



Docentes

João Paulo Barraca <jpbarraca@ua.pt>

André Zúquete <andre.zuquete@ua.pt>

Bernardo Cunha <mbc@det.ua.pt>

TEMA 10

Integração de componentes em páginas Web

Objetivos:

- Componentes de *Twitter Bootstrap*.
- Gráficos usando *Highcharts JS*.
- Mapas e elementos georeferenciados.
- Sistemas de gestão de comentários.

10.1 Páginas Web

As páginas Web atuais necessitam de possuir estrutura e estilos muito sofisticados. Cada vez mais a interação faz-se através de interfaces Web e menos através de aplicações que os utilizadores necessitem de instalar no seu computador. Bons exemplos deste ponto são aplicações como o Google Mail[1], Google Docs[2], ou o Microsoft Office 365[3].

Como visto na aula dedicada ao tema de páginas Web, existem components que permitem acelerar o desenvolvimento, fornecendo, de uma maneira simplificada, estrutura e estilo agradável. Estes sistemas permitem ainda melhorar a interação através da inclusão de componentes "pré-feitos" que aumentam a atratividade e funcionalidade de uma página. Isto reforça a ideia que o desenvolvimento de um sistema deve considerar que a existência de componentes já existentes, disponibilizado por outras entidades, não devendo o programador pretender desenvolver toda a programação e estilo. Só em casos

muito específicos se pode tomar esta atitude, sendo que o resultado será, no mínimo, um maior esforço de desenvolvimento e de manutenção do código desenvolvido.

As páginas *Web* são um dos domínios onde mais se utilizam recursos externos, sendo o resultado visível na quantidade e complexidade das páginas disponíveis na *Internet*. Este guião irá abordar alguns componentes que podem ser utilizados para enriquecer uma página *Web*, com foco em componentes dinâmicos para a representação de gráficos, mapas e interação com visitantes.

Para tal, irá fazer-se uso de uma página construída usando o *Twitter Bootstrap*, sendo que serão adicionados componentes de forma a enriquecer as suas funcionalidades.

Relembra-se que uma página mínima, fazendo uso de *Twitter Bootstrap* pode ser criada fazendo inclusão da página de estilos correta. De forma a suportar um modelo de interação mais rico, poderá também ser adicionada um conjunto de *scripts*.

O exemplo que se segue resulta numa página muito minimalista com *Twitter Bootstrap* mas que poderá servir de base para o restante guião:

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8">
    <title>LABI</title>
    <link
      href="//netdna.bootstrapcdn.com/bootstrap/3.0.3/css/bootstrap.min.css"
      rel="stylesheet" />
    <link rel="stylesheet"
      href="//netdna.bootstrapcdn.com/bootstrap/3.0.3/css/bootstrap-theme.min.css" />
    <script src="https://code.jquery.com/jquery-1.10.2.min.js"></script>

  </head>

  <body>
    <div class="container">
      Conteúdo
    </div>

    <script
      src="//netdna.bootstrapcdn.com/bootstrap/3.0.3/js/bootstrap.min.js">
    </script>
  </body>
</html>
```

De realçar que ao contrário do exemplo fornecido numa aula anterior, neste caso a página inclui componentes de Cascading Style Sheets (CSS)[4] (no topo) e de JavaScript

(JS)[5], no fundo. Existem igualmente uma marca `<div>` com a classe `container` que indica que todo o conteúdo deverá estar dentro deste elemento.

Exercício 10.1

Obtenha uma cópia do seu repositório na plataforma *Code.UA* e, lá dentro, crie um diretório com o nome **aula11**. Entre no diretório recém criado e crie uma página com o conteúdo do exemplo anterior. Verifique o funcionamento correto desta página.

A qualquer instante poderá adicionar a página ao repositório, o que deverá fazer no máximo no final da aula.

10.2 Componentes

Uma página *Web* é composta por várias seções, definindo a sua estrutura, sendo que em cada seção existirão componentes variados tais como menus, sub-menus, painéis, imagens, entre outros. A linguagem HyperText Markup Language (HTML)[6] fornece já um grande conjunto de ferramentas que possibilitam a criação de páginas ricas, no entanto por vezes é insuficiente aplicar as marcas de forma isolada, sendo necessário conjugar a marca (HTML), um estilo (CSS) e ações (JS).

Para uma lista completa dos componentes disponíveis para *Twitter Bootstrap*, consultar <http://getbootstrap.com/components>. Neste guião será abordado um número reduzido de componentes.

10.2.1 Menus e Sub-Menus

Os menus e sub-menus são vitais para a navegação de uma página. Tipicamente estes são implementados através do recurso a marcas `` e `li`, significando que no fundo os elementos de um menu são uma lista de items. Depois o estilo aplicado irá atribuir o aspeto típico de um menu.

Usando *Twitter Bootstrap* uma barra de navegação é criada a através de marcas `<div>` contendo depois uma lista de items.

```
<div class="navbar navbar-inverse" role="navigation" >
  <div class="navbar-header" >
    <a class="navbar-brand" href="#" >LABI</a>
  </div>
```

```

<div class="navbar-collapse collapse" >

  <ul class="nav navbar-nav" >
    <li class="active" ><a href="#" >Link</a></li>
    <li><a href="#" >Link</a></li>
  </ul>

  <ul class="nav navbar-nav navbar-right" >
    <li class="active" ><a href="#" >Default</a></li>
  </ul>
</div>
</div>

```

A marca `<div>` inicial possui classes (**navbar** e **navbar-default**) que servem para definir o posicionamento e o aspeto (borda, sombra). Esta marca é seguida de marcas que indicam o conteúdo, respetivamente: `<div>` com classe **navbar-header** indicando o nome da página; uma marca `<div>` com classes para definirem estilo; duas marcas `` para a construção do menu em si. Cada elemento (``) da lista será um elemento do menu.

Exercício 10.2

Adicione uma barra de navegação à sua página. O conteúdo da barra deverá ser colocado dentro da marca com classe **container**.

Também é possível adicionar sub-menus, sendo que para isto é necessário utilizar a classe **dropdown** num elemento do menú. Os elementos do sub-menu são implementados através de uma nova lista.

```

<li class="dropdown" >
  <a href="#" class="dropdown-toggle" data-toggle="dropdown" >Link <b class="caret" ></b>
</a>
  <ul class="dropdown-menu" >
    <li><a href="#" >Link</a></li>
    <li class="divider" ></li>
    <li><a href="#" >Link</a></li>
  </ul>
</li>

```

O resultado deverá ser o apresentado na Figura 10.1:

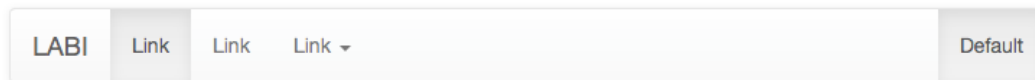


Figura 10.1: Barra de navegação

Exercício 10.3

Adicione à sua página o exemplo acima apresentado.

De forma a navegar na página, é possível criar ligações internas. Estas ligações funcionam através do atributo `id` de cada elemento. Por exemplo, se existir um elemento com `id "topicoA"`, uma marca `<a>` pode enviar o utilizador para esse elemento tendo `#topicoA` no seu atributo `href`. O exemplo seguinte demonstra este caso.

```
<a href="#topicoA" > link para o Topico A </a>
...
<h2 id="topicoA">Este é o Tópico A</h2>
```

Exercício 10.4

Na sua página, adicione marcas `<h2>` para cada um dos tópicos deste guião. Crie um número igual de ligações na barra de navegação, possibilitando ir rapidamente para cada tópico a partir desta barra.

Execute os restantes exercícios por baixo do tópico correspondente.

10.2.2 Popups

Os *Popups* são bastante úteis para apresentar mensagens importantes aos utilizadores. Tipicamente sobrepõem-se à página e necessitam de uma ação explícita para desaparecerem, tal como clicar num botão. Usando *Twitter Bootstrap* os *Popups* são implementados com recurso à classe `modal` aplicado à marca `<div>`.

Tal como demonstrado no exemplo que se segue, existe uma marca `<div>` inicial da classe `modal` que depois possui diversas outras marcas `<div>` com cada um das áreas do elemento. Pode identificar um cabeçalho (`modal-header`), um corpo (`modal-body`), e um

rodapé (**modal-footer**). Note também a existência de duas marca **button**, correspondendo respectivamente ao botão com o símbolo **X** que tipicamente existe no topo de uma janela, e o botão com o texto *Fechar*.

```
<div id="oMeuPopup" class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close"
          data-dismiss="modal" aria-hidden="true">&times;</button>
        <h4 class="modal-title">T\ 'itulo</h4>
      </div>
      <div class="modal-body">
        <p>Conteúdo</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-dismiss="modal">Fechar</button>
      </div>
    </div>
  </div>
</div>
```

Existem vários outros atributos, tais como **data-dismiss**, ou **aria-hidden** que servem para adicionar funcionalidades. O resultado poderá ser o apresentado na Figura 10.2.

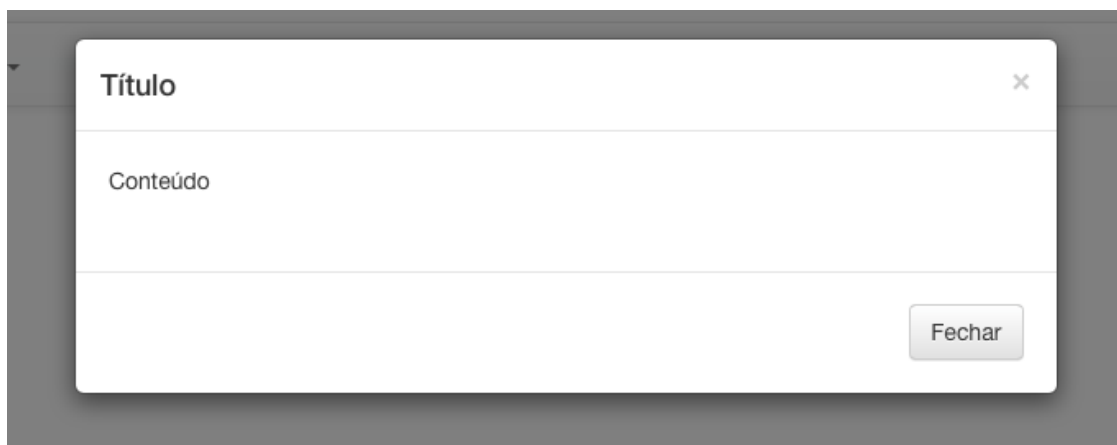


Figura 10.2: *Popup*

Exercício 10.5

Adicione o exemplo acima apresentado. Repare que nada é apresentado. No entanto pode ver no código fonte que isto se deve apenas a um atributo do estilo.

Os *Popups* não são sempre apresentados, no entanto, usando *Twitter Bootstrap* o seu conteúdo permanece na página. Uma maneira simples de o mostrar será através de um botão, usando o atributo **data-target**. O valor deste atributo terá de corresponder ao valor do atributo **id** do *Popup* a apresentar. Significa que o botão em causa activará o *Popup*.

```
<button class="btn btn-primary" data-toggle="modal" data-target="#oMeuPopup">
  Lançar popup
</button>
```

Exercício 10.6

Implemente um exemplo com um botão que ativa um *Popup*, e adicione-o à sua página. Verifique que o *Popup* já é apresentado.

10.3 Gráficos

Uma das limitações que nem a nova versão de HTML suporta é a inclusão de gráficos. No entanto tal é possível através da inclusão de recursos externos de JS, de que a biblioteca *Highcharts JS* é um bom exemplo.

Para adicionar um gráfico existem vários requisitos:

1. Incluir, na marca **head**, os recursos de JS que permitem desenhar gráficos. No caso do *Highcharts JS* é necessária uma linha:

```
<script type="text/javascript"
  src="http://cdnjs.cloudflare.com/ajax/libs/highcharts/3.0.2/highcharts.js"></script>
```

2. Existir um local na página que irá conter o gráfico (HTML). O atributo **id** irá identificar onde o gráfico será desenhado:

```
<div id="grafico-linhas" style="width: 400px; height: 300px;"></div>
```

Neste caso, o gráfico terá 400px por 300px de dimensão.

3. Existir uma configuração do gráfico e os dados (JS) incluída na página:

```
<script type="text/javascript" src="grafico-linhas.js"></script>
```

Sendo que o conteúdo poderá ser:

```
function desenhaGrafico() {
    $("#grafico-linhas").highcharts({
        title: {
            text: "Média de temperaturas",
        },
        xAxis: {
            categories: ["Jan", "Fev", "Mar", "Abr", "Mai", "Jun",
                        "Jul", "Ago", "Set", "Out", "Nov", "Dez"]
        },
        series: [{
            name: "Lisboa",
            data: [7.0, 6.9, 9.5, 14.5, 18.2, 21.5, 25.2, 26.5, 23.3, 18.3, 13.9, 9.6]
        }]
    });
};
```

Repare que neste exemplo, existem dois eixos e duas séries de valores. Respetivamente para as temperaturas de Lisboa.

4. Ser invocada a função de JS que desenha o gráfico

```
<button class="btn btn-primary" onclick="desenhaGrafico()">
    Grafico de Linhas
</button>
```

Ao compor este exemplo, o resultado deverá ser semelhante ao apresentado na Figura 10.3.

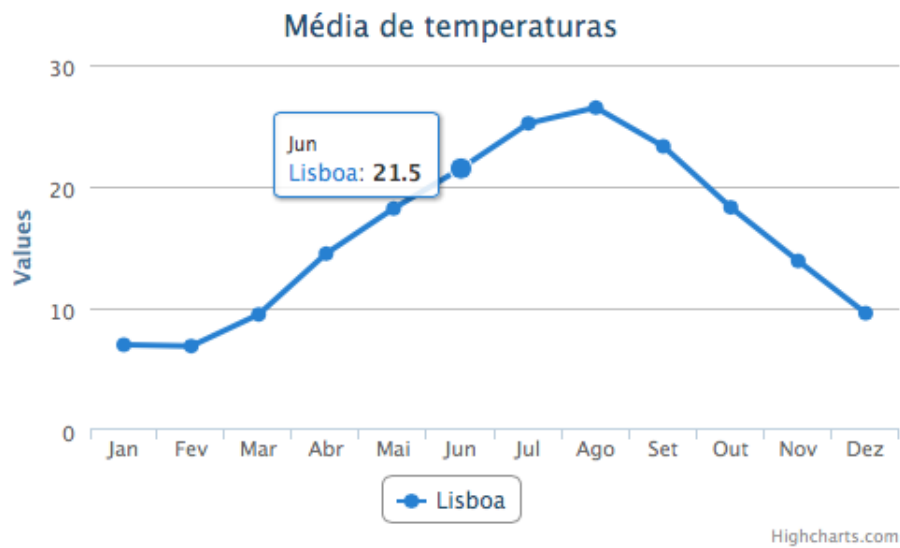


Figura 10.3: Gráfico de linhas

Exercício 10.7

Adicione um gráfico de linhas com as médias de temperatura do sitio onde cresceu. Pode consultar esta informação no endereço <http://www.weatherbase.com>.

Repare que no exemplo anterior apenas são mostrados os dados de Lisboa. No entanto é possível adicionar várias linhas *series*. Para isso basta que o campo **series** possua o seguinte formato:

```
series: [{
  name: "local1",
  data: [... valores ...]
},{
  name: "local2",
  data: [... valores ...]
}]
```

Exercício 10.8

Adicione uma segunda linha com as temperaturas de Aveiro de forma a comparar os dois locais.

A biblioteca *Highcharts JS* possibilita muitos outros tipos de gráficos[7]. Frequentemente, modificar a aparência do gráfico apenas requer se se altere o seu tipo, definido da seguinte forma:

```
$("#grafico").highcharts({  
  chart: {  
    type: "column"  
  },  
  title: {  
    text: "Média de temperaturas",  
  },  
  ...  
});
```

São possíveis vários valores, tais como: **pie**, **column**, **scatter**, **bar**, **area**.

Exercício 10.9

Duplique o ficheiro com a definição do gráfico para outros, representando os vários tipos de gráficos possíveis.

Inclua todos os ficheiros JS na sua página. Adicione um botão e um elemento para desenho de todos os gráficos. Verifique a utilização desta biblioteca.

Exercício 10.10

Inclua um outro tipo de gráfico referido em[8] na secção "*More chart types*"

10.4 Mapas

Outro componente muito comum em páginas *Web* são os mapas, ou outros elementos que apresentam informação geo-referenciada. Os mais populares são provavelmente o

Google Maps[9] e o *OpenStreetMap*[10].

A utilização destes elementos é algo semelhante à dos gráficos, visto que é necessária a inclusão de um recurso externo, a existência de um elemento para onde desenhar o mapa, e algum código JS para personalizar o conteúdo.

Sendo que é possível utilizar serviços como o *Google Maps* diretamente, por vezes é mais vantajoso utilizar bibliotecas que facilitem integração e utilização. Outra vantagem é que é possível utilizar vários mapas através sem modificações relevantes na programação.

Neste caso o recomenda-se a utilização de *Leaflet JS*[11], que adicionar elementos a páginas *Web*, independente do fornecedor de mapas.

Para utilizar *Leaflet JS* é necessário efetuar os seguintes passos:

1. Incluir a definição de estilos e de código do *Leaflet JS*.

```
<link rel="stylesheet" href="http://cdn.leafletjs.com/leaflet-0.7.1/leaflet.css" />
<script src="http://cdn.leafletjs.com/leaflet-0.7.1/leaflet.js"></script>
```

2. Criar um elemento onde colocar o mapa:

```
<div id="oMeuMapa" style="width: 500px; height: 400px"></div>
```

De notar que este elemento necessita de possuir um atributo **id** que irá permitir especificar no código JS qual o elemento específico a utilizar.

3. Finalmente é necessário incluir código JS que crie o mapa.

```
<script type="text/javascript" src="map.js"></script>
```

Sendo que o conteúdo do ficheiro **map.js** poderá ser o seguinte:

```
var map = new L.Map("oMeuMapa", {center: [40.633258,-8.659097],zoom: 15});

var osmUrl="http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png";
var osmAttrib="Map data OpenStreetMap contributors";
var osm = new L.TileLayer(osmUrl, {attribution: osmAttrib});

map.addLayer(osm);
```

Neste caso o valor 40.633258 refere-se à latitude, o valor -8.659097 refere-se à longitude e o valor 15 refere-se à altitude (ou *zoom*). O valor **oMeuMapa** refere-se ao atributo **id** do elemento destino do mapa[11].

[illegible]

Exercício 10.11

Adicione um mapa à sua página. Este mapa deverá mostrar a cidade de Aveiro e a cidade onde mora.

Adicione um mapa à sua página. Este mapa deverá mostrar a cidade de Aveiro e a cidade onde mora.

Adicione um mapa à sua página. Este mapa deverá mostrar a cidade de Aveiro e a cidade onde mora.

Ao mapa criado é possível a definição de funções que são chamadas após a ocorrência de eventos. Um evento útil para o restante guião é saber-se quais as coordenadas de um determinado ponto. Para isto, existe um evento denominado **click** que fornece imediatamente essa informação. A utilização de qualquer evento segue o exemplo seguinte:

```
map.on("click", nome-da-funcao);
```

Sendo que, se se pretender mostrar as coordenadas atuais num elemento da página com `id "coordenadas"`(ex, uma marca ``, pode-se fazer¹:

¹Para a lista completa de eventos, consultar <http://leafletjs.com/reference.html#map-events>

```
function mostraCoordenadas(e){
    var s = document.getElementById("coordenadas")
    s.innerHTML = "Latitude, Longitude = "+e.latlng.lat+", "+e.latlng.lng;
}
```

Uma das utilidades dos mapas é a apresentação de pontos (ou marcadores), indicando a localização de um ponto de interesse. Esta funcionalidade é muito útil para localizar eventos, locais de interesse, ou a morada de empresas e serviços.

Adicionar um ponto a um mapa requer que se defina as coordenadas do ponto, e de seguida que se adicione este ponto ao mapa. No exemplo seguinte é criado um *array* com vários pontos, que são depois adicionados ao mapa:

```
var pontos = [
    L.marker([40.633258, -8.659097]),
    L.marker([40.642729, -8.747899])
];

for(i in pontos) {
    pontos[i].addTo(map);
}
```

Também pode ser adicionado um pequeno texto ao ponto, que será apresentado quando o apontador clicar em cima do ponto. Para isto, é utilizado o método **bindPopup(msg)** em que o argumento **msg** pode ser qualquer texto ou mesmo código HTML:

```
...
L.marker([40.633258, -8.659097]).bindPopup("LABI@DETI")
...
```

Alternativamente, por vezes é útil ajustar a vista de forma a contemplar todos os pontos adicionados. Aqui é necessário criar um grupo com todos os pontos e depois invocar um método que ajusta a vista de forma automática.

```
...
var grupo = new L.featureGroup(pontos);
map.fitBounds(grupo.getBounds());
```

NOTA: Devido a um erro na biblioteca *Leaflet JS*, ao utilizar este método é necessário retirar a opção `center` a quando da inicialização do mapa.

Exercício 10.12

Adicione marcadores que identifiquem vários locais de interesse para si (morada em Aveiro, morada permanente, onde faz desporto/trabalha/pratica um instrumento). Não se esqueça de adicionar textos que descrevam o ponto e de ajustar o mapa aos pontos.

Frequentemente é desejável apresentar imagens diferentes dependendo da natureza do local assinalado. Isto é simples pois os elementos **marker** podem ser criados com uma lista de opções, onde se inclui o ícone²:

```
var iconeUA = L.icon({
    iconUrl: "http://xcoa.av.it.pt/ua.png"
});

...
L.marker([40.633258, -8.659097], {icon: iconeUA}).bindPopup("LABI@DETI")
...
```

Exercício 10.13

Adicione imagens personalizadas aos seus ícones. Na página <http://mapicons.nicolasmollet.com> encontra muitos e variados ícones. Terá de escolher os ícones, e colocá-los junto com a sua página.

Além de marcadores, também é possível adicionar polígonos que sinalizam uma área alargada e não um ponto individual. A metodologia é semelhante à criação de marcadores, mas os polígonos são criados através de um *array* de pontos. A Reitoria de Universidade de Aveiro pode ser indicada através de um polígono, da seguinte forma:

```
var reitoria = L.polygon([
    [40.63102, -8.65793],
    [40.63149, -8.65731],
    [40.63126, -8.65699],
    [40.63078, -8.65759]
```

²Para uma lista completa de opções, consultar <http://leafletjs.com/reference.html#icon>

```
    ], {  
      color: "red"  
    }  
  )  
  reitoria.addTo(map);
```

Exercício 10.14

Adicione um polígono que delimite o DETI.

Ao polígono que criou, use o método **bindPopup(msg)** para mostrar informação do departamento quando se clicar no polígono.

Exercício 10.15

Em vez de utilizar o tipo **polygon**, se utilizar o **polyline** o resultado será uma linha e não um polígono. Utilize estes dois tipos de objetos para indicar onde mora e qual o trajeto para a Universidade de Aveiro.

10.5 Comentários

Um aspeto interessante de se adicionar a um blog ou outra página com visitantes é a capacidade de se incluir comentários. Existem várias maneiras de obter esta funcionalidade, sendo que a mais simples é a adição de um componente externo em JS. Um exemplo é o serviço disponibilizado pelo Disqus[12], que apenas necessita da inclusão de código HTML e JS na página em questão.

Exercício 10.16

De forma a adicionar comentários à sua página, aceda a `http://disqus.com`, e identifique-se com o utilizador **alunolabi** e palavra passe **alunolabi**.

Este serviço é partilhado por todos os alunos pelo que é pedido que não sejam feitas modificações aos dados da conta!

Seguidamente, aceder `http://disqus.com/admin/universalcode/` de forma a obter o código a incluir na página. O valor da variável **disqus_shortcode** deverá ser **labi**.

Exercício 10.17

Envie a página que acabou de criar para o servidor **xcoa.av.it.pt**, no diretório `~/public_html`. Pode utilizar o comando **scp** ou o repositório **GIT** onde tem esta página.

Pode aceder à página através do endereço `http://xcoa.av.it.pt/~pXgY`. Verifique que é possível adicionar comentários.

Exercício 10.18

Aceda novamente ao endereço `http://disqus.com` e verifique as funcionalidades que lhe são apresentadas. Verifique que, por exemplo, ao adicionar comentários, também pode ter informação sobre os seus visitantes e moderar os comentários.

10.6 Para aprofundar

Exercício 10.19

Aceda à página do *Twitter Bootstrap* e implemente uma página local com todos os componentes extra que são apresentados.

Exercício 10.20

Aceda à página do *Fuel UX* (<http://exacttarget.github.io/fuelux/>) e implemente outros componentes que permitem enriquecer páginas *Web*.

Glossário

CSS	Cascading Style Sheets
HTML	HyperText Markup Language
JS	JavaScript

Referências

- [1] Google, *Google mail*, <http://www.gmail.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [2] —, *Google documents*, <http://doc.google.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [3] M. Corporation, *Office - office.com*, <http://office.microsoft.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [4] W3C. (2001). Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification, endereço: <http://www.w3.org/TR/2011/REC-CSS2-20110607/>.
- [5] ECMA International, *Standard ecma-262 - ecmascript language specification*, Padrão, dez. de 1999. endereço: <http://www.ecma-international.org/publications/standards/Ecma-262.htm>.
- [6] W3C. (1999). Html 4.01 specification, endereço: <http://www.w3.org/TR/1999/REC-html401-19991224/>.
- [7] H. JS, *Hicharts demos*, <http://www.highcharts.com/demo/>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [8] H. AS, *Highchars - interactive javascript for your webpage*, <http://www.highcharts.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [9] Google, *Google maps*, <http://maps.google.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [10] OpenStreetMap, *Openstreetmap*, <http://www.openstreetmap.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [11] V. Agafonkin, *Leaflet - a javascript library for mobile-friendly maps*, <http://leafletjs.com>, [Online; acedido em 28 de Novembro de 2014], 2013.
- [12] D. Inc., *Disqus - the web's community of communities*, <http://www.disqus.com>, [Online; acedido em 28 de Novembro de 2014], 2013.