Michael Gannon

Currently, there are countless productivity tools for Developers, to speed up and simplify the creation of code and occasionally they can become a crutch. In my opinion it's more important to develop a comprehensive understanding of the underlying technology before learning to use the tools that simplify the development process. That's the main reason I enjoyed this exercise as much as I did, it tested my comprehensive knowledge without the 'crutch' of Twitter Bootstrap.

The first thing I did before writing a line of code was examine the mockups and decide on how these were going to be implemented in HTML. So I broke the mockups into boxes and containers and decided what would hold what and guesstimated some unit values I would assign and sketched out how the DOM structure would appear with no CSS (attached). I also decided what elements were most appropriate for each bit of the design.

1. I created the container element to hold all the elements and stretches the width and length of the viewport, as specified by the brief.
2. I added a nav bar using the HTML5 tag <nav>, added the image provided, resized the image and added a margin of 14vw. I chose VW and VH as my units because they are relative to the size of the viewport they are being viewed on and making the site responsive would be easier this way.
3. I added the content div below the nav because the nav had to take up the full screen width at all times but the content needed a left and right margin on either side which I didn't want to interfere with the behaviour of the nav.
4. I broke the content into 'left' and 'right' and added all the relevant data in the same layout as the wireframe.
5. The next thing needed was the popup modal window.
   1. I created the modal container and drop shadow as layed out by the brief. I set the display selector to 'none' so the whole modal doesn't appear on page load.
   2. I binded a click event to button#button which changes the display to 'block' on click.
   3. I binded another click event on the x which reverts the modals display value to 'none' and applied the same click handler to the window so while the modal is visible and something other than the modal is clicked, the modal is removed. This works by using event.target. So within the window click handler there's an if statement to check whether the modal shadow has been clicked, and if it has then it dismisses the modal.
6. I got the modal fade in working using the CSS animation selector and CSS keyframes so animate each from 0% to 50% to 100%.
7. Next I created media queries to accomodate different screen sizes. I let content be the judge of when a media query was required. Since the brief specified that the right panel must remain affixed in the same location at all times the overall layout of the page stayed the same. As the screen was reduced in width I couldn't be as generous with the margins or padding or font size. I had some hidden <br> tags in step 1, 2 & 3 between the words; 'transaction' and 'info', 'receipt' and 'info' and between 'make' and 'payment'. The reason for doing this was that as the browser decided to move some of these words down to the next line it broke the progress bar beneath the words. So when the browser did this I made the br tags visible so each of the 3 phrases broke to the next line at the same time thus keeping the progress bar in tact.
8. I also removed the border and text around the euro and pounds images at the mobile screen size since space was not as available.
9. I added a fixed height to the logo in the nav bar to stop it being nudged up and down on screen resize which fixed the problem.

Problems encountered:

Keeping Steps 1-3 left aligned yet justified in the centre and joining the bottom-borders. How I solved it was floating the text left but also added padding to the left. This also joined the bottom borders of each step as per the brief.

Keeping the modal height big enough so it contains all the elements inside. I used media queries to resolve this.

Fade the modal out. Even though the brief specified just a modal fade-in I wanted to fade the modal out also. I tried assigning the modal a new class on 'x' click or 'window' click that fades the modal out. With more time I would have figured this out.

The steps 1-3 bottom border breaking on screen resize. The words from some of the steps would drop down to the next line so I included some <br> tags and set their display to none and when i needed to correct the steps bottom borders I set the display value of the br tags to block ensuring the steps stayed looking uniform

What I would have done differently:

I really like the mockup designs. I would have maybe displayed the left container on top of the right container below 500px width since it ends up looking squashed and reducing the font sizes below 12px makes them hard to read.