2017

CMP1130M Web Authoring

ASSESSMENT ITEM 2

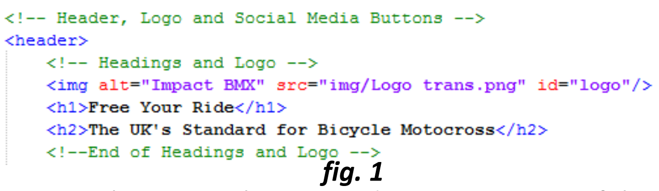
MATTHEW SEELIG SEE16600270

Table of Contents

Best Practice and Standards

Hypertext Markup Language

During the development stage of the website, after the initial concepts, sitemaps and wireframes had been made, it was relatively easy to follow best practice and standards. This was because I was creating the shell of the site and was only using HTML and basic CSS; as taught during my first few Web Authoring lectures. As seen in *fig.1*, indentations are present when appropriate. As well as this, nesting elements follows standards shown to me in the lectures and workshops. The nesting, specifically the nesting of img, h1 tags etc. follows W3C standards for the header tag (W3C, 2014). Also in *fig.1*, you can see that internal program documentation is of the highest standard, and is prevalent throughout each HTML, CSS and JS file.

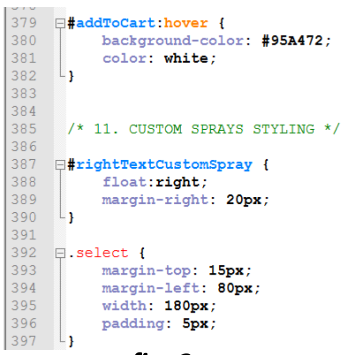


```
<!-- Header, Logo and Social Media Buttons -->
<header>
    <!-- Headings and Logo -->
    <img alt="Impact BMX" src="img/Logo trans.png" id="logo"/>
    <h1>Free Your Ride</h1>
    <h2>The UK's Standard for Bicycle Motocross</h2>
    <!--End of Headings and Logo -->
```

fig. 1

Cascading Style Sheets

Naming conventions are crucial to make the markup and css easy to interpret. I used camelCasing for my classes and id's (as shown in *fig.2*); a *'de facto'* standard set by the IT community. *'de facto'* standards are important because not all practices are outlined in official documents made by organisations such as W3C, which leaves gaps in standards and leaves room for misinterpretation. Community 'de facto' standards are much more flexible, as you normally use whatever is suitable/feels right for the developer (StackOverflow user Bojangles, 2011).



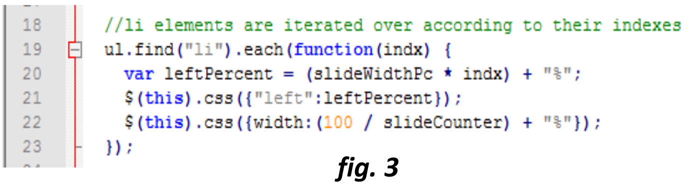
```
#addToCart:hover {
    background-color: #95A472;
    color: white;
}


/* 11. CUSTOM SPRAYS STYLING */

#rightTextCustomSpray {
    float:right;
    margin-right: 20px;
}

.select {
    margin-top: 15px;
    margin-left: 80px;
    width: 180px;
    padding: 5px;
}
```

fig. 2

JavaScript & jQuery

For Javascript, I followed *'de jure'* standards outlined by W3. The number of documents outlining standards for JS on the W3 site (W3C, 2017) is enormous, and was definitely outside of the scope of my project due to lack of expertise and time limitations. Instead, I used the w3schools JS conventions; a simplified version of the former (W3schools, 2017).

Fig.3 exhibits three standards that were followed; internal program documentation, camelCasing for variables, and spaces around operators. (*ibid*)



```
//li elements are iterated over according to their indexes
ul.find("li").each(function(indx) {
  var leftPercent = (slideWidthPc * indx) + "%";
  $(this).css({"left":leftPercent});
  $(this).css({width:(100 / slideCounter) + "%"});
});
```

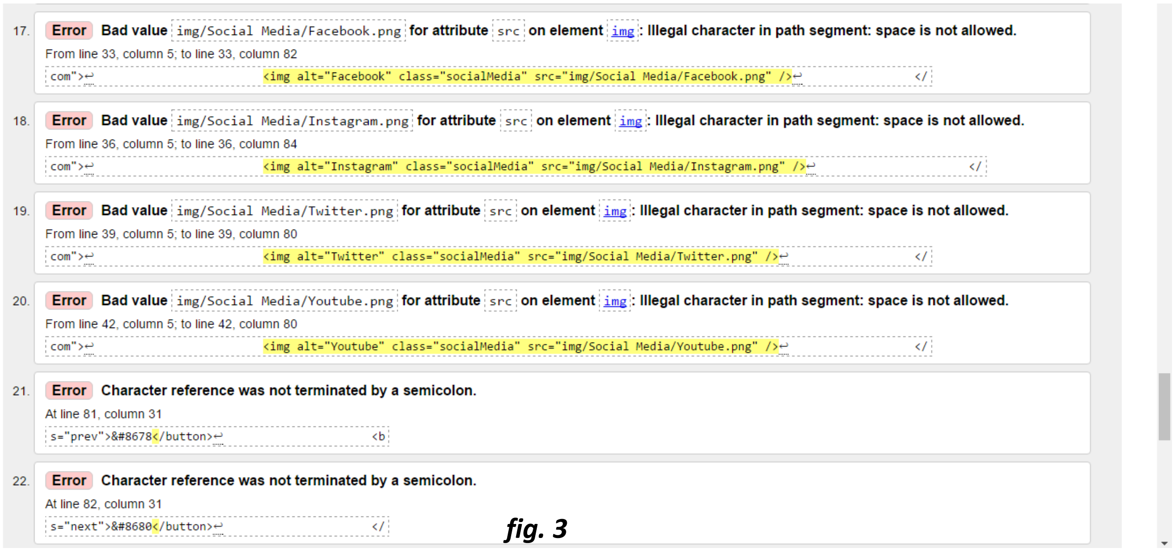
fig. 3

When using jQuery I used the same standards and practices, but used documentation from the jQuery libraries to make sure the correct syntax was being used e.g. the correct parameters. A good example would be the .replaceWith() function; following the jQuery documentation allowed me to use the function correctly (jQuery, 2017).

W3C Validation

During the latter stages of development, it was necessary to use validate my markup to make sure that the standards and practices were being met. W3C validation is perfect for this because it is industry standard, and can automatously check your files without chance of human error.

Figs 4 & 5 display A W3C validation, and the subsequent validation once the errors were fixed.



17. **Error** **Bad value** `img/Social Media/Facebook.png` **for attribute** `src` **on element** `img`**: Illegal character in path segment: space is not allowed.**
From line 33, column 5; to line 33, column 82
```
com">↩                    <img alt="Facebook" class="socialMedia" src="img/Social Media/Facebook.png" />↩                    </
```

18. **Error** **Bad value** `img/Social Media/Instagram.png` **for attribute** `src` **on element** `img`**: Illegal character in path segment: space is not allowed.**
From line 36, column 5; to line 36, column 84
```
com">↩                    <img alt="Instagram" class="socialMedia" src="img/Social Media/Instagram.png" />↩                    </
```

19. **Error** **Bad value** `img/Social Media/Twitter.png` **for attribute** `src` **on element** `img`**: Illegal character in path segment: space is not allowed.**
From line 39, column 5; to line 39, column 80
```
com">↩                    <img alt="Twitter" class="socialMedia" src="img/Social Media/Twitter.png" />↩                    </
```

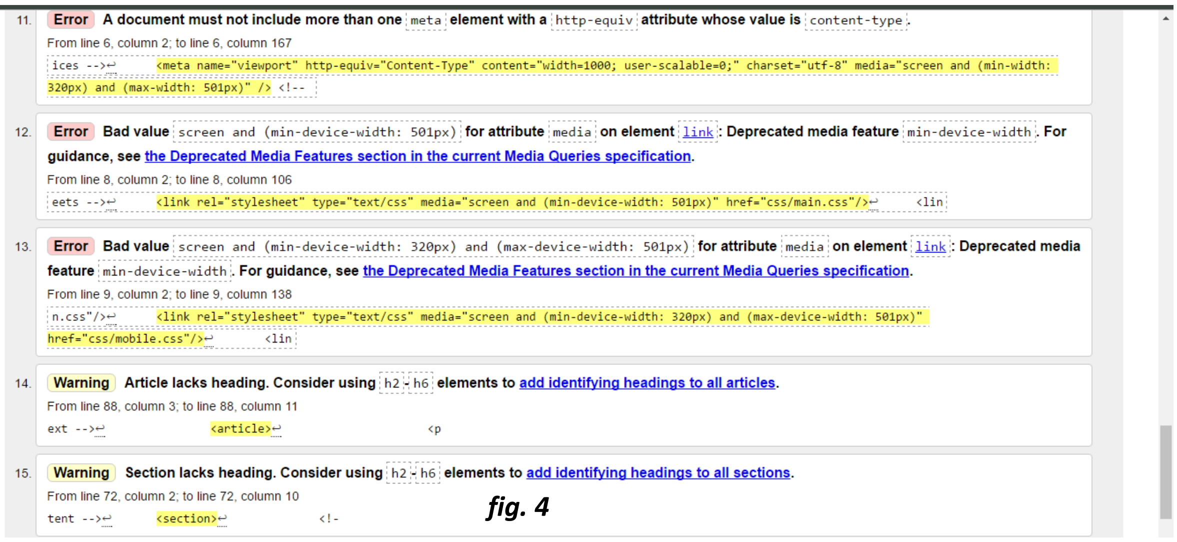
20. **Error** **Bad value** `img/Social Media/Youtube.png` **for attribute** `src` **on element** `img`**: Illegal character in path segment: space is not allowed.**
From line 42, column 5; to line 42, column 80
```
com">↩                    <img alt="Youtube" class="socialMedia" src="img/Social Media/Youtube.png" />↩                    </
```

21. **Error** **Character reference was not terminated by a semicolon.**
At line 81, column 31
```
s="prev">&#8678</button>↩            <b
```

22. **Error** **Character reference was not terminated by a semicolon.**
At line 82, column 31
```
s="next">&#8680</button>↩            </
```

fig. 3

It is worth mentioning that this software isn't perfect, and some errors e.g. <meta> errors, could not be fixed due to their importance/the nature of the error. I will discuss this in further detail later in the report. W3C validation screenshots for all markup can be found in a folder in the supporting material entitled *'W3C Validation'*.



11. **Error** **A document must not include more than one** `meta` **element with a** `http-equiv` **attribute whose value is** `content-type`.
From line 6, column 2; to line 6, column 167
```
ices -->↩        <meta name="viewport" http-equiv="Content-Type" content="width=1000; user-scalable=0;" charset="utf-8" media="screen and (min-width:
320px) and (max-width: 501px)" /> <!--
```

12. **Error** **Bad value** `screen and (min-device-width: 501px)` **for attribute** `media` **on element** `link`**: Deprecated media feature** `min-device-width`**. For guidance, see** the Deprecated Media Features section in the current Media Queries specification.
From line 8, column 2; to line 8, column 106
```
eets -->↩        <link rel="stylesheet" type="text/css" media="screen and (min-device-width: 501px)" href="css/main.css"/>↩      <lin
```

13. **Error** **Bad value** `screen and (min-device-width: 320px) and (max-device-width: 501px)` **for attribute** `media` **on element** `link`**: Deprecated media feature** `min-device-width`**. For guidance, see** the Deprecated Media Features section in the current Media Queries specification.
From line 9, column 2; to line 9, column 138
```
n.css"/>↩        <link rel="stylesheet" type="text/css" media="screen and (min-device-width: 320px) and (max-device-width: 501px)"
href="css/mobile.css"/>↩        <lin
```

14. **Warning** **Article lacks heading. Consider using** `h2`-`h6` **elements to** add identifying headings to all articles.
From line 88, column 3; to line 88, column 11
```
ext -->↩              <article>↩                <p
```

15. **Warning** **Section lacks heading. Consider using** `h2`-`h6` **elements to** add identifying headings to all sections.
From line 72, column 2; to line 72, column 10
```
tent -->↩        <section>↩            <!-
```

fig. 4

As you can see, many errors were fixed, apart from the <meta> errors (discussed later), and the warnings that can be ignored due to their context. W3C CSS validation was also used; screenshots are in the supporting material.

Current Web Standards & Specifications

Strengths

As with any industry, the use of standards is very beneficial because it provides a known level of quality, and allows for innovation and the minimisation of risks and errors. Current web standards & specifications define a common way of exchanging information, which makes web apps more flexible and compatible (romjon, 2017). HTML5 and CSS3 offer a larger variety of customisation than previous versions, such as the canvas element, which was a large part of the customisation of my website *fig. 5.*

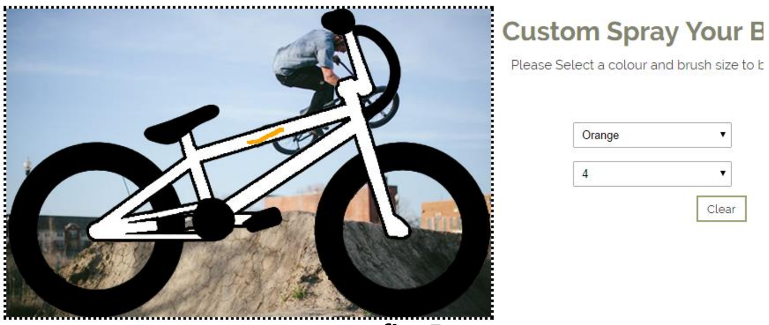


fig. 5

CSS3 also offers styles such as *box-shadow* which is demonstrated in *fig. 6.*

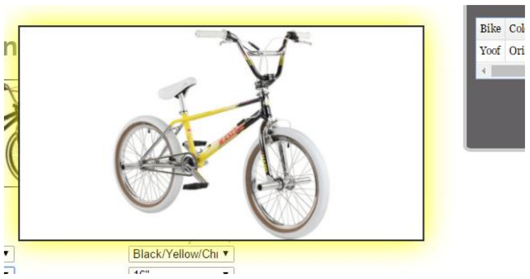
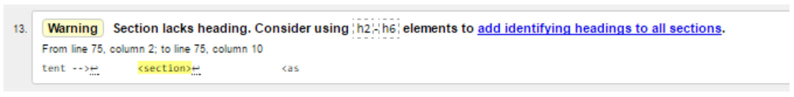


fig. 6

HTML5 has brought other new elements such as header and section, which allow developers to structure their site in a more interpretable manner, rather than using div tags with classes and id's like in previous HTML versions. CSS3 has also introduced RGBA colours and opacity (W3, 2011) which brought simple yet innovative design features to developers.

Weaknesses

With the introduction of a very modern HTML5, the support for older browsers can be an issue. Older versions of popular browsers can struggle or outright won't work with the new markup language. (Katz, 2010). Another issue is the restriction web standards can have on developers. Earlier I mentioned that meta tag errors that were shown were not fixable because I needed media queries in my meta tags. This is just one example of how validation can limit developers. Others warnings/errors that are displayed (*fig.7)* support this argument.



```
13. Warning  Section lacks heading. Consider using h2-h6 elements to add identifying headings to all sections.
From line 75, column 2; to line 75, column 10
tent -->↩       <section>↩            <as
```

fig. 6

There are also some known bugs in HTML5 and CSS3, one of which I came across myself. CSS style *box-shadow* produces a 1px white gap around its contents which cannot be removed without creating several more styles and elements (DaveE, 2014). This is displayed in *fig.7.*

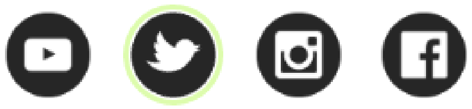


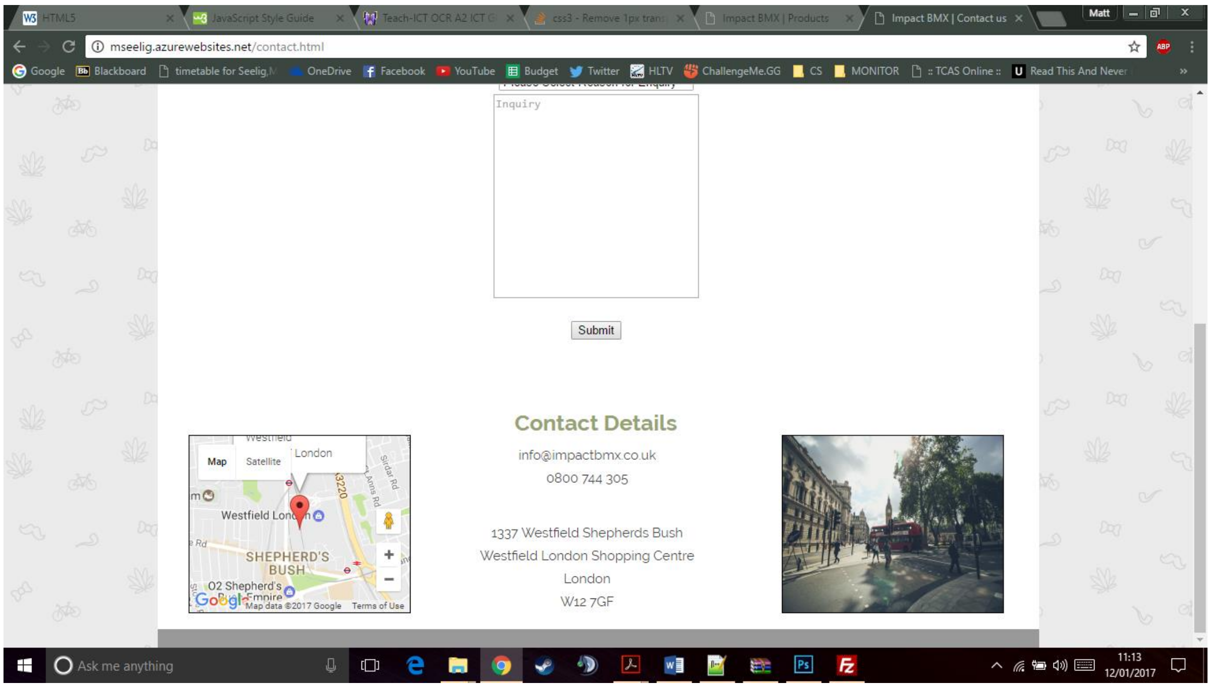
fig. 7

Interoperability

Below are examples of browser support

Chrome

Geo location working in chrome through use of *-webkit-*



Contact Details

info@impactbmx.co.uk

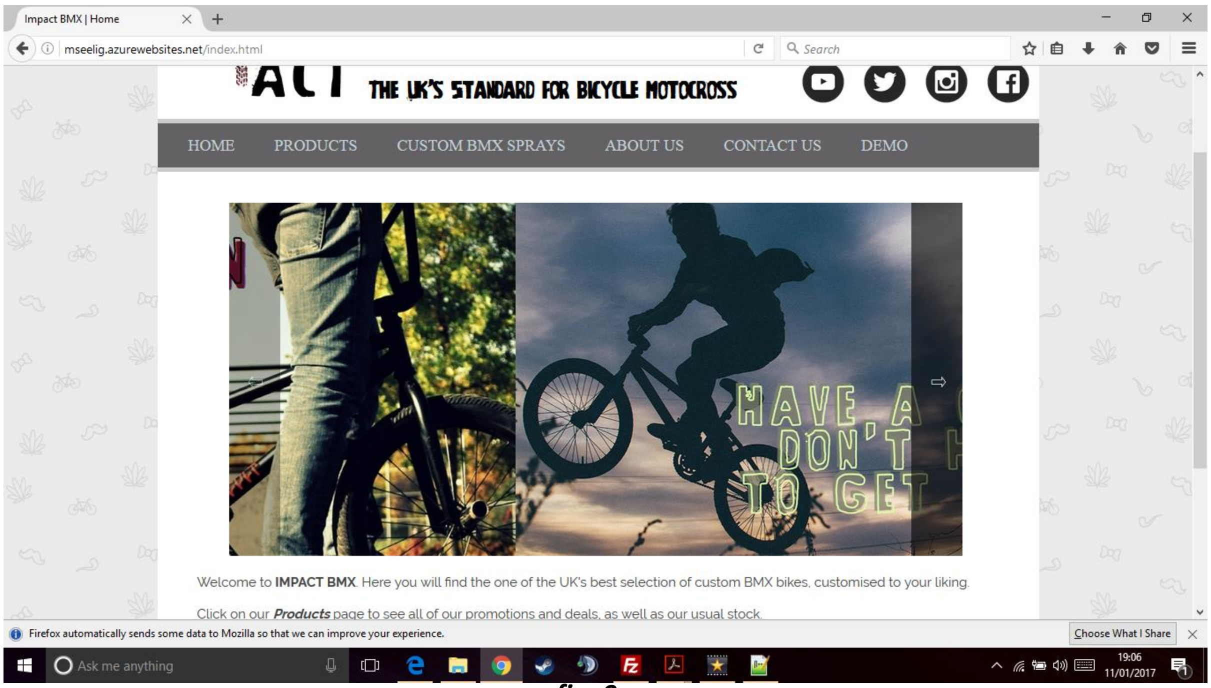
0800 744 305

1337 Westfield Shepherds Bush
Westfield London Shopping Centre
London
W12 7GF

fig. 8

Mozilla

Image slider working through use of *-moz-*



Welcome to **IMPACT BMX**. Here you will find the one of the UK's best selection of custom BMX bikes, customised to your liking.

Click on our ***Products*** page to see all of our promotions and deals, as well as our usual stock.

Firefox automatically sends some data to Mozilla so that we can improve your experience.

fig. 9

Safari

Transform scale and transition style through use of -webkit-

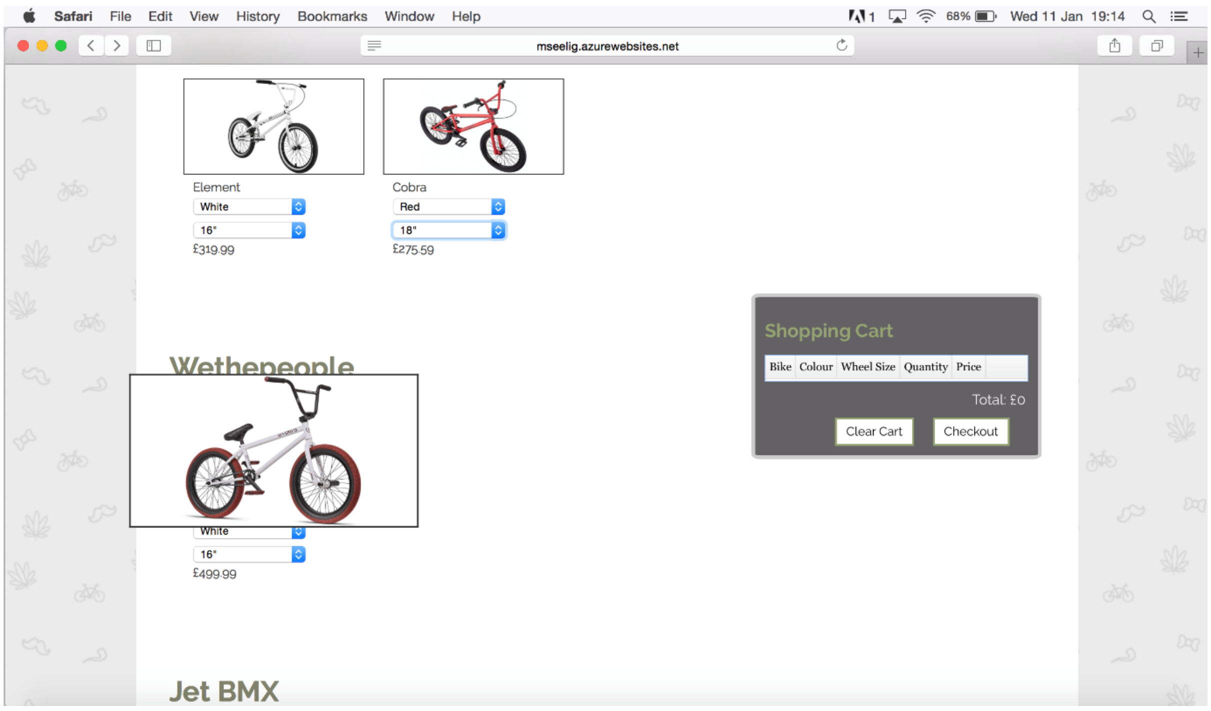


fig. 10

Microsoft Edge

Transform style through use of -MS-

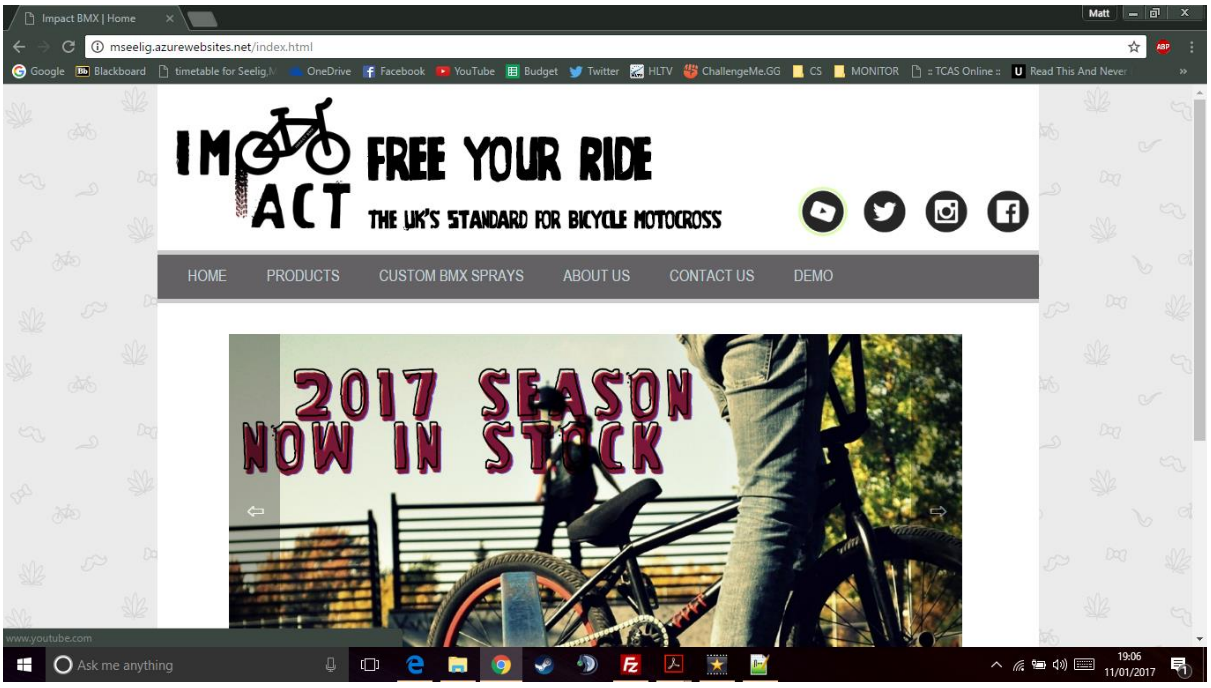
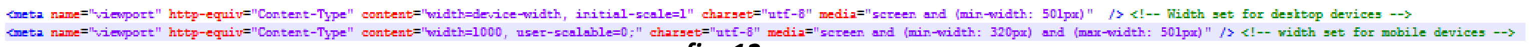


fig. 11

Note: Internet Explorer is no longer support, thus a developer doesn't need to consider IE for interoperability.

Mobile

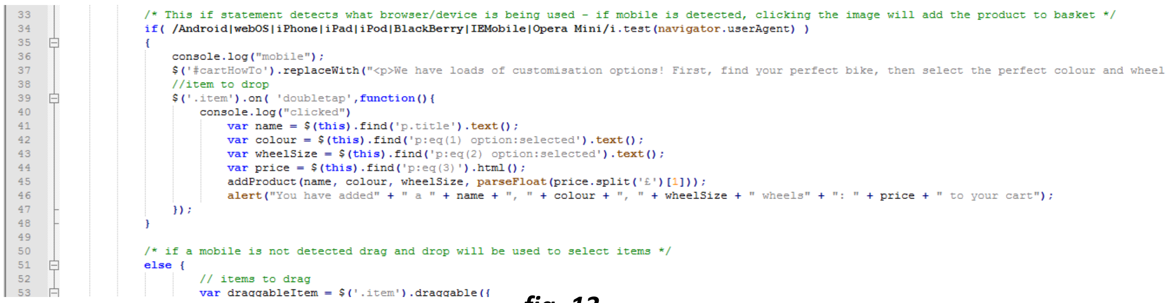
I included media queries in my meta tags to load separate css files for desktop and mobile.



```html
<meta name="viewport" http-equiv="Content-Type" content="width=device-width, initial-scale=1" charset="utf-8" media="screen and (min-width: 501px)"  /> <!-- Width set for desktop devices -->
<meta name="viewport" http-equiv="Content-Type" content="width=1000, user-scalable=0;" charset="utf-8" media="screen and (min-width: 320px) and (max-width: 501px)" /> <!-- width set for mobile devices -->
```

fig. 12

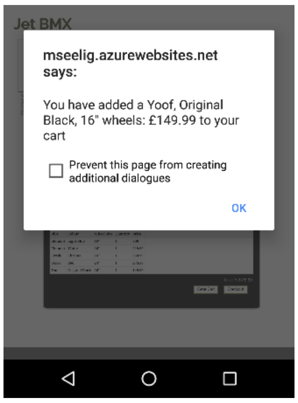
I also added if statements in my JavaScript so products are dragged and dropped for desktop, and clicked on with an alert for mobile.



```javascript
/* This if statement detects what browser/device is being used - if mobile is detected, clicking the image will add the product to basket */
if( /Android|webOS|iPhone|iPad|iPod|BlackBerry|IEMobile|Opera Mini/i.test(navigator.userAgent) )
{
    console.log("mobile");
    $('#cartHowTo').replaceWith("<p>We have loads of customisation options! First, find your perfect bike, then select the perfect colour and wheel
    //item to drop
    $('.item').on( 'doubletap',function(){
        console.log("clicked")
            var name = $(this).find('p.title').text();
            var colour = $(this).find('p:eq(1) option:selected').text();
            var wheelSize = $(this).find('p:eq(2) option:selected').text();
            var price = $(this).find('p:eq(3)').html();
            addProduct(name, colour, wheelSize, parseFloat(price.split('£')[1]));
            alert("You have added" + " a " + name + ", " + colour + ", " + wheelSize + " wheels" + ": " + price + " to your cart");
    });
}

/* if a mobile is not detected drag and drop will be used to select items */
else {
        // items to drag
        var draggableItem = $('.item').draggable({
```

fig. 13

This final *fig* exhibits the alert on an android phone using chrome browser.



Jet BMX

mseelig.azurewebsites.net says:

You have added a Yoof, Original Black, 16" wheels: £149.99 to your cart

☐ Prevent this page from creating additional dialogues

OK

fig. 14

Website URL

http://mseelig.azurewebsites.net/

Word Count (excluding cover page, contents, and references): **923**

References

Bojangles. (2011) *What is the standard naming convention for html/css ids and classes?.* [online]. Available from http://stackoverflow.com/a/6028289/7153114 [Accessed 27 December 2016].

DaveE. (2014) *Remove 1px transparent space from CSS box-shadow in IE11?.* [online]. Available from http://stackoverflow.com/a/22205418/7153114 [Accessed 17 November 2016].

jQuery. (2017) *.replaceWith()*. [online]. Available from http://api.jquery.com/replacewith/ [Accessed 10 January 2017].

Major, B. (2016) *A collection of mobile event plugins for jQuery.* [online]. Available from https://github.com/benmajor/jQuery-Touch-Events [Accessed 28 December 2016].

Katz, J. (2010) *Beyond the Hype: Understanding HTML5 and its Potential for e-Learning and mLearning.* [online]. Available from https://www.learningsolutionsmag.com/articles/465/beyond-the-hype-understanding-html5-and-its-potential-for-e-learning-and-mlearning [Accessed 12 January 2017].

Map-embed. (2016) *Embed Google Map - Add Google Maps to your Website.* [online]. Available from http://www.map-embed.com/ [Accessed 02 December 2016].

Romjon. (2017) *Web Standards*. [online]. Available from http://romjon.com/briefing/web-standards [Accessed 12 January 2017].

sweets-BlingBling. (2015) *What is the best way to detect a mobile device in jQuery?.* [online]. Available from http://stackoverflow.com/a/3540295/7153114 [Accessed January 06 2017].

W3. (2011) *CSS COLOR.* [online]. Available from https://www.w3.org/Style/CSS/specs#color [accessed 12 January 2017].

W3C. (2014) *A vocabulary and associated APIs for HTML and XHTML*. [online]. Available from https://www.w3.org/TR/2014/REC-html5-20141028/sections.html#the-header-element [Accessed 12 January 2017].

W3C. (2017) *JAVASCRIPT APIS CURRENT STATUS.* [online]. Available from https://www.w3.org/standards/techs/js#w3c_all [Accessed 07 January 2017].

W3schools. (2017) *JavaScript Style Guide and Coding Conventions.* [online]. Available from http://www.w3schools.com/js/js_conventions.asp [Accessed 07 January 2017].

Image references available in the supporting documentation .zip file in a separate document entitled *'Image References'*.