

MA Web Design &
Content Planning

REPORT:

IMPLEMENTATION AND PROMOTION

Helen Ng
SEPTEMBER 2017

Contents

I.	Site launch	2
II.	Stages of implementation	2
	Schedule.....	2
	Content collation and creation	2
	CMS development.....	3
	Front-end development.....	3
	Testing.....	4
III.	Problems encountered	5
	SVG illustrations.....	5
	Filter function.....	5
IV.	Changes from initial plan	6
	Design.....	6
	SVGs	6
	CSS.....	6
	Schedule.....	6
V.	Promotion	7
	SEO strategy.....	7
	Social Platforms	7
	Other	7

Site launch

Architecturally launched on the 29 August despite not being a finished product. There were still improvements to be made and illustrations to complete, however, core functionalities were working well and all written content I had originally planned to include was completed.

Stages of implementation

Schedule

For the prototype crit at the beginning of May, I created an overview schedule for implementation. However, after receiving feedback from our tutors who suggested that we should produce more detailed plans for each stage, I revised my overview and assigned each week with tasks I needed to complete. The initial weeks covered mostly content creation, then CMS and front-end development tasks in the following weeks, with time for testing and review factored in towards the end.

Content collation and creation

Due to the nature of the content, I knew that the process of researching, collating, and then creating would take longer to complete. I allowed myself until the end of May to finish the textual side of the content, and spaced out my illustrations over the months. By launch time, I wanted 15 building entries and three architectural styles featured on the site - a little less than I would have liked, although I knew that with the time given, it wasn't worth sacrificing the quality of content over quantity.

Research for written content started before May, but I only managed to gather information for the architectural styles and none of the individual buildings. I actually found writing the content one of the hardest parts of the implementation process as I'm not particularly confident with my style of writing, so I would spend a lot of time rewriting bits of text. Luckily, I managed to complete all the content writing apart from the *About* page by the end of May, and revisited the content every now and again to proofread and edit.

SVG illustrations are the other part of the content, and although I didn't find them as difficult as the writing, the process of creating them was very time-consuming. On average, each illustration took nearly 2 days to complete, and then a further couple of hours for the building previews. Research was needed to find the most suitable images to draw from, and at times, even the simpler buildings took much longer than expected. SVG optimisation was also key, as I knew that the more SVG illustrations there were on the site, the greater impact they would have on page load times. To ensure the SVGs were fully optimised, I manually simplified paths and shapes after creating each one, and then ran them through Jake Archibald's SVGOMG online tool to reduce the file size further. ^[1]

CMS development

For my CMS, I decided to opt for Craft as I felt there was an abundance of documentation and videos out there, in addition to a huge active community on Slack. I blocked out a full week in my schedule to concentrate on familiarising myself with the CMS and getting to grips with the idea of templates and TWIG syntax. During this period, I was still uncertain whether using a CMS was the right choice for my project. However, after a week of playing around, I saw the benefits of using one — especially Craft; creating relationships between content was easily achievable and it gave you full flexibility in creating content your way. With the super user-friendly interface and huge support system from the Craft community, it made my development process a little less daunting.

Before starting any development, I listed the must-haves and nice-to-haves on the site. Amongst the must-haves were a couple of major functionalities which took a little longer than expected to develop: the first being a filter where users could filter building entries by architectural style, century, type, and area; and the second being an interactive map with custom pins where users can click on a pin to activate a pop-up information box which links through to the main entry.

For the filter, I initially came up with a JavaScript solution — hiding and showing different entries depending on which filters are chosen — but when I tried to integrate it into Craft I encountered some difficulties with implementing it into the templates. I approached the Craft Slack channel for help, and one member advised that although the JavaScript solution was a neat way of doing this, it's not a long-term solution and wouldn't work well when the site eventually has hundreds of entries. He suggested going with a back-end filter which performs well with a large volume of entries and also inputs parameters into the URL so that users can always revisit that particular filtered result if they wanted to. I decided to go ahead with this approach and although it was much more difficult to develop, it was finally implemented thanks to many hours searching through the Craft forum and a blog post.^[2]

For the map, I was lucky enough to find exactly what I needed in the form of a plugin called SmartMaps.^[3] This plugin enabled me to use an address field to render a building's location on the front-end and was easily customisable. The plugin was fairly expensive, however, I felt it was worth the cost due to the ease of development and the support that was offered.

Front-end development

Another reason why I chose Craft was due to the complete flexibility of front-end development, I could design it how I wanted without any restrictions and could be confident that my design would look consistent across the pages. As I'd created wireframes and mock-ups for most of the pages on the site for the prototype crit, I had a good idea of how I wanted it to look. The design needed to be as clean and simple as possible for ease of responsive design but also to enhance readability and draw attention to the illustrations.

For the majority of my layouts, I used flexbox to create grids for components on the page; as I'd used it in previous projects, I was comfortable with how to develop with it and implement it. It is also widely supported across browsers which meant that I could be confident in my design rendering the same on multiple browsers without coming across too many problems. For CSS, I followed the BEM naming method and tried to write shorthand

properties wherever possible to keep the CSS clean and optimised. I made sure that comments were written throughout the file to clarify the different sections during development, although the final CSS file currently used is a minified version without any comments or spaces to reduce page load times.

At the beginning of the development process, I wanted to use as little Javascript as possible, and would try to opt for CSS solutions where I could. The original image pop-ups and back-to-top buttons were implemented via CSS, yet after user testing, I realised that JS solutions would be needed to enhance user experience for these two components.

Testing

Browser

During development, I interchanged between Chrome and Firefox, however I created a dev-site where I was able to test the site across all different browsers and devices making sure that the site performed as expected.

User-testing

I carried this out just before launch, and was able to gain some really valuable feedback from some users. I delayed launching the site by a couple of days to implement these changes which I felt had a positive impact on user experience. For example, the layout of the building entry pages were improved to encourage users to read the content and to balance the page visually; the layout of the listings page for the buildings page was also improved to enhance usability of the filter function; and the image pop-up was created for a better visual experience. I plan to carry on testing post-launch to see if I can further improve the site's usability and design.

Accessibility

I used the WAVE Accessibility tool to ensure that WCAG 2.0 standards were met. When I first tested the site, the initial results were better than I expected, I only had a few colour contrast errors and warnings about long alt texts and duplications, which have now been improved. Throughout development, I made sure that all mark-up was semantic and researched into the best accessibility practices. ^[4]

Problems encountered

SVG illustrations

These took much longer than expected as I wanted to include as much detail as possible and when testing on different browsers, IE was particularly problematic as the SVGS were distorted. I eventually found a solution where the SVGS had to be 'Exported As' rather than 'Save As', and the width and height attributes needed to be removed and the 'PreserveAspectRatio' attribute included. The process of having to research and implement the solution, then further optimising the SVGS and correcting the mark-up did contribute to missing the scheduled deadline for finishing the illustrations. Only 11 out of 15 were completed by launch at the end of August, which meant entries without illustrations had to have a placeholder message of 'Image coming soon...' instead.

Filter function

The back-end filter was a challenge and went through many development stages and iterations before it could be finalised. The initial research through the Craft documentation and forum took longer than expected, and then trying to figure out how to implement it within the template using TWIG syntax added on more time. After the first iteration, the filter failed to return any results; then on the second, the filter parameters did not appear in the URL; with the third iteration, the filter failed to function with the .htaccess file in place. A solution was eventually found and the filter functioned as hoped. Although I wanted to revert back to the Javascript solution at times, I'm glad I managed to implement this via the back-end and feel I have learnt a lot along the way.

Changes from initial plan

Design

Layout for some of the pages were altered to improve user experience; for example, the filter is placed as a sidebar on the buildings index page rather than at the top of the page, and building entry pages have been changed to enhance the flow and readability of the content.

SVGs and CSS animations

I initially wanted to inline SVGS for CSS animation purposes, however with the use of a CMS, I found it difficult to implement it this way and decided on including SVGs using the method instead. Choosing this method has not affected how the illustrations behave on the page, and have not lessened the impact of them despite no CSS animations.

CSS

I planned to use SASS to decrease development times via variables and mixins etc. However, I found that the implementation time took much longer than I thought and found myself falling behind schedule. I also realised that the file sizes created using SASS were larger than the original CSS file I'd created. I decided to stick with CSS and use BEM and shorthand properties, which allowed me to develop pages much quicker and be confident in completing front-end development on time.

Schedule

Although I had planned out each stage in detail, I still ended up deviating as I had underestimated how much time researching and trialling out different components would take. Due to errors and components not functioning properly and the time spent on fixing those, I did slip from the scheduled plan. This meant that the site was launched without some illustrations and improvements still to be made, but I needed the site to be live for a period of time to gain some data for analytics purposes.

Promotion

SEO strategy

I used SEOMatic, ^[5] a plugin which allows for easy SEO implementation within Craft and carries out tests to show you where SEO could be improved. This consisted of entering keywords and data for meta title and description tags, setting up OpenGraph meta data and images for social platforms, and making sure that information was provided for site identity purposes. In addition, I ensured that written content on the site was keyword-rich by trying to include the most relevant keywords. I found this fairly difficult as I didn't want to over-use certain keywords and be seen as a spam site, yet had to place certain keywords strategically to make sure that they were included. I have also included as many internal links as possible to improve navigation of the site and ensured that the site meets WCAG 2.0 standards to improve its prospects in Google search rankings.

Social Platforms

During my research for the site, survey results from a Google Form I sent out showed that the most popular social platforms people used were Facebook and Instagram. Before launch, I had created an Instagram account and posted photographs of buildings around London which I'd taken and included relevant tags to reach as many people as possible who were interested in architecture and illustration. Unfortunately, due to lack of time, I was only able to post a few images before launch, so I'm hoping to work on building up Architecturally's Instagram profile after launch and rally some more followers, by posting a mixture of photographic images and illustrations once a day.

Facebook was set up after launch, as initially I wasn't sure whether the site needed a Facebook presence. However, after some consideration I felt that it would help reach a wider audience and have a much better click through rate than Instagram. A post will be published at least once a day based on an individual building, which will include a small quote from the content and a preview of the illustration to encourage people to click through to the site.

The use of Facebook and Instagram will differ in terms of the mediums used; Facebook will be primarily focused on attracting users through snippets of content, whereas Instagram will use imagery to encourage users to click through to the site. I wanted each platform to offer different types of content so that each one could maximise its marketing potential and use different ways to engage users.

Other

Many of my promotional activities have mainly been through word-of-mouth, sending the site to friends and colleagues, and encouraging them to share with others. I am also planning to publish a post on BoredPanda.com which is a platform where people can share their creative work in art, design and photography. As this site gets thousands of views a day, I hope to attract at least a few dozen views and click-throughs to the site by posting about my illustrations.

References

1. Jakearchibald.github.io. (2017). *SVGOMG - SVGO's Missing GUI*. [online] Available at: <https://jakearchibald.github.io/svgomg/>
2. Coupé, J. (2017). *Combined searches and filters using Craft - Webstoemp*. [online] Webstoemp.com. Available at: <https://www.webstoemp.com/blog/combined-searches-and-filters-craft-cms/> [Accessed 11 Jun. 2017].
3. Double Secret Agency. (2017). *Smart Map - The smart way to handle maps & addresses*. [online] Available at: <https://www.doublesecretagency.com/plugins/smart-map>
4. Pickering, H. (2016). *Inclusive design patterns*. 1st ed. Freiburg, Germany: Smashing Magazine.
5. Welch, A. (2017). *nystudio107/seomatic*. [online] GitHub. Available at: <https://github.com/nystudio107/seomatic>