

Folha Prática 8: Responsive Design

OBJECTIVO

Neste laboratório iremos criar, através de *Cascade Style Sheets* (CSS), um **template dinâmico** para o website em desenvolvimento sobre a cidade que escolheu no início do semestre (e.g., Covilhã, Coimbra, Santarém). Este template terá três *layouts* possíveis, consoante a resolução do ecrã seja de um PC, tablet ou smartphone. Esta página servirá posteriormente de *template* para todas as páginas do site a desenvolver.

DESENVOLVIMENTO

Crie uma pasta Lab8 onde inserirá todos os conteúdos a desenvolver neste laboratório.

Deverá criar em CSS um único *template* para todas as páginas do site, para que estas tenham o mesmo aspeto. A escolha dos estilos a aplicar fica ao critério do aluno, sendo parte integrante no processo de avaliação.

O layout do seu site deverá ter 5 elementos, tal como representado na Figura 1:

- cabeçalho
- menu
- conteúdos
- resumo (o quê? onde? como chegar?)
- rodapé.

Em termos de conteúdos, o único elemento que mudará será “conteúdos”, sendo os restantes comuns para todas as páginas. Terá assim 5 ficheiros html, um por tópico, e não usará iframes, ao contrário do Lab7. Deverá incluir toda a informação.

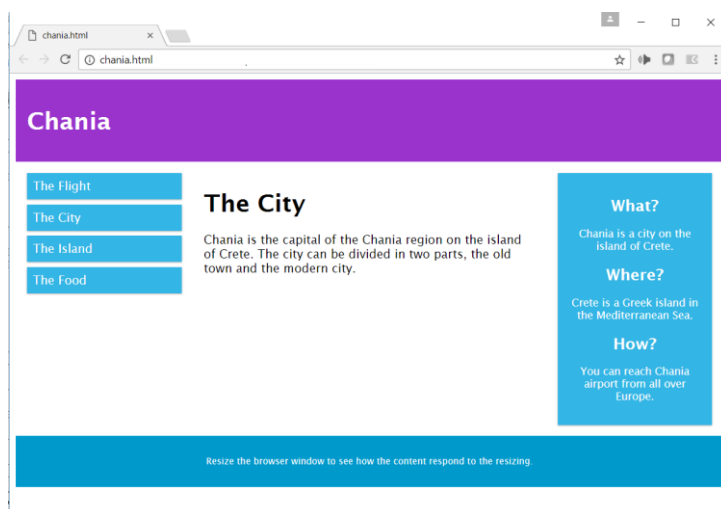


Figura 1. Página com 5 elementos.

Deve recorrer às etiquetas de estruturação do HTML5 para distinguir estes elementos. O conteúdo do *body* deverá ser o mais simples possível, fazendo-se uma utilização sistemática dos CSS (num ficheiro) para especificar todo o tipo de formatações (cores, tamanhos, tipos de letra, bordos, etc.) por forma a dar ao *site* um aspeto sóbrio, funcional e profissional.

1. Diferentes dimensões de ecrãs

Convém que o seu *site* seja visível adequadamente em diversos dispositivos, com características (nomeadamente a resolução) de ecrãs bastante distintas.

A melhor forma de se adaptar a apresentação da página às diferentes resoluções é através do uso de consultas de media (*media queries*), que permitem inquirir do tamanho do ecrã em que o website está a ser visualizado. Leia com atenção a página da W3Schools onde se explica a utilização das *media queries* (http://www.w3schools.com/css/css_rwd_mediaqueries.asp).

O uso de *media queries* pode ser feito de três formas distintas:

1. Colocando, num ficheiro CSS, as diversas regras dentro de uma regra *@media*:

```
@media only screen and (min-width: 36em) {  
  ...  
}
```

2. Colocando o atributo *media* na etiqueta *link*:

```
<link rel="stylesheet" href="styles/EcraGrande.css"  
      media="only screen and (min-width: 36em)">
```

3. Colocando o atributo *media* na etiqueta *style*:

```
<style media="only screen and (min-width: 36em)">  
  ...  
</style>
```

Utilize um ficheiro CSS para especificar as diversas regras.

Deve criar regras que serão aplicadas consoante a resolução dos ecrãs, tal como representado na Figura 2. Considere que um smart-phone tem menos de 600px, um tablet tem mais de 600px e um PC mais de 768px.

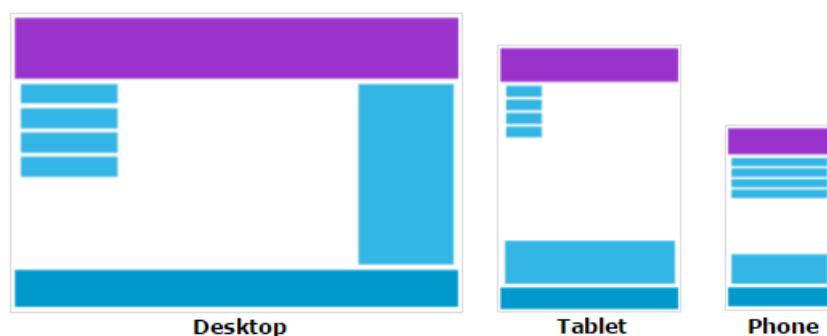


Figura 2. Três layouts possíveis, consoante a resolução do ecrã.

A melhor estratégia será “começar pequeno e ir crescendo”: devemos visualizar/trabalhar primeiro para os ecrãs pequenos e depois formatar para ecrãs maiores. A formatação por

omissão, sem *media queries*, deve criar um *layout* que seja visível nos ecrãs pequenos (decidir qual a informação mais importante e que deverá ser apresentada primeiro, ficando a restante disponível através do uso do *elevador*). Posteriormente, ao introduzir um *media query* para um ecrã maior, pode-se optar por layouts que envolvam diversas colunas, usar padding mais generosos, etc.

Uma das técnicas usadas hoje em dia consiste em dividir o ecrã em colunas de várias larguras possíveis. Deste modo, pode-se escolher a largura que se pretende para um determinado elemento. Veja o exemplo da secção *Another Breakpoint* da página da W3Schools sobre *media queries* (http://www.w3schools.com/css/tryit.asp?filename=tryresponsive_col-s).


Consoante a dimensão do ecrã, uma determinada largura de coluna é escolhida. Por exemplo, o “cabeçalho” ocupará sempre 100% do ecrã, mas o elemento “conteúdo” ocupará 100% da largura num smart-phone, 75% da largura num tablet e num PC 50%, tal como representado na Figura 2. Para diferenciar estes dois últimos casos, o elemento conteúdos define-se com duas larguras (<div class="col-6 col-m-9">), sendo apenas válida a largura que for válida. De facto, as medidas clo- e col-m são definidas uma para o ecrã tablet e outra para o ecrã PC.

Do exemplo http://www.w3schools.com/css/tryit.asp?filename=tryresponsive_col-s adapte o código ao seu website que está a desenvolver. Faça-o apenas para a página de introdução. Não use frameset nem iframes. Faça tudo de raiz com base neste exemplo, adaptando as cores às que escolheu.

Poderá obter mais informação sobre os *media queries* em: http://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Media_queries.

A melhor forma de testar se o seu *layout* se altera em função do tamanho do ecrã, consiste em modificar a dimensão da janela do Browser. Idealmente convém colocar a janela com dimensões pré-estabelecidas, para os ecrãs mais comuns dos dispositivos existentes.

Os *Browsers* mais comuns permitem a modificação da dimensão da janela através ou de opções do menu ou através do uso de extensões que se instalam no *Browser*, expandindo as suas funcionalidades:

- Nas versões mais recentes do Google Chrome e do Opera é possível, através da opção das “Ferramentas de Desenvolvimento” (Ctrl+Shift+I) e escolhendo o segundo ícone () correspondente a “Emulação de Dispositivo”, escolher as dimensões pretendidas para o ecrã. Nas versões antigas do Google Chrome poderá ser utilizada a extensão “Windows Resizer” disponível em: <http://chrome.google.com/webstore/detail/window-resizer/kkelicaakdanhinjdeammilcgefanh>.
- No Firefox pode-se ir ao menu “Tools” → “Developer” → “Responsive Design View”. O *site* será apresentado na janela tem por cima um menu que permitirá escolher as dimensões do ecrã.

Deverá testar o aspeto do seu *site* em diferentes dimensões através do uso destas ferramentas.

2. Imagens diferentes para terminais diferentes

Uma imagem grande pode ficar perfeita num grande ecrã de um PC, mas inútil num smartphone. Por que carregar uma imagem grande quando tem que escalá-la de qualquer maneira? Para reduzir a carga, pode usar media queries para exibir imagens diferentes em dispositivos diferentes.

A *media query* “min-device-width” verifica a largura do dispositivo, em vez da largura do navegador. Deste modo, a imagem não será alterada quando se redimensionar a janela do navegador, ao contrário de “min-width”.

No exemplo http://www.w3schools.com/css/tryit.asp?filename=tryresponsive_image_mediaq veja como é implementado de forma semelhante esta funcionalidade. Crie duas versões da imagem wordcloud da cidade que escolheu, uma pequena e outra grande. Na página que fez no ponto 1., utilize-as consoante o tipo de terminal.

Faça o mesmo para as fotografias, por forma a que, quando seleciona uma determinada fotografia para ser aberta, será aberta uma fotografia maior, mas de tamanho específico para o tamanho do terminal.

3. Menus dropdown

Uma das formas de apresentar sub-menus é recorrendo a menus *dropdown*. Leia a secção http://www.w3schools.com/css/css_dropdowns.asp.

No menu do site que acabou de desenvolver neste laboratório, crie um sub-menu dropdown inspirado em http://www.w3schools.com/css/tryit.asp?filename=trycss_dropdown_navbar assim como na aula teórica. Faça-o para os poemas do seu site, aparecendo os títulos dos três poemas disponíveis.

4. Estilização dos menus

Visualize o resultado do seguinte código fonte HTML.

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
  body{
    background: #b3b300;
    font-family: Helvetica, arial, sans-serif;
  }

  #nav ul {
    float: left;
    list-style-type: none;
    width: 490px;
    height: 44px;
    border-radius: 10px;
    background: #e3e300;
    box-shadow: 10px 10px 30px #333;
    text-shadow: 0 1px 0 white;
```

```

    }

    #nav ul a{
        display: block;
        padding: 10px 20px 10px 20px;
        float: left;
        color: #454545;
        font-size: 20px;
        text-decoration: none;
    }

    #nav ul a:hover{
        display: block;
        padding: 10px 20px 10px 20px;
        float: left;
        color: #ddd;
        text-shadow: 0 1px 0;
        background: #555;
    }
</style>
</head>
<body>
    <div id="nav">
        <ul>
            <li><a href="#">Inicial</a></li>
            <li><a href="#">Empresa</a></li>
            <li><a href="#">Produtos</a></li>
            <li><a href="#">Contacto</a></li>
        </ul>
    </div>
</body>
</html>

```

Entenda os seletores e as propriedades definidas no ficheiro CSS, em particular border-radius e box shadow.

Aplique estas estilizações ao seu site.

SUBMISSÃO DO LABORATÓRIO

- Submeta todos os ficheiros da pasta Lab7 no Moodle.
- Integre também a pasta do laboratório e todos os ficheiros na sua área publica webx, incluindo um link para cada um dos sítios web (fixo e fluido) na pagina index.html da pasta Lab7.
- Submeta também o link para o Lab7.