Samantha Grise

Prof. Jen Kramer

DGMDE27 Assignment 3 Comparison and Paper

April 17th 2017

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Custom Hand Coded Responsive Design | UIkit Responsive Design Framework | Winner |
| 1. Speed Of Setup | So long as the front-end developer has a firm grasp on html, css, and some javascript, they can moderately quickly put together a site with limited additional functionality. | The front-end developer needs a short adaptive period to familiarize with the setup and utilization of the framework. Beyond this initial period, the developer needs to implement the desired structure and features on the site. | Custom Hand Coded, with the caveat that the initial site must be very basic. |
| 1. Available Features | Available features are limited to structures and functionality built into html and css, and then additional features must be handcoded. | This is where UIkit shines. UIkit has many core features and components that can be quickly added to a site. | Hands down UIkit |
| 1. Clean CSS & HTML | How clean and well written the site will be is completely determined by the skill and thoroughness of the developer. | With UIkit the existing files and documentation help to set up a trend for clean code. Especially the inclusion of very highly commented files and the overall structure of the framework, the developer is setup for success. | UIkit |
| 1. Sticky Footer | With hand coding, this is very easy to implement, though at times tricky for the beginner. | There is no built in sticky-footer with UIkit, it must be hand-coded. | Hand coded wins |
| 1. Equal Height | With hand coding, the developer must write their own solution to make equal height columns, or integrate a third party solution. | UIkit very easily creates equal heights with data-uk-grid-match . | UIkit wins |
| 1. Popularity | Hand coding is used on all sites, that are not pre-built. | Currently UIkit is one of the most popular frameworks, but is significantly less popular than bootstrap or foundation, meaning a smaller community. | Hand coding wins |
| 1. Documentation | Documentation for a hand-coded site is limited to documentation from HTML, CSS, and other sources. There is no specific documentation for a hand-coded site. | UIkit does have documentation, though it is surprisingly limited. The developers of the framework made strong assumptions about the comfort of user with certain web technologies and more specifically with the utilization of the framework and how it fits together. Other frameworks like Bootstrap have far superior documentation than UIkit. | UIkit |
| 1. Customization | A hand-coded site, is endlessly customizable, the developers skills and desires are the only limitation. | There is a degree of customization built into UIkit and the files are so well documented it is not an issue to utilize. (variables, mixins, hooks) | UIkit |
| 1. LESS vs SASS | With hand coding, it is up to the developer which css preprocessor is used. | The most recent version of UIkit calls for LESS not SASS, which is arguably not as user friendly, but will have SASS later. | Hand coding wins |
| 1. Stability | Each web technology, HTML, CSS…etc goes through versioning, but we can be moderately safe in assuming the longevity of HTML and CSS. | UIkit is still being actively worked on, with its most recent version still in beta, this means that currently it is stable. | Hand coding wins |
| 1. JavaScript Functionality | Hand coded sites must plugin JavaScript or write code that integrates well with the site. | UIkit ships with many JavaScript functionality that automatically integrates with their core and components. | UIkit wins |
| 1. Size | The size of the site/files is dependent on what the developer writes. | When downloading UIkit’s bundle, you get everything. If you want to make your site smaller, you will need to remove erroneous files. | Hand coding wins |
| 1. Modern Web | With hand coding how up to date a site is with the latest standards/trends is entirely dependent on the developer. | UIkit does pretty well semantically, with its core and components, however it fails in the department that all of the styling is done with px. To remedy this, would require lots of editing of the UIkit files. | Hand coding wins |

In today’s modern world of the web we have many choices when it comes to the tools we utilize in creation of websites. There are numerous frameworks, plugins, code snippets, …etc. that can make the development process faster and easier, yet there are times when writing a site by hand is all you really need. In this paper, custom hand coding a site will be compared against UIkit, a responsive design framework.

Firstly, the initial determining factor for whether UIkit or a custom coded site would be ideal, is complexity. Will the site to be built be complex or simple, with minimal functionality? Simplicity meaning limited styling, features, and functionality. Should the site be minimal in design needs then perhaps hand coding the site would be more efficient. UIkit requires a degree of setup and implementation that could be more cost ineffective, than simply hand coding a simple site. However, setup of UIkit is not so complex, only requiring an adaptive period for the developer to familiarize themselves with the framework, that if the site in question is of a more complex nature, then the setup and implementation needed to get UIkit up and running is worth it, considering all of the time that would be saved hand coding added features and functionality.

UIkit truly beats hand coding in multiple key areas: features, functionality, customization, and documentation. With hand coding, all of the features and functionality desired must be written, or third party code must be found, vetted, and implemented without issue. UIkit is very successful, in that when added to a site, it comes with all of its core, components, and JavaScript functionality (additionally jQuery does need to be linked to), and the developer only needs to connect the files and add the code to their files.

UIkit comes with many useful features and functionality such as a pre written grid system, navigation, slideshow, dynamic grid, …etc. Hand coding is limited to the skillset of the developer and/or their ability to implement other code. Therefore UIkit allows for the quick implementation of features and functionality that would take significantly longer when hand coded. Likewise, these pieces all integrate and play well with each other out of the box.

The customization of these added pieces from UIkit is not difficult due to special css classes, JavaScript classes, and the clean format and exceptional commenting of the files in the UIkit bundle. In the SASS files there are variables, mixins, and hooks which make customizing UIkit very simple. Likewise the UIkit has documentation on each and every one of these additional pieces.

There are some areas where UIkit is not as successful: it does not have every feature that a developer could want, has moderate size, has moderate long-term stability, and has moderate popularity. One of the features UIkit lacks is a built in sticky footer. However it has so many other features, like equal heights, that this missing piece can be easily forgiven and just hand coded. UIkit also is very attached to px, whereas the rest of the development community has been moving towards ems, this is an issue that is harder to remedy since fixing all of the px in the UIkit files would be tedious at best. The use of px makes UIkit less responsive, which is a clear negative and creates a greater need for more code in the form of media queries.

The size of UIkit is also problematic, there is no easy way to download and implement just some of UIkit especially if using a cdn this means larger sites. Also the stability and popularity of UIkit is questionable. UIkit currently is doing very well. It is in active development and is moderately popular, but it is not nearly as popular as Bootstrap or Foundation, this means that there is a small community of developers using and contributing to UIkit. UIkit’s preference for LESS over SASS, the currently more popular css preprocessor, likely also does not help with UIkit’s popularity. Longevity of web technologies is often dependent on popularity, so the staying power of UIkit feels less stable than say Bootstrap, and obviously neither framework can be as stable as simple hand coding.

In short, hand coding will likely be around forever, but a framework like UIkit is so useful and powerful that hand coding with a framework is the clear winner of this battle. Utilizing UIkit, an often overlooked but powerful framework, may help the web community keep it around for a long time to come, letting us make our sites faster and more functional.