La web mostrará una página con nuestro nombre de usuario y el listado de videojuegos (tendremos algunos videojuegos ya dados de alta), pero solo se mostrará en el listado el identificador, el título y la valoración media.

`login.component.html` : recoge el name correspondiente al usuario en el formulario

```
<div>
|   <h2>Nombre de usuario: {{ name }}</h2>
</div>
```

`listado.component.html` : recorre la lista de videojuegos y la añade a la tabla con los atributos especificados

```
<tbody>
|   <!-- recorreremos la lista de videojuegos y añadimos a la tabla los atributos -->
|   <tr *ngFor="let videojuego of videojuegos">
|       <td class="medio">{{ videojuego.idVideojuego }}</td>
|       <td class="enlace">
|           <!-- Cuando hacemos click en el titulo, se activa el metodo ver detalle -->
|           <div (click)="verDetalle(videojuego.idVideojuego)">{{ videojuego.titulo }}</div>
|       </td>
|       <td class="medio">{{ videojuego.valoracionMedia }}</td>
|   </tr>
</tbody>
</table>
```

Vista HTML:

Nombre de usuario: ana

Listado de Videojuegos

Id	Título	Valoración Media
1	AOE2 Definitive Edition	9
2	Fifa 2023	7
3	Call of duty Modern Warfare 3	7
4	Animal crossing	5
5	Fornite	6
6	Red Dead Redemption 2	9
7	Super Mario Bros	7
8	Wii Sports	5
9	Tetris	3
10	Minecraft	5

Si pulsamos en un videojuego, nos llevará a su detalle, el cual podremos ver todos los atributos del videojuego. Esta página tendrá también un botón para volver.

Listado.component.html : Con *ngIf mostramos el detalle de los videojuegos y ocultamos la tabla con la lista de videojuego. Cuando clicamos un videojuego, se oculta la lista y aparecen los detalles. Cuando clicamos en volver se activa el metodo volver(), que oculta los detalles y vuelve a mostrar la lista.

```
<!--Si videojuegoSeleccionado es distinto de null, mostramos el div -->
<div *ngIf="videojuegoSeleccionado !== null">
  <br><br><hr>
  <h3>Detalle de Videojuego</h3>
  <!-- Sacamos los atributo de los videojuegos por la id del seleccionado -->
  <h3>{{ videojuegos[videojuegoSeleccionado].titulo }}</h3>
  <p>Id: {{ videojuegos[videojuegoSeleccionado].idVideojuego }}</p>
  <p>Compañía: {{ videojuegos[videojuegoSeleccionado].compania }}</p>
  <p>Imagen :</p>
  <p>Valoración media: {{ videojuegos[videojuegoSeleccionado].valoracionMedia }}</p>

  <button (click)="volver()">Volver</button>
  <hr>
</div>
```

Listado.component.ts : Con la variable videojuegoSeleccionado, le damos dos valores. Number, para recoger la id del videojuego que hemos seleccionado y usarlo para mostrar sus atributos. Null para usarlo en los *ngIf, y mostrar o no la lista o los detalles

```
new Videojuego(10, 'Minecraft', 'CB Projects Red y ...
]
// inicializamos la variable videojuego seleccionado a
videojuegoSeleccionado: number | null = null
// recibe el id de videojuego y cambia la id a -1, pa
verDetalle(id: number): void {
  this.videojuegoSeleccionado = id - 1
}
// cambia el estado de videojuego seleccionado a null
volver(): void {
  this.videojuegoSeleccionado = null
}
}
```

Vista HTML:

Nombre de usuario: ana

Listado de Videojuegos

Id	Título	Valoración Media
1	AOE2 Definitive Edition	9
2	Fifa 2023	7
3	Call of duty Modern Warfare 3	7
4	Animal crossing	5
5	Formite	6
6	Red Dead Redemption 2	9
7	Super Mario Bros	7
8	Wii Sports	5
9	Tetris	3
10	Minecraft	5

Detalle de Videojuego

AOE2 Definitive Edition

Id: 1

Compañía: Nintendo

Imagen :

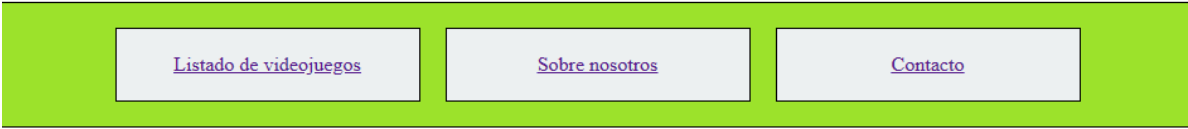


Valoración media: 9

Volver

La página web tendrá también una página que muestre los datos para contactarnos y otra página que hable sobre nosotros. A estas páginas accederemos mediante un menú de navegación (también en este menú tendremos el listado de videojuegos).

Vista HTML:



Requerimiento 2:

Para entrar en la página deberemos de estar registrados, se mostrará una página de “login” al principio donde deberemos de poner nuestro nombre de usuario y nuestra contraseña. La aplicación tendrá algunos usuarios por defecto.

Vista HTML:



Nombre de usuario:

Inicio de sesión

Nombre de usuario:

Contraseña:

Login.componentes.ts : Creamos los usuarios con su contraseña

```
// creamos los usuarios con nombre y contraseña
usuarios = [
  { nombre: 'javier', contrasena: 'javier' },
  { nombre: 'ana', contrasena: 'ana' }
]
```

En caso de que no podamos entrar en la página, se nos mostrará otra vez la página de “login” diciendo que no hemos podido acceder.

Vista HTML:



Nombre de usuario: as

Inicio de sesión

Nombre de usuario:

Contraseña:

Nombre de usuario o contraseña incorrectos

Login.componentes.html : Creamos un formulario con un método enviar() que comprueba que el usuario y la contraseña coincidan con los que tenemos creados. Si son correctos, oculta el formulario y muestra el menú de navegación. Si no, saca un mensaje de error y no se puede acceder al menú de navegación

```
<!--
  Cuando mandamos el formulario, se activa el metodo enviar, que comprueba los
  usuarios y contraseñas. Y cambia la variable login y nologin dependiendo se si
  es correcto o no.
-->
<form (ngSubmit)="enviar()">
  <label for="name">Nombre de usuario:</label>
  <input type="text" id="name" name="name" [(ngModel)]="name" required>
  <br> <br>

  <label for="password">Contraseña:</label>
  <input type="password" id="password" name="password" [(ngModel)]="password" required>
  <br> <br>

  <button type="submit">Entrar</button>
</form>
<!-- Si el usuario o la contraseña son incorrectos avisa con un h3 -->
<h3 *ngIf="noLogin">Nombre de usuario o contraseña incorrectos</h3>
</div>
<!-- Para poder usar los enlaces a otros componentes -->
<router-outlet></router-outlet>
```