GYMNASIUM

UX FUNDAMENTALS

Lesson 3 Handout

Wireframes And Prototypes

ABOUT THIS HANDOUT

This handout includes the following:

- A list of the core concepts covered in this lesson.
- The assignment(s) for this lesson.
- A list of readings and resources for this lesson including books, articles and websites mentioned in the videos by the instructor, plus bonus readings and resources hand-picked by the instructor.
- A transcript of the lecture videos for this lesson

CORE CONCEPTS

1. Wireframes act as sketches for web pages, or for screens in a mobile app.
2. Wireframes are tricky to use, even by seasoned web designers. Clients have a hard time with them. Designers have a hard time with them. We're going to talk about that, and how to make them easier for you and for your clients.
3. Prototypes are clickable wireframes. And they let you test ideas.
4. Wireframes and prototypes work best when you're working quick and you're working cheaply.

ASSIGNMENTS

Now that you have observed some people using a product and seen some of their challenges, it's time to start designing a better solution.

From this assignment forward, you'll be actively designing a website or smartphone app that provides a better and easier experience doing the task you've chosen (booking a flight or finding a restaurant and making a reservation).

PART 1: CHOOSE YOUR AUDIENCE

An essential component of UX design is knowing your audience. Normally you would determine your audience with your client, but for this project you'll choose it yourself. You can choose an audience like the people you interviewed, or you can choose something completely different.

Exercise: Complete this sentence:

"My user is a _____, who wants to _____."

For this exercise, your audience should be very specific. This will likely make your product a niche or very

narrowly focused product, and that's fine.

Here are some examples of acceptable audience choices:

- My user is a college student, who wants to travel home cheaply. (Airline reservation system)
- My user is a single person, who wants to impress a first date. (Restaurant finding system)
- My user is a parent of an autistic child, who wants to find a restaurant where their child can be loud. (Restaurant finding system)

Here are some unacceptable audience choices, because they're not specific enough:

- My user is a hungry person, who wants to find a good place to eat. This is unacceptable because it's too vague!
- My user is a busy traveler, who needs to book a flight fast. This is also unacceptable because it's too generic! Be more specific!

You'll be testing with this audience later (not during this assignment, but in a future assignment), So it may help if you choose an audience you are likely to have easy access to. For instance, it may be unwise to choose "college students" if you're never around them.

PART 2: CREATE WIREFRAMES OF YOUR PRODUCT, FOR YOUR AUDIENCE

For this assignment, you'll generate a set of wireframes depicting a better experience for the chosen task (booking a ticket or finding a restaurant), in the chosen medium (website or smartphone app), and based on your audience and their needs ("My user is a _____, who wants to _____.").

The design doesn't need to be feasible or easy to build; your focus is on a better interaction for your user. So feel free to think "outside the box" and impractically, if it might provide a faster or easier experience.

You may use any software you like to generate the wireframes, or can draw them on paper by hand. Wireframes must be monochrome; no use of color allowed. (We recommend sketching by hand.) No polished work is allowed. Sketches or simple boxes only.

Your wireframes should convey the process of completing the task and the improvements you would make, and should convey your ideas and improvements as clearly and simply as possible. A peer or client reading your storyboard must be able to completely understand the idea without additional explanation. This could mean adding callouts, descriptive text, cartoons, symbols, a user flow diagram, or anything else you see fit.

Each wireframe should have a prominent title (e.g., "Choose a Restaurant", "Search Results", etc.) and a number or letter to identify it (A, B, C, or 1, 2, 3, etc.) so you can refer to it elsewhere.

Collect your wireframes into a PDF and add a cover sheet with the attached template. Post the PDF to the UX Fundamentals forum board. (Note: PDFs can't be directly attached to forum messages in the UX Fundamentals forum; you'll need to use Dropbox or another sharing service to generate a web address, which you can insert into a forum post.)

UX FUNDAMENTALS WIREFRAMES

Your name _____

Project chosen _____

(Booking a flight or finding a restaurant and making a reservation)

Medium chosen _____

(Website or smartphone app)

Audience statement:

My user is a _____,

who needs to _____.

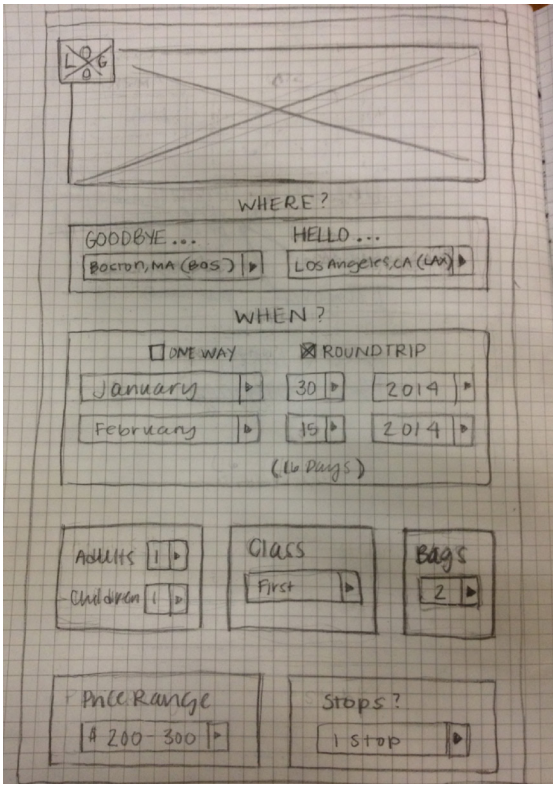
(Attach wireframes to this document; be sure to label and number each wireframe. Bonus: provide a brief description of each, a few sentences at most.)

ASSIGNMENT EXAMPLE

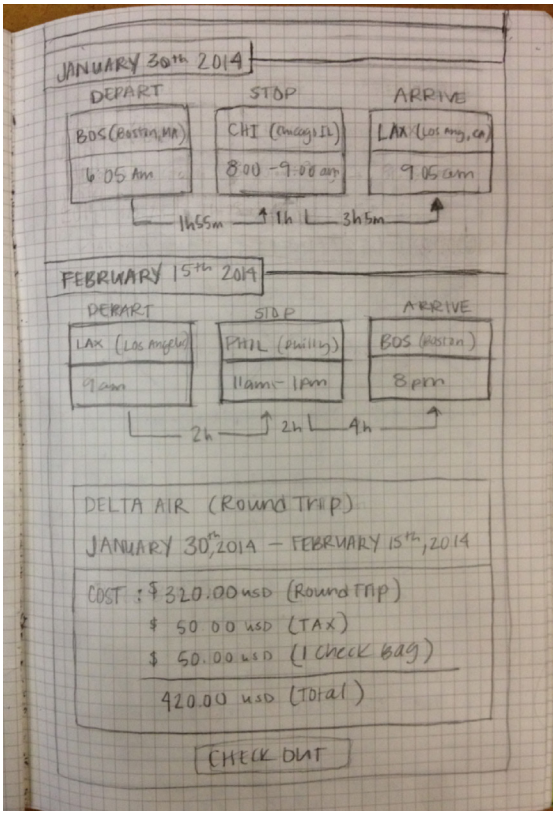
Here's an example of this assignment, from a student who completed the course:

*Wires 1 and 2 are Home Page examples. Wires 3, 4, and 5 are Summary Pages. Ideally the user would be able to book their flight without trouble and if they have any problems they can fix them along the way as opposed to going backwards.

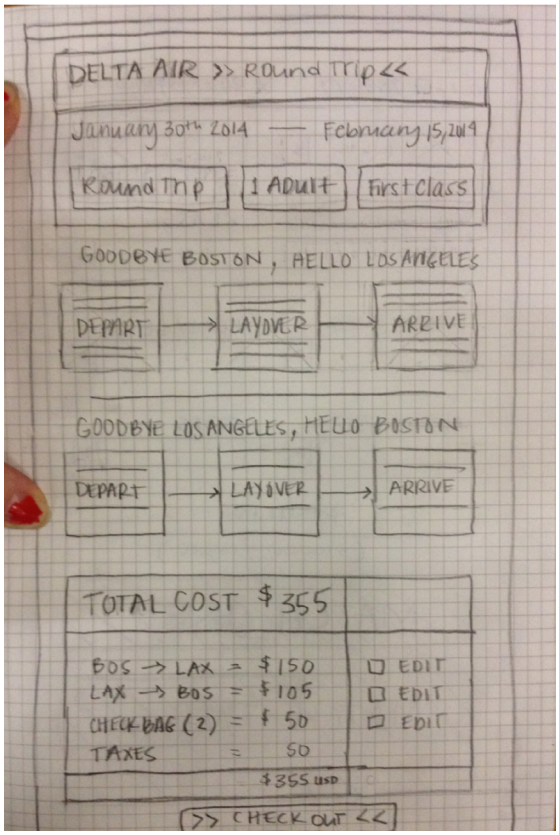
- Wire #1 - I sectioned off the most important parts of the flight details and made them larger so you are more aware of the choices you made.



- Wire #2 - I made it so you have the option of making your search a bit more refined. Once you make a certain choice regarding your trip it will automatically find flight matches on the bottom of that page.
- Wire #3 -Organizes a flight summary page. It has the date large at the top so you can be certain of the days you chose. At the bottom it has the choices you made and the option to edit them so you don't have to backtrack. It also breaks down the price of the flight.
- Wire #4 - Also organizes the summary page. It has almost the same layout as #3 but I decided to add the hours between flights to the information and divide the start date and the end date of the trip.



- Wire #5 - This wire has the flight options editable at the top and the price breakdown at the bottom.



RESOURCES

RESOURCES FOR WIREFRAMING

- (Resource list) All-in-one guide to wireframing (software, articles, etc.): http://www.gracesmith.co.uk/get-wireframing-the-all-in-one-guide/
- (Article) The secret to successful design sketches: http://zurb.com/article/227/the-secret-to-successful-design-sketches
- (Resources) http://wireframes.tumblr.com/
- (Resource) Turn any webpage into a wireframe: http://www.wirify.com/
- (Tutorial) A beginner's guide to wireframing http://webdesign.tutsplus.com/tutorials/workflow-tutorials/a-beginners-guide-to-wireframing/
- (Article) The "Boxing Glove" approach to wireframes: http://www.90percentofeverything.com/2008/01/02/the-boxing-glove-wireframing-technique/
- (Article) Using scissors and paper to create wireframes with clients: http://www.mstoner.com/blog/uncategorized/pwireframing_paper_wireframing/
- (Slideshow) The best tool for wireframing is the one you feel most comfortable with (also includes a good case study of a soup-to-nuts design process): http://www.slideshare.net/runger/the-right-way-to-wireframe-cidd-chicago-february-6-2013

- (Book) Communicating Design, by Dan Brown; the gold standard book about all sorts of UX deliverables: http://www.amazon.com/Communicating-Design-Developing-Documentation-Planning/dp/0321712463/
- (Software) Kit of photoshop elements that look hand-drawn: http://cloudcastlegroup.com/blog/basiliq
- (Software) Balsamiq — like Illustrator, but makes drawings look like they're hand-sketched: http://balsamiq.com/
- (Video) Wireframing over the shoulder — a start-to-finish timelapse: http://www.from-the-couch.com/post.cfm/title/wireframing-over-the-shoulder

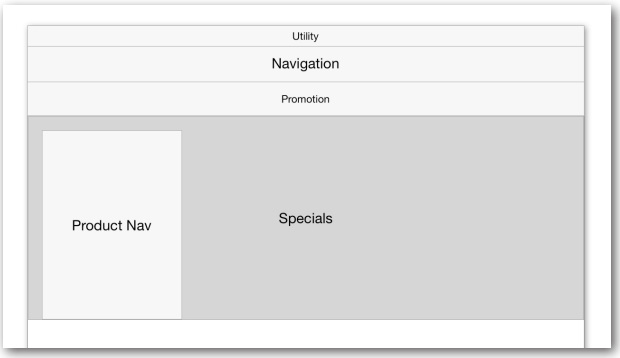
RESOURCES AND TOOLS FOR PROTOTYPING

- *Powerpoint/Keynote*
- *Acrobat (for making PDFs)*
- *Indesign:* http://www.smashingmagazine.com/2013/03/07/creating-wireframes-and-prototypes-with-indesign/
- *Templates for Keynote and Powerpoint:* http://keynotekungfu.com/
- *Prototyping with Apple Keynote:* http://keynotopia.com/
- *PDF prototypes:* http://amccrim.com/protopdf
- *Online tools:*
  - http://www.invisionapp.com/
  - http://uxpin.com/
  - http://www.hotgloo.com/
  - https://www.flinto.com/
- *HTML and CSS*
- *(Book) Prototyping: A Practitioner's Guide, by Todd Zaki Warfel* http://books.google.com/books/about/Prototyping.html?id=aieWBrFeRtUC

INTRODUCTION

(Note: This is an edited transcript of the UX Fundamentals lecture videos. Some students work better with written material than by watching videos alone, so we're offering this to you as an optional, helpful resource. Some elements of the instruction, like live coding, can't be recreated in a document like this one.)

This is UX Fundamentals, an online course developed by Aquent. This is lesson three. Today, we're talking about wireframes and prototypes. If you are a web designer and build websites, you have probably used one or both of these, wireframes or boxes on pages, that look something like this. They're often meant to reveal the underlying graphic design behind a finished website, consider them planning diagrams.



Now, a lot of folks have questions about the best way to use wireframes, or even why they use them at all. Wireframes and prototypes are very effective and versatile tools, but only if you know how to use them. That's what this lesson is about.

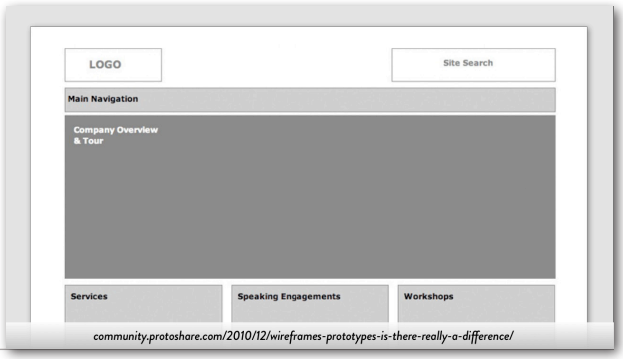
- Big idea number one: wireframes act as sketches for web pages, or for screens in a mobile app.
- Big idea number two: but they are tricky to use, even by seasoned web designers. Clients have a hard time with them. Designers have a hard time with them. We're going to talk about that, and how to make them easier for you and for your clients.
- Big idea number three: prototypes are clickable wireframes. And they let you test ideas.
- Big idea number four: wireframes and prototypes work best when you're working quick and you're working cheaply. We'll talk about what that means and how to do it. That's what's in store for today's lesson in UX Fundamentals.

BIG IDEA #1

WIREFRAMES ARE SKETCHES

Let's start with big idea number one, that wireframes are sketches. They're diagrams of web pages. Frequently the deliverable to an actual client looks something like this. It's a sheet of paper or a PDF and it's got boxes on it. Those boxes are labeled.

Like sketches, they're a way for us as a designer to be able to articulate ideas or even explore ideas before taking the expense of actually building the final product. Jason Santa Maria is a well known web designer that put it this way: *"Sketchbooks are not about being a good artist, they're about being a good thinker."* Wireframes, at their core, are sketches of web pages.



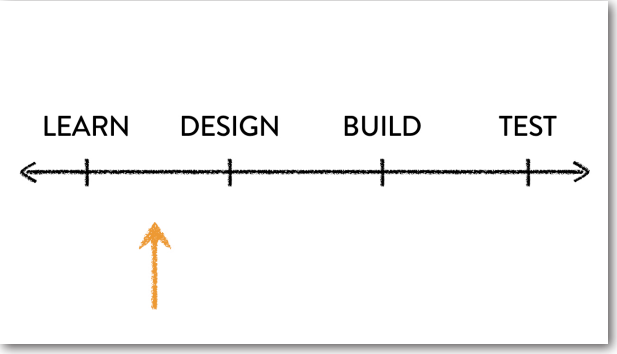
Here are a couple of real wireframes. This is one that's got gray boxes that are labeled with the sections of this particular page. Here's another one. The boxes aren't gray but it's the same idea; lots and lots of different boxes, each one labeled with what would be inside of it.

These wireframes have a couple of important jobs. They describe what goes on the page and how important each of those things are.

They do that by the position on the page and the size.

A good wireframe will also reveal the goal of each page. For instance, is the goal of that page to introduce a product to a new customer? Or to help them understand a choice they need to make? Or maybe to drive a click for a particular thing? That's what wireframes can help articulate.

I'm talking about pages as if they are just about websites. But wireframes work in the exact same way if you're building a mobile app that has screens. So whether it's pages or screens, wireframes work the same way. Making wireframes usually happens early in the design process. Every project is different and every company has a different kind of process, but generally speaking, a design process has a couple of different stages.



The first is a *learning phase*, sometimes called *discovery*, where you learn about the client, and the customer, and the project. Then a *design phase* where you're articulating some of those ideas into a solution. Then that solution needs to be built and then tested before it goes live.

Wireframes, like sketches, usually happen at the early stages of the design process. Hey, spoiler alert here. This process that we're talking about of learning, designing, building, and then testing is a common process in user experience design projects. More often than not it's not just a process that happens once but is repeated over and over again to get to a final product. We'll come back to this process in future lessons, but for now I just wanted to mention it's something you'll see again.

Another key characteristic of wireframes is that they have no graphic design. Comps are meant for design and that wireframes, which come before comps are not. At least that's what we've always been taught.

But it turns out, that's kind of a bit of a lie. Wireframes do have some graphic design. We'll talk about what that is and how to use it to your advantage.

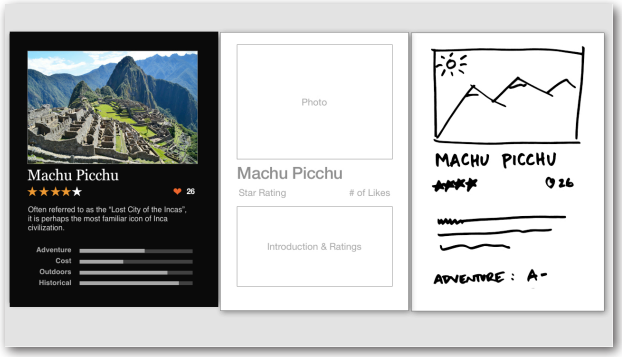
Given all of this, many designers say, why don't we just jump straight to comps? Why do we need to have wireframes? The simple reason is because comps are expensive.

So let's take our mobile app from the last lesson, "Travelly." We need to design a destination page, like for Machu Picchu, the ruins in Peru. We could comp up something like this, with ratings, and descriptive text, and lots of nice looking graphics.

We could show this to our client. If it's the first thing we show them they might say that it looks fantastic, but they might not. It might not fit some need or goal that the client or the project has.

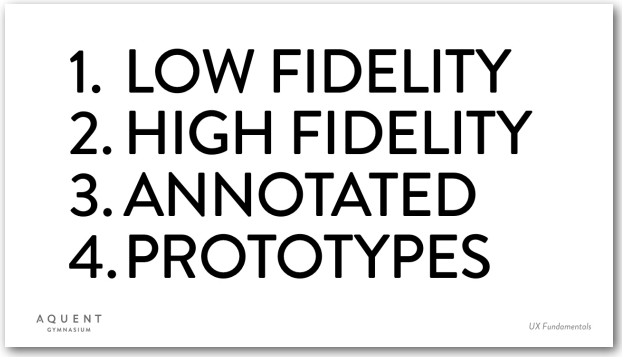
When you show comps to a client, you're not just showing one. You're showing lots and lots of different comps. Getting those wrong and having to redo them means hours and hours of wasted time, which means wasted money. In the beginning of a project the comp is maybe too much information. You may be able to deliver the same amount of information about what's on the page and what's important in a hand drawn sketch.

Fancy computer generated wireframes are essentially just that. They're fancier versions of hand drawn sketches. This is something that you can use today, that is to sketch more and comp less. In the amount of time that it took me to make this comp I could have come up with four or five different concept sketches, which would have revealed the same kind of information without all the time invested in making it look perfect.

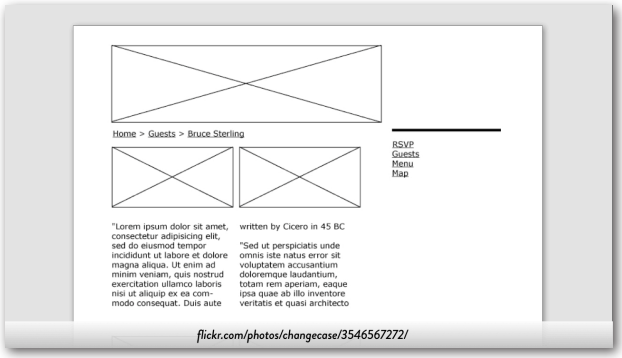


It's not just me saying this. Scott Childs, the creative director of Capital One says, *"Wireframing is the make or break a part of any UX project. It's where you want to fail fast, fail early, and fail cheap."* Scott's saying, we're probably not going to get it right the first time because we're human. So if you're going to get it wrong, get it wrong fast. Wireframing provides us some tools for doing this.

Let's get into the nuts and bolts of it. There are four different kinds of wireframing styles that you can use in your projects. I'm going to describe them in turn. The first are low-fidelity wireframes. They're the most basic, sketchiest kind of wireframes. Second are high-fidelity wireframes which provide more detail. Third are annotated wireframes, which describe each of the elements on the page in great detail. The fourth are prototypes, which are wireframes, but they're wireframes that are clickable and can actually do things.

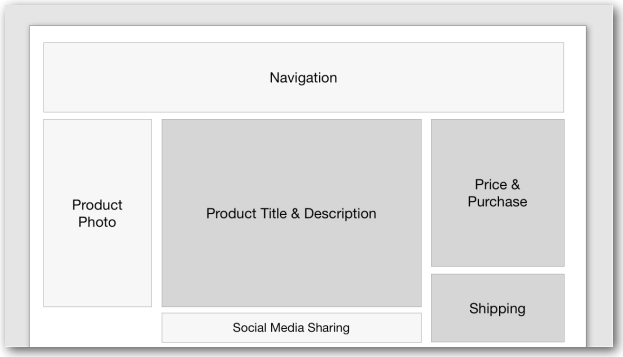


I'm going to go through each of these in a little bit more detail. Each of these has its own superpowers and its own weaknesses. Let's start with low-fidelity wireframes. Like all wireframes, these show what go on a page and how important each thing is. The examples that I showed you earlier are perfect examples of low-fidelity wireframes. There's not a lot of detail about what's in each of these boxes, just that the boxes exist.

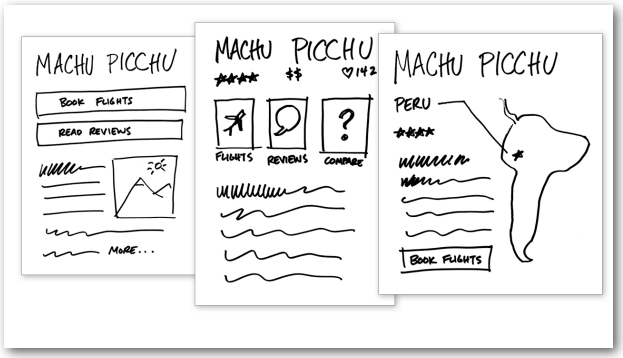


Here's another example of that with x'd out boxes representing images. It's got dummy text, lorem ipsum text. Wireframes don't need to be computer generated, however. This is a perfectly valid wireframe that's hand drawn, that shows the same level of detail about where things might go and what they might do.

Low-fidelity wireframes like this one can be very helpful for page layout. Now, you may be saying, wireframes aren't for graphic design. That's what we've always told our clients. Now come on, is this exactly true?

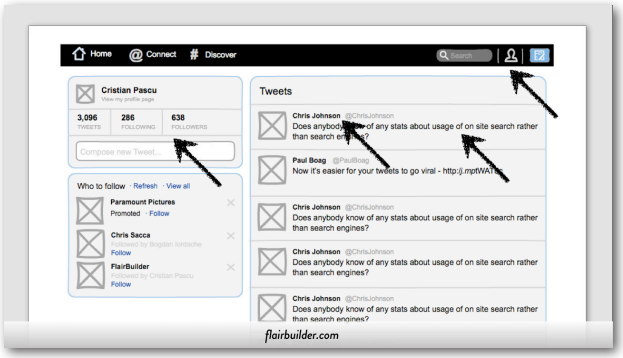


If we present wireframes to our clients that have elements laid out like this in a particular order, it seems like it has some relationship to the graphic design of the finished product. Wireframe purists will say no, it has no relationship. This just describes how important each element is. But more often than not, a wireframe laid out like this does pretty closely match a finished graphic design.



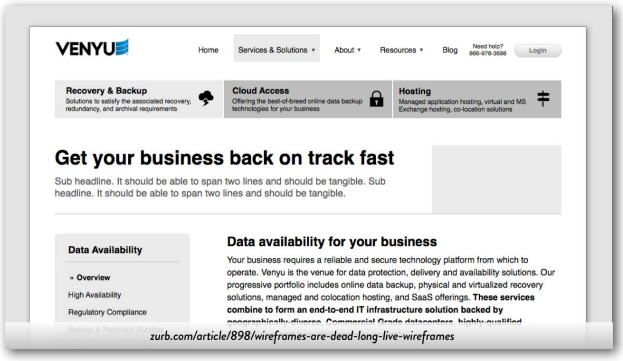
That's okay but wireframes can have some graphic design. How much graphic design they have is up to you. We'll talk about some techniques for managing that. Low-fidelity wireframes can also be really good for exploring concepts. Here are some sketches of that Machu Picchu destination page; three completely different concepts sketched in a very quick amount of time.

Low-fidelity wireframes can also give you a reality check on your content. Even lorem ipsum text like this can give you a sense of how much content you might need to fill the space. It's not perfect, but it can give you a yard stick.



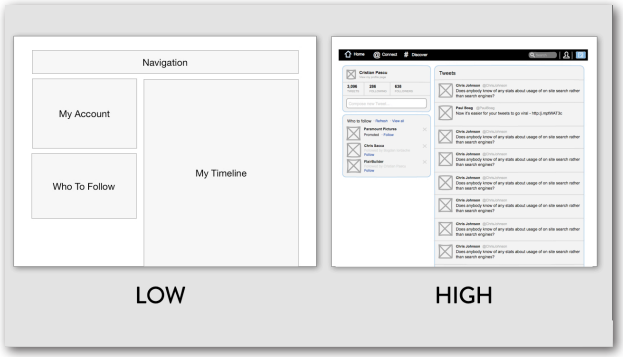
Now sometimes we need more detail, and in that case we turn to high-fidelity wireframes. These are wireframes that still show what goes on a page and how important each thing is. But the difference is, it shows it with a lot more detail.

Here's an example of a high-fidelity wireframe of a Twitter feed from twitter.com. You see the differences here? Instead of lorem ipsum text there's dummy text here that represents what actual text might look like. There's a lot of detail here about what would go in this particular box.



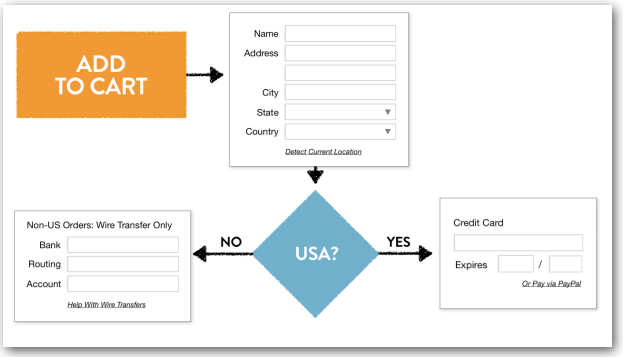
There's even typographical detail; some text is bold in some text is not. There are even hand-drawn versions of icons that might appear in the finished design. So in all of these areas there's a lot more detail. There's more detail about the content, more detail in typography, more detail about the graphics and about the page layout.

Here's that high-fidelity wireframe we just looked at and here's a low-fidelity version of the same wireframe. The difference is in the detail. Here's another high-fidelity wireframe, this one of Pinterest's homepage. The images are grayed out down below, but there's a good bit of detail about what the nav bar would look like.

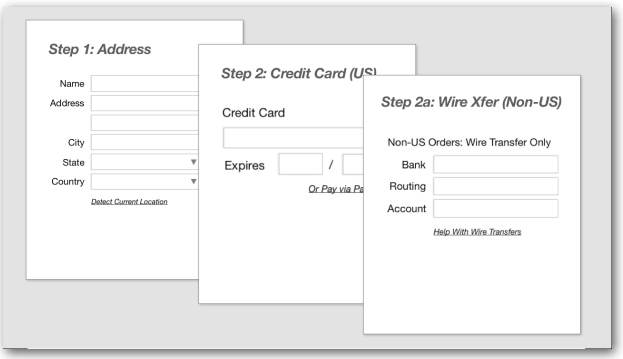


Here's a very high-fidelity wireframe that looks actually a lot like a comp. It's rendered in grayscale except for the logo and it has real text in most places. It even has real looking icons and navigation. This is a very high-fidelity wireframe, and higher fidelity wireframes are usually easier for clients to understand and easier for them to respond to because of the graphic design and because humans are visual creatures.

Here are the low- and high-fidelity versions of the Twitter feed we were looking at a second ago. Which do you think is easier for a client to respond to? The downside to high-fidelity wireframes is that they take longer to make. Let me pause here and say that wireframes, whether they're high- or low-fidelity, are not just good for single pages. They're also good for user flows or for how users accomplish tasks.

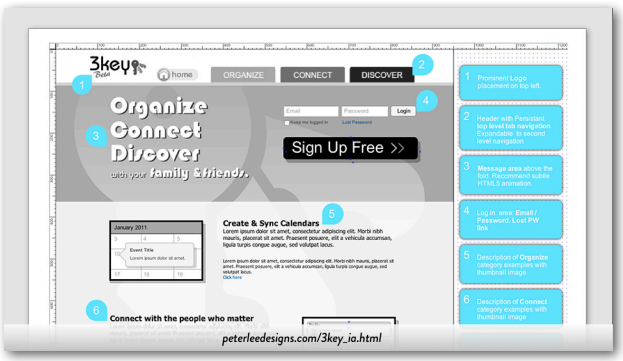


Do you remember user flow diagrams from our last lesson? Here's an example of one again. It looks like a flow chart. A customer might add a product to a shopping cart and then they have to enter their address. Let's say that based on whether they're in the United States or not, a couple of things have to happen.



If they are in the United States, then let's say they can pay by credit card. However, if they're not in the United States, they're overseas, then they've got to pay some other way like a wire transfer. Okay, so you've got a user flow diagram like this describing the process that your customer takes.

We might create little miniature wireframes around what happens when a customer enters their address, or enters their credit card information, or has to put in wire transfer information. This can help us figure out how easy the set of tasks is for a user to accomplish. Normally when we're delivering something like this to our client or to our team we might lay it out like this, step one, step two, and step three. My point here is that wireframes are okay even if they're just sections of pages, if it's some kind of an important interaction or important user flow.



The wireframes we've talked about so far have some significant drawbacks. There are some things that wireframes don't show. Even a relatively high-fidelity wireframe like this doesn't show animation. It doesn't show interaction. It doesn't show what happens if you hover over something or if there's a drop-down menu.

It also doesn't give any technical details about how a piece of dynamic content might be generated or about how a slide show might be driven. For all of that kind of detail we turn to our third kind of wireframe, the annotated wireframe. This is a special kind of wireframe because it's specifically about documentation.

Annotated wireframes give space to explain each part of the layout and how it's generated. Annotated wireframes often look like this, with each of these call-outs referring to some other document where you explain in excruciating detail what each of these things are and how they work. They can describe interactions and describe how something is generated. They're fantastic for developers, because they give a lot of detail about how to build the website.

But they're for explaining a design that's already been finalized. They're not very good at coming up with new ideas. Annotated wireframes can also take a very long time to make.

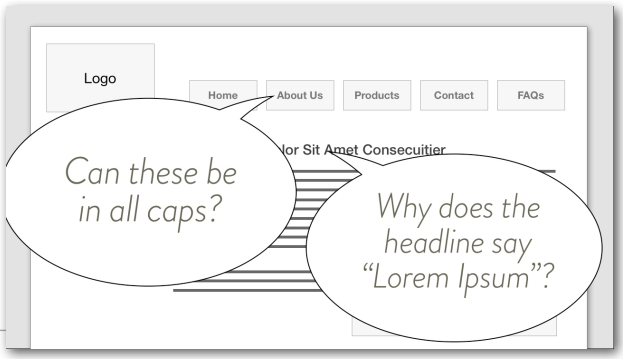
The fourth style of wireframes is prototypes. Prototypes are clickable wireframes. They're like faked websites. Here's one done in Keynote which isn't Macintosh's version of PowerPoint. It's an app, but it's completely fake. These are Keynote slides we're looking at right now. They're hooked up to make it appear like it's an app.

They're fantastic for testing ideas because you, and your clients, and your users can actually use a product, and test it out, and see how it works. They can take a long time to make, certainly longer than a sketch. But in the end they're almost always cheaper than real code.

So, four different kinds of wireframe styles, low-fidelity, high-fidelity, annotated wireframes, and prototypes. Which style is best? Which one should you use for your website?

These are tools, so it all depends on the problem that you need to solve and what you need to discover through your design process. Here are a couple of examples, though, to maybe guide you along the way. Let's say you have an app where customers can search and review trips. That calls for some low-fidelity sketches to figure out some ideas and then a prototype to test it and see how it feels.

How about a project with a new content management system for your company and a tech team that's not in the same country as you? You need high-fidelity wireframes and annotations to be very, very explicit and clear about what needs to happen in the new design. What about if you have a brochure website that just needs branding and doesn't have a whole lot of fancy interaction? Jump to loose comps



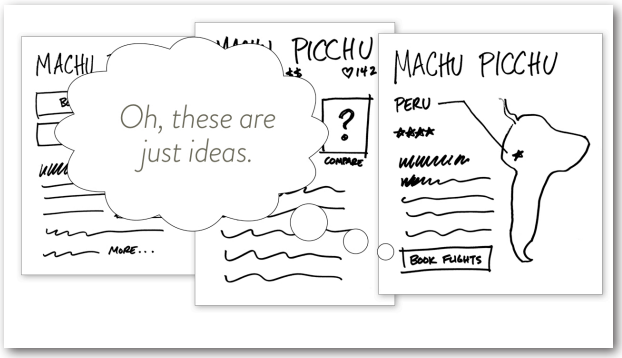
or high-fidelity wireframes so the client can see what they're going to get.

Or what if you have an update to an existing site? You don't do wireframes at all. Jump straight to comps for something like that. The point is, you can use different wireframe styles for different situations. You can fit the tool to the job at hand. Wireframes are useful planning documents in this way, and they're like sketches.

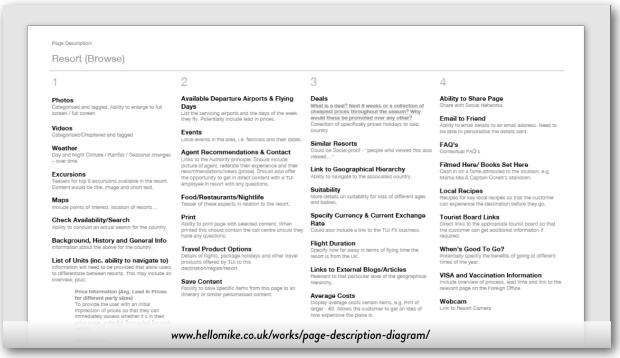
BIG IDEA #2

THEY'RE TRICKY TO USE

That takes us to big idea number two, which is that wireframes, even as simple as they seem, are actually quite tricky to use. We'll talk about why that is, and more importantly, how to make them better.

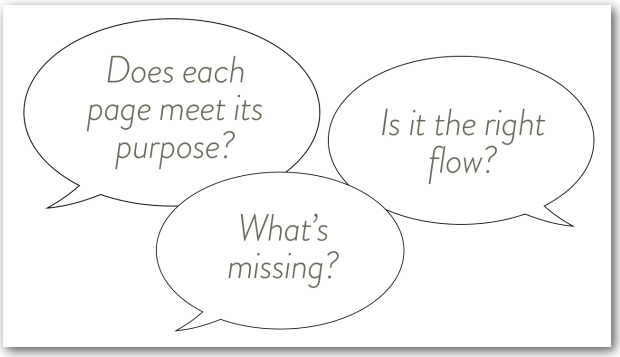


Wireframes have a number of limitations. A key one is that clients have a hard time understanding what they are. Makes sense, clients are not graphic designers. If you show a client a wireframe that looks like this, a perfectly legitimate wireframe, they might have all sorts of design-y questions, like what's this dummy text here? What does it mean? Can we change the style of this text or these buttons?



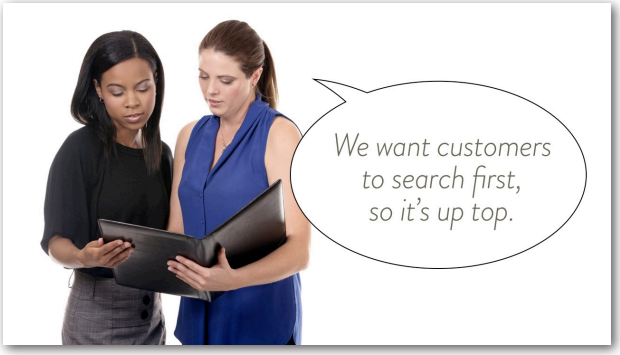
People are visual animals. This is why clients love to jump straight to mock-up. Iin fact, if you show them wireframes, sometimes they'll get frustrated and angry because they feel like they're not looking at design.

It's not just clients that have a hard time with them. Designers disagree about wireframes. They disagree about the style of wireframes that are most effective, especially when they're in the middle of a project, they get frustrated and want to jump to Photoshop and the tools that they know the best.



Sometimes designers do a little bait and switch with wireframes. I've been guilty of this myself. That's when a designer shows a client a wireframe that looks like this, and very carefully explains, hey, this is not for visual design. This is for the elements on the page. The designer pushes the client through that process. Then you know what they get back when the comp's finally done? Almost the exact same thing. It turns out it was actually for visual design.

So let's talk about clients and how to make things easier for them. First, the visual style that you give your wireframe is very, very important. If you deliver your wireframe in a style like this one, a high-fidelity wireframe with lots of details, the client is likely to think that has to do with the visual design, that that's how the site is going to look. If you provide a very low-fidelity wireframe, or hand-drawn sketches like this, the client is not going to think that's how it's going to look. They're going to think that they're just ideas.



So if you're trying to get ideas across, make your wireframes intentionally rough. If you use Illustrator to create wireframes, use the charcoal brush to make lines look uneven and imperfect. This piece of software is specifically for making wireframes. It's called Balsamiq. It lets you work like you would in Illustrator, but it makes the lines and the text look irregular and uneven, like they're hand-drawn. When it's time to make it look nicer, you flip a switch, and the typography and the line styles change to make it look tighter.

Another way to handle this is to get rid of the visuals from the wireframe entirely. These are called page description diagrams, and they serve the same purpose as wireframes by describing what goes on the page and how important it is. But it does it without any visuals at all.

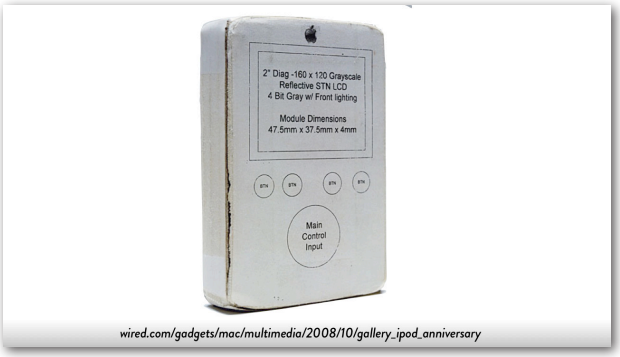
Another key to making wireframes less tricky is how you present them to the client, incredibly important. Many of us may find ourselves in the situation where you finish up your wireframes on a Friday, and you send them off to the client as you're ready to go home. Well, throwing them over the fence is a very bad idea. It's important to sit down with your client, either in person or over the phone, and go through each wireframe, talking through the goals of the page and the goals of the site.

You might ask questions like, is this page meeting its purpose? If you have a user flow, is it the right one? What's missing from these pages? If you have a wireframe like this, you might talk through, hey, this is where this headline is coming from. And this button down here on the bottom? Do we need this button down here on the bottom of the screen? These are the kinds of questions that you want to be talking through with your client. Don't just throw it over the fence.

It's also important to refer your wireframes back to the project's goals; that if something is in a particular location, it's there because of a goal in the project. Some designers like to get buy-in from their clients by collaboratively wireframing, by working on a whiteboard with your client, or modifying wireframes on your computer right there with your client.

If you have scenarios or user flows that you're describing in wireframes, be sure to walk through them with your client so they understand and can give you the feedback that you need. If you have a login form, explain what's happening in each element of that login form, and when they click something, the kind of place that they might go.

Now, your clients may want to talk about graphic design, even though they're looking at wireframes. They might be interested in the treatment of graphics, whether there's enough space for type, and the specifics of the layout of the wireframe. If they do this, that's okay. Take notes on all of these topics and then move on.



wired.com/gadgets/mac/multimedia/2008/10/gallery_ipod_anniversary

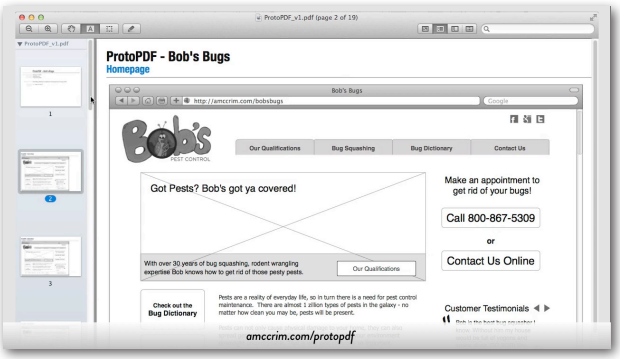
The thing you really want to watch out for, though, is if your client is quiet. If your client's not giving you feedback about your wireframe, that is very bad. What's happening is they're waiting until they see comps. This isn't good, because the entire point of wireframes is that they're cheaper to change than comps are. But wireframes only save you money in the process if you use them well. And unfortunately, they can be tricky to use.

keynotopia.com

BIG IDEA #3

PROTOTYPES LET YOU TEST IDEAS

Big idea number three is about prototypes and how prototypes let you test ideas. Prototypes are just wireframes that are clickable. That's it. They're fake websites or apps, like this one, that let you test out how an interaction works. Things about how an error message might come up, or how an animation or a transition might look and behave. It can act as a proof of concept for a new idea. It could be a way for you to show your clients and colleagues how a design might work. It can even be a way to test out an interaction with customers. Bring them in, sit them down with a prototype, and ask them to do something with it.



amccrim.com/protopdf

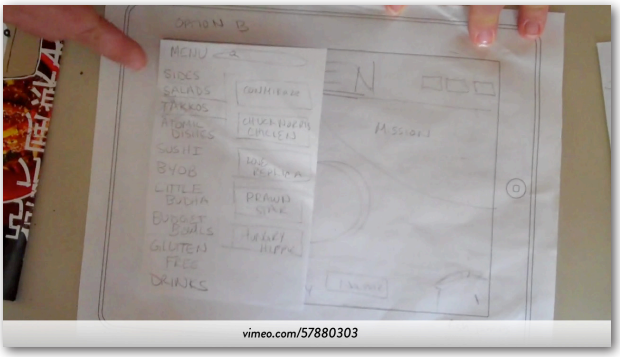
Now, they're not complete websites or applications, of course. They're fake, like this mock-up of the very first iPod. It's meant to give you a sense of how it feels and what it looks like. Fake websites are cheaper than the real thing. But prototypes still take time and energy to create, so it's usually best to test just one thing or a narrow range of important things, like navigation or shopping and checkout interactions, or a rating system, for instance. Things that are complex or that are especially popular or both.

When you're ready to make a prototype, well, there are lots of different ways to make them. One popular way is by using PowerPoint or Keynote on the Macintosh to make slides that turn into screens. You can do the exact same thing with a PDF using Acrobat or InDesign. If you know HTML and CSS, you can use that to make prototypes, and there's a whole crop of new custom tools that can be used to make proto-

uxpin.com

types, as well. I'll provide examples of each of these in the resources section here in the classroom, but let me go through them, one by one.

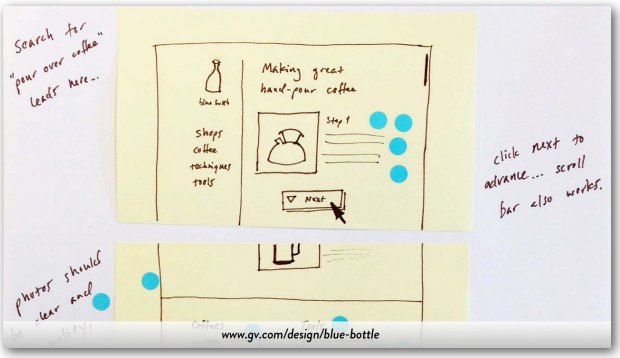
Using PowerPoint or Keynote to create prototypes is pretty straightforward. Here's an example of one in action that's created in Keynote for the Macintosh. There are two decks open here. The deck on the left is the prototype. I'm clicking through the individual screens. The deck on the right is for elements that I can copy and paste into it, like this button here, which is just a graphic. Now, I use keynote to hook this button up to go to the slide that corresponds to the state when that button is pressed. It clicks and it goes here to this slide, and then I can hook this button up so that it goes back to the original slide.



I can do the exact same thing with the PDF out of InDesign or Acrobat. Here's a PDF in which each page of the PDF is a page on the website. The buttons here jump me to the corresponding page, and so the end result is that it feels like it's a working website. But really, it's just a PDF.

If you know HTML and CSS, there's lots of different ways to make prototypes there, too. One cheap and dirty way to make HTML prototypes is to save out your wireframes or mock-ups as JPEGs. Bring them into pages. Another way is to use a framework like Bootstrap, which lets you build pages that look pretty good, pretty darn fast.

If you don't know HTML and CSS, that's okay, because there are a whole suite of new custom tools specifically meant for UX designers to create wireframes and then prototypes based on those wireframes. Some of them are free. Some of them are for pay. Some of them are online, and some of them are downloadable. I'll include some links to these in the Resources section of the classroom.

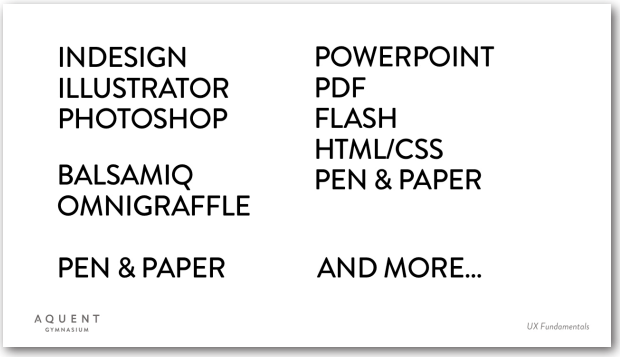


Here's another way to make a prototype; one that's not digital at all. You can create a paper prototype by printing out a wireframe or a mock-up or drawing it by hand, and then manipulating the sheets of paper when you want to demonstrate how an interaction works, going from one page to another, or fly-out menus, or any other kind of interaction. This is something you can do today; create paper prototypes just by printing out what you've already got. You can even test them on a stranger to see if an interaction that you've designed makes sense.

If you want to do it, here are a couple of tips. You should test a specific thing; not the entire website, but one specific kind of interaction. Ask questions that are open-ended and get them toward that. For instance, if you're testing an online store, you might ask, how would you buy a product?

Here's an example of the entire process in action. It's not a paper prototype but a Keynote prototype. It was for this company, Blue Bottle Coffee, a New York and San Francisco boutique that was selling coffee online, but their store wasn't doing very well. The designers had one week to come up with new design directions that might be more effective for their online store. Here's what they did. During the first day, they interviewed customers. They went out to prospective customers and asked them questions about how they purchase online and how they might purchase coffee. Back in the office, they collected the results and put them on Post-It notes on their whiteboard.

The second and third day, they sketched ideas based on those interviews; ways to solve problems for their customers. They drew storyboards about how customers might want to buy coffee, and they voted and chose the three designs that they thought might work the best. On the fourth day, they prototyped those designs using Keynote, creating slides that tested out the major interactions.



They also did some significant graphic design on these prototypes. These are very, very highly designed prototypes. Usually, prototypes don't have this good a polish. On day five, they tested them. They brought in a different set of consumers and had them try to buy coffee. They collected all the results on their big whiteboard and looked for the patterns that helped them get to a good design direction. They ended up redesigning the website, and their sales growth doubled.

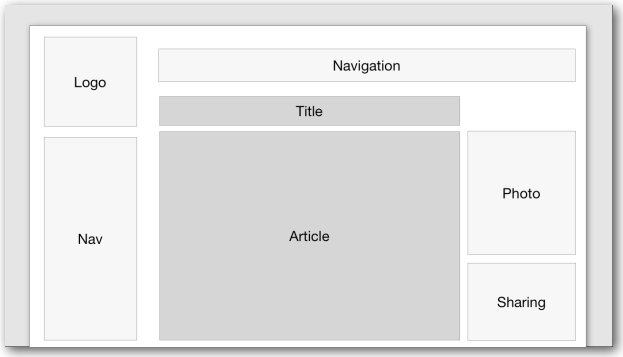
That entire research, prototyping, and interviewing process took five days and required nothing more technical than whiteboards and Keynote. Their five-day process involved learning about customers by interviewing them, designing by sketching, building prototypes, and then testing those prototypes with real people. If they had had two weeks, they might have done it twice. Hey, it's the same cycle we saw before. Learn, design, build, and test. I told you we'd see it again. Just like wireframes are a way to let you sketch ideas for your websites, prototypes are a way to let you test them.

BIG IDEA #4

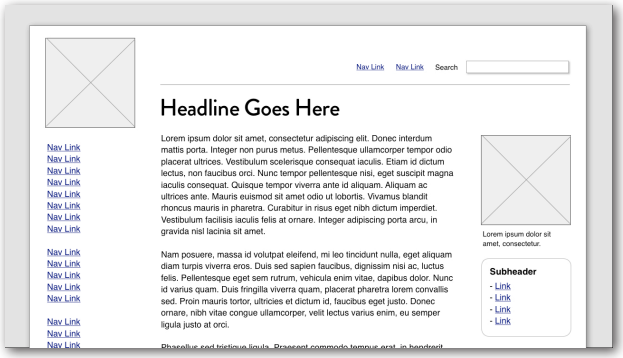
WORK FAST AND CHEAP

That brings us to our final big idea, which is that wireframes are most effective when they're fast and cheap. There are a million different tools that can help you make wireframes. Here are some of them. InDesign is great for making wireframes, and so is Illustrator or even Photoshop. There are specialized applications that are meantspecifically for wireframing, like Balsamiq or Omnigraffle, and then there's good old pen and paper.

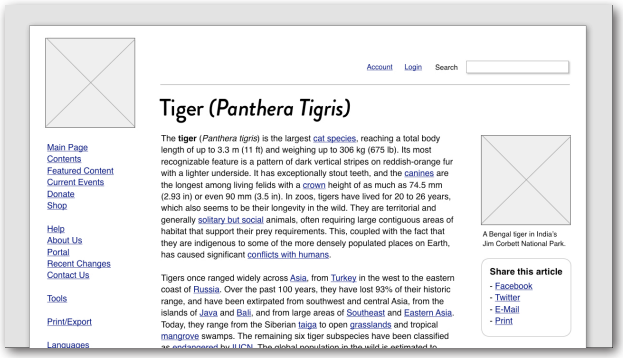
If you're making prototypes, you can make them in PowerPoint or Keynote. You can do them in PDF using InDesign or Acrobat. You can create them in Flash or HTML and CSS or, heck, pen and paper. This list is nowhere near complete. There are lots and lots of tools. So which is the best tool?



I'm about to give you my personal recommendation for the best tool for making wireframes and prototypes, and that is *whichever tool you're already using.* If you are already competent at any of these things, that's what you should be using to make your wireframes and prototypes. Assuming, of course, your company doesn't have standards, and then you're going to follow their standards. But if you're on your own or you get to choose, choose the tool that you're most comfortable in. Why? Because if you're comfortable, you'll be fast. If it's not fast, then you might be wasting time and money learning a tool.

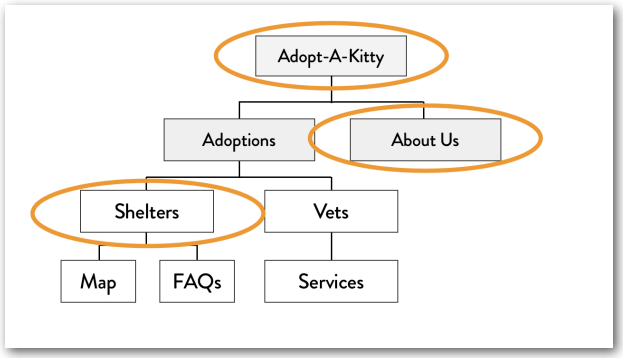


Here are four tips for working faster when creating wireframes and prototypes, no matter what tool you use. Tip number one: reuse and recycle elements, and I'm specifically talking about user interface elements. One great way to do this is to use libraries online. There are lots for free, and there's some for pay. These libraries are buttons and pull-down menus and nav items that you can use in your wireframes. Why spend all the time developing what a pull-down menu looks like? Just drag and drop it from somewhere else.



There are ones like this one which have some nice design polish. There are others that look more hand-drawn. There are even ones that are specifically meant for mobile devices. Over time, as you do wireframes and prototypes, you'll build your own library that you could reuse over and over again.

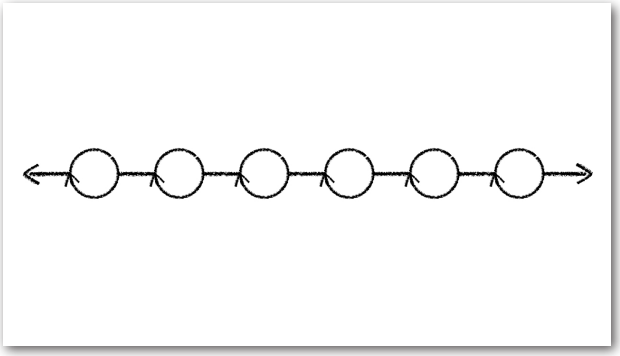
Tip number two: show the important content in your wireframe, and here's what I mean by that. Let's take a wireframe for a page, and there's a logo and a nav bar on the left-hand side. In the middle column, there's a title and then an article, and on the right column, there's a supporting photograph and some sharing (Facebook, Twitter-type stuff), and then a navigation bar that strips across the top. This is a very standard, typical, low-fidelity style wireframe. Now, there is no content represented in this wireframe. All we're seeing are gray boxes.



Here's another style of doing this exact same wireframe with lines and squiggles. It's the same layout, but I'm representing the content here graphically, with big squiggles for headlines and little squiggles for text.

Here's another way to represent the content. You could use dummy text. It could be lorem ipsum text, or text like, headline goes here. Finally, another way you can show content on a wireframe is to make it real content; to make it genuine content that's prototypical or an example of the kind of content you might find on this particular page.

All four of these are ways to handle content. Real content is a lot easier for clients to understand. If you show a client a wireframe like this, they can get the page layout and understand generally what it's trying to convey, but this offers a lot more detail. Granted, some clients may react to the specific text that's chosen, but it really gives you a way to test what the content will do in that space and how exactly the navigation will work. There's a benefit to this for you as a designer. Using real text makes you think harder about the design and about the content. It will make the solution that you provide probably stronger, but you don't have a whole lot of time, so don't do all of the text, because that would be a waste of your time.



Here's a rule of thumb. You should take the important content in your wireframe or prototype and make that content real. So for instance, here on this tiger page, the important content should stay real. The unimportant stuff, you might convert into squiggles and lines. This is the best of both worlds. It makes you think harder about the content and the problem that you're trying to solve, and it's easier for clients to understand.

Tip number three: if you're making a set of wireframes, don't just do the homepage or the homepage and a subpage. Here's an example of what I mean. You've got a website that's about cat adoption, and there are two main categories from the homepage; Adoptions and About Us. Underneath Adoptions, there are a couple other categories. One's for shelters and one's for vets. Let's say there's even more content underneath there. It would be very easy of us to just do a wireframe for the home page and a subpage, or maybe a third-level subpage, and think that that's all that we need.

But that can lead to a nasty surprise, that can happen down in that fourth level. Sometimes, navigation on this one can be tricky, and especially when getting back to the levels above it. A good practice is to do at least one wireframe for each level of navigation. Cover all the levels of navigation. Also, be sure to cover any important interactions that you have in your website.

The last tip is, plan to get your wireframes wrong. This is a little counterintuitive. Many designers, like myself sometimes, wish we could just go in a corner, make something beautiful, design an amazing product, get the client to accept it, sign off on it, and then move on to our next project. But that's not the way that projects usually go. Accepting that and planning for that is a way to, ironically, make them go faster and better.

I'm talking about the cycle of learning what the business problems are or what consumers want, and of designing something that solves those problems or meets those needs. Building that thing, even if it's just a prototype or a wireframe of it, and testing it, either technically testing it, or testing it with consumers or your client and then expecting to repeat that process. Instead of a straight linear start-to-finish kind of design process, it's cyclical, and you plan for multiple versions of your product or multiple iterations. Each time makes the product a little bit better.

This kind of design methodology is sometimes called *Agile*, and it's called this because, like it sounds, it's a design process that lets you change direction quickly and make your product better and better.

So those are four tips on how to work with wireframes and prototypes more quickly.

- Tip one, to reuse and recycle user interface elements like buttons and pull-down menus.
- Tip two, to show content, real content, for the stuff that's important on your wireframes and your screens.
- Tip three, to make sure that you wireframe out all the levels of your site's navigation and all of the important interactions.
- Tip number four, plan to get it wrong. Plan for iterations, and plan on using cycles to make your designs better and better.

That's big idea number four: wireframes and prototypes are best when you work fast and cheap. This is UX Fundamentals.

ASSIGNMENT

This assignment is in two parts. Complete instructions are in the classroom files, but I'll go over it briefly here.

Part one, do you remember the project you did user research for in the last session? Well, we're going to take that project, and you're going to choose an audience for the project. You're going to do that by completing this sentence: *"The user is a ___who wants to _____."* You get to decide who and what this is. It doesn't necessarily have to be someone that you interviewed.

However, when you create this sentence, when you fill in the blanks, it needs to be specific. A good example might be, "The user is a college student who wants to fly home cheaply". The audience is specific, college students, and the need is specific. They need to fly home cheaply. And price is more important than anything else.

A vague audience statement is not good. For instance, this audience statement, the user is a busy traveler who wants to book tickets easily is no good. It's too hard to grab a hold of this user and their needs. You'll need to choose something more specific. I've included more examples in the classroom materials.

Part two of the assignment is to create wireframes for your project, your medium, and your audience. So for instance, your wireframes might be for a website for college students who need to book a cheap flight home. Whatever your combination is, those are the wireframes you need to make.

Those wireframes can take any form. You can use any software. You can even draw them by hand. If you choose not to draw them by hand, that's fine too. You can make them look like boxes like this, or you can be more detailed, like this.

However, a couple of things are important. First, your wireframes need to make sense, and they need to make sense to other people. Avoid cryptic language. They need to convey the process that you're trying to describe. You're trying to solve the problem of a college student who needs to go home cheaply, for instance.

Your wireframes need to convey your solution to that problem, and how your solution is better than any other solution out there. In other words, your wireframes need to make sense. This is sometimes harder than it seems. Your wireframe should show the important content on your website, like the example that we looked at in class.

I've included more guidelines, instructions, and expectations in the classroom files. Check them before you start working. When you're done, package up your wireframes as a PDF and post a link to them in the forum so that I can see them, and so that your fellow classmates can too.