

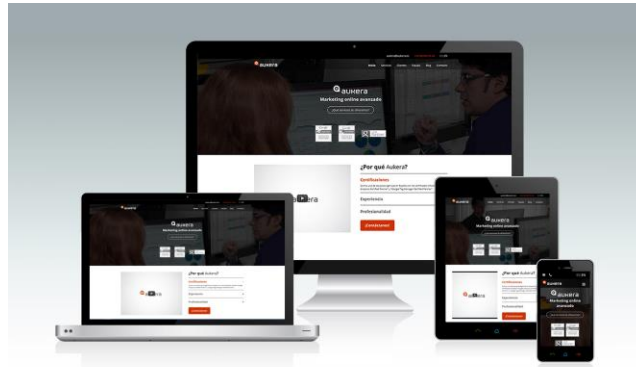
Marco Martin Rabadán

Raúl Rodríguez

PER9755

29/04/20224

Tarea Grupal: Web responsive



Noticias	2
Publicaciones	8
Responsive	10

Introducción

Decidimos hacer una web sobre una empresa de Ciberseguridad, nos parece una buena idea, ya que desde este módulo tenemos una buena visión de las actividades que se realizan y así poder cumplimentar todo el contenido de la web.

Hemos dividido el proyecto en:

- El Home con un index.html
- Quienes somos / empresa (el enlace va a la misma página las dos)
- Área de actuación
- Noticias
- Publicaciones

-Contacto / localización que decidimos ponerlo junto.

Nos hemos dividido el trabajo así:

Víctor Antón realiza la parte de INDEX Y QUIENES SOMOS.

Martín González Varela realiza la parte de ÁREA DE ACTUACIÓN y CONTACTO.

Marco Martin Rabadán realiza la parte de NOTICIAS y PUBLICACIONES.

Por lo que cada una de las partes te la explicamos cada uno con su PDF.

Noticias

En el head de la misma he vinculado el CSS styles.css que dará la forma visual de ver la página web. Así mismo se han vinculado las fuentes de Google fonts para utilizarlas en el diseño.

```
...<link rel="stylesheet" href="noticiaspublicaciones.css">
...<!--google fonts-->
...<link rel="preconnect" href="https://fonts.googleapis.com">
...<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
...<link href="https://fonts.googleapis.com/css2?family=Inconsolata:wght@200..900&family=Je
...<!--google font-->
...<link rel="preconnect" href="https://fonts.googleapis.com">
...<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
...<link href="https://fonts.googleapis.com/css2?family=Inconsolata:wght@200..900&family=Je
    display=swap" rel="stylesheet">
```

Comenzamos por el body en el que se verá el cuerpo de la página, para después utilizar un div con la clase contenedor que agrupa todo el contenido.

El header define la cabecera y después del mismo se ha creado un navegador que contiene los enlaces a las demás páginas

```

<header>
<nav>
<ul class="contenido-ul">
<li class="columna"><a href="#">area de actuación</a>
<li class="columna"><a href="quienes_somos.html">quienes somos</a>
<ul class="columna1">
<li><a href="quienes_somos.html">empresa</li></a>
<li><a href="noticias.html">Noticias</li></a>
<li><a href="publicaciones.html">PUBLICACIONES</li></a>
</ul>
</li>
<li><a href="#">CONTACTO</li></a>
</ul>

```

Se describe a continuación lo realizado en el css del navegador: Comenzamos importando las fuentes de Google fonts para usarlas en todo el documento con @import. Con el selector * damos varios estilos globales con el margin y padding a 0 hacemos que todos los elementos se peguen a la izquierda de la página y con el box-sizing: border-box; hacemos que los elementos se posicionen dentro de la caja y ponemos en mayúsculas toda la página. Ponemos los h1 y p del documento siendo el margin 0 y padding 10px para evitar saltos cuando se haga el responsive, siendo los “formatos” explicados.

```

*{
margin: 0;
padding: 0;
box-sizing: border-box;
text-transform: uppercase;
}

/* Formato */

h1, h2, h3, p{
margin: 0px;
padding: 10px;
}

```

La clase contenedor le ponemos un width: 90%; para que deje un margen, como un marco y centre los demás elementos. El header usa display: flex que sirve para organizar los elementos horizontalmente, así como un justify-content: space-between; para que dejen un espacio uniforme entre ellos.

```

/*contendor*/
.contenedor{
  font-family: "Teko", sans-serif;
  margin: 20px auto;
  width: 90%;
}
header{
  align-items: center;
  justify-content: left;
}

```

Seguimos con el navegador nav el cual se ha realizado un menú desplegable muy sencillo en el que en la parte de Quienes somos es posible desplegarlo y aparecen tres apartados para ir a otras páginas. Es el menú explicado en clase.

```

nav{
  background-color: rgb(57, 99, 55, 0.8);
  width: 100%;
  display: flex;
  justify-content: right;
  z-index: 5; /*Para superponerlo de la pagina principal*/
  flex-wrap: wrap;
}
.contenido-ul{
  display: flex;
  align-items: center;
  justify-content: space-between;
}

```

```

.menu{
  list-style: none;
  padding: 4.5rem;
  transition: all 0.3s ease 0s;
}

li{
  list-style: none;
  padding: 1.2rem;
  transition: all 0.3s ease 0s; /*all se aplica a todas las

```

Se ha utilizado el Display Flex ya explicado para poner en horizontal los elementos y se han centrado a la derecha de la página. Dentro de los li se ha utilizado el elemento transitorio: all 0.3s base 0s; para establecer una transición suave cuando se despliegue el menú, así como se eliminan los puntos de la listad con List-style: none;

```

6  ✓ .columna1{
7    ...display: none;
8    ...position: absolute;
9    ...margin-top: 1vw;
10
11  }
12
13  ✓ .columna1>li{
14    ...width: 100%;
15    ...transform: translateX(-20px);
16    ...border-bottom: 1px solid #fff; /*Para
17    ...text-align: center;
18    ...background-color: #24252a;

```

Con las clases columna 1 definimos las posiciones del desplegable, al ponerle una position:absolute; el cual se coloca en relación con el contenedor padre y la propiedad transform: translateX(-20px); se utiliza para mover el desplegable en su posición X.

Llegamos al contenido principal de la página:

```

.....<div class="contenido-principal">
.....<section id="main-content">
.....<header>
.....<p></p>
.....<h3>Noticias.</h3>
.....<p></p>
.....</header>
.....<article class="blanco discos">
.....<h2>Expertos en ciberseguridad advierten sobre vulne
.....El artículo detalla las preocupaciones expresadas po
.....dispositivos de Internet de las Cosas (IoT).<br>vul
p><ul><li>Que los dispositivos IoT están cada vez má
.....inteligentes y sistemas de salud.</li><li>Los expert

```

```

/*Características del aside*/
aside{
...width: 25%;
...background-color: #f4f4f4;
...padding: 5px;
}

```

```

#main-content{
...width: 75%;
}

```

En el que hemos dividido la misma con un section que con el identificador “#maincontent” ocupa el 75% de la página e irán los artículos y un aside que ocupa el 25% donde podemos ver

el nombre de las ultimas noticias y actualizaciones. También se podría utilizar para poner banners de publicidad. la clase .contenido-principal se ha realizado con un display: flex; y un flex wrap: wrap; el cual te permite controlar cómo se distribuyen los elementos dentro de un contenedor flexible.

```
/*Contenedor general del resto de la pagina*/
✓ .contenido-principal{
  ...display: flex;
  ...flex-wrap: wrap;
  ...z-index: -1;
}
/*Características de los headers y el section*/
✓ section header{
  ...display: flex;
  ...flex-wrap: wrap;
  ...height: 4rem;
}
```

También establecemos características como el alineamiento, la tipografía o el color de letra del section y header.

```
/*Características de los headers y el section*/
✓ .contenido-principal section header{
  ...
  ...align-items: left;
  ...text-align: justify;
  ...font-size: 2vw;
  ...font-family: "Teko", sans-serif;
  ...font-weight: 5vw;
  ...color: ■ whitesmoke;
}
```

Para darle vida he puesto animación con “:hover” en las dos primeras listas del primer artículo dando una sensación inmersiva, sin embargo, en las demás listas utilizo una clase para que no haga esta función o para quitar y poner los “list-type-style”.

```

.nodisc·li:hover{
|···list-style-type:·none;·/*Hace·que·algunas·lista
}
.nodisc·li{
|···background-color:·■whitesmoke;
}
.nada·li:hover{
|···background-color:·■whitesmoke;·/*Hace·que·alg
}
.negro·li:hover{
|···background-color:·□black;·/*Hace·que·algunas·
|···list-style-type:·disc:

```

He realizado en cada artículo un estilo de listas distinto para demostrar que se usan las funcionalidades distintas que he creado.

También he añadido hipervínculos a todas las imágenes llevándote al artículo que tienen relacionado.

Utilizo estas características para dar forma de círculo a algunas fotos:

```

/*Características·imagen*/
.ovalito·{
|···width:·120px;
|···height:·120px;
|···display:·flex;
|···border-radius:·100px;
}

```

Para establecer la posición del copyright fija:

```

168  /*Posicion·del·copyright*/
169  ∨ .copy{
170  |···position:·relative;
171  |···display:·flex;
172  |···left:·800px;
173  |···bottom:·50px;
174  |···width:·100%;
175  |···color:·■#fff;
176  }
177

```

Publicaciones

La primera parte del menú y el aside es común para las dos páginas, aunque varían algunas cosas como la disposición de los artículo o el estilo en el aside.

```
<header>
<nav>
<ul class="contenido-ul">
<li class="columna"><a href="#">area de actuación</a>
<li class="columna"><a href="quienes_somos.html">quienes somos</a>
<ul class="columna1">
<li><a href="quienes_somos.html">empresa</li></a>
<li><a href="noticias.html">Noticias</li></a>
<li><a href="publicaciones.html">PUBLICACIONES</li></a>
</ul>
</li>
<li><a href="#">CONTACTO</li></a>
</ul>
```

```
<header>
<nav>
<ul class="contenido-ul">
<li class="columna"><a href="#">area de actuación</a>
<li class="columna"><a href="quienes_somos.html">quienes somos</a>
<ul class="columna1">
<li><a href="quienes_somos.html">empresa</li></a>
<li><a href="noticias.html">Noticias</li></a>
<li><a href="publicaciones.html">PUBLICACIONES</li></a>
</ul>
</li>
<li><a href="#">CONTACTO</li></a>
</ul>
```

Para esta parte decide utilizar la etiqueta grid para poder hacer una reja, podía haber utilizado también otras características vistas en clase como el float:left,etc, solo que esta opción me resultaba más cómoda:


```

/*Parametros para hacer efecto zoom in en rejá 3x3*
.forma{
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  grid-gap: 40px;
  margin-top: 0px;
}

```

Así me va a hacer una disposición de 3 en cada fila repitiéndose hasta que no haya más.

Con lo visto en clase decidí hacer un muestrario de publicaciones que si clicas te llevan a la publicación correspondiente. Esta tiene una animación de zoom in visto en clase. Al pasar el cursor por encima aparte de hacer un zoom y cambiar la opacidad de los elementos también aparece una descripción de la foto con el fondo del texto en verde.

```

}
.forma: hover div{
  opacity: 0.4;
  transform: scale(1,1);
  transition: 1s;
}
.forma: hover div: hover{
  opacity: 1;
  transform: scale(1.25);
  pointer-events: auto;
}
.forma div img~span{ /*El span adyacente a img se oculte*/
  width: 100%;
  display: block;
  opacity: 0;
  margin: 0;
}

```

Al realizar la animación no conseguí hacer aparecer el enlace e intenté que se mostrase con el z-index por si se encontraba por debajo de todo:

```

}
.forma div img: hover+span{
    opacity: 1;
    pointer-events: auto; /*Con esto intento que se
    z-index: -1;
}
/*Footer*/

```

Responsive

A parte de tener en los contenedores principales la etiqueta `display: flex` y `flex: flexwrap`, al final del documento CSS se encuentra el código `@media` que sirve para que la web cambie dependiendo del dispositivo en el que se reproduzca.

```

/*Para resoluciones de mas de 1300*/
@media screen and (min-width: 1300px) {
    .contenedor {
        width: 90%; /*Reducir el ancho del contenedor principal para que no se extie
    }
    aside {
        width: 20%; /*Ajustar el ancho del aside*/
    }
    #main-content {
        width: 75%; /*Ajustar el ancho del contenido principal*/
    }
    section.forma div img {
        width: 100%; /*Ajustar el ancho de las imágenes en el contenido principal*/
        height: auto; /*Mantener la proporción de las imágenes*/
    }
}

```

En este caso quiero que el aside desaparezca si el ancho es menor a 768 px y la orientacion es en formato horizontal.

```

/*Para resolucion tablet en vertical*/
@media screen and (max-width: 768px) and (orientation: portrait) {
    .contenedor {
        width: 90%; /*Reducir el ancho del contenedor principal para que se adapte al ancho de la
    }
    aside {
        display: none; /*Ocultar el aside en tabletas en posición vertical para ahorrar espacio*/
    }
    #main-content {
        width: 100%; /*Ajustar el ancho del contenido principal para ocupar toda la pantalla*/
    }
}

```

Utilizaremos orientation: portrait cuando sea un formato en vertical y orientation:landscape cuando el formato sea horizontal para así intentar no tener dos css con responsives distintos.

```
....section.forma-div-img{
....|...width:100%;/*Ajustar el ancho de las imágenes en el contenido principal*/
....|...height:auto;/*Mantener la proporción de las imágenes*/
....}
....nav{
....|...overflow-x:hidden;
....}
....ul.li{
....|...overflow-x:hidden;
....}
....}
....}
```

Con overflow-x: hidden ; le indico que todo lo que se salga de su contenedor quede oculto.