COLORES:

Para la selección de los colores utilizados en la aplicación web se escogió una paleta de colores proporcionada por un sitio web llamado Color Hunt. Este sitió web proporciona una gran variedad de paletas de colores por lo que resulta de gran ayuda a la hora de empezar a tener en una idea sobre el diseño gráfico de la aplicación web.

Color Hunt se destaca además de por tener una gran variedad de paletas, por ser muy sencilla de utilizar. No obliga a registrarse y solo se necesita buscar entre el gran número de paletas que tiene aquella que más se ajuste a lo que se necesite.

Todo esto, sumado al conocimiento previo que se tenía sobre Color Hunt hizo que se decidiera utilizar sus paletas para el desarrollo del presente proyecto.

A la hora de buscar la paleta se tenía claro que se querían utilizar colores que no fueran muy claros. La utilización de colores claros en entornos poco iluminados provoca una mayor fatiga y desgaste a la hora de observar una pantalla, por ello, y dado que no se quiere que los usuarios puedan llegar a sentirse molestos por esto, se decidió utilizar una paleta que contara con diferentes tonos de azul y verde.

Estos colores, combinan a la perfección y permiten utilizar tonos claros sin que estos lleguen a ser molestos o perjudiciales para los usuarios de la aplicación. La paleta seleccionada con sus respectivos valores hexadecimales se puede observar en la figura NUMEROCOLORES.

De los colores seleccionados, se decidieron usar aquellos con tonos más claros para botones o pequeños iconos de la aplicación y aquellos con tonos más oscuros para el resto de las vistas. Además de los colores presentes en la paleta se decidió usar el rojo para la representación de acciones que consisten principalmente en eliminar, ya que dicho color resulta muy útil a la hora de alertar al usuario del peligro que conlleva realizar dicha acción.

Para poder aplicar dichos colores lo único que se tiene que hacer es copiar el valor hexadecimal de cada color en los archivos css asociados a las vistas.

ICONOS:

La correcta selección de los iconos que se van a utilizar en un sitio web es muy importante. Los iconos ayudan a que los usuarios tengan claro que representa cada cosa sin la necesidad de que se les tenga que poner un texto, por ello, es importante que puedan entender qué tipo de icono es, qué representa y a qué va asociado de la manera más rápida y sencilla posible.

Para la selección de los iconos se decidió utilizar Font Awesome, que cuenta con una versión de pago y otra gratuita. Al principio se iban a utilizar imágenes para la representación de dichos iconos, sin embargo, tras investigar un poco y ver diversos proyectos y páginas web, se pudo observar cómo en su mayoría utilizaban los iconos creados proporcionados Font Awesome.

Font Awesome no presenta dependencias con ninguna otra librería, solo necesita que los navegadores cuenten con el soporte @font-face lo que lo hace compatible con la mayoría de los navegadores. Cuenta con un gran número de iconos diferentes disponibles e intuitivos y gracias a la utilización de dichos iconos, el número de imágenes que se utilizan a la hora de desarrollar la aplicación se ven disminuido, lo que afecta positivamente a la velocidad de carga del sitio web (utiliza para crear los iconos una sustitución llevada a cabo por vectores).

Todo lo comentado anteriormente sumado a su licencia libre y a su fácil uso, hizo que se decidiera utilizar sus iconos a la hora de representar las vistas de la aplicación web Wishten.

Para poder utilizar sus iconos primero hay que registrarse y a continuación usar un kit. Un kit es un paquete donde van a estar guardados todos los iconos que se utilizan en la aplicación web. Para usar el kit en proyecto, solo se debe copiar y pegar el .js que te proporciona el sitio web en la sección correspondiente. Esto se puede ver en la figura NUMEROFONTAWESOME.

Una vez se ha obtenido el kit, lo único que queda por hacer es buscar el icono que se desee, copiarlo (Font Awesome proporciona la línea de código) y pegarlo en la vista que se desee utilizar. Hay que tener claro cuál es el nombre que tiene dicho icono para poder modificar la forma en la que este se ve o coloca desde el correspondiente css asociado a la vista en la que se encuentre. La línea que de código que te proporciona para representar cada icono se puede ver en la figura NUMEROICONO.

Si bien dicho icono se podría utilizar junto al elemento <span> en HTML5, cómo la línea de código se te proporciona directamente, se decidió dejar el uso del icono junto con el elemento <i>. Se puede cambiar dicha línea de código en función del entorno en el que se vaya a utilizar el icono, ya que Font Awesome ofrece también las opciones de react o vue.js, sin embargo, como para el desarrollo de Wishten se utiliza HTML para representar las vista juntos a css no es necesario tocar nada.

Estilo:

Para realizar las vistas de la página web se decidió desde un principio utilizar HTML5 y css. Se pensó utilizar Bootstrap para agilizar el proceso de creación de las vistas y poder centrase en las diversas funcionalidades, sin embargo, al final se no se utilizó por diversos motivos.

Uno de los principales motivos por los que no se utilizó Boostrap es debido a una de sus mejores características, los componentes y estilos predefinidos. Bootstrap es una toolkit muy utilizada que ofrece una gran cantidad de componentes, y estilos listos para usar, los cuales hubieran agilizado el proceso de desarrollo de Wishten. Sin embargo, el hecho de que dichas plantillas estuvieran presentes en numerosas páginas web hizo que se decidiera dar un toque más único a nuestra web y por ello no utilizar Bootstrap.

Investigando un poco aplicaciones como TikTok, Instagram u otras redes sociales conocidas, se pudo observar cómo todas ellas usaban vistas muy simples con unos pocos botones los cuales se encargan de mostrar mediante ventanas modales la diferente información asociada a ellos (acciones de editar, eliminar, crear, etc.).

Todas las plantillas que vimos de Bootstrap estaban orientadas a mostrar toda la información al usuario, y si bien visualmente era muy atractivo, no era exactamente lo que mejor se adaptaba a este proyecto.

Wishten busca que los usuarios puedan ver videos que les sean útiles para estudiar o encontrar ofertas de empleo o de prácticas en empresas, por lo que, lo que se busca, es que la información que se le muestre al usuario sea mínima y lo más informativa posible, ya que van a pasar el tiempo que pasen en la aplicación viendo el vídeo o la oferta que les interese.

Esta forma de mantener a los usuarios es muy utilizada en las redes sociales, de ahí que se hayan usado estas como ejemplo a seguir de cara a estructurar y definir las vistas de la aplicación.

Si bien por lo comentado anteriormente ya se tenía claro que no se usaría Bootstrap, al ver la utilidad que este representaba a la hora de adaptar las vistas a los diferentes tamaños de pantalla se planteó

Otro de los motivos por los que Bootstrap es tan utilizado es porque facilita mucho la adaptación de las vistas en diferentes tamaños de pantalla. Esto resulta muy útil ya que una aplicación web puede ser usada en un dispositivo móvil, ordenador, Tablet, etc. Sin embargo, la presente aplicación no busca estar orientada a otro dispositivo que no sea un ordenador, eso no quiere decir que no se hayan adaptado las vistas a los demás dispositivos, si no que, al no ser un motivo de tanto peso, la utilidad de Bootstrap para la representación de las vistas de nuestro proyecto disminuía.

Es cierto que todo lo que te ofrece Bootstrap es modificable, pero, dado que no había nada que se adaptara a lo que se buscaba, que ya se tenían conocimientos previos sobre el manejo de HTML5 y css que, al no tener plantillas predefinidas, se contaba con total libertad a la hora de representar las vistas y, los motivos comentados anteriormente, se decidió al final no utilizar Bootstrap.

La representación de las vistas usando HTML5 y css ha sido muy variada en Wishten, sin embargo, hay algunos componentes y estilos comunes a todas ellas y que son importantes de explicar para entender su estructura y respresentación.

Para poder representar correctamente los contenidos de cada vista, se utiliza el elemento <div>. Este elemento actúa como si fuera un contendor, por lo que se pueden meter otros elementos dentro de él. Al utilizar dicho elemento se consigue que en el css se puedan colocar de la forma que se desee los elementos que contiene, ya sean botones, iconos, textos, otros contenedores, etc. Si bien la vista de cada contendor es diferente hay varias líneas de código que se repiten en cada css a la hora de estilizar los contenedores y estas son las que se pueden observar en la figura NUMEROFLEX.

Utilizando dichas líneas de código se consigue que se pueda controlar y la disposición y colocación de los elementos dentro del contenedor (se les suele llamar flexbox), que dichos elementos aparezcan uno debajo de otro verticalmente (es decir, en columna), y el lugar del contenedor en el que se encontraran (en este caso en el centro del contenedor). Repitiendo está estructura y especificando otros detalles cómo la distancia entre los elementos, los bordes que estos tienen (redondos, sin bordes, etc.), o el color, se consiguen representar todas las vistas del proyecto.

Otro tema importante que comentar, es la capacidad y el cómo se ha conseguido que la aplicación web se adapte y se vea en una gran variedad de tamaños de pantalla. Para ello se ha utilizado lo que se muestra en la figura NUMEROMEDIA.

El @media es muy útil ya que permite definir un tamaño de pantalla determinado a partir del cual se va a tener que cambiar la disposición y colocación de los elementos de la vista. Cómo se ha comentado anteriormente, casi todos los elementos van dentro de contendedores y estos están dispuesto de una determinada forma (en fila, uno debajo de otro, etc.).

En ocasiones como sucede en Wishten con las ofertas, por ejemplo, hay contenedores que cuentan con un gran número de elementos alineados en fila unos detrás de otros. Para que al representar dichos elementos en una pantalla que cuente con un ancho más pequeño estos no se descoloquen, se superpongan unos a otros o desaparezcan, se decide que, cuando la pantalla tenga el tamaño que se ha fijado en el @media, dichos elementos en vez de estar alineados en fila, se alineen en columna (unos debajo de otros). De esta forma se podrá ver todo el contenido del contenedor sin problemas independientemente del tamaño de la pantalla (también puede suceder que estén en columnas y haya que ponerlos en fila, que haya que reubicar unos sí y otros no, etc.).

Por último, hay que comentar que para la representación de ventanas modales o de acciones cómo puede ser mostrar o no la contraseña, además de para otras funcionalidades se ha utilizado JavaScript, un lenguaje de programación utilizado para lograr páginas web interactivas.

Este lenguaje es muy flexible y ofrece numerosas opciones, pero su uso en Wishten a la hora de ayudar a representar las vistas (tiene también otras funciones diferentes dentro de la aplicación web) se puede resumir en la utilización de eventos para saber cuándo un usuario ha pulsado algo y en función de eso realizar una acción u otra. Para ello, se combina con HTML para coger el elemento en cuestión que se va a supervisar, controlar dicho elemento (utilizando un listener) y , en función del elemento que sea y de lo que se haga con dicho elemento, realizar una acción u otra.