UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

ETSE

Desenvolvemento de Aplicacións Web

El Córner

Jorge González Corbelle

Iago Feijóo Rey

Índice general

1 Descripción del proyecto:

"El córner" es una página web informativa orientada al sector deportivo. En ella se podrá acceder a información de diversos deportes, así como las clasificaciones de las distintas competiciones y estadísticas relacionadas con ellas o sobre los equipos o jugadores participantes..

El usuario que acceda a la página podrá navegar entre los distintos deportes que ofrece, pudiendo seleccionar algunos para ver información sobre los deportes seleccionados, así como competiciones o equipos. Además, hay un apartado donde se muestran las noticias más destacadas de los deportes, competiciones o equipos.

Existe la posibilidad de que un usuario sea socio de "El córner", registrándose en la página web. Esto permite al usuario acceder a contenido exclusivo como un apartado que muestra contenido de multimedia sobre los deportes, así como resúmenes o imágenes destacadas de diferentes eventos.

2 Preparación del proyecto:

2.1. Inventario de contenido:

En el siguiente diagrama se muestran los contenidos que serán accesibles desde nuestra página principal. Desde esta, se puede acceder a las distintas pestañas representadas. En "Deportes", el usuario podrá escoger el deporte y la competición de los que quiere ver información. A continuación, puede acceder al resto de ventanas informativas (Resultados, Equipo, Noticias) que muestran estadísticas específicas de los deportes que se ofrecen.

La ventana de "Foro" lleva a un espacio en el que los usuarios pueden ver y comentar sobre los temas más destacados del momento.

"About us" nos lleva a una pestaña de información sobre la página, así como formas de contacto para más información.

Además, hay una ventana de Inicio de sesión/Registrarse (denominada "Perfil")que permite que un usuario se convierta en socio de la página web dándole acceso a la ventana de "Multimedia", en la que se muestran resúmenes o imágenes destacadas de los eventos.

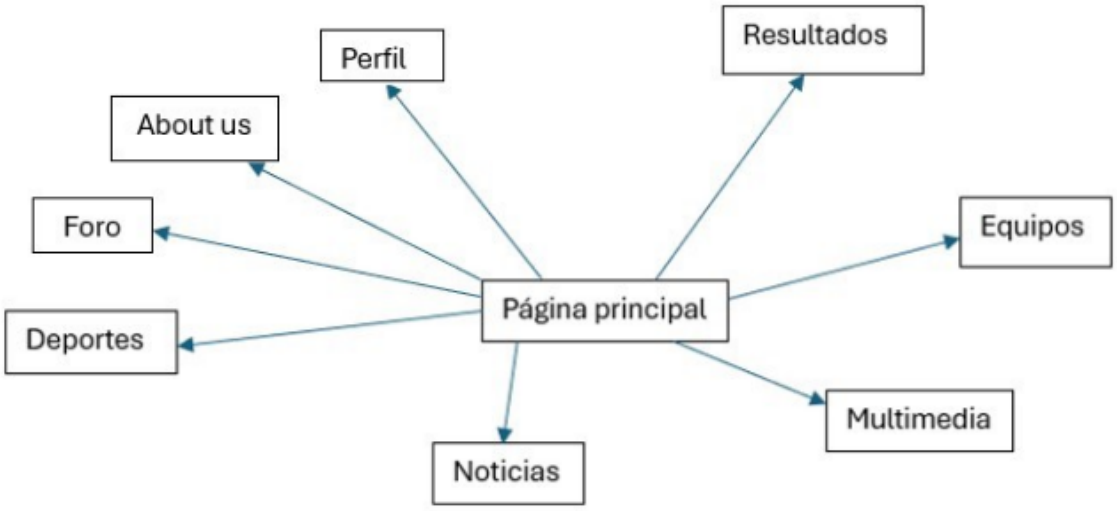


Imagen 2.1: Inventario de contenido de "El córner"

2.2. Arquitectura de Información (AI):

A continuación, procederemos a la presentación de un esquema en el que se puede apreciar la arquitectura de la información que seguiremos en el diseño de nuestra página web. Este esquema nos permitirá obtener una primera aproximación del número de páginas que necesitaremos crear.

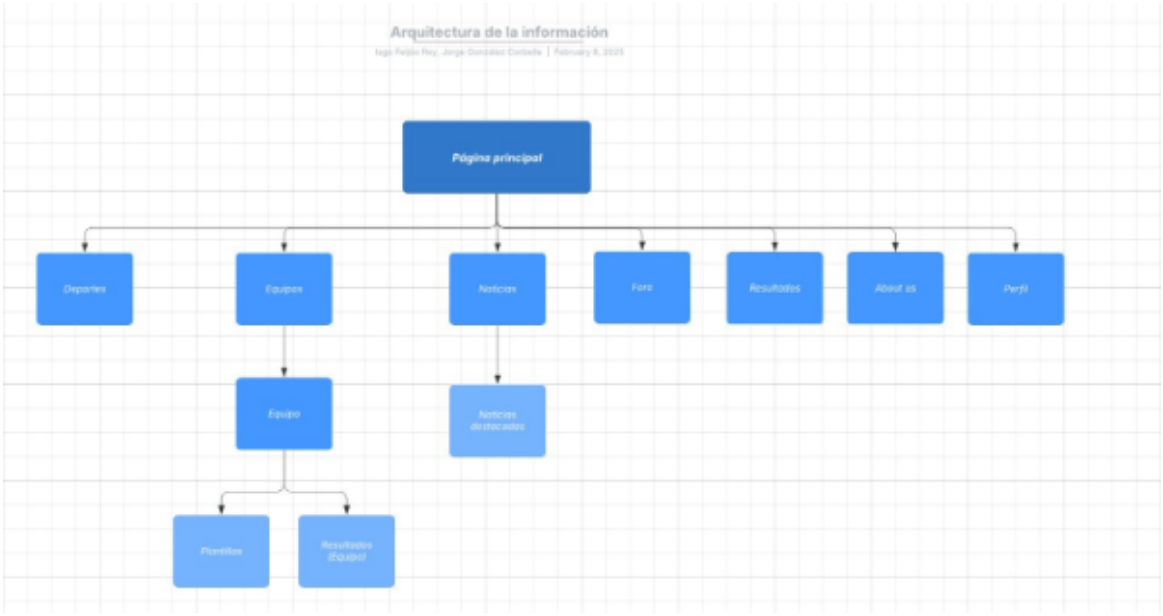


Imagen 2.2: Inventario de contenido de "El córner"

2.3. Mapa Web:

El siguiente mapa web de nuestra página muestra los sitios que son accesibles para los usuarios que visitan "El córner".

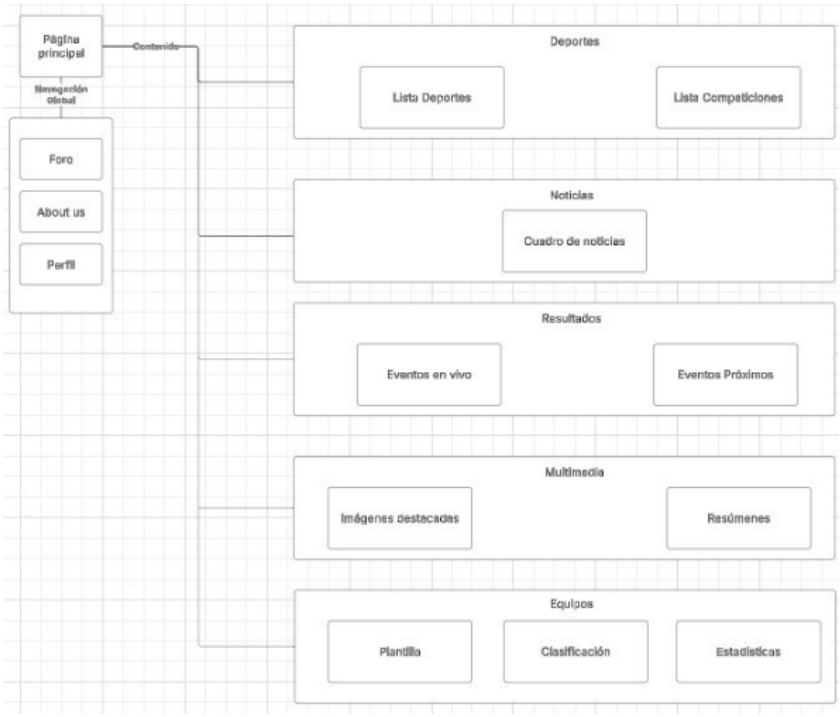


Imagen 2.3: Mapa Web de "El córner"

2.4. Prototipo manual (sketching):

En las siguientes figuras se muestra un prototipo de la página web en la que se recogen las ideas básicas de ella, incluyendo la apariencia final de las páginas que lo conforman.

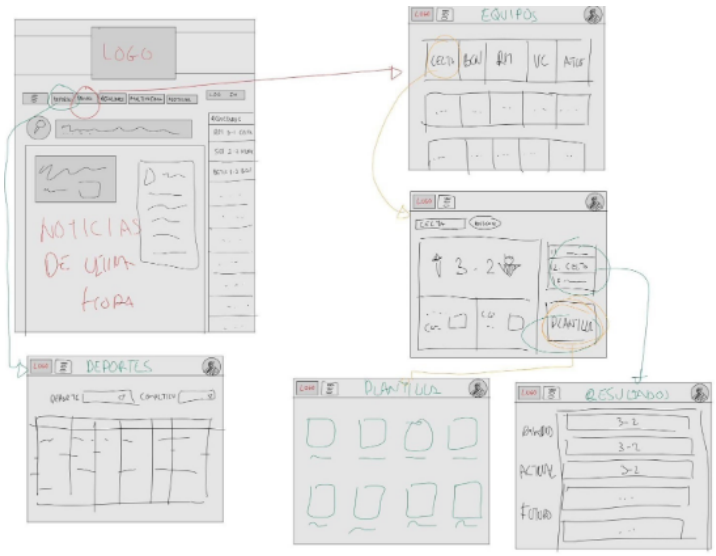


Imagen 2.4

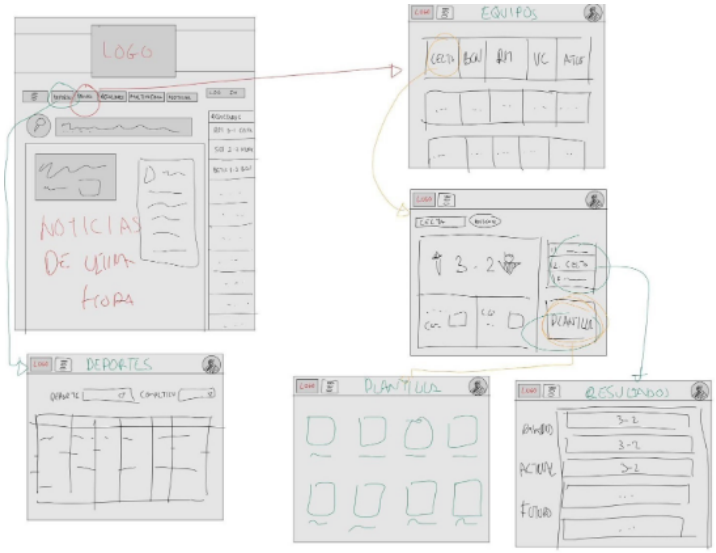


Imagen 2.5

2.5. Esqueleto (wireframe):

En el esqueleto se muestra como vamos a distribuír los elementos en las pestañas. Para ello, se ha utilizado un diseño basado en 24 columnas, para mantener las proporciones que usaremos para la página web final.

2.5.1. Página principal:

En la página principal se mostrarán las noticias de última hora destacadas (rectángulo de la izquierda), y a su vez los últimos resultados de las competiciones más buscadas (cuadrado de la derecha).

Imagen 2.6

2.5.2. Deportes:

En la pestaña de deportes se da a escoger de un desplegable el deporte y competición del cuál se quiere ver información. Esta información aparecerá en el recuadro grande.

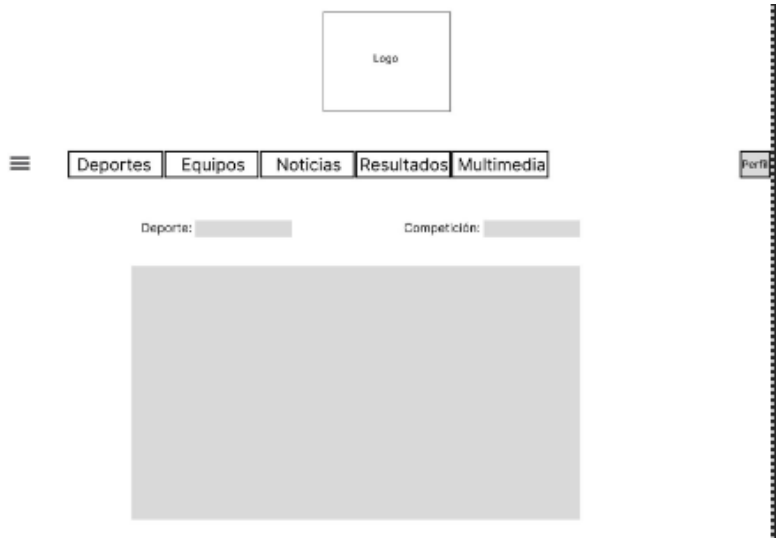


Imagen 2.7

2.5.3. Equipos:

En la pestaña de equipos se mostrarán, de varios deportes (en concreto los favoritos del usuario), los equipos de las competiciones correspondientes al deporte. El usuario puede seleccionar uno de los equipos para poder ver información detallada de cada uno, así como la plantilla, los últimos resultados y los próximos partidos.

Imagen 2.8

2.5.4. About us:

En esta ventana se ofrecerá información de nuestra página, así como un correo de contacto.

Imagen 2.9

2.5.5. Multimedia:

En multimedia, se muestran las imágenes destacadas del momento y los resúmenes de los últimos eventos, así como partidos o galas de premios.

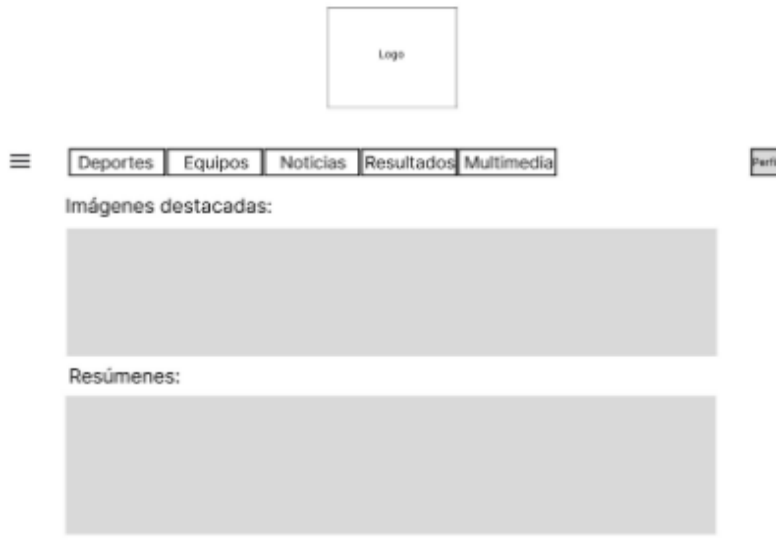


Imagen 2.10

2.6. Diseño final (Mockup):

Una vez hecho el prototipo de la página web, se diseña la apariencia final de la página. Para ello hemos utilizado la fuente de texto “Inter”, proporcionada por la aplicación “Figma”.

2.6.1. Página Principal:

Imagen 2.11

2.6.2. Deportes:

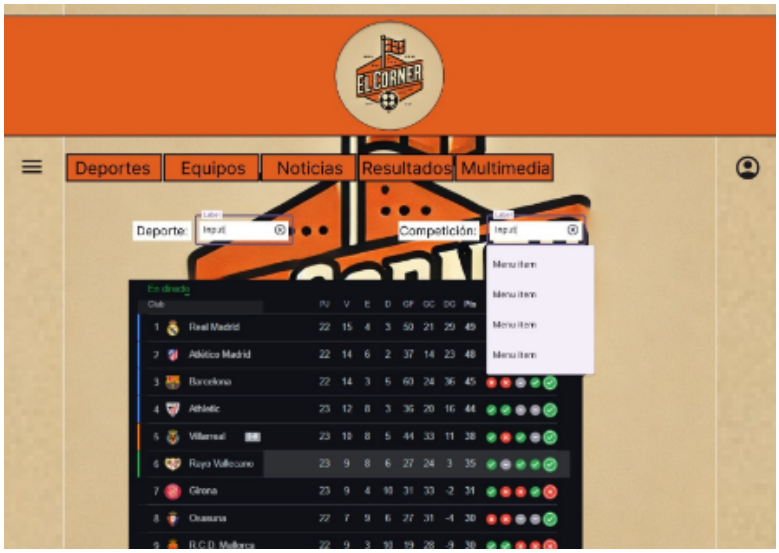


Imagen 2.12

2.6.3. Equipos:

Imagen 2.13

2.6.4. Equipo:

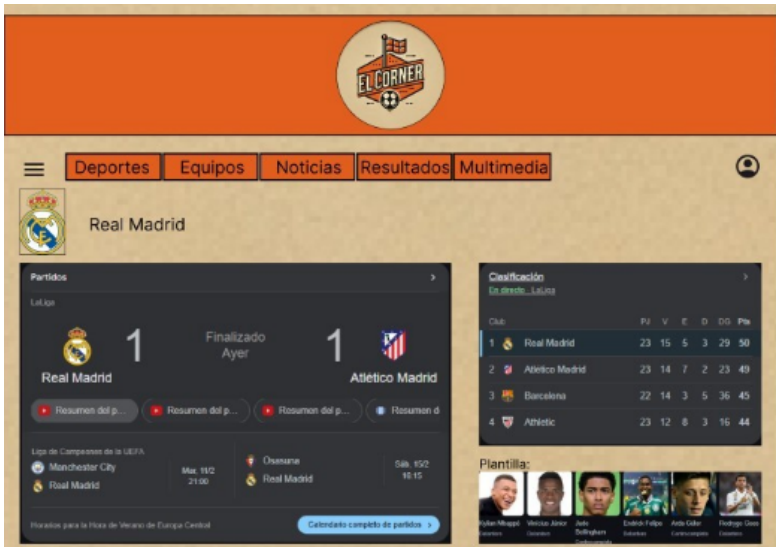


Imagen 2.14

2.7. Storyboard:

Ya con el diseño final de la pagina web terminado procedemos a realizar un storyboard que nos permite reflejar casos de usos reales dentro de la web.

En este caso hemos realizado el storyboard de cómo haremos para ver la plantilla del Real Madrid en La Liga de fútbol española (Imagen 1)

Imagen 2.15

2.8. Estructura de ficheros:

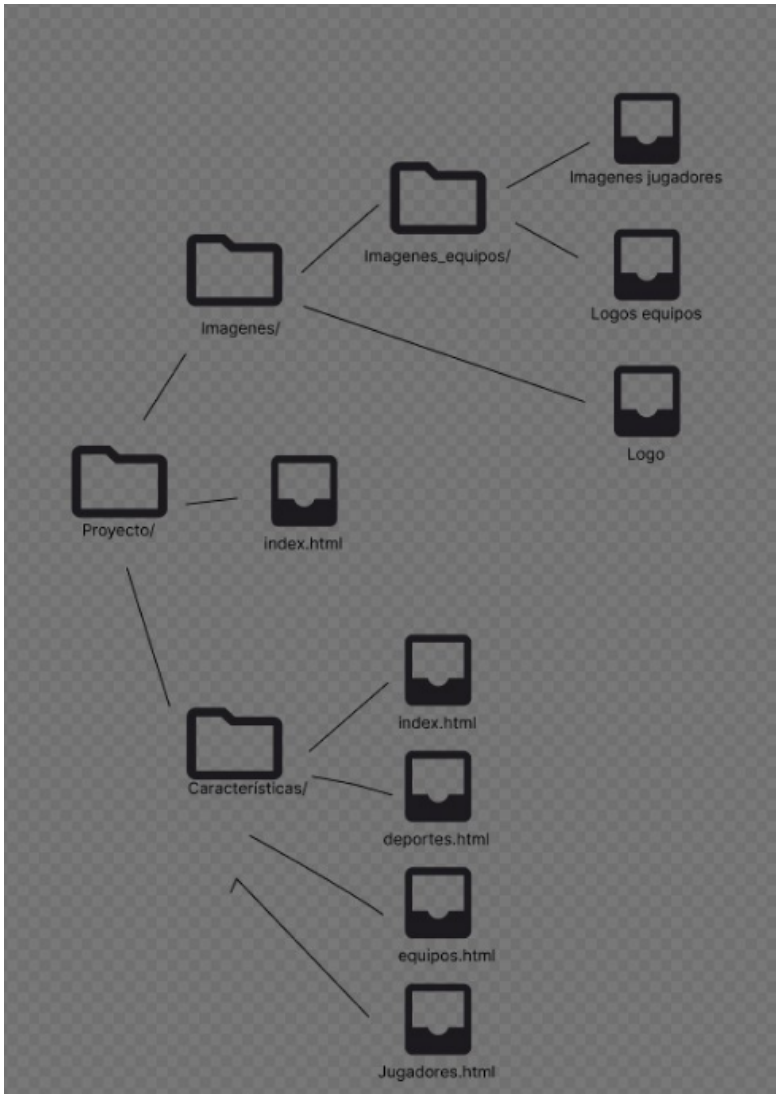


Imagen 2.16

3 Realización del proyecto:

3.1. HTML:

A continuación describiremos algunas páginas principales en cuanto a su código html y las relacionaremos con el elemento al que se corresponde y su situación en la ventana.

Para empezar analizaremos la ventana index o página principal:

```
        <!-- ------------------------------------------------    NAV    ----------
------------------------------------------------- -->
        <nav class="menuIndex">
            <input type="checkbox" id="nav-toggle">
            <label for="nav-toggle" class="opciones" >
                <img src="Imagenes/elementos_pagina_web/barras.png" width="40"
height="40" alt="Menu" onclick="fijarMenu()">
            </label>

            <div class="logo">
                <a href="index.html">
                    <img src="Imagenes/elementos_pagina_web/logo.png" width="80"
height="80" alt="ElCorner_Logo">
                </a>
                El Córner
            </div>
```

Imagen 3.1

```
            <ul class="list">
                <li><a href="Contenido/Deportes.html">Deportes</a></li>
                <li><a href="Contenido/Equipos.html">Equipos</a></li>
                <li><a href="Contenido/Resultados.html">Resultados</a></li>
                <li><a href="Contenido/Noticias.html">Noticias</a></li>
                <li><a href="Contenido/Multimedia.html">Multimedia</a></li>
            </ul>

            <div class="perfil">
                <a href="Contenido/login.html">
                    <img src="Imagenes/elementos_pagina_web/avatar.png" width="40"
height="40" alt="Imagen_Perfil">
                </a>
            </div>
        </nav>

        <!-- ----------------------------------------------------    MAIN (SECTION+)
 ------------------------------------------------------------- -->
        <div class="grid-container">

            <!-- ----------------------------------------------------    SECTION
(NOTICIAS)    ------------------------------------------------------------- -->
            <section id="breaking-news">
                <h2>BREAKING NEWS</h2>
                <article class="featured-news">
                    <header class="news-content">
                        <figure class="fuente-grande">
                            <img src="Imagenes/breaking_news/marcas/marca.png"
width="24" height="24" alt="Imagen_OkDiario">
                            <figcaption>Marca</figcaption>
                        </figure>
                        <div class="image-container">
                            <figure class="equipo-noticia-grande">
                                <img src="Imagenes/breaking_news/logos/madrid.png"
width="40" height="40" alt="Real Madrid">
                            </figure>
                            <figure class="noticia-grande">
                                <img src="Imagenes/breaking_news/madrid2.jpg"
alt="Real Madrid">
                            </figure>
                        </div>
                        <a href="https://okdiario.com/diariomadridista/real-
madrid/izquierda-radical-rabia-contra-dani-carvajal-dar-pregon-fiestas-san-isidro-
522726" class="news-title">
                            Fran González, el quinto portero más joven en debutar
con el Madrid, contra la 'maldición' del estreno
```

Noticias

Imagen 3.2

```
                    </a>
                    <time class="time">Hace 6 horas</time>
                </header>
                <div class="news-body">
                    Con 19 años, 9 meses y 10 días es el más joven en debutar
desde Iker Casillas.
                </div>
            </article>


            <div class="regular-news-container">
                <!-- Noticia 1 Fila 2 (Pequeña) -->
                <article class="regular-news">
                    <header class="news-content">
                        <figure class="fuente">
                            <img src="Imagenes/breaking_news/marcas/marca.png"
width="20" height="20" alt="Imagen_Marca">
                            <figcaption>Marca.com</figcaption>
                        </figure>
                        <div class="image-container">
                            <figure class="equipo-noticia">
                                <img
src="Imagenes/breaking_news/logos/juventus.png" alt="Juventus">
                            </figure>
                            <figure class="noticia">
                                <img src="Imagenes/breaking_news/juventus1.jpg"
alt="Juventus">
                            </figure>
                        </div>
                        <a href="https://www.marca.com/futbol/liga-
italiana/2025/03/31/juventus-pierde-gatti-fractura-compuesta-perone.html"
class="news-title">
                            La Juventus pierde a Gatti por una fractura
compuesta de peroné
                        </a>
                        <time class="time">Hace 13 horas</time>
                    </header>
                    <div class="news-body">
                        El defensa, titularísimo, se lesionó ante el Genoa y
estará fuera un mes
                    </div>
                </article>
            </div>
        </section>
```

Noticias

Imagen 3.3

```
<!-- ------------------------------------------------    ASIDE (RESULTADOS)    -
------------------------------------------------------- -->
        <aside id="resultados">
            <h2>RESULTADOS</h2>
            <h3>JORNADA 4</h3>
            <table>
                <thead>
                    <tr>
                        <th>Equipo Local</th>
                        <th>Resultado</th>
                        <th>Equipo Visitante</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>Cádiz CF</td>
                        <td>3 - 1</td>
                        <td>Villarreal CF</td>
                    </tr>
Noticias
                    <tr>
                        <td>UD Almería</td>
                        <td>2 - 3</td>
                        <td>RC Celta</td>
                    </tr>
                    <tr>
                        <td>Real Sociedad</td>
                        <td>5 - 3</td>
                        <td>Granada CF</td>
                    </tr>
                </tbody>
            </table>
        </aside>
    </div>

</body>

<!-- ------------------------------------------------    FOOTER    ----------
------------------------------------------------- -->
<footer>
    <span class="copyright">&copy; 2025 El Corner - Todos los derechos
reservados</span>
    <span class="about-link"><a href="Contenido/AboutUs.html">About Us</a>
</span>
</footer>

</html>
```

Imagen 3.4

Diagrama de Index:

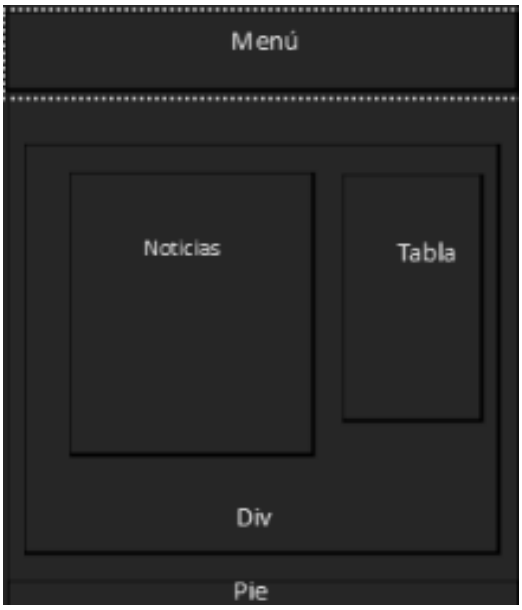


Imagen 3.5

Ahora resultados:

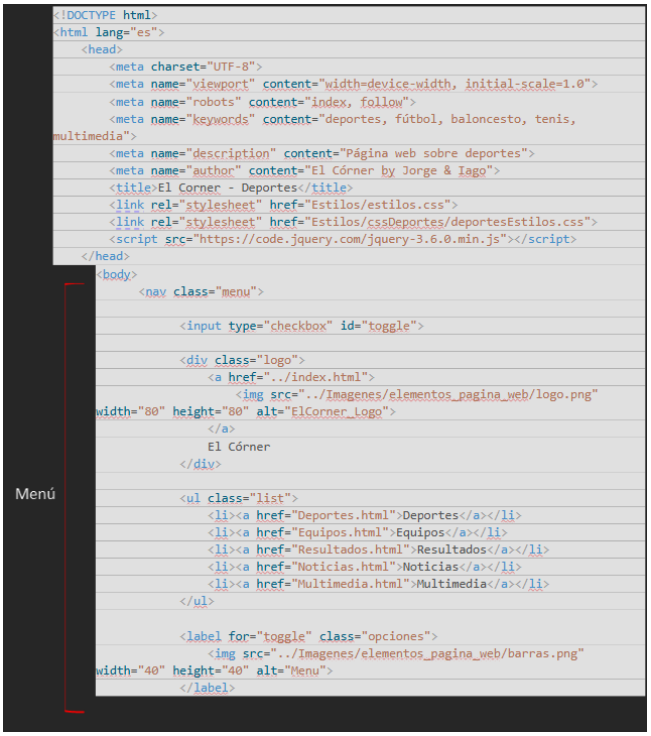


Imagen 3.6

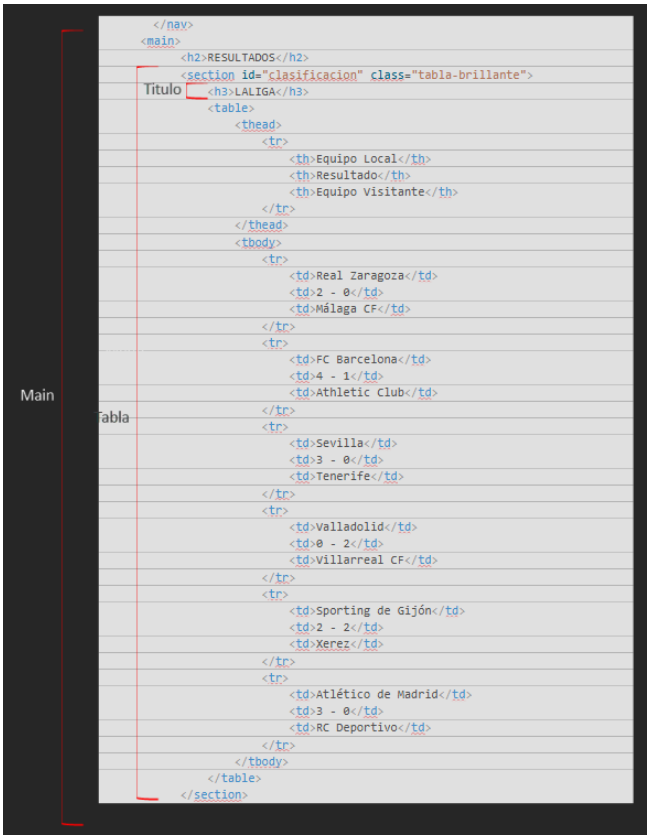


Imagen 3.7

Imagen 3.8

Diagrama de Resultados:

Imagen 3.9

Ahora la página de Deportes:

```
<!-- NAV -->
<nav class="menu">

    <input type="checkbox" id="toggle">

    <div class="logo">
        <a href="../index.html">
            <img src="../Imagenes/elementos_pagina_web/logo.png" width="80" height="80" alt="ElCorner_Logo">
        </a>
        El Córner
    </div>

    <ul class="list">
        <li><a href="Deportes.html">Deportes</a></li>
        <li><a href="Equipos.html">Equipos</a></li>
        <li><a href="Resultados.html">Resultados</a></li>
        <li><a href="Noticias.html">Noticias</a></li>
        <li><a href="Multimedia.html">Multimedia</a></li>
    </ul>

    <label for="toggle" class="opciones">
```

Imagen 3.10

```
            <img src="../Imagenes/elementos_pagina_web/barras.png" width="40"
height="40" alt="Menu">
        </label>
    </nav>

    <main>
        <section id="filters">
            <div class="filter-group">
                <label for="deporte">Deporte:</label>
                <select id="deporte" name="deporte">
                    <option value="futbol">Fútbol</option>
                    <option value="baloncesto">Baloncesto</option>
                    <option value="balonmano">Balonmano</option>
                    <option value="tenis">Tenis</option>
                </select>
            </div>
            <div class="filter-group">
                <label for="competicion">Competición:</label>
                <select id="competicion" name="competicion">
                    <option value="laliga">LaLiga</option>
                    <option value="premier">Premier League</option>
                    <option value="seriea">Serie A</option>
                </select>
            </div>
        </section>

        <section id="clasificacion">
            <h2>En directo</h2>
            <table>
                <thead>
                    <tr>

                    </tr>
                </thead>
                <tbody>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
```

Filtro

Tabla

Imagen 3.11

```
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                    <tr>

                    </tr>
                </tbody>
            </table>
        </sect>

    </main>

    <!-- ------------------------------------------------    FOOTER    -------
------------------------------------------------- -->
    <footer>
        <span class="copyright">&copy; 2025 El Corner - Todos los derechos
reservados</span>
        <span class="about-link"><a href="aboutus.html">About Us</a></span>
    </footer>

</body>
</html>
```

Tabla

Imagen 3.12

Diagrama de Resultados:

Imagen 3.13

Por último, la ventana de multimedia:

```
    <body>
        <nav class="menu">
            <input type="checkbox" id="toggle">
            <div class="logo">
                <a href="../index.html">
                    <img src="../Imagenes/elementos_pagina_web/logo.png" width="80"
height="80" alt="ElCorner_Logo">
                </a>
                El Córner
            </div>

            <ul class="list">
                <li><a href="Deportes.html">Deportes</a></li>
                <li><a href="Equipos.html">Equipos</a></li>
                <li><a href="Resultados.html">Resultados</a></li>
                <li><a href="Noticias.html">Noticias</a></li>
                <li><a href="Multimedia.html">Multimedia</a></li>
            </ul>

            <label for="toggle" class="opciones">
                <img src="../Imagenes/elementos_pagina_web/barras.png" width="40"
height="40" alt="Menu">
```

Imagen 3.14

Imagen 3.15

Diagrama de Resultados:

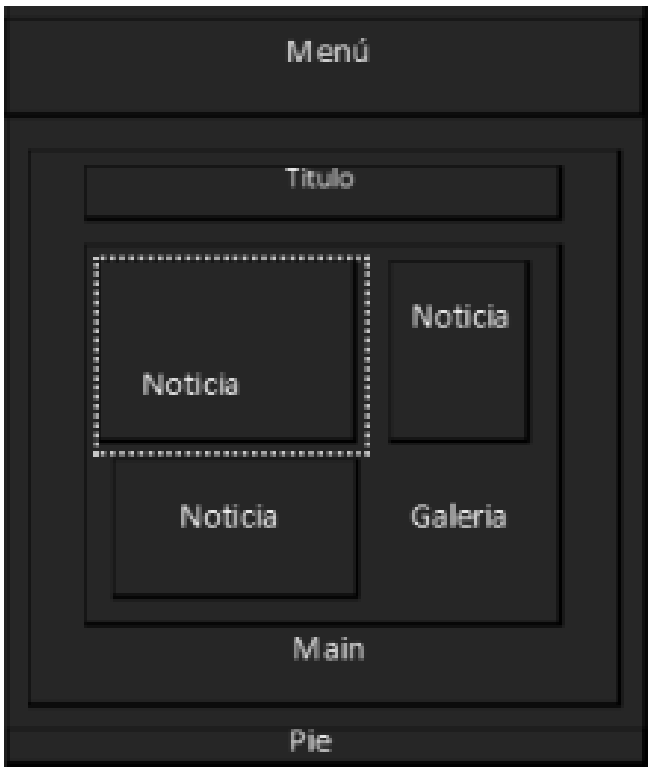


Imagen 3.16

3.1.1. Estructura de ficheros final

Aunque no dista muucho de la idea origianl, si que surgieron necesidades que nos obligaron a crear mas carpetas y expandir la jerarquía hacia niveles mas profundos y algo mas organizados.

Ahora nuestra estrucutra de ficheros cuenta con carpetas dentro de contenido para ordenar los archivos

según sean .css, .js o .html. También cuenta con nuevas carpetas para organizar las imaganes usadas.

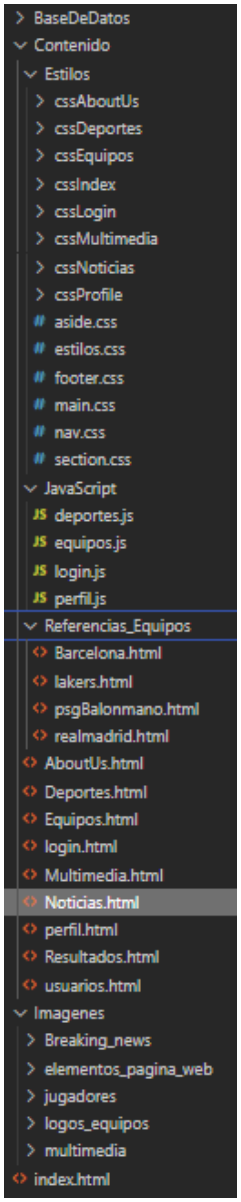


Imagen 3.17

→

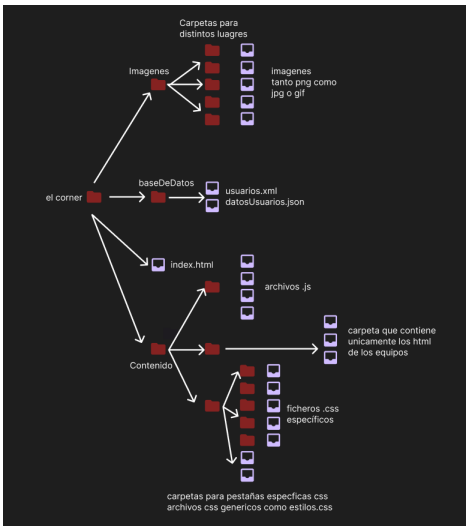


Imagen 3.18

3.2. Formateo mediante CSS:

Para dotar de estilo a la página se usa CSS, en concreto, se usa un archivo `.css` que se corresponde con el estilo general de las ventanas que conforman la página web, denominado `estilos.css`:

En él se incluyen los estilos de las etiquetas que son iguales para todas las pestañas de la página web. Por ejemplo, para el `aside` se definen aspectos como su posición, color del fondo, tamaño o el grosor y color del borde:

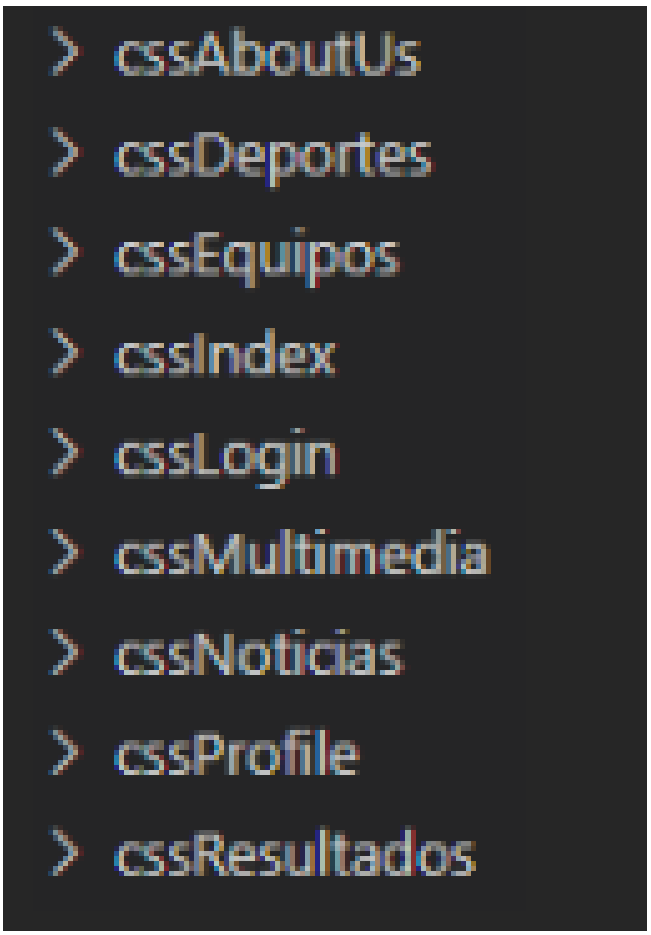
```
aside{
    float: right;
    flex: 40%;
    background-color: transparent;
    padding: 30px;
    box-sizing: border-box;
    border: solid 1px var(--gray-color);
    box-shadow: 2px 2px 5px var(--gray-color);
}
```

También, en el `footer`, se define que esté fijo en la parte inferior de la página, con fondo naranja, borde superior gris y texto centrado. El enlace `.ªbout"` dentro del `footer` se alinea específicamente a la derecha:

```
footer{
    position: fixed;
    bottom: 0;
    left: 0;
    width: 100%;
    background-color: #f6894c ;
    border-top: 1px solid #ccc;
    padding: 5px 0;
    text-align: center;
}

footer .about-link {
    text-align: right;
}
```

Después, para las etiquetas que no son iguales para todas las ventanas, se definen páginas de estilo concretas para cada una con el formato mostrado en la siguiente imagen, en la que dentro de cada una de las carpetas hay uno o más archivos `.css` que definen el estilo en esa página en concreto.



3.2.1. Creación de páginas responsivas:

En esta sección se detallan los trabajos realizados durante la segunda semana del desarrollo del sitio web "El Córner", enfocados en la adaptación de las páginas para distintos tamaños de pantalla y dispositivos. La responsividad es un aspecto fundamental del diseño web moderno, ya que permite garantizar una experiencia de usuario óptima tanto en ordenadores como en tablets y teléfonos móviles.

Para cumplir con este objetivo, se seleccionaron cinco páginas distintas del sitio, aplicando en cada una de ellas una técnica específica de diseño responsivo de las vistas en clase. De este modo, se asegura la integración práctica y variada de los diferentes enfoques disponibles, entre ellos:

- Uso de *CSS Multicol*.

```
.multicol {
    column-count: 2;
    column-gap: 40px;
    margin: 30px 0;
}


<p class="multicol">
    En El Corner, nos comprometemos a operar con transparencia y cumplir
     con todas las normativas legales aplicables. A continuación, te
     proporcionamos información detallada sobre nuestros términos legales,
     política de privacidad y política de cookies. Te recomendamos que leas
      atentamente esta sección para comprender cómo gestionamos tus datos y
      garantizar una experiencia segura y confiable en nuestra plataforma.
    En El Corner, la privacidad de nuestros usuarios es una prioridad. Esta
     política de privacidad explica cómo recopilamos, utilizamos y protegemos
     la información personal que nos proporcionas al visitar nuestro sitio web,
      registrarte en nuestra plataforma o interactuar con nuestros servicios.
    Recopilamos información personal, como tu nombre, dirección de correo
    electrónico y datos de navegación, únicamente cuando nos la proporcionas
     voluntariamente, por ejemplo, al suscribirte a nuestro boletín,
     participar en encuestas o dejar comentarios en nuestras publicaciones.
    Utilizamos tus datos para mejorar tu experiencia en nuestro sitio, personalizar
     el contenido que recibes, enviar actualizaciones y noticias relevantes, y
     responder a tus consultas o solicitudes. Nunca compartiremos tu información
     con terceros sin tu consentimiento explícito, excepto cuando sea necesario
     para cumplir con obligaciones legales.
    Implementamos medidas de seguridad técnicas y organizativas para proteger
    tus datos personales contra accesos no autorizados, pérdidas o alteraciones.
     Sin embargo, recuerda que ninguna transmisión de datos por Internet es
     completamente segura, por lo que te recomendamos actuar con precaución.
    Tienes derecho a acceder, rectificar, eliminar o limitar el uso de tus datos
     personales en cualquier momento. Si deseas ejercer estos derechos o tienes
      alguna pregunta sobre nuestra política de privacidad, puedes contactarnos
       a través de <a href="mailto:contacto@elcorner.es">nuestro correo
        electrónico</a> o mediante nuestro formulario de contacto.
</p>
```

Imagen 3.19

→

Imagen 3.20

- Uso de contenedores *Flex Container*.

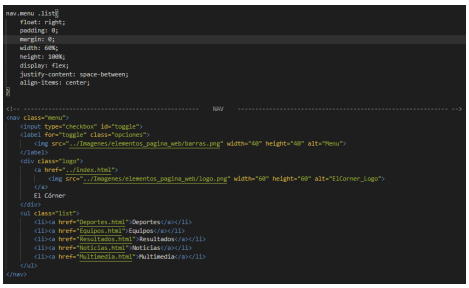


Imagen 3.21

→

Imagen 3.22

- Uso de *CSS Grid*.

```css
.grid-container {
    display: grid;
    grid-template-columns: 2fr 1fr;
    gap: 20px;
    background-color: rgba(255, 255, 255, 1);
    border: 1px solid rgb(0, 0, 0);
    padding: 20px;
    margin: 80px 20px;
    box-shadow: var(--shadow);
    border-radius: 10px;
}

.featured-news {
    grid-column: 1;
}

.regular-news-container {
    display: grid;
    grid-template-columns: 1fr 1fr;
    gap: 20px;
    grid-column: 1;
}

.regular-news {
    grid-column: span 1;
}

#resultados {
    grid-column: 2;
}
```

Imagen 3.23

→

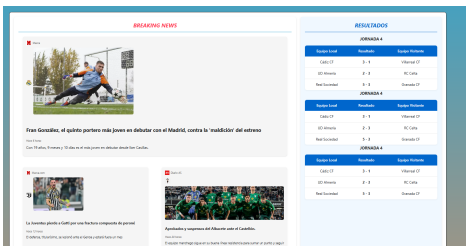


Imagen 3.24

- Uso de la librería *Bootstrap*.

```
<main>
    <div class="container mt-5">
        <!-- Sección de Fútbol -->
        <div class="row">
            <h1>Fútbol:</h1>
            <p>(Pulsa en el equipo al que quieras acceder)</p>

            <div class="col-md-4">
                <h3>La Liga:</h3>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
            </div>

            <div class="col-md-4">
                <h3>Premier League:</h3>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
            </div>

            <div class="col-md-4">
                <h3>Bundesliga:</h3>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
                <figure class="imagenEquipo">···
                </figure>
            </div>
        </div>

        <!-- Sección de Baloncesto -->
        <div class="row">
            <h1>Baloncesto:</h1>
            <p>(Pulsa en el equipo al que quieras acceder)</p>

            <div class="col-md-4">···
            </div>

            <div class="col-md-4">···
            </div>

            <div class="col-md-4">···
            </div>
        </div>
```

Imagen 3.25

→

Imagen 3.26: Botón después de pasar el mouse por encima

3.3. Dinamismo mediante JavaScript:

En esta sección se presentan los diferentes elementos dinámicos implementados en nuestra página web “El Córner” mediante el uso de JavaScript. Siguiendo los requisitos establecidos, hemos estructurado el desarrollo en tres etapas principales, cada una con características específicas:

3.3.1. JavaScript

Primeramente, hemos incorporado elementos dinámicos utilizando JavaScript. Hemos desarrollado varios efectos interactivos que mejoran la experiencia de usuario, haciendo uso de diferentes métodos de acceso al DOM para distintos tipos de eventos.

Efecto Blur/Focus en campos de texto

En este ejemplo se realiza un elemento visible en el que cuando se tiene el foco (el usuario lo selecciona) en el input de texto para introducir el nombre desaparece el contenido de placeholder (el texto predeterminado que hay en él). A su vez, cuando se hace clic en otra parte, es decir se pierde el foco, vuelve el texto. Para ello se usan los métodos de acceso al DOM querySelector y getElementsByName, y se atiende a los eventos de focus y blur.

```
<script>
    const usernameInput = document.querySelector('#username');

    usernameInput.addEventListener('focus', function() {
        this.placeholder = '';
    });

    usernameInput.addEventListener('blur', function() {
        if (!this.value) {
            this.placeholder = 'Usuario...';
        }
    });

    const passwordInput = document.getElementsByName('password')[0];

    passwordInput.addEventListener('focus', function() {
        this.placeholder = '';
    });

    passwordInput.addEventListener('blur', function() {
        if (!this.value) {
            this.placeholder = 'Contraseña...';
        }
    });
</script>
```

Imagen 3.27: Código JavaScript para gestionar los eventos focus y blur

El resultado es el siguiente:

Imagen 3.28: Estado inicial del input

Al hacer click

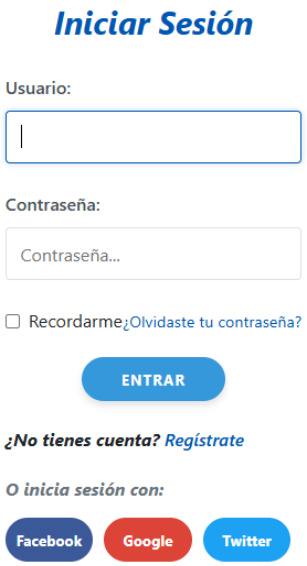


Imagen 3.29: Input después de recibir el foco

Efecto Hover en botones

Otro efecto visible es el siguiente:

```
const botonEntrar = document.querySelectorAll('.login-button')[0];

function resaltar() {
    this.classList.add('.login-button:hover');
}

function quitarResaltado() {
    this.classList.remove('.login-button');
}

loginButton.addEventListener('mouseover', resaltar);
loginButton.addEventListener('mouseout', quitarResaltado);
```

Imagen 3.30: Código JavaScript para gestionar los eventos mouseover y mouseout

Se aplica en el botón de “Entrar” en el apartado de login. Cuando se pone el mouse por encima se hace el

botón más grande, y cuando se quita vuelve a su tamaño normal. Para ello usamos el método de acceso al DOM getElementsByClassname y los eventos de mouseover y mouseout.

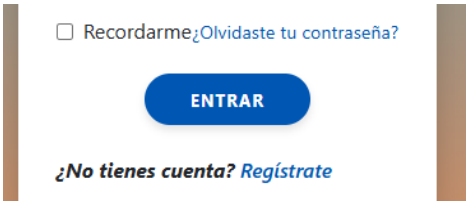


Imagen 3.31: Estado inicial del botón

→

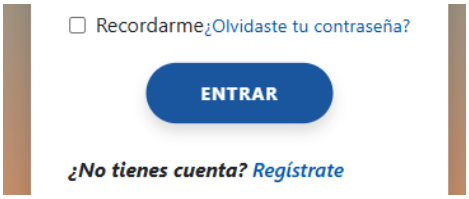


Imagen 3.32: Botón después de pasar el mouse por encima

Brillo mediante "onmouseover"

El siguiente elemento visible se corresponde con la iluminación (aumento de brillo) de las imágenes correspondientes con los equipos en la página Equipos.html. Para ello, guardamos en una constante todas las etiquetas img del código, y luego buscamos las que tienen como abuelo (la etiqueta padre de la etiqueta padre) una etiqueta figura con clase="imagenEquipo". En esas últimas, hacemos que se aumente el brillo de la imagen si se pasa el ratón por encima y se quita ese aumento si se quita el ratón de encima. Se usa el método de acceso al DOM getElementsByTagName y getAttribute.

```
<script>
    const imagenes = document.getElementsByTagName('img');

        for (let i = 0; i < imagenes.length; i++) {
        const abuelo = imagenes[i].parentNode.parentNode;

        if (abuelo.tagName.toLowerCase() === 'figure' &&
            abuelo.getAttribute('class') === 'imagenEquipo') {

            imagenes[i].onmouseover = function() {
                this.style.filter = "brightness(120%)";
            };

            imagenes[i].onmouseout = function() {
                this.style.filter = "brightness(100%)";
            };
        }
        }

</script>
```

Imagen 3.33: Código JavaScript para gestionar el brillo usando onmouseover

El resultado de este efecto es el siguiente:

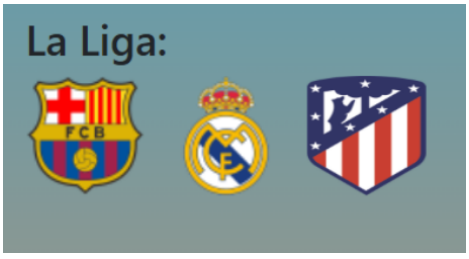


Imagen 3.34: Estado inicial del botón

→

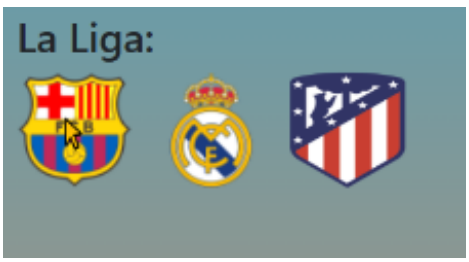


Imagen 3.35: Botón después de pasar el mouse por encima

3.3.2. jQuery y ES6

En la segunda fase, hemos mejorado la funcionalidad de la página implementando efectos visuales utilizando la librería jQuery.

JQuery

Para este elemento visible usamos jQuery, haciendo que en la ventana de resultados se recubra la tabla de un borde de 3 px cuando se pase el ratón por encima. Para ello se añade y elimina una clase CSS (borde-tabla) cuando el usuario pasa el ratón por encima de un elemento con la clase ".tabla-brillante", que se corresponde con las tablas que se muestran en la ventana.

```
<script>
    $(document).ready(function() {
        $(".tabla-brillante").hover(
            function() {
                $(this).addClass("borde-tabla");
            },
            function() {
                $(this).removeClass("borde-tabla");
            }
        );
    });
</script>
```

```
.borde-tabla {
    border: 3px solid #000000;
}
```

El resultado es el siguiente:

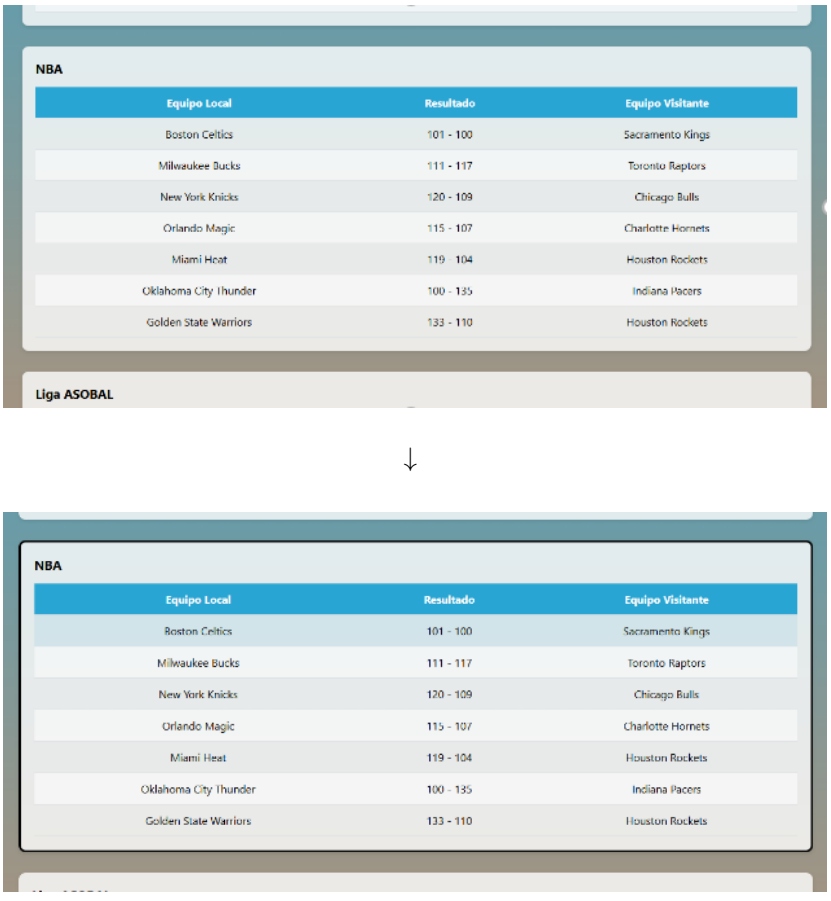


Imagen 3.36: Botón después de pasar el mouse por encima

JES6

Este elemento visible, usando JavaScript (ES6), corresponde al menú que aparece cuando se reduce el tamaño de la ventana y se hace clic en el icono de las tres barras. El menú que aparece da acceso a las distintas secciones de la página. Para implementar esta funcionalidad, se ha creado una función que alterna el valor de la propiedad position en el estilo del menú entre fixed y sticky. Esto permite que el menú se fije en la parte superior de la pantalla al hacer clic en el icono, y que vuelva a su posición original al hacer clic nuevamente.

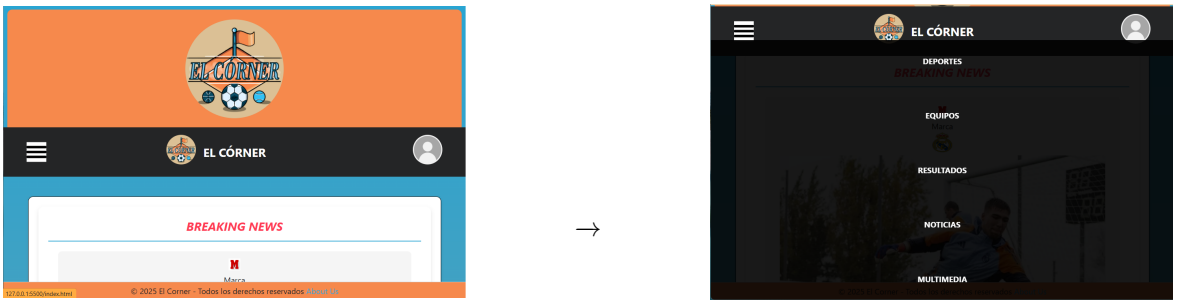
Para acceder al menú y modificar su estilo, se utiliza el método del DOM getElementsByClassName. Además, se incluye un desplazamiento suave (smooth).

```
<input type="checkbox" id="nav-toggle">
<label for="nav-toggle" class="opciones" >
    <img src="Imagenes/elementos_pagina_web/barras.png" width="40" height="40" alt="Menu" onclick="fijarMenu()">
</label>
```

```
<script>
    function fijarMenu() {
        var menu = document.getElementsByClassName("menuIndex")[0];

        if (menu.style.position === "fixed") {
            menu.style.position = "sticky";
            window.scrollTo({
                top: 0,
                behavior: "smooth"
            });
        }
        else {
            menu.style.position = "fixed";
            window.scrollTo({
                top: 210,
                behavior: "smooth"
            });
        }
    }
</script>
```

El resultado es el siguiente (ventana con tamaño disminuido para que aparezcan las tres barras):



3.3.3. Carga de contenido

En esta parte hemos realizado la lectura de datos externos en formatos XML y JSON. Para ello, hemos implementado diferentes métodos de carga:

- Uso del objeto XMLHttpRequest para carga asíncrona tradicional
- Utilización de la API Fetch

Todos estos desarrollos se han probado en un servidor local para comprobar funcionamiento en un entorno real ya que de no ser así el navegador impide el acceso a archivos locales.

XML

En este caso usamos XML con XMLHttpRequest como forma de cargar los datos, y se cargan datos XML y luego valida credenciales comparando el contenido del XML.

Primero, verifica que ambos campos (usuario y contraseña) estén completos, si falta alguno muestra un error. Luego, realiza una petición a un archivo XML con los datos de usuarios. Si consigue conectar, pasa los datos a

la función “validateUser” para comprobar las credenciales.

La función validateUser verifica las credenciales del usuario comparándolas con los datos almacenados en el archivo XML. Primero, recorre cada nodo en el XML, comparando los valores de nombre de usuario y contraseña. Si coinciden, almacena la información de sesión en el almacenamiento local, muestra un mensaje de confirmación y redirige a la página de perfil. En caso contrario, muestra un mensaje de error.

El código correspondiente es:

```
function handleLoginSubmit(event) {
  event.preventDefault(); // Prevenir envío del formulario por defecto

  const username = document.getElementById('username').value;
  const password = document.getElementById('password').value;

  if (username.trim() === '') {
    mostrarMensaje('Por favor, ingrese su nombre de usuario', false);
    return;
  }

  if (password.trim() === '') {
    mostrarMensaje('Por favor, ingrese su contraseña', false);
    return;
  }

  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState === 4) {
      if (this.status === 200) {
        const xmlDoc = this.responseXML;
        if (xmlDoc) {
          validateUser(username, password, xmlDoc);
        } else {
          mostrarMensaje('Error al procesar los datos de usuario.', false);
        }
      } else {
        mostrarMensaje('Error al conectar con la base de datos.', false);
      }
    }
  };

  xhttp.open("GET", "../../BaseDeDatos/usuarios.xml", true);
  xhttp.send();
}
```

```
function validateUser(username, password, xmlDoc) {
  const usuarios = xmlDoc.getElementsByTagName('usuario');
  let usuarioValido = false;

  for (let i = 0; i < usuarios.length; i++) {
    const user = usuarios[i];
    const usuarioNode = user.getElementsByTagName('nombre_usuario')[0];
    const contrasenaNode = user.getElementsByTagName('contrasena')[0];

    if (usuarioNode && contrasenaNode &&
        usuarioNode.textContent === username &&
        contrasenaNode.textContent === password) {
      usuarioValido = true;
      break;
    }
  }

  if (usuarioValido) {
    // Si el usuario es válido, guardar información en localStorage
    localStorage.setItem('sesionActiva', 'true');
    localStorage.setItem('usuario', username);

    // Mostrar modal de éxito
    mostrarMensaje('¡Has iniciado sesión correctamente!', true);

    // Redirigir a la página de perfil después de un breve delay
    setTimeout(() => {
      window.location.href = 'perfil.html';
    }, 1500);
  } else {
    // Si no coincide, mostrar modal de error
    mostrarMensaje('Las credenciales no son correctas', false);
  }
}
```

El código XML que lee es este:

```xml
<usuarios>
  <usuario>
    <nombre_usuario>jakub</nombre_usuario>
    <contrasena>123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>jorge</nombre_usuario>
    <contrasena>123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>arianeera</nombre_usuario>
    <contrasena>ari123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>pablo</nombre_usuario>
    <contrasena>pablo123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>miguel</nombre_usuario>
    <contrasena>miguel123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>javi</nombre_usuario>
    <contrasena>javi123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>paula</nombre_usuario>
    <contrasena>paula123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>iria</nombre_usuario>
    <contrasena>iria123</contrasena>
  </usuario>
  <usuario>
    <nombre_usuario>admin</nombre_usuario>
    <contrasena>admin</contrasena>
  </usuario>
</usuarios>
```

JSON

En cuanto a la carga de contenido en formato JSON (con jQuery) realizamos la siguiente funcionalidad:

Este fragmento de código carga y visualiza los datos del usuario al iniciar una página web. Primero, verifica si existe el usuario en almacenamiento local. Posteriormente, con una petición fetch, carga un archivo JSON que contiene los datos de usuarios. Busca coincidencias entre el usuario actual y los del JSON. Al encontrar una coincidencia, actualiza todos los campos del formulario (nombre, apellidos, email, etc.) con los datos correspondientes al usuario del JSON. En caso de no encontrar al usuario en la base de datos JSON, muestra un error. El código demuestra un flujo completo de autenticación y carga de datos personales, utilizando tanto el almacenamiento local del navegador como peticiones asíncronas para recuperar información de un servidor.

El código correspondiente es:

```javascript
$(document).ready(function() {
    const usuarioActivo = localStorage.getItem('usuario');

    if (!usuarioActivo) {
        console.log('No hay usuario activo en localStorage');
        return;
    }

    console.log('Usuario activo:', usuarioActivo);


    const nombreUsuarioGuardado = localStorage.getItem('usuario');
            if (nombreUsuarioGuardado) {
                document.getElementById('username-display').textContent = nombreUsuarioGuardado;
                document.getElementById('nombre').value = nombreUsuarioGuardado;
            }

            document.getElementById('username-display').textContent = "aaaaa";
            $('#nombre').val("aaaaa");

    fetch('../../BaseDeDatos/datosUsuarios.json')
        .then(response => {
            console.log('Respuesta del fetch:', response);

            if (!response.ok) {
                mostrarMensaje('Error al conectar con la base de datos.', false);
            }
            return response.json();
        })
        .then(jsonObj => {
            console.log('JSON cargado:', jsonObj);

            // Buscar el usuario activo en el JSON
            const usuarioValido = jsonObj.datosUsuarios.find(
                user => user.usuario === usuarioActivo
            );



            // Si encontramos el usuario, actualizamos los campos del formulario
            if (usuarioValido) {
                console.log('Usuario encontrado:', usuarioValido);

                // Actualizar el nombre de usuario en el encabezado
                document.getElementById('username-display').textContent = usuarioValido.usuario;

                // Actualizar campos del formulario de información personal
                document.getElementById('nombre').value = usuarioValido.nombre;
                document.getElementById('apellidos').value = usuarioValido.apellidos;
                document.getElementById('email').value = usuarioValido.email;
                document.getElementById('telefono').value = usuarioValido.telefono;
                document.getElementById('fecha-nacimiento').value = usuarioValido.fechaNacimiento;
                document.getElementById('ubicacion').value = usuarioValido.ubicacion;
                document.getElementById('bio').value = usuarioValido.acercaDeMi;

                // También podríamos actualizar la sección de equipos si es necesario
                // Esto dependería de cómo se estructura la información de equipos en el HTML

                console.log('Datos de usuario cargados correctamente');
            } else {
                console.error('Usuario no encontrado en la base de datos');
            }
        })

});
```

El código JSON que lee es este:


```json
{
  "datosUsuarios": [
    {
      "usuario": "jakub",
      "nombre": "Iago",
      "apellidos": "Feijoo Rey",
      "email": "iago@ejemplo.com",
      "telefono": "123456789",
      "fechaNacimiento": "1990-01-01",
      "ubicacion": "Calle de Iago, 123",
      "acercaDeMi": "Aficionado al deporte en general y seguidor del Celta desde peque\u00f1o. Me gusta comentar los partidos y participar en las quinielas.",
      "equipoFavorito": "Celta de Vigo",
      "equipos": "Real Madrid, Barcelona, Atletico de Madrid"
    },
    {
      "usuario": "jorge",
      "nombre": "Jorge",
      "apellidos": "gonzalez Corbelle",
      "email": "jorge@jemplo.com",
      "telefono": "987654321",
      "fechaNacimiento": "1992-05-12",
      "ubicacion": "Calle de Jorge, 456",
      "acercaDeMi": "Apasionado del f\u00fatbol y seguidor del Real Madrid. Me gusta analizar jugadas y debatir sobre el juego.",
      "equipoFavorito": "Real Madrid",
      "equipos": "Real Madrid, Sevilla, Valencia"
    },
    {
      "usuario": "pablo",
      "nombre": "Juan Pablo",
      "apellidos": "Martinez Martinez",
      "email": "pablo@jemplo.com",
      "telefono": "654321987",
      "fechaNacimiento": "1988-08-20",
      "ubicacion": "Calle de Pablo, 789",
      "acercaDeMi": "Fan\u00e1tico del Barcelona desde la infancia. Me encanta seguir las estad\u00edsticas y tendencias del equipo.",
      "equipoFavorito": "Barcelona",
      "equipos": "Barcelona, Villarreal, Betis"
    },
```