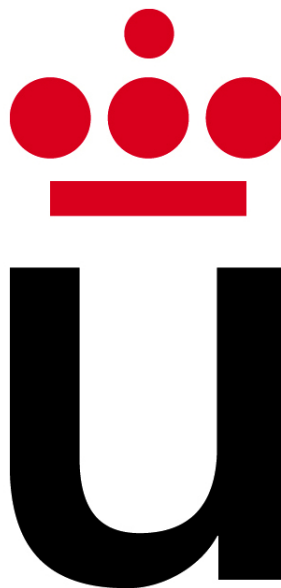


PRÁCTICA 1

Diseño de Aplicaciones Web

PÁGINA WEB INTERACTIVA: HTML, CSS, JQUERY Y BOOTSTRAP

GII+GIS: *Germán Alonso Azcutia*
GIS+MAT: *José Ignacio Escribano Pablos*



MÓSTOLES, 14 DE FEBRERO DE 2015

Índice

1. Página 1: CSS, HTML y jQuery	1
1.1. Organización de los archivos	1
1.2. Aspectos relevantes del HTML	1
1.3. Aspectos relevantes del CSS	1
1.3.1. Propiedades CSS de los div	1
1.3.2. Propiedades CSS para decoración	2
1.4. Imágenes	2
1.5. Menús desplegable con jQuery	2
2. Página 2: CSS, HTML, JS y BOOTSTRAP	2
2.1. Organización de los archivos	2
2.2. Aspectos relevantes del HTML	3
2.3. Aspectos relevantes del CSS	4
2.4. Diseño responsable	4

1. Página 1: CSS, HTML y jQuery

1.1. Organización de los archivos

Lo primero que hicimos fue crear el archivo `index_css.html` donde se incluye la página web y el archivo `style.css` donde se encuentran los estilos de los elementos de dicha web.

La estructura inicial del documento `index_css.html` es la siguiente:

```
<!DOCTYPE html>
<html>
<head>
<title>Result</title>
<link type="text/css" rel="stylesheet" href="style.css" />
</head>
<body>
</body>
</html>
```

Es decir, una estructura básica del documento HTML, en donde lo único destacable es que el archivo está enlazado con una hoja de estilos llamada `style.css`.

1.2. Aspectos relevantes del HTML

Hemos usado una estructura básica del documento HTML, lo más relevante es el uso de una hoja de estilos `style.css` y que para organizar la página web hacemos uso de la etiqueta `<div>` para definir los bloques de contenido y secciones.

La estructura de los `<div>` hechos sería como la siguiente:

```
<div id="breadcrumb">

Home
</div>
```

1.3. Aspectos relevantes del CSS

1.3.1. Propiedades CSS de los div

En las etiquetas `<div>`, las propiedades más utilizadas son `position`, `width` y `height`. Con `Width` y `height` hemos modificado el tamaño de las etiquetas "`<div>`" para adecuarlo así a las necesidades de la web, y con la propiedad `position` y jugando con sus valores `Relative` y `Absolute` conseguimos posicionar los elementos de manera correcta.

Además de estas propiedades, se han utilizado otras como `background-color`, `color`, `text-align`, `left`, `right`, `top` o `bottom`, que nos facilitan colocar de manera correcta los elementos en cada bloque.

Un ejemplo puede ser el siguiente:

```
#logo > h1{
  position: relative;
  top: -40px;
  right: -70px;
```

```

z-index: 1;
font-size: 1.5em;
color: #8f8f8f;
}

```

1.3.2. Propiedades CSS para decoración

Para decorar y que la página web tenga un aspecto similar a la pedida, se han utilizado propiedades como `font-weight`, `font-stretch`, `color`, `border-color`, `border-radius`, `margin-left`, `margin-right` o `text-align`, que nos permiten cambiar el grosor del texto y alinearlos, poner los márgenes necesarios o adaptar los bordes a nuestro gusto para que se parezca lo más posible a la página pedida.

1.4. Imágenes

Las imágenes que aparecen en la web se encuentran todas ellas en la carpeta `img` y para añadirlas a nuestra página usamos distintos métodos, según describimos a continuación:

- Las imágenes que ocupan todo el ancho de nuestra web, o el ancho que queremos usar, se han metido utilizando la propiedad `background-image` de CSS.
- Las imágenes que son mucho más pequeñas que el tamaño que necesitamos, utilizamos la propiedad de CSS `repeat-y` para que dicha imagen se repita a lo largo del eje y.
- Por último, otras imágenes como la del logo, están contenidas en una etiqueta `<div>` donde se inserta la imagen directamente en el HTML, sin tener que estar en las propiedades CSS, como en los casos anteriores.

1.5. Menús desplegable con jQuery

Para agregar los efectos pedidos (`FadeIn` y `DropDown`) en el menú, lo primero que tenemos que hacer es importar los archivos en el HTML: el primero es el archivo jQuery, necesario para utilizar sus propiedades, y el segundo es el script donde se implementa el comportamiento del menú.

```

<script src="js/jquery-2.1.3.min.js"></script>
<script src="js/menu.js"></script>

```

Para realizar el código nos basamos en el método `animate()` de jQuery, que nos permite modificar varias propiedades del CSS al mismo tiempo.

Gracias a esto, cambiamos los valores `height` y `opacity` en los eventos de ratón `mouseenter` y `mouseleave` para conseguir los efectos deseados.

2. Página 2: CSS, HTML, JS y BOOTSTRAP

2.1. Organización de los archivos

En este caso, los archivos creados fueron `index_bootstrap.html` y `custom.css` que personaliza la página con CSS. El head de nuestro archivo queda de la siguiente forma:

```

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width", initial-scale="1.0" />
  <title>Metaheuristics Research Projects</title>
  <script src="js/jquery-2.1.3.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
  <script src="js/collapse.js"></script>
  <link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"/>
  <link type="text/css" rel="stylesheet" href="css/custom.css"/>
</head>

```

Entre los archivos que necesitamos se encuentra un plugin de Bootstrap, Collapse.js¹ que permite que una página se adapte a las páginas de forma responsable escondiendo o mostrando elementos HTML.

Como se puede observar, nuestro CSS debe de estar debajo del de Bootstrap para sobrescribir los estilos del mismo, y que prevalezcan los nuestros.

2.2. Aspectos relevantes del HTML

La estructura de `index_bootstrap.html` es bastante similar a la del HTML anterior, pero con el añadido de utilizar las clases proporcionadas por Bootstrap para que se apliquen sus estilos.

Alguna de estas clases son: `container` y `container-fluid`, para definir la anchura de la página (fija o variable), `dropdown-menu` y `dropdown-toggle` o para el menú desplegable, `col-sm-2`, `col-sm-10` y `col-xs-12` para configurar el ancho de las columnas dentro de una rejilla (clase `row`).

Un ejemplo de como queda es el siguiente:

```

<div class="row" id="content">
  <div class="col-sm-2 hidden-xs" id="vertical_menu">
    <div id="sections"><h3>Sections</h3>
    </div>
    <ul class="nav nav-pills nav-stacked" id="menu">
      <li><a href="#description">Description</a></li>
      <li><a href="#members">Members</a></li>
      <li><a href="#external_researchers">External Researchers</a></li>
      <li><a href="#optimization_problems">Optimization Problems</a></li>
      <li><a href="#black_box_solvers">Black-Box Solvers</a></li>
    </ul>
  </div>
  <div class="col-sm-10" id="content_right">
    ...
  </div>
</div>

```

En donde se puede observar que tenemos una rejilla (`row`) para el contenido dividida en dos columnas, una para el menú vertical (`col-sm-2`) y otra para el contenido (`col-sm-10`). Además, se usa la clase `nav nav-pills nav-stacked`, que nos da un elemento de navegación mediante botones y de forma vertical.

¹<http://getbootstrap.com/javascript/#collapse>

2.3. Aspectos relevantes del CSS

Como el CSS viene dado por Bootstrap, en nuestra hoja de estilo `custom.css` nos dedicamos a que la página se parezca lo máximo posible a la web pedida, respetando el diseño responsable.

Al igual que en el CSS anterior, las propiedades más utilizadas son `background`, `background-color`, `color`, `text-align` y `padding` para organizar los bloques de manera correcta y otras propiedades como `font-weight`, `font-stretch`, `color`, `border-color`, `border-radius`, `margin` o `text-align` para los últimos retoques.

Si hay algo que se pueda destacar es la poca relevancia que tiene ahora la propiedad `position`, ya que no se necesita debido a que usamos el CSS de Bootstrap y el valor `inherit`, para poder quitar los efectos del CSS de Bootstrap que se solapan con los de nuestra hoja de estilos.

2.4. Diseño responsable

Para construir una web responsable, y que el diseño y zoom funcionen bien en dispositivos móviles y tablets, lo más importante es añadir

```
<meta name="viewport" content="width=device-width", initial-scale="1.0" />
```

dentro de la cabecera `<head>`. De esta forma, la página será escalable y tendrá un resultado óptimo para visualizarse en cualquier dispositivo.

Pero además, para que la navegación en dispositivos con un menor tamaño de pantalla sea más agradable, hay que llegar más allá.

En nuestra barra de navegación, hemos añadido la línea

```
<div class="collapse navbar-collapse navbar-ex1-collapse">
```

en el HTML, que agrupa los enlaces de navegación en un menú desplegable.

Además, en nuestro menú vertical, le hemos añadido la clase `hidden-xs`, que hará que desaparezca el menú en dispositivos móviles, pero sea visible tanto en tablets como en ordenadores.

La Figura 1 muestra la representación de `index_bootstrap.html` en un móvil y en una tablet.

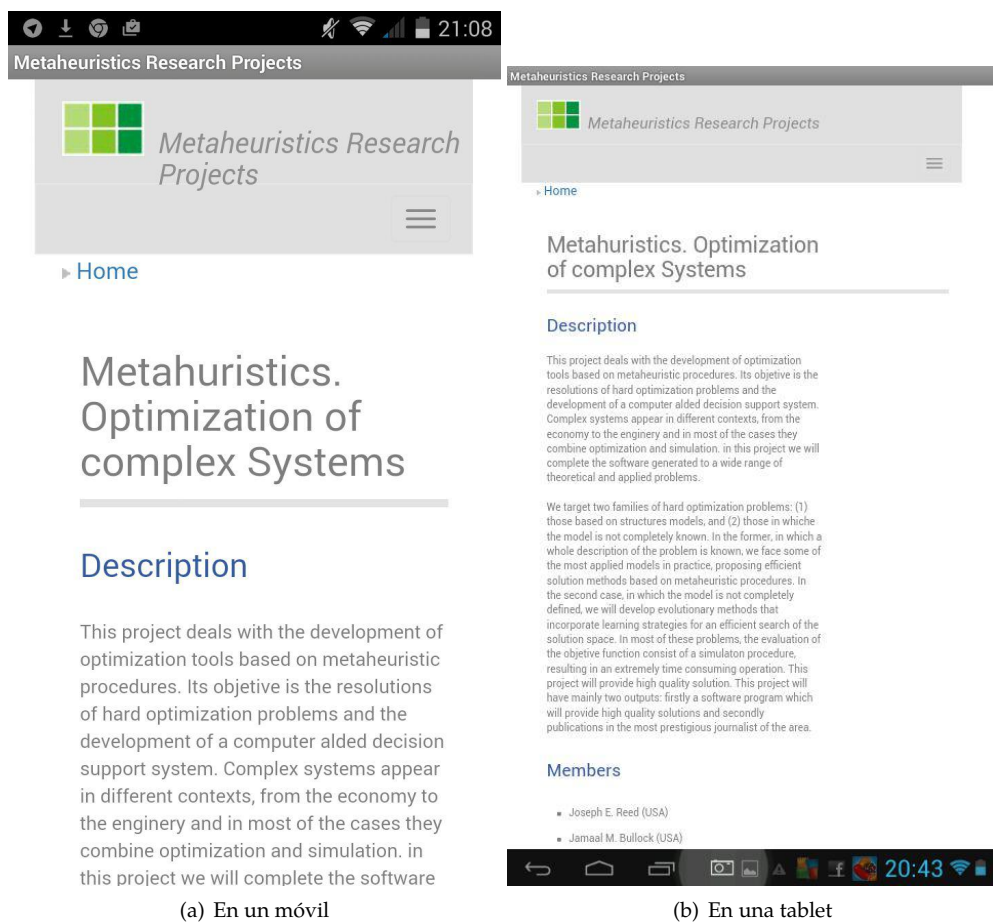


Figura 1: Diseño responsable de index_bootstrap.html