The Development and Testing

Changes

Changes made to the website compared to the original plan were quite minor. The design was followed exactly as planned during implementation. However, the design breakpoints were slightly changed.

The website has been designed to not go smaller than 360 pixels wide as majority of phones currently in the market don't go to any width smaller than that.

The website's first major design breakpoint has been set at 800 pixels wide with minor changes taking place when width is 600 pixels wide. This decision was taken after taking into consideration that it wouldn't really make sense to keep on treating an 800 pixels wide screen as a phone. The navigation bar and the content of the main area could expand and be allowed to flow properly from that width onwards.

The website's main content stops increasing in size after it reaches 1000 pixels wide. This was decided as going up to 2000 pixels lead to nothing but really spaced out elements that didn't complement the website's design but rather hurt it. So, a decision to stop increasing after 1000 pixels was taken and it has proven to be quite elegant on testing on various screen sizes.

Organisation

Templates

To maintain a consistent repetitive user interface throughout the website, templates were created and replicated in every page of the website. Templates were created for the header, the navigation menu, and the footer. They are the same throughout the whole website pages.

Figure 1- Header and Navigation

Figure 2- Footer

Figure 3- Mobile-design Header and Navigation button

Also, a style template has been created to facilitate the creation of the Recommendations page and the Founders page. Both contain different content. However, both pages follow the same template set in the stylesheets.

Cross-Browser Compatibility

To ensure that the website is compatible with different browsers and to avoid the confusion and the extra style manually added to fix certain browser issues, I decided to use Normalize in my website.

I prefer to use Normalize over Reset since it doesn't wipe everything but rather keeps the basic styles that are common between the browsers and resets the other attributes that have been edited by different browser manufacturers.

File Structure

As per the original designing document, the same file structure was followed throughout the implementation phase. Various folders for different file types were kept. Also, within these folders, sub-folders were created in case extra organisation was needed (e.g. images folder)

Debugging

During the implementation phase, testing was continually being done after every edit to the website.

For JavaScript debugging, I used the console.log() function and the Web Developer Tools found in Mozilla Firefox and Google Chrome to properly test my code and remove any potential bugs

For HTML and CSS debugging, I used the Web Developer Tools found in Mozilla Firefox and Google Chrome to properly test the HTML and the CSS to make sure they were doing their intended job.

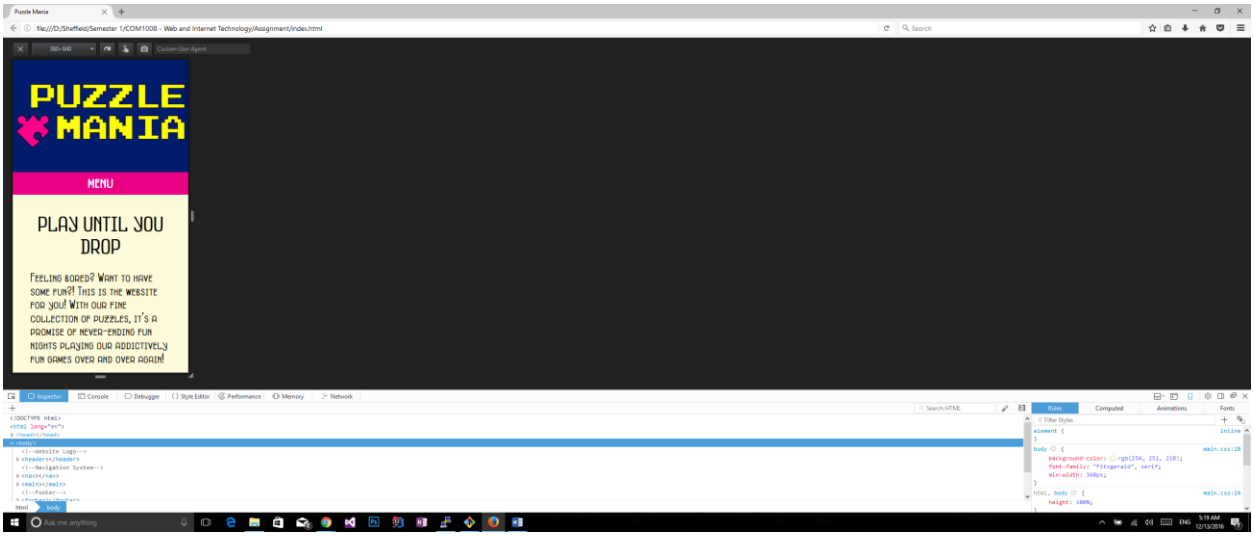

Figure 4 - Debugging HTML and CSS for Mobile-first CSS

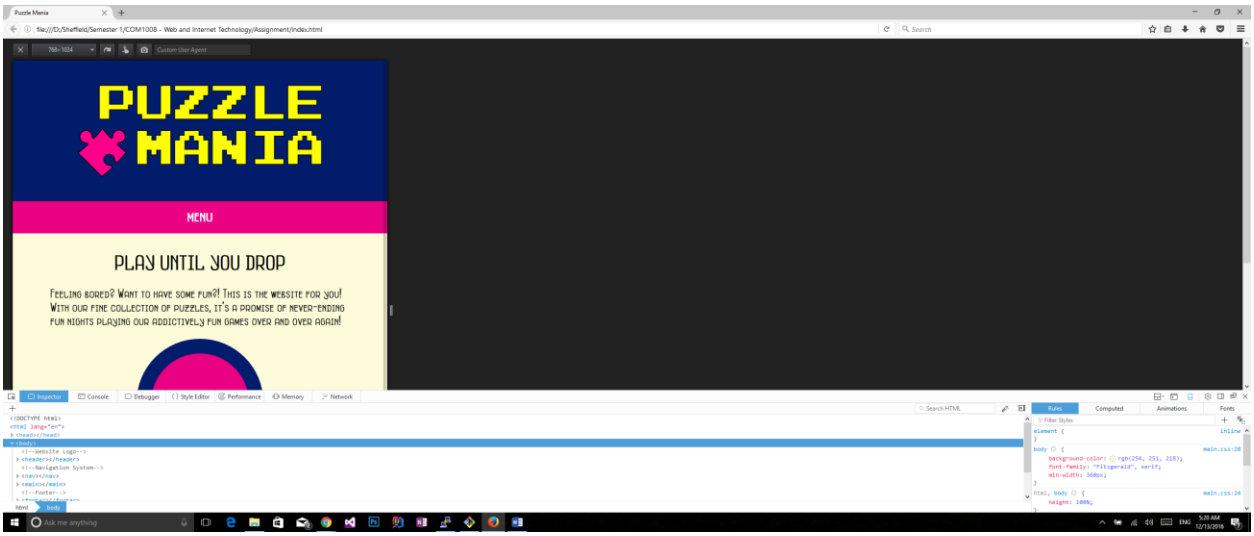

Figure 5 - Debugging HTML and CSS using Mozilla Web Developer tools to fix a tablet issue

Optimisation

To improve the website's speed, I made sure that all the images I created or added to the website were first compressed to ensure a faster image loading time leading to a more efficient website.

Also, I managed to keep my JavaScript (other than the game) to the bare minimum which is just being used in the navigation menu system on mobiles to avoid having the JavaScript code slow down the website.

Security

A potential security breach in the Contact Form would be Cross-Site Scripting. This would have been a problem if the user entered a string that contains HTML and JavaScript. This could have led to that code being executed on the web page causing a security breach that could have led to:

- Cookie theft
- Keylogging
- Phishing

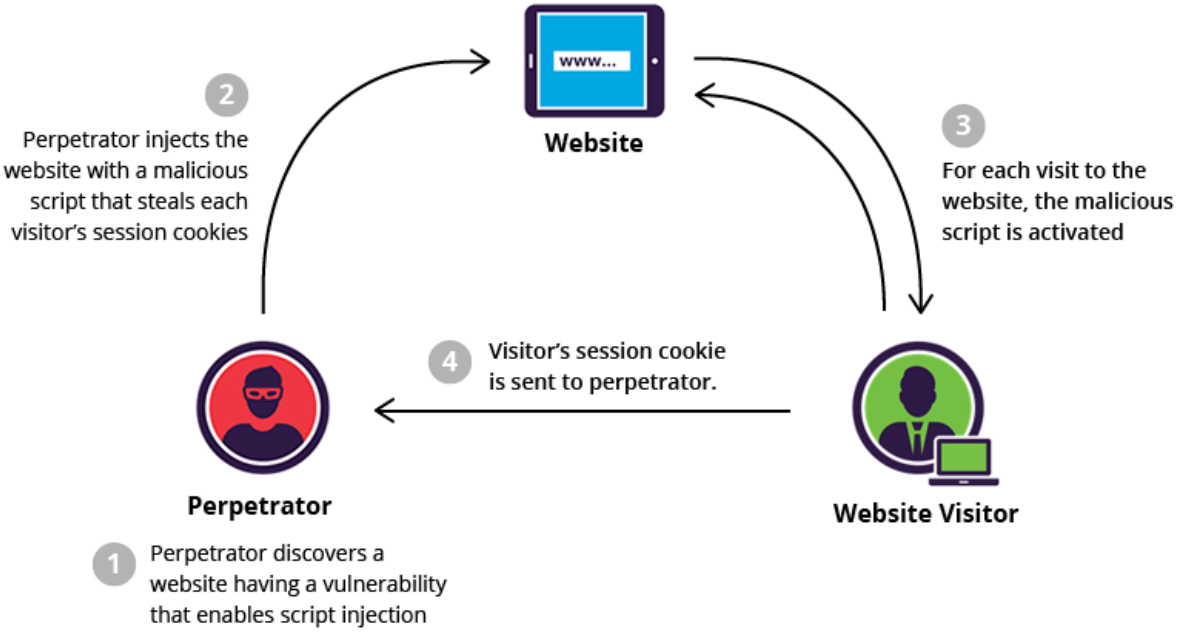

Figure 6- Cross Site Scripting

However, the script being used "FormMail.pl" has been updated to remove cross-site scripting vulnerabilities by converting all the characters into their HTML equivalents to prevent them from being interpreted as code. (As found in Lecture 18, COM1008)

Another security issue would be that the company address is laid out in public on the website which could lead to physical security breach which could then lead to the company's data being damaged or stolen. A good measure to take in this scenario would be to have the data centre for the company at a different remote location to try and prevent any potential breach of any kind.

Testing

I was required to test the website on the last two major versions of web browsers in my testing. So, I decided to use browsershot.org to test out my website and make sure that it's HTML & CSS were being rendered properly. Google Chrome and Mozilla Firefox both worked fine. However, Internet Explorer seemed not to respond to the external font being used in the website even when EOT format of the font was tried.

I also tested the website against Google's mobile friendly checks and received a 100/100 for mobile friendliness. (Using https://testmysite.thinkwithgoogle.com/)

Throughout the entire implementation process, the Web Developer Tools were constantly used to test each feature as it's being implemented and how it would affect the rest of the website. In the end, the website was tested using these tools then using some Google Chrome extensions that tested how accessible the website was.

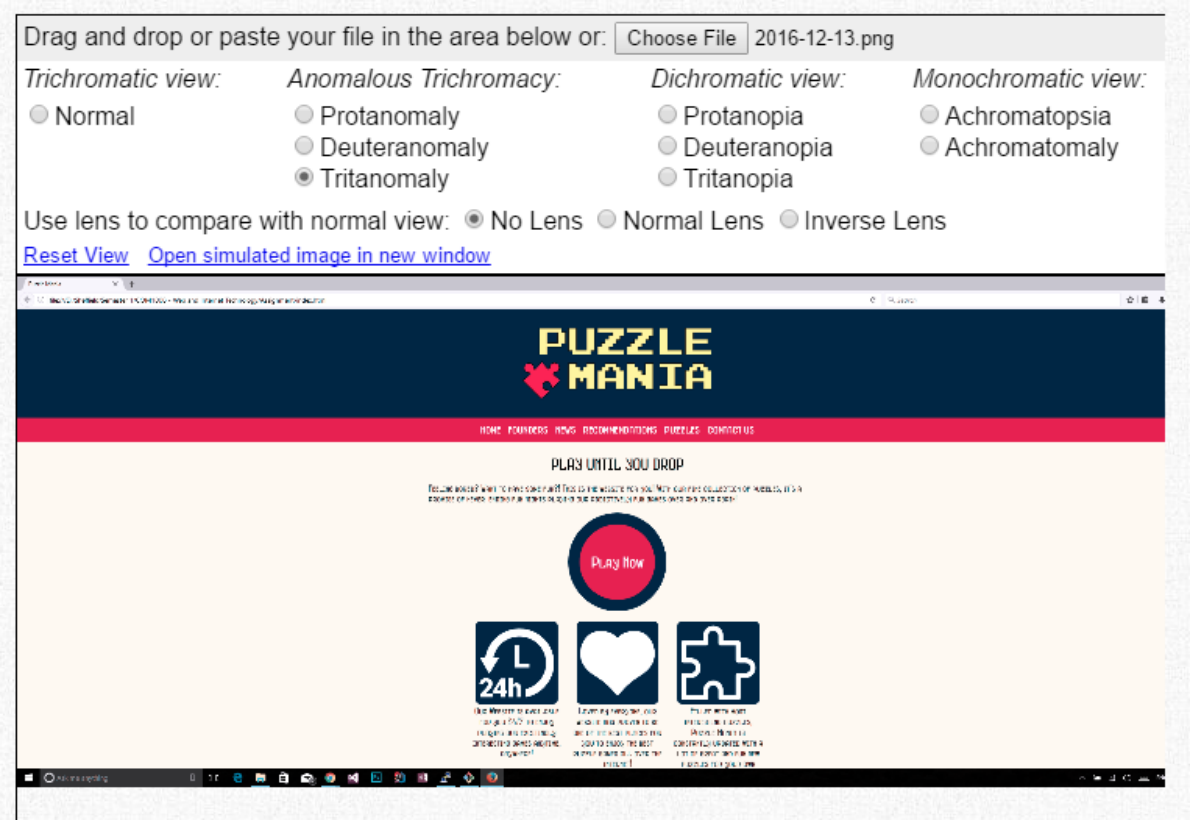

Figure 7- Color Blind Simulator check

Also, screenshots of the website were uploaded to a colour-blind simulator that showed how the website is to be perceived by a colour-blind person. This was done to confirm that the website is easily accessible.