



ENCICLOPEDIA
FRONT-END

LENGUAJES MODULE BUNDLERS PREPROCESADORES

¿QUÉ ESPERAS PARA APRENDER?

DECUBRIR FRONT-END

[Ver enciclopedia front-end](#)

Herramientas HTML y CSS

PEC 2: Web y recursos multimedia

Resumen

1. Portada
 - 1.1. Imágenes destacadas para diferentes dispositivos
 - 1.2. Elementos animados
 - 1.3. Favicon
2. Páginas de categoría
 - 2.1. Imagen destacada para diferentes dispositivos
3. Páginas en detalle
 - 3.1. Imagen destacada para diferentes dispositivos
 - 3.2. Imágenes en diferentes formatos
 - 3.3. Imagen con gestión de la dirección de arte
4. Header y footer
 - 4.1. Título, recurso gráfico con clip-path y animación
5. Estándares y calidad del código
 - 4.1. HTML 5
 - 4.2. CSS 3
6. Apéndice
 - 6.1. Url web enciclopedia front-end
 - 6.2. Url repositorio git web enciclopedia front-end
 - 4.3 Url boilerplate

1. Portada

1.1. Imágenes destacadas para diferentes dispositivos

En la portada tenemos 3 imágenes, una por cada categoría, que representan lenguajes, module bundlers y preprocesadores. Tenemos **5 breakpoints** (375 px, 740 px, 812 px, 1024 px y 1900 px) con los cuales indicamos al navegador cuando debe cambiar de imagen a las siguientes resoluciones 251 px, 354 px, 414 px, 444 px y 594 px.

Hemos elegido estos breakpoints y resoluciones después de hacer varias pruebas y comprobar el resultado con el **inspector de mozilla developers**. Hemos comprobado que en ésta página, 5 breakpoints aportan un resultado más fluido y responsive. Por ejemplo, la resolución de la imagen en el breakpoint de 1025 px es de 354x354px. Sin embargo, para una pantalla menor de 812 px es de 594x594 px.

Aunque a priori parece no tener sentido que la imagen para una pantalla más grande sea más pequeña, hay que tener en cuenta que este diseño de este template está a dos columnas para pantallas grandes y a **una columna para pantallas más pequeñas**. En el siguiente código se puede ver cómo hemos integrado las diferentes imágenes en nuestro código con el atributo picture.

index.html

```
<picture class="img-fluid rounded-circle" >
  <source media="(max-width: 375px)"
    srcset="./assets/img/portada/sass_251.jpg">

  <source media="(max-width: 740px)"
    srcset="./assets/img/portada/sass_414.jpg">

  <source media="(max-width: 812px)"
    srcset="./assets/img/portada/sass_594.jpg">

  <source media="(max-width: 1024px)"
```

```

srcset="./assets/img/portada/sass_354.jpg">

<source media="(max-width: 1900px)"
srcset="./assets/img/portada/sass_444.jpg">



</picture>

```

El **formato elegido para estas imágenes ha sido .jpg**, ya que no necesitamos una alta resolución, ni trabajamos con transparencias. En la siguiente tabla se puede apreciar la diferencia en el tamaño de las imágenes jpg entre la carpeta /img y /dist después de realizar npm run build.



251 px		354 px		414 px		444 px		594 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
14,4 KB	7 KB	21,8 KB	10 KB	25,2 KB	10,7 KB	13 KB	29,9 KB	17 KB	40,2 KB

Tabla 1.1: Optimización de imágenes portada - lenguajes



251 px		354 px		414 px		444 px		594 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
10,7 KB	5 KB	16,8 KB	7 KB	20,8 KB	8 KB	24,1 KB	10 KB	36,7 KB	14 KB

Tabla 1.1.2: Optimización de imágenes portada - module bundlers



251 px		354 px		414 px		444 px		594 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
11,2 KB	5 KB	19,6 KB	8 KB	22,2 KB	9 KB	23,4 KB	10 KB	34,9 KB	14 KB

Tabla 1.1.2: Optimización de imágenes portada - preprocesadores

1.2. Elementos animados

En cuanto a los elementos animados, encontramos el header que está compuesto de 9 burbujas que se mueven en diferentes sentidos y a diferentes velocidades. Por otro lado, también tenemos los call to action animados, ver página web <https://optimistic-liskov-890915.netlify.com/>.

Header

El efecto del header está **realizado en su totalidad a través de html y css**. En el html, tenemos 9 <div> con la respectiva clase, una para cada burbuja.

En el css, hemos asignado a cada clase una altura, una anchura, un radio, una posición y una animación. A cada animación hemos asignado un tiempo de acción. Después a través de los **keyframes hemos conseguido que se muevan** en tanto en el eje de la x como en el de la y, gracias a los valores bottom y left.

Por último en el css, hemos añadido la **propiedad transform para hacer que la burbuja se tambalee** ligeramente como aparecen en las burbujas de la página [css animation rock](#). A continuación, mostramos el código css de la burbuja 1 y el resultado del header en la imagen 1.2.

style.css

```
header.masthead .bg-circle-1 {
  width: 200px;
  height: 200px;
  border-radius: 50%;
  position: absolute;
  left: 0px;

  animation: bubble-1 10s linear infinite alternate;
}

@keyframes bubble-1 {
  0% {
    transform: scale(1);
    left: 0px;
    bottom: -100px;
```

```
}  
20% {  
  transform: scaleY(0.95) scaleX(1.05);  
  left:0px;  
  bottom:0px;  
}  
48% {  
  transform: scaleY(1.1) scaleX(0.9);  
  left:0px;  
  bottom:100px;  
}  
68% {  
  transform: scaleY(0.98) scaleX(1.02);  
  left:0px;  
  bottom:200px;  
}  
80% {  
  transform: scaleY(1.02) scaleX(0.98);  
  left:0px;  
  bottom:300px;  
}  
97% {  
  transform: scale(1);  
  left:0px;  
  bottom:400px;  
}
```

```

100% {
  transform: scale(1);
  left:0px;
  bottom:500px;
}
}

```

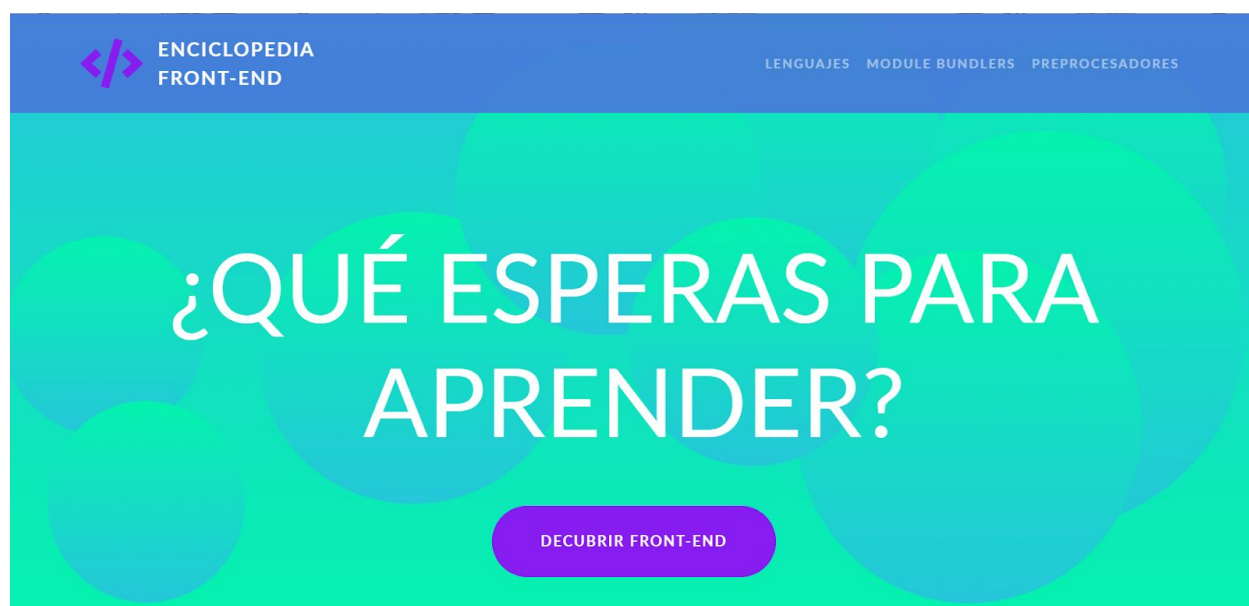


Imagen 1.2: Header animado de burbujas

Call to action

No nos olvidemos de los botones, a los que también hemos añadido una animación. Están presentes en casi todas las páginas de este proyecto, pero hemos elegido la portada para explicarlos porque es la **página más representativa ya que aparecen todas las animaciones** de los botones.

En primer lugar, tenemos la animación del **botón primario que se encuentra en el header**, como podemos ver en la imagen superior 1.2. A este call to action le hemos añadido una animación a través del keyframe como podemos apreciar en el siguiente código.

style.css

```
@keyframes primerboton{
  0%{
    opacity:0;
    transform:translateY(40px);
  }
  100%{
    opacity:1;
    transform:translateY(0);
  }
}
```

Por último, tenemos el **botón secundario**, el que es el que enlaza al contenido de las páginas de categoría y detalle como se puede apreciar en la imagen 1.2.1. A este call to action le hemos añadido una animación rotate a través del keyframe que podemos ver más abajo.

style.css

```
@keyframes rotate {
  0% {
    transform: rotate(0deg) translate3d(0, 0, 0);
  }
  25% {
    transform: rotate(3deg) translate3d(0, 0, 0);
  }
  50% {
    transform: rotate(-3deg) translate3d(0, 0, 0);
  }
}
```

```

}
75% {
    transform: rotate(1deg) translate3d(0, 0, 0);
}
100% {
    transform: rotate(0deg) translate3d(0, 0, 0);
}
}

```

Lenguajes

Descubre todo lo que necesitas saber sobre los lenguajes de front-end. Principalmente HTML, HyperText Markup Language, es el componente estructural clave de todas las webs de internet. Sin él las páginas web no pueden existir. CSS Cascading Style Sheets, es lo que le proporciona estilo a HTML. JS, Usando solo HTML y CSS tus webs serían páginas estáticas, con JS tus páginas web son interactivas.

SABER MÁS



Imagen 1.2.1: Call to action secundario animado

1.3. Favicon

Después de realizar una rápida búsqueda en Internet, hemos encontrado la página de [w3.org](https://www.w3.org) donde explica cómo añadir un favicon a nuestro proyecto. Antes de nada, hemos creado un logo en Inkscape con una **resolución de 32x32 píxeles**. Para visualizar el favicon que hemos creado

en la pestaña del navegador como aparece en la imagen 1.3, hemos insertado el siguiente código en nuestro template html.

index.html

```
<!-- Favicon -->  
<link rel="icon" type="image/png" href="./assets/img/favicon.png">
```

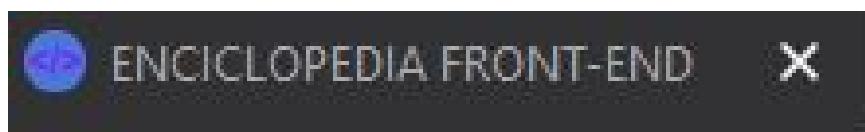


Imagen 1.3: Favicon

2. Páginas de categoría

2.1. Imagen destacada para diferentes dispositivos

En esta web tenemos 3 páginas de categoría, cada una de estas contiene un banner y 3 imágenes destacadas, una por artículo. Los **banners tienen 7 breakpoints** (414 px, 742 px, 768 px, 900 px, 1204 px, 1500 px y 1905 px) con los cuales indicamos al navegador cuando debe cambiar de imagen a las siguientes resoluciones 414 px, 812 px, 768 px, 896 px, 1204 px, 1262 px y 1905 px.

Por otro lado, las **imágenes destacadas de los artículos tienen 5 breakpoints** (414 px, 740 px, 900 px, 1204 px y 1900 px) que corresponden a las siguientes resoluciones 330 px, 510 px, 390 px, 530 px y 635 px.

Hemos elegido estos breakpoints y anchura de imágenes después de hacer varias pruebas y comprobar el resultado con el **inspector de mozilla developers**. Hemos comprobado que en la página de categoría para el banner el resultado más fluido y responsive eran 7 breakpoint mientras que para las imágenes destacadas de los artículos solo necesitamos 5.

En el siguiente código se puede ver cómo hemos integrado las diferentes imágenes en nuestro código con el atributo picture.

lenguajes.html

```
<!-- BANNER -->

<picture class="img-fluid rounded mb-3 mb-md-0" >
  <source media="(max-width: 414px)"
    srcset="./assets/img/articulos/html_330.jpg">

  <source media="(max-width: 740px)"
    srcset="./assets/img/articulos/html_510.jpg">
```

```
<source media="(max-width: 900px)"
      srcset="/assets/img/articulos/html_390.jpg">
```

```
<source media="(max-width: 1024px)"
      srcset="/assets/img/articulos/html_530.jpg">
```

```
<source media="(max-width: 1900px)"
      srcset="/assets/img/articulos/html_635.jpg">
```

```

```

```
</picture>
```

```
<!------- IMG DESTACADA ARTICULO ----->
```

```
<picture class="img-fluid rounded-circle" >
```

```
<source media="(max-width: 375px)"
      srcset="/assets/img/portada/sass_251.jpg">
```

```
<source media="(max-width: 740px)"
      srcset="/assets/img/portada/sass_414.jpg">
```

```
<source media="(max-width: 812px)"
      srcset="/assets/img/portada/sass_594.jpg">
```

```
<source media="(max-width: 1024px)"
```

```
srcset="./assets/img/portada/sass_354.jpg">

<source media="(max-width: 1900px)"
srcset="./assets/img/portada/sass_444.jpg">



</picture>
```

El **formato elegido para estas imágenes ha sido .jpg**, ya que no necesitamos una alta resolución, ni trabajamos con transparencias. A continuación se puede apreciar la diferencia en el tamaño de las imágenes jpg entre la carpeta /img y /dist después de realizar npm run build.



410 px		768 px		812 px		896 px		1204 px		1262 px		1905 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
15,2 KB	5,52 KB	30,6 KB	10,07 KB	10,07 KB	3,90 KB	14 KB	5 KB	31 KB	11 KB	33 KB	10 KB	40 KB	12 KB

Tabla 2.1: Optimización de imágenes categoría - banner - lenguajes



410 px		768 px		812 px		896 px		1204 px		1262 px		1905 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
14,2 KB	4,81 KB	30,6 KB	10,04 KB	11,07 KB	4,30 KB	15 KB	5,71 KB	31,4 KB	10,3 KB	35,8 KB	11,7 KB	46,1 KB	14,8 KB

Tabla 2.1.1: Optimización de imágenes categoría - banner - module bundlers



410 px		768 px		812 px		896 px		1204 px		1262 px		1905 px	
/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist	/img	/dist
16,2 KB	5,1 KB	31,7 KB	5,42 KB	14,07 KB	6,04 KB	16 KB	10 KB	34 KB	11,7K B	33 KB	12,6 KB	40 KB	17,3 KB

Tabla 2.1.2: Optimización de imágenes categoría - banner - preprocesadores

* Nota: Debido a la falta de tiempo, no he podido continuar añadiendo a la documentación las tablas con la listas de imágenes optimizadas.

3. Páginas en detalle

3.1. Imagen destacada para diferentes dispositivos

Esta página web contiene 3 páginas en detalle, que corresponde a html, parcel y sass. Cada página en detalle cuenta con un **banner superior que cuenta con 7 breakpoints** (414 px, 742 px, 768 px, 900 px, 1204 px, 1500 px y 1905 px) con los cuales indicamos al navegador cuando debe cambiar de imagen a las siguientes resoluciones 414 px, 812 px, 768 px, 896 px, 1204 px, 1262 px y 1905 px.

3.2. Imágenes en diferentes formatos

En cuanto a los formatos, hemos decidido mantener el **banner con el formato jpg**, ya que no necesitamos una alta resolución, ni trabajamos con transparencias. Hemos insertado dos imágenes en el cuerpo del texto.

En la página que habla sobre **html**, tenemos **en primer lugar una imagen de un código en png**, ya que es necesario que tenga una alta resolución para poder leer lo que muestra el programa. **La segunda imagen está en jpg** porque es una imagen que no necesita una alta resolución, ya que muestra solo un workshop, ver resultado en el siguiente enlace <https://optimistic-liskov-890915.netlify.com/html.html>.

Para la página de **parcel**, hemos utilizado en **primer lugar** una imagen procedente del gif que tiene la página de parcel, lo hemos guardado en **png, ya que esta imagen tiene transparencias**. En **segundo lugar**, tenemos la imagen de una pequeña conferencia sobre parcel, como no es necesaria una alta resolución **hemos utilizado jpg**, se pueden apreciar las diferencias en la página web <https://optimistic-liskov-890915.netlify.com/mbparcel.html>.

En la página que habla sobre **sass**, tenemos **en primer lugar una imagen de un código en png**, ya que es necesario que tenga una alta resolución para poder leer lo que muestra el programa. **La segunda imagen está en jpg** porque es una imagen que no necesita una alta resolución, ya que muestra solo un workshop, ver resultado en el siguiente enlace <https://optimistic-liskov-890915.netlify.com/sass.html>.

3.3. Imagen con gestión de arte

En cuanto a la gestión de arte en las páginas en detalle, corresponde a la segunda imagen que se encuentra en el cuerpo del texto, las imágenes de las tres páginas corresponde a workshop o conferencias sobre los distintos temas y se recortan las imágenes para enfocar al presentador en los formatos más pequeños, como podemos apreciar en la siguiente imagen 3.3.



Imagen 3.3: Gestión de arte

Ejemplo del código insertado para realizar la gestión de arte utilizando el elemento picture.

sass.html

```
<picture class="img-fluid rounded mb-3 mb-md-0" >
  <source media="(max-width: 414px)"
    srcset="./assets/img/detalle/sass_art_direction_330.jpg">

  <source media="(max-width: 740px)"
    srcset="./assets/img/detalle/sass_art_direction_510.jpg">

  <source media="(max-width: 900px)"
    srcset="./assets/img/detalle/sass_art_direction_690.jpg">
```

```

<source media="(max-width: 1024px)"
      srcset="/assets/img/detalle/sass_art_direction_930.jpg">

<source media="(max-width: 1900px)"
      srcset="/assets/img/detalle/sass_art_direction_1110.jpg">


</picture>

```

4. Header y footer

4.1. Título, recurso gráfico con clip-path y animación

Hemos rediseñado el footer, añadiendo el título y el logo, elementos que no se encontraban en el pec anterior. Además se han integrado **dos clip-path y una animación**, para la elaboración de los clip-path hemos usado la página [CSS clip-path maker](#).

Gracias al **elemento transition en CSS**, hemos podido crear una animación, cuando pasamos el ratón sobre el footer, el clip-path de la versión desktop **cambia de forma**, se puede ver en las imágenes 4.1. y 4.2.

Por último, se han **invertido los colores de la degradación** de fondo en el estado “hover”, este cambio, así como los anteriormente citados, se puede apreciar en siguiente código.

style.css

```
.footer {
```

```
padding: 120px 0;
text-align: center;
background: linear-gradient(0deg, #4878D9 0%, #25C7D9 100%);
background-repeat: no-repeat;
background-position: center center;
background-size: cover;
font-family: 'Lato';
color:white;

clip-path: polygon(31% 20%, 53% 0, 76% 19%, 100% 0, 100% 100%, 0 100%, 0 21%, 14% 0);
transition: clip-path 1s;
}

.footer:hover {
background: linear-gradient(0deg, #25C7D9 0%, #4878D9 100%);
clip-path: polygon(35% 0, 56% 16%, 81% 0, 100% 20%, 100% 100%, 0 100%, 0 0, 16% 21%);
}
```



Imagen 4.1: Footer reposo



Imagen 4.2: Footer hover

5. Estándares y calidad del código

5.1. HTML 5

Falta de tiempo para controlar la validación

5.2. CSS 3

Falta de tiempo para controlar la validación

6. Apéndice

4.1. Url web enciclopedia front-end

<https://optimistic-liskov-890915.netlify.com/index.html>

4.2. Url repositorio git web enciclopedia front-end

<https://github.com/inmafernandez/enciclopedia-front-end>

4.3. Url boilerplate

<https://github.com/inmafernandez/boilerplate>