En vez de añadir enlaces a las fuentes, descargué la fuente Fira Sans de Google Fonts y la añadí a mi carpeta raíz para tenerla siempre disponible en mi proyecto, evitando riesgos de que, si un día Google cambia el link o elimina la fuente, que la letra deseada en mi proyecto también cambie. Para ello he usado @font-face, y he creado el mismo estilo para diferentes grosores que vamos a usar en el ejercicio, aunque, también, se pudo haber definido un nombre de estilo diferente por cada grosor.

Como en la entrega anterior, he definido un reseteador de márgenes interiores (padding) y exteriores (margin), así como, he ocultado, por defecto, los puntos de las listas que vaya a crear en mis páginas. También, he ocultado la línea que aparece para los enlaces, y definí el color blanco en vez del azul. Ya que en todo el ejercicio y en todas las páginas se va a usar la misma fuente de letra, la puse en mi reseteador y poder ahorrar código en el resto del proyecto, modificando donde corresponda, solamente el tamaño y grosor de la letra. Al igual que en el anterior ejercicio, sigo la metodología BEM para definir las clases y subclases para que se vea claro directamente desde el CSS en lo que se está trabajando.

He usado las propiedades background-image para seleccionar la imagen de fondo, size define el tamaño donde 100% es el ancho y 1500px es el alto que fui ajustando para dejarlo como se aprecia en la muestra. Background-attachment: fixed se usa para que la imagen se mantenga fija, aunque se haga scroll en la página y se siga mostrando. Background-repeat puse para que no se repita a modo mosaico, y position, para centrarla.

En index, tanto en el header como en el footer, he creado una clase padre donde definí el tamaño de la caja, y después creé una clase hija donde pude usar display-flex y posicionar los elementos acorde a las necesidades.

En el header, conseguí que la caja morada se mantuviera dentro de la imagen de la guitarra a base de definir los márgenes y padding. Lo más complicado ha sido definir el pseudoelemento para crear la línea debajo del texto que representa la página en la que nos encontramos actualmente. Tal como ya hice en la PEC anterior, establecí al elemento padre con position relative y al hijo con absolute, y ahí definí los parámetros para crear la línea que buscaba. Como ya tenía preestablecidos efectos para los enlaces, he tenido que añadir especificidad para conseguir cambiar los efectos solamente en el elemento seleccionado. Por otro lado, tras conseguir ubicar la imagen de la guitarra, mencionar que se puede jugar con los padding para aumentar el espacio y que se vea más contenido de la imagen. Mantuve las mismas clases en los header del resto de páginas y solo tuve que cambiar la etiqueta donde se encontraba la palabra de la lista del navegador de la página en la que nos encontrábamos, pero los estilos se mantenían iguales.

Acto seguido, continué trabajando con el footer, para tener hechos todos los elementos comunes de las páginas. Creé clases que fueran comunes para las demás páginas, ya que los estilos son los mismos. Creé una caja padre y dos hijas, y apliqué los estilos necesarios para cuadrar los elementos. Box-sizing: border-box fue fundamental para establecer el margen que necesitaba del lado izquierdo sin que el tamaño del footer excediera en horizontal el tamaño total de la pantalla.

En el página de tickets, he creado el formulario. Dejé el atributo action=”” vacío en la etiqueta <form>, y por tanto, los datos se envían a la misma página actual en la que se encuentra el formulario. Lo suyo es dejar vacío el campo y manejar los datos con un lenguaje de programación como Javascript. El atributo action debe estar presente para que se procesen los datos correctamente.

De cara a la accesibilidad y usabilidad de la página, se puede evitar el uso de atributos for=”” y id=”” para asociar ambas etiquetas, ya que la asociación se logra de manera implícita envolvendo los input dentro de las etiquetas label, asocié explícitamente estos valores ya que necesitaba separar las etiquetas label para separlos en cajas diferentes y poder posicionar el label encima del input. Añadí el atributo placeholder donde corresponde, para aclarar al usuario lo que tiene que escribir en cada campo, tal como indica la imagen ejemplo. Por otro lado, dentro de la etiqueta <select>, el atributo name es necesario, ya que esa información se utiliza al tramitar el campo y valor del lado backend, id por otro lado, no es necesario, ya que al igual que en las etiquetas input, se asocia con label, en este caso, label ya engloba select y option, pero igualmente añadí las etiquetas for y id. Dentro de la etiqueta option, el atributo value=”” tampoco es necesario, ya que, si no se especifica este atributo, el contenido visible de la etiqueta option se enviará como valor por defecto, pero lo añadí para facilitar el manejo de datos.