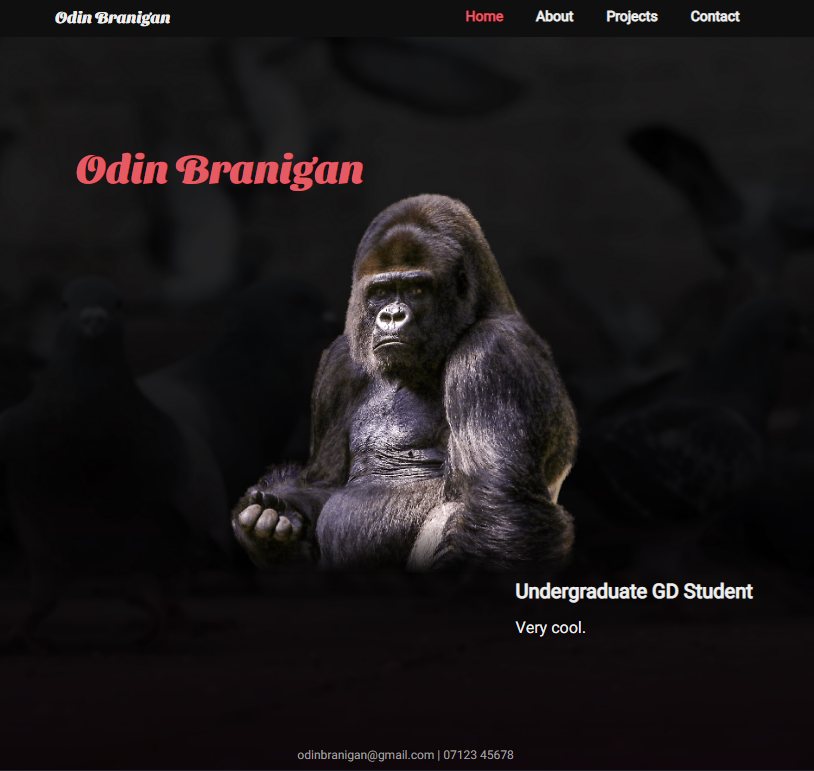
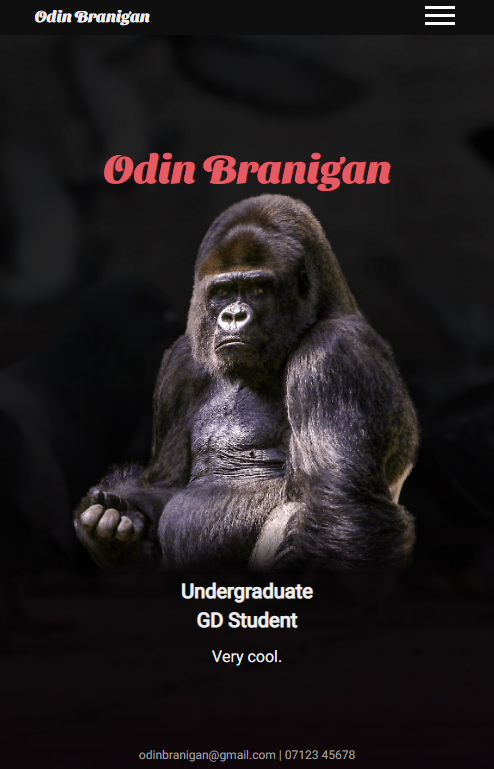
Website Analysis and Report

# Initial Specification

When beginning to plan on what my website will include and how content will be laid out I wanted to make sure to include a flexible and interactive navigation system, and a way to show 3d models which would also be interactive. I also wanted to make sure the website was able to display content correctly at different resolutions as I knew not being aware of this would give me problems later on.

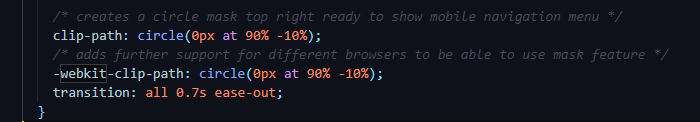
# Analysis

When initially visiting the site, you are greeted by the Home page which involves an interactive navigation bar and several other elements below. The navigation menu scales correctly at different resolutions and will also switch to a burger menu for mobile users allowing for easy and simple navigation for these different resolutions. This also makes sure that the screen does not get too busy and cluttered at smaller sizes.

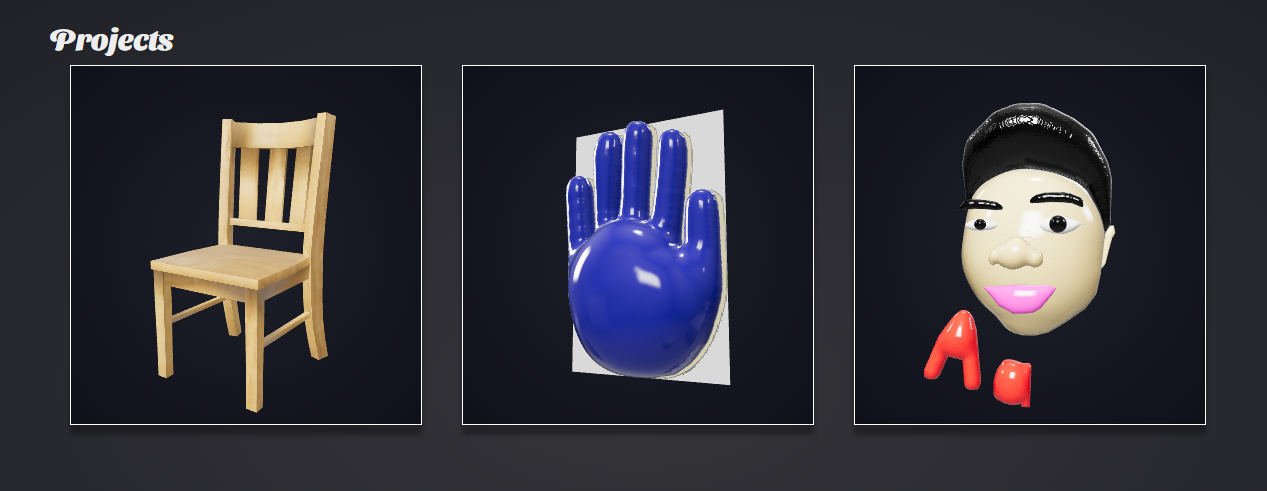
 

With the main content on the home page, I am also able to clearly see the content at different resolutions. By using a grid to position these elements, I was able to easily move them to different grid places depending on the resolution. I also did not want the central image to maintain a minimum size even if the window is very tiny and so I was able to do this with the min-height property, otherwise it would shrink until you could no longer see it. Although it can be good for elements to scale with different resolutions I did not want this one to shrink too much, especially when the surrounding text retains its size.

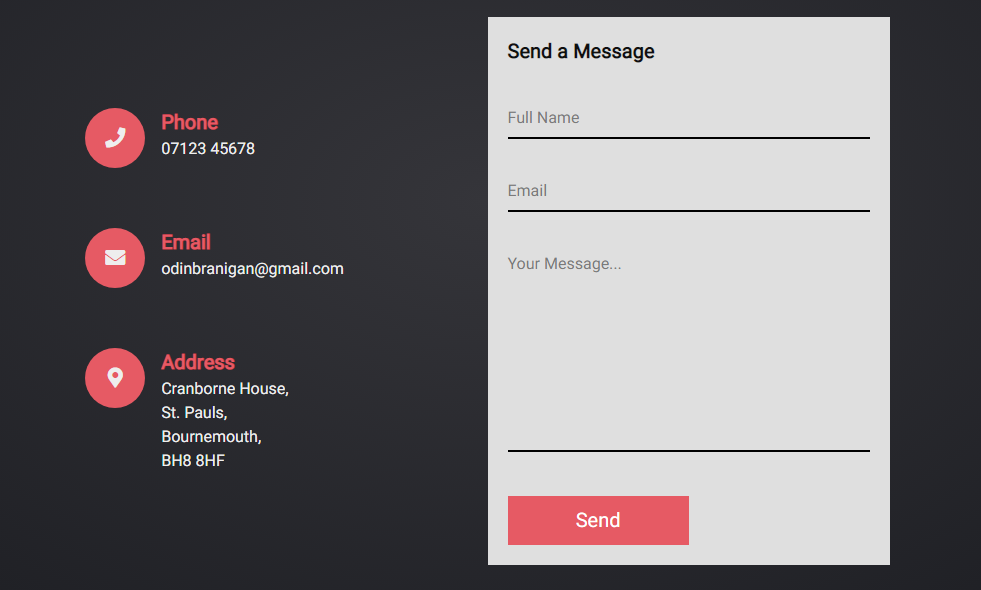
To also make this interesting and start using JavaScript for the first time, the burger menu also includes an animated transition when opening it, making use of a mask to gradually but quickly reveal the navigation links by toggling a class.



Whilst using this mask feature I also included another line to add further support for browsers that use WebKit just in case there were issues displaying this effect on certain browsers. I have made sure to test this on Firefox and Google Chrome and did not encounter any problems.



Moving on to the Projects page, this is where I was able to use JavaScript to display interactive 3D models. I was able to do this by importing JavaScript from *https://modelviewer.dev/* as this is not a built-in feature. This gives a new element, *<model-viewer>,* which I was able to edit however I wanted, and allowed me to add .glb and .glTf files. If I could change anything about this page I would have liked to turn this simple layout into an interactive slideshow instead. This would make it more interesting, and also give larger previews of each model. As well as that, I found on my PC that having more of these model-viewer elements on screen at once quickly made the webpage slow.



Onto the Contact page, this is where I got to use icons and forms for the first time. After researching how to use icons I found FontAwesome.com which provide and allow you to easily add icons directly from their website.

All I needed to do was link this JavaScript which this company had provided to me. By using this method, I do not need to find and save images, making it much more manageable and also easier for in future if I want to continue using icons elsewhere. I also decided to use these as they can be scaled without losing quality as they are vectors, unlike images which can lose quality easily when being scaled to different sizes.

I was also able to make the form more responsive and interesting from a “*Responsive Contact Us Page Design using Html CSS | HTML Responsive Web Page Design”* tutorial by Online Tutorials which introduced me to icons and useful CSS such as “*:focus”*. I have also made sure to include mobile support which was done easily as I had used a flex display for sections of the page and I could simply change the direction of it from row to column. This would put the form below my contact details, be big enough for mobile users, and still keep everything centred and neat.

# Research

FontAwesome.com – icons and JavaScript

*\*:focus {*

*outline: none;*

*}*

CSS provided by CEich at https://stackoverflow.com/questions/3397113/how-to-remove-focus-border-outline-around-text-input-boxes-chrome

Free to use .gbl files found on TurboSquid and