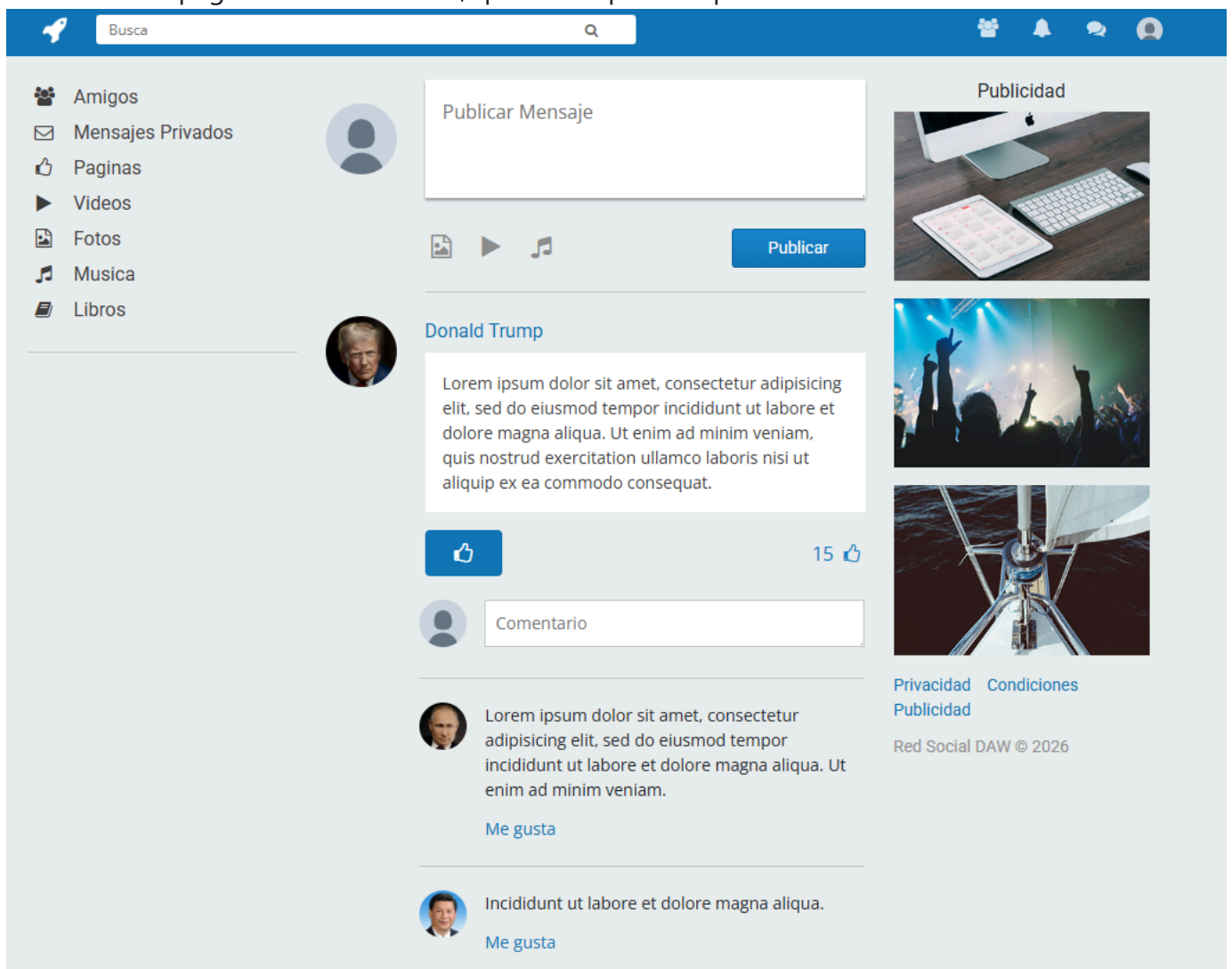
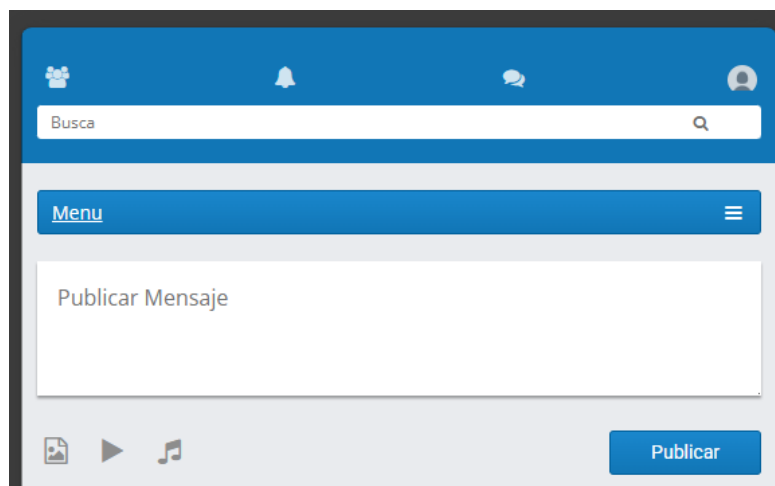


# Práctica Red Social

- Haremos una página estilo red social, que se adapte a dispositivos móviles.



- Cuando se hace small el menú de la izquierda desaparece y aparece un hamburger.



- La publicidad desaparece cuando el dispositivo se hace pequeño.

# Guía para la realización de la práctica:

- Recursos para la realización de la práctica:
  - Tendremos una carpeta con imágenes donde guardaremos tres imágenes de la publicidad (medirán 332x220)
  - Tendremos un logo (24x24)
  - Dos imágenes para las publicaciones (las que queráis, pero no muy grandes).
  - Tendremos fotos de paisajes.
  - Fotos de personas (200x200) para los avatares.
  - **Podéis usar las que os paso o yo, o cualquier otras fotos que encontréis**
- Estructura de la página:
  - Los iconos los usaremos de fontawesome (nos vamos a la página de fontello, seleccionamos los iconos que nos queremos bajar, pulsamos descargar y se nos generará un zip, extraemos las carpetas css/ y font/ y las incluimos en nuestro proyecto).
  - Fuentes: He usado la de roboto y open sans: Me voy a la página de google fonts, las selecciono, copio en enlace y lo añado en mi html.
  - Bootstrap añadiré los estilos de bootstrap usando CDN.
  - por último haré referencia a mi hoja de estilos (css/estilos.css)
  - Debe quedar algo así:

```
1 <head>
2   <meta charset="UTF-8">
3   <meta name="viewport" content="width=device-width, initial-scale=1.0">
4   <title>Red Social DAW</title>
5   <link
6     href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/bootstrap.min.css"
7     rel="stylesheet" integrity="sha384-..." crossorigin="anonymous">
8   <link href="https://fonts.googleapis.com/css?family=Open+Sans|Roboto"
9     rel="stylesheet">
10  <link rel="stylesheet" href="css/fontello.css">
11  <link rel="stylesheet" href="css/estilos.css">
12 </head>
```

## HEADER

- Se trata de implementar esta parte:

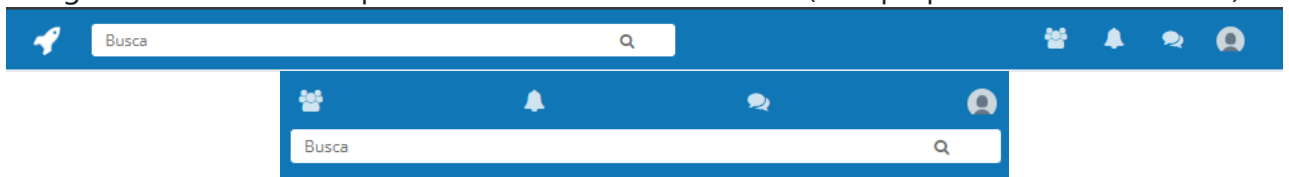


- Hay que fijarse que entre el input de búsqueda y los iconos hay un espacio.
- Cuando la pantalla se hace más pequeña el espacio desaparece.
- El header lo meteré dentro de un container, dentro de este container meteré un row y dentro de este tres `divs` con el valor col a cada uno (para que le dé el mismo espacio).
  - Una para el logo

- Otra para el input: Dentro del `div` del `input` pondré un `form` y dentro del `form` otro `row`. Dentro de este `row` anidado irá un `col` de tamaño 10 para el `input` y un `col-2` para el botón
- Otra para el menú, dentro pondré enlaces con los iconos y cargaré mi imagen (le doy un tamaño dentro del html de 22x22 para que encaje bien).
- Estilos del header:
  - Al header le pondremos un color de fondo azul.
  - Para el `input` le diremos que abarque el 100% (`width: 100%`), además podemos jugar con diferentes reglas como `width`, `background`, `border`, `font-size`, `padding`, `height`, `line-height`, `border-radius`, `color`, `font-family`.
  - Para quitar los los espaciados entre columnas que nos da bootstrap por defecto hay que usar la propiedad `g-0`
  - Para los iconos de la derecha podemos poner a los enlaces (iconos) un color blanco con opacidad). Cuando hagamos hover lo podemos poner blanco.



- Para colocar los elementos del header lo haremos con utilidades de bootstrap:
  - El logo se mostrará en dispositivos medianos en adelante (más pequeños no se mostrará).

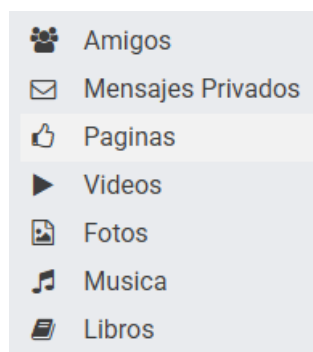


- En la columna de buscar tendremos que trabajar con los espaciados. En extra pequeño tendrá un tamaño de 12, cuando sea `sm` de 8, cuando sea `lg` en adelante 6
- En la columna de los iconos habrá que hacer algo parecido, lo que pasa que los iconos, al ser enlaces se nos van a poner a la izquierda, por tanto, aplicaremos flexbox para colocarlos
- Cuando la pantalla sea pequeña la barra tiene que cambiar de orden, y cuando sea mediano también tiene que cambiar.



## BARRA LATERAL IZQUIERDA

- Solo haremos la parte del html y el css, posteriormente haremos la parte del javascript que oculta el menú e inserta el hamburguer.

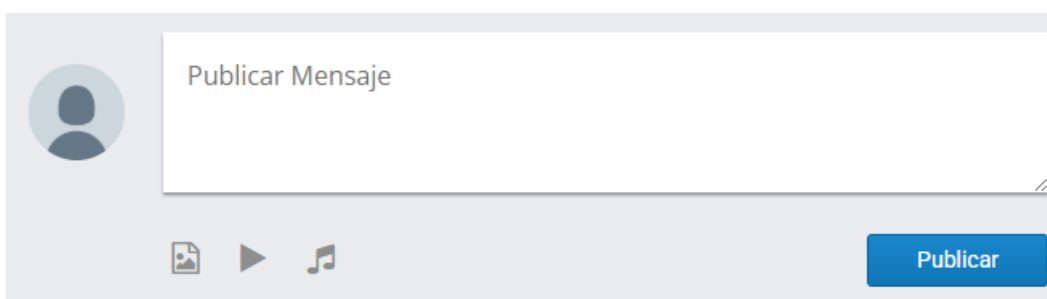


- Crearemos una row donde pondremos las tres columnas:
  - La de la izquierda.
  - El contenido principal (en contenido central).
  - La parte de la derecha (publicidad).
- Para cada elemento del menú usará enlaces añadiendo cada icono en la `class`.
- Ahora le tendremos que dar estilos: jugad con los paddings de cada elemento.
- Para que estén uno encima del otro y no seguidos usad `display: block;`
- Por último, pondremos que a partir de un tamaño medio en adelante ocupe 3, por debajo ocupará 12.

## CONTENIDO PRINCIPAL

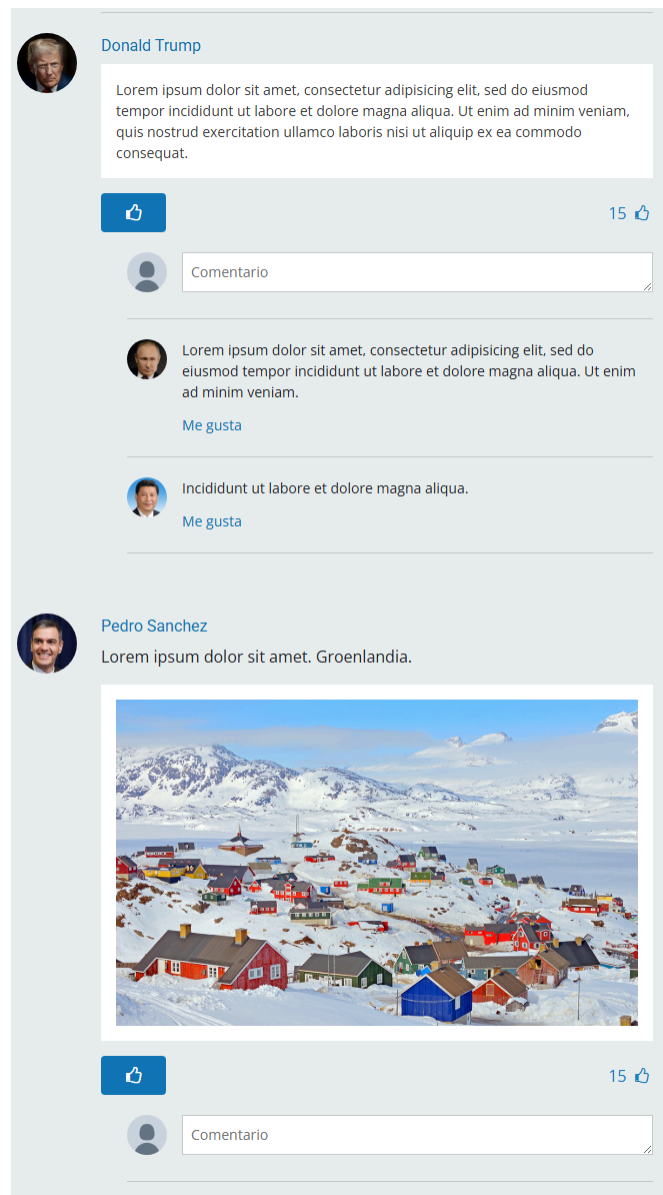
- Esta sección se compone de varias partes.

### Formulario de publicación:

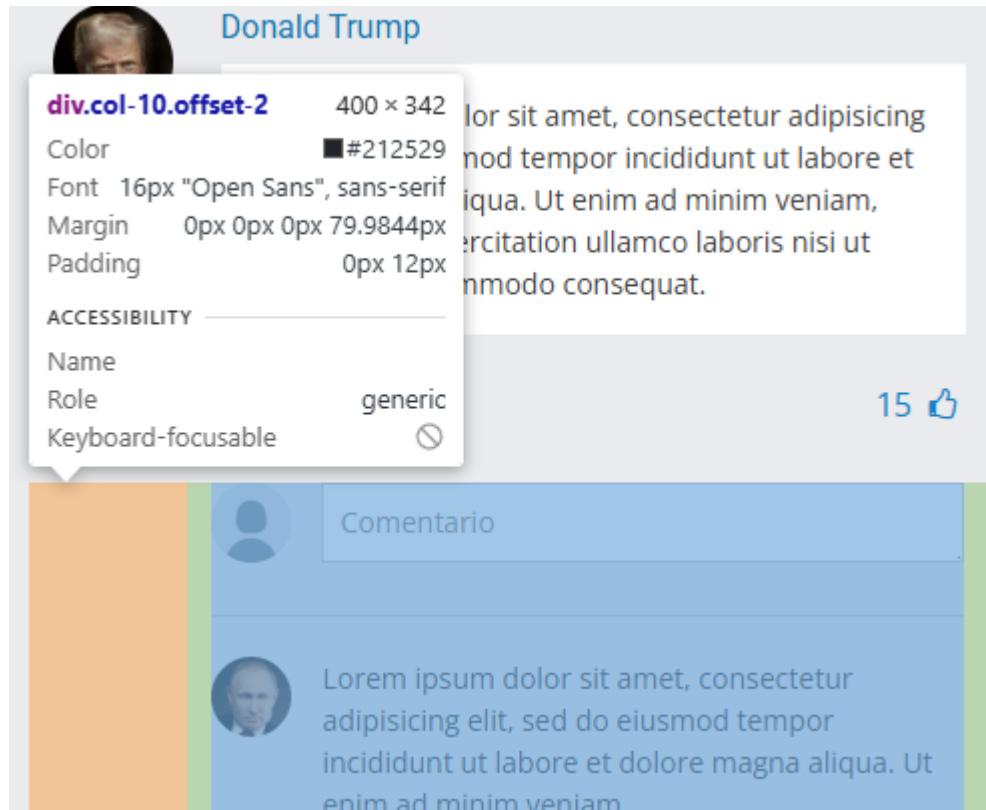


- Se compone de una fila y dos columnas. En la primera columna estará la foto (le pondremos `col-auto` para que ocupe justo lo que es la foto) y en la segunda columna estará el `text-area`, los iconos y el botón. Para que los iconos y el botón queden separados usar flexbox. Los iconos los podéis poner como enlaces (con la clase que le corresponda en fontello).
- La foto de perfil en el formulario de publicación le he puesto dentro de un enlace. En los estilos le doy el tamaño que quiero en concreto (60x60), recordad que para darle el tamaño que queramos en los estilos tenemos que poner `display: inline-block;`
- El botón es un botón normal al que le he puesto estilos propios (propiedades usadas: `border`, `padding`, `font-size`, `font-family`, `background`, `color`, `border-radius`, `cursor`)
- Para que haga el efecto de invertir colores he cambiado de posición el gradiente.

## Publicaciones:



- Para la publicación vamos utilizar una row y dentro dos columnas:
  - Una para la foto con col-auto. La foto la he metido en un enlace (seguir la misma estrategia que con la imagen de perfil).
  - Otra para el enlace (nombre) y el comentario (jugar con paddings, color, etc) y la caja de botones (parecido a lo que hicimos con el anterior layout de botones).
  - Para los comentarios que van dentro de cada publicación he creado una fila dentro de la fila ya creada



- Cada comentario es una fila (row), para decirle que las columnas no tengan espaciados le pondré `g-0`.
- El espaciado que hay a la izquierda de las columnas lo podemos conseguir dándole a la columna donde están los comentarios un tamaño de `col-10` y un `offset-2`
- Al colocar la imagen en el comentario de la imagen hay que ponerle en los estilos un `width` de 100% para que no se salga de su contenedor

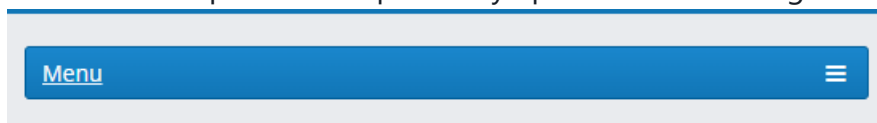
## BARRA LATERAL DERECHA

- Le tenemos que decir (con bootstrap) que en un dispositivo pequeño en adelante tenga un `display none`, y en uno largo o mayor debe ser block.



## ADAPTACIÓN A RESPONSIVE

- Se trata de que el menú de la izquierda desaparezca y aparezca el hamburguer:



- Para ello antes del contenido principal vamos a agregar una fila, que será donde pongamos el botón. El botón será un enlace. Le pondremos que no se muestre en un tamaño mediano en adelante.
- Para separar el texto "Menú" del icono hamburguer usaremos flexbox.
- Para el icono hamburger usar `icon-menu`.
- Ahora trabajaremos con la barra lateral izquierda (crear una clase en nuestro css). Meteremos esta clase en un media query ( `@media screen and (max-width: 767px)` )
- Cuando pulsamos el botón hamburger queremos que salga el menú y para ello le daremos el siguiente estilo:

```
1 main .fondo-enlace { /*Esta es la parte del fondo que sale tras el menú */
2   position: fixed;
3   width: 100%;
4   height: 100vh;
5   top: 0;
```

```

6     left: 0;
7     cursor: pointer;
8     background: rgba(175, 217, 236, 0.5);
9     z-index: 99;
10    display: none;
11  }
12
13  main .fondo-enlace.active {
14    display: block;
15  }
16
17  main .barra-lateral-izquierda {
18    width: 85%;
19    position: fixed;
20    z-index: 100;
21    left: 0;
22    top: 0;
23    background: rgba(0,0,0,.85);
24    height: 100vh;
25    margin: 0;
26    padding: 0;
27    transform: translateX(-99.99%);
28    transition: all .3s ease;
29    overflow: scroll; /* si se hace más pequeño el usuario puede hacer scroll hacia
30  abajo */
31  }
32  main .barra-lateral-izquierda.active {
33    transform: translateX(0%); /*Cuando esté active se mostrará*/
34  }

```

- Usaremos ahora Javascript para poner y quitar una clase que llamaremos active cuando pulsemos sobre la barra. Para ello crearemos una carpeta js donde guardaremos nuestros javascript y dentro un fichero con el siguiente nombre: `menu.js`:

```

1  document.addEventListener('DOMContentLoaded', function(){
2    /* Guardamos los elementos HTML para luego trabajar con ellos */
3    const boton = document.getElementById('btn-menu');
4    const fondoEnlace = document.getElementById('fondo-enlace');
5    const barraLateral = document.getElementById('barra-lateral-izquierda');
6
7    if(boton) {
8      /* Cuando se haga click en el boton menu */
9      boton.addEventListener('click', function(e){
10        e.preventDefault();
11        /* Añadimos o quitamos la clase CSS active
12        al fondo y la barra lateral */
13        fondoEnlace.classList.toggle('active');

```



```

14         barraLateral.classList.toggle('active');
15     });
16 }
17
18 /* Lo mismo pero cuando se pulsa el fondo del menu para cerrarlo */
19 if(fondoEnlace) {
20     fondoEnlace.addEventListener('click', function(e){
21         e.preventDefault();
22         fondoEnlace.classList.toggle('active');
23         barraLateral.classList.toggle('active');
24     });
25 }
26 });

```

- Y al final o principio de nuestro html añadiremos la referencia a este fichero:

```

1 | <script src="js/menu.js"></script>

```

# EVALUACIÓN

- Practica asociada a RA2, RA3, RA4, RA5 y RA6.
- **Corrección por pares:** La corrección se hará por pares, cada alumno corregirá a otro basándose en una escala que será proporcionada a continuación
- **Puntuación:** La puntuación máxima que se puede obtener de la práctica será 9 sobre 10, el otro punto se obtendrá de la corrección a un compañero. El punto solo se asignará si el alumno que realiza la corrección ha aprobado su propia práctica.
- **Fraude:** Si alguna de las correcciones fuera fraudulenta se penalizará al alumno que ha corregido con los puntos que haya "regalado" a su compañero. Se considera corrección fraudulenta cuando la corrección del alumno difiere de 2 puntos o más sobre la corrección objetiva. Por ejemplo, si la nota objetiva es un 4, y le has puesto un 8, se restará a tu nota esos 4 puntos de diferencia.
- **Plagio:** El plagio supondrá un cero tanto al plagiador como al plagiado. Se considerará plagio el hecho de que el 25% o más del código sea idéntico.
- **IA:** Si se detecta que la totalidad de la práctica se ha realizado con IA, se puede pedir al alumno una defensa oral de la práctica, donde demuestre su autoría. La calificación podrá ser modificada en función del resultado de dicha defensa.

## Criterios

- **HEADER (2.5 puntos)**
  - **Estructura del header** con logo, buscador y menú de iconos correctamente implementados (1 punto)
  - **Comportamiento responsive** del header: logo desaparece en small, buscador y menú cambian de orden y tamaño según breakpoints (1 punto)
  - **Estilos personalizados:** input con estilos propios, iconos con hover, espaciados correctos (0.5 puntos)
- **BARRA LATERAL IZQUIERDA (1.5 puntos)**
  - **Estructura y estilos** del menú lateral con iconos, correctamente posicionado (0.8 puntos)
  - **Responsive:** desaparece en móvil y aparece el botón hamburger (0.7 puntos)
- **CONTENIDO PRINCIPAL (2.5 puntos)**
  - **Formulario de publicación** con foto de perfil, textarea, iconos multimedia y botón correctamente maquetados (0.7 puntos)
  - **Publicaciones:** estructura correcta con foto de perfil, nombre, texto/imagen, botón de like (1 punto)
  - **Comentarios:** correctamente anidados con offset, foto y textarea/respuestas (0.8 puntos)
- **BARRA LATERAL DERECHA (0.5 puntos)**
  - **Publicidad y footer** correctamente posicionados y ocultos en dispositivos pequeños (0.5 puntos)
- **MENÚ MÓVIL CON JAVASCRIPT (2 puntos)**

- **Funcionalidad del menú hamburger:** el menú se desliza correctamente al pulsar el botón (1 punto)
- **Fondo oscuro y cierre:** aparece el fondo semitransparente y el menú se cierra al pulsar fuera (0.5 puntos)
- **JavaScript:** código correctamente implementado con `menu.js` (0.5 puntos)