Thomas Sherlock & Jamie Meininger

Coursework 3 Report
Degree Apprenticeships Microsite

Technical Aspects of Development

This site is built using HTML5 and CSS3. The interactive aspects of the site come from a small amount of JavaScript loaded as scripts at the bottom of each page they're needed on. We have opted to create a more complex site which closely follows one of our designs from the Design stage. To do this, we used several technologies to assist in the fast delivery of the project.

Backend

There's not strictly a backend server, as GitHub pages completes all page building for us, meaning at a bare minimum, no other files are needed other than an `index.html` to have a running site.

Bootstrap 4

We opted to use the latest version of Bootstrap 4 to provide a basis for our site layout. Information on bootstrap can be found here. This gives our site the famous grid structure, using CSS to split the webpage into 12 columns, allowing us to develop a better site layout with more ease. Using bootstrap also gave us other benefits too:

- Glyphicons - we used these all over the site as small vector graphics and icons. They're bundled with bootstrap and can be scaled up or down in size. Colour - Bootstrap comes with some basic colour libraries to be used. Although we overwrote a lot of these, basic text and background colouring as black or white was done through bootstrap.
- Enhanced Forms - Bootstrap provides an advanced form user experience which we put to good use within the contact page. It allows for more advanced data entry, and better look and feel of standard form types.
- CDN Delivery - we opted to use a CDN for delivery of bootstrap. This means that the load will not be on our GitHub server for downloading the CSS, but also offers the chance that the load speed will be reduced if the user has visited another site with Bootstrap on it, as browsers will cache the CSS. For users on mobile, this will greatly improve their user experience, as slow loading was noticed as a main factor for decreased site traffic in the design stage.

Google Fonts

Google's free font library was used in our site to provide a better font service. This was included in the top of all our pages as a `<link>` tag.

Go.js Framework

To create the interactive mindmap elements on the schemes page, we utilised Go.js, a third-party framework for creating graphics. To create this, a separate .js script file was created in the `js` folder, and referenced in a script tag on the schemes page. A canvas element is used inside a div which creates the mindmap element. To fill the mindmap with data, a JSON object with all the data and the links between them was added to the external script file.

Intro.js

This was used to create a simple introduction screen for new users. Attached to a specific div is a small pop up that explains how to interact with the site, which can be navigated through or simply closed. This was used on the casestudies.html page.

File Structure

The `gh-pages` branch contains all the code & files needed to run the site. All html files are included in the root directory to conform to the GitHub Pages requirements, and to aid in ease of use for maintainers.

All CSS has been refactored into separate CSS files for each page. These are stored, with the Bootstrap CSS files, inside the `/CSS` folder. The .css file has the same name as the .html file for ease of use. The Bootstrap CSS files are placed in here in case the CDN delivering Bootstrap goes down. Minified versions of this CSS are also placed in here to reduce page loading time.

JavaScript files have been included in a separate `/js` folder for ease of use. Again, the full bundled files and minified versions have been included in here to speed up loading if the CDN goes down.

All images have been included in a separate `/res` folder. They are currently all placed at the root level here, to allow for shorter URLs when referencing them. We did this to ensure the root level folder would not be swamped with excess files, and to ensure they were slightly obfuscated by the extra folder.

Process from Design to Development - Delivery details

The site is split into 5 content pages, plus a menu. These files are stored in the root directory as explained above. To develop a full website from a design, we first selected a design. We chose the second of the designs, which gave us more than 5 pages, featuring lots of content in a slightly deeper navigational structure. We also selected this design as it allows us to include full page images which illustrate the benefits of a Degree Apprenticeship.

Each page was developed in the same way:

- First an empty page was created with the basic `<html>`, `<head>`, `<title>` and metadata tags. Bootstrap and GoogleFonts was also included here.
- The header & footer was inserted.
- Divs were created with the class "container" and "row". Then to add content, divs with the col class were used to split the page correctly.
- Content added as "Lorem ipsum" placeholder text.

- W3 validation was used on each page to ensure the HTML was at a correct standard.

Header/Navbar

On all pages (except for the menu), a header has been included. This was mentioned in the design stage, however for development a "hamburger button" was used instead of the word "menu" on a button. This, when clicked, links to the menu page. The navbar contains this button on the top right in green, and uses whitespace and a lack of content on the rest of it to draw the user's attention away from it and onto the main page's content.

Footer

As with the header, all pages except the menu and index (home) pages have a footer implemented on it. This footer contains some basic information about the site - a site map for users with accessibility issues, as well as social media and contact links for the social networks, etc. This uses the bootstrap 3 column layout to give a uniform feel to it. This uses the house style of white text on a green background. The footer is coded to have a fixed position on the bottom of the page, meaning the page length will stretch to hold the footer under the main content. The footer was not included on the paper prototypes as it was at a greater level of detail than the prototype, so we decided to create a new design for it.

Homepage - index.html

Developing the index page was very similar to the designs chosen. There were some minor changes made to increase the usability & user experience of the site. The page still contains a large full-page image, with a simple logo and button on it, which entices the user in to use the site. The developed page however, has a narrow bar at the top and bottom of the page, which uses whitespace to draw the user's eye to the image. There is also a "hamburger" style button in the top right, which is present on every page as the header/navbar. In terms of colour schemes, we upheld the main design principles and used the same colour scheme throughout the page.

Menu - menu.html

The menu page was also developed identically to the design, with 4 different images being used as the background for 4 menu items. For this, as per the design, the menu hamburger bar is not used to ensure a clean feel. To develop it, CSS was used to create a hover able button, so that when the mouse hovers over a

button, the entire background shows a different image.

We opted to split the menu onto a separate page, so that when the menu button is clicked from another page the user is redirected to a whole new page. This was done to solve several issues:

- Avoid code duplication - this will ensure that when developing, the code for the menu is stored in a single file, and not duplicated unnecessarily throughout the other page's HTML. This means that development (& changes to the menu page) can be done centrally on one file, yet the effect will happen on all pages.
- Accessing the menu from multiple pages - as above, the menu can be accessed in the same way from multiple pages

- Speeding up page loading, in 2 ways
    - not loading menu on every page will increase the speed of every page, as the images on the menu page will not need to be loaded on the other 5 pages. not loading menu when not clicked on will increase the speed of all pages that the user accesses directly (for example through hitting the `/contact.html` page directly.) as, if the user does not click onto the menu, the images will not be loaded unnecessarily.
    - Caching of images - The first time the user clicks on the menu, the browser (assuming it's recent) will cache the CSS of the page, including the CDN version of bootstrap, and the images. This means that on subsequent visits, the CSS and images will load faster, making the menu as fast as possible.

About - about.html

This page was not designed in any great depth in our chosen design. However, in the alternative design we opted not to use, a page for explaining Degree Apprenticeships was designed and planned. However, through consultation with our planned users, we tested an alternative layout that adhered to both our Design Principles and the Gestalt principles, which used a split page between an image and a small block of text. More content can be viewed from the user's scrolling down, showing a similar layout with different content & different images. To split the page in half vertically, the bootstrap column attribute was used which split the page into 2.

Case Studies - caseStudies.html

This page also very closely follows the designs. A large div contains an image showing the timeline for a real apprentice. This can be interacted with through finger scrolling on a mobile device, or click scrolling on a computer. To change the contents of the timeline, apprentice images sitting in a div on the left are shown, and can be clicked on to show a different timeline. A small amount of CSS was used to make the apprentice images briefly pulse in size when hovered over, to entice users to click onto them.

To explain how to do this, a JavaScript framework called intro.js was used to display an overlay with instructions on it. intro.js allows you to attach this to a specific div, so each section has a brief introduction that can be clicked through to explain to new users how to use the site. In a production site, this would be linked to a user's cookies and would only display on the first visit.

Contact - contact.html

The contact page has been developed to the exact specification of the design. A contact page has been implemented, which, if connected to a backend, would use a HTTP POST to send information to the site's backend server. A small amount of JavaScript has been used to mock up this form submission, using a

JavaScript function called switchVisibility () to show/hide a form submitted message. The bootstrap 4.0 form classes were used to great effect here to design a more responsive form.

Schemes - schemes.html

This page was designed to have a full screen interactive mind map on it. To develop this, an additional framework in the form of Go.js was used to create the mindmap. Through some basic user testing, it was also decided that a small menu on the side would be needed to give some details on how to interact with the mindmap

Note: some content on this page (such as clickable company names & exact industry information) has not been included, as such data was not readily available at the time of development.

Running this site

To clone files locally:

- First, run `git clone https://github.com/mrtomsherlock/web-design-coursework.git` in a folder of your choice in terminal/cmd.
- Then navigate to this repository through terminal/cmd and type `git checkout master`
- This will change the files in your source repository to the ones in the gh-pages branch, which GitHub uses to build the site from.

Running it:

As we are using GitHub pages as our web server, there are no server files on the repository. This means to develop/view the site locally, one can simply open the relevant HTML pages from the folder they're stored in.

GitHub repository

The GitHub repository for this site is

We are using the `master` branch to host all code for the live site. The master branch does not contain relevant code for this assignment.

Published Site URL

The published site is running at this url. Users can access this through any web browser once logged in through the