CSS Framework or Custom CSS design

Today, creating modern a website there is a focus on responsive design, meaning the website will typically include media queries to provide responsiveness, a grid system for layout, and images that are sized appropriately for the screen. So, what is the best approach to creating a new responsive website, using a CSS framework or writing custom CSS? In this article, I will compare a CSS framework, uikit, vs. custom CSS looking at specific components used in typical websites.

# Comparison Grid

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **CSS Framework (uikit)** | **Custom CSS/JS** | **Which is better?** |
| Handling responsive design across multiple devices with varied screen sizes | uikit includes six predefined media queries for breakpoints based on possible devices screen size | Must determine, through trial and error, appropriate breakpoints to create as many media queries as required | Since uikit provides a starting point for breakpoints and the breakpoints can be adjusted for the content using a framework is better and more time efficient |
| Adaptation of content for responsive design using a grid system | uikit provides several grid-based solutions, including a standard grid system and the ability to adapt the standard grid system with a dynamic grid or a parallax grid. uikit’s grid system provides adjustability of all grid components including gutters, source ordering, and wrapping | A grid only system could be download, but it may conflict with other components or a custom grid system could be written. Writing a custom grid system would require knowledge of how the grid elements will interact on a page and require tweaking to make the grid look correct on all devices | A predefined grid system is a better option as implementation is straight forward and you can adapt the grid system to fit a specific project. Also, unique features, such as parallax or even columns do not require writing any JavaScript |
| Handling images in a responsive design with JavaScript | uikit does not include any built functionality for handling responsive images | Must download a JavaScript solution or custom write a solution. An alternative is using HTML, but you do not have the ability to customize the HTML code | A downloaded solution would be the best alternative when working with responsive images. The package would typically take into consideration any cross-browser issues and JavaScript based solutions are typically encapsulated from interfering with existing code |
| Providing navigation that works across multiple sized devices | uikit includes a navbar that is partially responsive, in that it will shrink as the screen size gets smaller, but eventually the links begin to wrap. uikit includes the ability to create an offcanvase navigation that can incorporated for smaller screen sizes. uikit does not include a dropdown menu for smaller screens | Navbars can be investigated and downloaded or a custom written navbar can be generated. Again, a downloaded navbar can include components that cause conflicts with exist code in a project. |  |
| Handling form layout and interactive form components | uikit has an extensive list of form helper. From the core form to the numerous components that can enhance the core form. The uikit form makes controlling the layout and components of a form simple using markup similar to standard HTML form | Custom designing a form |  |