

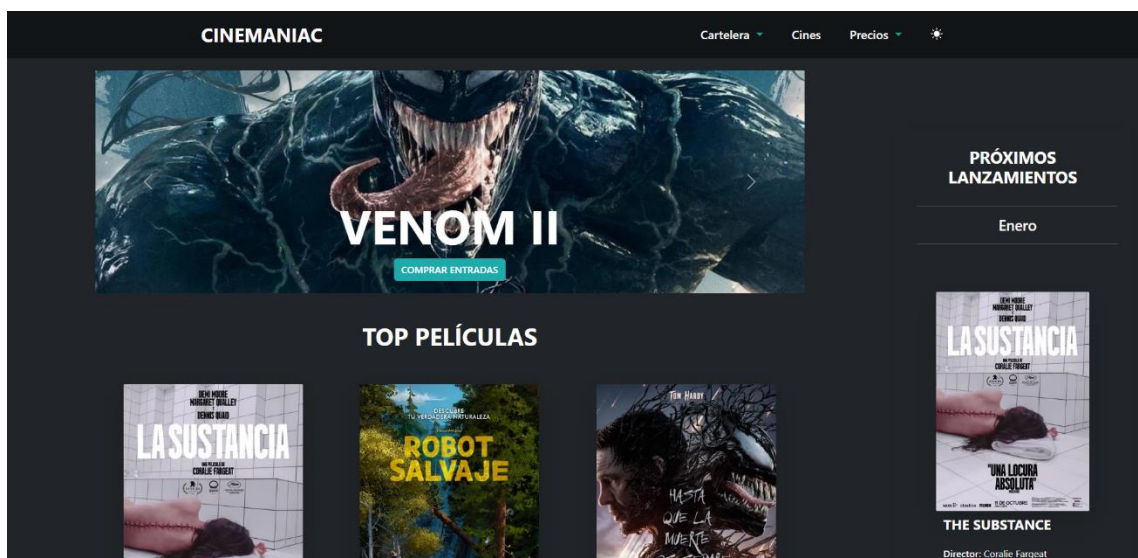
1. Descripción del proyecto:

Se trata de la página web de un cine llamado Cinemaniac. En la página principal aparecen las películas que actualmente se pueden ver y al lado hay un pequeño menú donde se anuncian los próximos estrenos. En la página de socio un usuario podrá suscribirse para recibir notificaciones y correos sobre esos próximos estrenos. También hay una sección para consultar la localización de los distintos cines que existen de Cinemaniac, una pestaña para consultar los precios, una zona para comprar entradas y una página de preguntas frecuentes entre otras.

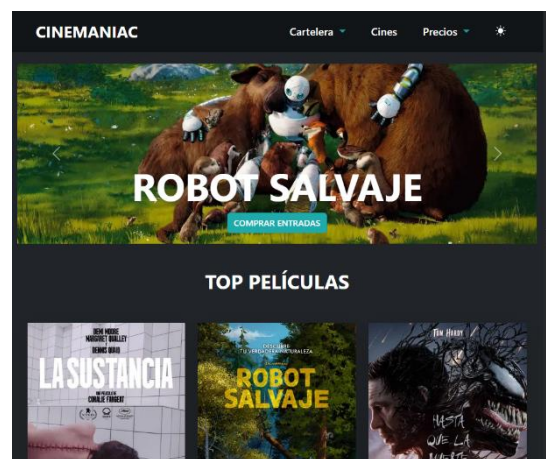
2. Diseño responsive:

index.html:

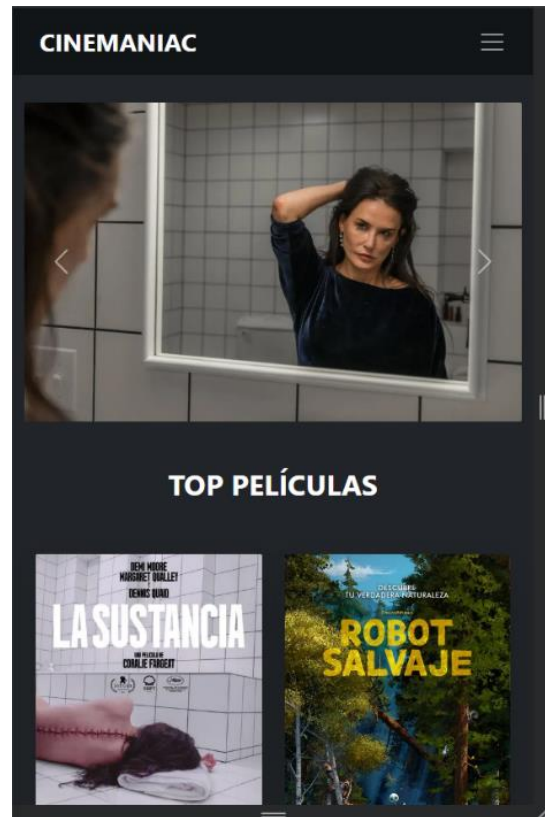
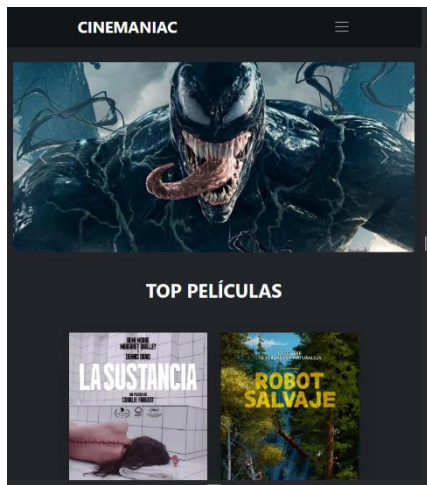
Tamaño Extra Large ($\geq 1200\text{px}$): En la parte superior hay un menú de navegación para acceder a las otras pestañas. Las películas aparecen en 3 columnas, y a la derecha hay un iframe que despliega los próximos lanzamientos (proximos.html).



Tamaño Large ($\geq 992\text{px}$): En la parte superior hay un menú de navegación para acceder a las otras pestañas. Las películas aparecen en 2 columnas, y el iframe de la derecha desaparece.

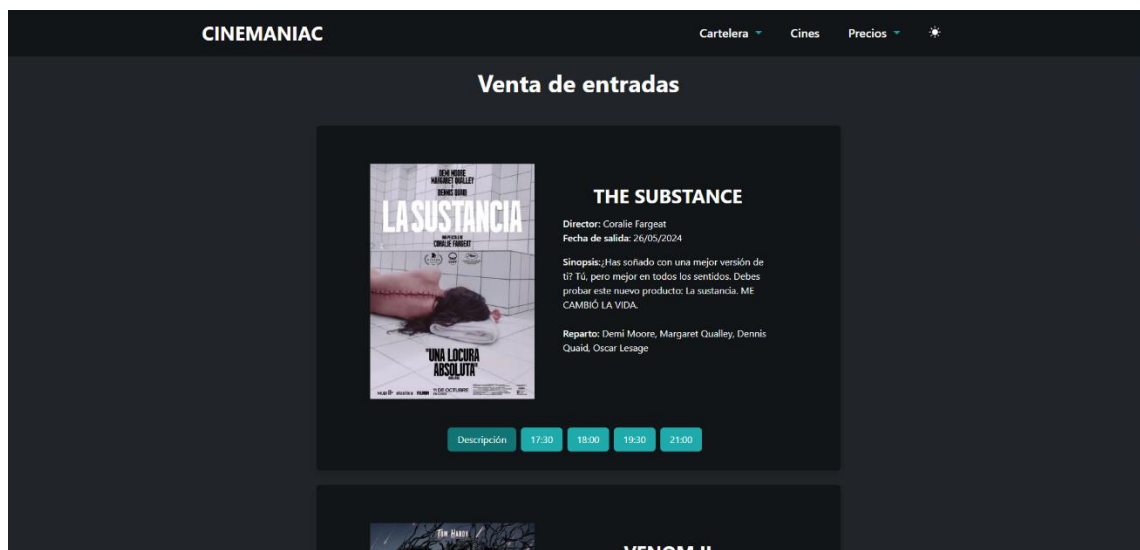


Tamaño Medium y Small ($\geq 768\text{px}$ y $\geq 576\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Las películas aparecen en 2 columnas, y el iframe de la derecha desaparece.



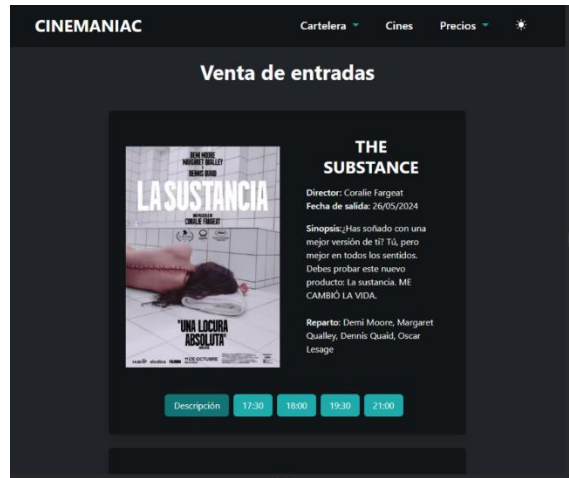
entradas.html:

Tamaño Extra Large ($\geq 1200\text{px}$): En la parte superior hay un menú de navegación para acceder a las otras pestañas. Las películas aparecen unos contenedores donde sale el título de la película, el cartel de la película y su descripción. En los contenedores se puede seleccionar la hora para poder comprar las entradas.

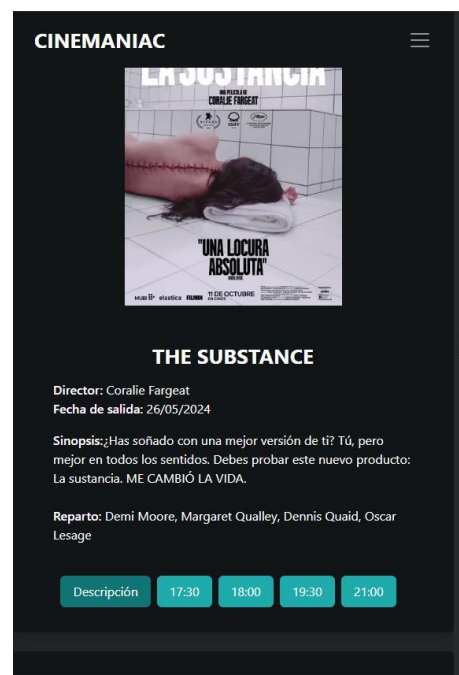
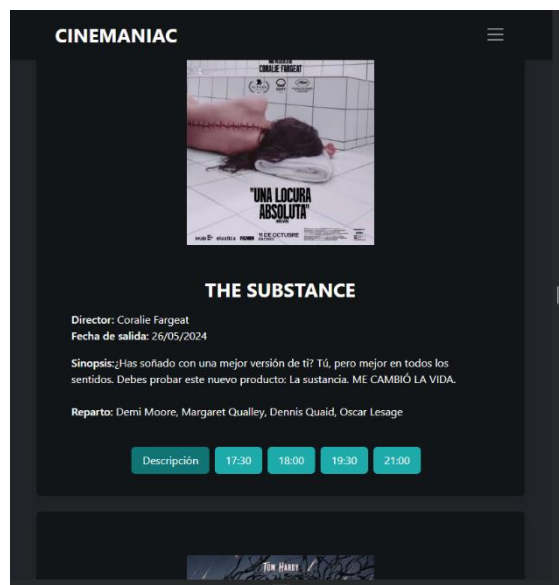


Documentación Bootstrap

Tamaño Large ($\geq 992\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Las películas aparecen unos contenedores donde sale el título de la película, el cartel de la película y su descripción. En los contenedores se puede seleccionar la hora para poder comprar las entradas.

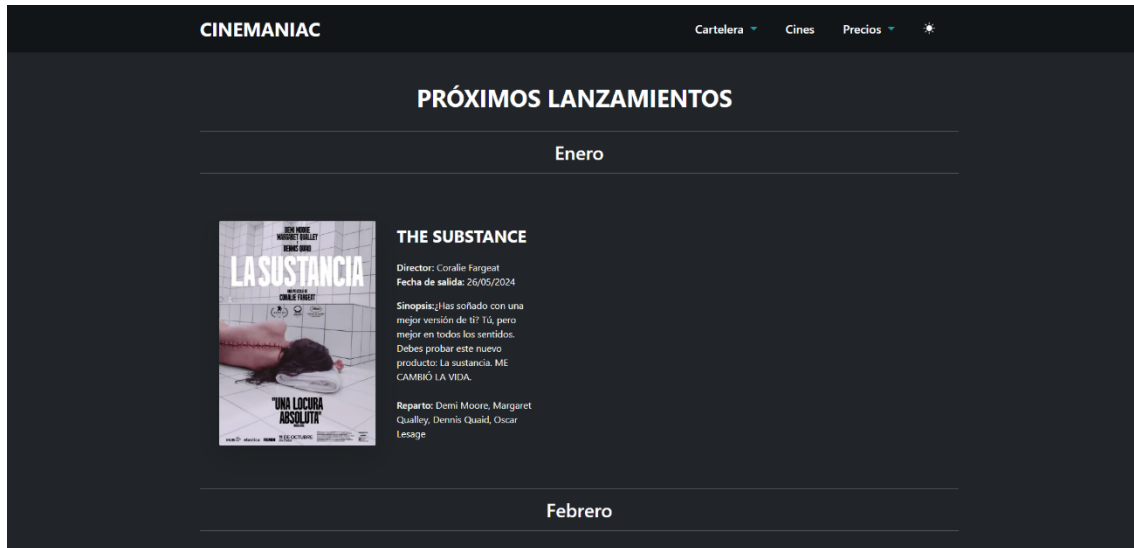


Tamaño Medium y Small ($\geq 768\text{px}$ y $\geq 576\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Las películas aparecen unos contenedores donde sale el título de la película en debajo del cartel de la película junto con su descripción. En los contenedores se puede seleccionar la hora para poder comprar las entradas.

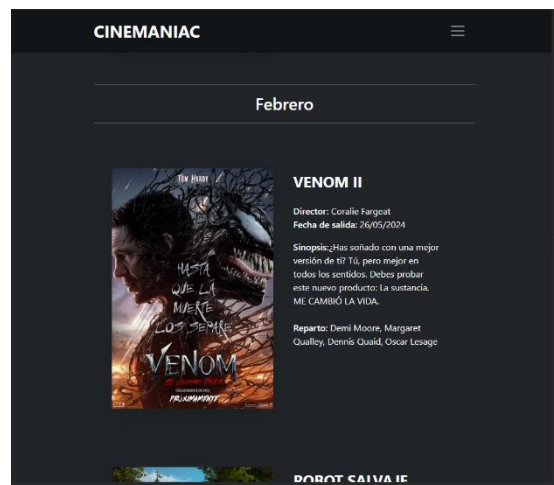


[proximos.html](#)

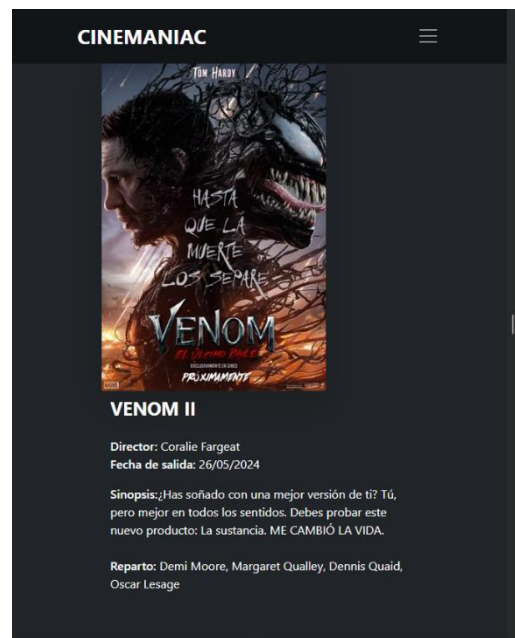
Tamaño Extra Large ($\geq 1200\text{px}$): En la parte superior hay un menú de navegación para acceder a las otras pestañas. Los próximos lanzamientos salen en filas que se dividen por el mes de salida en un máximo de 2 columnas por mes.



Tamaño Large y Medium ($\geq 992\text{px}$ y $\geq 768\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Los próximos lanzamientos salen en filas que se dividen por el mes de salida en un máximo de 1 columna por mes.

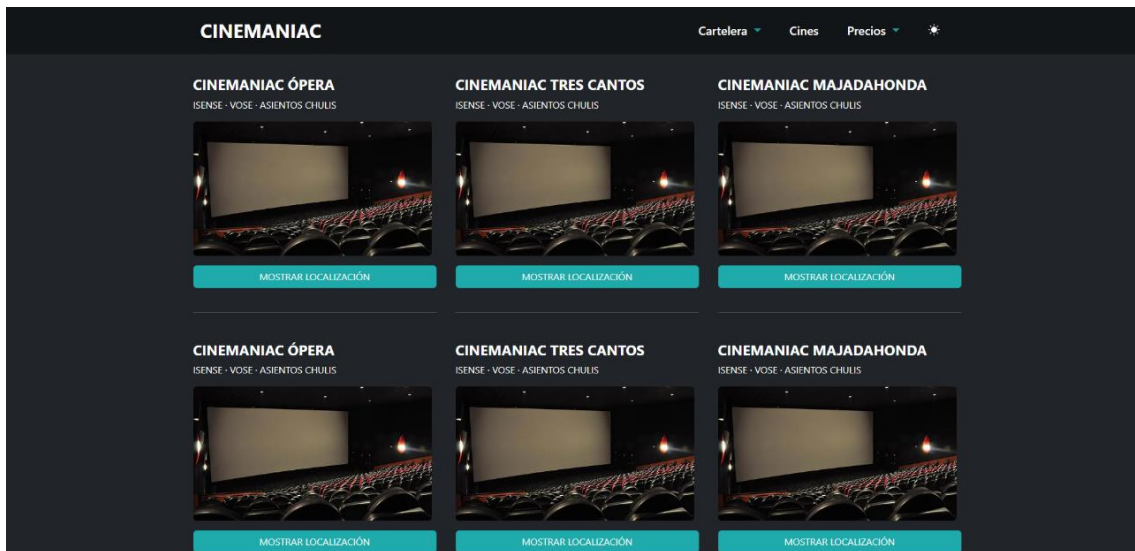


Tamaño Small ($\geq 576\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Los próximos lanzamientos salen en filas que se dividen por el mes de salida en un máximo de 1 columna por mes y el título y la descripción se desplazan debajo del cartel.

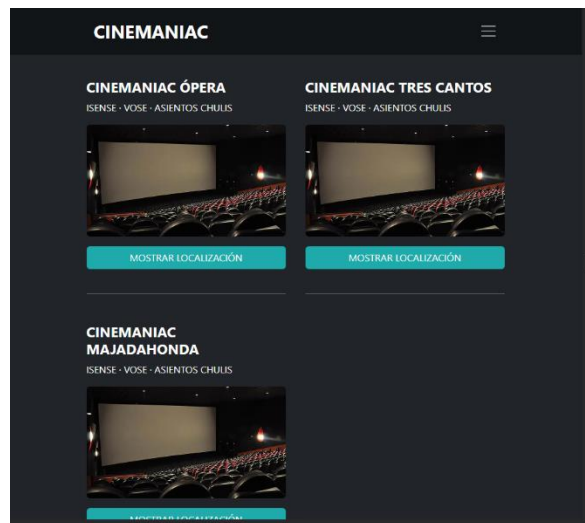


cines.html

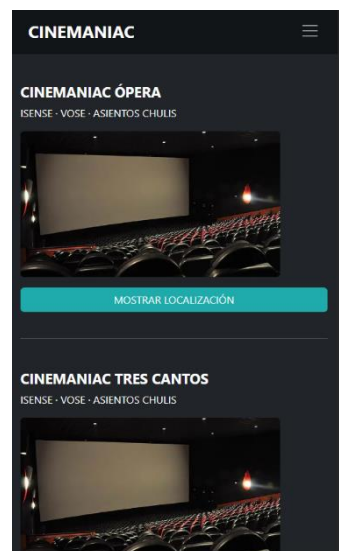
Tamaño Extra Large ($\geq 1200\text{px}$): En la parte superior hay un menú de navegación para acceder a las otras pestañas. Los cines aparecen en varias filas de 3 columnas cada uno.



Tamaño Large y Medium ($\geq 992\text{px}$ y $\geq 768\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Los cines aparecen en varias filas de 2 columnas cada uno.



Tamaño Small ($\geq 576\text{px}$): En la parte superior el menú desaparece y se transforma en uno desplegable desde la derecha para acceder a las otras pestañas. Los cines aparecen en varias filas de 1 columna cada uno.



preguntas.html:

En pantallas menores a md las pestañas nav para navegar entre acordeones se muestran de forma horizontal por encima del contenido ocupando toda la pantalla (12 columnas) al igual que los acordeones.

En pantallas de tamaño entre md y lg las tabs se muestran de forma vertical a la izquierda del contenido, con las tabs ocupando 4 columnas y el contenido 8.

En pantallas lg o de más píxeles la distribución es igual pero las tabs verticales pasan a ocupar 3 columnas y el contenido 9

De esta forma se da todo el tamaño posible contenido principal mientras los tabs se mantienen visibles, pero sin ocupar más espacio del necesario.

El cambio de aspecto de las pestañas se realiza con 2 grupos de tabs distintas, ocultando un grupo u otro según el tamaño de pantalla.



Documentación Bootstrap

precios.html:

Mediante **table-responsive** Bootstrap adapta el tamaño de las celdas según la pantalla y su contenido, y en pantallas pequeñas si el tamaño total de la tabla es demasiado ancho nos permitirá desplazarnos por ella de forma horizontal.

CINEMANIAC				CINEMANIAC				Cartelera	Cines	Precios	
PRECIOS para socios				PRECIOS para socios							
Nombre	Validez	Taquilla	Online								
Día Laboral	Lunes a Jueves	9.80€	8.80€								
Fin de semana y festivos	Viernes a Domingo	10.60€	9.60€								
Abono Mensual	Todos los días	50€	No disponible								
PRECIOS para no socios				PRECIOS para no socios							
Nombre	Validez	Taquilla	Online								
Día Laboral	Lunes a Jueves	9.80€	8.80€								
Fin de semana y festivos	Viernes a Domingo	10.60€	9.60€								
Día del espectador	Miercoles no festivos	5.50€	5.40€								
Mayores de 65 años	Todos los días excepto festivos	7.00€	No disponible								
Estudiantes y carnet joven	Lunes a Jueves	7.00€	No disponible								
				FAQ		Hazte Socio		Aviso legal			
				Precios		Eventos		Política de privacidad			
				Quiénes Somos		Salas premium		Descarga la app			

5. Componentes adicionales

Accordion: Se utiliza para organizar el contenido de cada tab de forma que solo se vea una sección a la vez, y el usuario pueda desplegar o colapsar el contenido que elija.

Al abrir una sección las otras se cierran, esto se realiza utilizando `data-bs-parent`. En la sección de preguntas frecuentes se utilizan dentro de cada tab para mostrar y ocultar distintas preguntas.

.accordion define el contenedor principal de Accordion

id="accordion1" son identificadores únicos y necesarios para su funcionamiento

.accordion-item define cada sección del Accordion, dentro está el encabezado y contenido

.accordion-header se utiliza para el encabezado de cada sección y también necesita de un id único

.accordion-button nos sirve para expandir o colapsar el contenido asociado y cambia de aspecto si el contenido se esta mostrando o no

data-bs-toggle da función al botón, y **data-bs-target** define un identificador único para el contenido asociado al botón

aria-expanded="true" Nos sirve para mostrar un elemento expandido por defecto, en este caso el primer botón de acordeón

.accordion-collapse se utiliza para la sección del contenido del acordeón

.collapse para las animaciones de expansión y colapso, y **show** permite tener un contenido visible por defecto, en este caso de nuevo en el primer item

.accordion body almacena el contenido de cada sección, en este caso texto.

tabalist: Dentro de esta página encontramos una sección para la compra de entradas para cada película disponible en la cartelera.

Cada película está creada de un contenedor formado por una imagen y una “tab-content”. La clase “tabs” de Bootstrap nos permite poner y cambiar la información de una página usando un pequeño menú de navegación.

Dentro del “tab-content” aparecerán cuatro pestañas distintas “tab-pane”. La primera contiene: El título de la película, la fecha de estreno, el director y una descripción de la trama de la película. Las otras pestañas “tab-pane” contienen las horas en las que se proyecta la película. Al seleccionar una hora aparece una selección de asientos y puedes comprar la cantidad que desees.

Cada “tab-pane” contiene un role para especificar el cual de las secciones es el “tabpanel” y cuál es el “tablist”.

Cada botón del “tablist” tiene una clase llamada “nav-link” que sirve para indicar cual está seleccionado actualmente.

Modales: Las modales son los pop-ups que podemos ver en la pestaña de cine. Estos funcionan de modo que al hacer click al botón de ver localización, el div con la clase modal-content se despliega y muestra la información que había oculta, en forma de pop up en el centro de la pantalla. Al clicar en el botón de X, este se cierra y la página vuelve a su forma original.

Navbar responsive: El navbar que hemos encontrado en los elementos de Bootstrap aparece como una barra horizontal cuando estamos en dispositivos grandes, y se encoge y esconde tras el botón de las 3 rayas horizontales cuando el dispositivo es de tamaño pequeño. Contiene, además, 2 menús dropdown que desplegarán páginas a las que se puede acceder. En dispositivos pequeños, la disposición del menú cambia de horizontal a vertical.

Carousel: El carousel de la página principal es otro elemento que pudimos encontrar en la documentación de Bootstrap. Este posee 2 botones a la izquierda y a la derecha que al ser clickados o pulsados pasarán a la siguiente imagen. Además, en el caso de no pulsar ninguno, pasarán automáticamente a la siguiente imagen al cabo de unos segundos. A estas imágenes también les acompaña un título con el nombre de la película y un botón para acceder a la página de películas para poder comprar una entrada.

6. Uso de Sass

- *Uso de variables*

Hemos utilizado variables principalmente para gestionar los valores de los colores que a aplicarse a los distintos elementos de la página web. Al tener un documento donde se encuentran las variables, es más sencillo cambiar los estilos globalmente en caso de querer hacerlo sin la necesidad de buscar cada referencia de un color o forma en el código.

En cuanto a las variables de color, hemos hecho una distinción entre las que se aplican en temas claros y aquellas que se aplican en temas oscuros de manera que se pueda cambiar entre ambos de forma rápida y sencilla.

```
variables.scss > ...  
  
// colores tema oscuro  
$primary-color-dark: #1FABAB;  
$secondary-color-dark: #def5ff;  
$tertiary-color-dark: #212529;  
$background-color-dark: #212529;  
$text-color-dark: #ffffff;  
$hover-color-dark: #35d7df;  
  
// colores tema claro  
$primary-color-light: #0d6efd;  
$secondary-color-light: #bae8f9;  
$tertiary-color-light: #9eabb9;  
$background-color-light: #b4dada;  
$text-color-light: #212529;  
$hover-color-light: #1e9e9e;  
  
//invariables  
$black-bg: black;  
$border-radius-value: 5px;
```

- Utilización de Mixins

Hemos usado mixins para hacer el código de estilos más modulable y que pueda ser reutilizable (como si fuera un método de java) evitando sobre todo tanta repetición de propiedades. Tenemos los siguientes:

- **Dimensiones:** sirve para establecer las dimensiones de los elementos pasándole como parámetros la anchura y la altura.

```
@mixin dimensiones($width: auto, $height: auto) {  
  width: $width;  
  height: $height;  
}
```

- **Dimensiones máximas:** mismo funcionamiento que el anterior, pero este limita el ancho y alto máximo de un elemento.

```
@mixin maxDimensiones($max-width: auto, $max-height: auto){  
  max-width: $max-width;  
  max-height: $max-height;  
}
```

- **Estilos para las imágenes:** contiene anidado los dos mixins anteriores para controlar las dimensiones de las imágenes. Además, podremos controlar la posición de las mismas.

```
@mixin image-style($height, $max-height, $object-position: 100% 80%){  
  @include dimensiones(auto, $height);  
  @include maxDimensiones(auto, $max-height);  
  object-fit: cover;  
  object-position: $object-position;  
}
```

- **Barra de scroll personalizada:** mediante el uso de las variables globales podremos definir sus colores y forma. También podremos definir si anchura.

```
@mixin scrollbar($width, $bg-color, $thumb-color){
  &::-webkit-scrollbar {
    width: $width;
    border-radius: $border-radius-value;
    background-color: $bg-color;
  }

  &::-webkit-scrollbar-thumb {
    background: $thumb-color;
    border-radius: $border-radius-value;
  }
}
```

- **Media Queries:** utilizamos un mixin para gestionar las media queries de una forma más rápida, pudiendo así adaptar el diseño a las diferentes pantallas.

```
@mixin deviceSize($size){
  @if $size == 'sm'{
    @media (max-width: 400px){
      @content;
    }
  } @else if $size == 'lg'{
    @media (min-width:1610px){
      @content;
    }
  }
}
```

- **Temas claro y oscuro:** estos mixins nos ayudan a aplicar tanto el tema claro como el tema oscuro que afectan principalmente al fondo del cuerpo de la web, a las letras de menú de navegación y al botón de cambio de tema.
 - *¿Cómo se ha realizado el cambio de tema?*

Documentación Bootstrap

Para ello se han creado variables de color para ambos temas. Posteriormente, se han creado estos dos mixins en los que se van a aplicar dichos colores en función del tema activo y que se van a incluir en el archivo principal scss. Para poder lograr dicho cambio de forma dinámica desde la interfaz de usuario, hemos utilizado un script de JavaScript. En este script, se recoge el id del botón que genera el cambio y se le añade un EventListener de tipo “click” de manera que, al pulsarlo y tras comprobar qué estilo tiene el body (mediante la clase light-theme o dark-theme), esta se cambia por su opuesta, aplicándose los estilos correspondientes.

```
// Mixin para aplicar el tema claro
@mixin light-theme {
  background-color: $background-color-light;
  color: $text-color-light;

  // Otros estilos del tema claro
  a {
    color: $primary-color-light;
  }

  button {
    background-color: $primary-color-light;
    color: $text-color-light;

    &:hover {
      background-color: $hover-color-light;
    }
  }
}

const themeToggle = document.getElementById("theme-toggle");

themeToggle.addEventListener("click", () => {
  const body = document.body;

  // Cambiar entre light y dark theme body
  if (body.classList.contains("light-theme")) {
    body.classList.remove("light-theme");
    body.classList.add("dark-theme");
  } else {
    body.classList.remove("dark-theme");
    body.classList.add("light-theme");
  }
});
```

```
// Mixin para aplicar el tema oscuro
@mixin dark-theme {
  background-color: $background-color-dark;
  color: $text-color-dark;

  // Otros estilos del tema oscuro
  a {
    color: $primary-color-dark;
  }

  button {
    background-color: $primary-color-dark;
    color: $text-color-dark;

    &:hover {
      background-color: $hover-color-dark;
    }
  }
}
```

- **Modularización del código**

Hemos creado varios archivos scss que afectan a los diferentes elementos de la página web con el objetivo de dividir por regiones la aplicación de estilos de manera que, en caso de necesitar realizar cambios en alguna de las partes, sea más fácil, rápido y visual acceder al código. En estos archivos se van importando tanto las variables como los mixins para poder utilizarlos. Finalmente, todos los archivos scss se importan en un main.scss para generar así toda la hoja de estilos (css).

- **general.scss:** Este archivo contiene los estilos generales que se aplican a toda la página web donde se gestionan clases comunes como márgenes y el uso de temas que explicaremos más adelante. Se aplican las variables, y mixins de los distintos temas.
- **navbar.scss:** En este archivo se gestionan los estilos del navbar (barra de navegación). Se aplican los mixins de dimensiones y márgenes.
- **imagenes.scss:** Archivo que se encarga de los estilos relacionados con las imágenes para que se ajusten correctamente al diseño. Se aplican mixins para gestionar las dimensiones (dimensiones máximas también) de las imágenes
- **carrousel.scss:** Aquí se definen los estilos del carrusel de imágenes que aparece en la pantalla principal. Se aplican mixins para gestionar las dimensiones máximas y la estética de las imágenes.
- **devicesSize.scss:** Este archivo contiene los media queries que se utilizan para adaptar el diseño de la web a los diferentes tamaños de pantalla.
- **main.scss:** Es el archivo principal que importa todos los archivos SCSS anteriores y los compila en un único archivo CSS.