

Usability in Web Design

Prepared for the Open Source Development and Documentation Project

John MacKenzie, Andrew McAlister, Sahil Desai, Kevin McCarthy

Executive Summary

Of all the elements of web design, usability requires a high sense of design expertise. The importance of usability cannot be articulated enough. Page layout, speed and accessibility are three of the most vital constituents of usability. These three portions weigh in to a well designed web site. If constructed correctly these concepts can save time and maximize ease of use among those who make use of the site. As a result, this minimizes both the cost for the designer and animosity amidst users.

Introduction

In the not so distant past, web design began evolving into more of a technique and moved away from being an art. With this advent of technological savvy, came the important issue of which part of a website matters the most. Jakob Nielsen, who is considered the grandfather of usability, provides a simple and clear answer, "users experience the usability of a site before they have committed to using it and before they have spent any money on potential purchases" (New Riders Publishing). Usability is not only the key to good web design but also the determining factor in whether or not a user will decide to continue to use a site.

Before continuing, it would be useful to note the following technologies: CSS (Cascading Style Sheets) -- this can be thought of as a template that influences the look and basic outline of each page in a site (The World Wide Web Consortium), Javascript -- a new programming language mostly utilized in web programming, that gives a web page the abilities and advantages of a regular application that one would use offline on a computer (Ginda), XML (Extensible Markup Language) -- This is in its most basic form, the backbones of a website, it is the markup language that determines how all the text, images and content on the site is displayed, and PHP (PHP Hypertext Preprocessor) -- this is a scripting language; it allows a web designer to perform simple functions that process different pieces of information inputted by a user (Wikipedia).

The layout of a page has the ability to impress a user more than the other aspects of a site, simply because it is the first aspect of the site that they encounter. Furthermore, it will determine whether or not a user is comfortable using a site.

Another influential characteristic of a user's experience is how fast each section of the site responds. The many technologies mentioned above, including the non-graphics oriented PHP, Javascript and CSS all assist in speeding up individual parts of the site.

The last of these design elements is accessibility. Accessibility refers not only to the portion of the population with physical disabilities (auditory, visual, etc), but to those people who have a slower internet connection or limited experience with the internet.

Page Layout

The goal of a web page is usually one of the following:

1. Convey some useful information
2. Allow someone to purchase something
3. Allow someone to accomplish something (Web Applications)

All of these tasks require careful interaction and communication with the user. If a site is conveying information, the text can be as clearly written and easy to understand as possible, but if it's buried behind a confusing interface or not properly emphasized on the page, it won't be understood – or worse – ignored. The user's ability to get to the information they want, make their purchase, or use the application, is called usability.

Don't Make Me Think

Steve Krug, a usability expert and the author of one of the foremost books on usability, says: "People often ask me 'What's the most important thing I should do if I want to make sure my web site is easy to use?' The answer is simple... It's don't make me think." Web pages, according to Krug, should be self-explanatory. The user should be able to glance at the page and understand how to do what he wants to do. Every second he stares at the page confused is another second he is tempted to click the back button and head to greener pastures. Emphasizing features most users are looking for, such as search boxes for example, and de-emphasizing less important features, like copyright notices or support information (Krug).

Be Consistent

One great piece of advice to make a site more usable is to be consistent. To make a site immediately self-evident to users, don't reinvent the wheel. Site creators can benefit from the user's experience with other web sites – copy them! If a web designer invents their own interface - no matter how intuitive and cool it is – their users will have to think about how to use it, since they haven't seen it before. And that becomes confusing. Experts in the field agree that "Consistency is critical" when designing sites to be more usable (Gaffney). CSS is the perfect technology to make a design consistent. By allowing someone to create one file to describe what they want their site to look like, CSS simplifies site creation. Since the site's creator can put all of their formatting and design code in one file, they can tell all their pages to look to that file to know how to display its information. This gives a consistent look and feel to the entire site. The designer doesn't have to duplicate information across multiple pages, yet the layout will be the same on every page.

Be Concise

It's always a good idea to cut down on the number of words on a page. Reading takes time, and users generally don't want to spend the time to wade through pages of text. Less text means a smaller page, so users don't have to scroll. It reduces the amount of material on the page, so users don't have to search the page for the information they are looking for. What kind of content should be cut? If a site is an online newspaper or webzine, the designer doesn't need to cut back on their actual content- their users are there to read it. Krug recommends cutting back on "Happy Talk" and Instructions (Krug 46). "Happy Talk" is introductory text. "Welcome to our site! I hope you enjoy it here. Please click on the menu on the left of the screen to go to some pages." Users aren't there for that – imagine putting a DVD in and having someone come on and say, "Welcome to this DVD! We really hope you enjoy watching this movie. You will be seeing a menu that will let you select different options, including starting the movie! Again, thank you for inserting this DVD into your player."

Speed

One of the major complaints that users have with larger sites is the speed at which pages load. Even as the speeds at which users' connect to the internet are increasing, web sites are getting bigger at this same rate. One major source of the increased size is the graphical layout. In the past web designers would create a graphic layout for a site, and then for each page in that site would simply duplicate the layout initially created. This had been successful until the layout of pages on the internet began to get very graphics intensive. This resulted in a slowdown of page loading (Graham).

A disadvantage of these old approaches was that even if one small detail of a page changed, the user had to wait for the entire page to reload. CSS solved this very issue. By having commonly occurring images and simple layout characteristics imprinted in the CSS file, there was less of a need to include this information in every page of the site. Instead a user would only have to load the first page once and then would not have to wait for the computer to retrieve the images and data more than once (Moss).

While developing a web application that uses languages such as PHP, one can encounter a few obstacles in attempting to speed up the application. Since all data must be sent to and received from the server the user might end up staring at a blank page for longer than needed. Possible reasons for this are server connectivity issues or the server is simply unreliable (Wikipedia). In contrast to this server-side approach to web design a more reliable and faster solution is that of client-side web applications. These include Javascript and Flash which are faster because all they require of the server is to deliver the application initially to the user and then the user's computer does the rest (Ginda).

"Ajax, or Asynchronous JavaScript and XML, is an approach to web application development that uses client-side scripting to exchange data with the Web server. As a result, web pages are dynamically updated without a full page refresh interrupting the interaction flow. With Ajax, you can create richer, more dynamic web application user interfaces that approach the immediacy and usability of native desktop applications." (McCarthy)

As McCarthy made clear, these new AJAX web applications are allowing developers to expand their horizons by creating faster, more usable environments in websites. Reiterating an earlier point, the key objective in AJAX is to give the end user an experience that mimics the speed of an offline application.

Accessibility

A different, but related issue to usability is Accessibility – many users are disabled or handicapped, whether it is because of an older web browser that does not support newer web sites, or by an actual physical handicap (e.g. blindness). Blind people can navigate the web by using browsers that have the ability to read text on the web aloud. However, these browsers cannot read back data that is inside images, and often get confused by complex layouts with extensive formatting (Wikipedia). There are a few technologies that can be utilized to help these users . One is to use CSS, by separating the content of the page from the formatting, the screen-readers have an easier time with the page. Older browsers that don't understand CSS can still display the data on the page, although they cannot display the formatting.

Accessibility can be tricky when developing a web application. When using the server side approach (PHP/ASP), the web application is limited to the abilities of the browser in which it will be run. This can sometimes lead to clunky awkward interfaces to which the users is forced to adapt. However, when using the client-side approach (Java/Flash), the programmer has the ability to create much more

interactive and “native” interfaces for the user to interact with. Such advantages would include not having to open a new web page everytime one needs to view new information, mimicking a desktop application (Garret).

Trenton Moss, in explaining why accessibility makes business sense, makes two major points: “An accessible Website will make your client money” and “An accessible Website will save your client money” (Moss). He mentions CSS as the key factor to how this happens:

"An accessible website separates the content (HTML) and presentation (CSS) of each page ... Now, to adjust the layout of the website, developers only need to make changes to the CSS file, rather than amending each and every page of the site, saving considerable time (and, therefore, money)" (Moss)."

When the most important consideration is cost, it is imperative that CSS be used to make designing easier and faster. As a result the website will receive a higher number of visitors which in turn will help the web developer money in the long run (Webcredible).

Conclusion

Usability has become increasingly important as the internet becomes a part of our daily lives. Not only will a usable site make users happy, the converse is true: a difficult-to-use site will drive users away. When creating a site, it is crucial to keep the users' best interest in mind. Look at the site from their perspective, and try to think like a user. A page should be concise, consistent, navigation should be self-evident (Users should not think more than they need to). Test the site on the server to make sure it is not too slow, and if it is, consider using client-side technology or AJAX to speed the site up to improve the user's experience. Think about accessibility - which browser is being used? It is also important to keep the interests of disabled users in mind.

Bibliography

Krug, Steve. *Don't Make Me Think: A Common Sense Approach to Web Usability*. New York: New Riders Press, 2000.

Gaffney, Gerry. “Why Consistency is Critical.” February 25, 2005.
<http://www.sitepoint.com/article/why-consistency-is-critical/> (September 29, 2005).

Moss, Trenton. “Secret Benefits of Search Engine Optimization: Increased Usability”, November 15, 2005. <http://www.sitepoint.com/article/increased-usability/> (September 29, 2005).

New Riders Publishing. “Designing Web Usability” May 4, 2001.
<http://wdvl.internet.com/Authoring/Design/Usability/> (September 29, 2005).

Masse, Roger E.. “Guide to Usability for Software Engineers” November 17, 1998.
<http://www.otal.umd.edu/guse/> (September 30, 2005).

The World Wide Web Consortium. “W3C” September 30, 2005. <http://www.w3c.com> (September 30, 2005).

Neilson, Jakob. “useit.com” September 30, 2005. <http://www.useit.com> (September 30, 2005).

Mullet, Kevin and Darrel Sano. *Designing Visual Interfaces*. Mountain View: Prentice Hall, 1995.

Graham, Paul. "The Other Road Ahead." September 2001. <http://www.paulgraham.com/road.html> (September 29, 2005).

Wikipedia. "Web Applications." September 26, 2005. http://en.wikipedia.org/wiki/Web_application (September 29, 2005).

Garrett, Jesse James, "AJAX: A New Approach to Web Applications", <http://www.adaptivepath.com/publications/essays/archives/000385.php> (September 30, 2005).

McCarthy, P. (2005) "Ajax for Developers: Build dynamic Java applications." <http://www-128.ibm.com/developerworks/library/j-ajax1/?ca=dgr-lnxw01Ajax> (September 26, 2005).

Ginda, R. (2005) JavaScript. <http://www.mozilla.org/js/> (September 27, 2005).

Adaptive Path, LLC. (2005) Adaptive Path. <http://www.adaptivepath.com/> (October 7, 2005).

Usability. (2005) Usability.gov. <http://usability.gov/pdfs/chapter15.pdf> (October 7, 2005).

Webcredible. (2004) Why a CSS website layout will make you money. <http://www.webcredible.co.uk/user-friendly-resources/css/css-website-layout.shtml> (October 23, 2005).

This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/2.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.