

## Índice

Documentación del proceso de desarrollo.....	2
Justificación de las decisiones tomadas en el desarrollo.....	3
Decisiones sobre la estructura de archivos.....	3
Decisiones sobre cuando usar Bootstrap.....	3
Importación de Bootstrap.....	3
Customize de Bootstrap.....	4
Soporte de grid.....	4
Decisiones de diseño del sitio web.....	5
Página: Portada.....	5
Página: La Banda.....	5
Página: La Historia.....	6
Página: Gira 2021.....	6
Justificación de la selección de la metodología y guía de estilos y memoria de la aplicación de los criterios elegidos.....	6
Conclusiones.....	7
Anexo I. Enlaces a las fuentes de las imágenes con licencias gratuitas o Creative Commons.....	7

## Documentación del proceso de desarrollo

Para el desarrollo de la página se han seguido los siguientes pasos:

1.- Clonación de Boilerplate de la PEC1.

He continuado utilizando las recomendaciones de mdo junto con la nomenclatura de BEM. Por ello he mantenido la instalación y configuración de stylelint. También he continuado usando la librería de iconos fontawesome y el módulo para añadir opciones de accesibilidad al front-end “accessibility”

2.- He añadido la dependencia bootstrap para poder utilizar la librería de componentes  
`npm install @popperjs/core bootstrap@next --save.`

3.- Para facilitar el desarrollo he continuado utilizando parcel versión dev para poder apreciar los cambios de forma dinámica en la página web. Parcel está integrado con el precompilador Sass.

También se han utilizado las herramientas para el desarrollador de Firefox Developer edition para analizar la aplicación de los estilos a los elementos html. Se ha utilizado asimismo la herramienta para comprobar como se ajustaba la página de forma responsive a los diferentes dispositivos móviles.

Una vez finalizado el desarrollo se ha procedido a generar la versión de producción con parcel build

Se ha utilizado Github como repositorio

Se puede acceder al repositorio en:

[https://github.com/ptousd/eines2\\_pec2](https://github.com/ptousd/eines2_pec2)

Para paso a producción se ha utilizado Netlify.

Se puede acceder al proyecto en la url:

<https://thirsty-benz-a83646.netlify.app>

Una vez en producción se ha analizado el código con las siguientes herramientas:

Para corregir problemas de html

<https://validator.w3.org/>

Para corregir problemas de css

<https://jigsaw.w3.org/css-validator>

Para validar accesibilidad de las páginas online

<https://achecker.ca/checker/index.php>

A partir de aquí se ha hecho Continuous Deployment hasta conseguir que los validadores no encontrasen problemas.

# Justificación de las decisiones tomadas en el desarrollo

## Decisiones sobre la estructura de archivos

Se ha utilizado la misma estructura de carpetas que la propuesta en el Boilerplate. Al principio estudié la posibilidad de utilizar la estructura de archivos de SMACSS o ITCSS pero se ha descartado dado lo reducido del proyecto. Al utilizar la librería de componentes Bootstrap todavía se ha reducido más el tamaño del archivo css.

## Decisiones sobre cuando usar Bootstrap

Con el objetivo de aprender a utilizar al máximo la librería se han utilizado sus componentes en casi todo el proyecto. El resultado es que prácticamente no he tenido que escribir código css. Sólo en tres excepciones:

- 1.- El logo, ya que en este caso me ha parecido más simple al tener que aplicar la fuente de “`$font-family-headings`” al texto, y un tamaño de imagen muy reducido.
- 2.- En el poster para poder llevar a la práctica lo aprendido en el curso. Aunque no creo que bootstrap tenga la suficiente flexibilidad para trabajar de esta forma con grid.
- 3.- En la página de los componentes del grupo por indicación del enunciado de la práctica.

En el escaso código css escrito se ha utilizado BEM para nombrar los selectores y se han seguido las recomendaciones de @mdo.

## Importación de Bootstrap

He seguido las instrucciones para incorporar sólo las partes de Bootstrap que necesito y reducir el tiempo de carga de las páginas.

En main.scss

```
@import "bootstrap/scss/_functions";
@import "bootstrap/scss/_variables";
@import "bootstrap/scss/_mixins";
@import "bootstrap/scss/_utilities";
@import "bootstrap/scss/_root";
@import "bootstrap/scss/_reboot";
@import "bootstrap/scss/_buttons";
@import "bootstrap/scss/_carousel";
@import "bootstrap/scss/_close";
@import "bootstrap/scss/_containers";
@import "bootstrap/scss/_forms";
@import "bootstrap/scss/_grid";
@import "bootstrap/scss/_list-group";
@import "bootstrap/scss/_modal";
@import "bootstrap/scss/_nav";
@import "bootstrap/scss/_navbar";
@import "bootstrap/scss/_helpers";
@import "bootstrap/scss/_type";
@import "bootstrap/scss/utilities/_api";
```

En main.js

```
import 'bootstrap/js/dist/carousel';  
import 'bootstrap/js/dist/modal';  
import 'bootstrap/js/dist/collapse';
```

## Customize de Bootstrap

He customizado la paleta de colores:

```
$primary: #17a095;  
$secondary: #6590aa;  
$success: #c49815;  
$dark: #444;
```

```
$theme-colors: (  
  "primary": $primary,  
  "secondary": $secondary,  
  "success": $success,  
  "dark": $dark  
);
```

Y las fuentes de base para el texto

```
$font-family-base: 'Montserrat', serif;
```

## Soporte de grid

El poster de la página de portada se ha desarrollado con grid siguiendo las explicaciones del curso. Al ser un layout no soportado por navegadores antiguos se ha utilizado la querie:

```
@supports (display:grid) { ... }
```

Para la aplicación de estilos en caso de no soportarse grid inicialmente se intentó

```
@supports not (display:grid) { ... }
```

siguiendo las instrucciones de MDN

El problema surge con Internet Explorer que no es capaz de interpretar la querie y no tienen en cuenta los selectores incluidos.

Para salvar el problema se ha decidido primero escribir el código para los navegadores que no soportan grid sin querie de supports, para luego reescribir el código para aquellos navegadores que sí lo soportan con supports grid, siguiendo el patrón que se conoce como "progressive enhancement".

Cuando no hay soporte grid se ha utilizado el layout flex en su lugar.

## Decisiones de diseño del sitio web

La página se ha dividido en 4 grandes áreas. Una cabecera, un menú de navegación, un contenido y un pie de página. Utilizando las etiquetas semánticas propias de HTML-5 para mejorar la accesibilidad.

Los menús se han implementado con Navbar de Bootstrap

Aunque el diseño inicial según los wireframes es sobre dispositivos de escritorio, se ha utilizado un desarrollo first-mobile ya que he considerado que es la mejor manera de que se vea bien en dispositivos pequeños.

Se han utilizado dos puntos de ruptura con media-queries: el primero es para pantallas de resolución 768px que son las que más se aproximan a un entorno tablet, y el segundo para 1440px que es para equipos de escritorio. Se han utilizado estos puntos de ruptura ya que se consideran estándares.

Primero se ha desarrollado la web sin puntos de ruptura y probado en dispositivos tipo móvil. En segundo lugar se ha añadido el punto a 768px para adaptar el funcionamiento a tablets, para finalmente añadir el punto de ruptura para visualizar en formato escritorio.

Todo ello para seguir la metodología first-mobile.

Se ha añadido javascript para utilizar la función de accesibilidad de la dependencia accessibility. Cualquier esfuerzo por mejorar la accesibilidad de una página vale la pena. Aunque hay que decir que con este módulo el esfuerzo ha sido mínimo.

### Página: Portada

Se ha utilizado un diseño tipo póster para dispositivos a partir de tamaño tablet.

### Página: La Banda

Se ha utilizado un diseño en flex wrap para dispositivos a partir de tamaño tablet.

Para cada componente del grupo se ha asignado un tamaño fijo de 45% en tablet (2 columnas) y 30% en escritorio (3 columnas), con flex-grow y flex-shrink a 0. Al tener 8 miembros en una distribución de 3 columnas quedan 2 miembros en la última fila. Para mantener una apariencia de tamaño de columna homogénea se han utilizado para flex-grow y flex-shrink el valor 0 que impide que se reparta el espacio disponible. Se ha considerado la mejor opción de diseño.

```
.banda {  
  margin: 0;  
  padding-right: .5em;  
  padding-left: .5em;  
  list-style: none;  
  
  @media only screen and (min-width: 768px) {  
    display: flex;  
    flex-flow: row wrap;  
    justify-content: center;  
  }  
}
```

```
}  
  
}  
  
.miembro {  
  margin-bottom: 1em;  
  
@media only screen and (min-width: 768px) {  
  flex: 0 0 45%;  
  margin-bottom: 1.5em;  
  margin-left: 1.5em;  
  display: flex;  
  flex-direction: column;  
  justify-content: space-between;  
}  
  
@media only screen and (min-width: 1440px) {  
  flex: 0 0 30%;  
}  
  
}
```

## Página: La Historia

Página muy sencilla en la que se ha explorado el uso de **Utilities** y **Content** de bootstrap

## Página: Gira 2021

Lo más destacable es que se han utilizado los componentes Carousel, Modal y Forms

## Justificación de la selección de la metodología y guía de estilos y memoria de la aplicación de los criterios elegidos.

Para el desarrollo de la práctica se ha seguido la guía de estilos de mdo en combinación con la metodología BEM.

Se ha escogido mdo como guía por considerarla muy completa y además fácilmente trasladable a reglas de stylelint.

Como metodología se utiliza BEM ya que se considera que el contenido de la página se adapta muy bien a una división del tipo: bloque – elemento – modificador. Además también permite fácilmente mejorar la reutilización del código, mejora la modularidad y la estructura de nuestro CSS y reduce mucho la especificidad en los selectores.

He escogido esta metodología por delante de OOCSS ya que me ha gustado mucho la forma de nombrar los selectores y una vez seleccionada para el naming es más fácil continuar con la misma metodología.

Quizás para proyectos más grandes es mejor utilizar OOCSS al facilitar más la división entre container y contenido, y entre estructura y skin. Aunque lo cierto es que para tomar una decisión final necesitaría la experiencia de trabajar en proyectos más complejos en ambas metodologías.

Como ya he indicado, dado el reducido tamaño del proyecto, no se ha utilizado ni SMACSS ni ITCSS para la división de los archivos css.

La metodología no está reñida con el uso de Bootstrap ya que esta librería está básicamente enfocada a la reducción al máximo de la codificación css utilizando selectores completamente reutilizables y con mínima especificidad.

En el caso de Bloques de estructuras más complejas es cuando se ha utilizado BEM. Tal como ya he indicado en tres bloques: logo, poster y banda.

## Conclusiones

He podido comprobar que bootstrap es una herramienta pensada para olvidarse de escribir css. Solamente con conocimientos de HTML es posible crear páginas muy completas y visualmente atractivas.

Aunque su uso pueda parecer una decisión errónea por el riesgo de crear sitios web clonados de otros que usa bootstrap, la flexibilidad que ofrece en cuanto a configuración es lo suficiente alta como para que sean lo suficientemente originales.

Además siempre se puede compaginar su uso con la utilización de nuestros propios selectores. Bootstrap para facilitar la codificación y nuestros selectores para conseguir un efecto diferenciador.

Un ejemplo lo encontramos en esta misma práctica en el caso del poster de la página principal.

Indicar también que la aparición de los layouts: grid y flex han facilitado enormemente el diseño responsive de las páginas web. Algo fundamental en el panorama actual con tan alta variedad de dispositivos.

## Anexo I. Enlaces a las fuentes de las imágenes con licencias gratuitas o Creative Commons

Conciertos

[https://upload.wikimedia.org/wikipedia/commons/7/78/Concierto\\_OT\\_BCN\\_2018.png](https://upload.wikimedia.org/wikipedia/commons/7/78/Concierto_OT_BCN_2018.png)

[https://upload.wikimedia.org/wikipedia/commons/6/63/Natalia\\_Lafourcade\\_en\\_concierto.jpg](https://upload.wikimedia.org/wikipedia/commons/6/63/Natalia_Lafourcade_en_concierto.jpg)

La historia

<https://pixnio.com/people/crowd/music-concert-performance-music-stage-festival-nightclub-audience>

Guitarra logo

<https://pixabay.com/es/photos/guitarra-guitarra-el%C3%A9ctrica-2957224/>

Guitarra eléctrica

<https://pixnio.com/es/fondos-de-pantalla/guitarrista-musica-guitarra-electrica-altavoz-musico-instrumento-sonido>

Acustica

<https://pixnio.com/es/gente/mujeres/mujer-guitarra-clasica-guitarra-acustica-bajo-musica-guitarrista-melodia>

Saxo

<https://pixabay.com/es/photos/saxof%C3%B3n-instrumento-musical-m%C3%BAsica-3246650/>

bateria

<https://pixabay.com/es/photos/bater%C3%A1-conjunto-personas-hombre-2599508/>

piano

<https://pixnio.com/es/diverso/piano-musico-artista-sonido-mano-pianista>

acordeon

<https://pixnio.com/es/media/musico-acordeon-contacto-directo-musica-instrumento>

violin

<https://pixnio.com/es/gente/mujeres/mujer-artista-violin-musica>

cantante

<https://pixabay.com/es/photos/cantante-silhouette-concierto-1595864/>

Campo mallorca

<https://commons.wikimedia.org/wiki/File:Mallorca0-2Leganes.jpg>

Plaza Toros

<https://commons.wikimedia.org/wiki/>

[File:Die Arena des Colisseu Balear umfasst 11.000 Quadratmeter - panoramio.jpg](#)

Pabellón multiusos

[https://commons.wikimedia.org/wiki/File:Multiusos\\_Fontes\\_do\\_Sar\\_-\\_02.jpg](https://commons.wikimedia.org/wiki/File:Multiusos_Fontes_do_Sar_-_02.jpg)