COM1008: Web and Internet Technology
RESIT, Summer, 2023

1. Introduction

This assignment will test your ability to create a website using Mobile First Responsive Web Design. It will test your understanding of stages of creating a website, as well as your coding skills in HTML5, CSS, JavaScript and the Canvas.

This is an individual project. The work you submit must be your own individual work. *See Section 5 for further details on this.*

2. The Website

You will develop a website for your own fictitious web design company.

The following is a list of pages that must be included on the web site:

- A home page (named index.html);
- A founders page;
- A news page;
- A demo page for a canvas demo;
- A contact page;
- An accessibility page;
- A legal and security page;
- A design page;
- A testing page.

The main menu at the top of each page should provide links to the following pages: home; founders, news, demo.

There should also be a menu in the footer that provides links to the following pages: accessibility; legal&security; design; testing, contact.

Further details about each of these pages is given in the following sections.

2.1 Home page

- A home page that welcomes people to the website of your company. What will you have on this page that extols the virtues of your company?
- Invent a logo for your fictitious company. (*Hint*: This logo should appear on every page of your website.)
- This page will demonstrate your ability to create an attractive page that loads quickly.

2.2 Founders page

- A page with details of the founders of the fictitious company.
- This should include photographs and short biographies for three people.
- All the information can be fictitious to match the fictitious company you are creating. The photos can be three different pictures of you – you can use the same picture and digitally alter each picture if you prefer. (Note: You should make it clear that the profile information is fictitious by including a note to this effect on the page.)
- Consider carefully how you lay out the photographs (images) in relation to surrounding text. Make it look neat and professional.

2.3 News page

- A news page about the fictitious company.
- This should have at least three fictitious news items. Each fictitious news item should be accompanied by a photograph related to the fictitious news item. The fictitious news items should be about the company, e.g. voted the best company in the region, won an award for something, etc. Try to base these on things you read online about other companies so they are at least realistic.
- You should use CSS for typographic effects for the news stories. Be adventurous. Pay careful attention to which HTML5 semantic element it would be appropriate to use for news items.

2.3 Canvas demo page

- A separate page that demonstrates the JavaScript and Canvas work you do for com1008. ***Note:*** This is the **only** page where JavaScript can be used in your assignment. You must not use JavaScript on any other page of your website.
- An interactive match-the-pairs game should be produced..
- Multiple copies of the same image (e.g. the company logo) should be displayed on the canvas

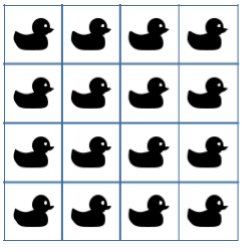

Figure 1. A 4x4 grid of images

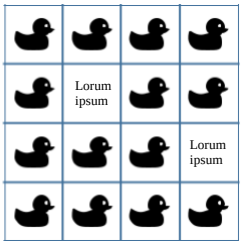

Figure 2. Two pieces of text 'lorum ipsum' revealed by clicking on the images

as a 2D grid of images – a 4x4 grid of images, giving a total of 16 images, will suffice (see Figure 1 for an example where the logo of the company is a duck).

- The images that are displayed can be either (i) image files (e.g. jpg or png) drawn using the Canvas command to draw an image or (ii) they can be drawn using any Canvas drawing commands that draw lines, circles and rectangles. The logo does not need to be complicated – many logos are quite simple.
- When the user clicks on an image in the grid of images (i.e. clicks on any part of the relevant grid cell containing the image), it is replaced by a phrase (e.g. 'HTML5 compliant', or 'Promoting Responsive Web Design') that describes some positive aspect of your fictitious company.
- The user then clicks on another image (grid cell) and it is also replaced by a phrase.
- If the two phrases match, then the phrases stay displayed on screen (see Figure 2). If the phrases do not match, then, after a short delay, the text phrases disappear and the images are redisplayed.
- The game finishes when all the pairs of phrases are matched, so that no images are displayed. A message should be displayed that the game is over.
- This page serves to demonstrate your fictitious company's ability to create interactive graphical experiences on a web page. Through the text that is displayed they will also learn a little about the positive attributes of your fictitious company.

2.5 Contact page

- A contact page, which includes the fictitious contact details of the company (e.g. fictitious address) and a form for users to send comments to you via your e-mail address.
- A user should input their e-mail address in a text box in the form and input their comments in another text area on the form. The form should also include a button, which, when clicked by the user, sends the contents of the two text areas to your company's e-mail address. (Note: Since the contact details for the company are fictitious, the form should be sent to your University e-mail address. This button is for appearance only as we do not need the email to actually be sent.)
- Make sure the form is stylish – you are again demonstrating that your company can use HTML and CSS to make a form look good.

2.6 Accessibility page

- This page covers two things: (i) it includes your accessibility statement for your website; (ii) it includes the reasoning behind your choices.
- You may have addressed accessibility in a range of ways on the website and you should include what you have done on this page. Give reasons for the accessibility approaches you have chosen. Give references to support your reasoning (e.g. the relevant W3C url and others).
- Make sure you also state what accessibility issues may be handled by other software, e.g. screen readers, and give links to advice that could be followed to make this work.
- Use page sections as appropriate.
- This page should be a maximum of 250 words.

2.7 Legal and security page

- State how you have addressed any legal issues related to your website, e.g. copyright for pictures, statement about GDPR.
- State how you have considered security in relation to your website, e.g. security of contact form information.
- Give references to support your reasoning (e.g. official information on copyright, GDPR, etc).
- Use page sections as appropriate. (Hint: you should be considering which of HTML5's semantic elements is appropriate to use.)
- This page should be a maximum of 250 words.

2.8 Design page

- A description of how you designed the website using 'Mobile-First Responsive Web Design'
- Assume you are using up-to-date browsers. Do not consider old browsers.

You must include the following in separate headed sections on this web page:

- Introduction: A brief statement about the general idea that influences your design. Are you aiming for a serious site or a more relaxed site? Perhaps this relates to the kind of fictitious company you have invented?
- Site Map: draw the site map for your website and justify the structure. The structure should be quite straightforward given the pages described above. Don't overcomplicate it.
- Design mock-ups: You must use a mobile-first approach for the assignment. Create design mock-up diagrams that show, *as a minimum*, the mobile design and the desktop design. You could use wireframes for this, paint software (e.g. Photoshop) or hand-drawn sketches (that are then

scanned in or photographed) to produce diagrams similar to the following examples:

- The diagram labelled "Normal, Narrow, Mobile" in the Overview section at: http://webdesignerwall.com/tutorials/responsive-design-with-css3-media-queries/comment-page-1
- The diagram labelled "*The "extreme" versions of the new website design"* at: https://www.smashingmagazine.com/2013/03/building-a-better-responsive-website/;

It is ok if your diagrams are more sketch-like than these.

You must decide how many breakpoints and tweakpoints to use in your design and write a short justification of this and how each affects the design. We expect to see careful consideration of this. Tweakpoints can be important in showing small changes to your design as the browser window is resized. (Note: the exact position of the breakpoints could be changed during the development stage, and you may also add more tweakpoints.)

You must justify your design decisions. (Note: You do not have to give designs for every page if some of the pages are very similar. Just say that page X is similar to page Y.)

- Menu System: A consideration of the menu system that is being used. What style have you chosen and why? How will you deal with both a small mobile screen and a large desktop? How will your navigation area adapt to each display resolution? You must **not** use JavaScript for the menu. The menu should be implemented using **only** HTML and CSS. *If you use JavaScript we will turn this off and your menu will fail.*
- This page should be a maximum of 500 words, so use pictures appropriately.

2.9 Testing page

You must include the following in separate headed sections:

- Optimisation: for example, have you considered page and image loading times and what to do about this? Discuss this (and include examples by using the web developer tools in the browser).
- Debugging: you should make use of html and css validators. What were the results? Can you explain them?
- Browser testing: discuss tests on different devices and different browsers and include screenshots of this – you only need to consider up-to-date versions of web browsers in your testing. For the purposes of this assignment, it is acceptable to use emulation tools for testing (e.g. the Responsive Design Mode available in Firefox or Chrome's Device Mode). If you use emulation tools, then consider different emulation tools in different browsers.
- Accessibility: you must also consider accessibility testing and show the results of this with screenshots included as pictures on the testing page. You should make use of WAVE (https://wave.webaim.org/) and include a report of its output and what you did about this. Also show the output when you have fixed any problems it highlights.
- This page should be a maximum of 500 words, so use pictures appropriately.

2.7 General requirements

You must satisfy the following when constructing your website:

- The overall website design must be consistent.
- The website must be legible, e.g. is there good contrast between text and background?
- Each html <head> element must include a meta element identifying you as the author(s). Also include comments to this effect in other file types.
- You must use the HTML5 semantic elements when structuring each webpage, e.g. header, nav, main, footer, article, section, etc. These must then be styled in the relevant stylesheet(s).
- Appearance must be controlled by the linked stylesheet(s), **not** by inline styles.
- Use of @media queries – are these used in a structured and controlled way? (Remember, it is mobile first, so we will be looking for use of min-width. If you are using max-width, you have probably misunderstood mobile-first RWD.)
- Economic use of properties in a stylesheet, where appropriate, e.g. margin a b c d, rather than setting the top, right, bottom and left margin separately
- All source code should be well organised and neatly laid out, e.g. using *indentation*.
- You must include appropriate comments in your source code. As an example, a comment might describe the purpose and reasons for using a particular CSS rule or use of a particular HTML element. We will look at these comments carefully since the comment text will be unique to you and reflect your understanding. However, do not comment on simple pieces of source code

where the meaning is obvious – use your common sense here.

3. Handin via Blackboard (via Resit link)

- The website must be put in a single zip file called **surname_firstnames_com1008.zip**, e.g. **Smith_Jane_com1008.zip**.
- The home page of the website must be index.html, so that it is easy for us to identify which file to load first.
- Make sure you include every file, including all relevant images. (Remember: using relative addresses on your website for the links between pages and resource files such as images is important so that the website can be easily copied onto a different server.)
- Do not include videos, as including these will create a large zip file, which will crash Blackboard, given the number of students handing in work – there is previous experience of this and it wasn't pretty. If you want to use videos, then put them somewhere like YouTube and then put a single image on your website and a link to the YouTube video.
- Remember to identify in every separate file that *you* wrote the code.
- **Please check your zip file is correct before you hand it in.** When you have created the zip file, you should check that it has been created correctly by unzipping it again in a different folder on your computer and checking that the pages open and work. There have been cases in previous years where the zip file was corrupt or the file structure had been flattened.
- **Also, please check that the correct zip file has been uploaded to Blackboard.** Uploading the correct file is your responsibility. We can only mark what is uploaded. If you upload the wrong file then we can only mark that, which may lead to failure. If you upload a file that is not your own, it will be classed as unfair means.

4. Marking

4.1 The Website (50%)

The majority of marks are for producing a website that fulfils all the requirements. Read them *carefully*.

- General (20%) – includes look & feel, RWD behaviour, content, menu, accessibility considerations, quality;
- HTML (15%) – includes comments, layout, content of <head> element, use of semantic elements, quality;
- CSS (15%) – includes comments, organisation, layout, economic use of properties, RWD, use of @media (mobile-first), quality.

4.2 Accessibility, Legal & security, Design, Testing (25%)

- The *content* of these pages will be marked separately.
- You must include each of the things asked for in the above descriptions. Take heed of the list of sections required on the design and testing pages.
- Justifications should be given where appropriate (e.g. why were particular breakpoints chosen?), but these should be brief as there is a maximum number of words for the design and testing pages. Use diagrams to help explain things (as these do not count towards the word limit). A good diagram embodies attention to detail. The accompanying description should give reasons for choices, not waste words on describing what is obvious in the diagrams.
- Justify accessibility statements and legal & security issues by citing references on the relevant pages. The list of references does not count towards the word count.
- The design mock-ups should be neat and reasons for the breakpoint(s) and any tweakpoint(s) given.
- Make sure to include accessibility testing pictures on the testing page, as well as pictures of your website running on different screen resolutions and browsers.

4.2 JavaScript Demo page (25%)

This will include:

- Using appropriate coding structures, as well as appropriate comments and layout (5%)
- How well the basic application works and the completeness of this, including quality and creativity of the work (20%)

5. Practical considerations

5.1 Unfair means

The standard Department rules for use of unfair means will be applied, as described in the undergraduate student handbook:

https://sites.google.com/sheffield.ac.uk/comughandbook/general-information (Menu: General Information, Assessment)

We are aware that there are lots of HTML, CSS and JavaScript tutorial sites on the Web. Do **not** copy them, since that would be plagiarism. Instead, only use them for learning.

Do NOT use Bootstrap or jQuery or any other similar frameworks/libraries for creating web sites. This will be treated as plagiarism for the purposes of this assignment.

You may use CSS reset or normalize. If you do, you need to make sure you comply with the license for each of those and make clear that they are not your work. Include any requested attribution in the relevant files that you use.

5.2 Late handin

Standard Department rules will be used for late handin – see:

https://sites.google.com/sheffield.ac.uk/comughandbook/general-information
(Menu: General Information, Assessment)

5.3 Code reuse

You may reuse HTML, CSS and JavaScript code that we wrote that is given in our lecture materials, as long as it is not code that is from another source that is being used to illustrate something – we may have used it to illustrate something, but you will not have permission to reuse it. If there is any doubt, assume you cannot reuse it.

5.4 Text editor

There are plenty of Web design tools available. The expectation for this assignment is that you will use a code editing environment to develop your website (e.g. Visual Studio Code or Notepad++ or similar). You might use more sophisticated tools to support your design process, e.g. drawing design wireframes, but not to develop your code for you. Typically, it is extremely obvious when one of these sophisticated tools has been used in code development as the files created contain lots of extraneous HTML and CSS, rather than only including what is required.

As part of the deliverables you will have noted that we expect code that contains comments. These will demonstrate whether or not you understand the more complex bits of HTML, CSS and JavaScript that you have produced.

5.5 Relative links

Make sure you use relative URLs in your code when referring to your own resources or to other files that you are developing – the reasons for this are explained in lectures. Links to relevant external URLs should still be given in full, e.g. http://www.w3.org/.

5.6 Keeping your work private

Your website should be developed on your own computer, **not** in a publicly-accessible folder.