# The importance of sketching

**Author – Professor** [**James Williamson**](http://www.lynda.com/James-Williamson/128-1.html)

When learning web design, it's natural to want to focus on technical skills like learning languages and writing code. However, that means that designers often overlook some of the low tech skills that can be just as important to becoming an effective web designer. In this movie, I want to focus on sketching and why it's so valuable to web designers. Creating websites isn't just about writing code and mastering languages. A site's content strategy must be planned. Site architecture should be mapped out, and user experience accounted for.

If you don't have a good planning process in place, it doesn't matter how good your code writing skills are. Your sites simply won't be effective. For me, that's where sketching comes in. Sketching can be one of the most efficient and effective planning tools in a web designer's arsenal. Using a program like Photoshop or creating HTML mockups can be time consuming and constraining. Sketching, on the other hand, is loose and free. You can express ideas freely, iterate through them quickly and explore concepts without the fear of making costly mistakes.

Sketching is extremely low cost, and unlike using Photoshop or coding mock-ups, anybody can do it. That makes it a fantastic tool for initial planning stages and collaborative workflows. Sketching is especially beneficial during the early stages of initial planning. You can sketch out dozens of ideas without really worrying about whether or not they're any good or whether they're going to be effective. This simple process will spur creativity and allow you to come up with unexpected solutions. Of course, sketching isn't just for initial planning.

I keep a sketchbook handy throughout the development process, as it's an easy way to test ideas or spur on inspiration. My favorite benefit of sketching is the way it allows clients or coworkers to focus on ideas instead of design decisions. If you create a mock-up in Photoshop, or put together an HTML prototype, people tend to focus on things like typography, color choices, or placeholder images instead of the idea that you're trying to communicate.They can't help it. I mean, it's just human nature. By using a sketch, you can really focus on that one big idea or give them multiple variations on how to present a piece of content.

This allows you to find which solutions to pursue. Without getting sidetracked by unnecessary details. All you really need to get started with sketching is a pen or a pencil and a sheet of paper. That's it. However, if you really want to create effective sketches, materials do matter, so I have a couple of items that I want to recommend to you. Now, if you're the type that really likes to draw with pencils, I would recommend getting a professional quality set of pencils with varying degrees of hardness. You also want to get some professional erasers that are going to erase content on the page without printing a lot of dust or tearing the paper.

Now, with these pencils, you can use the softer pencils to draw those really thick bold lines and you can use harder pencils to draw the soft, thin lines.Now, I personally love drawing with markers. And if you're going to do that I recommend getting a really high quality set of gray markers that come in gradation of values. You can see these guys come in like darker markers to lighter markers, which is great. This is great for indicating areas of emphasis or shading certain areas, and then you can come back over the top with them.

With either multi-colored markers, or thicker markers to really get your point across. I love drawing with these markers. Now quality paper matters too. And I recommend getting a sketch book, and this thing as you can tell is pretty large. It's big enough to convey your ideas. You want some high quality paper and being able to have a sketch book like this with perforated paper that you can tear out and share with people, is a real advantage. Now if you'd like a little bit more of a starting point, there are a lot of templates out there that'll simulate mobile devices for you or give you some grid layouts to choose from.

You're also going to want a good collection of rulers to help you draw straight lines and angles, and I find it very helpful to add post-it notes or sticky tabs to my sketches. These can be used to simulate user interactivity, element states or modal windows. If you're a visual designer, I'm guessing that nothing I've said so far is news to you. You're probably used to exploring ideas through sketching. If you're not, you might be hesitant to give sketching a try. I don't subscribe to the, you don't have to be any good at drawing to sketch school of thought. I mean it's true to a degree. But at the end, you do have to draw effectively to communicate your ideas.

If you can do that through rectangles and stick figures, that's awesome. But, if you can do it with fidelity, that's even better. What matters is that you practice. Like any other web design skill, your sketching will get better the more you do it. So, grab a pen, grab some paper, and start sketching out your ideas.

# Anatomy of a website

When I first started to design websites, one of the things that really kind of frustrated me was that everyone just seemed to assume that I already knew how to do certain things. For example, no one ever really told me the proper way to structure my websites. It was just one of those things, I guess, that everybody just sort of assumed I knew how to do. So I want to make sure that I showed you the basic structuring and anatomy of a simple site. This is going to helpyou, I hope, plan and structure your own project. As well as ensure that you're structuring things in the most efficient way possible.

So you can see, on my desktop, I have a folder called explore, and I've got a little website in here. By the way, I don't develop on the desktop. It's just kind of the cleanest. Way to show this to you. So if I open this up, here are my files and folders and this is actually a website. Not a very complex one, but it is a website. Now they're a couple of things that I want to point out to you. Number one, it's really just a collection of files and folders. For example, right here inside the folder I've got index.htm. Index.htm is kind of the home page of any site.

We used to have one of either two files, index.htm or default.htm. And I know that there's some people out there still using default, but index is certainly the most popular of those now. It's just, if you place an index.htm or .php or .cfm or whatever. Web server you are using. If you place that where they call the root folder and the root folder is just a folder that contains your website, then that's your homepage. That's the page that browsing to that particular site will go to.Alright. Now, you will also notice that I have other files that are what we call top level files.

In this case, mission, tour, resources. And that means they're in the same directory as the index file. However, we do have some files in subdirectories. In this case, there's a directory called resources. And if I open that up, I can see that I have an FAQ file in there, and a links file in there. Okay, so what does that have to do with the structure of the actual site? Well let's go ahead and check that out. So, let me sort of rearrange these and what I'm looking at right here is the live explore california site. Now this is a site I did years ago for one of the Dreamweaver courses.

It's just a nice sample site. It's still online and it serves the purpose of what we need to take a look at here. Okay, so as I mentioned before, we have a subdirectory called resources, right?So if I click on resources. I can see that it goes to the resources.htm file. Well, that's this file right here. That's perfect. However, if I click on the FAQ link, right here, notice that that now goes into a resources directory and finds the file FAQ.htm. So, if I open up that directory, you can see the correlation between the structure of the site.

And the address bar up top. So it's just mirroring that. Again it's nothing more than a collection of files and folders. Now, you may have noticed that some directories are named slightly differently. For example, all these directories up top have underscores in front of their names and they have things like CSS, images, video, scripts. Those are all resources and that means that they're not individual webpages themselves, they're resources that those webpages are using. For example, the CSS directory includes the site's CSS files, the images folder, as you would imagine, contains all of the images that are being used by the site.

And there's a couple of reasons why we name folders that way. Number one, the underscore indicates to me that that is a resource and not a section of the site, so it's not me trying to organize the site into logical sections. It's actually resources that the site is using. The other thing is by giving them an underscore it moves them all up the top any directory listing. So it's very easy for me to find and filter out my resources so that I can access them quickly or even ignore them if I really want to focus on the rest of the site.

I can just either sort of flip the order around or just sort of ignore all those underscore directories. It makes it very, very simple. To separate those. Okay, now this time as I mentioned before this is a really simple site and one of the things you're going to notice is that if you develop a site like this or a nice simple static site. And you get a web host and you link to theroot directory of your web host in anticipation of uploading this file. There's going to be some files already there that you may not recognize or might not know what those files are doing. So I'm going to really quickly jump over to my control panel for this particular site and take a look at the site as its live online.

Now you're going to notice that there are a few directories and things over here that at aren't actually in this folder as I mentioned before. In that folder, I've got sort of a stripped down version of the site. The other thing is, the live version of the site, I'm not the only instructor that'sbeen adding to this over time. So, there's a good bit of cruft that I could probably go in and get rid of some of this stuff. However, what I really want you to focus on right now, if I scroll down, here we go, is files like these. We have the htaccess file. We have these 400, 401, 403, 404 files, those types of things.

Alright, so, if you create a brand new site in a hosting company, chances are if you open up the directory you're going to find a default index.htm page that you can get rid of or overwrite. And then you're going to find a bunch of files like this. Like the 400, the 500 files, the 403s, the .htaccess file, those type of things. So the htaccess file is a configuration file that allows you toset certain things for your server, such as caching and things like that. Typically, you won't need to modify that. And if you do need to modify it. You're probably going to know a good bit about, websites at that point and be equipped to do it.

But, if not you might want to just research what the HT access file is, the things that you need to do in order to modify, and what it can do for you. It's one of those things that you shouldn't be scared to touch but it can have some dramatic impact on your site performance. So, you want to make sure that you're doing the right thing. Okay, I want to turn our attention a little bit closer to these error pages. So, that's exactly what these are, they're error pages. If I go, for example.Let's say I go back to my tour and let's say I'll try to look inside the images directory. Okay, now we did this in the last movie and you can see I have indexing turned off.

No indexing. So essentially I'd get a 403, a permission denied. Now if I look at the site, I'm seeing this file right here, the 403 file. Now the file that I'm actually seeing this page, that's just the default. If you don't ever touch them, that's what people are going to see. You;re probably used to seeing things like that, because there are a lot of sites out there that don't take these files and modify them. But if you want to you can. You can download these files and do something a little different with them and actually integrate them into your site a little better. If for example, I go to my resources and let's say I click on the privacy policy.

Okay, I don't have a privacy policy. I, on purpose didn't upload one. And, so, what I am looking at right here, this is what they call the 404 file. And this is if I download the 404 file from the server and modified it and added my stuff to it and then overwrote it and now, this is what I am getting. So essentially in addition to structuring your own files within your site. Be aware that there are other files in here, default files like the fave icon, things like that. That you're going to need to modify yourself.

When you get a new site take a look at the existing structure of it. Research some of those elements and think about what you might want to change. Before you come in and overwrite it with everything in your file. Alright now not all sites are as simple as the on I'm showing you, but for the most part. That's really how the majority of all sites online are structured understanding why sites are structured a certain way is really going to help you as you organize plan and create your own sites.

# Dealing with a multidevice world

Perhaps the biggest change in web design over the past ten years has been the arrival of these guys. You know, prior to smartphones, the biggest design challenge I really had to worry about was which screen resolution I was going to target. But now, since phones and tablets have begun consuming web content, we have an entirely set of screens to worry about. Techniques for designing for these multiple devices are still evolving. But it's a reality that's not really going to change. So as designers, we have to come up with new ways for planning and creating our sites.

That means if you're new to web design, congratulations. You don't really have to change anything. Instead of having to relearn how to design sites, you can just focus on designing for the multi-device world. The truth is, however, we're still in a state of flux. And not all solutions or best practices, are really fully evolved just yet. Because of that, you should make sure that your process of learning web design also involves staying current with how web design techniques are changing to deal with an ever-growing list of web-capable devices.

One of the biggest changes triggered by the rise of mobile devices, is how we approach planning our sites. Many designers now consider mobile first when planning sites, rather than a desktop. This movement has been championed by Luke W, who has coined it the term mobile first in reference to planning and structuring sites with mobile in mind first and then working up in screen size. There is evidence to back up the soundness of this plan as increasingly web content is consumed on mobile devices more than on desktop.

Regardless if you're planning your site design with mobile first, desktop first or as somewhat of an equal partner. You still need to have a strategy in place for how your content should look and function across multiple devices. Yeah, I'm not going to pretend that this is easy folks. It's not, but if you focus on this approach as you learn web design, it simply becomes the natural way that you work. While the current focus seems to be on mobile devices and smaller screens, it's overlooking an emerging trend.

We're quickly approaching an always connected world. There's an increasing array of web content devices that don't really fit traditional molds. Some don't have screens at all, yet still consume Web content. That means that in the near future, we're going to be designing for, not only devices like phones and laptops, but also cars, shoes, watches, glasses and, yeah, even household appliances. Making sure that your content is structured with standards-compliant code, and written with clear semantics, is critical to making sure that your content will be able to be accessed by this growing array of devices.

It's also likely that in the near future we're going to see new standards and new APIs designed to help us create content for these sort of non traditional devices. In the meantime, make sure that you focus on writing clean semantic code and learning best practices and techniques for displaying content across all these devices. It's the world we currently find ourselves in, and it's not likely to change.

# What is content strategy?

## Author – Professor [Patrick Nichols](http://www.lynda.com/Patrick-Nichols/1845050-1.html)

When you look at a website or use an app, you're interacting with content. Content is the information conveyed in copy, navigation and the visual design of digital experiences. Content strategy is the concerted effort to present that information in meaningful, useful and relevant contexts within the user experience. Content strategy is a discipline in many creative agencies and mid to large scale companies. This content strategist role combines elements of information architecture and copywriting, along with editorial method. But even if you're not called a contentstrategist, you'll use content strategy in creating websites and digital experiences.

Understanding content strategy will allow you to deliver the right information at the right time and in the right place. Information architecture and content strategy really go hand in hand. Specialists in each discipline often work closely together in a user experience team. In smaller engagements, one person may combine the expertise of these two disciplines into a single role. Content strategy places information in meaningful contexts. That way when users have a website or other digital experience, need information, they'll find it when and where it's needed.

# Working with wireframes

The wire frame is a document that maps how information, assets and objects are arranged within a container. That container can be a webpage, an app, or other digital experience. Wire frames are extremely helpful in planning the relative heirarchy of content within a shared space. You can use software like Omnigraffle or Visio to create wire frames, or you can sketch them on a tablet or piece of paper. What matters is that your wire frames have a clean look and feel that's easy for others to read at a glance. Though different wire frames can have different visual styles, most share a few basic design elements.

If you're new to wire framing, follow these design conventions. Show copy as wavy lines or placeholder text. Some people put real copy in their wire frames. But I recommend against it. You want reviewers to focus on relative placement, not real copy, and if you think reviewers won't be distracted by real copy, think again. The eye tends to gravitate to it by instinct. If you need to show some copy, use placeholder text instead. A good placeholder text language is Lorem Ipsum. You can find passages of it to use online.

For large navigation sections, if the labels are final you can use them in the wire frame. But as with copy, sometimes it's better to avoid too much detail. The same applies to links, which you'llwant to note if there're important page elements, but probably won't want to show in detail. Add images as rectangular boxes with a sample sketch or large X spanning each form. If you have videos, show them as rectangles with the sample play button instead of an X. These are just a few examples to help you get started. Follow whatever conventions will render your wire frames universally helpful and easily understood for all key contributors.

If you're working with an information architect, he or she may build and manage the wire frames. Work closely together to ensure content elements are prioritized properly. For more on wire frames, check out Chris Nodder's course, User Experience Fundamentals for Web Design. Wire frames are useful tools for planning digital experiences. Whether laid out in software or sketched on paper, they help you see and explain the relative placement of content within a shared space.