**“Standards and Best Practices for Modern Web Development”**

Visiting a website is simple and straightforward; although creating a website has a certain quality or standard that we should practice, it is necessary to learn these standards and practices for the betterment and understandability customers or users.

**Best Practices:**

1. **Understand the big picture.** Some developers love to code so much that they skip the first step — planning — and end up writing code over and over again. First ask, what’s the overall intention, or bigger picture of the project? This leads to understanding priorities and not getting bogged in the insignificant details that may steer you off course.
2. **Start with the user experience.** Paper or a mock-up are cheap to change compared to the cost of scrapping a project your users don’t want to use. A good rule of thumb to remember is that 70 percent of development efforts should be spent on research and design, and then the final 30 percent on programming. By understanding the potential user, and creating a user-friendly design, the development phase will significantly less complex!
3. **Write smarter, not more.**Don’t make something too clever. Every line of code you write should have a definite and necessary purpose. Overproducing code is the figurative equivalent to asking for bugs. Each piece of code should solve a problem or provide a feature necessary to the function of the page.
4. **Don’t reinvent the wheel.** Imagine you’re building a house. Would you rather start with a bare piece of land, or a nearly constructed house where all you can focus on laying carpet, painting and installing appliances and landscaping? In application development, a framework provides a structure in which you can build on instead of starting from scratch. Not only does it save time, but it helps less experienced developers build better quality applications and reduces potential for errors.
5. **Be nimble to avoid pitfalls.** Lean, Agile, Scrum, XP — you pick. Don’t let a huge feature hold your timeline hostage and aim to be “always shippable.” In other words, instead of hitting a grand-slam home run, swing for hundreds of singles and doubles. Development is a process of continuous improvement, and incremental updates to a working project creates less bugs than waiting until it’s built out in full.
6. **Consider the future — but not too much.** Don’t over-engineer for some theoretical future need, but conversely don’t let today’s demands close off any future innovation. It’s inevitable that the user interface (UI) you design today will be outdated a few years, or even a few months from now. Design and develop the backend in a way that you can adjust the UI without having to do a full-fledged overhaul, which can be extremely costly. Also expect to make incremental updates, whether it be slight designs to font to adding a completely new feature.
7. **Write testable code.**Testing is an essential tool in the toolbox of any serious developer. However, some code is written in a way that it’s hard, or even impossible, to test. Make sure to follow-up testing best practices and methods to write clean, easy-to-read code.
8. **Decide what you’re optimizing for.** This depends on the project, of course. Is this code intended to last a while, or is it just a quick prototype?  If the former, then the answer is probably not that we are optimizing for highest performance or utmost reliability (yes, we shouldn’t crash, but we don’t have to spend months proving that it’s resilient to the ridiculously improbable). Instead, optimize for code readability and extensibility. The next person that works on your code needs to understand it even if you have forgotten — and that might be you, coming back to it months later.

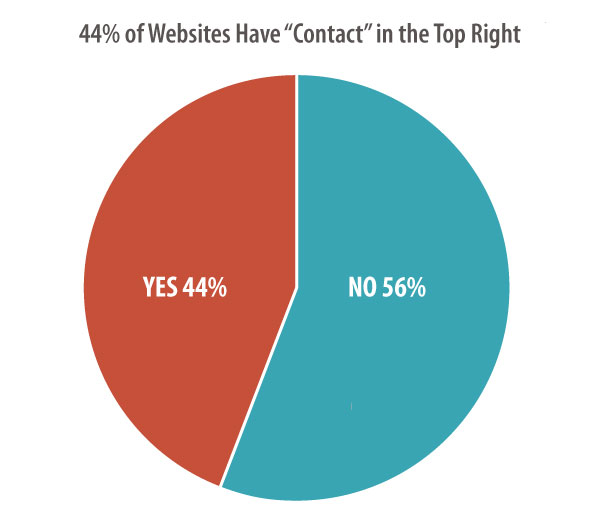
### Best Practices for Designing the Website:

### 1. Logo in the top left

100% of the websites researched had a clickable logo in the upper left corner of every page on the site. That’s a standard!

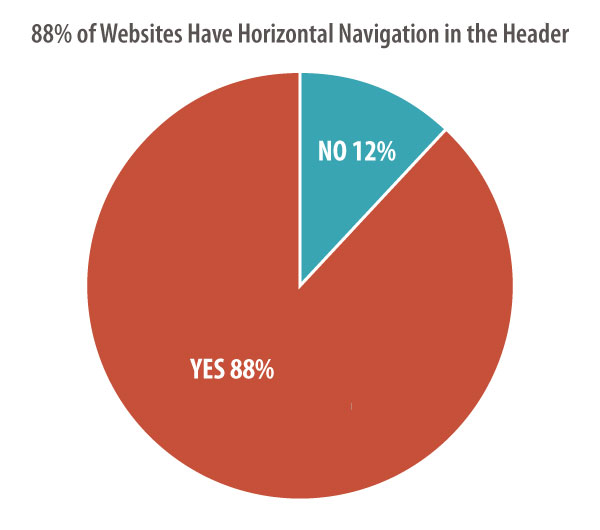
### 2. Contact in top right

44% have the contact button or link in the top right corner of every page. Although this placement is very common and considered best practices, it cannot be considered standard.



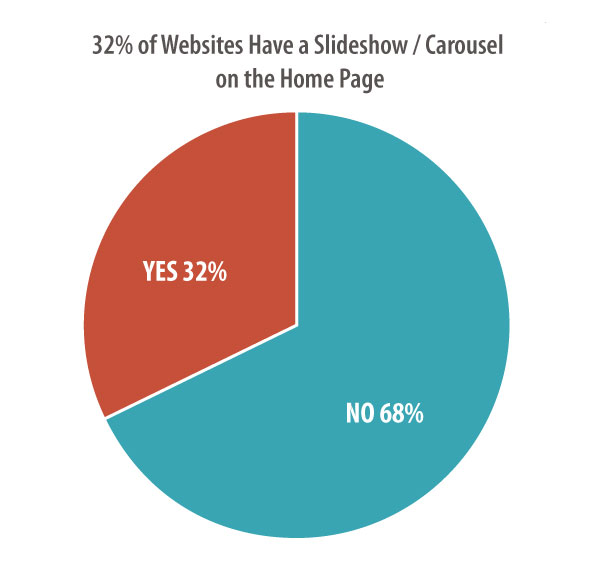
### 3. Main navigation across the top

88% of the websites had the main navigation located in the header at the top of every page, making horizontal top-level navigation a web design standard.



### 4. Home page slideshow

32% of the websites have a home page slideshow (also known as a carousel) with a rotating series of images and messages.

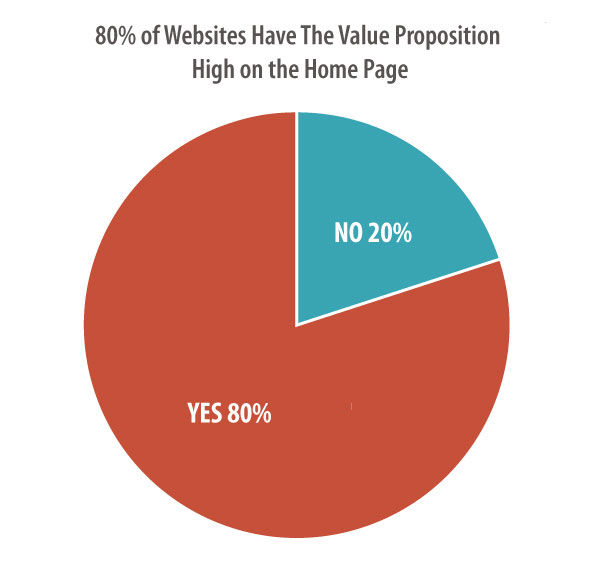


This is one that Orbit designers are watching carefully, as more sites seem to be favoring a static featured image, rather than a rotating series of images.

Research [is mixed](http://www.nngroup.com/articles/auto-forwarding/) on the effectiveness of each option. Results vary! Choose the best option for your site, your message and your visitors.

### 5. Value proposition high up on the home page

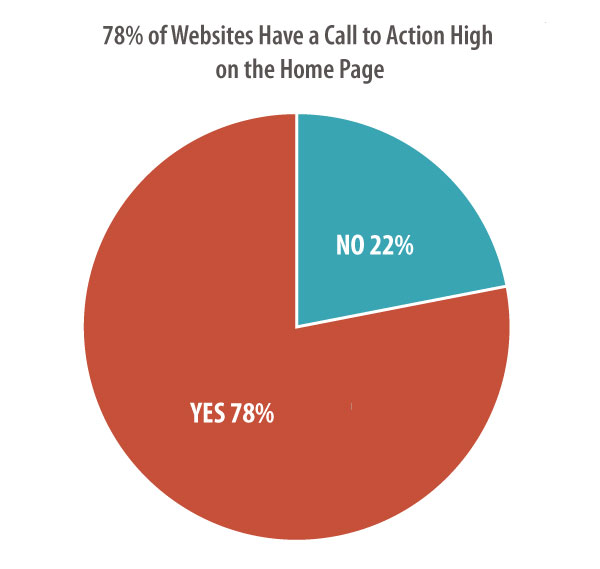
80% of marketing websites have an explicit value proposition located high on the home page. So the majority of websites explain their value to visitors “above the fold.” The remaining didn’t have an clear value proposition at all.



Any web designer will tell you that there is no standard pixel height for browsers. Therefore, there is no fold. But of course, some design elements appear high on pages and are generally visible to the majority of visitors without scrolling.

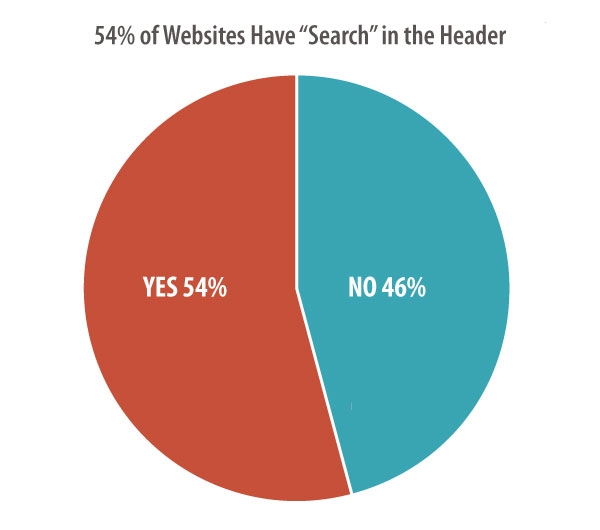
### 6. Call to Action high up on the home page

78% of the websites had visually prominent calls to action. The percentage fell below our threshold for standard, it’s certainly a convention.



### 7. Search feature in the header

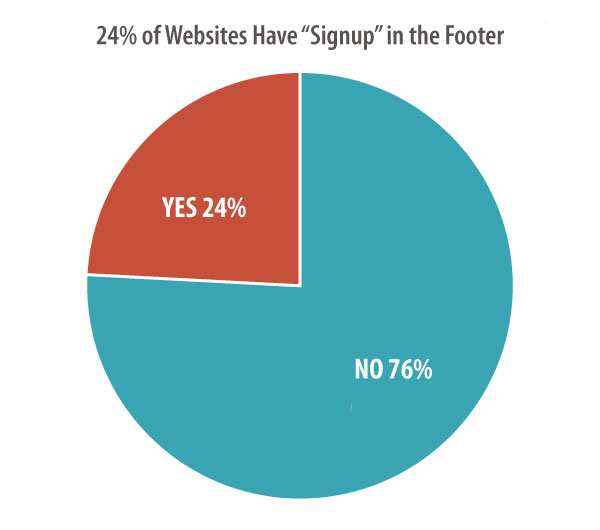
54% of websites have a search feature in the header. About half of all marketing sites do not have a search feature that appears “globally” on every page either as a link, icon or search box.



This isn’t surprising to us. Search tools aren’t necessary unless the website contains a large amount of content. A search tool is often a “crutch” for a poorly organized website.

### 8. Signup box in the footer

24% of websites allow visitors to sign up and subscribe to email updates in the footer. So this is a common place to gather email addresses, but not a convention or a standard.

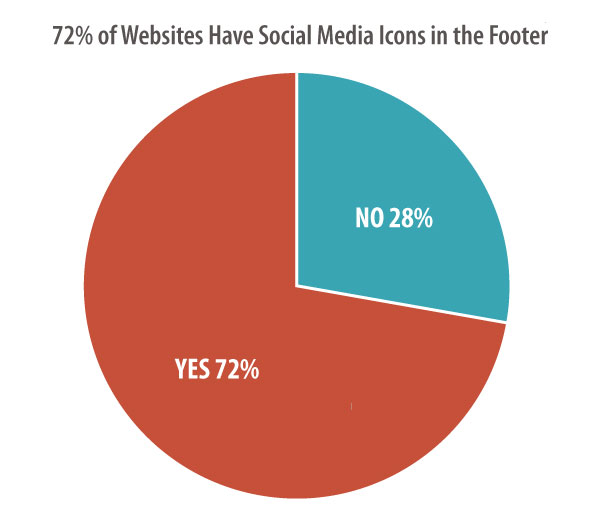


The most common content for footers is copyright, privacy, legal, sitemap and contact links. Visitors expect to find contact information in the bottom right or bottom center of websites.

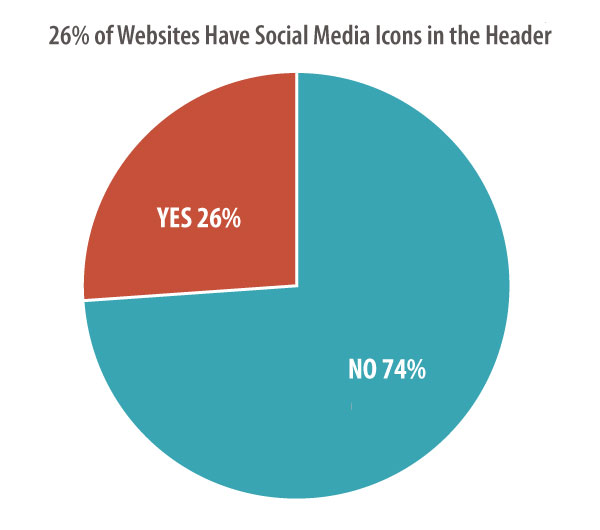
Want to a better footer? Here are our [Footer Design Best Practices](https://www.orbitmedia.com/blog/website-footer-design-best-practices/), and 27 things you can add to the bottom of your pages.

### 9. Social media icons in the footer

72% of the websites include icons for social media websites in the footer. This almost makes these a standard design element.



26% of the websites included social media icons prominently in the header.

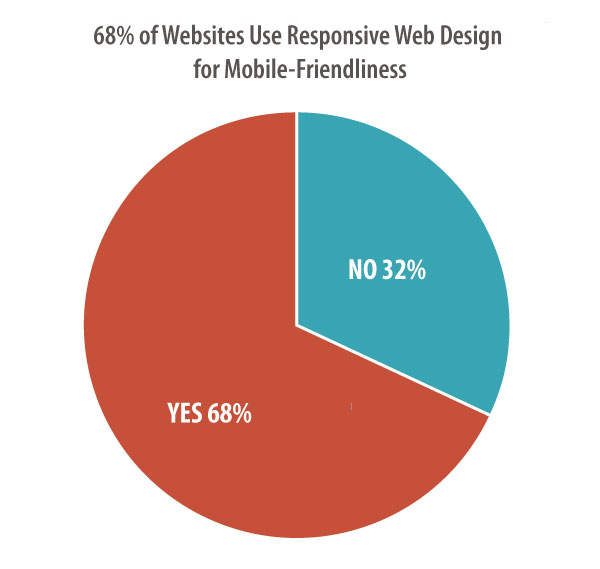


As in the footer, clicking any of these icons takes the visitor to the social media site. For this reason, this is a design element that can [cost you traffic](https://www.orbitmedia.com/blog/website-social-media-integration/), increasing bounce rates and hurt results.

We recommend adding social media icons in the footer. To further reduce visual prominence, the full-color version can appear only after the visitor moves the mouse cursor over the icon.

### 10. Responsive design

68% of websites are mobile-friendly using [responsive web design](https://www.orbitmedia.com/responsive-web-design). This gives visitors a great experience regardless of the device – phone, tablet or desktop.



It’s a combination of design and programming that is difficult to add after a site is built. More often, it’s part of a redesign, which may explain why it’s a convention, but not a standard.

Responsive design has been best practices for years. We’re glad to see this become more common and we expect this feature to be standard eventually.

### Best Practices for Designing the Website (part 2):

### https://www.snapagency.com/web-design-usability#temp

1. Page Layouts
   1. [Template Consistency](https://www.snapagency.com/web-design-usability#temp)
   2. [Visual Hierarchy](https://www.snapagency.com/web-design-usability#hier)
   3. [Call to Action](https://www.snapagency.com/web-design-usability#cta)
   4. [Cross-Market](https://www.snapagency.com/web-design-usability#cross)
   5. [Page Width](https://www.snapagency.com/web-design-usability#width)
2. Navigations
   1. [Logo Link](https://www.snapagency.com/web-design-usability#logo)
   2. [Primary Navigation](https://www.snapagency.com/web-design-usability#primary)
   3. [Secondary Navigation](https://www.snapagency.com/web-design-usability#secondary) [Tertiary Navigation](https://www.snapagency.com/web-design-usability#tertiary)
3. Forms
   1. [Number of Fields](https://www.snapagency.com/web-design-usability#fields)
   2. [Requirements](https://www.snapagency.com/web-design-usability#require)
   3. [Method](https://www.snapagency.com/web-design-usability#method)
   4. [Design](https://www.snapagency.com/web-design-usability#design) [Opt-out](https://www.snapagency.com/web-design-usability#opt) [Automatic Error Checking](https://www.snapagency.com/web-design-usability#check)
4. Homepage
   1. [Flexible Media Area](https://www.snapagency.com/web-design-usability#fma)
   2. [Above & Below the Fold](https://www.snapagency.com/web-design-usability#above)
   3. [Navigation Paths](https://www.snapagency.com/web-design-usability#paths)
5. Contrasts and Colors
   1. [Color Palette](https://www.snapagency.com/web-design-usability#palette)
   2. [Emphasis Color](https://www.snapagency.com/web-design-usability#emphasis)
   3. [Color Connotations](https://www.snapagency.com/web-design-usability#connotations)
6. Scrolling
   1. [Page Length](https://www.snapagency.com/web-design-usability#length)
   2. [Infinite Scrolling](https://www.snapagency.com/web-design-usability#infinite)
7. Search
   1. [Why Bother?](https://www.snapagency.com/web-design-usability#bother)
   2. [Placement](https://www.snapagency.com/web-design-usability#placement)
   3. [Search Method & Usability](https://www.snapagency.com/web-design-usability#searchmethod)
8. Readability and Content
   1. [Text Formatting](https://www.snapagency.com/web-design-usability#format)
   2. [Content Layout](https://www.snapagency.com/web-design-usability#layout)
   3. [Images & Videos](https://www.snapagency.com/web-design-usability#images)
9. Mobile
   1. [Responsive Website](https://www.snapagency.com/web-design-usability#responsive)
   2. [Usability](https://www.snapagency.com/web-design-usability#usability)
   3. [Page Speed](https://www.snapagency.com/web-design-usability#speed)
   4. [Consider an App](https://www.snapagency.com/web-design-usability#app)
10. Users and Targeted Audience
    1. [Customer Target Profile](https://www.snapagency.com/web-design-usability#customer)
    2. [Location](https://www.snapagency.com/web-design-usability#location)
    3. [Gender](https://www.snapagency.com/web-design-usability#gender)
    4. [Age](https://www.snapagency.com/web-design-usability#age)
    5. [Income](https://www.snapagency.com/web-design-usability#income)
    6. [Targets Digital Location](https://www.snapagency.com/web-design-usability#targets)
    7. [Social Media](https://www.snapagency.com/web-design-usability#social)
    8. [Market to Your Ideal Customer](https://www.snapagency.com/web-design-usability#market)
    9. [Checkout & Decision Fatigue](https://www.snapagency.com/web-design-usability#fatigue)

**Standards:**

* **Brand Standards** Colours, type and tone are specific to every business. You should have a style guide for your website and stick to it.
* **Efficient Code** web standards brings efficient coding to web design. If all programming was limited to HTML, the result would be bloated and ungainly files that require longer than necessary download times. This not only affects the user's online experience, but it can actually increase the cost of accessing online content for those users who pay for downloads by the megabyte.
* **Coding Compatibility** whether it's a web page, an add-on, or a software application, almost every piece of programming passes through the hands of many different designers. Having an accepted standard of coding ensures that all programmers are speaking the same "language," and that code remains streamlined and efficient regardless of the author.
* **Simplified Maintenance** this further supports the value of standardized coding practices. Lean and efficient code is easier, and less expensive, to update and maintain. With style and layout information specified in one place (ie CSS elements) updates do not need to be applied to every page of an active site.
* **Device Compatibility** the adherence to web standards ensures that online content can be quickly and efficiently reformatted for different browsers and devices. This is particularly important now that mobile devices have become so popular with online users.
* **Search Engine Compatibility** all web developers want their online content to be easily searchable by Google and other search engines. In order for any search engine to rank and return relevant content in a user search it must scan that content with a web crawler. If that web crawler cannot interpret the content, rankings will naturally suffer. By following the approved standards set down by the W3C, developers can more easily ensure that the content they produce can be quickly and accurately scanned by the search engine bots.
* **Accessibility** finally, web standards also addresses the issue of accessibility for the visually impaired. Users with visual impairments often use screen readers to access content on the web. These readers literally read the web page aloud. Adherence to the accepted web standards governing semantic structure ensures that can easily decipher online content, keeping the web open and accessible to the visually impaired. This is one of the few aspects of the web standards project that has been codified into law.

Web Design Standards: 10 Best Practices on the Top 50 Websites. (2018, January 11). Retrieved March 19, 2018, from <https://www.orbitmedia.com/blog/web-design-standards/>

Introduction to Web Standards. (2018, February 13). Retrieved March 19, 2018, from <https://www.whoishostingthis.com/resources/web-standards/>

Bouwkamp, K. (2015, September 23). 8 Web Development Best Practices To Avoid Derailing Your Project. Retrieved March 19, 2018, from <http://www.codingdojo.com/blog/web-development-best-practices/>