

VDWP PROJECT

MA Digital Design Media

Lloyd Stevens

Birkbeck University | 2018

Table of Contents

INTRODUCTION.....	3
COMMON SITE FEATURES	4
STYLESHEETS.....	4
LAYOUT	5
<i>Grid / Responsive Design.....</i>	5
HIDING AND SHOWING CONTENT.....	16
SECTIONS.....	17
WRAPPER	17
CURRENT USER LOGIN	18
LOGO / BANNER	18
NAVIGATION.....	19
<i>Desktop Navigation</i>	19
<i>Responsive Navigation.....</i>	19
<i>Call to action buttons.....</i>	24
PAGE INTROS	25
<i>Accessibility</i>	25
FOOTER	26
<i>Footer Links.....</i>	26
<i>Social Media Links.....</i>	26
<i>Site Map.....</i>	27
SSI.....	28
JAVASCRIPT.....	29
<i>Efficiency of code.....</i>	29
<i>Separation of code.....</i>	29
CHAT FEATURE.....	30
SVG IMAGES	32
PAGES	33
INDEX (HOME)	33
<i>Sections</i>	33
<i>Accessibility</i>	34
NEWS & EVENTS	35
<i>Sections</i>	35
<i>Accessibility</i>	37
CURRICULUM.....	38
<i>Sections</i>	38
<i>Accessibility</i>	38
EMPLOYER ENGAGEMENT	40
<i>Sections</i>	40
<i>Accessibility</i>	40
ABOUT US	41
<i>Sections</i>	41
ACCESSIBILITY	54
<i>HTML5 and Form Accessibility.....</i>	54
<i>Field Sets</i>	54
<i>Lists</i>	54
<i>Placeholders</i>	54
<i>Labels</i>	54
<i>Date</i>	54
<i>JavaScript Jump Menu</i>	54
TESTING	55
ISSUE WITH MENU LAYOUT	55
SPEED TESTING.....	57

<i>Pingdom.com</i>	57
<i>PageSpeed Insights - Google</i>	57
HTML VALIDATION.....	59
<i>Date Warning on About Us Page</i>	59
CSS VALIDATION	59
ACCESSIBILITY	60
ARIA.....	60
JS	60
LINE HEIGHT	60
BACKGROUND IMAGES AND COLOUR CONTRAST ISSUES.....	60
NAVIGATION AND CONTRAST ISSUES.....	63
NAVIGATION BURGER MENU	63
TAB INDEX	63
EVALUATION AND FURTHER IMPROVEMENTS	65
<i>Minification</i>	65
<i>Improvement of Code</i>	65
BIBLIOGRAPHY	66

Introduction

This is the documentation to accompany the website developed for the VDWP project as part of the MA Digital Design Media course at Birkbeck University, London.

The project is based upon a school provision for vocational learning at Central Foundation Boys School, Islington. The site is a promotional site for the provision at the school, which is titled "Silicon Hack."

The site can be found at the following location on the dreamhost server assigned at the beginning of the module

<http://lsteve07.wdux1718.bbkweb.org/index.shtml>

The Image submission for this project can be found here

http://lsteve07.wdux1718.bbkweb.org/VDWP_Photoshop/

To begin with the site was adapted from the use of code from the previous web project WDUX, from the MA Digital Design Course, submitted by the same Author.

Some code was removed from the original two-page prototype developed from the last project, other parts of the code were reused and adapted. The menu bar and logo were two such areas that were adapted for this version of the site.

Common Site Features

Stylesheets

- There are three stylesheets applied to this site
- Google Font style sheet
 - This has been taken from googles font API and has been used to apply the motserrat font style to the text on the site
 - https://developers.google.com/fonts/docs/getting_started
- Main.css
 - This contains all the unique styling for the site
 - It comes after normalize so that it overrides any styling within this sheet
- Normalize.css
 - This is a corrective style sheet applied to remove the browser defaults that can cause issue
 - I used the normalize.css provided by Nicholas Gallagher at the following location
 - <https://necolas.github.io/normalize.css/8.0.0/normalize.css>
 - Different browsers have different defaults for the CSS
 - Using normalize.css ensures that the CSS is reset for styles that may need this and new formats can be applied
 - It means that development is that little bit easier as you are safe in the knowledge that the styles you are seeing should be reflected across each browser on the whole
 - It doesn't reset everything, only the areas that may produce inconsistencies across the various browsers
 - I chose this over Eric Myers reset as this would involve then reapplying new styles to every element. Normalize saves some of the time on this as it doesn't reset everything
 - Also normalize takes into account HTML5 whereas many of the older reset styles would not have done so as they were created prior to HTML5.
 - <http://nicolasgallagher.com/about-normalize-css/>
 - <https://meyerweb.com/eric/tools/css/reset/>

Layout

Grid / Responsive Design

- The site has been designed using a gridlayout for the most part to support the appearance and structure
- There has also been use of flexbox in places within grids
- There is on occasion nested grids
- Where required there has been use of the box model to further enhance the layout of elements within the site
- Below is a description that outlines the steps involved to setup the layout
- It also refers to previous versions of the site, these can be found in the respective folders on the server for the final site
 - http://lsteve07.wdux1718.bbkweb.org/previous_versions/

Responsive Design 1

- To begin with I used the code developed for the WDUX part of the MA Digital Media Design course that was previously submitted by myself earlier in the year
- I started by utilising much of the original code for the Logo (which is marked as banner as an ID) and also the navigation of the site
- From here my aim was to firstly recode the navigation and header section of the site so that it was mobile responsive starting
- I researched the various breakpoints that could be used
 - <http://www.websitedimensions.com/>
- I initially decided to go with two break points for the overall design of the site, one for desktop and then one for mobile
- I setup the code to hide the links, apply now and also the site logo when it reached the first breakpoint
- This was then replaced with the title {SH}
- Originally I had created an icon for this but then realised that it could easily be replicated using CSS styling instead
- At this point I wanted to make the navigation disappear and a burger menu to appear in its place
- I also wanted some form of drop down menu
- I did some research to this then added in a dropdown menu using following source
 - https://www.w3schools.com/howto/howto_js_dropdown.asp
 - The styling on this was not ideal at this point but the drop down functional
 - This version can be seen within VDWP v3

```
<div class="dropdown">
<button onclick="myFunction()" class="dropbtn">{ SH }</button>

<div id="myDropdown" class="dropdown-content">
<ul>
    <li><a href="#" id="active">Home</a></li>
    <li><a href="#news">News & Events</a></li>
    <li><a href="curriculum.html">Curriculum</a></li>
    <li><a href="about_us.html">About Us</a></li>
</ul>
</div>
</div>
```

- JavaScript for dropdown

```
/* When the user clicks on the button,
toggle between hiding and showing the dropdown content */
function myFunction() {
    document.getElementById("myDropdown").classList.toggle("show");
}

// Close the dropdown menu if the user clicks outside of it
window.onclick = function(event) {
    if (!event.target.matches('.dropbtn')) {

        var dropdowns = document.getElementsByClassName("dropdown-
content");
        var i;
        for (i = 0; i < dropdowns.length; i++) {
            var openDropdown = dropdowns[i];
            if (openDropdown.classList.contains('show')) {
                openDropdown.classList.remove('show');
            }
        }
    }
}
```

Responsive Design 2

- On reflection I wasn't happy with the drop down from a design and UX perspective and wanted to attempt a different approach
- I decided that to move on from here I would first need to remake much of the site layout as it was proving difficult to work backwards
- I thought it best to look into Grid Layout to utilise the responsive of this design
- I knew from the previous project that grid layout can be set to alter its width dependent on the width of the screen
- I went to the mozilla developer network to get some grid layouts to experiment with
 - https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Realizing_common_layouts_using_CSS_Grid_Layout#A_flexible_12-column_layout
- I utilised the 12 column layout link above, the example of which is seen below
- The border style and background colours below were CSS that came with the mozilla resource
- I found this useful to keep within the CSS and uncomment when testing as it allowed me to easily see where the grid items were without having to go into the developer tools as often within chrome/firefox
- The CSS is still found within the main.css file, this is for both maintainability and also to show the technique to the reader for the purposes of this assignment.
- It would be removed from a live final version of the site

The header

[Nav 1](#)

[Nav 2](#)

[Nav 3](#)

Sidebar

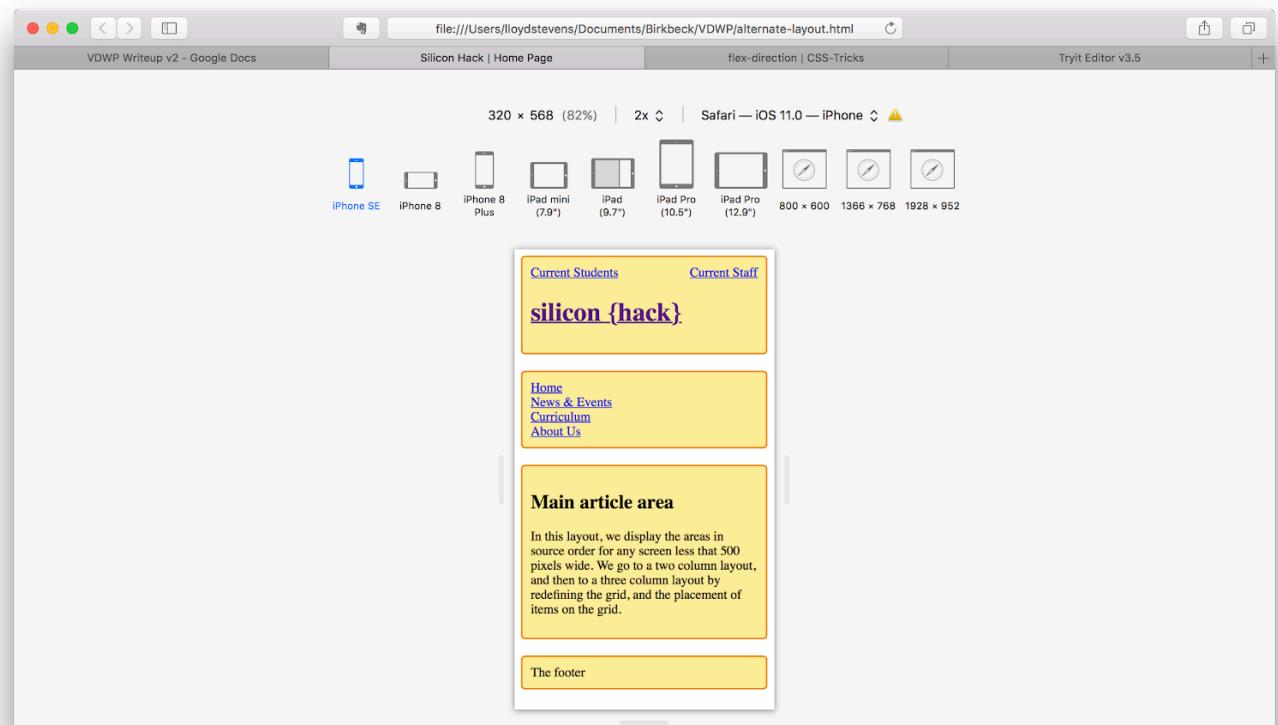
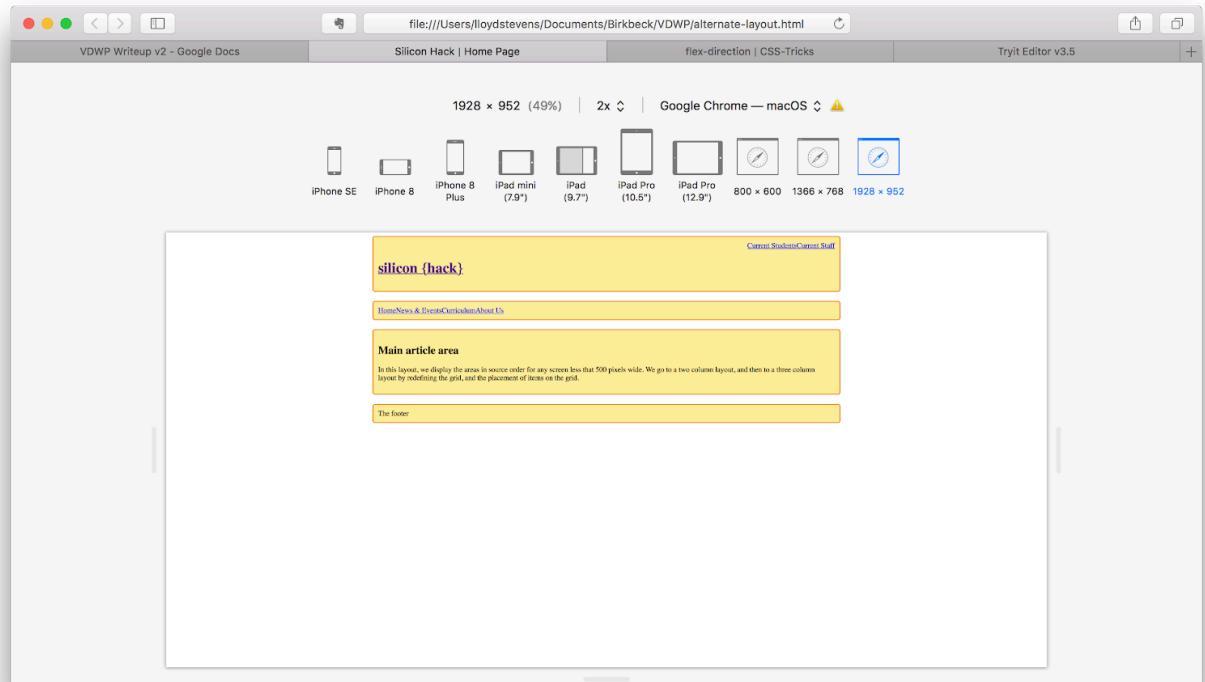
Main article area

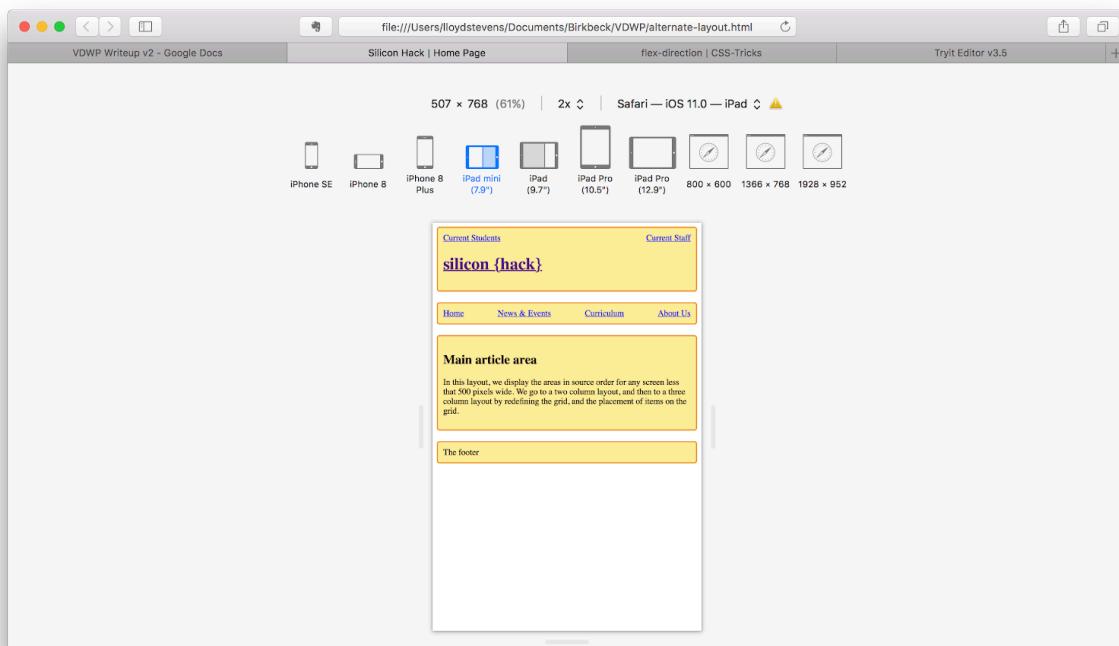
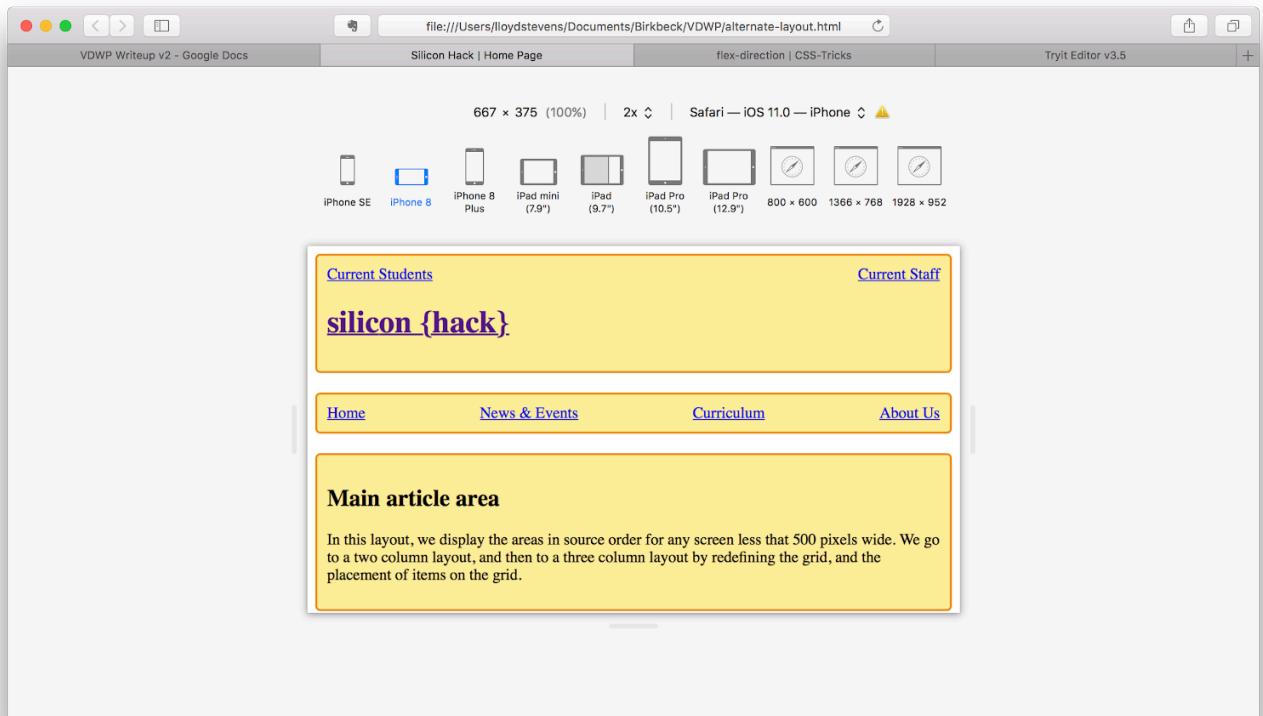
In this layout, we display the areas in source order for any screen less than 500 pixels wide. We go to a two column layout, and then to a three column layout by redefining the grid, and the placement of items on the grid.

Advertising

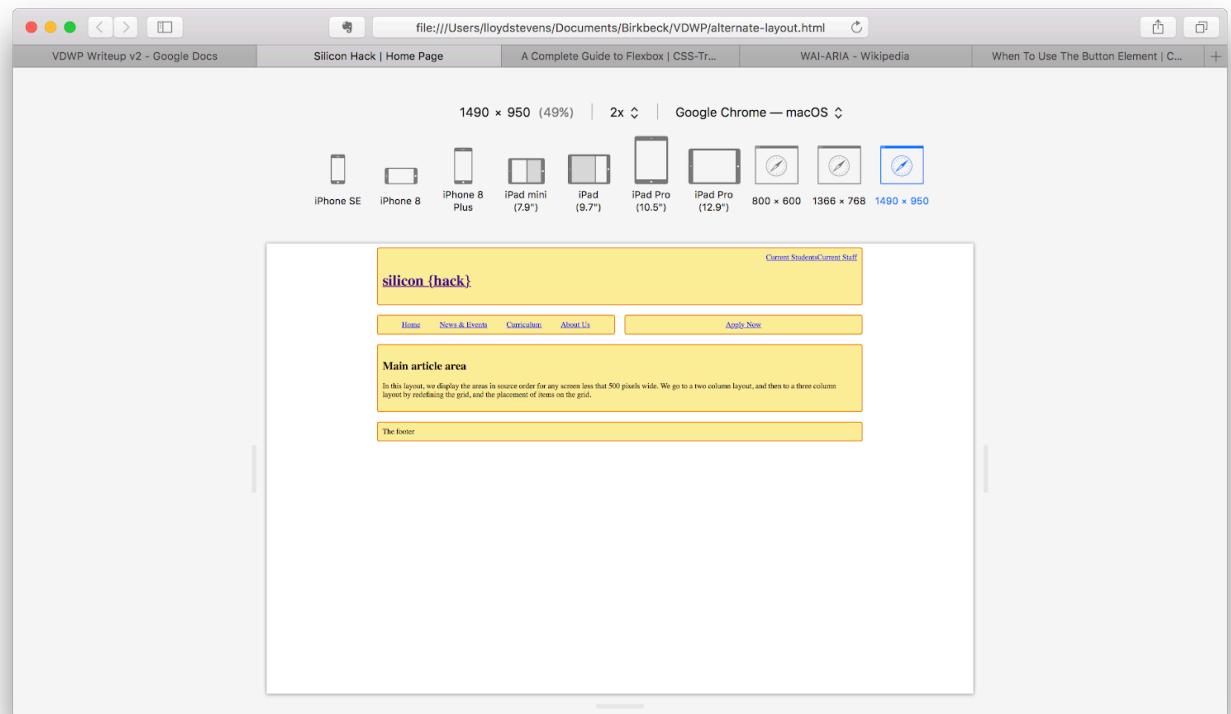
The footer

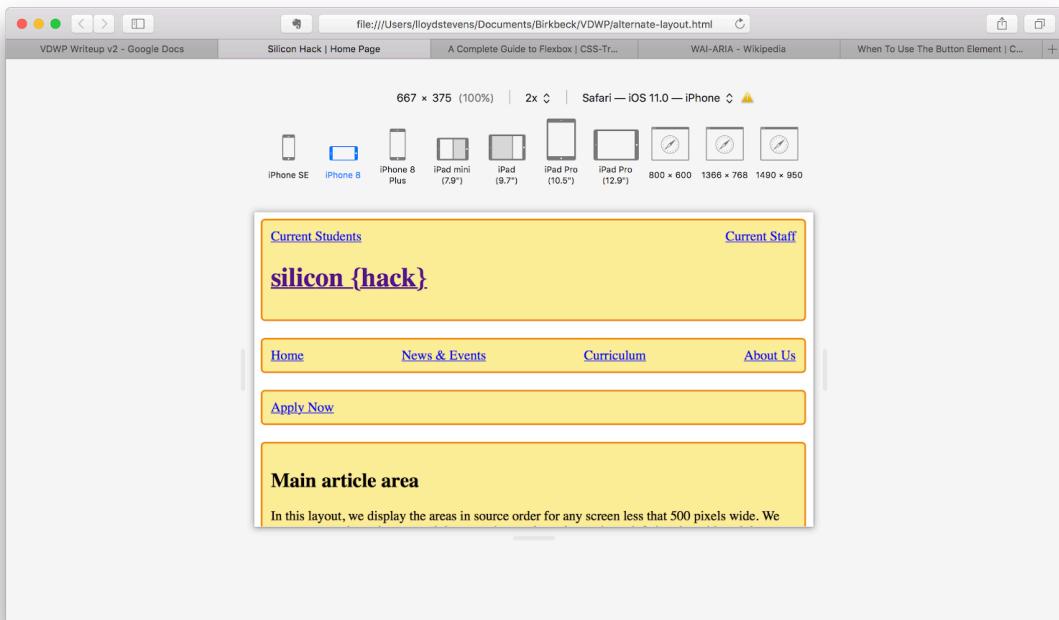
- I removed the side bar and kept a section for
 - current user navigation
 - The banner/logo
 - An area for main content
 - An area for footer material
- In VDWP v4 you can see that I created two files to work with using the above layout
 - Alternate-layout.html
 - alternate.css
- I altered the design to make it appear as a single column, although a 12 column grid was still used as I wanted to be able to position items according to this grid later
- Each column was designated with 1fr, this means that it would adapt according to the screen width and assign an equal amount of space
 - <https://alligator.io/css/css-grid-layout-fr-unit/>
- I implemented media queries to this design to start testing responsiveness
- The media query will trigger when the screen width hits a certain breakpoint
- At this point I was still designing the desktop layout first and working backwards on the mobile sized layout repositioning items on the grid where appropriate.
- The figures below show testing of the responsive design with different screen sizes using safari.



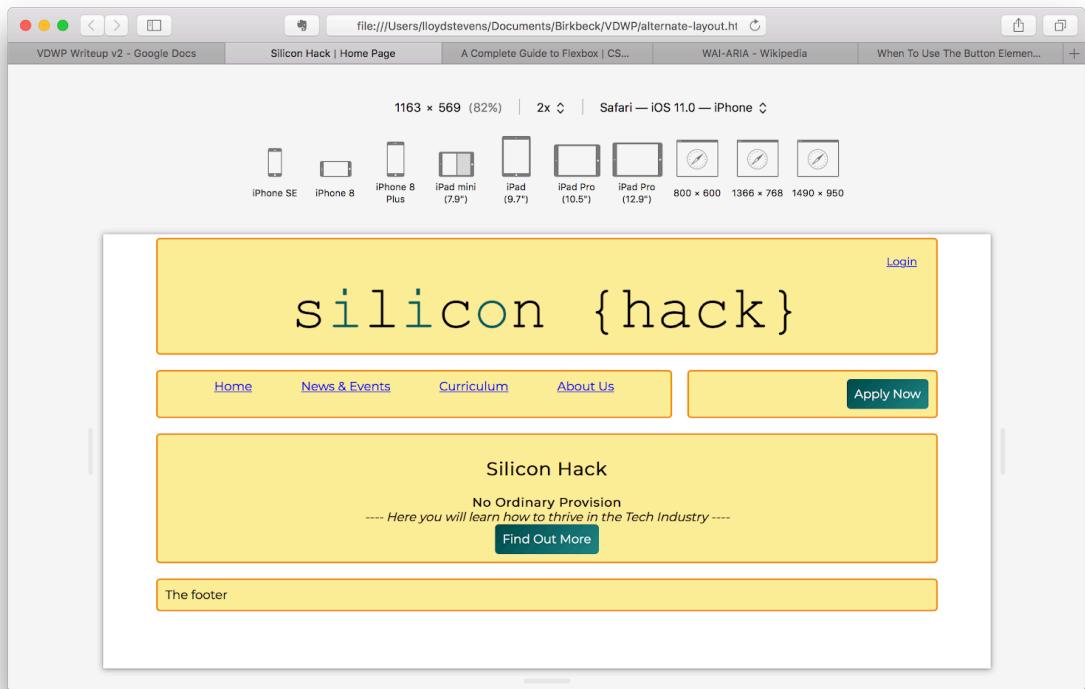


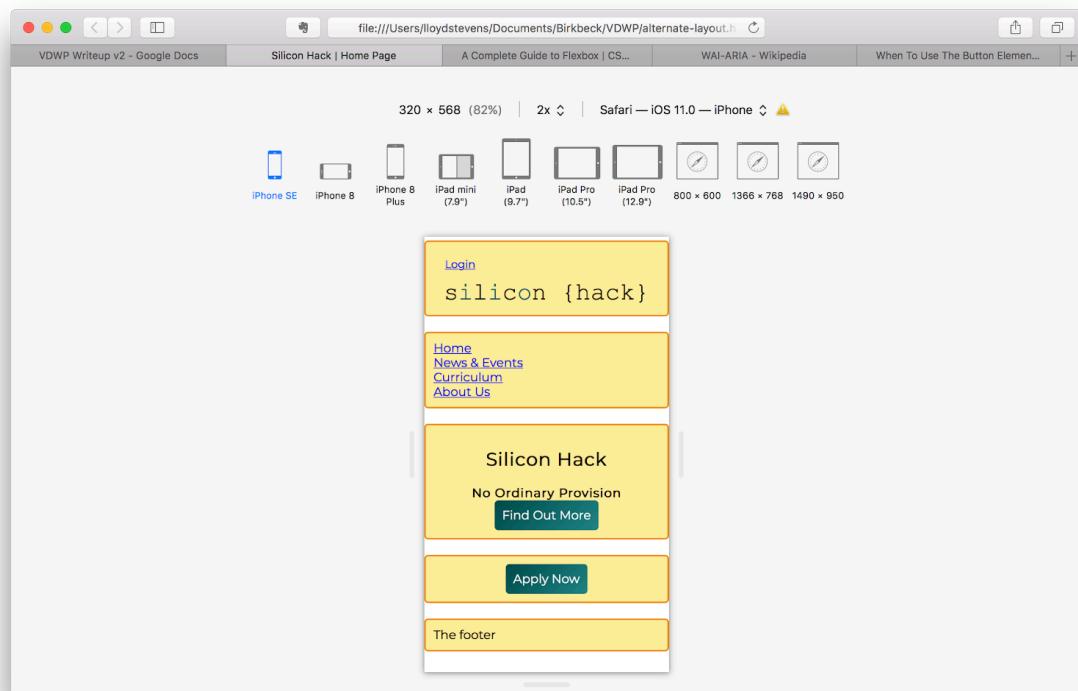
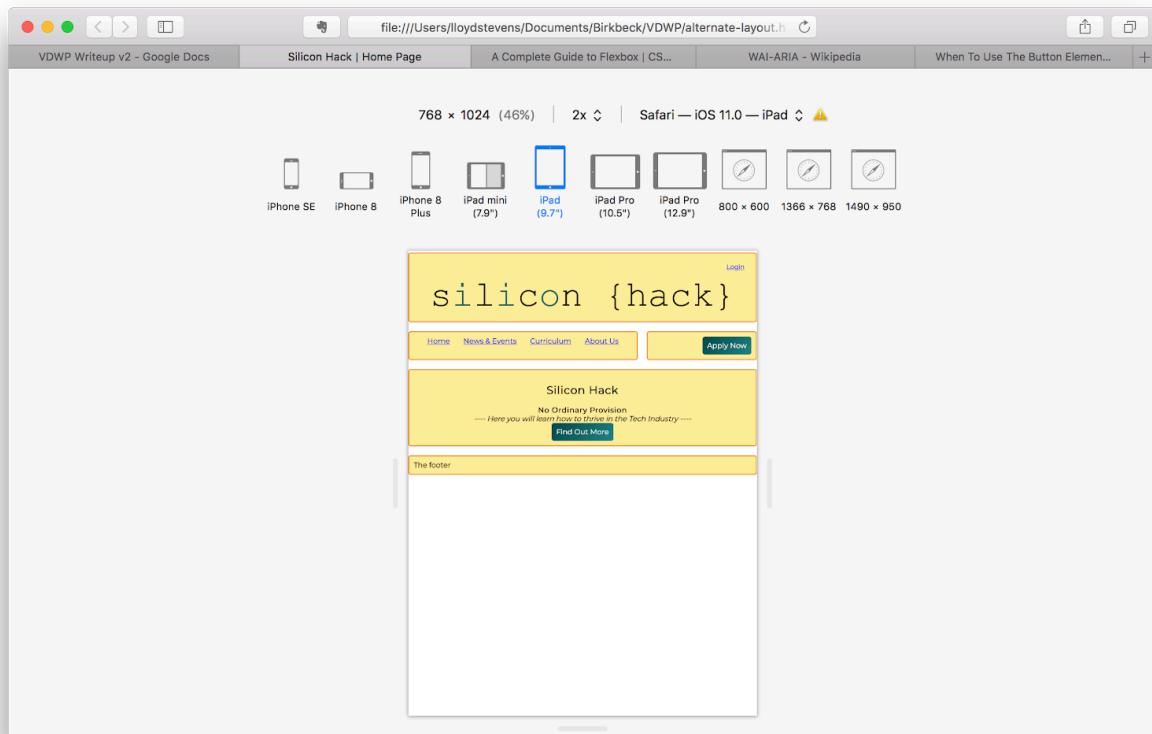
- From here I added in the apply now call to action button and started to position the
- I wanted the button to appear on the right hand side on the main desktop menu
 - Here I realised that I was using this as a button so researched whether I should be using the button HTML element instead
 - However this is not the correct usage due to the semantic meaning of the button element being for forms
 - Using the button element would give it less meaning for accessibility purposes
 - <https://css-tricks.com/use-button-element/>
 - From the article above I found that adding the class button, in my case “call-to-action-button” helps to give it more meaning
 - I also added the role=”button” for ARIA purposes
- I used media queries here to trigger the movement of the apply now button
- So for the mobile the apply now button moves its position in the grid by utilising the grid positioning of grid-row and grid-column.



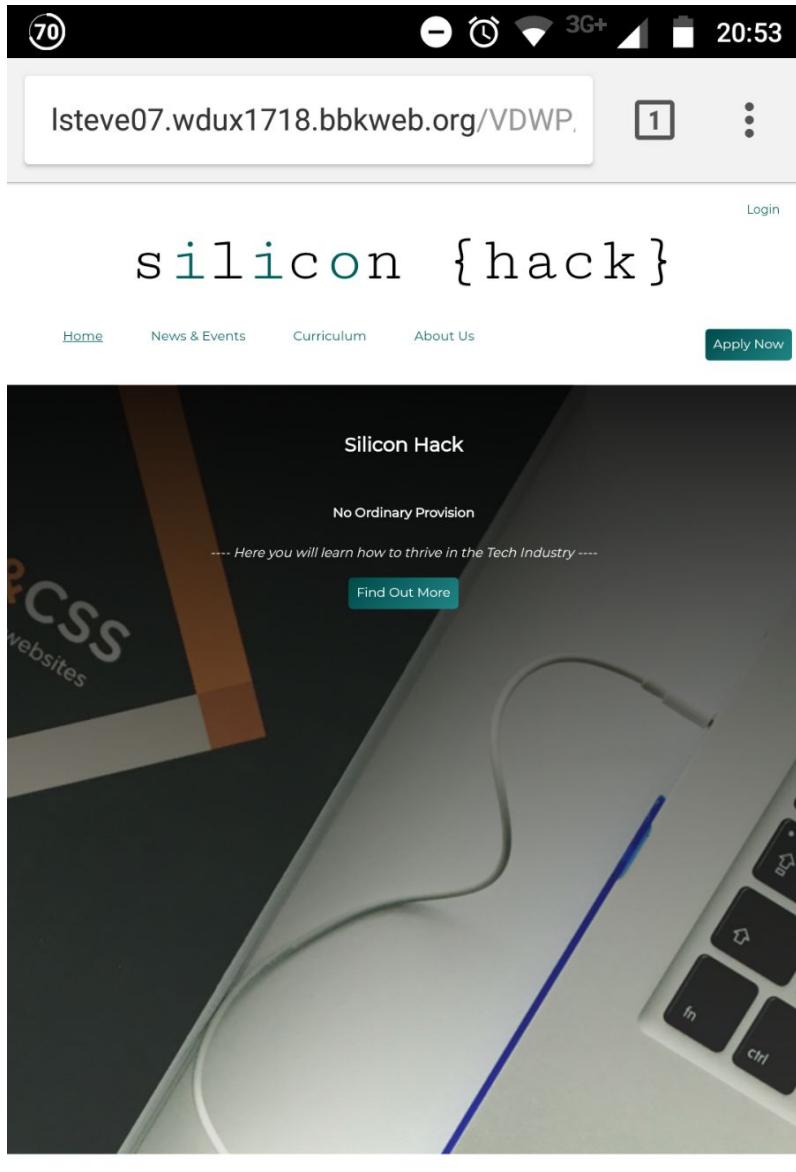


- Once this was done I started to incorporate the design I had setup for VDWP v3 previously, with the styling of the button
- I decided upon a three breakpoints here, one for mobile, one for a tablet/laptop and one for desktop
- When hitting the mobile I removed some text also adjusted the layout depending on the width of device
- Within the next images you can see the movement of apply now to a different position on the mobile layout as I was experimenting with the best appearance of the design.





- From here I removed the formatting that I was using for testing purposes and started to style the pages
- I uploaded to the live server but had an issue with live testing on mobile devices
- The figure below shows the site displaying the desktop version on Android Motorola phone
- I had the same issue on iPads and iPhones



- I found out through research that I needed this meta tag in the head
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
 - <https://stackoverflow.com/questions/13002731/responsive-design-media-query-not-working-on-iphone>

- Once added this fixed the issue on devices.
- From here I was able to move towards a mobile first development
- Doing some research on breakpoints I found this article
<https://responsivedesign.is/strategy/page-layout/defining-breakpoints/>
- I setup the breakpoints based upon when the pages started to lose shape rather than on a particular device size.
- I believe this to be more efficient a design process as devices change regularly and therefore designing to one particular device e.g. iPhoneX would give the site only a years lifespan at that specific size.
- So rather than design to a specific size I have chosen breakpoints where necessary to improve the appearance of the site.

Hiding and Showing Content

- Within the CSS there have been four classes created
 - Content-desktop
 - Content-mobile
 - Content-tablet
 - Content-desktop-tablet
- These are used to set the content that should be displayed on each breakpoint
- Not all of them have been used for this version but have been included for maintainability purposes

For example for mobile the CSS states

- Content-desktop
 - none
- Content-mobile
 - Display: inline-block
- Content-tablet
 - none
- Content-desktop-tablet
 - None
- If an element had the class content-mobile it would appear on the screen, such as the burger menu
- If an element had the class content-tablet then it would not appear on the mobile version of the site
- This has been used in some of the areas of the site to hide and show content where required
- This is something that can be streamlined within the CSS as there are inconsistencies with its usage
- For instance I have used #main-nav to hide the main navigation on the mobile breakpoint but I should consider using one of the above classes if I were to make the site more maintainable in future

Sections

- The site has been designed using sections to order the content
- Sections are a HTML5 element
- They provide more semantic meaning than using regular div elements
- It makes the pages easier to read also when maintaining to find a main section that then contains a variety of divs within it
- Within sections headings start at H1
- This is why each page has several H1 tags

<https://webdesign.tutsplus.com/articles/the-truth-about-multiple-h1-tags-in-the-html5-era--webdesign-16824>

https://www.w3schools.com/html/html5_semantic_elements.asp

Wrapper

- As previously mentioned the layout chosen was focussed on a 12 grid layout
- Most sections of the site are based on this grid layout
- The term wrapper was taken from the mozilla site previously identified
- Within the css this is set as a grid display and 12 columns of equal size 1fr.
- This allows the grid to fill the page with equal columns
- Using the .wrapper > * I was able to ensure that all elements that were direct children of wrapper were positioned across all columns
- This was done at the start so as to provide the mobile development focussed on one singular column
- I was then able to use breakpoints to alter the layout of the grid where necessary.
- An example of this can be seen when looking at the staff images on the about us page
- On the mobile initial breakpoint each staff image spans the full grid row i.e. across every column and then is positioned one row below the other
- At the 900px breakpoint I have altered the layout of the staff images so two images appear on row 2, two images on row 3 and a final image on row 4
- The last image had to span 8 columns and be positioned from column 3 instead of 2 or 7 respectively so as to ensure it was central to the images above.
- The use of the media queries and movement of grid items happens during various sections of the site.

Current User Login

- Within the HTML you will see a current user section commented out
- This was initially planned for development but was not feasible in the timescale for this project
- It is commented out to allow for ease of inclusion in the next version of the site
- It can be seen in the explanation of the initial responsive design setup but not in the final developmental version of the site
- In a live version it of course would not be within the CSS but was kept here as a method of showing the reader the intention of the current user navigation as well as for maintainability / upgrade purposes

Logo / Banner

- The logo for the site is labelled as and ID of banner within the HTML
- Initially in the previous project it was realised that this could be styled using text and not an image
- This was followed through from the previous project
- Some minor changes to the logo took place here
- Decided upon the green colour to be used for the logo and site
 - #006060
 - Used the colour swatch that was implemented for the image section of VDWP to choose the appropriate colour.
- The H1 tags were removed from the logo as on all pages other than index it was not the main title of the page and so wasn't appropriate to appear as H1 ahead of other titles
- On the majority of pages it is clickable
- It is not clickable on the home page as this would provide a link to the page it is on, which can confuse users
- On the home page it is not classed as H1 as there is already a H1 tag that is listed as Silicon Hack within the page intro and thus it was deemed unnecessary to repeat.
- This is why the logo is within a div
- The spans are used to target the individual letters to colour them within the CSS

Navigation

Desktop Navigation

- The navigation has been setup using the previously design navigation within the previous project
- It is setup using a simple unordered list and the various links
- This is within the html 5 Nav tag to give it meaning
- Below this in the html is the navigation for the apply now call to action button
- This is in a list with only one item, it might be thought that this is unnecessary usage of a list but in future there might be more than one button in this part of the site and so it was added for maintainability
- The CSS is used to provide colour and remove the text decoration of the navigation
- There is also a signifier provided through the use of hover to provide the user feedback when moving their mouse
- When the user is on a specific page the current page link is underlined so to identify to the user where they are in the site
- When attempting to align the navigation bar centrally inside its grid on the desktop view
- Initially I used a flexbox to attempt this
- When achieving the alignment I found it achieved this but then it caused issues on the horizontal alignment
- I later went to the use of grid layout but this also caused issues with the vertical alignment
- I later resorted to using padding, I wanted to stay consistent with grid layout and flexbox in an attempt to reduce the need of media queries to alter the layout of the navigation and mean it would be useable at a number of breakpoints due to automation
- However the time it was taking to research this compared with the time to create a variety of breakpoints to resolve the issue meant I sacrificed this piece of automated functionality

Responsive Navigation

Tablet & Smaller Laptop Navigation

Burger Menu

- For this I inserted the burger icon that was created and reduced the size accordingly using the CSS
 - On reflection the size of the menu is too large and this should be reduced in the image using photoshop rather than in CSS
 - On further reflection this should be fully coded using HTML and CSS, it was something that was not realised until later during the project and would be implemented for a future version of the site.
- I used the content-mobile class to show this in the mobile breakpoint only
- Used the following source code from W3Schools
 - https://www.w3schools.com/jquery/eff_toggle.asp
- I added a new line to set the initial state to hide the navigation, this I found at
 - https://www.w3schools.com/jquery/jquery_hide_show.asp
 - The value within toggle of 300 is the number of milliseconds
 - This gives the animated affect when showing the content

```

$(document).ready(function() {
    //set nav to hide
    $("#burger").hide();
    //when clicked set nav to show
    $("#burger-button").click(function() {
        $("#burger").toggle(300);
    });
});

```

- This provided suitable functionality which can be seen in Version 9 of the site.
- However when hitting the higher breakpoints the use of hide was continuing to hide the navigation
- I found out that it is possible to run an if statement based upon the viewport
- <https://responsivedesign.is/develop/javascript/conditionally-load-javascript-based-on-media-query/>
- Pseudo code
 - IF Viewport < 900
 - Burger.JS
 - Else
 - Do nothing
- Adding in the following code and testing stopped the hide function but also stopped the rest of the script when resizing the browser.
 - if (document.documentElement.clientWidth < 900)
 - Further testing showed that the script would only run on the first load and would require a reload of the page to reapply the script to display or hide the relevant content

As I have setup my site with various breakpoints I was able to ensure that the burger menu kick in at 900px.

I used the following code to ensure it would only work at the correct screen size, this was in response to the issue of `$("#burger").hide();` hiding the menu at all breakpoints.

The issue I have now is that it won't re-adjust unless reloading the screen, therefore if a browser window is below 900px and resized or a tablet is rotated then it would take a refresh of the page before navigation adapted accordingly.

```

if (document.documentElement.clientWidth < 900) {

    $(document).ready(function() {
        //set nav to hide
        $("#burger").hide();
        //when clicked set nav to show
        $("#burger-button").click(function() {
            $("#burger").toggle(300);
        });
    });
}

```

I did some more research and found `window.onresize()`

```
// only run the following code if the viewport is less than 900
window.onresize =function(){

    if (document.documentElement.clientWidth < 900)

    {

        $(document).ready(function(){

            //set nav to hide
            $("#burger").hide();
            //when clicked set nav to show
            $("#burger-button").click(function(){

                $("#burger").toggle(300);
            });
        });

    }
}
```

Testing produced two issues

- When resizing below 900 it worked and removed the menu and showed the burger icon, however when resizing back above 900 it didn't reshown the menu
- This was resolved with the below

```
// only run the following code if the viewport is less than 900
window.onresize =function(){

    if (document.documentElement.clientWidth < 900)

    {

        $(document).ready(function(){

            //set nav to hide
            $("#burger").hide();
            //when clicked set nav to show
            $("#burger-button").click(function(){

                $("#burger").toggle(300);
            });
        });

    }
else {
    // used to reshown menu if the window has been resized above
900
    $("#burger").show();
}
}
```

- I inserted an else statement to show the menu if it went back above 900
- This appeared to be functional upon further testing
- The other issue was the appearance of the menu upon the click was causing it to hide and reappear several times.

At this point I have media queries setup that move the menu from a horizontal layout to a vertical layout.

When it gets to a viewport of <900 it shows the burger icon and then hides the navigation

- The burger menu works with the JS until I resize the viewport, if the navigation is "closed" so to speak i.e. hidden.
- It will then remain hidden even when I get to the position above 900.
- Then of course the icon has gone and thus the navigation is not useable.

This can be seen in VDWP 11

- I tried some further experimentation to try and resolve the issue
- The issue occurs with the menu being hidden
- A potential solution is to have two menus instead of one
- one for the horizontal and desktop navigation which will be hidden <900
- And thus one for the burger menu, which would be vertical and only come into play <900 when the burger menu is clicked
- This I think would be relatively easy to sort out, however it would mean two menus and thus more repeated code.
- I managed to get this working for the most part, however there was an issue where the hidden menu would appear for a split second on load. This wouldn't portray a good impression to the user.
- I decided that I would stop this approach persevere with the media queries approach
- I reset the code that can be found in VDWP 14
- I then realised that there were issues with some of the labelling of the ID and Classes and so altered these all to match focussing on the main-nav ID for the menu rather than the newly created burger ID
- This version can be seen in VDWP 15
- I still ran into the same problem that I had initially
 - When the browser is resized if the menu was closed it wouldn't refresh and display
 - The media query wasn't being reactivated for this ID for some reason

Adding the following code into javascript managed to fix the issue

```
$ (document) .ready(function () {
    //when clicked set nav to show
    $("#burger-button") .click(function () {
        $("#main-nav") .toggle(300);
    });
});

// to capture a resize of the window to over 900
$( window ) .resize(function () {
    if (document.documentElement.clientWidth > 900)
    {
        $("#main-nav") .show();
    }
});
```

To summarise the above.

- The media queries have worked fine with the showing and hiding of the main-nav in general
- This has worked on load and on resize with no problems until I implemented the JQuery code
- The burger menu worked when loading the page at the higher breakpoint, resizing down and clicking
- It also worked on load of the correct breakpoint
- Where it fell down was when using the burger menu and then increasing the size of the breakpoint
- If the toggle was set to hide when resizing it continues to hide the Main-nav
- This is why the code above works as it re-forces the #main-nav to show when the window is resized to above the 900 breakpoint

The code above was then adjusted to ensure it contained JQuery alone instead of a mix of javascript

```
$ (document) .ready(function () {
    //when clicked set nav of main-nav id to show
    $("#burger-button") .click(function () {
        //toggle the main nav on or off, 300 is
        the speed to which the animation comes in
        $("#main-nav") .toggle(300);
    });
});

// to capture a resize of the window to over 900
$( window ) .resize(function () {
    //check that the window width is greater than 900
    if ( $( window ) .width() ) > 900
    {
        // if it is then show the main nav, turns it back
        on if it was in a hidden state from the above toggle function
        $("#main-nav") .show();
    }
});
```

Call to action buttons

- There is a class within the CSS called .call-to-action-button
- This has been taken from the previous project WDUX as described previously
- The colour scheme of button has been altered to match the colour swatch
 - #218686
 - #004b4b
- It has been styled so that there will be an underline text decoration when hovered over
- This is to provide a signifier to the user

Page Intros

- Each page has a section that intros the page.
- There is an intro image with text that relates to the page
- Each one also has a little tag line that is aimed at enticing the user to continue reading
- Some of the pages have a call to action button, this depends on what the page is and whether it is felt that there is need for it.
- When implementing the background images it was clear that it could cause some contrast issues
- Therefore research using the following stackoverflow link led to the use of a gradient to darken the image so it would allow suitable contrast against the foreground text.
- The gradient allowed a darker area at the top moving to lighter at the bottom
- This also allowed the bottom part of the image to be seen offering some style to the intro image for the user to see the image more clearly.
 - <https://stackoverflow.com/questions/26621513/darken-css-background-image>
 - Used above link to make the image darker so the contrast of text would be suitable for accessibility purposes
- The image itself was set so that it would be 60vh, this was measured so that the image would cover a large proportion of the screen when including the site navigation and logo
- Each intro contains the heading for the page itself
- The intro itself is based within a grid wrapper so will position according to that, within the CSS it is always set to span the whole page
- Within the intro there is a div that contains a flex box has been used to position content
- This has been setup using a class called flex-columns
- All content with this class are set to appear in a column one row after each other
- They have also been set to appear centrally within the site
- Padding and margin has been applied to each of the elements within the intro to ensure they are appropriately spaced out
- Within the mobile view the em element that provides a short tagline on each page is removed once it hits the smaller breakpoints as it felt it was a more appropriate style

Accessibility

- When testing accessibility there are errors coming up for contrast for the intros
- This is as the text is white and the backgrounds are also white
- There is a background image set in CSS which is accounted for by the WAVE checker
- Each background image has a default fall back to Black to provide contrast
- There is also the gradient applied as described above to ensure the contrast is present when the image appears or when the gradient black to white appears if the image fails

Footer

Footer Links

- The footer contains links to two external sites
 - Central Foundation Boys School which is the parent school of the offering Silicon Hack
 - Policies on the Central foundation boys school site, this is because Silicon Hack follows the same policies
 - The other link goes to the site map, which is explained further down

Social Media Links

- The social media links found in the footer are made up of an image sprite
- A gif was created as part of visual side of this assignment
- Several divs were created in the footer, one to represent each link to the social media icon
- CSS is used to set the background property as the image sprite GIF
- CSS is then used to position the image sprite and size the are the div will show
- This will then only show the image of the relevant icon for the social media link
- Each div is also set as an anchor element to point to the relevant site
- For the purposes of this assignment I have linked directly to the site and not the specific page, this is because Silicon Hack does not have social media accounts set up at present.
- When testing accessibility through WAVE the following error came up

If a link contains no text, the function or purpose of the link will not be presented to the user. This can introduce confusion for keyboard and screen reader users.

- This was because I didn't want to show text with the images
- However this would cause issues for people not using a regular visual browser
- I did not want to add visible text to the page as this would detract from the visual design
- Therefore I setup the following class

```
.text-hidden{  
    font-size: 1px;  
    text-decoration: none;  
    color:white;  
}
```

- This allows the text to be seen by non-visual browsers but then hidden visually
- This corrected the error coming up
- A new error came up regarding contrast issue with the newly hidden text.
- This is understandable as the text has been set to white to deliberately cause the contrast issue.
- This can therefore be ignored

Site Map

- When setting up the site map I utilised a tool to scan all the pages for the site
 - <https://www.xml-sitemaps.com/>
- Used as a basis for the eventual site map
- I removed large amount of the code provided and streamlined the code just to the detail that is seen within this site
- The site map uses grid layout in the form of the wrapper previously described

SSI

- I setup the SSI for the site
- I took out the header and footer of the site into two separate files in the SSI folder
- There are issues that I can see from this already, where I want to have individualised changes I currently cannot do so due to the header and footer being outlined as separate documents
- One way around this is to have separate include files for different sections of the site
- The original can be seen in version 10
- I decided to break up the SSI's to make it more efficient as currently it is not suitable to do minor changes, e.g. the site title
- The update is in Version 12
 - Document info SSI
 - Document details
 - Head details
 - Header
 - Current User Navigation
 - Site Banner (Logo)
 - Footer
 - Links
 - Central foundation
 - Policies
 - sitemap
 - social
- As I was designing the site I realised that some of the sections I may wish to repeat on other pages
- In particular the buttons for the contact I decided to remove into its own shtml file to use as an SSI include.
- I want to use this to future proof the ability to quickly shift these links to other pages in the future

<https://www.htmlgoodies.com/beyond/webmaster/article.php/3473341/SSI-The-Include-Command.htm>

JavaScript

Efficiency of code

- The JavaScript and JQuery has not been tested using a automated service in the same way as the HTML and CSS has
- This code is functional for the purposes but in places could be made more efficient, this in part explained where realised in other parts of this document
- There is repeating code in some of the scripts and speed of development was taken over performance speed in these cases
- In a future iteration of the site there would be need to use a DRY (don't repeat yourself) approach to the code for both JavaScript and JQuery
- Also it would be better for maintainability purposes to utilise only one form of JavaScript
- JQuery as a framework would allow faster development but may not improve performance in some instances
- However it would be a better form of the language as it will provide other tools within its library to support further development on of the site.

<https://code.tutsplus.com/tutorials/3-key-software-principles-you-must-understand--net-25161>

<https://stackoverflow.com/questions/2340412/jquery-vs-javascript>

Separation of code

- There are several files created using both JavaScript and JQuery
- These have been separated from content of the HTML and linked as separate files
- This allows the code to maintainable but does potentially have a performance impact
- Each time a new HTTP request will be required unless the code is cached on local machine
- They could all be placed into one file to improve load times as this would allow the file to cache on the first access to the site
- However this has not been done as the site is yet to go live
- There is also a potential argument that even though it would improve load times for future visits it may make the site slow on the first visit
- This first impression is important in the case of a school wishing to sign up students
- Therefore this is another reason the files have not been amalgamated
- Whilst it may provide some improved performance there are other areas of the site that deserved attention thus this will be investigated in a future iteration of the site.

<https://stackoverflow.com/questions/15236767/one-big-javascript-file-or-multiple-smaller-files>

<https://www.quora.com/Is-it-better-to-merge-all-JavaScript-files-into-one-or-to-keep-them-separate-in-regards-to-load-times-convenience-is-not-an-issue>

<https://softwareengineering.stackexchange.com/questions/223918/is-it-better-to-put-the-js-code-on-the-html-file-or-in-an-external-file>

Chat Feature

- I installed the chat widget from tawk.io
 - This is a free service
 - The monetisation for this company comes from offering agents to monitor and run the chat for you instead
 - This tool allows you to style in the same format and I used the standard colour scheme that I have been using from the colour swatch #006060
 - There
 - <https://www.tawk.to/>
-
- There was not a great deal required to the installation
 - The Widget code was inserted prior to the closing body tag as outlined in the tawk documentation
 - For this I made a new JS file called tawk.js to keep the separation of JS from the HTML
 - This then provided a functional chat service to the site.
 - This isn't without its own issues.
 - The chat service privacy policy outlines that they wouldn't share data with other organisations but they would be using data collected for their own analysis
 - This raises questions about potential child protection issues
 - The feature has been left in for the purpose of submission but would likely be replaced by a different chat tool in the future or removed altogether as a feature.

```
//Start of Tawk.to Script

var Tawk_API=Tawk_API||{}, Tawk_LoadStart=new Date();

(function() {

    var
s1=document.createElement("script"), s0=document.getElementsByTagName
("script") [0];

    s1.async=true;

    s1.src='https://embed.tawk.to/5ac4fdb1d7591465c709312c/default';
    s1.charset='UTF-8';

    s1.setAttribute('crossorigin','*');

    s0.parentNode.insertBefore(s1,s0);

})();
```

- It is difficult to explain the code for this without looking at the other pieces of code involved
- It starts by declaring a variable called TAWK_Api and assigns this to a previously created TAWk_API or an empty object
- Another variable called TAWK_LoadStart is assigned the latest date and time

- After this a function is loaded that sets s1 to be equal to a newly created script element
- s0 is set to the first value of the element with the tagname of script
- s1.async=true
 - this means that it will begin executing and any other statement will be executed even if the s1 hasn't been completed
- s1.src will pull the relevant code from the tawk website
- s1.charset sets the character set as 8 bit Unicode
- When researching the next line I found out that crossorigin is part of a feature called CORS
- This Cross Origin Resource Sharing allows two websites to share communication from the browser
- This is the feature that will allow the chat to be ran at another domain on the domain for silicon hack
- This means both sites can access the data even though they are on different domains
- The last line refers to a the parentNode and insertBefore.
- This implies that the script is working with the Dom and it will insert the value before the parent node
- From this code alone it is unclear to fully describe how this script is working
- I believe that the script is first checking to see if there has been a chat instigated from this machine recently, if there has it will allow the continuation of the chat, if there hasn't it will setup a new chat
- I believe it does this by using the DOM and the Nodes within the DOM to check for previous communication. It refers to the file that will be kept on the server of TAWK to identify what has previously been written by the current person accessing the chat and display it on the screen.
- As so much of the code is hidden from the user when setting up the API, it is therefore slightly risky to include this in the site for Silicon hack
- More investigation of TAWK as a company is required if TAWK is to be used safely
- It has been left in this assignment for the purposes of demonstrating the feature in action.

<https://stackoverflow.com/questions/251402/create-an-empty-object-in-javascript-with-or-new-object>

https://www.w3schools.com/js/js_dates.asp

https://www.w3schools.com/jsref/met_document_getelementsbytagname.asp

<https://stackoverflow.com/questions/1478295/what-does-async-false-do-in-jquery-ajax>

<https://www.html5rocks.com/en/tutorials/cors/>

<https://developer.mozilla.org/en-US/docs/Web/API/Node/insertBefore>

<https://developer.mozilla.org/en-US/docs/Web/API/Node>

https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction

<https://stackoverflow.com/questions/9979172/difference-between-node-object-and-element-object>

SVG Images

- Initially the plan was to use SVG images for the icons on the home page that provided link to the contact section of the about us page
- Used the following site to add the svg as links
 - <https://codepen.io/commondream/pen/dFgri>
 - This involved adding a class to the SVG images
 - The class then created a clickable link over the area that the SVG was contained
 - This still caused me issues though and the links were not working to the image
- Research led me to find that implementing SVG images can be tricky

Pages

[Index \(Home\)](#)

Sections

#comp-education

- This has been setup as a grid using the wrapper class
- This section is in place to give an introduction to the site and the offering of silicon hack
- Along with the text content there is an image that resizes at different break points
 - Utilising the grid layout the logo is positioned in different places and across a different number of columns
 - A % is then used to size it appropriately also depending on the breakpoint
- There is also a youtube video that will introduce the site
 - This is a placeholder video at present that will need to be replaced once a suitable video has been created by the school
 - This video also repositions according to the breakpoint using the gridlayout
- Below this there is a call to action button that will take the user to the about us page, this is so that they can read more about the staff who will support them and find out about the area
- This will then hopefully prompt them to make contact with the school

#main-news

- The main news section will contain a video that gives an overview of the latest news from the school
- At present this is using the same video as the above section just as a placeholder
- It will be replaced when a suitable video is available
- This has been set to cover the width of the site at all breakpoints in the current version of the site
- The section has been setup as a grid using wrapper that's content spans the full width of the site so that it can be adapted easily in the future if more content was to be added

#contact

- The contact link section provides a set of icons that either take the user to the form on the about us page or directly to the chat feature
- Each icon represents a button and this has been highlighted with an Aria role for the purposes of accessibility
- Also each icon has been given alternative text related to its role rather than describing its appearance, this will allow users who utilise the accessibility feature to understand the purpose of each icon
- An error in accessibility testing comes up to state that two links near each other go to the same page.
- Currently this is true, the links in their present form do not open the form at the correct contact type, this will be done in a future iteration and will involve setting a cookie to ensure the data can pass from one page to the next to identify the user request
 - i.e. each link goes to the form in the “general” view

- This accessibility issue is one that can be ignored during this iteration and would be resolved once the form correctly opens at the correct view
- The contact section is setup as a grid with 5 equal sized columns using a different class to the wrapper one used previously
- A new class was setup for this section called five-col-section-header
- This was used to spread the heading for the section across the five columns that were applied to it
- In the desktop version the icons are spread across the whole page
- Text is aligned centrally so that it fits under the correct item
- There is padding applied to each image within the CSS, this is to ensure that they appear to be consistent in layout
- Initially I used flexbox to centre the items
- Due to some of the images being a non-symmetrical shape and a transparent background it meant that they didn't appear to be central when positioned
- The padding rectifies this
- Within the mobile view the buttons have been put into position on the grid so that they appear in a 2, 2, 1 formation
- With the layout of a 5 column grid it means the last image can be placed centrally on its own
- Each one has been assigned a row and column to ensure this could happen

Accessibility

Contrast Issues

- There are contrast issues that come up in relation to the intro sections and also the footer, see these parts of the report for further detail.

News & Events

- I setup the grid as previously done in other pages using wrapper
- I added background images in a similar way to the intro sections on each page choosing to darken out the image to ensure the front text was viewable
- For each image/grid area I set the height to 50vh which means it would adapt dependent on the size of the screen and thus be responsive to smaller devices.

Sections

#news

- This section has some detail about the page and then each news item is setup as a wrapper so as a nested grid within the sections wrapper
- Within this the heading is set to have a background image in a similar manner to the intro
- This was implemented as the intial usage of the styled green heading found in other areas of the site didn't look good with so much content on the page
- The height of the image was restricted to 40vh
- This meant that the appearance would give it a slightly squarer shape on mobile but it was a view that I liked after trial and error
- The two images that went with each news item were styled side by side using grid layout on the desktop view but then one above the other on the mobile view
- As they moved down their size was also adjusted using the css to position them in the grid and also a % value
- I used two classes news-image-1 and news-image-2 to cover all of the news images easily with minimal code

#events

- Initially I designed a table for this section
- However, there were issues with making this responsive for phones and still viewable in a suitable format
- This can be seen in the image below



Come to an Event

Each term we hold a number of events that you could attend to get an idea of what we offer.

Check out the events below and then sign up through the form on the About Us page

Event	Date	Time
Taster	March 13th	7pm
Taster	April 25th	7pm
Open Evening	July 14th	7pm
Open Evening	July 21st	7pm
Hackathon	Jun 23rd	9am to 12pm
Hackathon	Jun 24th	9am to 12pm

[Sign Up For an Event](#)

© Central Foundation Boys School
Policies

- I designed the table so that the rows would change colour when hovered over, the code of which is seen below
- The table was styled using CSS
- The majority of styling is original
- There is some inspiration and lines taken from the following source

<https://freshdesignweb.com/free-css-tables/>

```
#events-table tr:hover:not(th) {  
    background-color:#a4d9f5;  
  
}  
#events-table th {  
    text-decoration: underline;  
    background-color: #006060;  
    font-weight: 300;  
    color: white;  
    padding: 5px 50px;  
}
```

- The way to make a table responsive appeared to quite cumbersome and I had the idea that it would be much quicker and easier to complete in a grid instead.
- Each grid item would contain the table heading or table row
- On reflection I realised that this would appear suitable it would not be semantically correct and thus could confuse understanding in browsers that were not visual

- I also had two design issues with the table in the above format
- The first being that it would appear that a row could be clicked to start the sign-up process, a feature that although I would like is not one that was programmable in the time left to complete the project.
- Secondly, I felt that the events section was being overpowered by the news and would potentially be missed by the user.
- To keep consistency and to overcome the above issue I decided to design the events section in the same manner as the news section above
- However this time rather than two images I had two columns of text
- There was also a call to action button to take the user to the about us page
- This would take them directly to the form
- Unfortunately, at this point, there wasn't time to code in a solution to open the form at the correct event, this will be a future development.

Accessibility

- The sign up buttons were flagging up issues as they were adjacent links going to the same place
- However this is necessary as it is a call to action after each point to get the user to perform some kind of sign up to an event

Contrast Issues

- Contrast issues on this page have been explained under in relation to the intro sections and also the footer and so can be ignored.
- There is a similar issue with the contrast of the headings that utilise backgrounds in CSS and again these can be ignored as the WAVE checker does not factor in the CSS background
- There is a fall back on each of these and also the gradient has been applied to provide suitable contrast of text over the image

Curriculum

Sections

- The main section on this page has been designated a wrapper
- Within it are divs what learn and digital media
- Both of these are set to span a number of columns
- On the desktop version they leave space between the edge by using the css to control the number of columns they span
- When they go down the breakpoints they will span the whole page

#what-learn

- This section gives some detail about what will be learnt at the school.
- It has a relatively basic layout using paragraphs and list elements

#digital-media

- This contains links to external pdf documents
- This details the units studied as part of the digital media part of the course
- This was setup as a nested 12 column grid using wrapper

#IT

- This contains the links to external pdf documents
- This details the units studied as part of the IT part of the course
- This was setup as a nested 12 column grid using wrapper

Common Use CSS

- Here I started to develop my own personalised CSS for layout, it was nearer the end of the development process that I created this page
- I setup col-1 and col-2 classes that were used to layout the items in two columns across this 12 column grid
- Although I haven't used bootstrap within the project I have used it previously and it is something that I started to see myself doing, further evaluation of this can be found in the evaluation at the bottom of this documentation.

Accessibility

Home Adjacent Link

- This is an error that comes about because the home button is next to the Silicon Hack Logo that also links to the home page
- This is an issue that can be ignored as they both serve a purpose

PDF

- Accessibility threw out a number of errors relating to PDF documents being present and providing warning to the user before opening
- Initially a small message was left near the links at the top to provide warning
- However this could easily be missed
- Researching led to the W3.org guidance on new windows
- Although different to the above issue it was felt that this way was simple but yet effective

- (PDF) was added to each buttons text
- Although it doesn't necessarily provide a backout it would still work when JS is turned off
- <https://www.w3.org/TR/WCAG20-TECHS/G201.html>
- This still threw up the error but it was partially resolved with this message that would be "viewable" both visual and through screen readers.
- There was a knock on effect as this now meant that at certain widths the buttons were not consistent in size
- This was a side effect that would have to be corrected in a future iteration due to time
- For the next iteration there would be a different page for each unit and HTML detail outlining the unit content rather than the link to the external PDF
- This would allow the summarising of the unit in pupil friendly speak and also resolve the accessibility issues with the use of PDFs.

Contrast Issues

- Contrast issues on this page have been explained under in relation to the intro sections and also the footer and so can be ignored.

Employer Engagement

Sections

- A relatively simple page in comparison to some of the others in layout
- This utilises grid but displays all content in one column

#employers-section

- Here there is detail outlining what the offering is and how employers can help

#join-us

- This section is in place to call the employer to join us and support
- It directs the user to the about us page contact form
- As described with many of the other links to this form it does not open the form in the correct view
- The heading of this section has been styled in the same way as the intros and headings within the news and events which has been described previously

Accessibility

Contrast Issues

- Contrast issues on this page have been explained under in relation to the intro sections and also the footer and so can be ignored.
- There is a similar issue with the contrast of the headings that utilise backgrounds in CSS and again these can be ignored as the WAVE checker does not factor in the CSS background
- There is a fall back on each of these and also the gradient has been applied to provide suitable contrast of text over the image

About Us

Sections

#more-about-us

- This section gives some more detail about the offering and the school
- It is there to give a bit of a flavour of what can be studied
- It is a relatively simple section that provides basic layout from top to bottom in all the views
- There is a call to action button to take the user to the curriculum to find out more
- On reflection this might not be the best way to move users around the site as it takes them away from the contact form.
- This is something that would be need to be analysed over time and adapted if necessary i.e. not enough contacts were being made

#our-staff

- The our staff section is setup as a 12 column grid using wrapper as described under wrapper (see this further up to understand how this works)
- The images have been styled using the CSS so they take a circular format
 - NB the image for Mr Lau is a placeholder using an image of Miss Lassami
- When the user clicks on the text or image of a person it slides out more detail
- I used the Jquery slide toggle found on
 - https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_slide_toggle
- I adapted this within a seperate js file and called it in the bottom of the about_us page.
- To adapt I added a line to hide the content on load as it was initially showing
- I then created a separate set of code for each person, this is not the most efficient way as there is repeating code, it should be possible to do this using functions and a call possibly using something like get.Elementby however for the purposes of getting functionality and developing the full site I have left this as a future development.
- To futher improve this page after the above was completed I used :hover to alter the cursor property to be a pointer when the mouse is over the div that contains the image
- This offers the signifier to the user that allows the should prompt the user to click the image or heading to reveal more detail about the staff member

#map

- The map section contains detail about where the provision is based and an embedded google map
- This is a grid layout using wrapper and thus a nested grid
- However at present it just uses a simple top down view in all the different screen sizes
- As it is using a wrapper it does have maintainability for future

#general-contact

- The general contact section is a form that allows the user to send a general message
- Or change the form using a select type in at the start of the form
- The form has been styled using field sets to

PHP Form

- A form was setup in HTML and PHP was applied for the submission using the TECTITE

#apply-now-form

- This form is for prospective students to apply
- When selected it display parts fo the form that allow them to input their grades for certain subjects
- There are also areas within here that will hide or display dependent on the slection fo radio buttons this is outlined in the explanations further down

#event-signup-form

- This form is for prospective students to apply for an event
- When selected it will display the area to choose which event they wish to attend
- When choosing the event it will offer feedback to the user in the page
- This part of the form isn't ideal as it isn't connected to any kind of database
- A better way would be for users to come straight from the events page and have the detail required filled out but this was beyond the scope of this project
- The JS related to this is described further down below

#employers-form

- This form is for employers to contact to offer their support in some way
- When selected it will give the fields for the employer to fill out

Form elements

Form Functionality

Dynamic Forms

- There are a number of forms for users to fill out information
- Parts of these forms will have common headings
- Rather than create several forms I decided it would be possible to create one form and then adjust the contents of the form dependent on how the form was accessed or a drop down within the form.
- Research took me to this stack overflow link
 - <https://stackoverflow.com/questions/37390865/change-form-fields-based-on-dropdown-values>

▲ Late to the party here, but here is another way

0

```
// hide all the divs
$('div').hide()

// Show and hide selected div
$('#cases').change(function () {
    var value = this.value;

    $('div').hide()
    $('#' + this.value).show();
});
```

Also created a [demo](#)

The code starts by hiding all the divs, this would need altering for my implementation

- It then uses the id of cases to capture the value of #cases when changed. In this case cases is a drop-down menu and so it will then take the value.
- The code then hides all the divs and the final line uses concatenation to append the value of the dropdown to the # so as to show the div with the matching drop-down.
- For this to work it would rely on the ids matching the values within the drop-down of the form.
- I created the form for each of the various fields required using the following resource for support with styling and HTML5 form features
- <https://www.html5rocks.com/en/tutorials/forms/html5forms/>
- I then used CSS to hide each of the Divs relating to the fields that didn't come in the standard contact us form.
- I then used the previously found code to create the following code

```
//the form changes as the function is changed
$("#form-menu").change(function () {
    var choice = $( "#form-menu option:selected" ).val();
    if(choice=="apply-now")
    {
        //show apply now fields
        $("#event-signup").hide();
        $("#apply-now").show();
        $("#Contact-Us").hide();
        $("#employers").hide();
    }
    else if(choice=="event-signup")
    {
```

```
//show event sign up fields
$("#event-signup").show();
$("#apply-now").hide();
$("#Contact-Us").hide();
$("#employers").hide();
}
else if(choice=="Contact-Us")
{
//          show contact us fields
//      NB this is currently empty and code is here purely
for maintenance purposes to provide functionality in future, however
the change will still hide the other fields that aren't relevant
$("#event-signup").hide();
$("#apply-now").hide();
//          $("#Contact-Us").show();
$("#employers").hide();
}
else if(choice=="employers")
{
//show employers fields
$("#event-signup").hide();
$("#apply-now").hide();
$("#Contact-Us").hide();
$("#employers").show();
}
})
});
```

Other Dynamic Form Features

- When accessing the apply now form there are fields that would not be relevant to all
- Again I didn't want to take up screen space with unnecessary fields
- Also I didn't want the user to be confused and fill in details not necessary to them
- I decided to use radio buttons to change the appearance of the form
- A search found this source on stack overflow with the following code
 - <https://stackoverflow.com/questions/20468457/show-form-based-on-radio-button-select>
 - <http://jsfiddle.net/LbrCf/>

```
var radios = document.getElementsByName("radios");
var cardpayment = document.getElementById("cardpayment");
var internetpayment = document.getElementById("internetpayment");
cardpayment.style.display = 'block'; // show
internetpayment.style.display = 'none'; // hide
for(var i = 0; i < radios.length; i++) {
    radios[i].onclick = function() {
        var val = this.value;
        if(val == 'radio1' || val == 'radio2'){
            cardpayment.style.display = 'block';
            internetpayment.style.display = 'none';
        }
        else if(val == 'radio3'){
            cardpayment.style.display = 'none';
            internetpayment.style.display = 'block';
        }
    }
}
```

- I then used the above to adapt my own code for two areas on the form
- The apply now for where students fill out results for science which they will have studied either separate science or combined science.
- Also where students will have filled out whether they have studied ICT or Computing or in fact neither.

Separate and Combined science results

```
// the form changes if certain radio buttons are selected
var radios = document.getElementsByName("radios");// 
var combinedscienceresults
= document.getElementById("combinedscienceresults");
var seperatescienceresults
= document.getElementById("seperatescienceresults");

combinedscienceresults.style.display = 'block'; // show
seperatescienceresults.style.display = 'none'; // hide

for(var i = 0; i < radios.length; i++) {//
    radios[i].onclick = function() {//
        var val = this.value;// 
        if(val == 'combinedscience') {//
            combinedscienceresults.style.display = 'block';
            seperatescienceresults.style.display = 'none';
        }
    }
}
```

```

        }
        else if(val == 'seperatescience'){//
            combinedscienceresults.style.display = 'none';
            seperatescienceresults.style.display = 'block';
        }

    }
}

```

ICT and Computing Results

```

// the form changes if certain radio buttons are selected
var ictradios = document.getElementsByName("ictradios");//  

var ICTgrades =  document.getElementById("ICTgrades");
var Computinggrades =  document.getElementById("Computinggrades");

ICTgrades.style.display = 'none'; // show
Computinggrades.style.display = 'none';// hide

for(var i = 0; i < ictradios.length; i++) {//
    ictradios[i].onclick = function() {//
        var val = this.value;//
        if(val == 'ICT') {//
            ICTgrades.style.display = 'block';
            Computinggrades.style.display = 'none';
        }
        else if(val == 'Computing') {//
            ICTgrades.style.display = 'none';
            Computinggrades.style.display = 'block';
        }

        else if(val == 'noict') {//
            ICTgrades.style.display = 'none';
            Computinggrades.style.display = 'none';
        }
    }
}
}

```

Both work in the same way, so I will describe the ICT/Computing code only.

- The first variable relates to the value pulled from the radio buttons which is stored in ictradios.
- This is achieved using `getElementsByName`
- Where it looks for the name ICT radios and assigns its value to the variable
- This is an array, an array is a variable that can contain multiple values of the same data type
- The key part to the JS here is the plural elements
- This means as it finds a new value it will append it to the array rather than just overwrite
- in this case of 3 items as there are three buttons
- If no buttons are selected the array would contain "", "", "" i.e. 3 empty array items
- If one of them were selected the value of an item would change to contain the value in the radio button
 - ICT

- Computing
 - noict
- There are two other variables set
 - ICTgrades
 - Computinggrades
- Both of these use getElementByID these are used later to access the elements with the ID's in this case both elements are Divs
- The next line uses these items straight away to hide the two divs that relate
- The HTML also has the noict radio button checked by default which is the state that would see the two hidden buttons
- This means I do not have any code to perform an action on load but only on the event if the radio button is changed
- The for loop loops through each array and checks whether the radio has been clicked
- When clicked this then stores the value in a variable called val
- Selection statement (if else if statement) is then called to check whether the value equals ICT, Computing or noict using the equality operator ==
- If that condition is true it will then jump into run the two lines of code within its block
- Depending on the value it will either hide the ICT grade entry through its div value, the computing grade entry or both

https://www.w3schools.com/jsref/met_document_getelementbyid.asp

https://www.w3schools.com/jsref/met_doc_getelementsbyname.asp

<https://techterms.com/definition/array>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Comparison_Operators

https://www.w3schools.com/js/js_if_else.asp

Email Verification

- When reading the above articles to solve the previous problems I found the following code (see below)

Verify Email input code

```
var elem = document.getElementById("email_addr_confirm");
elem.addEventListener("blur", verifyEmail);

function verifyEmail(input) {
  input = input.srcElement;
  if (input.value != document.getElementById('email_addr').value)
  {
    // the provided value doesn't match the primary email address
    input.setCustomValidity('The two email addresses must
match.');
  } else {
    // input is valid -- reset the error message
    input.setCustomValidity('');
  }
}
```

- Researching onblur I found that the term relates to the event of leaving a form field.
- Therefore the code below will verify the two emails match when it moves from the second email address data entry

- https://www.w3schools.com/jsref/event_onblur.asp

The form contains fields for Email Address, Confirm Email Address, Date, and Number of Guests. The 'Email Address' and 'Confirm Email Address' fields are highlighted in blue, indicating they are being compared. A validation message box displays the error: "The two email addresses must match.".

The code didn't take much to get working. It just requires the matching of the id's of each of the fields that are to be verified against each other.

The code works by getting the email address of the element with id email-addr-confirm when the user leaves the text box

It then invokes the verifyEmail function with the value found

The if statement checks whether the value found in email-addr-confirm matches the value in email-addr.

If they do not match an error message is assigned using set.Custom.Validity

If they match set.Custom.Validity is set to empty

Reading the article, along with other research, it explains that when using the set.Custom.Validity function an empty value will allow the form to be submitted but a value results in the error message being displayed.

- https://www.w3schools.com/js/js_validation_api.asp

One down side of this method of validation is that it only points out to the user when they click submit instead of while completing the form. However the positive is that it doesn't require submission to the database for the check and thus will not require reloading of the form. This also has the potential to be more secure as no data is being sent and resent back and forth. If the data was submitted it would require storing the data in the form temporarily until the confirmation of a completed submission. An invalid result would require sending data back from the server again and then back to the server. This number of sends would in theory increase the chances of data being intercepted.

I found a way to overcome this by using

```
input:invalid {  
  
background-color: lightcoral;  
  
}
```

- However this gives the form a style of appearance that is quite offputting in my opinion and could potentially alienate a user prior to any data entry.
- I therefore removed this type of signifier that there is invalid data.
- To achieve this there would need to be some more javascript to adjust the CSS value on exit from a field using the *blur* command identified previously

Displaying Event Details on Form

- When selecting an event I wanted to show some feedback to the user so they were sure what the event they were selecting was
- I used some javascript to get the details and display them

```
function displayEvent() {  
    var eventChoice = document.getElementById("event-choice")  
  
    var displayResult =  
    eventChoice.options[eventChoice.selectedIndex].text;  
  
    document.getElementById("event-details").innerHTML = displayResult;  
  
}
```

- The above code gets the value of the dropdown index and saves it in the variable event choice
- IT then uses a call on the index to store the text of the drop down selected in the displayResult variable
- This is then used to alter the HTML of the DIV event-details using .innerHTML.
- Within the related HTML the event onchange is used to then call the above function]
- So every time an event is selected it will then place the detail in the form so the user can see they have the right event

https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_element_innerhtml

<https://stackoverflow.com/questions/1085801/get-selected-value-in-dropdown-list-using-javascript>

Form Styling

- I researched some button styles and used this link to get some ideas for the drop down at the start of the form
- I wanted something that would stand out to the user but be in keeping with the sites colour scheme

<https://codepen.io/ericrasch/pen/zjDBx>

- When styling the select menu there were issues.
- I wanted a suitable signifier so that the user would clearly be able to choose the right form they wished to fill out
- Firefox caused issues when using padding so that the content was not viewable
- <http://jqueryui.com/selectmenu/>
- Therefore, I had to remove padding on the styling
- I reduced the size of the field so that it would be 50% of the screen size, this helped to make it stand out more within the page
- I also increased the size of the text to make it more obvious to the user.

Validate date field - although not needed if setup db in mysql

- To ensure that students who were too old or young could not sign up the min max attributes were used for the DOB entry

```
<label for="DOB">Please enter your DOB</label>
<input type="date" name="DOB" id="DOB" min="2002-09-01"
max="2001-09-01">
```

<https://stackoverflow.com/questions/17182544/disable-certain-dates-from-html5-datepicker>

PHP Form Script

- The form emails the user on submission
- I implemented the PHP form script from Tectite.com.
 - [FormMail](#) (9.15), a PHP script available from www.tectite.com.
- When testing I ran into the following error

Your form submission was processed by [FormMail](#) (9.15), a PHP script available from www.tectite.com.

GET method has been disabled. Forms must use the POST method.
Alternatively, reconfigure FormMail to allow the GET method.

- I realised I had forgotten to declare the form method and so added in the POST method

```
<form action="http://lsteve07.vdwp1718.bbkweb.org/php/formmail.php"
method="post">
```

- After this running the form resulted in receiving the following error

Please contact us (lsteve07.vdwp1718.bbkweb.org) directly since this form is not working.

lsteve07.vdwp1718.bbkweb.org apologizes for any inconvenience this error may have caused.

An error occurred while processing the form .

- This occurred as I had misplaced the hidden fields outside of the form tags

```
<!-- PHP Form - contact fields -->
<input type="hidden" name="env_report"
value="REMOTE_HOST,REMOTE_ADDR,HTTP_USER_AGENT,AUTH_TYPE,REMOTE_USER
" />
<input type="hidden" name="recipients"
value="stevens.lloyd@gmail.com" />
<input type="hidden" name="subject" value="Silicon Hack: Contact"
/>
<!-- ----- -->
```

```
<form action="http://lsteve07.vdwp1718.bbkweb.org/php/formmail.php"
method="post">
```

- Once I had put these in and tested this resulted in the standard message

Thanks! We've received your information and, if it's appropriate, we'll be in contact with you soon.

- However when checking the email not all the data was appearing
- I went into the form settings and made some changes to my code after inspecting
- The fields that were not displaying did not have a name value set so first I ensured these were set
- Also the email address had to be named "email"
- The name field had to be named "realname"
- When inspecting the result I realised that it wasn't clear which event had been chosen.
- Therefore I added a new option which would default none selected.
- An Error email was arriving in conjunction with the email containing the form information

The following error occurred in FormMail :

Unable to create check file "/tmp/fm8c33b3.txt": fopen(/tmp/fm8c33b3.txt): failed to open stream:
Permission denied

- Research found that this was a server issue and would require PHP being altered on the dreamhost server
- Using the source below I switched off the check for updates within the form using the variable
 - CHECK_FOR_NEW_VERSION
- And set the value to false.
 - \$CHECK_FOR_NEW_VERSION = false;
- This stopped the error coming through

- <https://www.tectite.com/vbforums/showthread.php?682-Unable-to-create-check-file-error-message&highlight=Unable%20create%20check%20file>

Form Confirmation – Success and Error Pages

- When the form is submitted the user is taken to either the confirmation page or error page
- The confirmation page allows the user to return to the home page or further down click to perform a new contact
- The error page gives the user opportunity to return to the form either by a call to action button or via the various links within the page in the form of the contact us buttons.
- This is achieved by using two hidden form fields
- The field named “good_url” redirects on a successful form submission to thanks.shtml
- The field named “bad_url” redirects when there is an error with submission to the page error.shtml
- <https://www.tectite.com/fmhowto/redir.php>

Form Submission

- The form uses a POST Method
- On the whole POST is amore secure method to send the data
- It doesn't not store the data sent within the browser and it is more difficult to be hacked
- <https://www.differencetech.com/difference/GET-vs-POST-HTTP-Requests>

Accessibility

HTML5 and Form Accessibility

- Although not all the HTML5 elements are compatible with older browsers they do degrade suitably so that issues will be ignored by the browser and display and provide functionality of the forms in a suitable manner
- <https://code.tutsplus.com/tutorials/rethinking-forms-in-html5--net-11620>

Field Sets

- Using the detail from Mozilla I structured the forms so that they were appropriate for screen readers using field set and logical for the user visually using legends with appropriate headings

https://developer.mozilla.org/en-US/docs/Learn/HTML/Forms/How_to_structure_an_HTML_form

Lists

- Lists allow the user to choose an item from a predefined lists
- This aids the data entry
- However it isn't limited to the data within the list

Placeholders

- These have been used to guide the user when completing the form

Labels

- Each element on the form has been designated a label using for
- When clicking on the label it will place the cursor in the element that's related in most browsers

Date

- The date field isn't compatible with some browsers
- This doesn't mean data cannot be entered but it doesn't mean the validation used in HTML5 won't work
- Also the range that is restricted won't be in place that is implemented here
- This can be rectified with serverside validation and the use of a DB
- This was out of the scope of this project but will be considered in the future

JavaScript Jump Menu

- One error that came up pointed to a JavaScript jump menu
- This was the dropdown that allowed the user to choose the type of contact
- As the JavaScript was not forcing a reload of page or a new page then this issue was not actually taking place and can be ignored

Testing

- During development I continuously tested the HTML and CSS using the following tools
 - <http://wave.webaim.org/>
 - <https://jigsaw.w3.org/css-validator/>
 - At around 400 lines of CSS the order at which I was organising the code started to get a bit difficult to maintain
 - This was particularly true once the media queries were in place
 - I spent a good proportion of time rearranging the CSS with a personalised method of organisation using comments.
 - After this there was little need to use the CSS checker until the later versions
 - Most errors in the code were easily identifiable until the CSS file reached around 1000 lines
 - The use of the comments to layout my code supported my ability to quickly find the areas that would be causing issues and it was often easy to spot the errors, such as an occasional missing ;
 - However as I began to further develop media queries with more detail it became more difficult to manage.
 - The movement of comments, as identified in the issue below, meant that at times there was potential to miss small errors in the code
 - Therefore it was vital for me to use the CSS validator to support the continued maintenance of the code through the development stages.

Issue with Menu Layout

- The responsive menu was functional until a change occurred in one of the versions late on
 - It then stopped functioning at the higher breakpoint, keeping the menu appearing as it did in the 900px breakpoint
 - I used the CSS validator at first to inspect the CSS
 - This found an obvious error that didn't arise from standard blackbox testing.
 - Main.css was linked as ../main.css - when using the SSI's because the file containing the path was in a different folder I assumed I would need to use the path directing to the folder from the SSI file
 - However I should have been using the virtual path, the path that the browser will see as the work from the server will be coming from the

<http://cs.wellesley.edu/~cs110/reading/SSI.html>

- Once corrected I had two errors in the CSS at this point.
 - The first error listed as 786 was actually online 774. This was an extra /* that hadn't been closed properly
 - This would have occurred as I was rearranging my code.
 - The second error listed on 1091 was due to an extra RBrace } on the media query at the 900x breakpoint

- Fixing these two issues results in successful parsing of the CSS at version VDWP v19
- However this did not fix the issue with the menu layout.
- At this point I manually studied the code relating to the 1000px breakpoint as this is where the issue was occurring
- I firstly alter the apply-nav selector to grid-row 1 to see if this was having an impact

```
#apply-nav {
    grid-row: 1;
    grid-column: col-start 10 / span 3;
}
```

- This didn't impact at all
- So I decided to see if a change of background colour on body was impacting the page
- I.e. to see if the media query was "firing" correctly

```
body {
    background-color: red;
}
```

- This had no impact
- I then moved this up through each of the breakpoints, but no impact was occurring at all.
- I then thought to clear the browsing history as this had caused me a few problems in the past
- This changed the body, on further testing it also ensured the breakpoint was now working with the apply-nav button
- Therefore the issue was the previously identified errors found using the W3C CSS checking service

After further development I ran the CSS test again and received the following errors

W3C CSS Validator results for http://lsteve07.wdux1718.bbkweb.org/news_events.shtml (CSS level 3 + SVG)

Sorry! We found the following errors (2)	
URI :	Line
http://lsteve07.wdux1718.bbkweb.org/main.css	
70 footer	Parse Error [nav]
72 footer	Parse Error [footer) { text-align: left; }

- This was due to incorrect usage of the :not selector
- It was altered from the following incorrect form to the latter correct form.
- Once done this produced a pass result for CSS.

```
ul li:not(nav, footer) {
```

To

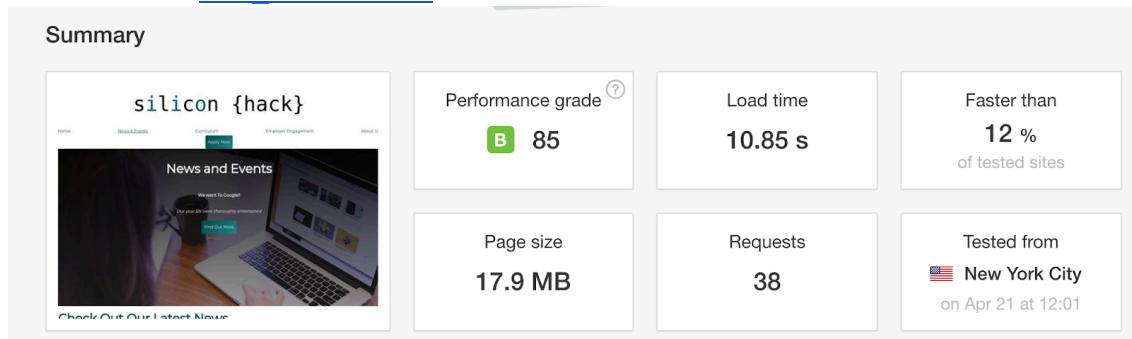
```
ul li:not(nav) :not(footer) {
```

Speed Testing

- Whilst testing I was conscious that the news and events page was loading slowly
- I decided to run a speed test on this page

Pingdom.com

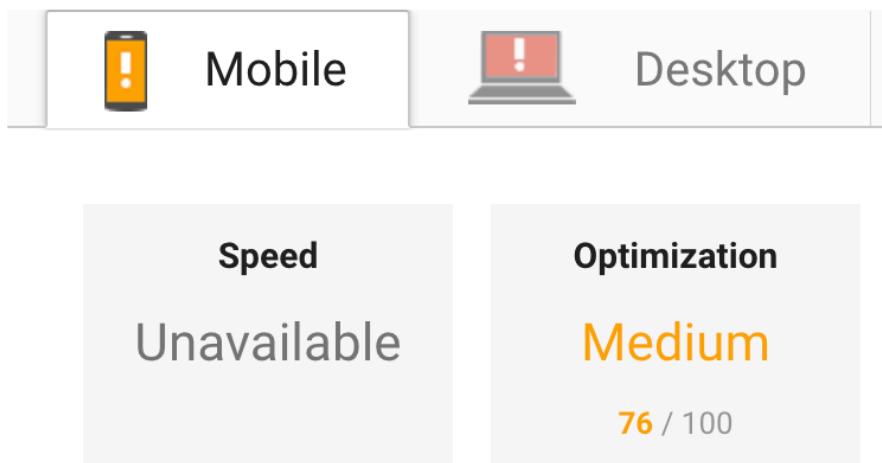
- Utilising the site pingdom.com I found that the site was taking 10.85s to load, as can be seen in the image below.
 - https://tools.pingdom.com/#!/dAw6wE/http://lsteve07.wdux1718.bbkweb.org/news_events.shtml



- Although listed as faster than 12% of tested sites, i.e. it wasn't the slowest. I still find this to be an unacceptable load speed
- Even though the images had been made progressive and some compression applied it was still producing too slow a load.

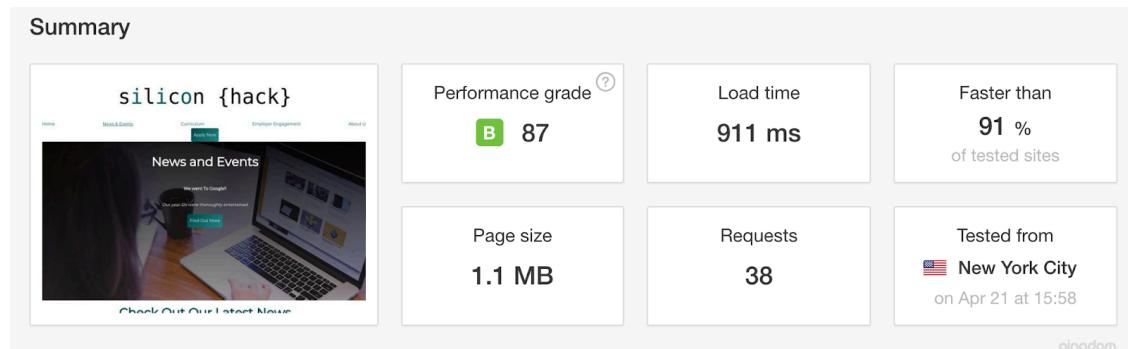
PageSpeed Insights - Google

- Researching through developers.google I found some further detail
 - <https://developers.google.com/speed/docs/insights/OptimizeImages>
- This brought me to googles PageSpeed Insights tool which is potentially a more reputable and reliable source of detail than the previous used tool from Pingdom.com
- However running this did not provide detail of speed
- It did outline that the mobile version was "Medium" in relation to optimisation (see image below)



- This also gave me further suggestions as to how to improve the optimisation of the page

- The key area to improve given by google was to further optimise the images on the page
- Further examining the images I had I realised that they were at a size that was well above the width that most users would reach
- Some of the images were over 5000px wide and even though compressed and made progressive they were still taking too long to load
- To resolve this I set a max width of the site to 1366px
- I also reduced image quality to 10% for each image
- This was chosen based on photoshops web-most-common template
- I then proceeded to alter the max width of all the images greater than 1366px to this size.
- After this was done I retested the news page again using pingdom
- This was the page that was noticeably slower, so I knew that improving this page would have a major impact on overall site performance and also the impression a user gets of the site.
- Page size was reduced to 1.1mb and load time to 911ms. It was now faster than 91% of tested sites.



HTML Validation

- Validating the HTML using <https://validator.w3.org> brought up errors with the iframes for YouTube
- Research found the following
- <https://www.htmlvalidator.com/CSEForum/viewtopic.php?t=2144>
- Changing the attribute *allowfullscreen* to
 - *allowfullscreen="allowfullscreen"*
 - rectified the issue

Date Warning on About Us Page

1. **Warning** The `[date]` input type is not supported in all browsers. Please be sure to test, and consider using a polyfill.
From line 365, column 41; to line 365, column 113
`<input type="date" name="DOB" id="DOB" min="2002-09-01" max="2001-09-01">`

- This warning comes up on the about us page.
- It is only a warning so is not invalid however due to the date field not being supported by all browsers a future version of the site would need to check validation of this server side.
- This could be done by adapting the PHP script for the contact us form.

All other pages return no errors or warnings.

CSS Validation

- Validating the CSS using <http://jigsaw.w3.org/css-validator/> returned no errors

Accessibility

Aria

- Aria is a method to allow content to be more accessible to people with disabilities
- In this site I have used aria to define when a link or an image is actually a button
- This especially important where I wanted to have styling of buttons where it could not be achieved using the regular button tags
- The call to action buttons are design using regular `a` tags
- This isn't suitable for accessibility purposes
- For each a tagged button I have altered the role to button
- I also have used aria to support accessibility for the buttons on the home page
- Each image is used to represent a button and therefore the role of button has been applied to these also
- This has also been applied to the contact links on the home page

<https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA>

JS

- The use of the SSIs and separate JS files means that I can only load JS files when required
- Some of the JS is not required on all pages
- Therefore to load the code separately could slow down performance
- On the about-us page for example I load two scripts after the SSI for the footer
- These two files are only needed on this page

```
<!--#include virtual="ssi/footer.shtml" -->
<script src= "js/confirmemail.js"></script>
<script src="/js/staff-toggle.js"></script>
```

Line height

- When reading the site at a line height of 1 I felt it was difficult to make out clearly
- Research resulted in finding these two pieces of information in a stack overflow quote that refers back to W3

Many people with cognitive disabilities have trouble tracking lines of text when a block of text is single spaced. Providing spacing between 1.5 to 2 allows them to start a new line more easily once they have finished the previous one.

Therefore an initial line height was set to 1.5 but then after going through each and getting some feedback from a variety of potential users it was decided that 2.0 was a better fit.

Background Images and Colour Contrast Issues

- Whilst styling the news pages I initially wanted to use background images with white foreground text

- However although not coming up in accessibility checks due to using CSS background as a method of styling with an image I was still conscious of the backgrounds
- As previously described this was overcome using the RGBA alpha channel and a gradient from top to bottom
- Utilising a black gradient from 0.9 down to 0.7 for most images I was able to achieve a gradient that still added some styling but allowed text to be displayed near the top of the image and be readable on the whole.
- This was achieved in both the intros and the news pages and can be seen below

<https://simplyaccessible.com/article/pitfalls-colour-contrast/>

Salesforce

We had a fantastic day out at Salesforce, right up at the top of heron tower.

With fantastic views of all of the city we learnt about cloud computing, the way teams were made up in salesforce, and of course we had a fantastic lunch spread out for us with as much chocolate as we could eat.

A Student



Google Trip

It was Google! Need I say more?!?

An amazing trip with some exciting talks and insights as to how the world of Google works. We learnt about how they are moving into AI with deep mind. We also had a go on the famous google slide :)

A Student



Salesforce

We had a fantastic day out at Salesforce, right up at the top of heron tower.

With fantastic views of all of the city we learnt about cloud computing, the way teams were made up in salesforce, and of course we had a fantastic lunch spread out for us with as much chocolate as we could eat.

A Student



- Further reflection made me reconsider this approach however
- I felt that the contrast was too great in the news items
- As an introduction with minimal text I felt this was ok but for somewhere with more detail I decided that a different approach should be taken

- Therefore I decided to create each into a heading section only and then add two images rather than one to go with the short description written by the students
- I then made the heading with background to be 40vh so that it would separate the content and give some of the style I was seeking
- The background colours have been adjusted with the gradient technique described earlier so that there is a slight blacking out of the content to give contrast to the heading text for accessibility purposes

Navigation and Contrast Issues

- Testing brought up contrast issues with the hover for the navigation that was being applied to the main navigation and also the links to the contact us form
- WAVE identified that to meet AA and AAA a darker colour would be needed
- I changed from the original orange selected from the colour swatch
 - #e08337
- To the darker orange of
 - #b1550a
- This then met the AAA for larger text, which the link texts are.

Navigation Burger Menu

- The burger menu flagged up an error of missing an alternative when testing through Wave
- An alternative text was added but as this also had an Aria to identify this as a button it wasn't a major issue.
- The other error it would throw out was that it was an image without a link, on reflection this should not have been set as an image but could have been designed using CSS and HTML alone, thus saving on both file storage, bandwidth and any confusion that might occur with this type of image for a user.
- In a future iteration of the site it would be redesigned using HTML and CSS

Tab Index

- When designing the site I tested the keyboard access of the pages
- The section the contact buttons was ordered with the apply-now button at the start using the grid, however in the html it was in a different order, this meant that the tab order was incorrect. I therefore went back into the html and reordered this.
- The tab index for a site is important for accessibility purposes
- When designing the HTML I ensured that the structure of the site was considered for the purposes of the user and a logical transition of tabbing through content
- This is especially important when using grids as the items can be repositioned in any order
- This would then cause issues when tabbing through
- Where there might have been a change of order I ensured that this was reflected in the HTML rather than forcing the change through the use of tabindex
- This was particularly important for the use of grid layout as the CSS can be used to reposition grid items into a different order from that in the HTML
- Where I decided on a change of layout I made sure the HTML reflected this also at this stage rather than using CSS to move items around the grid

- If I were to carry out a change in the future I would also need to ensure that the tab index was suitable to coincide with the visualised structure or alter the structure of the HTML document.
- This could be achieved with the tabindex property
- https://www.w3schools.com/tags/att_global_tabindex.asp

Evaluation and Further Improvements

Minification

- The CSS, HTML and JS files have not been stripped of empty space and comments for the purpose of this project
- To ensure improved load times for the final version of the site then a minification service would be used to provide this
- At present it has not been applied as the site is yet launched

<https://www.incapsula.com/cdn-guide/glossary/minification.html>

Improvement of Code

- As previously pointed out in the above there are areas where the code can be improved
- Over the course of the building of the site it started to get difficult to remember which classes and IDs did what in the CSS and thus there will be instances where this is repeated code
- There will also be cases where there is potentially redundant code
- All of this will have some performance impact on the site in relation to download speeds
- It will also make the site more difficult to maintain
- As a developer over time it would make sense to start to develop a personal “house style”
- I started to do this in places with classes such as five-col-section-header and the various classes to show and hide content on different devices
- This is something I should investigate more and invest time in as a developer
- In the short term it would be quite laborious a job slowing down development but in the long term it would see much improvement in the further development of a site created this way and also of future sites that could use this personal standardised CSS formatting
- The same of course can be said of JavaScript where there can be time saved developing a personal library of code to be utilised in future projects

Bibliography

<https://css-tricks.com/almanac/properties/f/flex-direction/>

<https://en.wikipedia.org/wiki/WAI-ARIA>

https://developer.mozilla.org/en-US/docs/Web/Accessibility/An_overview_of_accessible_web_applications_and_widgets

<https://css-tricks.com/use-button-element/>

<https://stackoverflow.com/questions/13002731/responsive-design-media-query-not-working-on-iphone>

<https://philipwalton.github.io/solved-by-flexbox/demos/vertical-centering/>

<https://discourse.wicq.io/t/a-way-to-hide-text-visually-but-not-from-screen-readers/510/10>

<https://responsedesign.is/strategy/page-layout/defining-breakpoints/>

<https://www.nccgroup.trust/uk/about-us/newsroom-and-events/blogs/2016/may/why-background-images-are-slow-to-display-and-how-to-make-them-appear-faster/>

<https://www.abetterlemonadestand.com/optimizing-images-for-web/>

<https://philipwalton.github.io/solved-by-flexbox/demos/vertical-centering/>

https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Box_Alignment_in_CSS_Grid_Layout

<https://css-tricks.com/almanac/properties/a/align-items/>

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

<https://stackoverflow.com/questions/27539262/whats-the-difference-between-align-content-and-align-items>

<https://codepen.io/pen/>

<https://css-tricks.com/navigation-in-lists-to-be-or-not-to-be/>

<https://philipwalton.github.io/solved-by-flexbox/demos/vertical-centering/>

<https://css-tricks.com/navigation-in-lists-to-be-or-not-to-be/>

<http://www.webaxe.org/leave-accessibility-to-experts-please/>

https://www.youtube.com/watch?v=aS_5IhneMHC

<https://helpx.adobe.com/photoshop/using/matching-replacing-mixing-colors.html>

<https://forums.adobe.com/thread/1017335>

<https://helpx.adobe.com/photoshop/using/positioning-elements-snapping.html>

<https://helpx.adobe.com/uk/photoshop/how-to/add-motion-blur-effects.html>

<https://www.youtube.com/watch?v=NVKju7nZvDI>

https://www.w3schools.com/cssref/css_colors.asp

<https://www.youtube.com/watch?v=63Wdt10lGoY>

<http://paletton.com/#uid=72POu0kw0k5o8s3sWo7FpfCPpaa>

<https://www.wikihow.com/Centre-Objects-in-Photoshop>

<https://www.favicon-generator.org/>

https://www.w3schools.com/css/css_image_sprites.asp

<https://css-tricks.com/css-sprites/>

<https://xcentral.com.au/our-blog/2013/july/07/choosing-the-right-laptop-screen-resolutions-explained-simply>

<https://graphicdesign.stackexchange.com/questions/45217/how-do-i-divide-a-canvas-into-9-equal-squares-in-photoshop>

<https://helpx.adobe.com/photoshop/using/gradients.html>

<http://www.prismo.ch/comparisons/desktop.php>

<https://websitespeedexperts.com/advantages-and-disadvantages-of-svg-images/>