**Literature Review**

This section explains some of the In order to allow user interaction at the implementation stage of the project, some sort of technique will need to be adopted that will allow for manipulation of the client side elements to convey the illusion of a user customising a cycle.

JavaScript would be one of the first tools that may come to mind, which is a scripting language that was designed for use within a browser on the client side. The programming language is especially useful when implementing website interactivity and validation. JavaScript is a low level language and does not provide advanced page manipulation and decoration functions, and provides nothing concerning animation. (Dubois, 2015) There have also been common, known problems with issues related to cross-browser incompatibilities which can often result in certain functions and features not working the way it which they were intended unless fully testing and implementing code to cater for each different browser. (Advantages of using pure JavaScript over JQuery, 2015)

JQuery will provide a main research topic for the project, understanding into why jQuery has become so popular compared to alternatives. In order to compare the tool, the architecture of the library and its capabilities will need to be implemented to determine how certain functions actually work. Learning jQuery, (Swedberg and Chaffer, 2013), discusses the tool in depth along with a guide of how to implement it. The book was published in 2013, up to date with latest version of jQuery, which means the book can also be used as a tool to explain how to implement certain functions and features. Not only will the book provide implementation techniques, these methods will follow the best practices as one of the authors, Karl Swedberg, is a member of the jQuery Foundation board of Advisors. This is a project run be volunteers that work together to constantly adapt, improve and progress the tool with their ambition to make jQuery ‘the best JavaScript tool possible. (jquery.org, , 2015a) This isn’t the first book the duo of authors have written; there are several other books that discuss previous versions of jQuery along with a handy reference guide of implementation techniques along with how jQuery components can be integrated with Ajax. While this is extremely useful, jQuery’s website offer online documentation with similar contents and these will be more current as the site has since been updated. (jquery.org, 2015b)

In 2008, Mary Meeker, an analysis for at Kleiner Perkins Caufield Byers who reviews technology trends annually, made a huge and substantial prediction; “Mobile to overtake fixed internet access by 2014”. Mary helped create the ‘Internet Report’, which became known as the ‘bible’ for internet investors and helped build her reputation within the industry. (Meeker, 2008)  
Today, according to Perez, we have now move past the mobile tipping point and developers and businesses are not adopting mobile first and responsive design techniques. (Perez, n.d.) This further emphasises on how important it is for businesses to ensure the web sites/applications now address their mobile users and ensure they experience the same experience as they have now become the most crucial target users. For this reason, the final artefact will ensure that it in turn adopts a responsive design, as this will allow for the same site to dynamically respond to the user’s viewport/screen size.

A report written by Jill Karlsson, a student of Uppsala University, Sweden, discusses in detail the techniques of implementing responsive sites along with the most common frameworks available to developers. This report compares the tools providing a rough outline of the some of the features that can be discussed in the final research aspect of the project. The report is not necessarily the most reputable source; however it’s supported and hosted by the Anglian Ruskin Library. The report is also current as it was published in June of this year. (Karlsson, 2014)

The following book helps to explain technical design slightly further; Responsive Web Design with HTML5 and CSS3 (Frain, 2012)written by Ben Frain, a freelance frontend web designer/developer. The book offers a highly in depth ‘How-to’ guide on modern responsive web development techniques. The book is reviewed by a highly reputable Web developer, Ed Henderson, a computer science graduate who specialises in web development along with a passion for design. Kamrujaman Shohel, another highly knowledgeable reviewer who started his career working with a multimode group (Microsoft Department) as an analyst also supports the book. This helps provide confidence in the reliability of the source along with ensuring that information is recognised and supported by reputable professionals. It also helps emphasise how important responsive design has become and why developers should begin adopt this technique.

In order to combine modern design techniques, JavaScript library interaction and usability along with the features of responsive design are web development frameworks. An article written in Smashing Magazine by the author Jen Kramer, discusses in detail, as a developer, weather we should use the responsive web frameworks available to us and why they may or may not make the development process more efficient. (Kramer, 2014) Smashing magazine is an online magazine for professional developers that discuss the latest and biggest current news, with a focus on “useful techniques and best practises” within the IT and technology industry. (Gremillion, 2015)

In the article it argues the reasons against web frameworks and how they only make the life of the developer more difficult; it further leads onto how they can benefit the developer, improves cross-browser and cross-device compatibility, along with the alternatives. As the article discusses the alternatives, this will provide key frameworks to research into for the research element of the report along with what a developer should look for in a framework. This will ensure that project can kepp within the short time frame, and time is used effectively researching and implementing worthwhile frameworks. Most frameworks sole purpose is to provide tools/techniques to allow a developer’s website to become responsive, however some act as a rapid development resource to allow a website to evolve from plans into a prototype in a short period of time. This is possible as frameworks allow developers the ability to use pre-styled components and templates that can be simply implemented rather them having to design each component.

JQuery UI is a collection of engaging widgets and essential interaction-helpers that can help drastically reduce the amount of code and time spent developing an interaction interface and design. (jquery.org, 2015a) This integrates alongside jQuery, offering styled components that can be use together to create user interaction on a site. While there isn’t currently a need for this particular tool at the implementation stage, a large amount of time will be spent researching and implementing jQuery and this will help reduce the amount of time spent to create controls to trigger jQuery events. Another book published by PACKT publishing, jQuery UI 1.6 written by Dan Wellman offers an in depth technical ‘how-to’ guide on jQuery UI and how it can be used alongside jQuery. (Wellman, 2009) This is a lead on book from learning jQuery that immediately presents some basic jQuery and asks the reader whether they understand is fully. It further explains some of the most popular functions, demonstrating how they can be used and also how their styles can be customised to fit in with other website themes and maintaining a consistent design. The book again is reviewed by reputable sources and Dan Wellman plays a crucial role in improving and developing the open source library meaning that the book demonstrates best practices. The only problem with this book is that as technology is constantly evolving and changing, some of the practical implementation examples may differ to the latest version of jQuery, however the online jQuery UI documentation can be used to help as this provides the latest version. (jquery.org, 2015b)

In order to implement a framework or JavaScript library there are two main methods; the first is downloading the full or customised library or the second is using a content delivery network (CDN). The book, A Practical Guide to Content Delivery Networks, (Held, 2010) describes a modern CDN as a “group of geographically dispersed servers deployed to facilitate the distribution of information generated by web publishers in a timely and efficient manner”. As the project will be discussing how to use different types of tools, sometime should be spent on understanding the best way to implement them. This book provides an in-depth explanation of the advantages and disadvantages of CDN’s which will allow a decision to be made during the implementation stage of whether CDN’s are the best technique to be used to integrate the tools into a project or whether a local version should be used.

The project will focus on researching into the available tools and demonstrate the difference between them and their capabilities. Some of the tools that will be researched can be supported by reputable reports; however some are so new that there are limited areas that can be researched into that specific tool so some websites will be used especially during the implementation stage.

**Methodology**

**Introduction**

This section describes project scope along with the implementation structure that will be used during the development of the final artefact. Research and evaluations are made of the tools stated within the scope along with justifications of why the will be chosen to implement the final solution. A detailed project Gantt chart further structures project progress into a timescale along with outlines of possible contingencies.

**Scope**

(research, practitioner interview, prototype feedback)

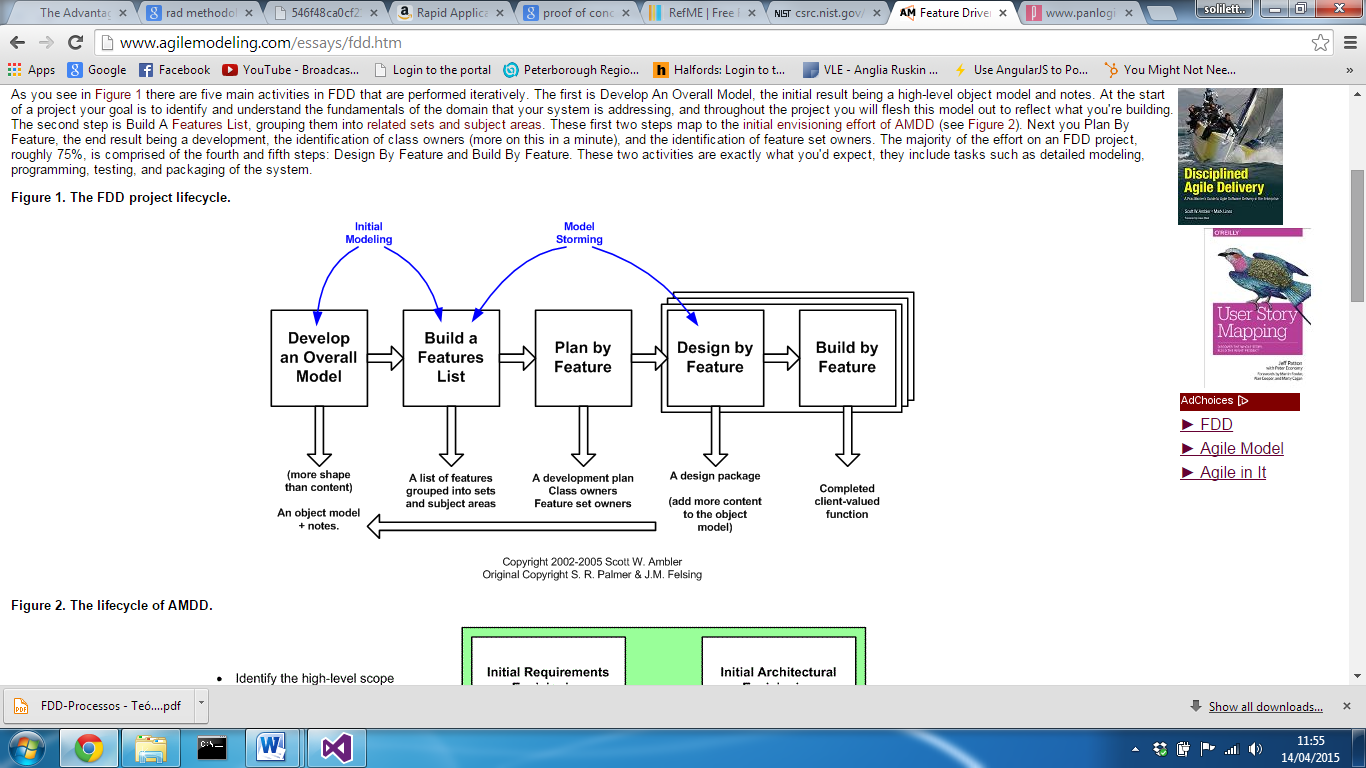
**Development methodology**Before undertaking the main project, a basic prototype will be created following the proof of concept (POC) approach demonstrating the feasibility of the artefact. This ensures that the final artefact is possible and focuses directly on implementing the main system requirement, with minimal priority on the user interface. This is a common approach used by web development firms in order to ensure that their vision of the artefact meets that of the clients. (Panlogic Digital Engineering, 2015) This approach will be crucial as the main requirement of the artefact will be developing an interface that provides the user the ability to customise a cycle.

The methodology that will be adopted for the main project will be rapid application development (RAD). The approach was first formalised by James Martin (Martin, 1991), who believed that it refers to a development life cycle designed for high quality systems with faster development and lower costs than traditional approaches. This is an alternative to the waterfall method that focuses more emphasis on the development process rather than planning. This model reduces planning required and focuses on making adjustments based on knowledge gained as the project progresses. This will be crucial during the development stages as most of the frameworks and tools used will be researched and implemented, and at each stage this could have a large impact on future enhancements of the artefact.

Over time, the approach has evolved and became used as an umbrella term to encompass a number of methods and approaches by many different venders applying their own interpretation. (Harris, Eichora, Goodwin and Henson, 1997)

During the artefact development, alongside RAD, the feature driven development process will be adopted. As the project will have a large emphasis on researching into different tools and identifying the most suitable, knowledge and experience will be gained by the developer, and this will have a large impact on the structure and design of system functions as the system progresses.

Feature driven development approach ensures that the system will progress after a function or feature has been fully implemented, regardless of whether that feature has any direct dependencies. This is an agile technique introduced in 1999 via the book Java Modeling In Color with UML (Coad, Lefevre and DeLuca, 1999). There a five key principles that are performed iteratively, with the first consisting of an overall model of the system with a main goal of understanding the true fundamentals of the system. The second step is to form a feature list, grouping them in relation to their functions along with an iterative order of approach. After the feature list has been built a plan can then be created with the final steps design by feature and build by feature. As their names describes the basic design of implementation of the feature can be achieved and further the feature can be build and tested usually following a testing approach. (Feature Driven Development (FDD) and Agile Modeling, 2015) Each feature will be tested to ensure it meets the requirements of the system before starting the process again.



**Project Management**

Gannt chart

Meeting with supervisors (minutes)

**What needs to be achieved?**

The desired artefact will be a web based system as this will allow users to access the system without the need of installing or setting it up. The website should include the most common components consisting of a navigation bar and a main body to display content. It will also need to respond to users of all devices using the most common and popular browsers.   
This section will explain how this can be accomplished using a website, implementing some of the latest techniques that make the development process more efficient and easier.

**What is a Web development Framework?**A web framework is library of features and templates designed to reduce a developer’s workload making the development cycle more efficient. Frameworks exist to save the developer from inventing the necessities, providing features that can be customised to meet the requirements. (WebMonkeyStaff, 2010)  
Within the web development environment there are several different frameworks designed for different areas of a website design. One of the main topics that developers currently face is how they can target the mobile device audience.   
With the number of mobile users, who currently navigate the internet, now overtaking those who use desktop devices, huge emphasis is placed on optimising the user experience for those users. Typically, a mobile device will display a full site where the user has to zoom and drag the page while reading content, which for the user can become quite frustrating.



(Figure 1 - Perez, 2015)

In order to combat this, many websites are now adopting some kind of technique in order to provide the best experience for each user.   
Progressive Enhancement (PE) is a web design strategy that makes an effort to create a website that is accessible for all devices, following a layered approach with a set of core principles emphasising the accessibility of content and functionality. (Parker, 2010)

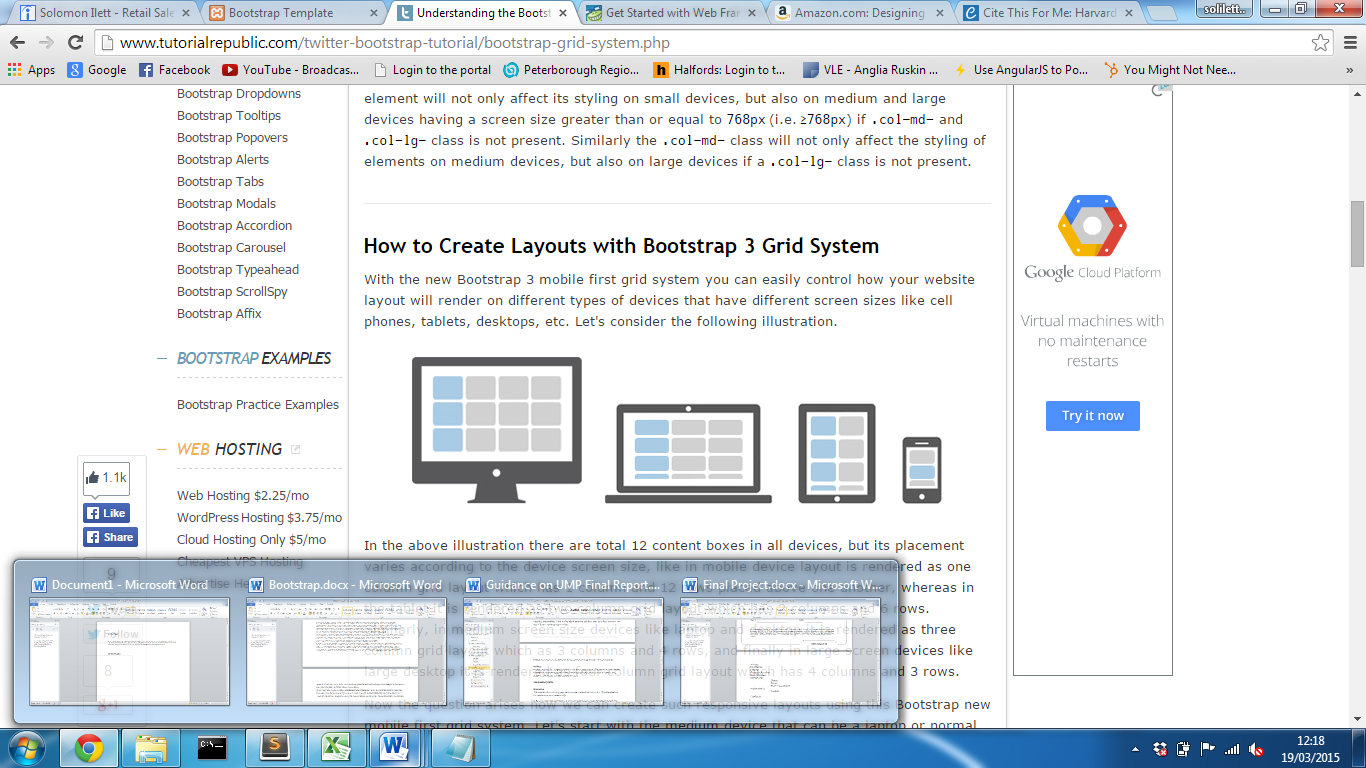
**Web Application Frameworks**

HTML Document Object model defines every element within a website structure as a node. For example, the document itself can be defined as a document node and each element “<body>” or a “<div>” can be defined as an element node. Each element node contained inside a parent node, becomes the child of that node. (HTML DOM Element Objects, 2015)

Each element node can then be assigned an attribute node, usually a class or a unique id which will further apply a specific styling or allow JavaScript to manipulate that element. The fundamentals of an application framework are to further apply attributes to defined element nodes, which overall reduces the work load and efficiency of developing a website. They provide the developer with the base structure, functionality and styles which can then be customised to meet the requirements of the system.

**Bootstrap**

Bootstrap today has become one of the popular frameworks for creating websites and web applications, (Meanpath.com, 2015) offering a collection of free tools to ‘kick-start’ the front-end development. The bootstrap site describes bootstrap as a ‘sleek, intuitive and powerful mobile first front-end framework for faster and easier web development’. (Mark Otto, 2015)  
The framework was originally developed for Twitter application by twitter’s team of developers and was later made open-source on Friday 19th August 2011. There have since been several updates, with one of the most substantial, responsive design. (Bootstrap, 2015)  
Responsive web design, originally defined by Ethan Marcotte, incorporates design techniques that allow a website to respond to the need of users or device viewport. (Web Fundamentals, 2015) Typically, a website will detect the user’s device viewport and either displays a desktop or mobile version of that site; responsive website design provides a structured layout that adapts to the screen size providing one site for all device screen sizes.   
Bootstrap framework provides a grid based system that allows a developer a fast and easy solution to create a site layout following a row and column design. When the viewport/screen size changes each column will resize and stack to fit the screen.

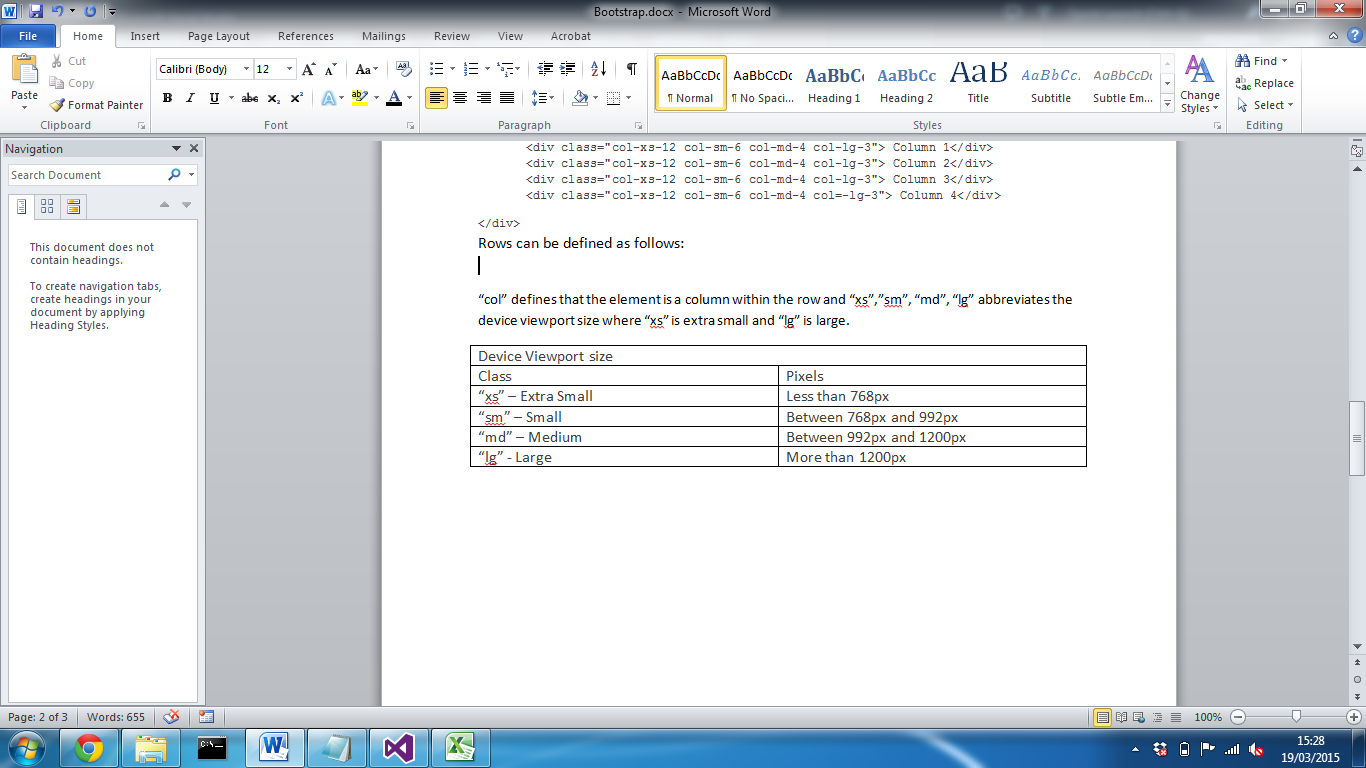


(Figure taken from …)

A typical desktop site can be split into 12 columns where a column can be defined by adding a class to an element.  
 i.e:

<div class="row">

<div class="col-xs-12 col-sm-6 col-md-4 col-lg-3"> Column 1</div>  
<div class="col-xs-12 col-sm-6 col-md-4 col-lg-3"> Column 2</div>  
<div class="col-xs-12 col-sm-6 col-md-4 col-lg-3"> Column 3</div>  
<div class="col-xs-12 col-sm-6 col-md-4 col=-lg-3"> Column 4</div>

</div>  
Rows can be defined as follows:

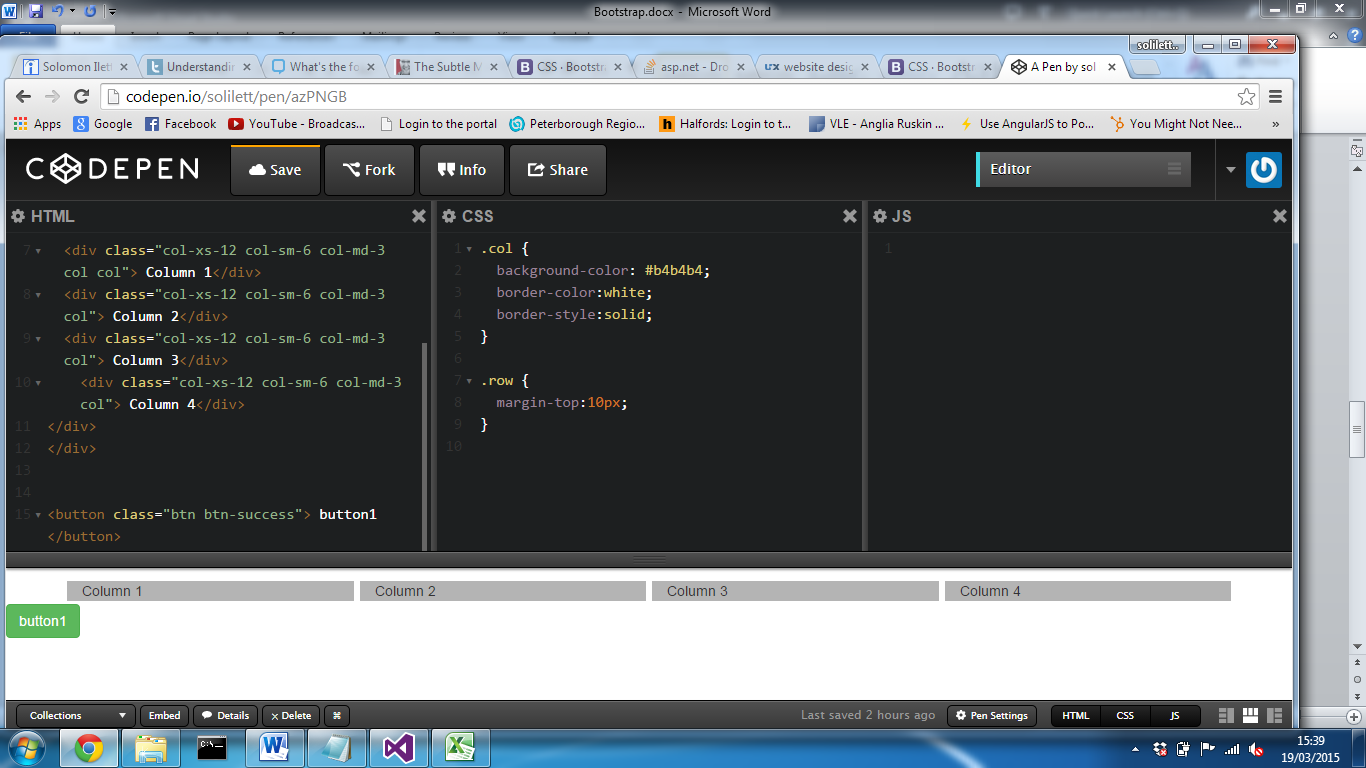
“col” defines that the element is a column within the   
 “md” abbreviates the device viewport size (see below).

|  |  |
| --- | --- |
| Device Viewport size | |
| Class | Pixels |
| “xs” – Extra Small | Less than 768px |
| “sm” – Small | Between 768px and 992px |
| “md” – Medium | Between 992px and 1200px |
| “lg” – Large | More than 1200px |

(Boostrap, 2015)

“4” defines the amount of the screen that the column will be allocated.

Any content contained within the column will then automatically respond based on the screen size. This means that the developer doesn’t have to create CSS media queries, they can simple define the class and the framework will do the rest. This immediately reduces the workload for the developer, as without this, they would have to implement a media query and then test to ensure contents change the way in which they are intended and each element could consequently then be effected by changing a particular element node.   
  
Following the same principles of assigning an element a class, bootstrap can also transform other elements by adding CSS styling to make them more professional and reduce the amount of work for the developer.   
  
<button class=”btn btn-success”> Button1 </button>



This applies styling to the button changing background colour, fore colour, font-family as well as its hover behaviours. This provides styling that may exceed an amateur web developer’s ability, but also reduces a lot of time for an experienced user. All bootstrap elements require containers to wrap content; this can either be fixed width; by default this places a 100px margin on both the left and right. The second is fluid which takes full advantage of the screen size and stretches to with a minimal margin.

Bootstrap provides almost every common DOM element, with default styles and JavaScript manipulation that can be simply implemented by adding a class attribute to an element node. One of the most noticeable features of the bootstrap framework is the collapsible Navbar that responds to the view port device. The colours and layout on Navbar links can different based on the particular style; however the fundamentals are the same. Typically, the Navbar positions list items horizontally across the top of the page and when the viewport is less than 768px, the list items then become items contained within a toggled drop down menu. This is extremely helpful when there are several list items contained within a unordered list element, as rather than squashing them on a mobile device screen, they then become vertically positions down the page.

The bootstrap framework is further supported by the latest browser versions improving cross browser compatibility; supports the four most popular internet web browsers including Internet Explorer 7+ , Safari, Firefox 4+ and Google Chrome. (Bootstrap, from Twitter 1.1.0 Documentation, 2015) This ensure that’s any components and features used during the development stage will be supported ensuring that the system can be used for different users of different browsers.

When implemented, the framework provides element nodes with default styling that has a number of simple but effective styles that create an overall professional theme. This differs based on the particular class assigned to a element, however one of the key elements is the font. This has a large impact on the overall presentation of a site, and immediately makes a site more attractive. Regardless of what a developer decides to implement, bootstrap allows a site to become consistent as all of the components offered are designed to fit with one another. (BOOTSTRAP REFERENCE)

In order to implement the framework, there two main methods, as stated in the literature review, however each method can offer a completely different experience. Bootstrap is customisable when downloaded and stored within the project. The framework can be customised so that only the parts required are included in that version of bootstrap. The main advantage is that if the developer only requires a minimal amount of bootstrap features, a lightweight version of the framework can be used, reducing both load time of the project and overall project size. There are so many features offered by the framework, that it would be extremely hard to implement every component. A developer can simply check a tick box for the required component and the downloaded version will only include them specific parts.

After the components have been selected, jQuery plugins can be selected; certain features of the jQuery plugins are dependent on what specific components are including; for example, if you require the dropdown component, both the dropdown styling and jQuery plugin need to be downloaded to implement it into a working solution. Finally, the colours of the required components can then be customised; these include elements colours, typography and icons. As mentioned, bootstrap styling is written using the LESS CSS pre-processor so the structure of certain colours can be completely dependent to that of the colour it’s contained with. (See LESS section for more details) After all customisations have been made, that unique version can then be downloaded and included within the project.

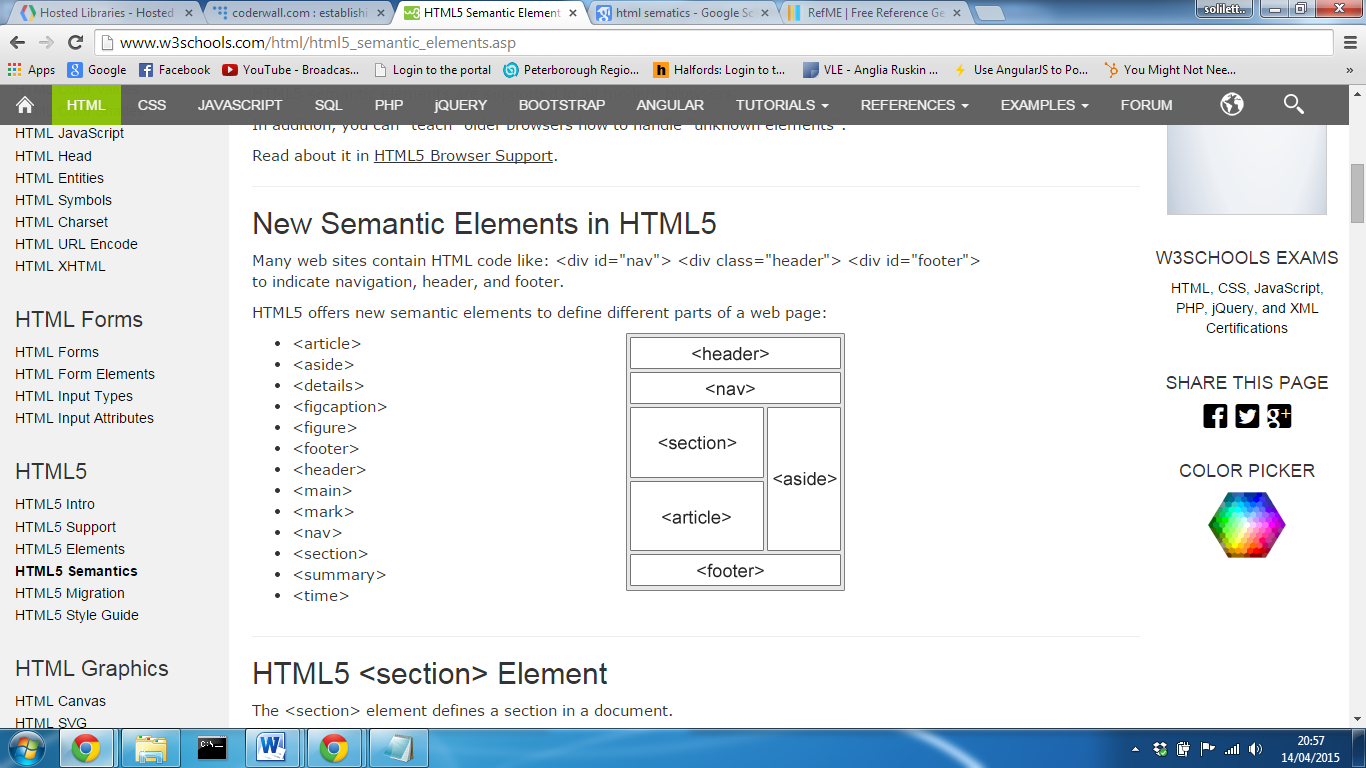
At this point, you may be begin to wonder what happens if you have downloaded a custom version and you want to implement an extra feature or change the colour of a component; Bootstrap customise website offers an easy-to-use, well organised user interface that allows a user to upload a previously customised version of bootstrap. (BOOTSTRAP CUSTOMISE SITE) All previously customised components will be checked on the site along with all the changed colours, fonts and icons. This means that the required change can then be altered and a new version of the framework can be downloaded.   
The alternative to downloading a local version is a Content Delivery network. While this may make the overall project smaller in size, and load style quicker through cache, the CDN will be a full version of the library and the advantages of using a content delivery network maybe completely worthless if the project it is used for only uses a low number of its features.

Bootstrap getting started documentation, provides the HTML mark up with the most popular to begin implementing the framework. The examples include starter templates, common components, responsive grid layout, login templates and other useful basic website designs. One of the most significant factors to consider when using a 3rd party framework is support; without it you will struggle to optimise the full use of the tool and will spend a lot of time fixing problems because a function or feature hasn’t been implemented properly. Bootstrap offer a huge documentation of how to use the tool along with fixes for some of the common problems incurred by developer who use the tool (REFERENCE). Not only that, but as bootstrap is one of the most popular framework currently used, many developers post solutions to problems they have incurred using the framework. (see appendix 2)

It is essential to include document stylesheet links and JavaScript references in a specific order, following best practises. One of the common factors to consider is at what point to add these into a project. Following best practises, stylesheets are loaded first in document head, then main contents and finally the JavaScript. This is crucial with bootstrap as a browser typically halts page parsing until an external script has been executed and often this can take slightly longer than parsing just the HTML document. (How browsers work, 2015) If the JavaScript link was placed higher than the body element, it could result in the page taking longer to load.

After realising the advantages of the bootstrap framework, it may become worthwhile considering its flaws before immediately implementing it into a project. As mentioned previously, each element node can be assigned a class and bootstrap styles will then be applied to that element; while this is true, certain characteristics of that styling can become adverse, if the parent of that element is not that of what bootstrap styling requires. For example, if implementing a div with a “jumbotron” class alone will mean that the width of the div will automatically be the same as the viewport, with no margins and no left or right padding. In order to add change this, an extra div with the “container” class needs to be added as the parent of the class; this will then add margin and padding to the div. While only a basic component, nearly all aspects of bootstrap framework require specific parent or child nodes which can make styles verbose and lead to lots of HTML output that’s not perfectly semantic. An ideal page that follows good semantics is one than making each element node easily identifiable and readable for developer and browser. (HTML5 Semantic Elements, 2015) This can be achieved by using self-explanatory class names and ID’s or using HTML5.

HTML5 offers new semantic element nodes that can be rendered by the browser and easily understand by a developer. The diagram below displays some of the most commonly used elements that separate the main body content into logical sections.



From w3schools

In order to maintain the semantics in the page, while still using the bootstrap framework, now becomes increasingly difficult as placing class attributes into tags breaks the semantic logic. A rival pre-processor of LESS.js is a CSS extension called SASS. Dan Tao, a member of the coderwall site, wrote an article demonstrating how using SASS, using the “@extend” function, allows a set of CSS properties, to be shared from one selector to another (Tao, 2013). The concept allows the developer to avoid having to right multiple class names on HTML elements and identifies a node by its HTML5 element node. (Sass: Sass Basics, 2015) This allows the framework to be fully implemented without the need of a heavy mark up.

When discussing bootstrap with a practitioner who works for a large UK based vehicle insurance comparison site (see appendix 1), emphasised the fact that often, a HTML document can be overcrowded with common attribute names, which makes the structure harder to follow and ignored best semantic practices. During the practitioner discussion it was stated how they found integrating the bootstrap framework into the virtual environment was quite complex, it was far easier for them to start from the ground up. This was emphasised in a blog for developers where it was stated the integration process can collide with previous setup and it could result in far extra work dealing with errors which could defeat the point in even using the framework. (5 reasons NOT to use Twitter Bootstrap, 2015b)

As the framework provides default styles, layouts and characteristics, it can often become extremely difficult to customise the look to take away the “Twitter look”. Websites that use the framework often present a similar look and feel and often large organisation want a website that is personal to them and reflects their brand effectively. In order to conquer this, many developers overwrite the style properties, however this can take a lot of time and a lot of additional code; this again goes against the sole purpose the framework is used and it may be easier manually design their website from scratch.

**Foundation**

While bootstrap currently may be the most popular framework, there are also alternatives available. Foundation, created by an organisation called ZURB, an interaction design company founded in 1988. (Crunchbase.com, 2015) The framework originated from their ZURB style guide and in 2011, Foundation was released which including Global CSS, jQuery plugins, common elements and best practices to enable rapid prototyping. (Zurb,2015) Very much like bootstrap, ZURB realised that mobile design was becoming more popular and more important; this lead to Foundation becoming the first open-source framework to be responsive and the first to be mobile first. (Zurb, 2015)  
Very much like bootstrap, foundation offers the similar responsive grid system, where content can be structured in rows and columns and components that can be implemented by assigning a class attribute to an element node. While there is no huge difference in the way the fluid grid systems behave, foundation offers some extra features that make them slightly ahead of the game.

The block grid system gives the developer the ability to evenly separate contents of a list within a grid and it will adapt regardless of the screen size. Very similar to bootstrap, each common screen size can be identified using “small”, “medium”, “large” etc, and based on specific attribute assigned to that grid, the contents will respond accordingly but still maintain an even margin. Bootstrap splits a row into 12 columns and foundation framework does the same by default (REFERENCE) however, there is no limitation of how many columns are stored in a row. Using the grid system, the developer can state how many columns they want in a row rather than with bootstrap each row always consists of 12 and as a developer you have to ensure either all columns add up to twelve or offset them to align them centre.

For example, below demonstrates a scenario where on desktop devices, 5 images are displayed horizontally.

**Bootstrap Grid System**

On large screen, five separate column elements can be defined assigned them will 2 sections each out of a total of 12 in the row. The first column can then be offset, which will then leave 1 empty column before and after all of the content. Using bootstrap each separate column has been defined inside a parent row container.

<ul class="row">

<li class="col-lg-2 col-lg-offset-1 ">

<img src="dummie.jpg" width="100%"/>

</li>

<li class="col-lg-2">

<img src="dummie.jpg" width="100%"/>

</li>

<li class="col-lg-2">

<img src="dummie.jpg" width="100%"/>

</li>

<li class="col-lg-2">

<img src="dummie.jpg" width="100%"/>

</li>

<li class="col-lg-2">

<img src="dummie.jpg" width="100%"/>

</li>

</ul>



**Foundation Block Grid System**

Using foundation rather than placing an attribute on each list item element node, the grid attribute can be defined on the “<ul>” tag.

<ul class="large-block-grid-5">

<li><img src="http://i.imgur.com/AAwx2W0.jpg"></li>

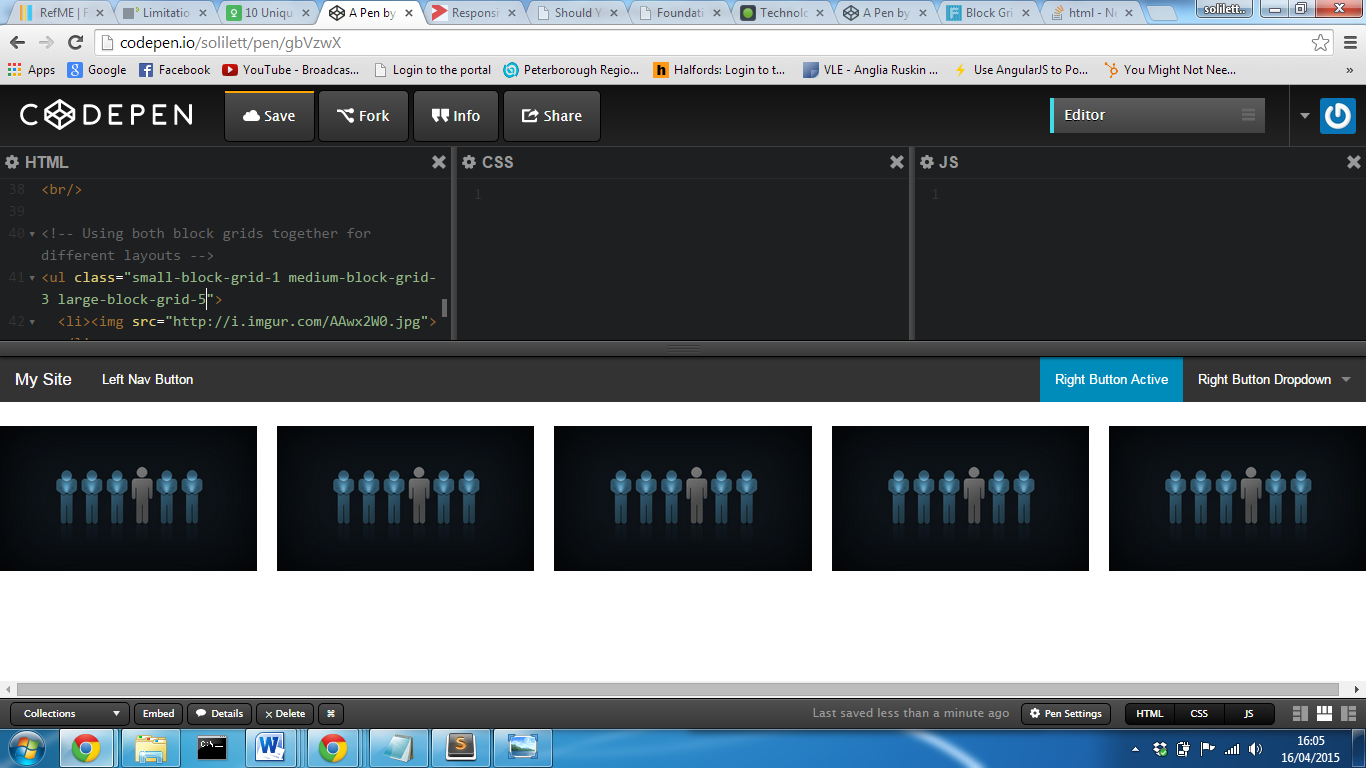
<li><img src="http://i.imgur.com/AAwx2W0.jpg"></li>

<li><img src="http://i.imgur.com/AAwx2W0.jpg"></li>

<li><img src="http://i.imgur.com/AAwx2W0.jpg"></li>

<li><img src="http://i.imgur.com/AAwx2W0.jpg"></li>

</ul>



This makes implementing the grid system a lot easier as the grid can be defined on the parent element and as long as there are list items contained within, style properties will be applied. Foundation’s block grid system utilises the full width of the screen whereas, because on the bootstrap grid the first content is offset, there are two empty columns. While this could be solved overwriting the width properties, frameworks are designed to make developers life easier thus adding extra work load.

The great thing about foundations block system is any number within the 12 columns of the screen can be defined and the framework will automatically ensure each elements is of the same width taking into consideration equal separation gaps between each row. As the amount of columns is defined in one place, it also makes changes a lot easier to implement as only one elements needs to be altered; bootstraps grid system could potentially mean that the whole structure of all content inside a row will have to be altered. This is only one of currently two types of grid systems offered by foundation, however this demonstrates how foundation are slightly ahead of the curve as suggested on their website (Foundation, 2015). They also offer a similar fluid grid system that’s practically identical to bootstraps grid system, so as a developer you have the ability to choose the best one to meet the set requirements.

When using components, Foundation provides a large array of different styles for buttons, where the developer can not only change the colours of a particular button, but also the style; for example the font or the border radius. (REFERENCE) This can have a huge impact on the design of the website providing a different overall representation. One of the problems with many frameworks, as stated with bootstrap especially, they can often create a certain look, and become far too similar to other websites that have used the same website. This methods provides the ability to change features to preferred styles and properties without the need over overwriting CSS styles and can be set within the main website document.

The frameworks also integrates abide validation directly into the framework; this is a HTML5 form validation JavaScript library that supports the native API by using patterns and required attributes. (ZURB, 2015) This can be implemented within a project by adding a link the foundation abide JavaScript library included in the latest download of foundation or by a link to a hosted CDN. By placing “data-abide” informs the library that you wish to validate user input for the inputs contained inside the form tag. Any input element inside that form can then access the library features by placing certain mark up inside input tags.

For example, if a user is required to enter a email twice to ensure they match, using the abide validation the second email can be checked to ensure it is equal to the first entered email. This can be performed by adding into the second input, “data-equalto=’[the first email input]’”. An error message class can then be inserted as the sibling of the entire field surrounding the input, which will then display if the validation returns a false value.

<form data-abide>

<div class="email-field">

<label>Email <small>required</small>

<input id="email" required type=”email">

</label>

<small class="error">Your email must match the requirements</small>

</div>

<div class="email-confirmation-field">

<label>Confirm email <small>required</small>

<input required type=”email" data-equalto="email">

</label>

<small class="error">The email did not match</small>

</div>

<button type="submit">Submit</button>

</form>

Each input can also be validated to ensure that it is of the correct format using one of two methods; either placing a regex pattern inside the input element which will be checked on submission of form or by adding a type equal to the library of predefined data patterns. These include payment cards, payment card security number, URL’s, domains and even hex colours. HTML5 already support new featured to validate by an email, however this extends the client side validation and reduces the amount JavaScript validation that normally has to be implemented on website that allow for user account or online transactions. This is only the principles of how the abide form validation can be used alongside the framework and only demonstrates implementing it in the mark up; additional jQuery can be added to change what happens when the validations returns an invalid data response which means the developer can then simply output any particular error message of their choice.

In response, bootstrap supports HTML5 validation, however doesn’t include any form validation to the extent of what foundation offer. This isn’t the only feature offered by foundation, there is also an image feature called ‘Interchange’, with its main features allowing images of different sizes, quality and resolution to be display based on the users device. (ZURB, 2015b) This is the same process of arbitrary HTML content, but offers the options of setting your own media queries to trigger what image to display. (Built-in Browser Support for Responsive Images, 2015)

**Alternatives**

Web application frameworks, like foundation and bootstrap are designed to make the process of turning an idea into a fully working prototype within a fraction of the time it would normally take. You do not have to be a web guru to create a professional looking website, but simply someone who understands the basics of web design and how to use the framework. It offers essential components that are necessary for a website and allows a user to simply define which one they want. They also benefit advanced developers as they can use the framework to get the basic outline of the design and structure of a site in a short period of time, and then customise it to fit their needs. They can also only used certain features of the frameworks; for example just the grid system to allow them to create responsive content.

However as stated earlier in the report, for most developers often more time is spent on overwriting existing style properties to make the site more personal to them. This can often lead to question whether using a framework is worth it at all and whether a website designed from scratch could meet the requirements and take less time to develop. Manually creating a site allows the design to be specific to what the site was intended to do and provides the developer the ability to create bespoke design rather than something that is similar to other websites available on the internet. As stated, web frameworks like bootstrap can often look very similar and require a lot of customisation to overcome this. This is only an option if the developer has a high knowledge in web design and the time in order to develop a fully working website from the ground up.

Large organisations, just like how bootstrap was created, often develop their own frameworks to keep different parts of their website consistent. This was emphasised during the practitioner discussion (See appendix 1), as they were currently designing their own framework that all developers could use that was specific to their product. The problem they had to overcome was the fact that each team developing in different areas of the organisation had a preferred framework or technique to use which lead to design being completely different from page to page. This could make it extremely difficult for new employees to understand how certain functions and structures work as such a huge array of techniques are used, that it would be extremely difficult to find someone with that expertise. While, this is a great solution, it could further become a problem, as each new developer would have to learn how to use the framework, whereas because tools like bootstrap are so widely use, developers potentially could already have that knowledge.

Often, developers use only parts of a framework to help make development easier and quicker. After talking to a practitioner, he stated how the use of grid systems help implement responsiveness into their site a lot easier without the need manually code it which would take much longer. It was stated how they avoid using bootstrap components, because it’s hard to overcome the “bootstrap look” and easier use what they have already. Frameworks can often be extremely hard to integrate into an existing system and often mean a lot of time is spent on fixes problems that have occurred by altering an attribute or element structure.

**Justification**

**Should a framework be used?**

The advantage of a framework is that it places higher importance on website features over design allowing a developer to spend more time on developing the main functions of a site rather than focussing on how the content is presented. They are designed to take a concept and make it a prototype more efficiently.   
As the project primarily focuses on implementing user functions, allowing the user to customise a cycle, more time should be prioritised on developing these features. Using a framework will reduce time spent on the User interface of the artefact allowing basic styles and layouts to be implemented without the need to spend a large proportion of time on design. It also helps integrate responsive design, ensuring content will adapt to different user devices without the need to create custom media queries for each page. This project will start from the ground up which eliminates integrating the chosen framework into an existing system which can often create problems. This will also not only make it easier for a single developer to understand, but will also allow others with knowledge in the chosen framework to have some understanding of the page layouts and structures.

After the required functions have been implemented, extra time can be spent on overwriting base styles if deemed necessary. This method will ensure the artefact meets the basic requirements within the given timeframe.

**What framework to use?**

Foundation claim to be the “most advanced, responsive front-end framework in the world” and bootstrap claims they are the most “popular mobile first” framework. Immediately this makes it hard to choose which one is the best to use. They both offer very similar tools, both are responsive, both use a grid system and both offer a huge array of components. While foundation has been around for longer, more people have favouritism for bootstrap (GitHub, 2015) and the more developers interested in it and using it, the more support that will be available. Bootstrap has become so popular that even Microsoft has begun integrating it within Visual Studio 2013 ASP.NET development environment.

After researching into both, foundation offers more features and component options, in particular with the grid based system, and however there is no real way of distinguishing which framework is the best. It mainly comes down to a particular developer or organisation preference and which particular one they prefer using. Due to developer experience using the Bootstrap framework, this will be the chosen for the main artefact implementation. This will reduce time spent on developing knowledge into alternatives and will more advanced techniques to be implemented and the developer will already have fundamental knowledge using it.

**Customise Framework**

While there will be less importance on how content is presented to the user, the site must still look professional and provide the user will a great user experience. Without this, the user may not want to use any of the system functions because of a potentially poor user experience. Using a web application framework will reduce the workload on the user interface and efficiency of implementation; however customisations will still be required to ensure a user can perform site functions.   
As mentioned, time will be allocated to overwriting framework base styles in order to create a unique user experience specific to the artefact. This will consist of implementing CSS style properties to elements.

**Webpage Design**

Cascading style sheets allows a developer to define the look and feel of a web page separating design and the mark up of content. (REFERENCE) This allows the developer to implement styles including fonts, colours, dimensions and other properties based on the attribute assigned to an element. With modern web design today, this has now become a necessity not only to improve the overall look of a website, but to improve usability. For example, hover properties can convey to the user that the link the mouse is currently on performs an action or takes them to another page.

**Cascading Style Sheets**

In order to implement any style properties, there are two main solutions; inline styling and an external cascading style sheet. For every element tag within a DOM, a style property can be assigned which will apply styles to that particular element node. This is what is known as inline styling and while effective it doesn’t follow best practices especially with the maintainability of the system. (Kyrnin, n.d.) The main issue with inline styling alone is that each element has its own style property so if the same styles are required for multiple elements, code would need to be duplicated; this immediately increases the overall file size of the webpage. If after implementing the inline styles its further required to make changes, each element style property would then need to be altered. Not only is this inefficient, mistakes could be made when altering each element and the elements will no longer be consistent.   
External CSS allows for the content to be separate to design, allowing changes to be made to the styles of a webpage, without touching the HTML document. In a large business environment, this allows the responsibilities to be split between developing the HTML content and designing the way the page is presented. After a webpage or website has been implemented, technology and system requirements are liable to change and developers must be able to easily make changes; using external style sheets, multiple element nodes with the same properties can be assigned as long as the inherit the same attribute node. This not only reduces the amount of duplicate code but also allows changes to be made in one area which is a direct advantage over inline styling (See appendix 3 for more information).

**CSS Pre-processors**

While stylesheets are more efficient and maintainable than inline, as a developer it can often seem that the process is still inefficient and in order to maintain some sort of semantics in the mark-up, it often results in repetitive CSS styling. Pre-processors are a way of enhancing style sheets adopting new features that allow for easier implementation of styles for a web document. After some initial research, there were two main contenders.   
Both Less.js and Sass provide similar features however the way in which they are configured differ (See appendix 4). The purpose of pre-processers are to extend the standard CSS syntax providing addition features and function that can be compiled to product a CSS file that can be linked to a project. Currently there is limited published literature discussing them, however, during the practitioner interview, the question was asked, what’s you opinion on pre-processors? The first response was “why would a developer not use a pre-processor?” This is a valid point, as standard while they extend and offer extra features to the CSS language, they can also handle standard css. This means that while there are extra features available, new developers still have the ability to use standard syntax. This potentially results in a process that can assist in helping new developers progress their skills as the applications used to implement these tools can offer error checking that can help solve errors easier.

**Less.js**

Less.js is an open source dynamic stylesheet language which extends the conventional CSS language. The framework is backed by large firms including Twitter who when developing bootstrap, they used a Preboot, an open source pack of mixins and variables used in conjunction with LESS, which made design faster and easier. (Team, 2015) Not only have bootstrap developed there whole framework around LESS, Microsoft Visual Studio 2013, also supports the pre-processor into the development environment.

One of the main features offered, is the ability to store variables. Variables can be declared and then used throughout and if required, a change only needs to be made in one place rather than many. This is extremely useful for colours, as colour variables can be declared at the top of the document, and changes where the variable is called will automatically adopt the variable value. Another example how LESS can be used is the mixins function; this allows a block of properties to be contained inside essentially a method. When the mixins is assigned to an attribute, those properties will be inherited by the element. These can also be “guarded” which means they only take effect when a certain condition is true. Less can also perform self-referencing recursion where a mixin can call itself with an updated value creating a loop. (Less.js, 2015)

When structuring the CSS, proprieties can be structured in the same way they are presented in the HTML document, where siblings of the parent can be contained inside the parent tags. This helps make layout of the file more readable and makes implementing additional styles inside existing declared properties easier and reduces repeated code. While these are only some of the most basic features, they demonstrate how pre-processors can improve the development environment, integrating basic programming syntax, and incorporate them into the design of a website.

**Sass**

Sass claims to be the most mature, stable and powerful professional grade CSS extension language in the world (Sass, 2015). The extension was one of the first extensions available to developers and provides a great support documentation of how to setup and integrate the language into the development environment.

While Less.js offers the ability to include some technical abilities, it is minimal in comparison to SASS. SASS is the one of the direct rivals of Sass which further extends the CSS language and provides some extra features to make CSS more efficient. The framework includes actual logic statements and looping operators which while can do these things, the syntax becomes increasingly difficult to understand and less logical. (Less vs Sass? It’s time to switch to Sass, 2015)  
The most difficult part of SASS in comparison to Less, is configuring Ruby on rails to compile the CSS, as mentioned in appendix 4, however if the project was developed in Microsoft Visual Studio, a package can be installed that will compile SASS within the development environment.

**Justification**

Both tools offer fundamentally the same features however, when implementing SASS, its offers more advances, efficient and logical operators that follow familiar logical syntax that most programmers would be familiar with. While this makes a clear winner between the two products, configuring Less.js is far simply than setting up SASS. The Winless GUI makes the process of setting up and running the Less.js compiler simple and easy to start up and begin developing. The application will automatically store previous session states including the files to compile and all the developer has do it is run the application.   
As the project will place more importance on implementing system functions over the system user interface, using Sass would result in a large proportion of the time dedicated to adopting the language and syntax, and as stated in an article on the Zing Design website, Less.js follows common syntax very similar to CSS but extends it to provide extra functions.   
Both offer similar basic features and in relation to the final artefact implementation both will offer colour manipulation, mathematics and parameter lists along with a more logical CSS structure. Adopting Less.js will allow for changes to be made easier, CSS can be minified so page content will be rendered quicker and Less.js will integrate with Bootstrap more conveniently. While a version of bootstrap using SASS is available, it would result in a local version of the framework to be stored rather than allowing a Content Delivery network to be used. For these reason, Less.js will make implementation easier, and will add value to the project and make overwriting base styles of the bootstrap framework easier at the final stages of the development process.

**Manipulating the DOM**

One of the main functions of the artefact will be allowing the user to interact with a cycle, providing the ability to customise it by changing components and their colours. In order to allow users to select options and for a physical representation of the final outcome to be display, some kind of client DOM manipulation method which need to be adopted that has the ability to change element attributes, contents and the style properties in real time, without the need of refreshing the entire page.

**JavaScript**

**JQuery**

**KnockDown.JS**

**Angular.JS**

**Justification**

**Configuring chosen frameworks and tools**

Content Delivery Networks (CDN’s)

**Implementation**

Manipulate the cycle to change colours/components

Detect mouse position over components to access properties

Make system usable on different devices

Membership facility

Save, edit and delete customised cycle design

**Evaluation**

**Recommendations**

**Bootstrap**

Large business

* Harder to implement the entire framework into an existing system
* Adopt grid system will make responsive site design simpler

Small Business

* Fast and efficient to take a concept into a prototype
* Best suited to working from the ground up

Advantages

* Reduces workload on UI
* Browser Support
* Mobile First
* Great support

Disadvantages

* Doesn’t follow good semantics
* Mark-up can often be overcrowded with addition mark up
* Bootstrap is heavily used so site style can often be similar to other bootstrap sites

Less OR SASS

Less if less advanced user as easier to configure and follows syntax similar to CSS

Sass if UI is of a main priority – Can integrate easier into visual studio – As mentioned by practitioner

**Either can be integrated into Visual Studio with ease**

**JQuery –**

**Advantages**

**Disadvantages**

**jQuery UI reduces UI workload for certain functions offered by JQuery**

**AngularJS**

* **Seperates developmet allowing each developer the ability to integrate at the end**
* **Extremely efficient**
* **Backed by Google**

**Appendix 2**

|  |  |
| --- | --- |
| **Framework** | **Amount of followers** |
| Bootstrap | 310 thousand Users |
| Zurb Foundation | 44,000 |

(figures taken from Twitter (Twitter,2015)

**Appendix 3**

**External CSS – Applying styles to multiple elements**

This briefly explains the principles of how cascading style sheets allow properties to be applied to element node using both attributes and its structure.

Rather than duplicating code and assigning style properties to each element, DOM elements can be access via the attribute assigned. The stylesheet can then apply styles either by assigning them directly to the id or class of that element or by the layout they are structured on the web page.

**HTML Mark-up**

<ul id="parent">

<li id="child"><a href="#" class="link">Link1</a></li>

<li><a href="#" class="link">Link2</a></li>

</ul>

An id attribute can be assigned properties by identifying the element by a hash and its id name. Its best practices to ensure when assigning an element with an id, it is unique. This ensures that, if manipulating the DOM, that element can be identified.

**CSS**

#parent, #child {

color:red;

width:100px;

}

Class can be assigned to multiple element nodes and allows the same styles to be applied by assigning style properties that class. They can be identified in by using a full stop and the name of that class.

**CSS**

.link {

text-decoration:none;

}

The element can also be identified by the way it is structured on the DOM, by identifying its parents.

ul li a

{

text-decoration:none;

}

The above example will apply the same style as using a class however as the elements are identified by their structure; any content with the same layout will also inherit these styles.

With inline styling you are limited to the styles that can be applied to elements especially when assigning hover behaviours as this isn’t possible using inline styling.

If its required to apply a property to a class that’s used in other areas of the system, but the element required to change has a different layout, they can be identified both using there structure and class name.

**CSS**

ul li .link {

text-decoration:none;

}

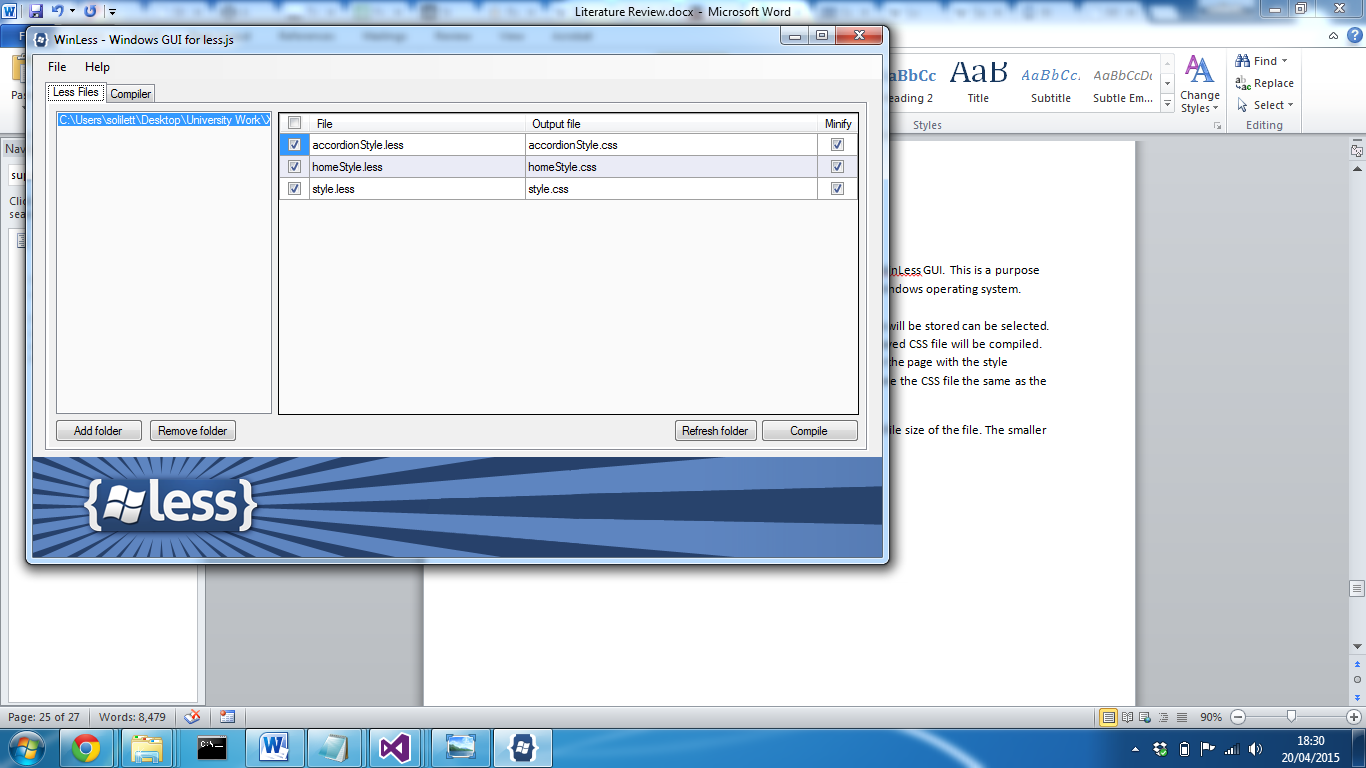
**Appendix 4**

**Less.js Configuration**

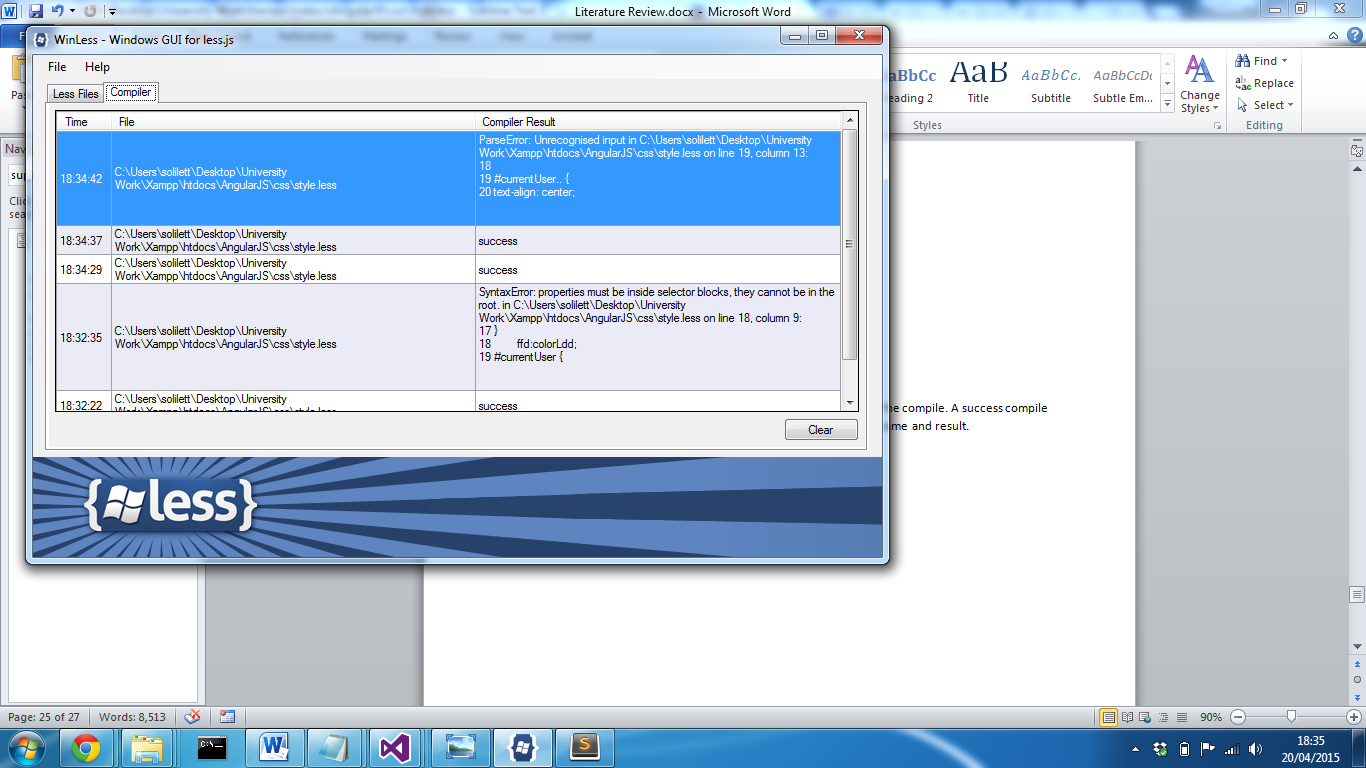
In order to implement Less.js is extremely simple through the use of WinLess GUI. This is a purpose made application designed to compile a LESS file into a CSS file on a Windows operating system. (Lagendijk, 2015)  
After downloading a installing Winless, the folder in which the less file will be stored can be selected. The application will automatically watch the file and when the file is saved CSS file will be compiled. This can then be linked to the HTML document which will then render the page with the style properties stored in the CSS file. The application will automatically name the CSS file the same as the Less file however the extension will be .css.

The CSS file compiled can be set to minified which reduces the overall file size of the file. The smaller the file size, the faster the browser will render the HTML page styles.

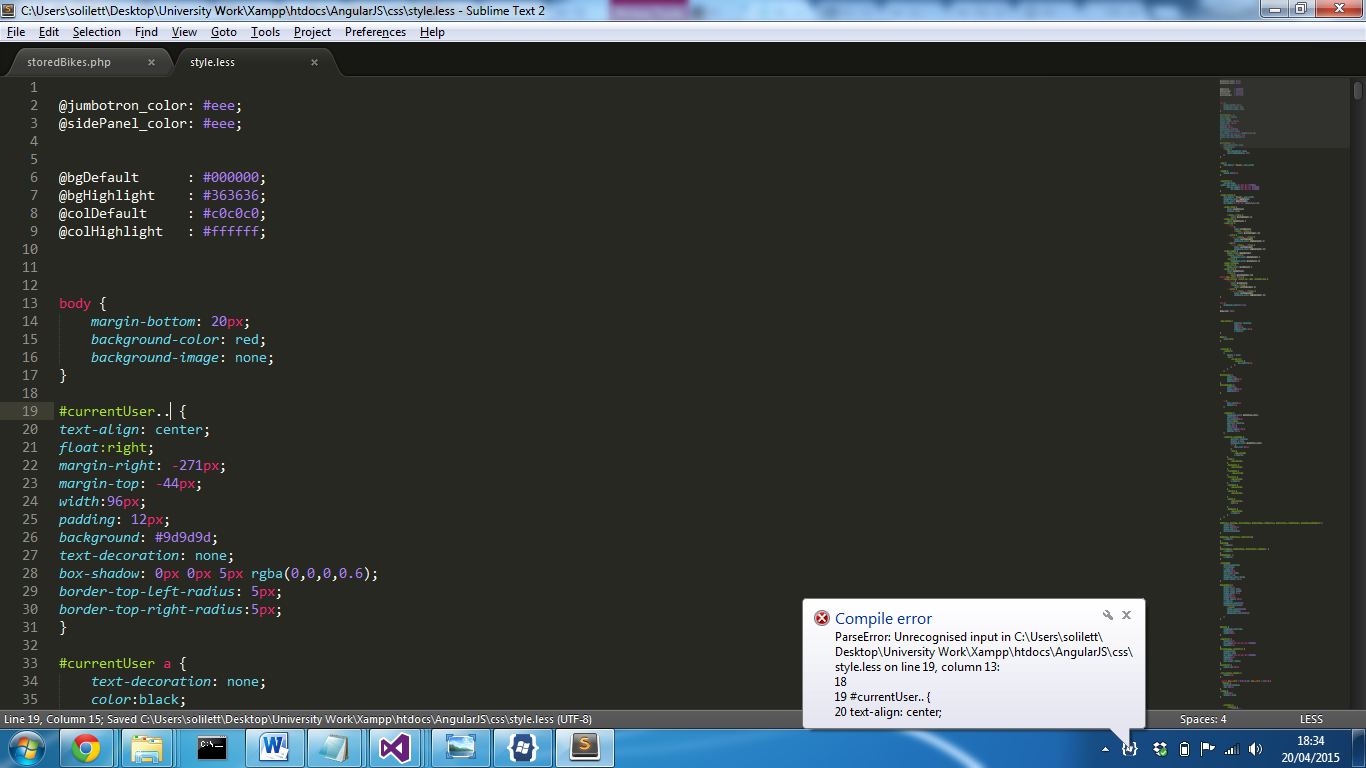
The winless interface



Every time the less file is saved, the application provides the result of the compile. A success compile is displayed in the compiler tab on the application along with the file name and result.



Whenever the application compiles and returns a compile error, a message prompt is displayed displaying the error along with the line on which the error is stored.



**SASS Configuration**

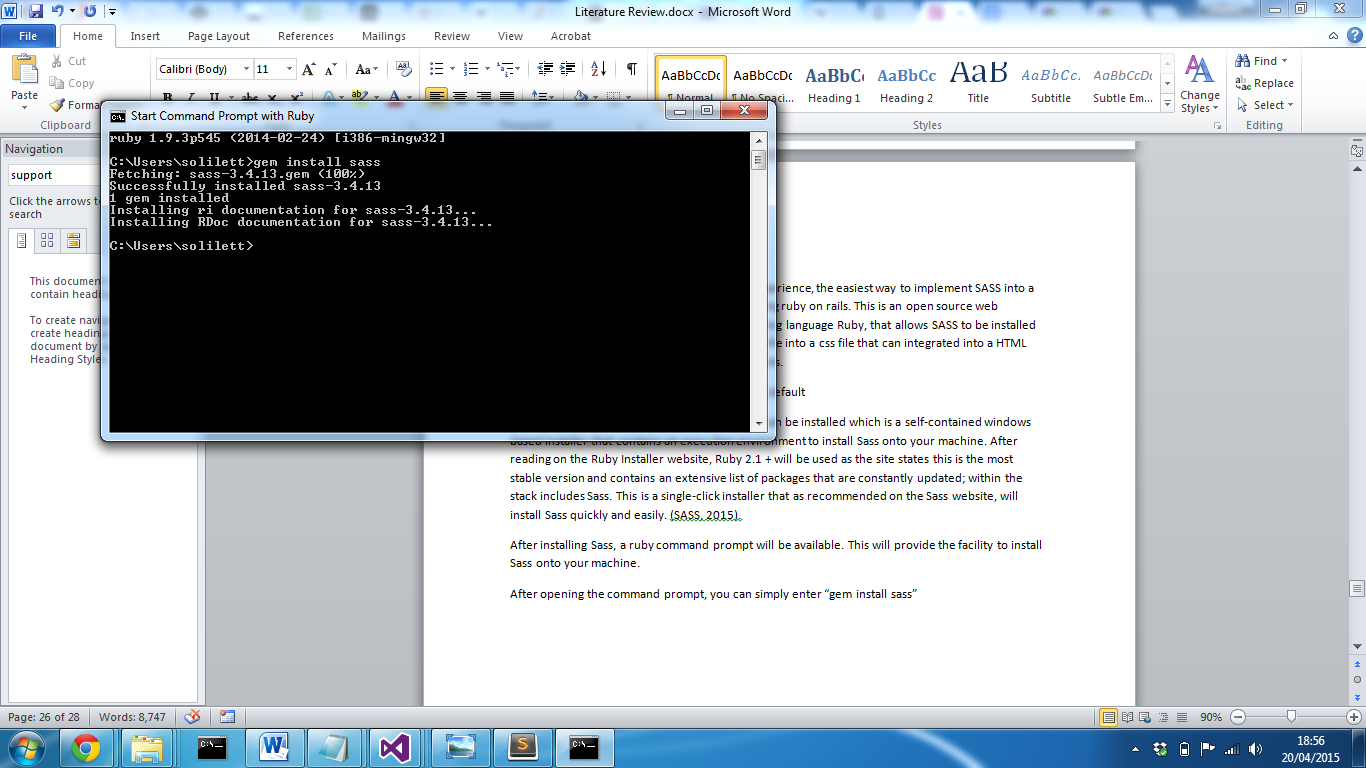
According to several tutorials and developer experience, the easiest way to implement SASS into a HTML project using a text editor, is achieved using ruby on rails. This is an open source web application framework written in the programming language Ruby, that allows SASS to be installed and run on a computer that will compile a SASS file into a css file that can integrated into a HTML project to apply styles and behaviours to elements.

Support: Rails 3.1+ provides support for Sass by default

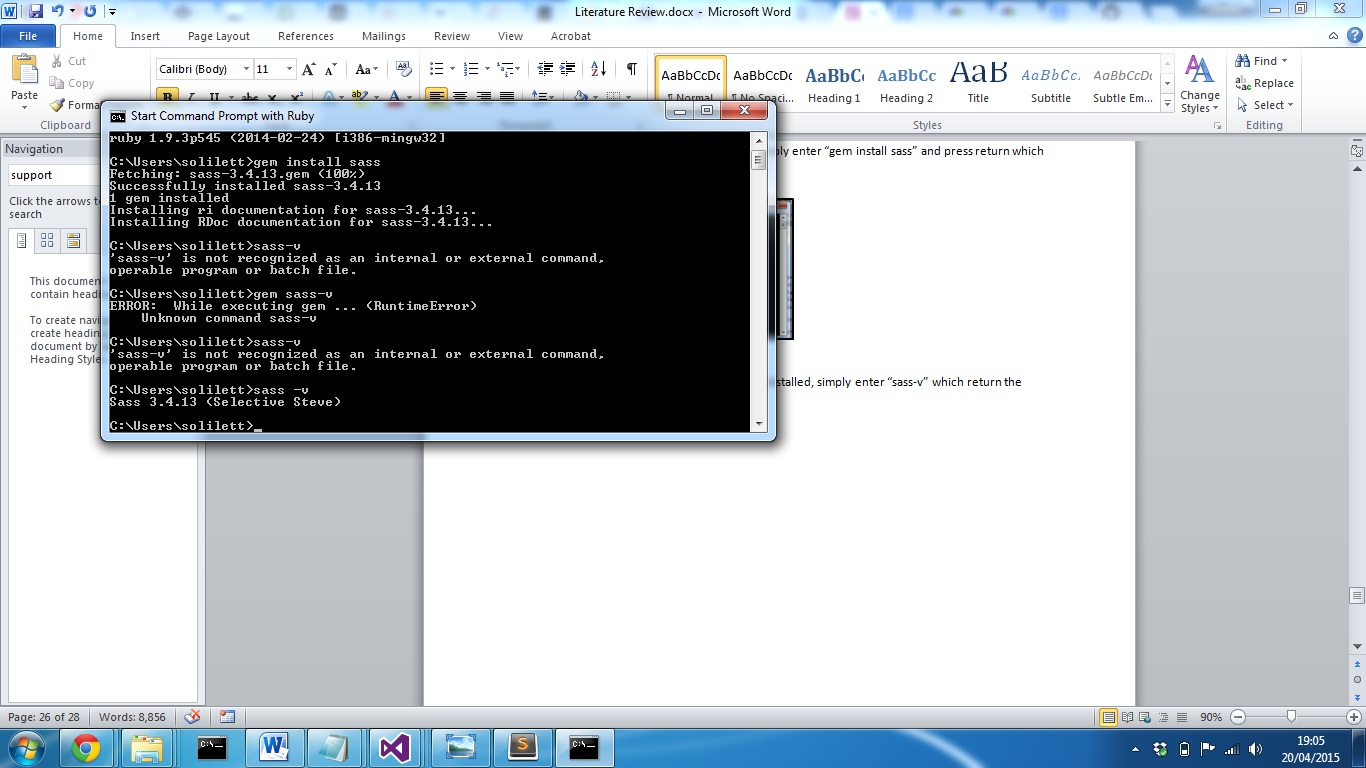
On a windows operating system, Ruby installer can be installed which is a self-contained windows based installer that contains an execution environment to install Sass onto your machine. After reading on the Ruby Installer website, Ruby 2.1 + will be used as the site states this is the most stable version and contains an extensive list of packages that are constantly updated; within the stack includes Sass. This is a single-click installer that as recommended on the Sass website, will install Sass quickly and easily. (SASS, 2015).

After installing Sass, a ruby command prompt will be available. This will provide the facility to install Sass onto your machine.

After opening the command prompt, you can simply enter “gem install sass” and press return which will run the installation process.

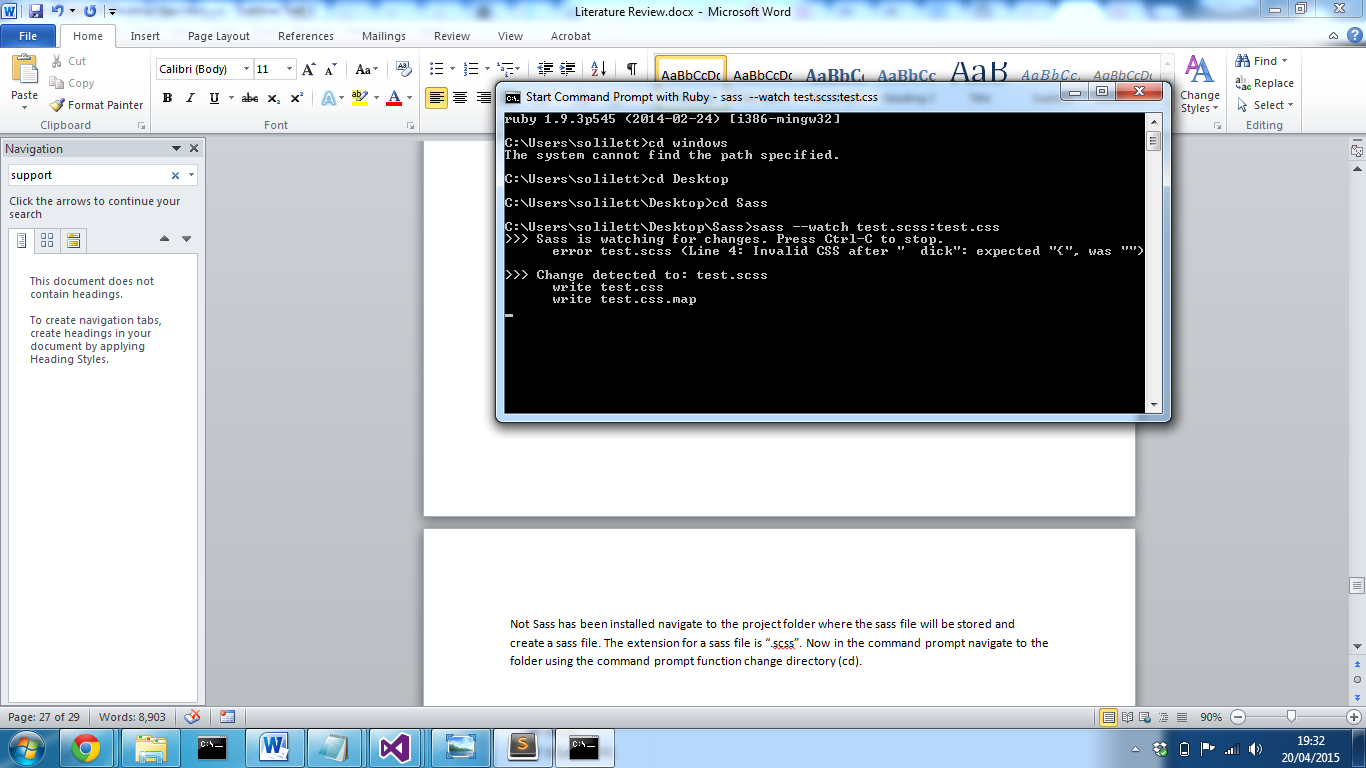


In order to double check if sass has successfully installed, simply enter “sass-v” which return the version of sass installed on your machine.



On initial setup, problems where encountered which led to Sass unable to be installed onto the machine. This was caused of an issue with the ruby installer that had been downloaded. After attempting to resolve the issue and being unsuccessful, the application was uninstalled and a new version was downloaded and installed. The possible cause for the problem could have been the file did not download correctly. After installing the new download, Sass installed with no problems.

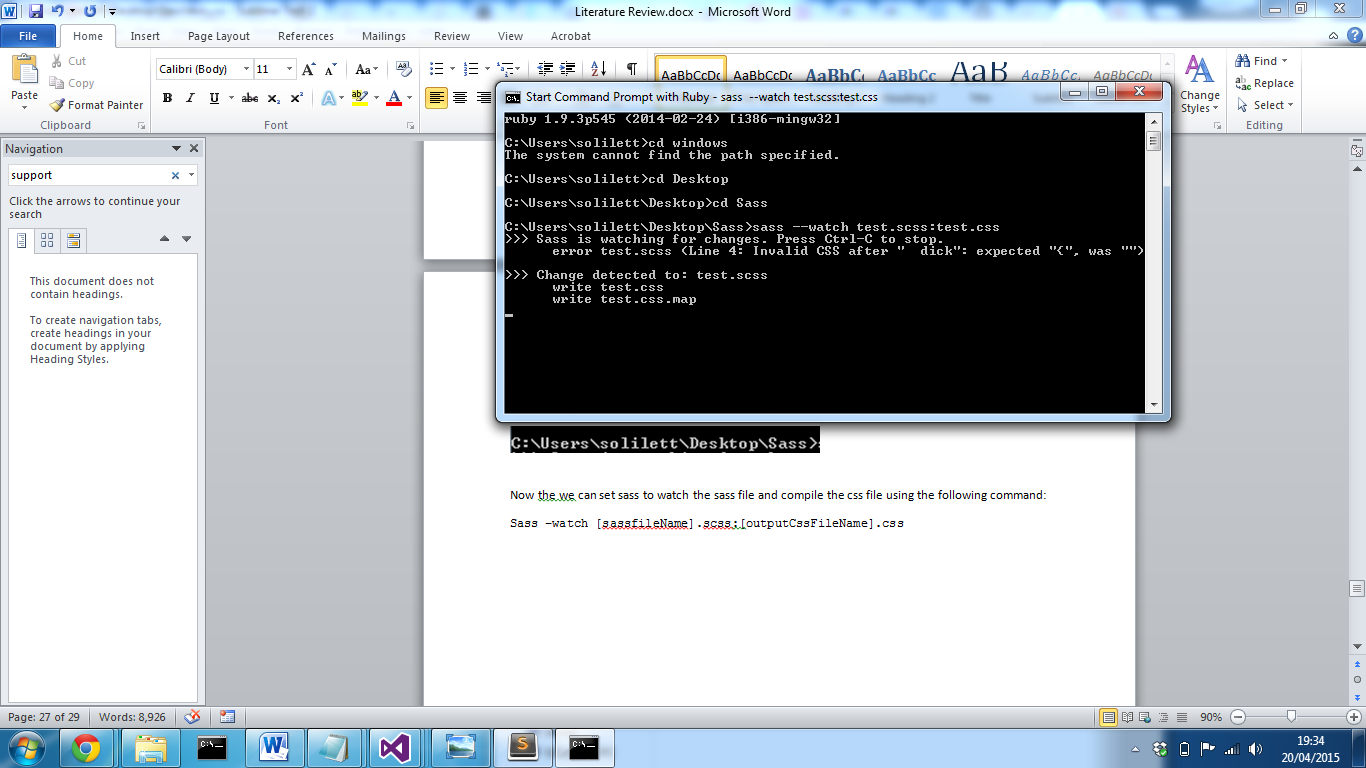
Not Sass has been installed navigate to the project folder where the sass file will be stored and create a sass file. The extension for a sass file is “.scss”. Now in the command prompt navigate to the folder using the command prompt function change directory (cd).

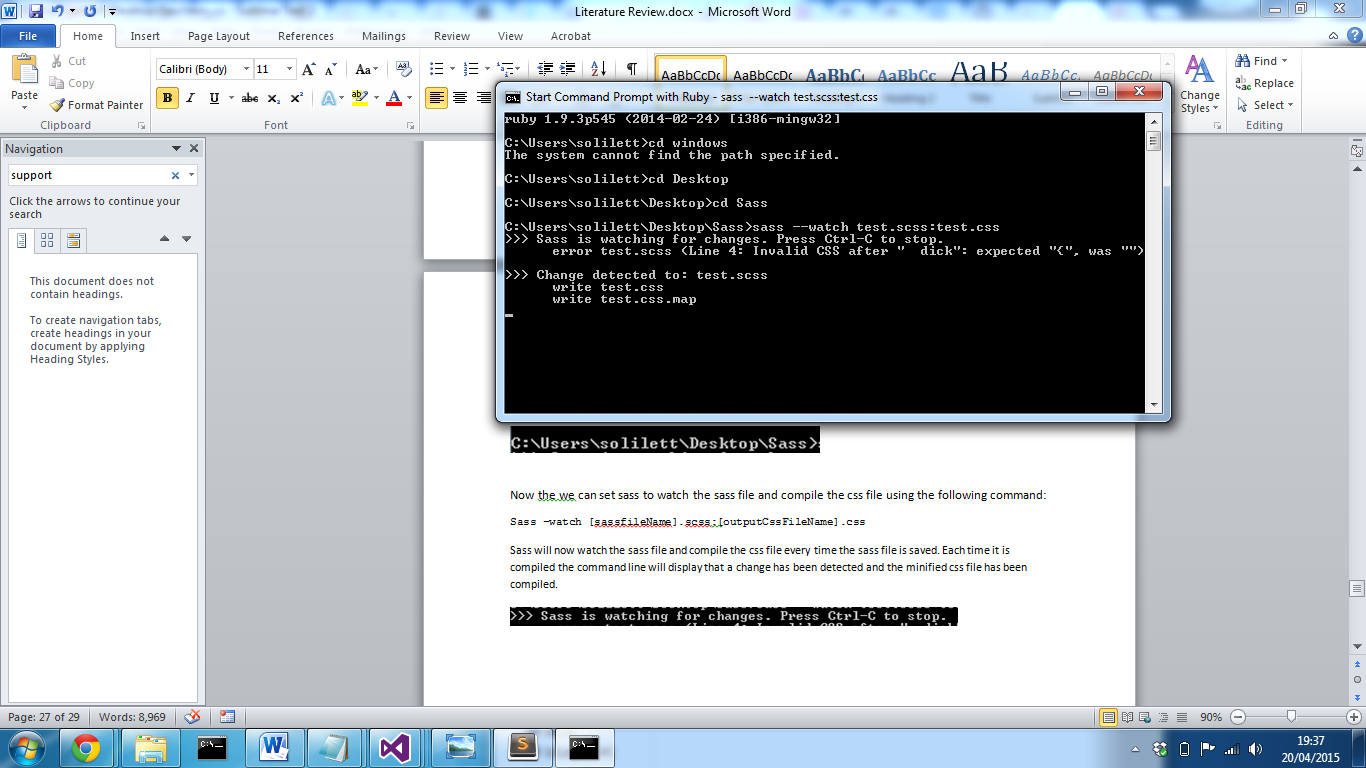


Now the we can set sass to watch the sass file and compile the css file using the following command:

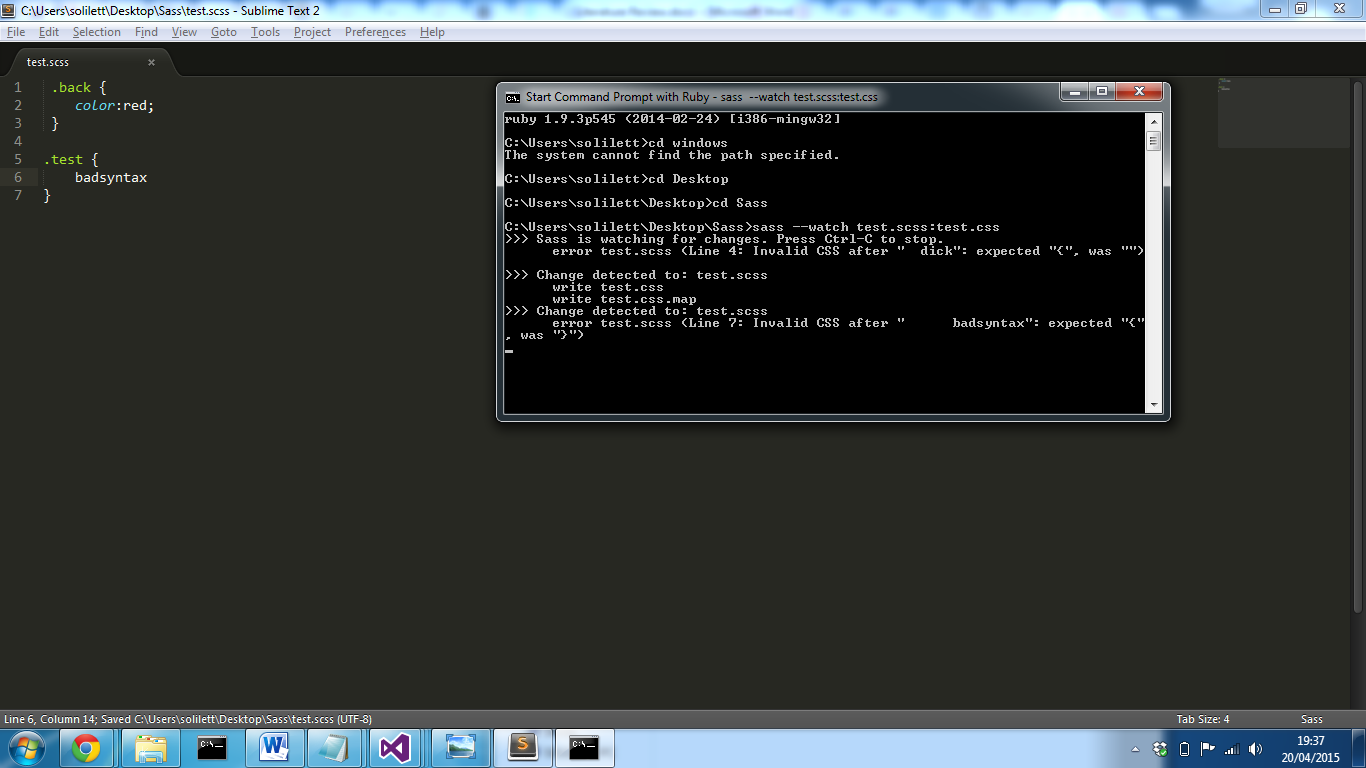
Sass –watch [sassfileName].scss:[outputCssFileName].css

Sass will now watch the sass file and compile the css file every time the sass file is saved. Each time it is compiled the command line will display that a change has been detected and the minified css file has been compiled.





When errors are detected the compiler will provide the line with the error and a possible solution



Sass will continue to watch the file until it is either it is closed or told to stop watching the file. The output css file can then be linked to the html project and the css properties will be applied.

**Appendix 5**

**Practitioner Interview**

Date: 05/03/2015

Participant Name: Mark Rainey

Role: Front End senior Developer

Company: Compare theMarket (Part of the BGL group)

**Discussion Notes:**

**Background of role**

As this is a large IT organisation, each product on their site is separated into teams and each responsibility is separated into logical areas. This can often mean the system structure can differ from product to product, even page to page. This results in a consistent theme for each product, however often both the front end mark up and backend design can be completely different; this is often down to the fact that the organisation offers developers the freedom to developer in an area of their expertise. This results in projects that use some of the latest tools and techniques that meet the requirements set. Mark mentioned a possible solution to the problem, is developer their own unique web application framework that keeps the UI structure and layout consistent across the whole business.

**Responsive Tools:**

Flex box is a layout mode providing for the arrangement of elements to predictably change, when the page layout must accommodate different screen sizes and different display devices. This is a direct alternative to bootstrap grid system; however the concept for responsive design differs slightly. With bootstrap, the grid system works by triggering elements properties to change when a screen or viewport is of a certain size. This is often through the use of media queries and JavaScript functions. Flexbox’s concept evolves around the idea if an element can no longer fit on a screen, it must change its properties and position accordingly through the use of CSS3.   
A key advantage stated by the practitioner about responsive site design was the fact that responsive systems are much easier to maintain. If a change is made it will have effect on all devices rather than if the website is split into two systems, one for mobile and one for desktop, changes would need to be made on each system. It also means that style properties could respond differently on different devices so they make break the consistency of the theme. This is what they stated was called device driven responsive design.   
They found setting up flexbox extremely difficult at first and it took several attempts to get it to work properly

**JQuery vs AngularJS**

During the interview, the practitioner discussed a great benefit of angular.js compared to jQuery that to this point had not been considered. JQuery requires an element to be identified by an attribute node, in order to trigger an event or for an element to be manipulated. For a large firm, this can often mean that a small change like changing a class name in the mark up can result in an a impact in the JavaScript not functioning correctly. Using angularJS a databinding control can be placed inside an element tag that is separate to an attribute node. A front end developer would have no reason to need to alter the control and allows the two areas to work completely separate, with no direct dependencies.

**SASS**

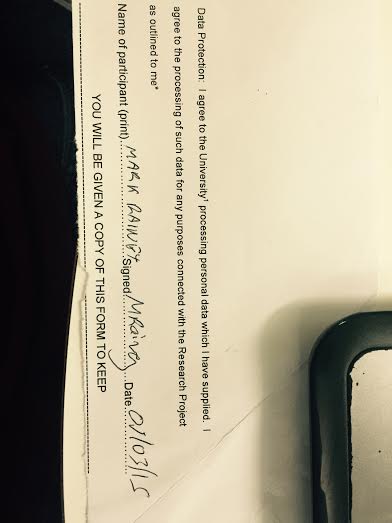
The practitioner used SASS over LESS for no particular reason; SASS was already integrated into the development environment so this was the easier compiler to use. The practitioner emphasised on why not using some sort of CSS extension would be bad practise, and as SASS structures styling identifiers in parent/child structure, the same way it’s structured in a HTML document it logically makes more sense.

**Bootstrap**

The practitioners initially demonstrated the large amount of use bootstrap grid system is used throughout the compare the market product side of the website and explained how well it worked making their site responsive. He also demonstrated how they had implemented a whole section of their site using bootstrap that presented the exact same design as other areas of the site where bootstrap hadn’t been used. He stated how bootstrap works extremely well for a new project or website, but the integration for an established site became extremely difficult and complex. He said the grid system is superb but the attributes required to use the frameworks capabilities completely goes against semantic principles. In a large organisation like theirs, many different developers can often work on one site, and it can become hard if there isn’t clear semantics.   
They also stated how they had used a bootstrap version that style properties and behaviours had been compiled using sass rather than less. This helped the integration into their project environment a lot easier.  
  
**Single page applications**

They try use single page application design with angularJS routing as much as possible, as tests have shown that content loads quicker and provides a better user experience. They also prefer the fact that data isn’t passed around from page to page.

**CDN’s**CDN’s are always used unless they require only a certain sections of a framework or tool. They stated how they had created their own cdn for all images used on all projects within the organisation so all developers have access to the same logos and brand images. This was crucial for them as if they decided to, for example, tweak their logo, they only had to change it in one place, and all areas of the website would then change to the new logo.

**Practitioner Data protection Acceptance**

**Appendix 6**

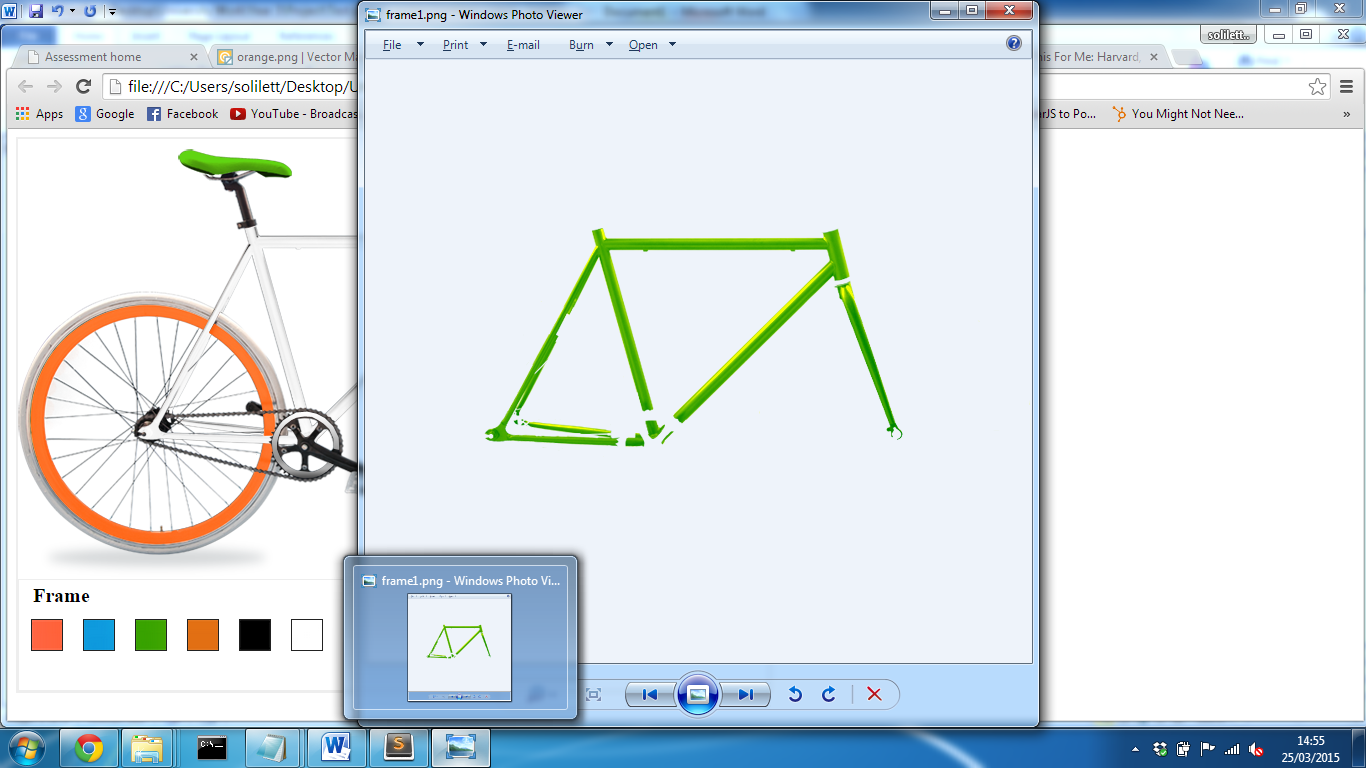
**Proof of Concept**

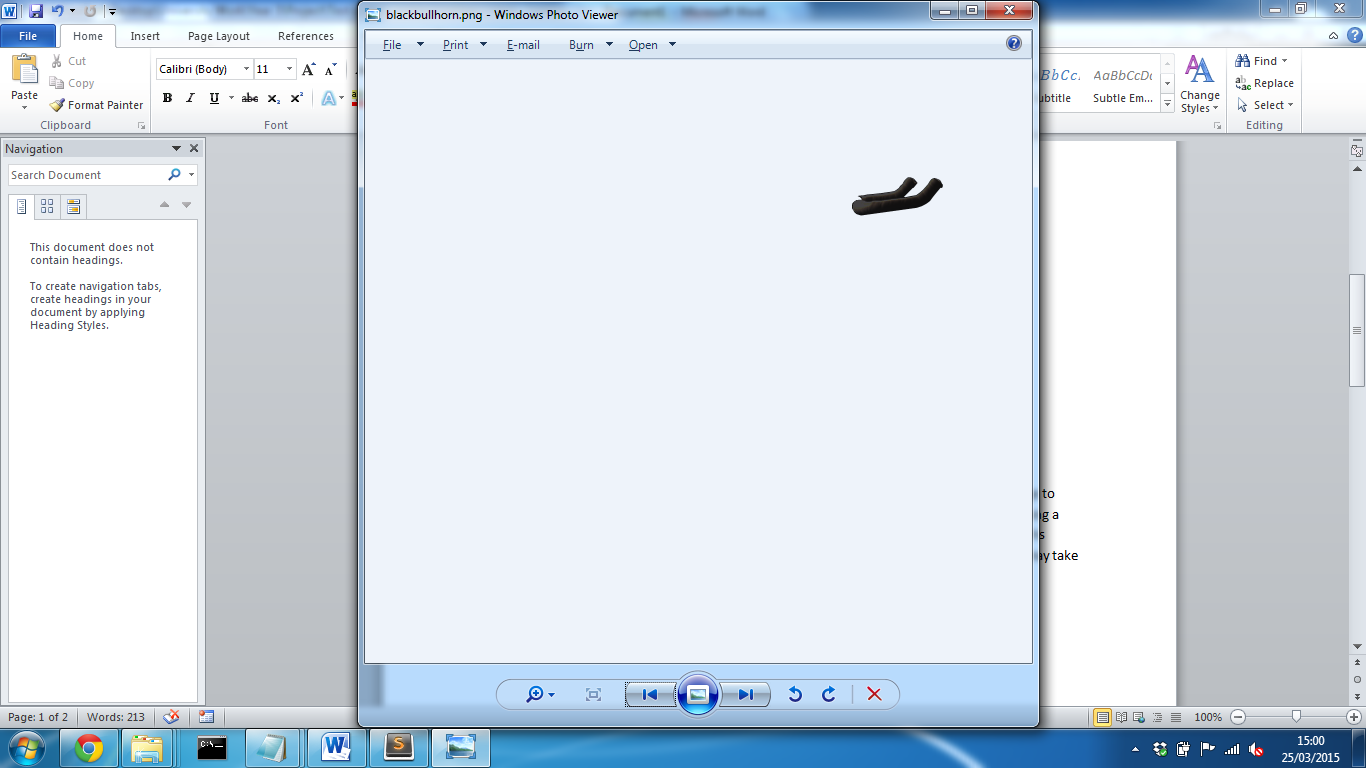
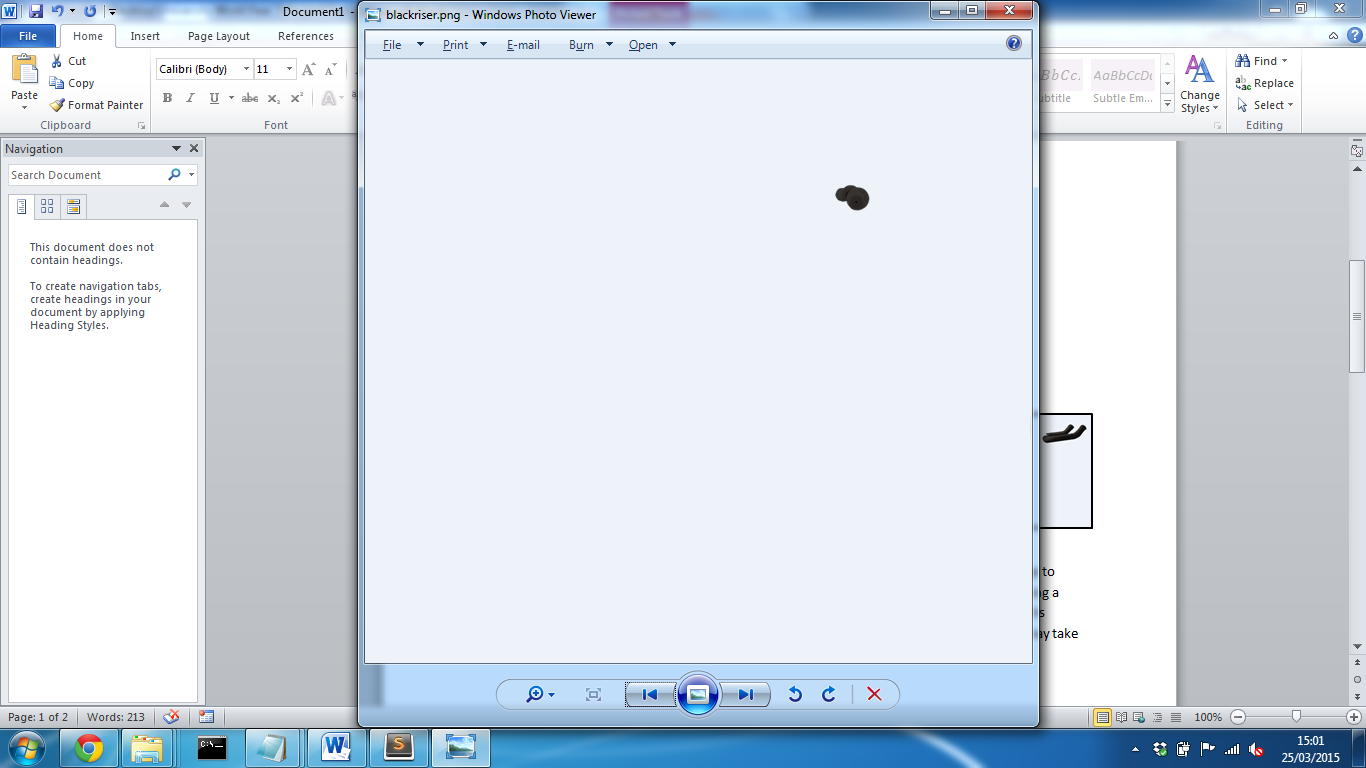
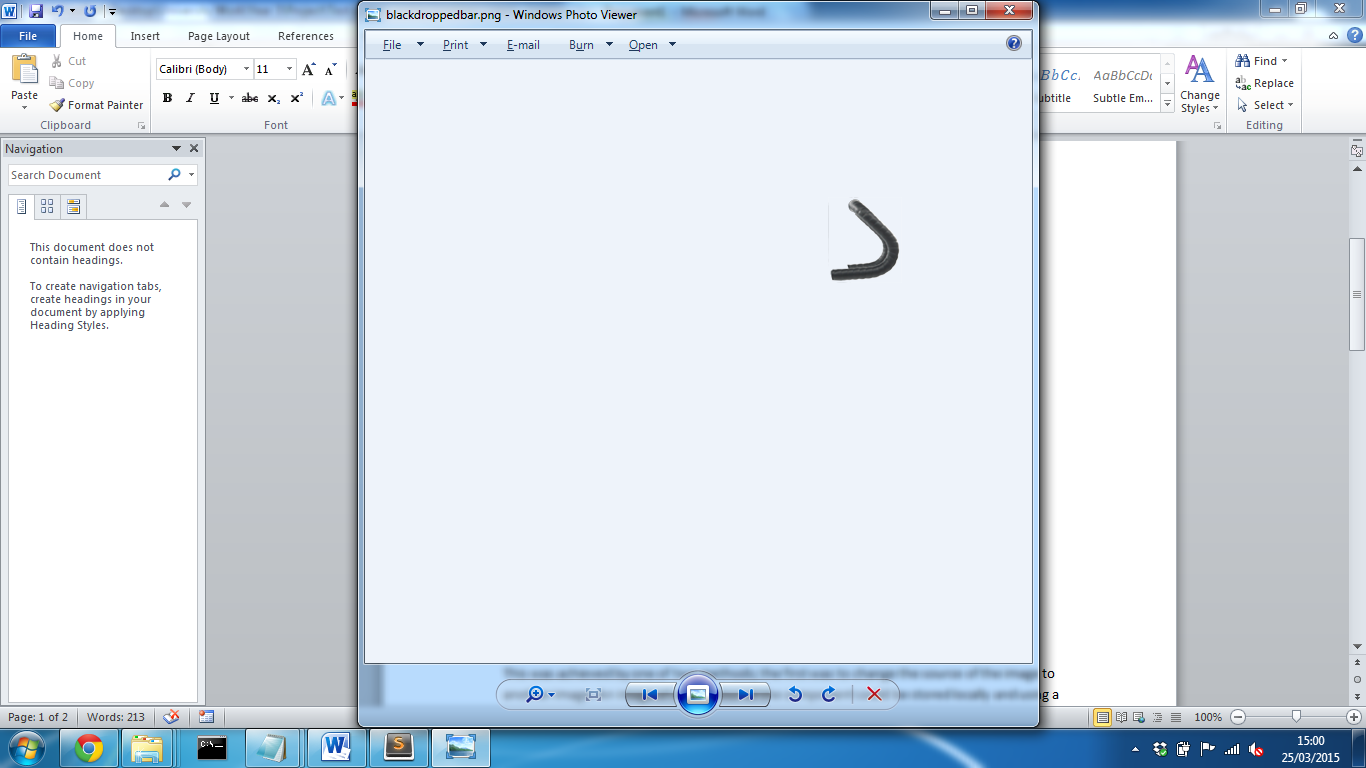
Before development of the final artefact began, the initial stage followed the proof of concept method, which demonstrates the feasibility of the desired system. This method allows a prototype to be created to ensure that the final artefact requirements are reach, but doesn’t provide a deliverable. (Janssen, 2015)  
This meant the first development task was to implement a working solution that allowed an image to be manipulated, providing the illusion the cycle component was changing colour.   
In order to allow each component to be modified, the cycle needed to be split up into each component that the user can interact with. The cycle images consisted of the following images:

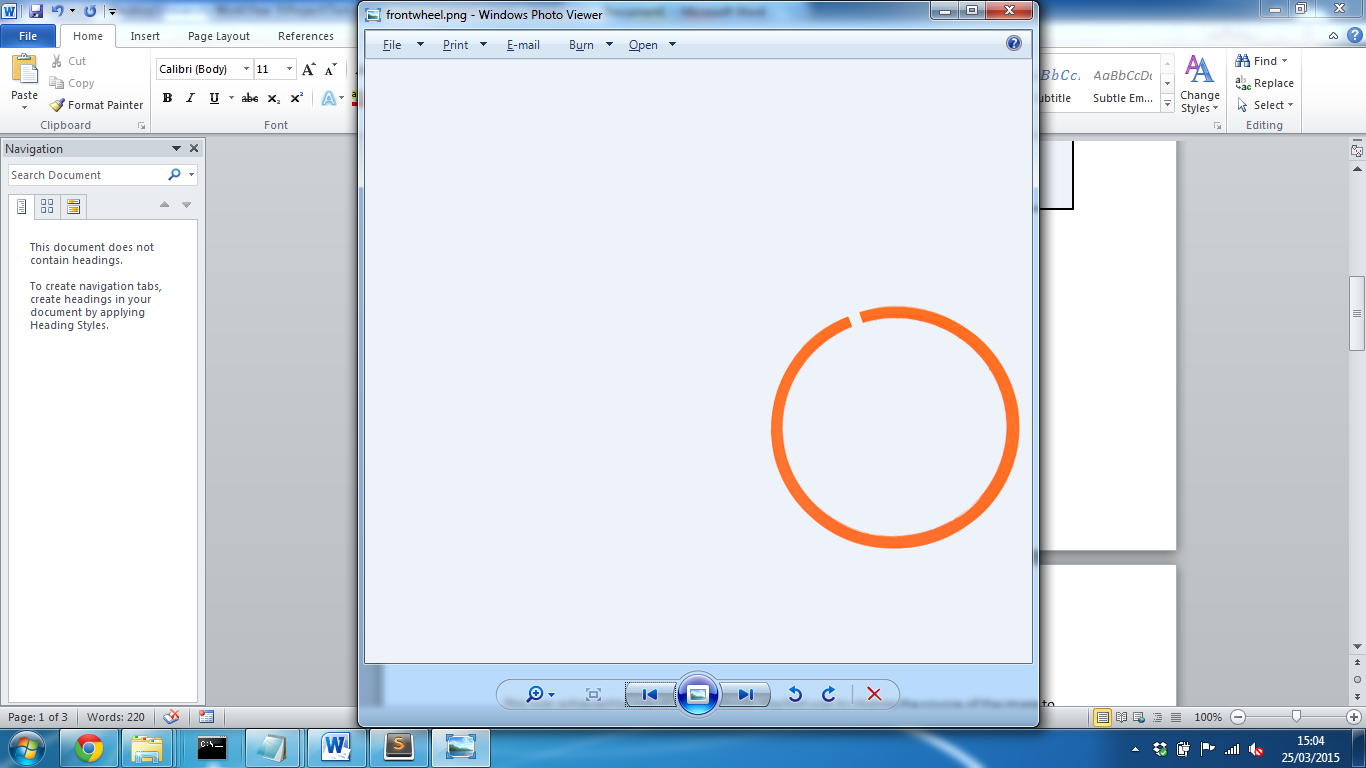
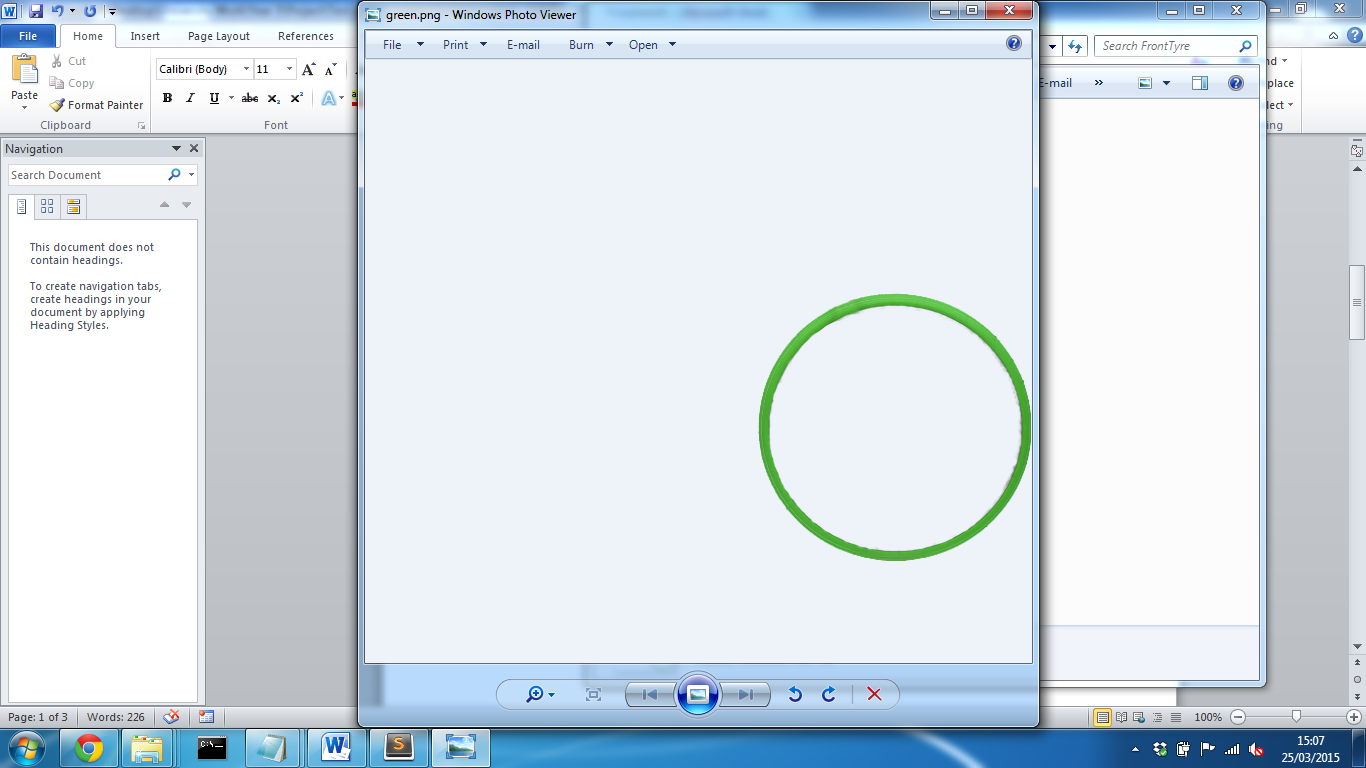
* Master Image



* Frame



* Handlebars - Each type: riser, dropped and bullhorn
* Wheels and Type (Front Tyre)



Each component was edited using Adobe Photoshop and then cut from the entire master image. It was the placed onto a blank canvas with the same dimensions as the master image in the same position. Photoshop allows for part of the image to be traced based on its colour, in perspective to the surrounding colours i.e. Background. After selecting the image can then be placed into a new canvas and Photoshop will automatically place the selected image in the same location as the other canvas, if the new canvas is of the same dimensions. Each image could then be masked and a new colour could then be overlayed on top; this meant that the colour could be changed and the shades of the original image could still be present to create a realistic image. Each image was saved as a PNG, which allows the background to be transparent.

After each component had been extracted from the master and saved in a structured folder layout, the cycle could then be implemented into a webpage. In order to create the illusion that all components are one image of a cycle, they needed to be layered on top of one another.

<img id="master" class=”component” src="bike-master.png" />

<img id="frame" class=”component” src="frame/green.png"/>

<img id="frontwheel" class=”component” src=”frontWheel/orange.png" />

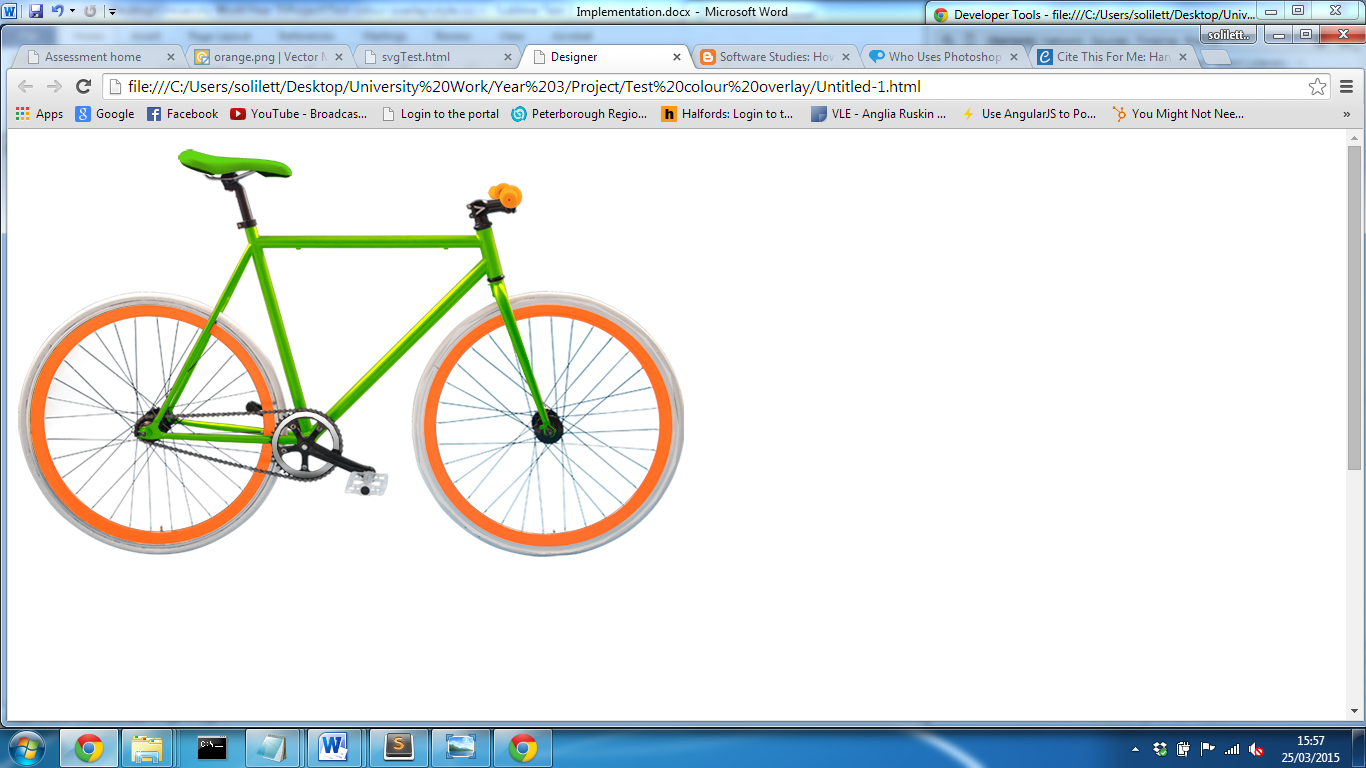
<img id="frontTyre" class=”component” src=”frontTyre/white.png" />

<img id="rearwheel" class=”component” src="rearwheel/orange.png" />

<img id="rearTyre" class=”component” src="rearTyre/white.png" />

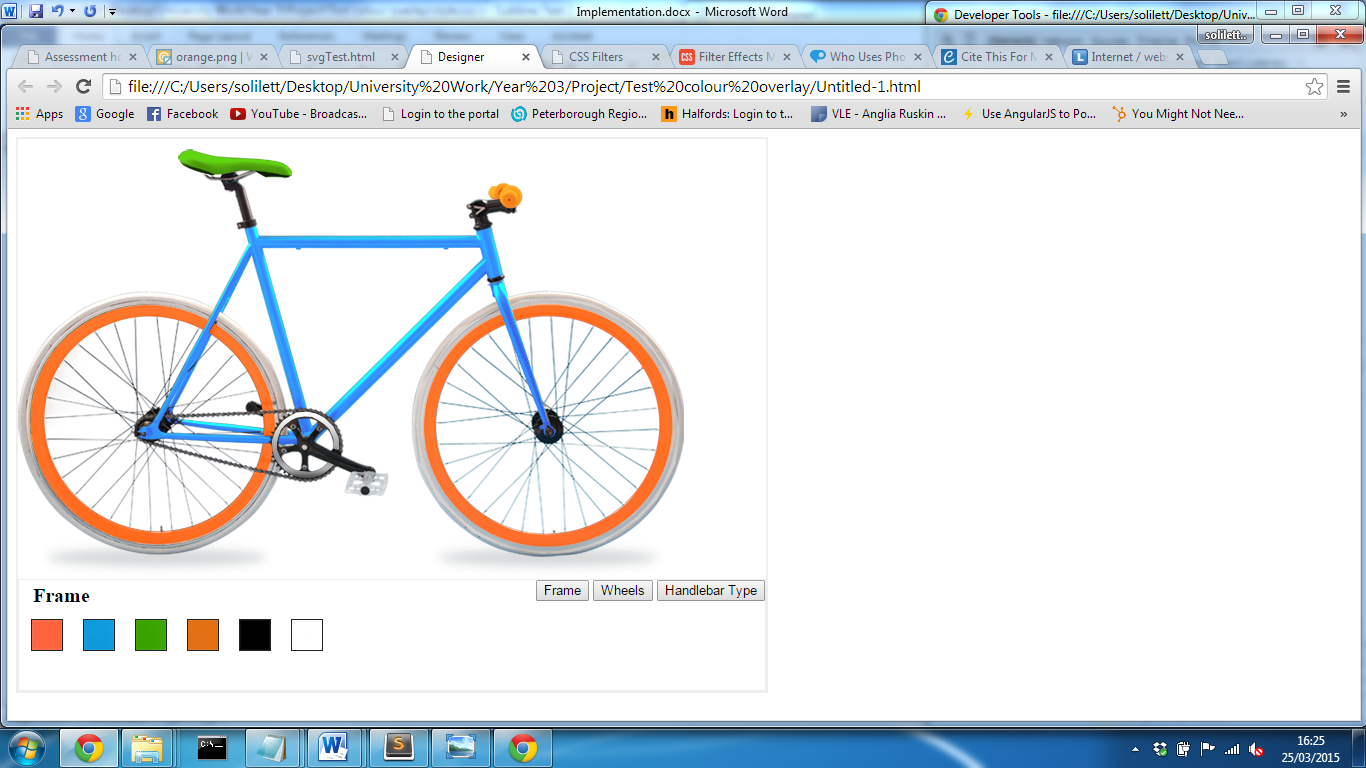
<img id="handlebar" class=”component” src="riser/orange.png" />

<img id="saddle" class=”component” src="saddle/green.png" />



The class “component” was then set to “position:absolute” which positions all components in the same place. The z-index of each component could then be set so that the master was at the back and components where then layered on top.

The next task to be implemented was the user interaction to allow a component colour to be changed. This was achieved by one of two methods; the first was to change the source of the image to another image. Using a JavaScript library the image source could be modified which would display a different colour component. While this was a feasible option, this meant that a large number images would need to be stored; it also meant that content may take some time to load.   
The second option was to use filter effects; these are a way of processing an element’s rendering before it’s displayed to the user. (Dev.w3.org, 2015) The Hue rotation filter allows for a degrees parameter to be specified between 0 and 360; any value between these values will then render a different colour to the user. The hue rotation value can be applied to a specified element through the css.

**CSS**

#frame {

webkit-filter:hue-rotate(120deg);

}

By setting the hue rotate parameter to 120deg allows the frame to transform from green to blue. While this works extremely well for a multitude of colours, black and white colours cannot be achieved. Although other filters are available, one filter alone will not achieve these colours and only one filter can be applied to an element. In order to overcome this problem, both methods can be used; this will mean that when black or white is selected the image source change and when any other colour is the hue rotation filter will be applied. While this reduces the amount files stored, user experience could be affected as the user may notice the time taken for the image to render. This method would be used in the final solution as when changing the handle bar type, there would be no other option.

After accomplishing a method to manipulate the colour, a function needed to be implemented that allowed for user interaction. In order to achieve this, a set of colour tiles were displayed to user and jQuery library was used to handle events. Using the jQuery library, a large array of pre-determined JavaScript functions are accessible; while JavaScript alone would be suitable, based on the developer’s previous experience, jQuery required far less lines of code.

$("#frameTileBlue").click(function(){

$("#frame").attr("src","frame/orange.png");

$("#frame").css("-webkit-filter","hue-rotate(120deg)");

});  
…

Using the jQuery library, a function can be specified based on a certain event. The above code will, on the click of an image with the id of “frameTileBlue”, will run a function to set the source of the images with the id “frame” to orange.png and apply the hue rotate filter to turn the frame blue.

To see implementation of the proof of concept prototype find video “proof of concept.avi” in videos folder.

**Bibliography**

Anon, 2015. *5 reasons NOT to use Twitter Bootstrap*. [online] Zing Design. Available at: <http://www.zingdesign.com/5-reasons-not-to-use-twitter-bootstrap/> [Accessed 15 Apr. 2015].

Anon, 2015. *Advantages of using pure JavaScript over JQuery*. [online] Available at: <http://programmers.stackexchange.com/questions/166273/advantages-of-using-pure-javascript-over-jquery> [Accessed 1 Apr. 2015].

Anon, 2015. *Bootstrap, from Twitter 1.1.0 Documentation*. [online] Available at: <http://bootstrapdocs.com/v1.1.0/docs/#about> [Accessed 14 Apr. 2015].

Anon, 2015. *Build software better, together*. [online] GitHub. Available at: <https://github.com/search?q=stars:%3E1> [Accessed 20 Apr. 2015].

Anon, 2015. *Feature Driven Development (FDD) and Agile Modeling*. [online] Available at: <http://www.agilemodeling.com/essays/fdd.htm> [Accessed 14 Apr. 2015].

Anon, 2015. *How browsers work*. [online] Available at: <http://taligarsiel.com/Projects/howbrowserswork1.htm#The\_order\_of\_processing\_scripts\_and\_style\_sheets> [Accessed 14 Apr. 2015].

Anon, 2015. *HTML5 Semantic Elements*. [online] Available at: <http://www.w3schools.com/html/html5\_semantic\_elements.asp> [Accessed 14 Apr. 2015].

Anon, 2015. *HTML DOM Element Objects*. [online] Available at: <http://www.w3schools.com/jsref/dom\_obj\_all.asp> [Accessed 14 Apr. 2015].

Anon, 2015. *Less vs Sass? It’s time to swtich to Sass*. [online] Zing Design. Available at: <http://www.zingdesign.com/less-vs-sass-its-time-to-switch-to-sass/> [Accessed 20 Apr. 2015].

Anon, 2015. *Sass: Sass Basics*. [online] Available at: <http://sass-lang.com/guide> [Accessed 15 Apr. 2015].

Anon, 2015. [online] Panlogic Digital Enginerring. Available at: <http://www.panlogic.co.uk/wp-content/uploads/2011/11/Our-approach-to-prototypes-and-proofs-of-concept.pdf> [Accessed 14 Apr. 2015].

Anon, n.d. Built-in Browser Support for Responsive Images. [online] Available at: <http://www.html5rocks.com/en/tutorials/responsive/picture-element/> [Accessed 16 Apr. 2015].

Anon, n.d. Welcome to Twitter - The people, topics and conversations you care about. Available at: <https://twitter.com/> [Accessed 16 Apr. 2015].

Coad, P.,, Lefevre, E., and DeLuca, E., 1999. *Java Modeling in Color with UML: Enterprise Components and Process*. 1st ed. United Kingdom: Prentice Hall PTR.

Dubois, J., 2015. *Introduction to JQuery*. [online] Available at: <http://www.oracle.com/technetwork/java/intro-jquery-166845.html> [Accessed 27 Mar. 2015].

Frain, B., 2012. *Responsive web design with HTML5 and CSS3 learn responsive design using HTML5 and CSS3 to adapt websites to any browser or screen size*. Birmingham: Packt Publishing Ltd.

Gremillion, B., 2015. *Smashing Magazine — For Professional Web Designers and Developers*. [online] Available at: <http://www.smashingmagazine.com/> [Accessed 27 Mar. 2015].

Harris, J.B.,, Eichora, F.L.,, Goodwin, D.L., and Henson, J.L., 1997. *Use of Rapid Application Development Techniques*. [online] Available at: <http://www.dtic.mil/dtic/tr/fulltext/u2/a331871.pdf> [Accessed 14 Apr. 2015].

Held, G., 2010. *A practical guide to content delivery networks*. Boca Raton: CRC Press.

jquery.org, jQuery F.-, 2015a. *About jQuery UI*. [online] Available at: <http://jqueryui.com/about/> [Accessed 1 Apr. 2015].

jquery.org, jQuery F.-, 2015b. *jQuery API Documentation*. [online] Available at: <http://api.jquery.com/> [Accessed 26 Mar. 2015].

jquery.org, jQuery F.-, 2015c. *jQuery Foundation*. [online] Available at: <https://jquery.org/> [Accessed 26 Mar. 2015].

jquery.org, jQuery F.-, 2015d. [online] Available at: <http://jqueryui.com/> [Accessed 1 Apr. 2015].

Karlsson, J., 2014. *Responsive web design with CSS frameworks*. Sweden: Uppsala Universitet.

Kramer, J., 2014. Responsive Design Frameworks: Just Because You Can, Should You? *Smashing Magazine*. 14 Feb.

Kyrnin, J., n.d. Avoid Inline Styles for CSS. *About Tech*. [online] Available at: <http://webdesign.about.com/od/css/a/aa073106.htm> [Accessed 20 Apr. 2015].

Lagendijk, M., 2015. *Windows GUI for less.js*. [online] Available at: <http://winless.org/> [Accessed 20 Apr. 2015].

Martin, J., 1991. *Rapid application development*. New York: Maxwell Macmillan International.

Meeker, M., 2008. *Technology / Internet Trends*. Morgan Stanley.

Perez, S., n.d. Majority Of Digital Media Consumption Now Takes Place In Mobile Apps. [online] Available at: <http://techcrunch.com/2014/08/21/majority-of-digital-media-consumption-now-takes-place-in-mobile-apps/> [Accessed 27 Mar. 2015].

Swedberg, K., and Chaffer, J., 2013. *Learning Jquery Fourth Edition*. United States: Packt Publishing.

Tao, D., 2013. *Bootstrap without all the debt*. [online] Available at: <https://coderwall.com/p/wixovg/bootstrap-without-all-the-debt> [Accessed 15 Apr. 2015].

Team, T.C.L., 2015. *Getting started | Less.js*. [online] Available at: <http://lesscss.org/> [Accessed 14 Apr. 2015].

Wellman, D., 2009. *JQuery UI 1. 6: The User Interface Library for JQuery*. Birmingham: Packt Publishing.

Wilde, B., 2015. *JQuery vs. JavaScript: What’s the Difference Anyway?*[online] Available at: <http://blog.udemy.com/jquery-vs-javascript/> [Accessed 27 Mar. 2015].

ZURB, and inc. ZURB network also includes zurb.com 2015a. *Abide Validation | Foundation Docs*. [online] Available at: <http://foundation.zurb.com/docs/components/abide.html> [Accessed 16 Apr. 2015].

ZURB, and inc. ZURB network also includes zurb.com 2015b.*Interchange | Foundation Docs*. [online] Available at: <http://foundation.zurb.com/docs/components/interchange.html> [Accessed 16 Apr. 2015].