UXPin

THE GUIDE TO MOCKUPS

Mockup types, methods and best practices

INDEX

CHAPTER ONE

Introduction

A quick note from the authors

W hile the terms may be used interchangeably, mock-ups are not wireframes. And they definitely aren't prototypes. Mockups are the models of web design and one of the most effective mediums for communicating visual design. Whatever they lack in interactivity is more than made up for in terms of visual clarity.

In this book, we'll share a wide breadth of expert commentary, theories, practices, and real-life screenshots of mockups. We'll compare mockups to other design deliverables and discuss the pros and cons of each. For more experienced readers, we've included best practices for creating mockups in Photoshop and Sketch. We hope that you see mockups as a collaborative design tool rather than just something to throw to developers.

Mockups represent the structure of information, visualizes the content, and demonstrates basic functionalities in a static way. We've included screenshots of mockups (more of which you can find at UXPorn) and plenty of discussion around different types of mockups and methodologies for creating them. We've also included our own preferences, and described our own design process practiced at UXPin.

We'd love your thoughts on what we've written. And feel free to include anyone else in the discussion by sharing this e-book.

Happy designing,

Jerry Cao
(co-written by Matt Ellis & Narek Khachatryan)
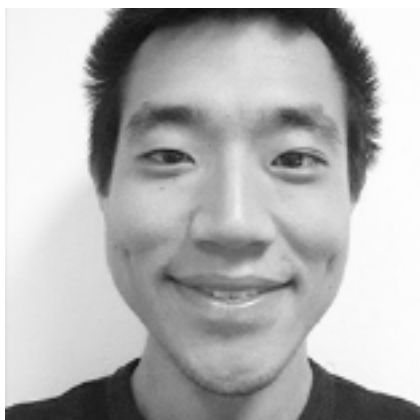
Jerry Cao is a content strategist at UXPin where he gets to put his overly active imagination to paper every day. In a past life, he developed content strategies for clients at Brafton and worked in traditional advertising at DDB San Francisco. In his spare time he enjoys playing electric guitar, watching foreign horror films, and expanding his knowledge of random facts. Follow me on Twitter.

With a passion for writing and an interest in everything anything related to design or technology, Matt Ellis found freelance writing best suited his skills and allowed him to be paid for his curiosity. Having worked with various design and tech companies in the past, he feels quite at home at UXPin as the go-to writer, researcher, and editor. When he's not writing, Matt loves to travel, another byproduct of curiosity.

Narek is a California-based product designer and user experience advocate. He's currently building UX optimization tools for mobile apps with his team at Simplitial Technologies. When off the grid, you'll find him reading, climbing rocks and exploring the great outdoors. You can follow him on twitter @narekk.

What is a Mockup?

Its definition, benefits, and role in the design process

Before we get into how to make the most of your mockups, it helps to know what exactly they are — and what they're not. As discussed in *The Guide to Wireframing*, mockups, wireframes, and prototypes can be confused with each other, making it difficult to find accurate information on each. However, all are integral parts of the UX design process, and so each should be given due attention.
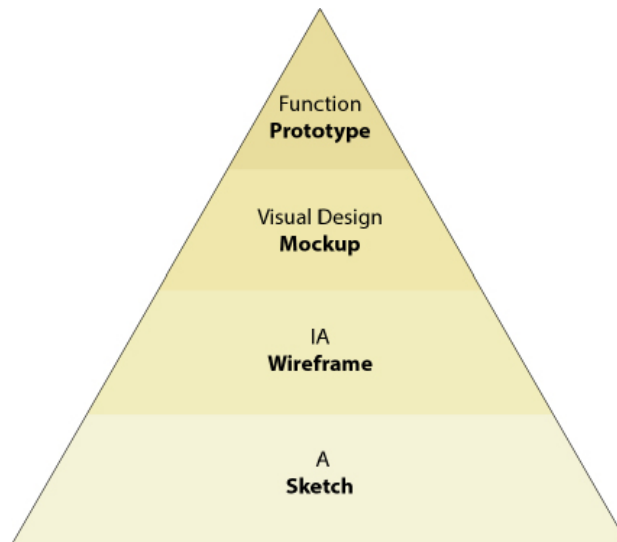
Photo credit: uxpin.com via uxbooth.com

We'll start our exploration of mockups by giving a broad overview, discussing the terminology, and showing how they fit into the design process as a whole.

A Word about Wording

Getting back to basics, before a website or app is released, it goes through three preliminary stages. These stages allow the team, and sometimes a few select users, to test its appearance, structure, and functionality before its release. Ideally, these preliminary stages allow you to fix any problems while they're still small, and fine-tune your design to communicate your message most clearly. However, because these phases are similar, they're often confused with each other. That's why we'll take a moment here to clarify the proper terminology for each. **Marcin Treder, CEO of our own UXPin**, lists the three formative stages of the UX design process as wireframes, mockups, and prototypes. Let's take a deeper look at each stage.

I. WIREFRAMES

Wireframes are typically a low-fidelity, bare-bones blueprint, usually represented with gray boxes and placeholders for detailed content. Their goal is to help establish what goes where, without consuming too much time on aesthetics just yet. A good wireframe should explain:

- **how the content is grouped**
- **how the information is structured**
- **the most basic visuals involved in the UI interaction**
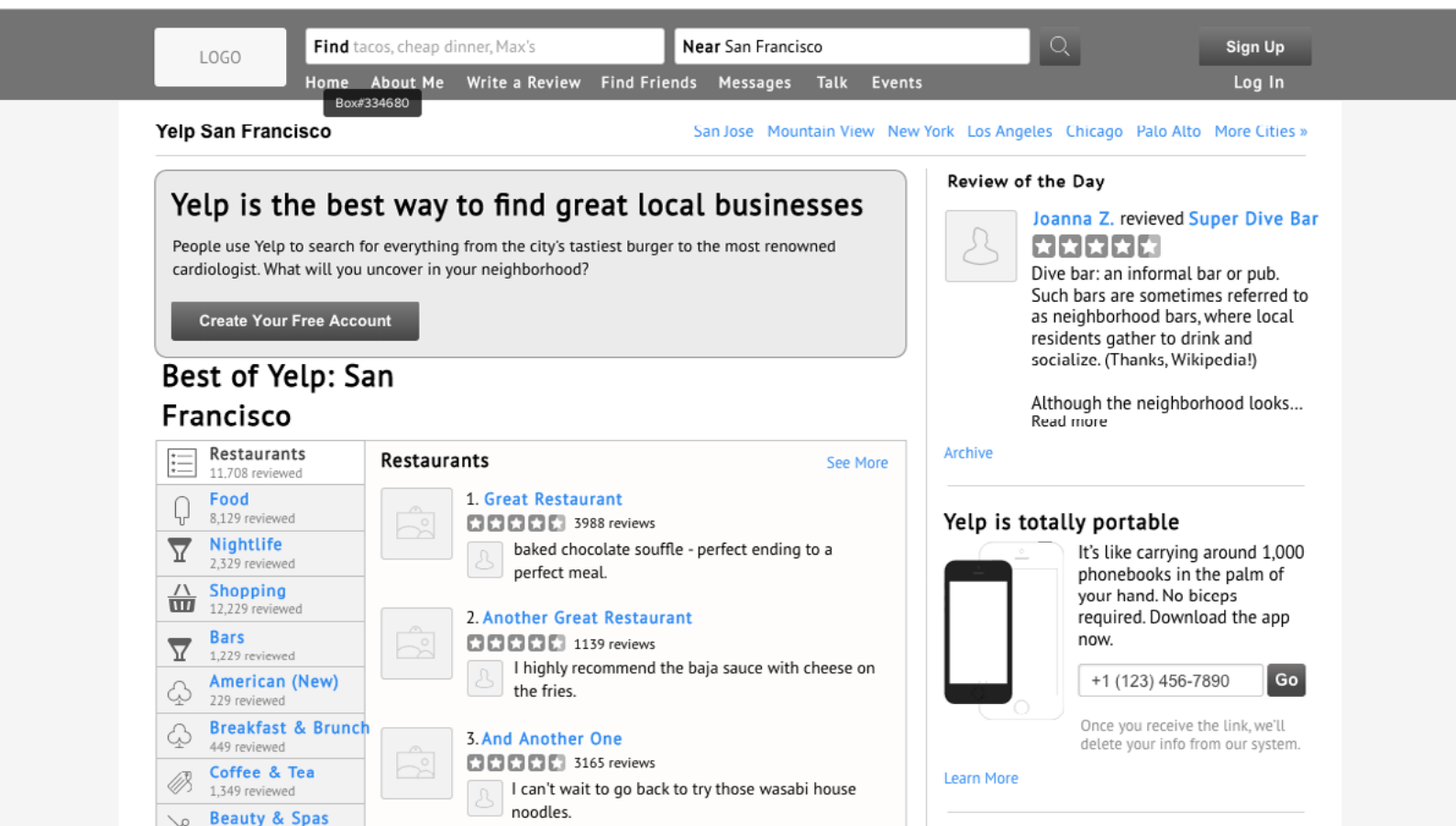
source: *User Testing & Design*

The purpose of a wireframe is to map out concretely for the entire team how the website should be designed. As described in the free *Guide to Wireframing*, the main goals of wireframing are documentation and design structure, but can also be shown to clients and stakeholders to get feedback while changes aren't

painstakingly difficult. To see how high-fidelity designs can start as wireframes, check out UXPorn and hover over each design to see the transformations.

> *"Wireframes are lo fidelity. Mockups are mid to hi fidelity."*

II. MOCKUPS

The main character of this e-book, the mockup is typically a mid- to high-fidelity representation of the product's appearance, and shows the basics of its functionality. Mockups fill in the visual details (such as colors, typography, etc.) and are usually static. By looking at a mockup, you should get a good idea of how the final product will look and a rough idea of how it might function (even if the functions aren't yet working). A mockup could be considered a high-profile visual design draft.
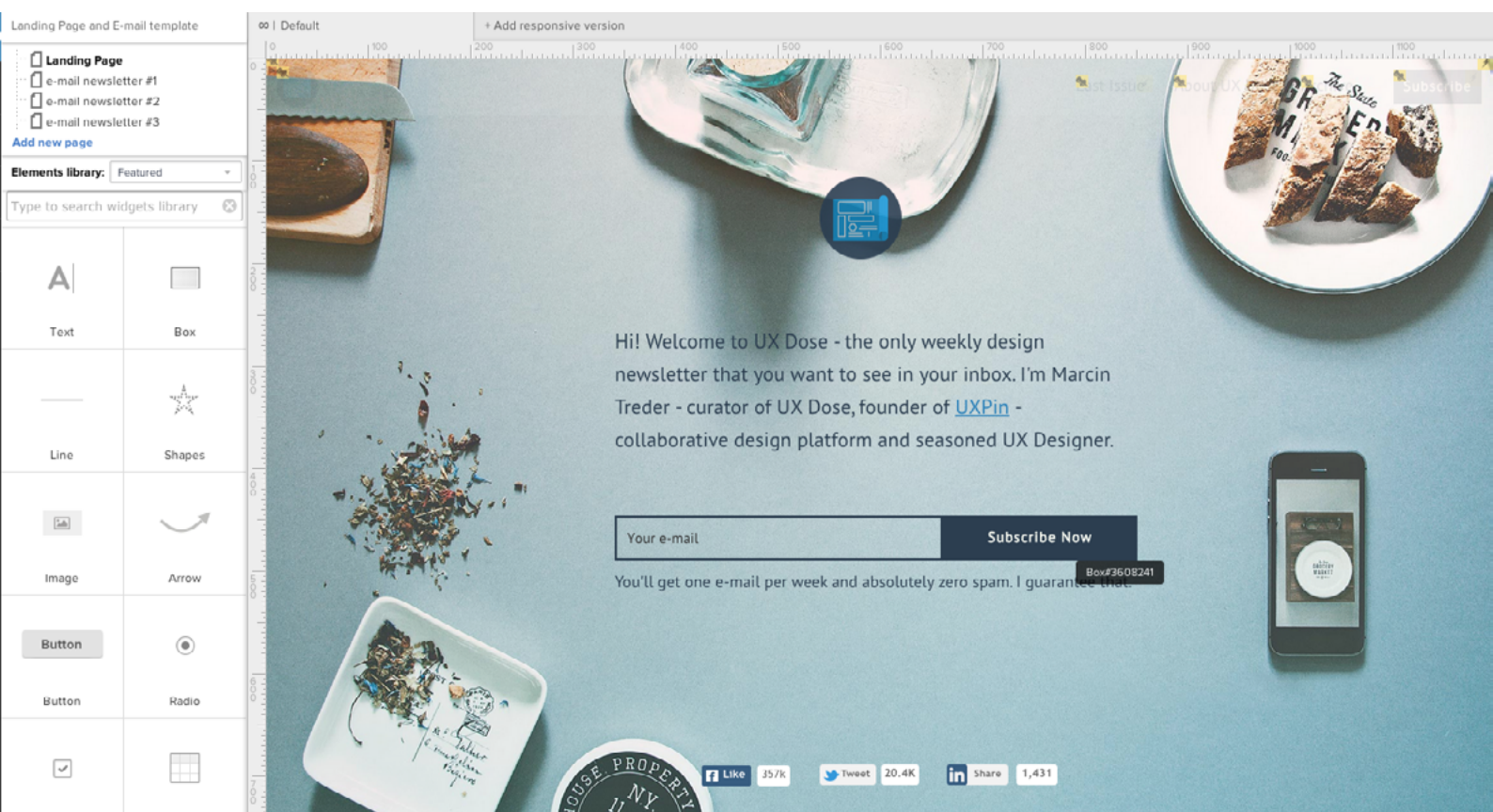
source: Mockup via UXPin

The wireframe lays the foundation and the mockup adds visual richness. While the mockup furthers the wireframe's purpose of documentation and organizing the team's vision, it does have an extra advantage the wireframe does not: with its superior visuals, the mockup is more impressive to stakeholders and investors, and so better at generating interest.

While the wireframe is visually stunted, the mockup is close to the final version in appearance, though it lacks the functionality to be advanced in the prototyping phase. In this way the mockup is, in essence, the bridge between the wireframe and the prototype.

*"Wireframes are skeleton. Mockups are skin."*   TWEET THIS

III. PROTOTYPE

The end of the beginning stages, the prototype can be a low- or high-fidelity representation of the product that includes functionality and the finer point of the UI design. In addition to the information structure and visualizations of the previous two phases, the prototype introduces more depth to the early UI, allowing users to:

- **experience actual content**
- **interact with the UI in a way similar to the final product**
- **predict and solve usability problems before further development**

When it comes to finding the right fidelity for prototypes, remember that going low-fi will let you test and tweak faster, while going hi-fi will get you as close to the final product as possible without sinking resources into development. A common design process is starting with a low-fi prototype (similar to what

Apple does by [creating hundreds of early prototypes](#)), and then iterating into high-fidelity prototypes. That way, you reap the benefits of [customer-driven design due to early testing](#) (as advised by **notable entrepreneur Andrew Chen**) as well as the [clear specifications demonstrated by high fidelity](#) (as explained by **SVPG Partner Marty Cagan**).
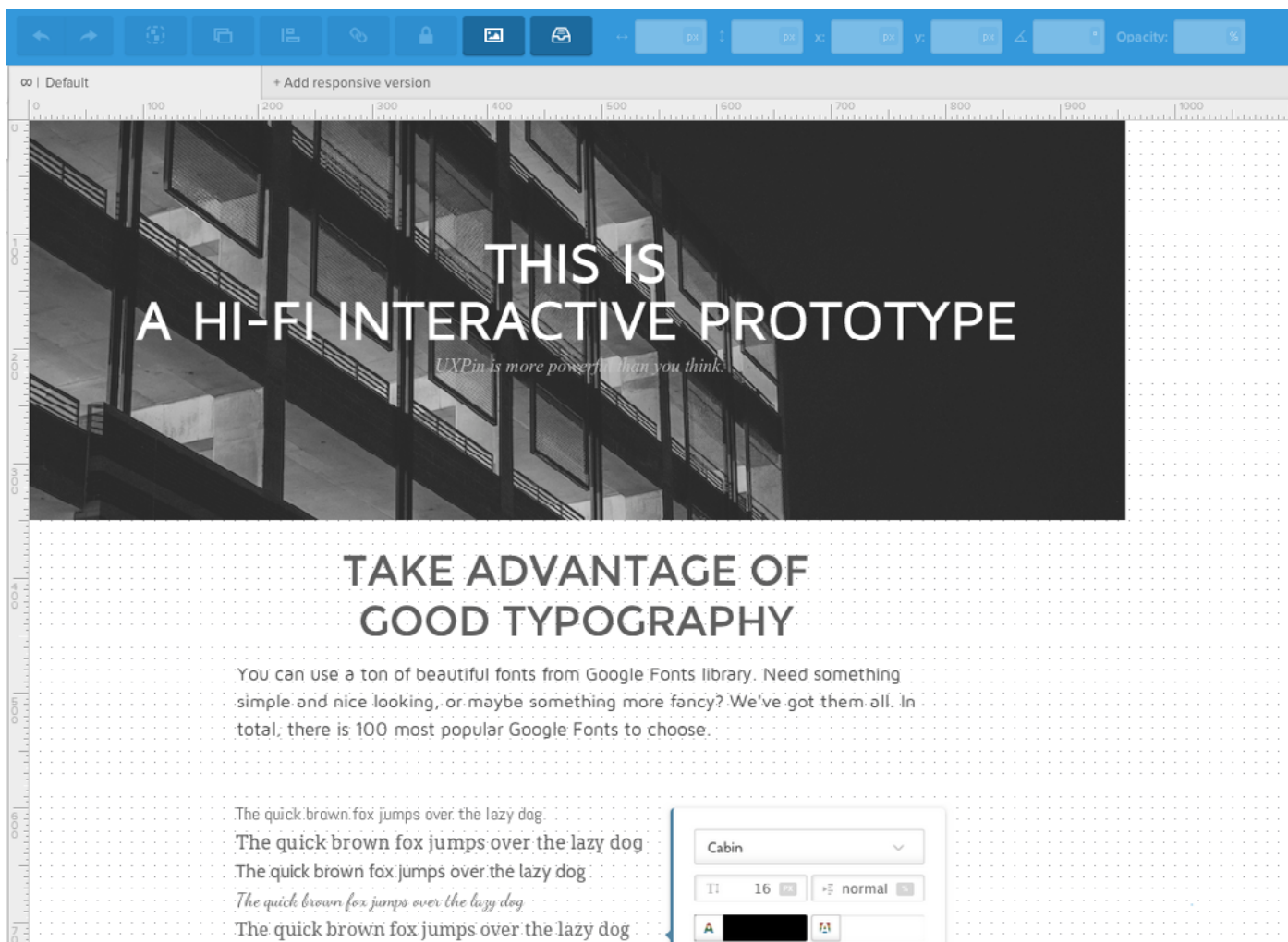
source: *[UXPin](#)*

A high-fidelity prototype should be one step below the final product; it should look and function as closely to the final version as possible.

"A mockup shows how it looks. A prototype shows how it works."

TWEET THIS

Why People Confuse Wireframes, Mockups, and Prototypes

While the distinctions of each phase are clearly defined, the truth is that even seasoned designers sometimes confuse the names of the different phases. Perhaps the most common mix-up comes with wireframes and mockups.

There's no doubt that wireframes and mockups are similar: they both are static phases of UI design, they both deal primarily with how the site looks, and they both don't require functionality. The main difference is in their quality. **Libby Fisher, freelance web developer and design blogger**, describes the mockup as the "decorated" version of a wireframe.

While wireframes are merely shapes and placeholders, mockups are designed to give the viewer an accurate impression of what the final product will look like. To help illustrate the differences, we'll examine a sample of each, taken from our *User Testing & Design e-book*, which chronicles our exercise of improving **Yelp** based on usability testing.
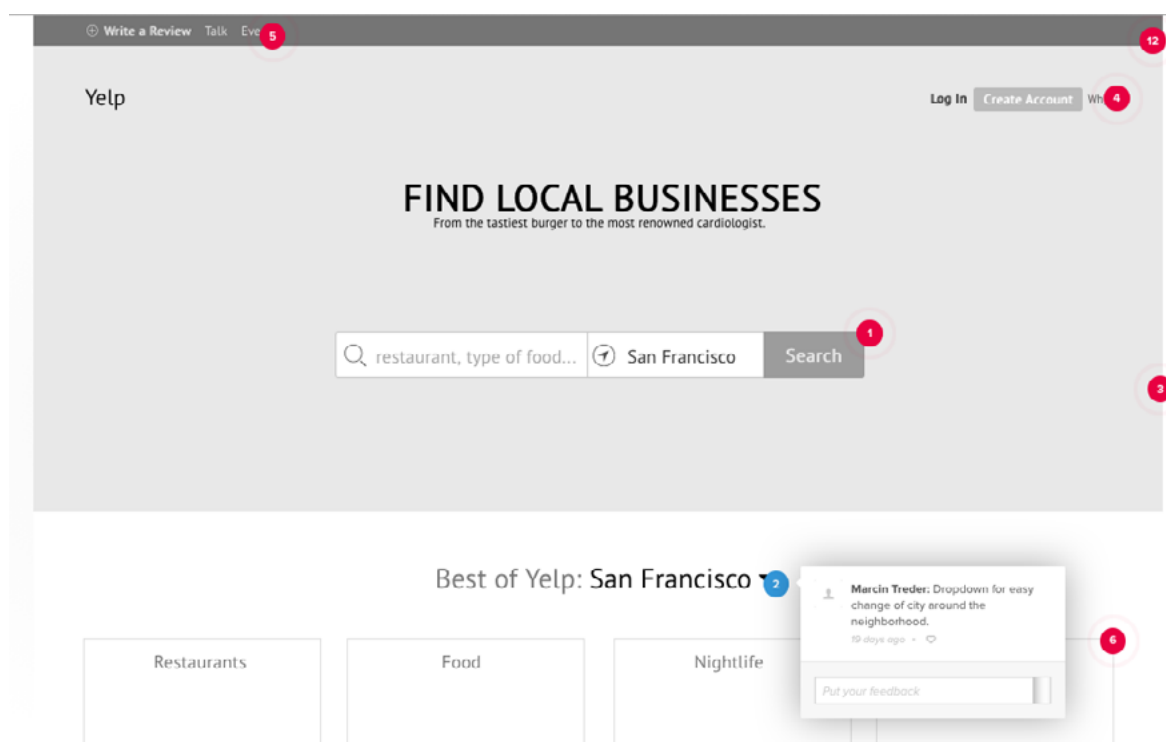

source: UXPin Low-Fidelity Yelp Design

We created the above wireframe as a roadmap for the improvements we wanted to make on Yelp's website. Compared with the current Yelp site, you can see how the wireframe is the skeleton of the new site with a larger Search bar and simplified categories.

Notice the low fidelity and lack of detail — flourishes aren't necessary at this stage. Think of wireframing as doing the heavy lifting, and save the finesse for mockups. Our goal was to first revise the structure of the site indicated by the qualitative insights and quantitative insights of the usability testing. The numbered red dots are our comments, explaining the changes so everyone is on the same page. This wireframe was later turned into a lo-fi prototype by adding interactions from our animations libraries.
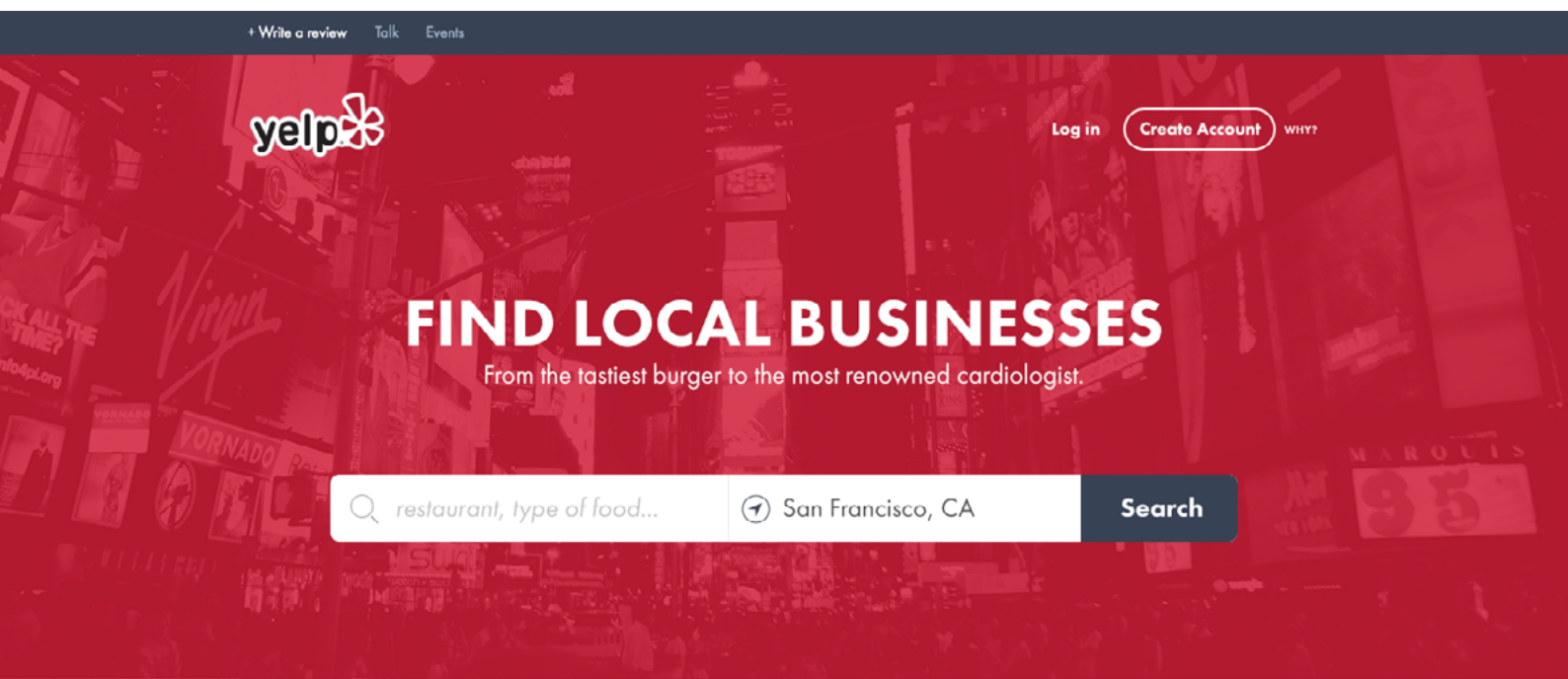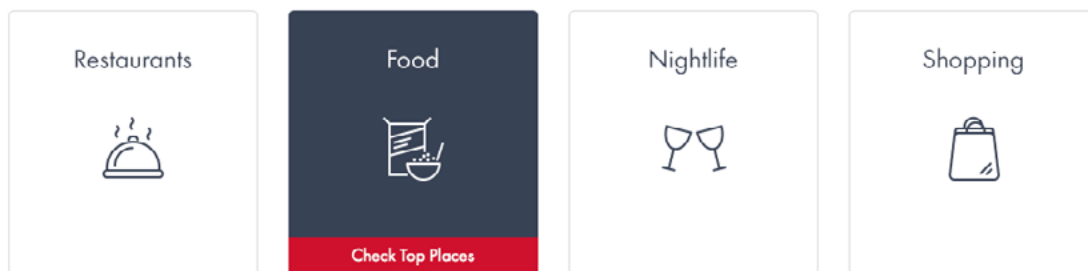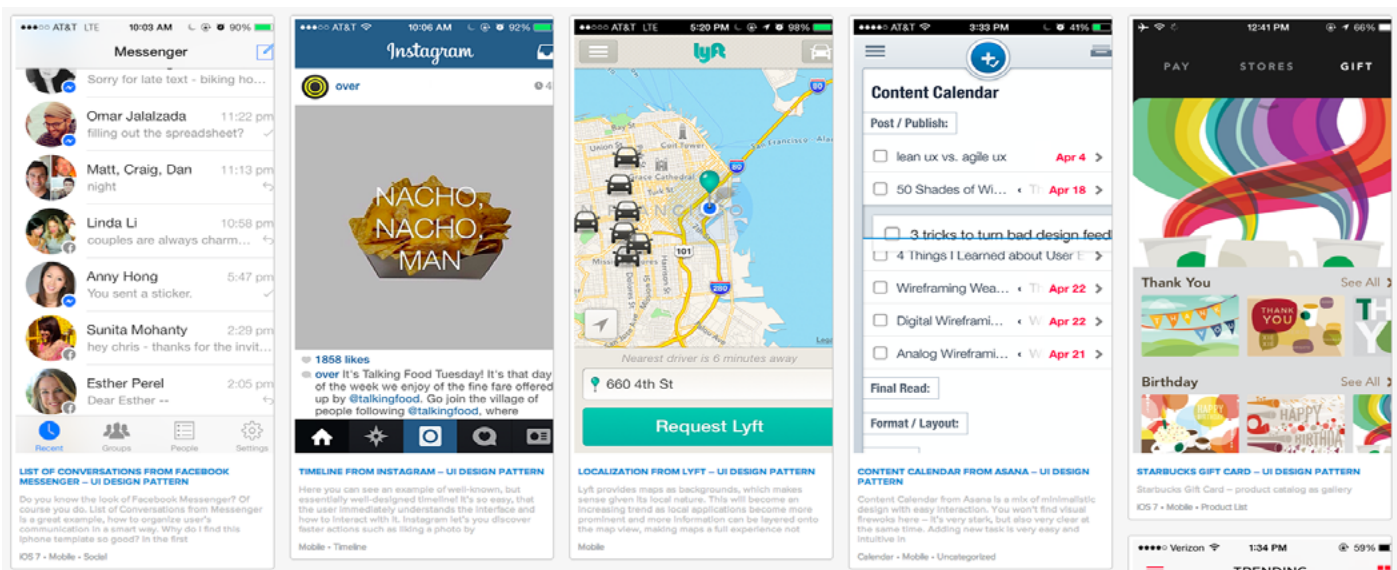

source: UXPin

Here is our mockup. The key difference, obviously, is the graphic detail — colors, icons, photos, etc. These visuals aren't just cosmetic, they also give an idea of how the functionality works. For example, even though it's not a prototype, you can alter a few elements to give stakeholders a taste of the interactive states (look at what we did to the "food" category to show what it looks like when clicked). Compare this mockup to the wireframe: the pieces are all the same and all in the same places, but the look and feel of the mockup is a lot more like the final product — and that is the fundamental difference between wireframes and mockups.

Don't Mock Mockups

While some designers hold the opinion that mockups aren't necessary (especially for the rapid prototyping of Lean UX), mockups are extremely helpful for exploring visual design decisions before you need to live with the consequences of code. Because mockups are a transitional phase between wireframes and prototypes, they are a sandbox for visual experimentation. There's a lot of details to keep track of in the UX design and development process, and overlooking the mockup phase can lead to subpar visuals.

source: UXPorn

Anthony Tseng, editor-in-chief for UX Movement, believes in [the power of both mockups and wireframes](#). In his opinion, design is "the synthesis of form and content." The wireframe outlines the basic structure of the content, while mockups present a vision of the form. Thus, a skilled designer will spend the appropriate amount of time perfecting both, and that means using wireframes and mockups.

The importance of mockups is worth elaborating upon. **Bima Arafah, freelance web designer and design author**, [explains why mockups are an essential part of the process](#). According to him, mockups are worthwhile for several reasons:

- **Organization of details** — Mockups can help reveal any clashing visual elements in a way that mirrors the final design. As discussed in our *[Web UI Best Design Practices](#)*, fine details such as color, contrast, and visual hierarchies should be determined in the mockup stage (where they can be easily changed) and solidified in the development stage.

- **Design implementation** — How does your initial design perform? From a usability perspective, a mockup lets you test the visual details and change them before it's committed to code.

- **Immersion in the user's perspective** — As you add detail to the wireframe (or maybe you jump right into a mockup), you are constantly looking at and altering a design that is closer to the final state. It's a subtle difference, but an important one, since a high-fidelity mindset helps you make design decisions from the user's point of view.

- **Flexibility** — Revisions that carry over from wireframing can stack up pretty quickly, but making them in a mockup is (comparatively) easy compared to CSS or HTML.

A lot of the criticism against mockups comes from the time and energy it takes to create something that eventually needs to be rebuilt in HTML or CSS. But with the availability of mockup tools, like Moqups for lower fidelity or UXPin for all types of fidelity, designers can create mockups faster with premade element libraries.

Context for Mockups

With this discussion of the benefits of mockups, it's necessary to mention that there are some pitfalls to watch out for, namely in how it opens the product up to unjust criticism. **Luke Wroblewski, product designer for Google**, explains how the presentation of a mockup will affect the feedback it receives. A mockup is static, but final products are rich and interactive. When showing a mockup to stakeholders, you're inherently at a disadvantage since the mockup currently exists in a vacuum. That's why you should frame the presentation in the context of what problem the product design is solving — otherwise stakeholders may request features simply because they're focusing on a tree and can't see the forest yet.
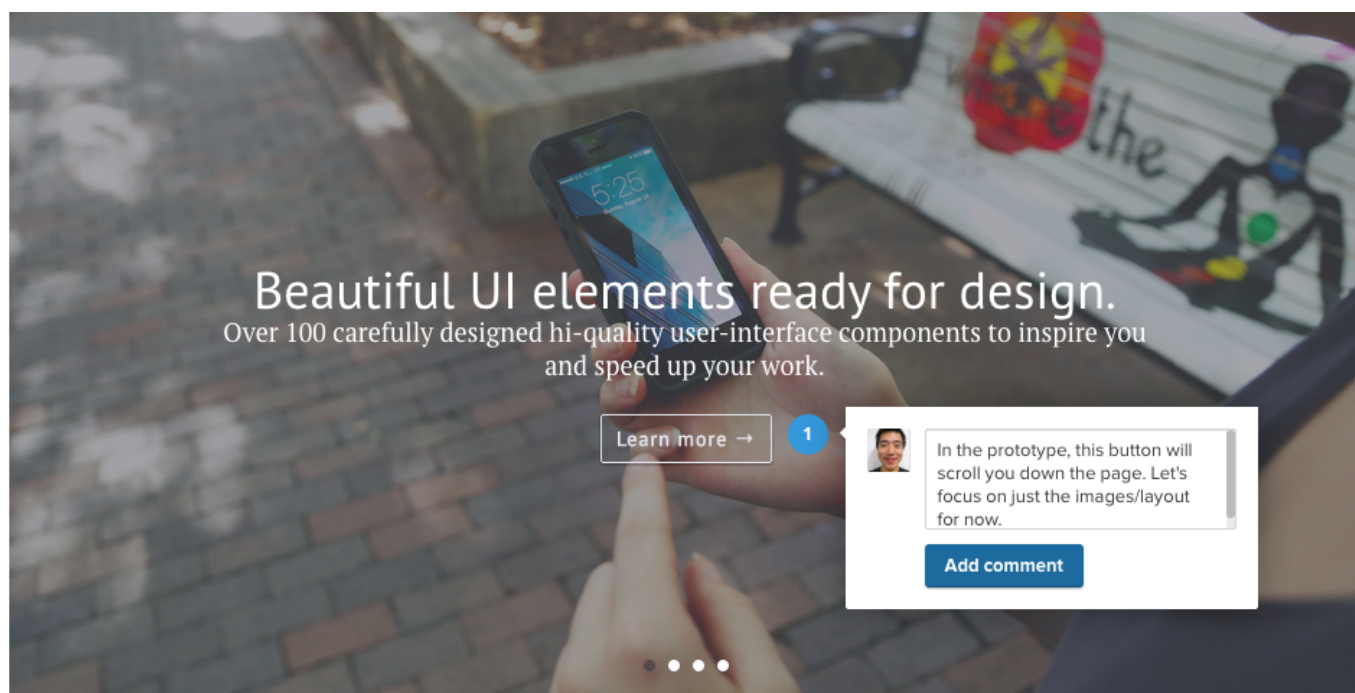
source: Collaboration via UXPin

Remember that mockups are still a design deliverable and just a means to the final design. This will guide the feedback more towards how it accomplishes its goal, and less towards criticism on the mockup itself.

You also don't want to let stakeholders have the final say in the mockup decisions, as this can lead to a dreaded [design by committee](), or at least diminishes the value of the designer in the eyes of the stakeholders. Find the right mockup tool for the right fidelity (mid or high), conduct usability testing to back up your decisions, and always provide context when presenting the mockup.

Takeaway

As the middle phase between the wireframe and the prototype, the mockup is a key component not just for the design team, but also for the stakeholders. It helps find and fix visual inconsistencies earlier in the process before they become too costly.

By now you understand what a mockup is (and is not), how it fits in to the development process, and why you shouldn't ignore it. Equally important, you also want to create context when presenting mockups since the design isn't final and the functionality can't be experienced.

In the next chapter we'll break down the anatomy and fidelity of mockups.

CHAPTER THREE

The Makeup of a Mockup

Dissecting the anatomy and the differences in mockup fidelity

W hat's a mockup really made of? While mockups can be created in differ-
ent ways to moldto the designer's stylistic vision, there are a few com-
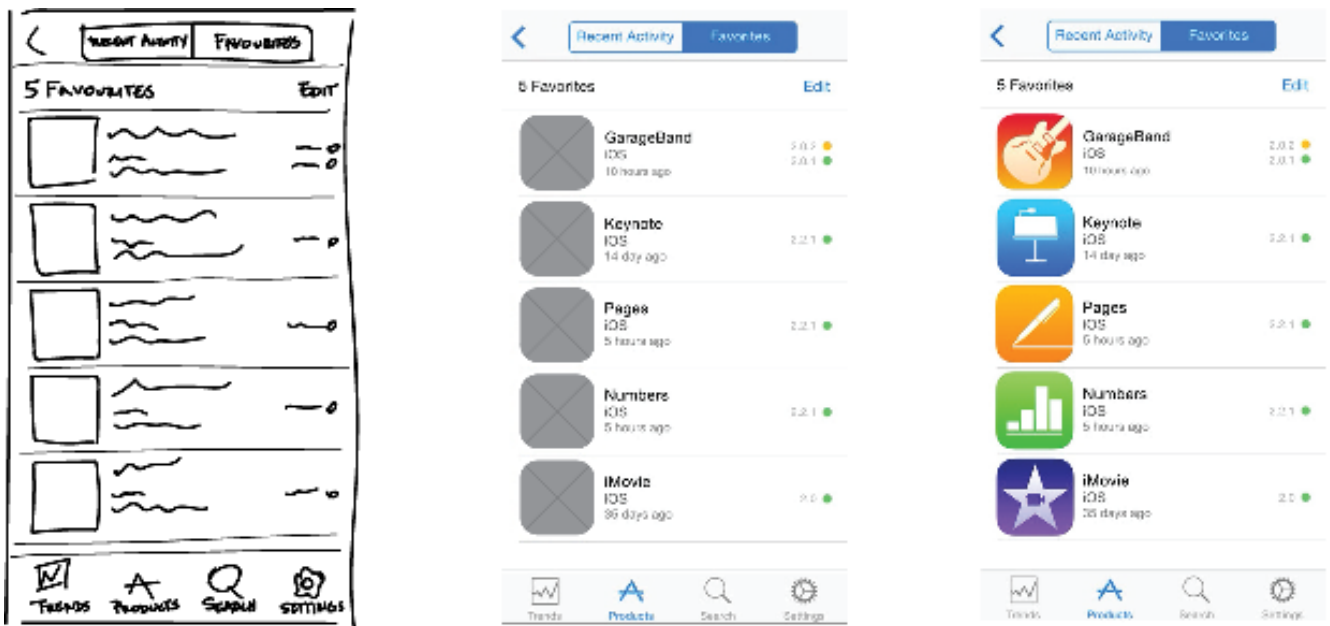mon components you don't want to neglect.

Photo credit: uxpin.com via http://www.jfarny.com/imagesNew/inrix_progression.jpg

In this chapter, we'll explain piece by piece what to include in your mockup,
and then discuss the differences in fidelity so that you can choose the option
that works best for your needs.

The Anatomy of a Mockup

What should be included in a mockup? The answer is simple: the same visual
details you'd normally include in a website or app.

As a visual representation of your product, a mockup should look the part.
While sacrificing interactivity, a mockup's purpose is to show clearly what the
final product will look like, so this requires the inclusion of the elements of
that final product. And because mockups are so centered around appearance,
you should apply the same rules for aesthetics that you would for the finished
product.

Moreover, a mockup should hint at usability and function — while not necessarily utilizing it (that's what prototypes are for). Including the proper buttons, color changes, and page flows will give stakeholders an impression of how the final product can work.

When planning your own mockup, be sure to take into account visual hierarchy, navigation, and any applicable UI patterns.

I. VISUAL HIERARCHY

Visual hierarchy is an essential element for website UI design, and so it is also essential for mockups, as they share the same visual significance. We discussed visual hierarchy in detail in our e-book *Web UI Design Best Practices - UI Design from the Experts*, but we'll briefly reiterate what's most relevant for mockups.
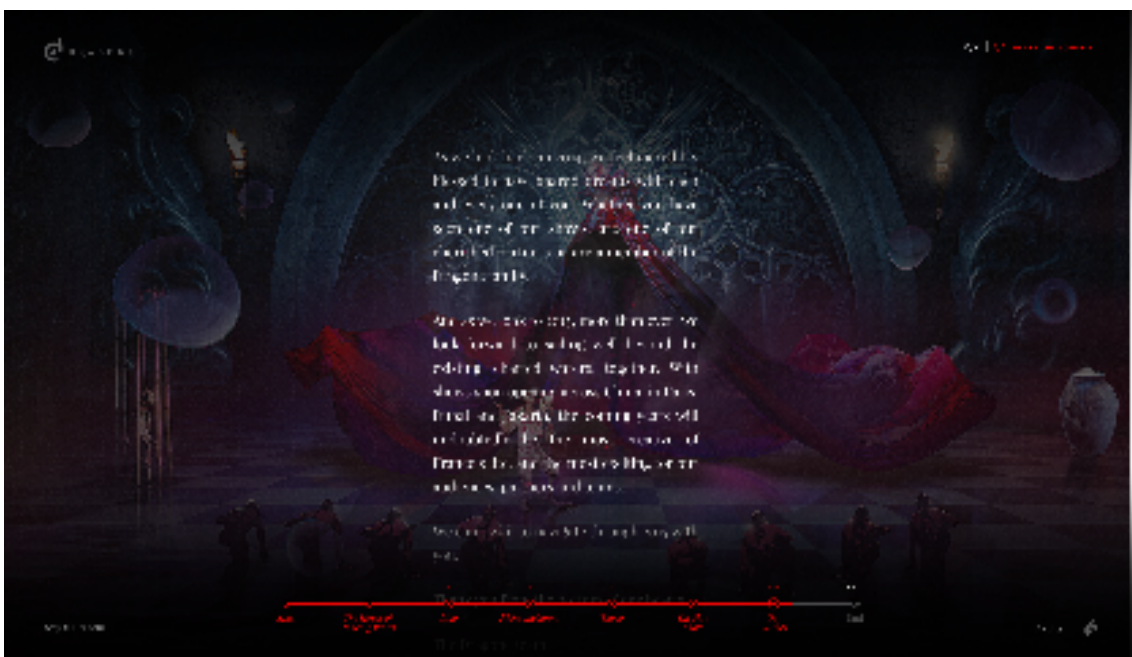

Photo credit: http://happy2015.dragone.com/ via awwwards.com

As far as the basics go, it's important to remember these elements:

- **Layouts** — Modern advancements in eye tracking have pointed out two main patterns for website viewing, the F-Pattern (for text-heavy content) and the Z-Pattern (for general content). By predicting where your users' eyes will go, you can plan out your site's layout with greater control. **Steven Bradley of Vanseo Design** explains how to best use these layouts.

- **Contrast** — Using both light/dark and warm/cold color contrast will make your mockup more visually stimulating, and when done properly can help you guide your users' eyes where you want, for example, to a call-to-action. **Dmitry Fadeyev, founder of Usaura**, gives more details on contrast in UI design.

- **Color** — With their suggestive moods and capability to attract/repel your users' sight, colors are a powerful tool for any designer — especially the overall color scheme, which will determine the site's personality and have an enormous impact on the UX. For the basics on the use of colors in web design, **Thomas Cannon of tuts+** wrote an excellent overview of color theory.

- **Typography** — Considering that the content of most websites is the written word, typography plays a large role in the overall user experience. Using the right sizes and fonts are almost as important as the writing itself. **Mark Boulton** describes some helpful tips for optimizing your typography.

- **Spacing** — Don't overload your page with too many visuals, no matter how good they are. Your elements need breathing room to reach their full potential, so a good designer knows how to strike a balance between too much and too little.

There's a lot to say on the importance of visuals in web design — in fact, entire books have been written on the subject. These are just the basic elements that are most applicable to constructing a mockup, but if these seem unfamiliar to you, more research into the topic might be helpful.

"Don't overload your page with too many visuals. Breathing room emphasizes what's important."

TWEET THIS

II. NAVIGATION

Whether you go mid or high fidelity, you need to show what the navigation will look like — it is, after all, the information pathway for your website or app.
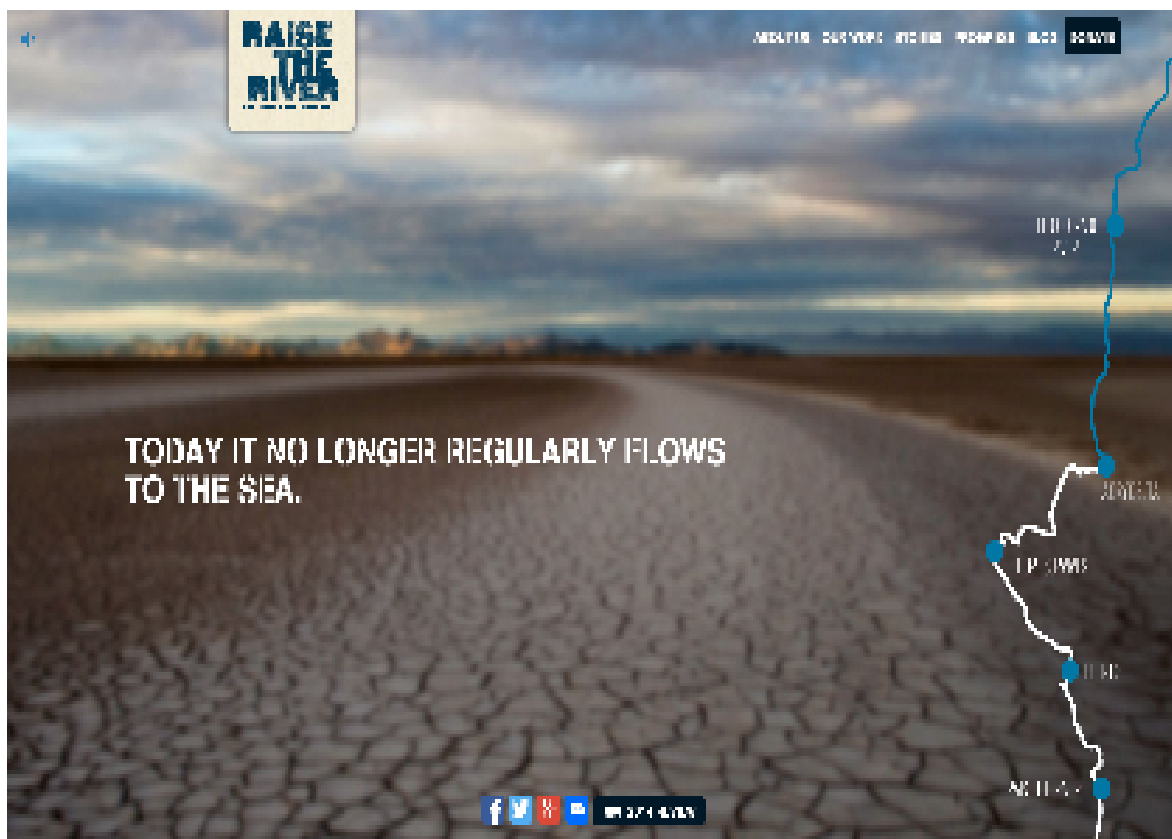

Photo credit: *http://raisetheriver.org/* via awwwards.com

Richard Rose of Fresh Consulting emphasizes that the navigation must reveal content, show clear placement of action, and prioritize clarity over visual flair. Site navigation breaks down to four main categories:

- **Menus** — The most basic form of navigation for users, this is typically where they will look first when trying to get around. **Vandelay Design** shows [40 of the best menus](#) to inspire your mockups.

- **Breadcrumbs** — If a user enters your site on a page that's not the homepage, a breadcrumb trail will provide the necessary reference points to orient themselves. *Smashing Magazine* explains the [three types of breadcrumbs, and how and when to use them](#).

- **Filters** — Content-heavy sites have more navigation needs than others, and so filters help shoulder some of the extra responsibility.

- **Links** — Links are generally helpful for connecting related but otherwise unconnected content.

If you're looking for examples of common navigation patterns, or perhaps inspiration for creating one yourself, [this UI pattern library collects some of the best](#).

III. UI PATTERNS

A lot can be said about UI patterns, but simply put, they are design best practices used by successful sites and apps to solve common UI problems. Most designers approach them as starting points or inspiration, since they're not designed to be drop-in templates (if they were, websites and apps would all be cookie-cutter).

Photo credit: *Patterntap.com*

In *Web UI Design Patterns 2014*, we go into detail about finding and applying UI patterns, and examine 63 cases from sites like **Amazon**, **Facebook**, and **Pinterest**, and others. UI design patterns usually fall within four categories that apply to all websites and apps:

- **Getting Input** — Patterns for receiving input, like form fields and controls, are the backbone of interactivity.

- **Navigation** — As mentioned above, patterns like menus and breadcrumbs allow users to orient themselves and explore with confidence.

- **Data & Content Structuring** — These are the patterns that organize and display your content, infusing your product with your own personality and style.

- **Social Sharing** — Everyone wants their product to be shared on social media, and these patterns can encourage this.

UI patterns are also effective ways to defend design decisions — when presenting mockups, it can be quite convincing to show popular sites that use similar design patterns. Your mockup won't look the same as those sites, but you'll have visual proof of the rationale behind the layout.

"UI patterns help defend your design decisions."

There are lots of patterns to keep track of, with more new ones popping up each day. Even veteran designers look to pattern libraries like UI Patterns, Pattern Tap, and UI kits (like our own) for quick reference and inspiration.

Mockup Fidelity

The best way to differentiate the types of mockups is in the fidelity. **Designer Brad Frost** best summarizes the progression of fidelity with an analogy to sculpture by subtraction:

> *"You start out with a big slab of rock, and slowly chip away to get the rough shape of the form you're creating. You take another pass at it to get a better sense of the object you're trying to extract from the stone. You take another pass to start getting a bit more detailed. Eventually, you start honing in on a particular aspect of the form: a face, the arms, or the torso. Slowly but surely you detail each section of the sculpture until you've arrived at the final form."*

In a piece for UX Booth, **Tyler Tate, UX leader for Nutshell and TwigKit**, says that design methods "exist on a continuum of fidelity," beginning with lo-fi sketches and ending with hi-fi HTML prototypes. This, then, places mockups at mid- to hi-fidelity, wedged between lo-fi wireframes and hi-fi prototypes.

To apply too much fidelity prematurely in the process will waste time and money, whereas not applying enough will leave some avenues not fully explored and puts the wrong thoughts in stakeholder's minds ("This page looks a little bare, I better tell the designer what to do!"). So, the optimal level of fidelity should be only what is required to get the job done in the least amount of time — what Tate calls "being lazy in a good way."
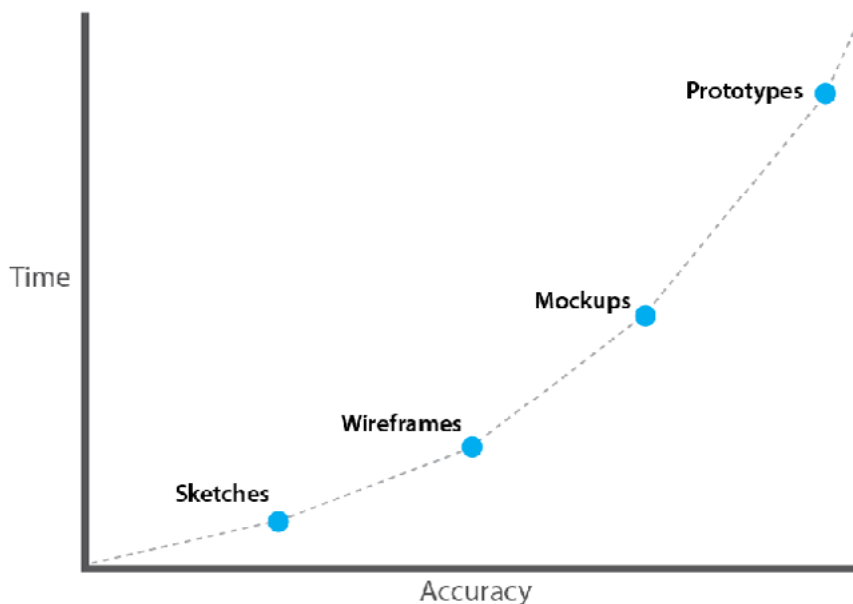

Photo credit: uxpin.com via www.uxbooth.com

I. On Paper: The Concept Mockup

Lo-fi mockups, called "concept mockups," exist outside the traditionally accepted form of mockups, but can be useful nonetheless. In the *Guide to UX Design Process and Documentation*, we explain that some companies skip the wireframing phase and replace it with a concept mockup, which they then improve until it becomes a high-fidelity mockup.

As you can see in this diagram, a concept mockup is basically a sketch with

more refinement and slightly more fidelity than a lo-fi wireframe. You can see in the drawing below that the concept mockup fills in some detail (like rough human figures), which wouldn't be required in a wireframe (where a large X across the images might suffice), but helps get the designer thinking towards higher fidelity.

Photo credit: Sketch. Jeremy Keith. Creative Commons.

Concept mockups are especially helpful if you prefer starting on paper but plan on digitally creating your mid- or hi-fidelity mockup. It takes a little more time to sketch, but lets you cover more visual ground — it's one thing to tell stakeholders that a blank box represents an image, and it's another to fill in a few figures to show them. And if you plan on wireframing before the mid- or hi-fidelity mockup, you can still take a photo of your concept mockup and then cut-and-paste the parts you'd like to insert to show more detail.

II. IN THE DIGITAL SPACE: HIGH-FIDELITY VS. MID-FIDELITY

MOCKUPS

There are really two schools of thought that divide how to create a digital mockup. Those that believe the mockup should represent the end product exactly (high fidelity), and those that see the mockup phase as more transitional and should not take too much time (mid fidelity).

In an article for The Next Web, **Joel Falconer, features editor for TNW**, explains the pros and cons for each style. The hi-fi mockup (which can be considered pixel-perfect), plans out many of the design decisions right at the start, including font size, color scheme, margin sizes, etc. You can see the level of detail we include in one of our own landing pages below (color, typography, background image, etc.).

source: High-fidelity Mockup via UXPin

These realistic designs are beneficial for several reasons:

- **Make the detailed decisions sooner** — You give yourself more time to iterate and perfect the visual design when you start exploring textures, gradients, colors, and fonts before there's pressure to code anything.

- **Easier to present to non-designers** — Clients and stakeholders prefer this style because what they see is what they'll get. Anyone can understand high-fidelity.

- **More accurate for developers** — The developer can see what the final product should look like, which means the high-fidelity mockup can serve as a visual specifications sheet. A hi-fi mockup nicely complements the text-heavy functional specifications document described in *The Guide to UX Design Process & Documentation*.

However, there is a downside, and one that shouldn't be overlooked: hi-fi mockups take longer. **Sergio Nouvel, UX Lead at Continuum**, believes that going hi-fidelity isn't time-efficient since everything needs to be rebuilt anyways. One could make the argument that high-fidelity mockups save time in the end through better organization and planning, but someone with limited time or resources might not want to take the risk.

The second school of thought, the mid-level fidelity mockup, also has its distinct advantages:

- You go from the conceptual phase to the implementation phase more quickly.

- It allows for more flexibility in the implementation phase without sacrificing the guidance of the mockup outline.

- It reduces the risk of unnecessary designs that need to be scrapped when

beginning from HTML and CSS.

source: Mid-fidelity Mockup via [UXPin](#)

Yet this, too has its drawbacks. The lack of finality means that more decisions are up in the air, so a lot of that time you saved during the mockup phase will have to go into fine-tuning these later on. Additionally, clients and stakeholders won't be as impressed with a mid-fi mockup as they would a hi-fi one, so that's worth keeping in mind (especially since it takes only around [50 milliseconds for a website to make an impression](#)).

> *"It's not about hi or lo fidelity. It's about the right fidelity for the job."*   TWEET THIS

The choice of how much fidelity or how much time to spend making your mockup is up to you. Consider your specific needs and restrictions to gauge just how much quality you want to put in at this phase. And keep in mind Tate's advice: use the minimum level of fidelity required to get the job done — just think deeply on what "getting the job done" means for your project.

Takeaway

Mockups will be as varied as the products they represent, so there's not really a standard for how they should look or function, only guidelines. You'll want to incorporate the same visual principles as your final web design, as this is the part of the process conducive to appearance, and you don't want to neglect essentials like navigation or UI patterns either. When choosing how much fidelity to make your mockup, consider first your individual needs, then examine how you want mockups to fit into the process as a whole.
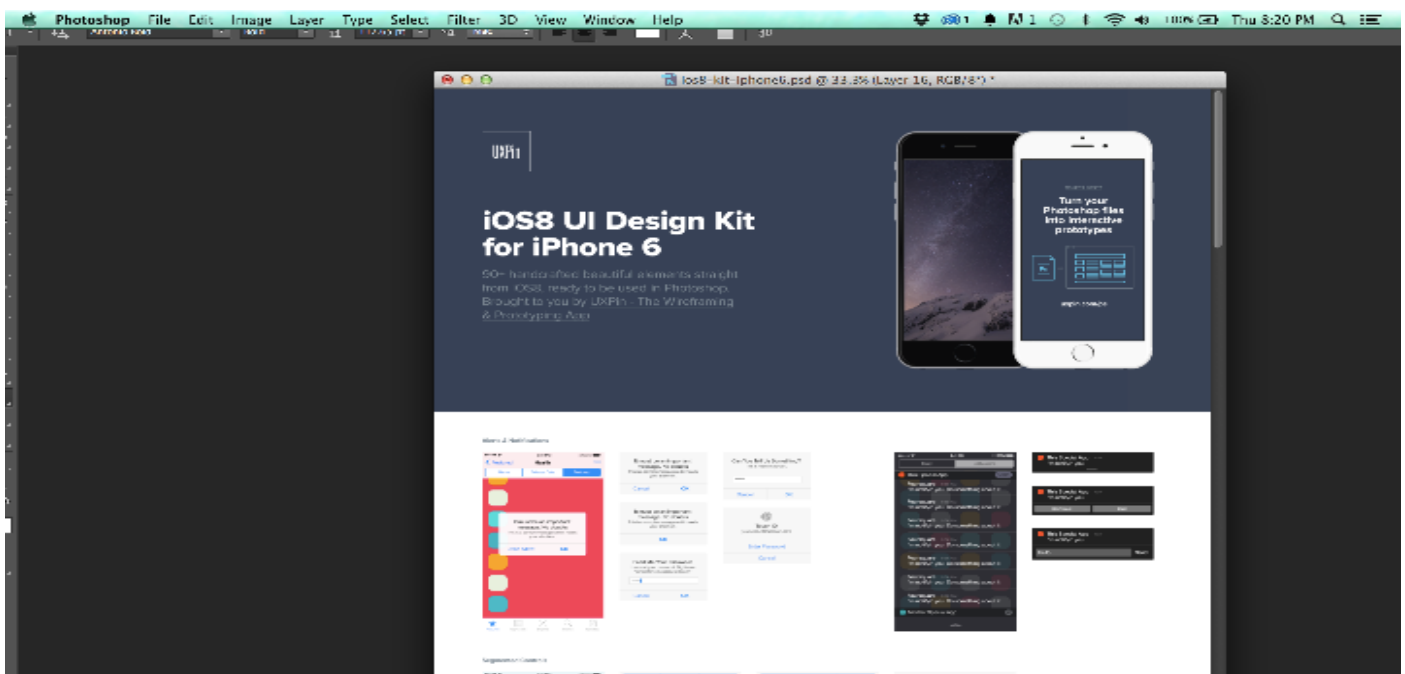

Photo credit: http://www.uxpin.com/ios8-ui-kit.html

Now that you have a solid understanding of what mockups are and how they're made, in the next chapter we'll talk about the pros and cons of three separate methods to creating mockups.
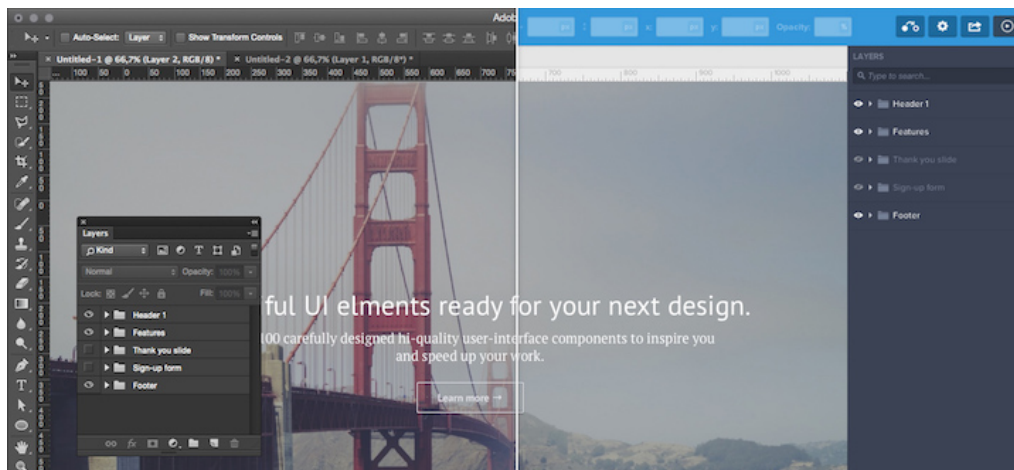
CHAPTER FOUR

3 Methods for Creating Mockups

Where and how to build your mockup

There's lots of ways to support a mockup. It's true there is no "best" way, but depending on certain UI and UX designer's styles and preferences (and the design process), some will work better than others.


source: [Turning Photoshop Mockups Into Interactive Prototypes](#)

In this chapter, we'll look at the pros and cons of mockup tools, graphic design tools, as well as coded mockups that start to blur the lines with prototyping.

Using Mockup Tools

Using a tool like our own [UXPin](#), or other solutions like Moqups or Balsamiq, will provide you everything you need to build your mockup and facilitate the entire process. These tools are designed to make the creation process as easy as possible, so you can focus more on stylistic decisions and less on how to manipulate the program. Both experts and beginners feel most comfortable with mockup tools, as beginners prefer their ease of use, while experts appreciate the designs specifically tailored to their advanced needs.
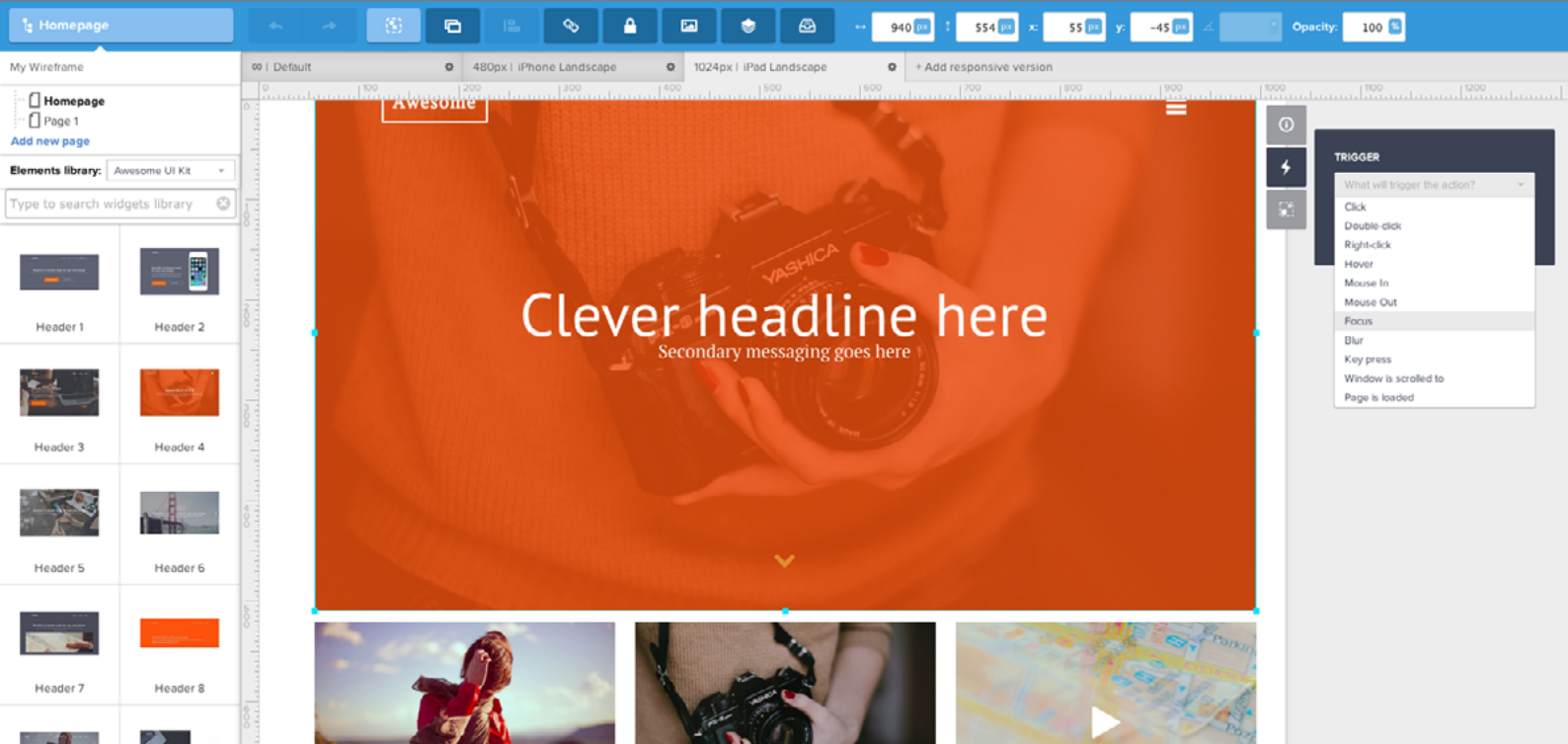
source: *Creating Quick Wireframes & Mockups*

Moqups and Balsamiq provide more functionality than non-design tools that are sometimes used for wireframes and mockups (such as Keynote), but they are limited to only low fidelity designs. They can, however, be quite useful if the goal is to create low-fidelity wireframes very quickly. When it comes to mockup tools, many are targeted for wireframing than true mockups. We'll discuss the specific tools' strengths and weaknesses in the next chapter.

With built-in collaboration, UXPin offers dozens of element libraries to save time, and its a simple matter of dragging, dropping, and customizing to create your mockup. Our 20+ element libraries cover both web and mobile.

Using Graphic Design Software for Mockups

Some designers swear by software like Photoshop, Sketch, or Illustrator, especially those particularly skilled or familiar with tools that offer control down to the pixel. As **Nick Pettit of Treehouse** points out in a piece explaining mockup types, graphic design platforms work best if you're aiming for the highest level of realism and visual fidelity.
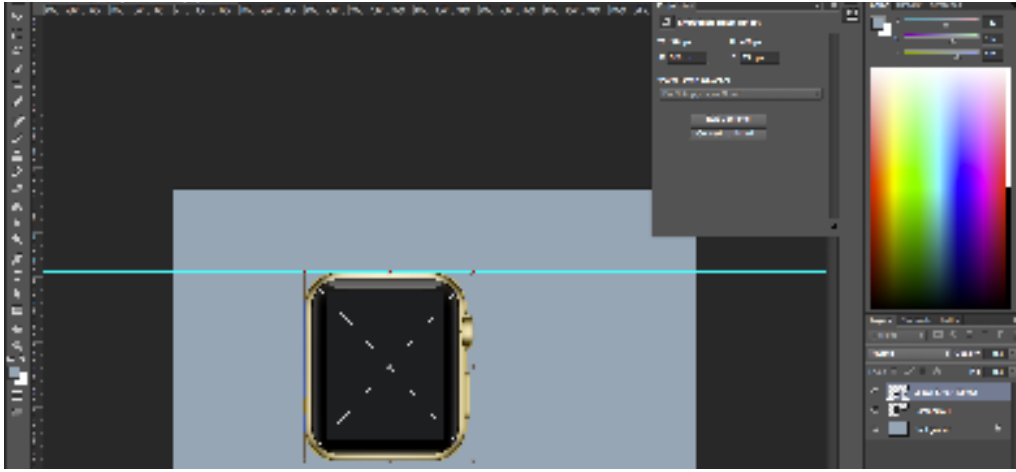
Photo credit: uxpin.com

Working in graphic design software gives you access to an almost endless selection of highly defined colors, so if you're working within the restrictions of a rigid and preset color scheme — for example, under particular branding rules — then these programs may be your best option. More than color options, these programs offer far more visual tools, allowing you to tackle the minutiae of detail.

But the drawback to using this type of software is that it can be difficult to translate when it comes time to start coding the design. What worked in Photoshop (elements like fonts, shadows, gradient effects, etc.) may not work in code, which can waste time in figuring out solutions for the prototyping phase.
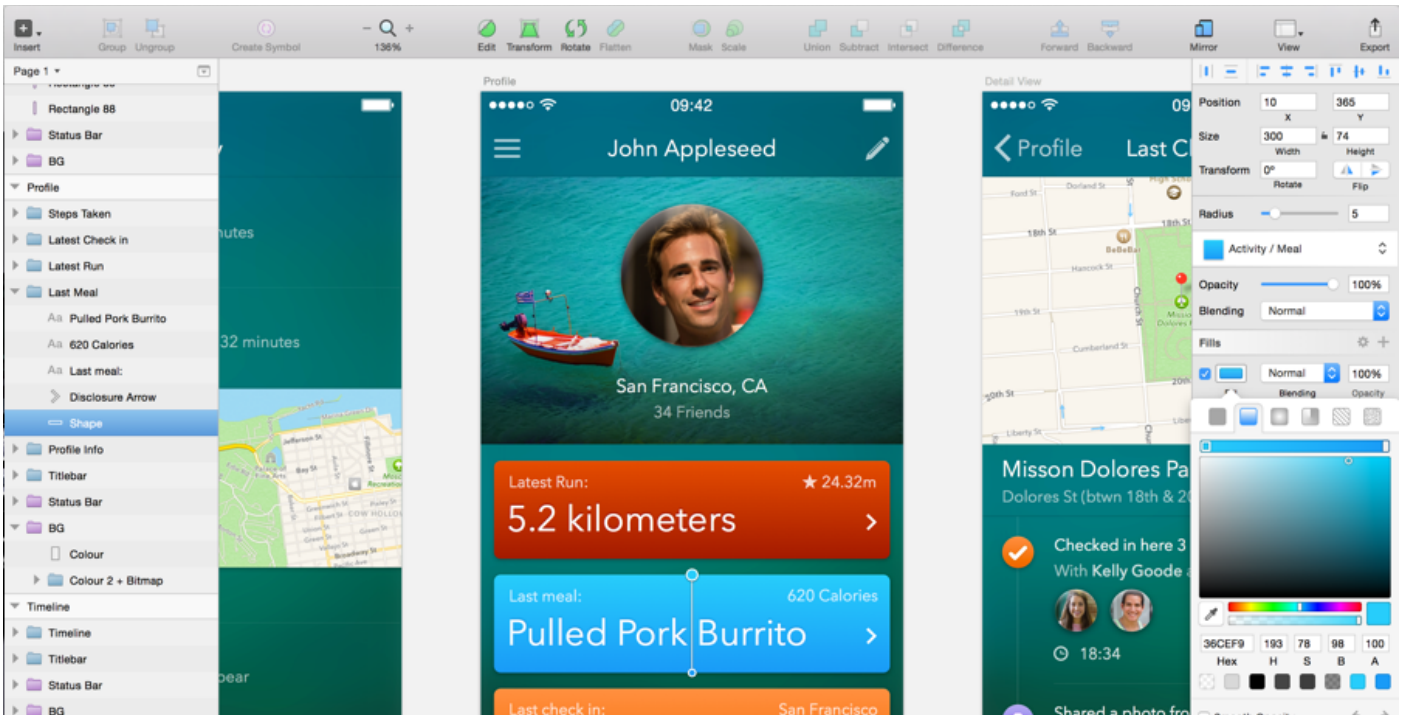
source: *Sketch App*

For this reason, graphic design software is only recommended if high fidelity visuals are the top priority. If you have a style-heavy page, it might help to hammer out the specific visual details in the mockup process (in which case Photoshop or Sketch will give you the most options). Just take a look at **Hubspot's** list of the 53 Most Beautiful Homepages, and it's easy to understand why it sometimes helps to sort out all the visual details sooner rather than later. Similarly, if you're dealing with a nitpicky or hard-to-please client, presenting them with a gorgeous and impressive mockup might win them over more easily.

It's also worth mentioning that mockups created in Photoshop or Sketch can be dragged and dropped into the prototyping phase with UXPin. This lets you easily animate all layers (no flattening) with a few clicks, and ensures you don't need to "start from scratch" when it's time to prototype. For more details, take a look at the tutorials for Photoshop integration and Sketch integration.
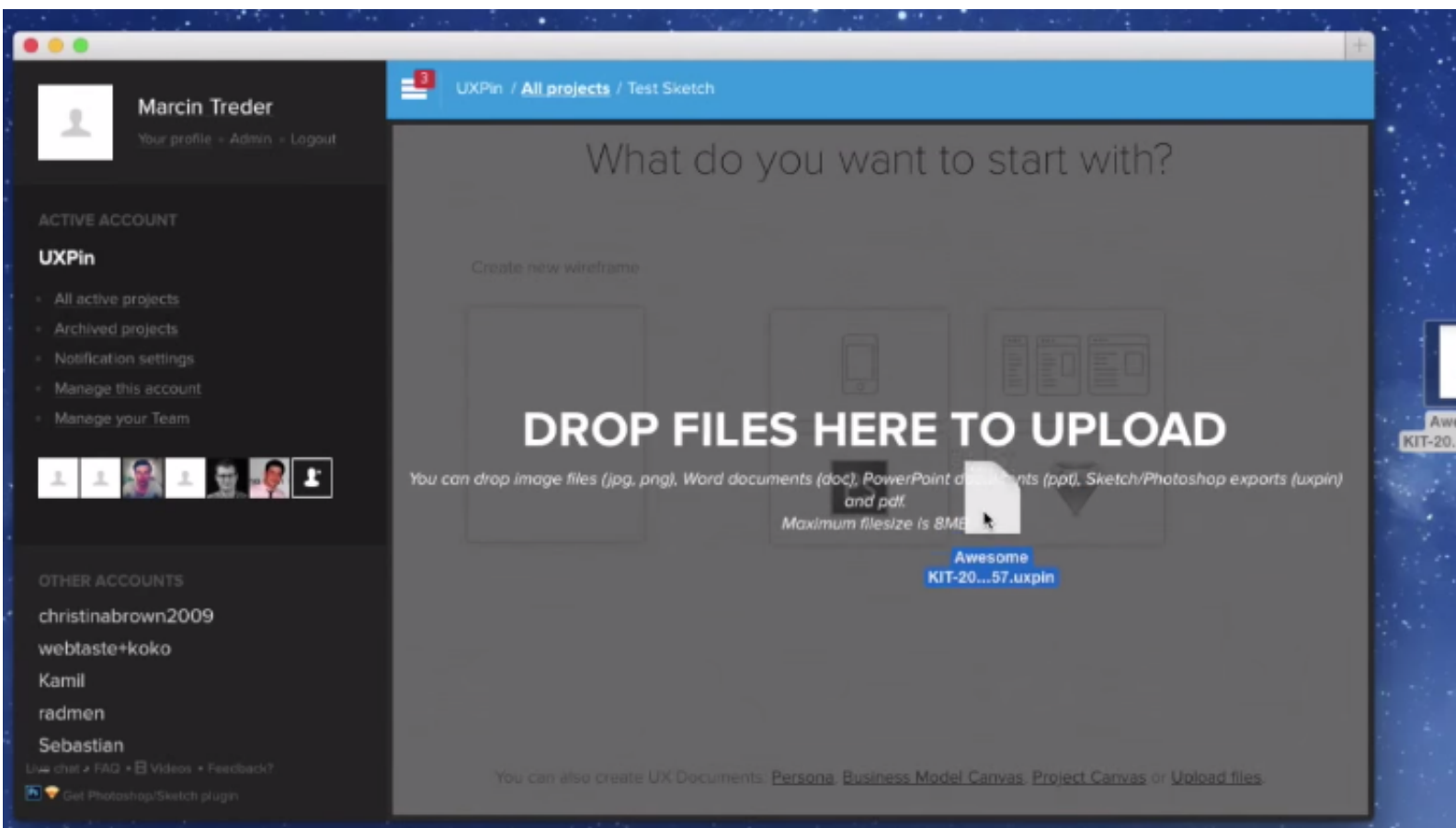
source: *Turning Sketch Mockups into Interactive Prototypes*

If visuals are *not* your only priority, you might be more efficient using a tool that allows you to do the wireframing, mockups, and prototyping all in one place. Graphic design software can be more trouble than its worth for mockups unless you're looking for optimum visualization — you'll definitely need to communicate regularly with your developer since these tools aren't designed for collaboration.

"Developers aren't WYSIWYG monkeys. Don't just hand them a mockup and wish them luck"

TWEET THIS

Coding Your Mockups

For starters, if you're mainly a designer and not comfortable with coding, then this obviously isn't even an option. The coded mockups are not the default choice — most coding can be postponed until prototyping (if you're creating an HTML/Javascript prototype) or even later (if you use a prototyping tool). But despite the complexity and potential obstacles, there are many respectable designers who advocate introducing coding into the mockup phase.
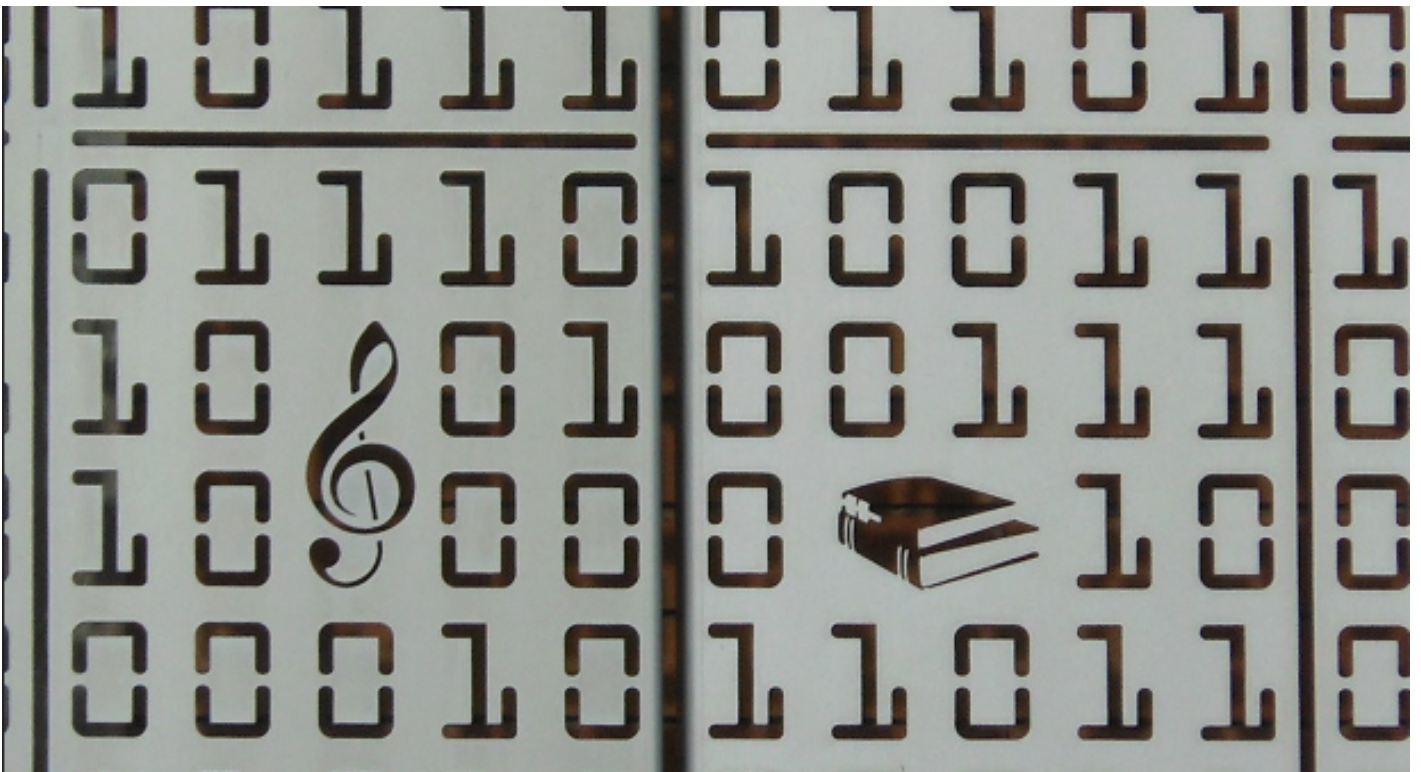
Photo credit: „Binary". Michael Coghlan. Creative Commons.

As **Joel Falconer** [shows us on TNW](#), building a structure in HTML and then moving to CSS for the basic layout can streamline the entire development process — in theory at least; many would argue that focusing only on visuals during the mockup phase saves time in decision-making and organization down

the road. But those who prefer coding early understand that more mathematical changes such as sizing can be implemented more easily when the code is already written.

One of the biggest proponents of coded mockups is **Ash Maurya, Founder/ CEO of Spark59 and speaker on development**. [When describing how he prefers to create mockups](), Maurya defends the HTML/CSS route, and makes some solid points:

- **Feasibility** — "Flashy" mockups (sometimes literally if they're done in Flash) designed only to look good can be difficult to translate into code, resulting in wasted time and effort. Elements like gradients, fonts, and effects — while easy to create in other tools — can be cumbersome or even impossible to recreate when coding. Starting in code lets you know right away what you can and cannot do.

- **Quick Iteration** — According to Maurya, coding actually saves time by simplifying the iteration process, although [others disagree in that it adds time in dragging out design decisions]().

- **Minimizes Waste** — "Creating a mockup in anything other than the final technology in which the product is delivered creates waste." Because the mockup is going to end up in HTML/CSS anyway, Maurya suggests adding fidelity there.

But as we mentioned before, mockups with coding are not a popular strategy, for more reasons than the difficulty of coding. **Tony Thomas, Lead Designer at Medialoot** [explains why in a compelling post on the subject](). Among the reasons designers prefer to leave coding out of mockups, these three are the most common:

- **Limits Creativity** — The strength of building with code can also be its weakness: knowing concretely what you can and cannot do will cut off many avenues before they're fully explored, while designing openly might inspire new ideas you wouldn't have had otherwise. Sometimes when you like an idea enough, you try harder to figure out how to recreate it with code, whereas the idea never would have even surfaced if you started in code.

- **Limits Experimenting** — It is a lot harder to try out new and different ideas in CSS/HTML than in other simplified programs. Just as with the above point, the freedom to create will often inspire new, more, and better ideas. Just ask yourself how many UI designs and iterations can you create in 30 minutes of Photoshop or Sketch versus CSS/HTML?

- **Dilutes Design Process** — Not everyone is great at multitasking. The mockup phase is all about iterating on the appearance of the site, and worrying about coding at the same time can get distracting. Some people prefer leaving things for the right place and time: visual design during the mockup phase, coding during the prototyping or development phase.

> *"Coded mockups aren't for everyone. Ask yourself how many designs can be done in 30 min of PS/Sketch vs. CSS/HTML."*　　**TWEET THIS**

Again, it's up to you when to introduce coding. Just make sure you know your design priorities and keep the developers updated on how you're prioritizing features.

Takeaway

Don't make the mistake of thinking all mockups are the same. Simple decisions about platforms, fidelity, and coding will all produce significantly different results. Know what you want and what your goals are before you even begin the design process — if you want a tool that supports all three phases, it's best to start out using it than to begin halfway through. Likewise, if you need a stellar, fully realistic mockup, keep in mind that you'll be using a graphic design editor at some point.

CHAPTER FIVE

Best Practices
for Mockup Tools

Practical tips for creating mockups with Photoshop & Sketch

Based on what we've reviewed, by now you should have a better idea of which platform will best help your needs. While previously we have described the different tools for mockups, here we'll explain the best practices to use for Photoshop and Sketch (two of the most popular graphic design tools for mockups). If you're still deciding between the two, here's an [interesting comparison between them](#).
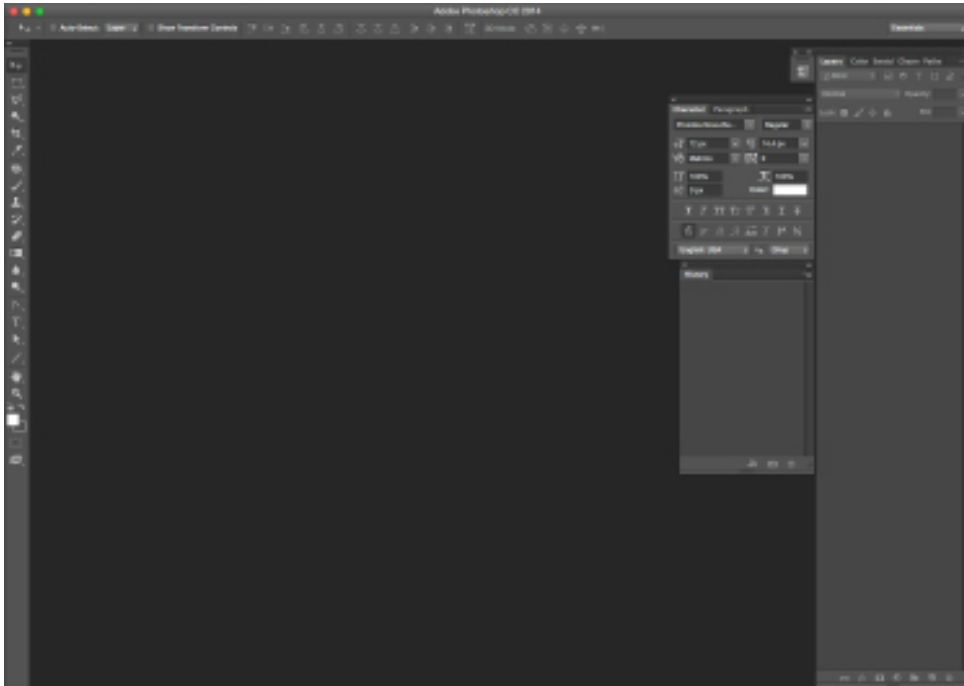

Photo credit: [uxpin.com](#)

For Photoshop, we'll describe 7 helpful tips, include helpful resources, and explain how to make to make the one-person tool more friendly for collaboration. For Sketch, we'll dive into 10 tips and provide links to our favorite resources.

"Design requires content and form: wireframes outline the content, and mockups add finesse to the form."

TWEET THIS

Tips for Photoshop Mockups

Photoshop has a large following of devoted users, thanks to its deep wealth of features, options, and user-created content. This post by **CreativeBloq** curating the [250 best Photoshop resources](#) speaks volumes about Photoshop's popularity. Many pixel-perfect die-hards still feel more comfortable designing their mockups in Photoshop over software specifically created for that purpose.

Photoshop can be used to make phenomenal mockups, as long as you know the relevant features — some of its features can seem like overkill for web design unless you're editing images heavily. Let's take a look at some tips for creating mockups in Photoshop, and how to make the experience more collaborative.

I. CREATING MOCKUPS IN PHOTOSHOP

If you're one of those Photoshop die-hards, here is some valuable advice about adapting your favorite software into mockup creation, given by some power users. We've described some best practices below based on this [piece by PlasticMinds](#) and our own experiences.

1. Take advantage of Shapes & Shape Layers
Much in the same way a wireframe provides the basic outline for a mockup, shapes and shape layers are a good starting point for your PS mockup. As a bonus, all layers are also preserved if you [import your Photoshop file into UXPin](#).

Photoshop gives you a variety of options for creating these basic shapes, and they can be easily resized for edits later. [Jake Rocheleau agrees](#), and points out that if you look closely at most published websites, you can still find the original shapes used for headers, navigation bars, buttons, and sidebar widgets.
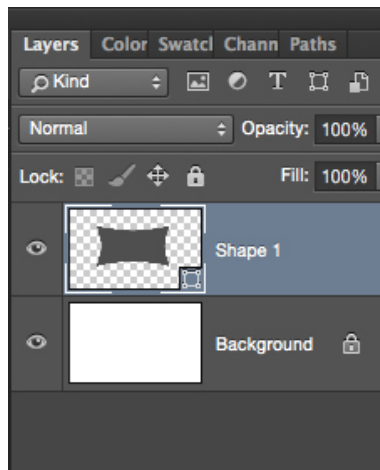
Photo credit: *uxpin.com*

[This tutorial](#) explains the essentials to Shapes & Shape Layers.

2. Use Crisp Antialiasing for Fonts (When Possible)

This prepares your mockup for browser rendering. But it may not work for all fonts, so when in doubt, check with your developer.
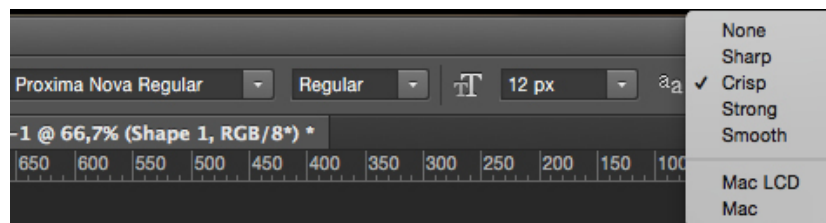

Photo credit: *uxpin.com*

To learn more about font anti-aliasing, check out [this tutorial](#). If you'd like to learn more about typography in the specific context of web design, check out the free e-book *Web UI Design Best Practices*.

3. Choose Web-Safe Fonts as a backup

It won't matter how great your font looks if it's not compatible with CSS/ HTML. Browser text, on the other hand, can be indexed by search engines, used by screen readers (for the visually impaired), and make for easier translations. It's best to start out using them (except for Comic Sans or Papyrus, of course).
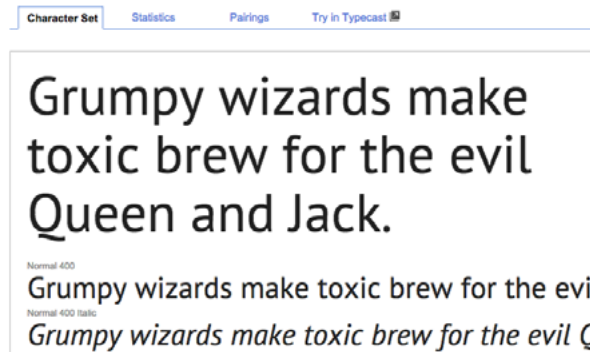

Photo credits: http://www.google.com/fonts/specimen/PT+Sans via awwwards.com

Google Fonts is the largest source of free web-safe fonts (check out **awwwards**' selection of the 20 best). On the other hand, Adobe Typekit is well worth the money — our company has an annual subscription and the 4200+ fonts are more than enough.

"Use web-safe fonts as a backup. Except Comic Sans or Papyrus."

TWEET THIS

4. If there's several versions of the page, use different groups

If you're doing multiple versions of a mockup, usually you'll just be modifying the main content. The header, footer, and sidebar probably won't be touched. To speed things up and help you stay organized, you can create folders like:

- **content - home page**
- **content - feature photo**
- **content - call to action**

In each version, you can just turn on or off the visibility. Whenever we work in Photoshop, this tends to improve our mockup efficiency.

5. Use Existing Resources

Premade toolkits full of UI design elements can save a lot of time designing new graphics and in transitioning to the prototyping phase. These downloadable kits come with premade buttons, controls, navigation, UI design patterns, and other UI elements to simplify the process. The right combination of UI kits, templates, fonts, and other elements will give you the right materials to tweak as needed.

You can find these kits in various libraries, four of our favorites listed below. We also created our own Awesome UI Kit and iOS 8 UI Kit (works with iOS 5s, 6, 6+ and Photoshop & Sketch).

- **Line 25** — list of 35 sites that provide templates, patterns, and UI kits for Photoshop

- **Freebiesbug** — excellent resource of free UI elements and kits for Photoshop

- **BlazRobar** — requires account creation, but is still a free resource for UI elements and kits for Photoshop

- **GraphicBurger** — helpful collection of free templates, icons, mockups, and UI kits for personal and commercial use
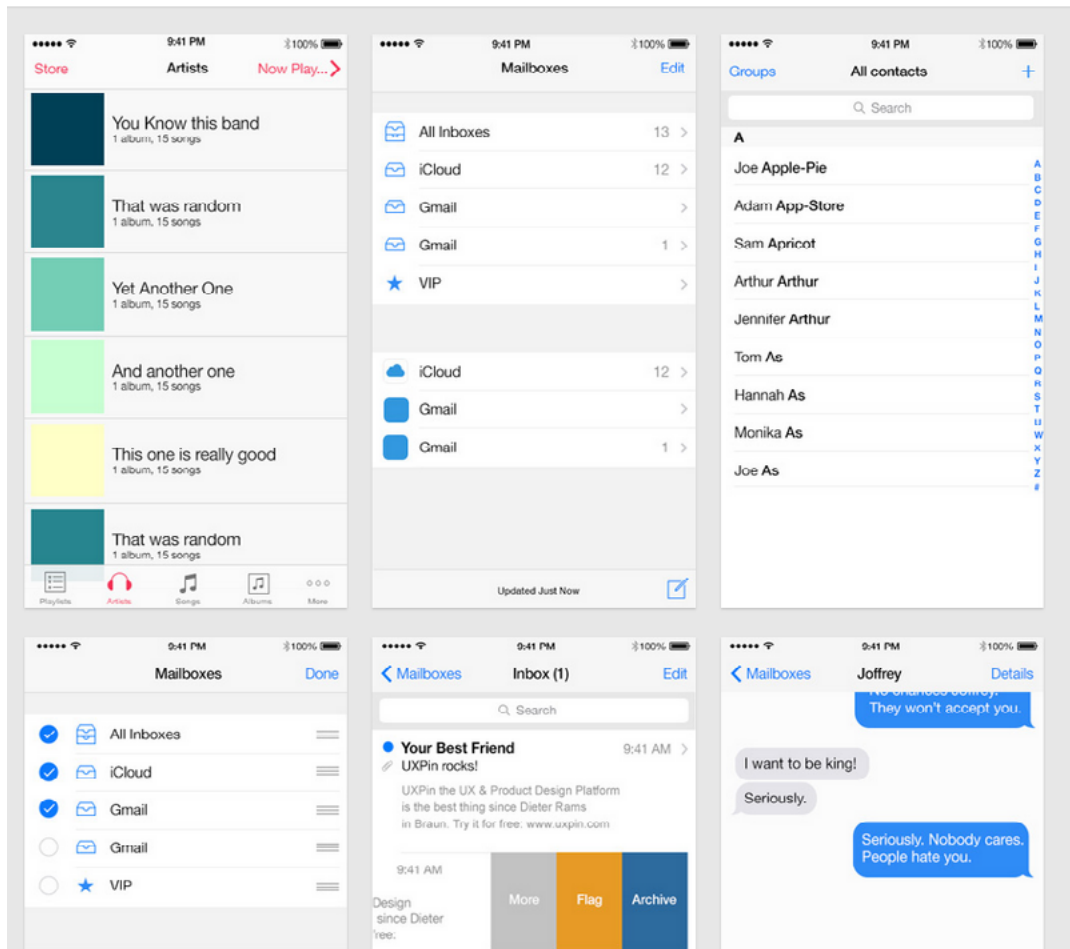
source: *Free iOS 8 UI Kit*

6. Use Smart Objects

As an additional tip, **Jake Rocheleau, creative designer and web developer**, explains the importance of Smart Objects in an article for SpyreStudios. Because vector-based images can be resized to any resolution, they work great for mockups. Importing a vector-based image into PS can make a Smart Object, which lend themselves to mockups because their pixels are "freer." Smart Objects also enable smart filters, which are recommended over regular filters — smart filters do not apply effects onto the pixels themselves, unlike regular filters.

If you'd like to learn more, **Martin Perhiniak, an Adobe-certified designer at Tuts+**, explains 10 things you must know about Smart Objects.
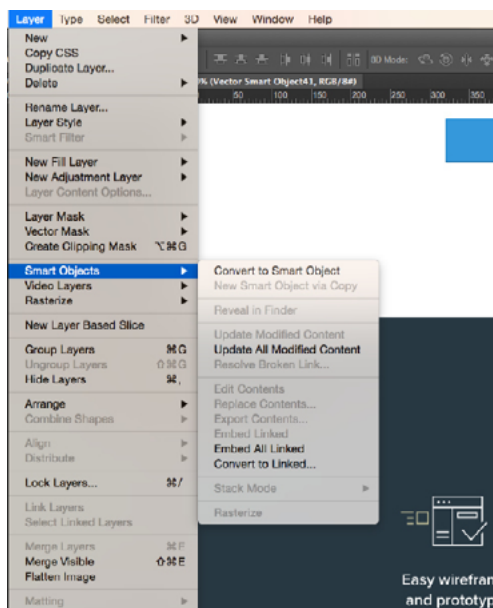

Photo credit: uxpin.com

For a comprehensive collection of tutorials for Photoshop techniques, check out this list of 18 resources from **Smashing Magazine**. These tips will work great when you're grinding away at your computer... but what about when it comes time to hand over the file to someone else?

II. COLLABORATING WITH PHOTOSHOP MOCKUPS

Photoshop is typically a one-person show, whereas the wireframing-mockups-prototyping process is highly collaborative. Design is never an island, so there are a few things you can do individually in PS that will make it easier for others later on.
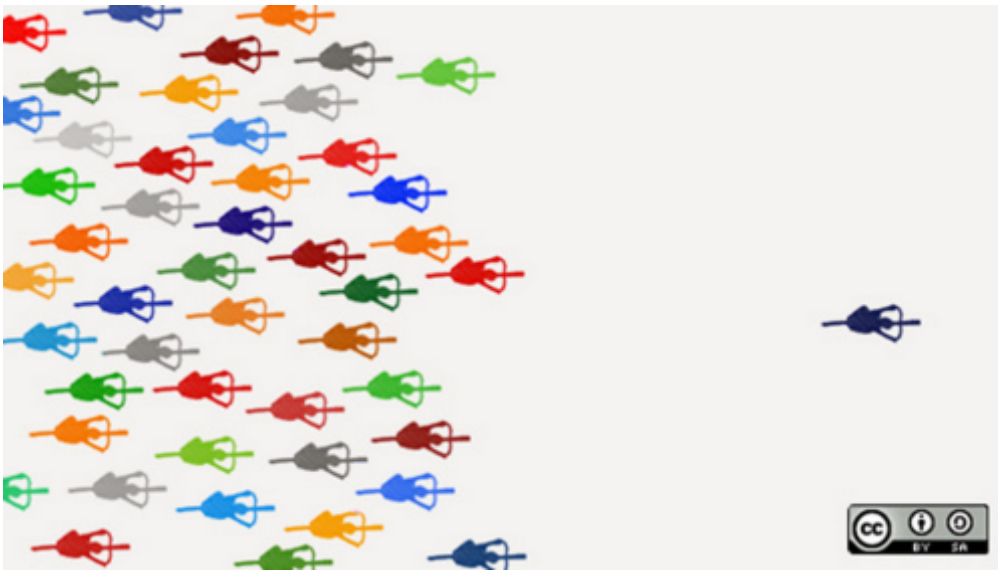

Photo credits: „Escaping creativity without giving up on collaboration" opensource.com. Creative Commons.

One of the most important considerations for collaborating in PS is the grid system. **Chris Spada, designer for Plaid**, explains that handing off a well-structured and organized mockup is crucial to designer and developer collaboration. In order to make sure a site will be compatible across different resolutions, its best to use a grid system. Moreover, a grid system will provide some extra structure, which can only make the otherwise chaotic process run smoother.

You also want to make sure there's a visual plan that all team members can work off of — in other words, a wireframe. As discussed in the Guide to Wireframing, having a shared wireframe prior to a PSD will help keep the workflow on track and streamline productivity.

Designm.ag provides some additional tips for keeping a PSD mockup efficient. The naming conventions on the document should be standardized, especially because they're being shared across the entire team. Including the version number and even the date (if applicable), plus categorizing the different files in separately labelled folders, will help keep everything in order and easily accessible. Likewise, layer titles within the document should also be named properly for the same reasons.
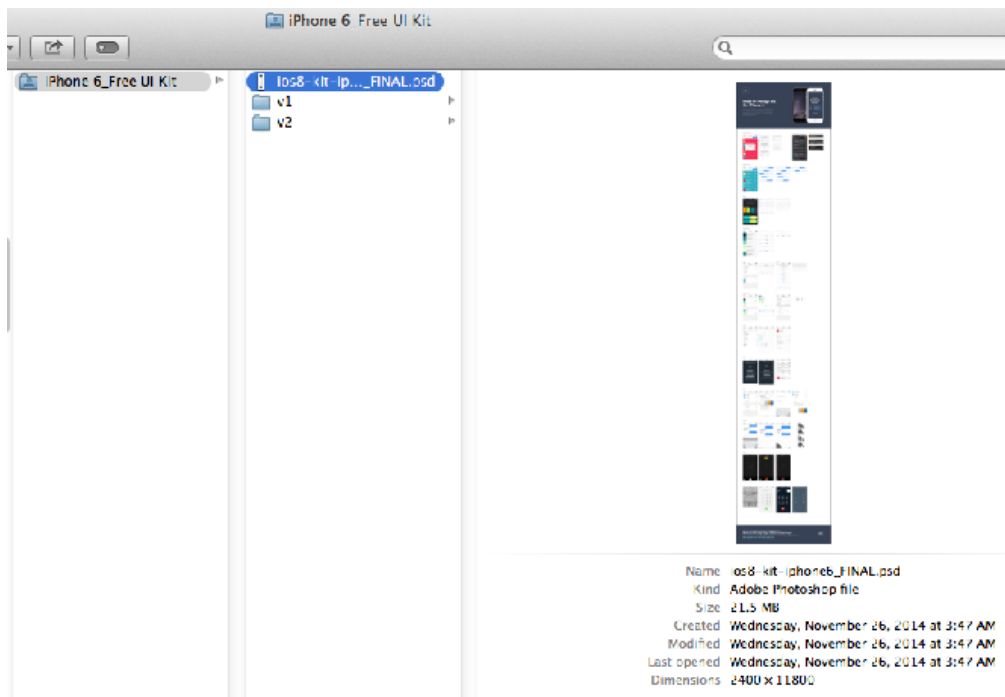
Photo credit: http://www.uxpin.com/ios8-ui-kit.html

Remember that design collaboration is absolutely crucial — it's not 1998, so it's quite an outdated (and dangerous) notion that you can hand off a PSD to a developer as if that person is a WYSIWYG machine. If you finalize the mockup in a graphic design tool, keep an open line with developers to regularly verify the technical feasibility of the design — what's on the screen may render differently in other devices, browsers, and platforms.

"Developers aren't WYSIWYG machines. Don't just hand off your mockup and wash your hands of it."

TWEET THIS

While the above are tips for Photoshop mockups, if you're looking for more general Photoshop tips, you can read this practical manual on Photoshop etiquette.

Tips for Sketch Mockups

While Photoshop is the current heavyweight champion, [Sketch](#) is the up-and-coming contender, quickly winning over the favor of the Mac crowd. Sketch is a UI design app for Mac, and has been [receiving a lot of online attention lately](#) (like from **Etsy**'s designers).

Photo credit: http://bohemiancoding.com/sketch/

If you're designing a mockup in Sketch, a lot of the same rules for Photoshop apply. However, Sketch offers some diversity that Photoshop or other tools do not (just being 100% vector-based already saves a ton of time when designing for web and mobile). Derived from our own experience designing in Sketch, here's 10 tips that have helped us quite a bit when creating mockups.

1. Use good naming conventions.

When you set objects to export, they are exported into the folders you group them, with the layer names given. To prevent headache when exporting assets for development, stick to layer names that match your naming conventions, and use folders that will make sense to the developer.
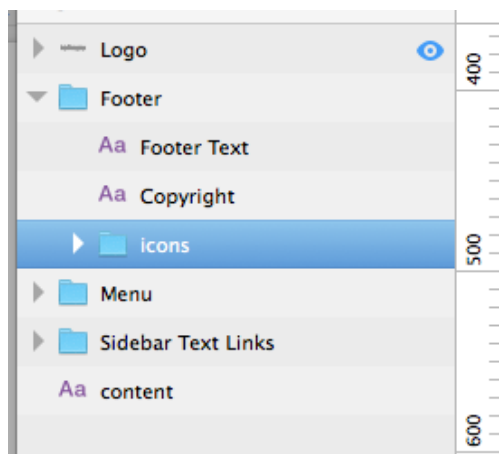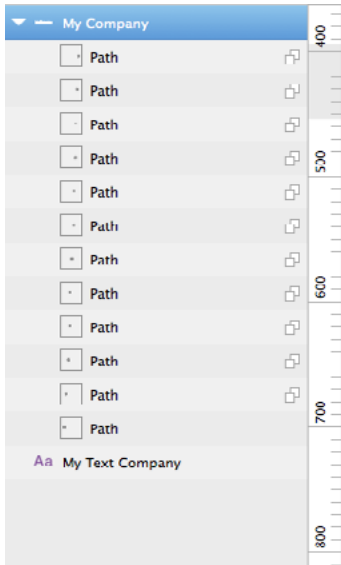

Photo credit: uxpin.com

As discussed in 7 Tips for Sketch users, it can be as simple as inserting a prefix "_ic" before all icons and using the prefix "img_" before all images. Whatever naming system you decide, make sure you discuss it with the developers first so you can find what's most intuitive for everyone. This will make for a more organized process of exporting the assets and importing them to Xcode.

2. Manipulating logotypes? Convert to outlines first.

If you're manipulating or resizing logotypes, make sure you convert it to outlines first. In other words, Sketch stops treating this as a string of text, and treats it as a group of vectors (which allows for more customization). You can then tweak specific letters, adjust spacing better, and resize without losing your proportions.

3. Create Text Styles for headlines and other repeating text types.

Sketch lets you create Text Styles that you can apply to any layer of text in any artboard or page. This is great if you're creating a mockup for a website UI, web app interface, or mobile UI.
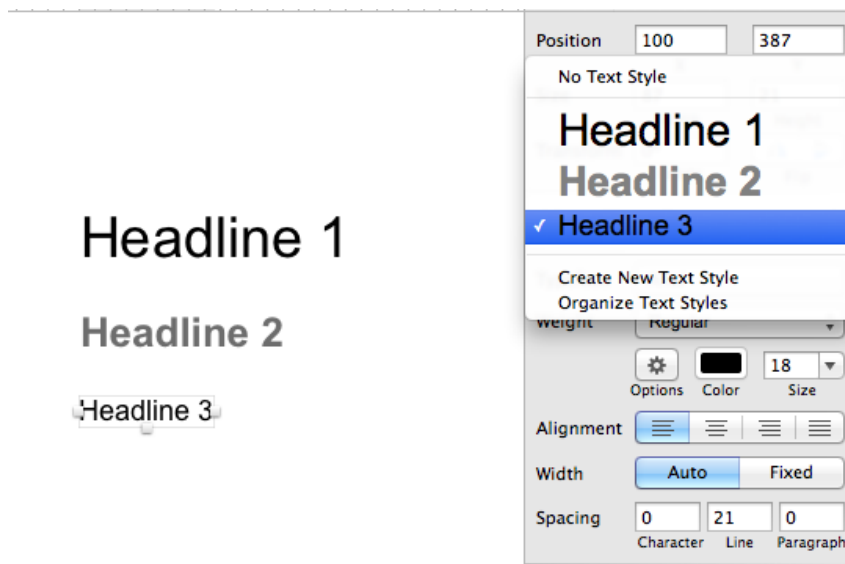
Here is an excellent step-by-step tutorial on how to create Text Styles, complete with plenty of screenshots and GIFs.

4. Create symbols for repeating patterns or UI elements.

Symbols are like Text Styles for groups of layers or non-text objects. You can turn any button, box, or entire UI frame into a symbol. This way you just need to modify that element in one place, let's say your navigation, and it will apply across all artboards and pages.
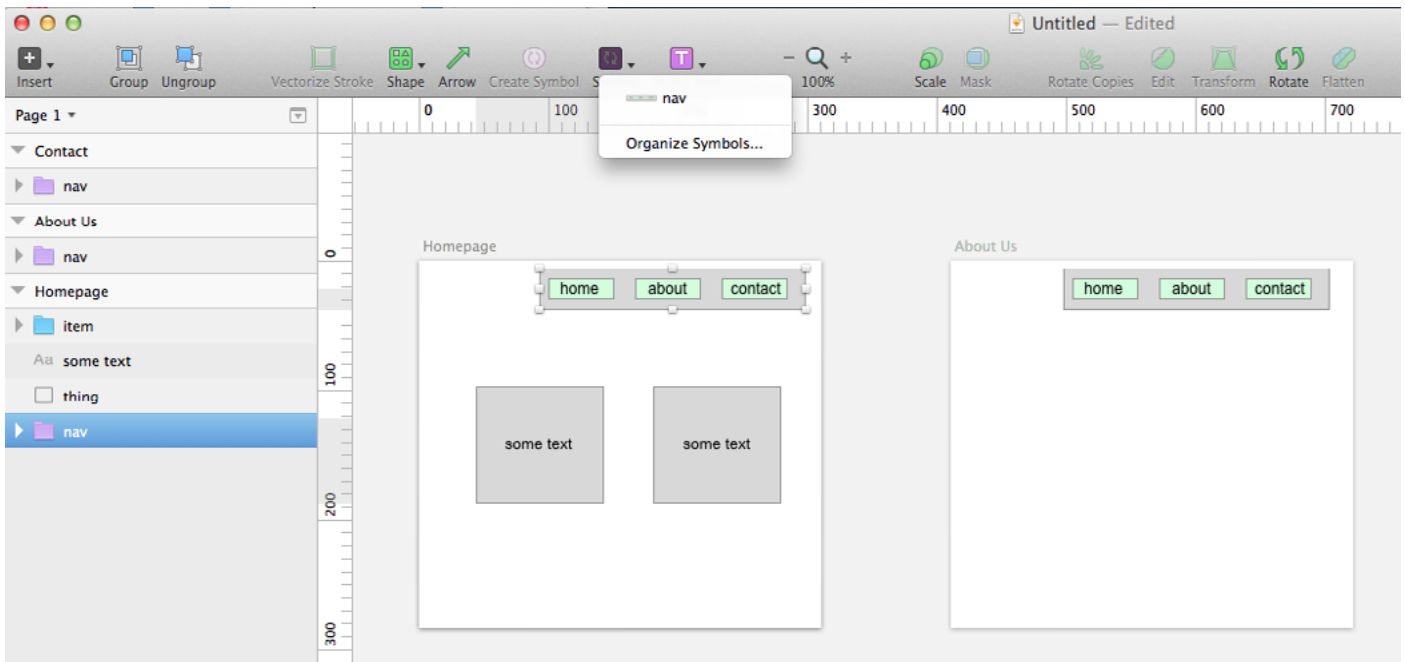

Photo credit: uxpin.com

5. Architect your product by taking advantage of multi-screen view

Zoom out at a bunch of artboards so you can quickly see your flows and product depth. Use this high-level view to design an onboarding flow or user journey. Then, dive in and design screens one by one. Later, you can zoom out to get a high-level overview. As discussed in *Web UI Best Practices*, doing this from time to time helps you stay focused on the big picture.
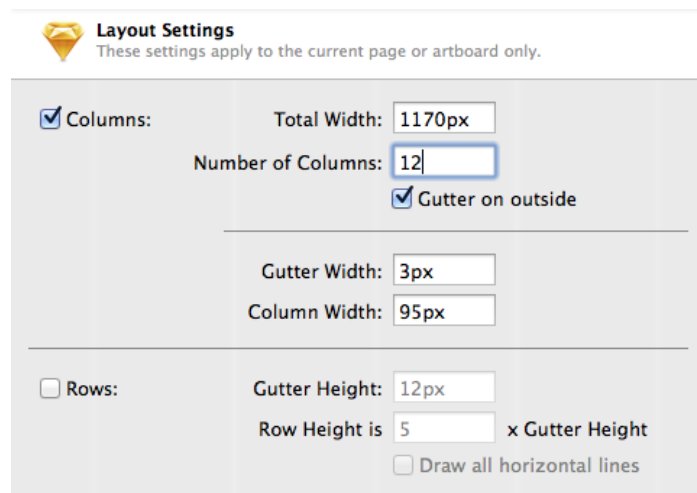

Photo credit: uxpin.com

6. Use layouts and your developer will love you.

Seriously — design based on a web grid like [Foundation](#), [Bootstrap](#), or [960gs](#) and your developer will be grateful. Our developers sit right next to our designers and this helps save time and sanity during the mockup collaboration process.

Turning design into code is a whole lot easier when you don't have to build a bunch of custom layouts. Of all of these front-end frameworks, Bootstrap is the most popular and versatile. Foundation is great too, especially with the new Foundation Mobile. For more on the differences, **Treehouse** has a [great piece](#) comparing and contrasting Bootstrap and Foundation. As for 960gs, it used to be more popular, but is beginning to be outdated (**Tuts+** actually explains [how to migrate from 960gs to Foundation](#)). We invite you to check them all out and discuss with your developers before you pick one.

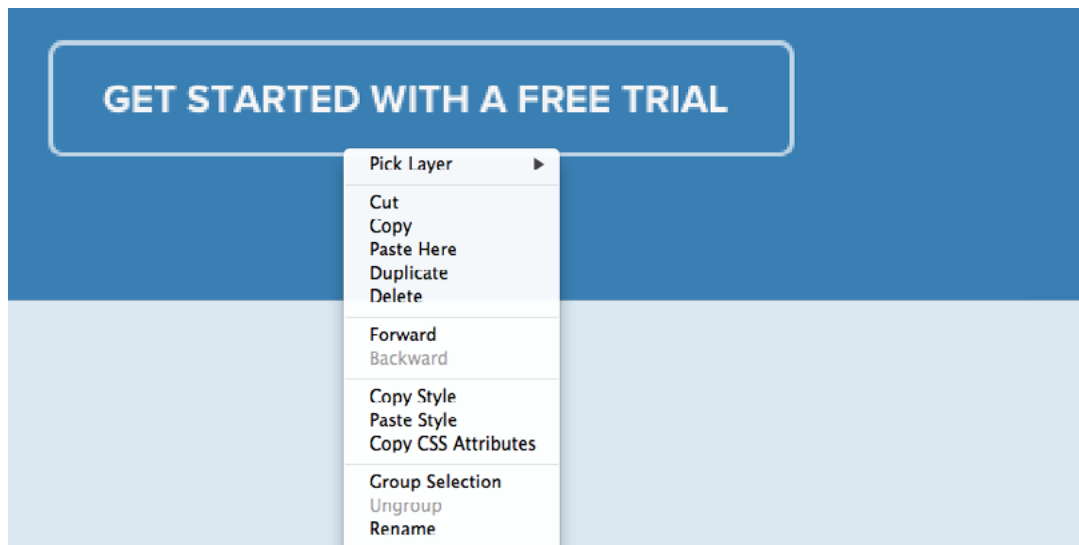*Photo credit: [uxpin.com](#)*

If you're building in Bootstrap, set up your layout as in the picture above, with these parameters:

- Total Width: 1170px
- Number of Columns: 12
- Gutter on outside: checked
- Gutter Width: 3px
- Column Width: 95px

7. Export CSS Attributes

This is a lifesaver if you're strapped for time. Select any object or text layer, right-click it, and select **Copy CSS Attributes**. Fire up your text editor or notes and paste it in.

You can do this for all the complex layers in your mockup to make it dead simple for you or your developers to code. This is especially necessary when you're dealing with gradient backgrounds or shadows.

Photo credit: uxpin.com

8. Take advantage of Plugins and UI Kits

There are a bunch of great plugins that will speed up your design workflow.

As you might expect, we use UXPin Export (which also integrates with Sketch) to quickly turn high-fidelity mockups into an interactive prototype to get feedback from the team. While we also dogfood UXPin for mockups, some of our designers come from a visual design background so Sketch (and Photoshop) are second nature.

If integrating with UXPin to help simplify your workflow, check out this tutorial. If you want to discover new plugins, we recommend you check out Sketch Toolbox. This little app lets you manage and install your Sketch plugins.
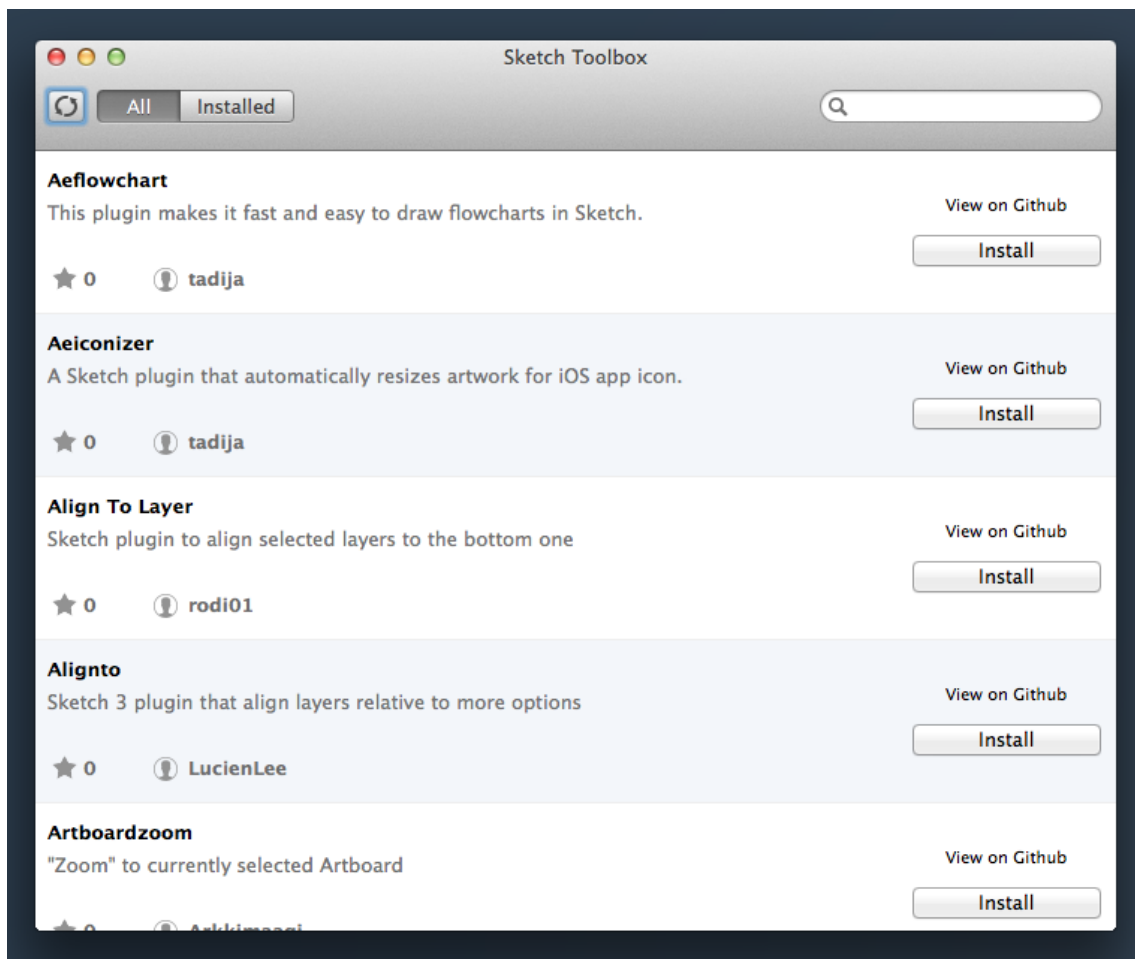
UI Kits will help you quickly create prototypes and early designs. You can customize them and add visual treatments later, but why reinvent the wheel? One of our favorite UI kits is the Bootstrap V3 GUI, which you can find on SketchAppSources. If you're designing for iOS 8, check out our free iOS 8 UI Design Kit (quite helpful since the elements work with iPhone 5s, 6, and 6+, and are compatible with Sketch and Photoshop).

9. Engage with the Sketch community

While there's tons of helpful tutorials, there is no substitute for learning from those with experience. Paying it forward is a rule of thumb all of the best designers follow because they have been on the receiving end of it themselves. We've included some of our favorite Sketch resources and community sites.

Sketch Resources:

- [Sketch App Sources](#)
- [Sketch Resources](#)
- [Sketch Shortcuts](#)

Sketch Community

- [SketchMine](#)
- [Brilliant Sketch](#)
- [Sketch Tricks](#)
- [Sketch Casts](#)
- [Sketch Tips](#)

10. Make it work

The most important rule of all: if it works, use it, and if doesn't, don't. We discovered, then tested, then finally embraced Sketch through a series of tests on projects back in 2012. So far, it's worked great for us. If these tips work for you, great! If not, don't force it.

If for whatever reason you're not satisfied with the other options available to you, Sketch might be worth checking out. If you're already a Sketch convert, take these tips to heart when designing your next mockup.

Takeaway

Work with the tool you feel most comfortable with, which is not to say that one is better than the other, but that the user is better with one than the other. To us, that's the most important practice of all. After all, design tools are made for one purpose — to make designing easier for everyone.

> *"Talk to your developers. What's on the screen may render differently in other devices, browsers, and platforms."*
>
> TWEET THIS

And no matter what tool you choose, always collaborate. The power of the mockup is that it's a universal medium for visual communication, so use it to open up dialogues about technical feasibility with developers. A few words exchanged at the right time can go a long ways to avoiding the misery of late-night redesign.

If you're interested in a cloud tool built specifically for wireframing and prototyping, feel free to check out UXPin. Collaboration is built-in, so it's easy to work with developers and product managers as the design goes from wireframe to mockup and prototype. Based on our experience with graphic design tools like Sketch and Photoshop, we've also included dozens of in-app element libraries and UI kits.

CHAPTER SIX

How to Use Mockups in the UX Design Process

Adapting mockups for different projects and processes

Traditionally, the process goes from wireframing to mockups to prototyping, with variation in fidelity for each stage depending on the team. But like most design practices, others have strayed from tradition and discovered new and helpful alternatives. We'll explore two such alternatives and give advice on each, then explain how mockups can make it easier to create style and brand guides.

Implementing Mockups in the Process

As discussed in *The Guide to UX Design Process & Documentation*, mockups can be more than just a step between wireframes and prototypes. In fact, even this transitional function can vary in importance and implementation.
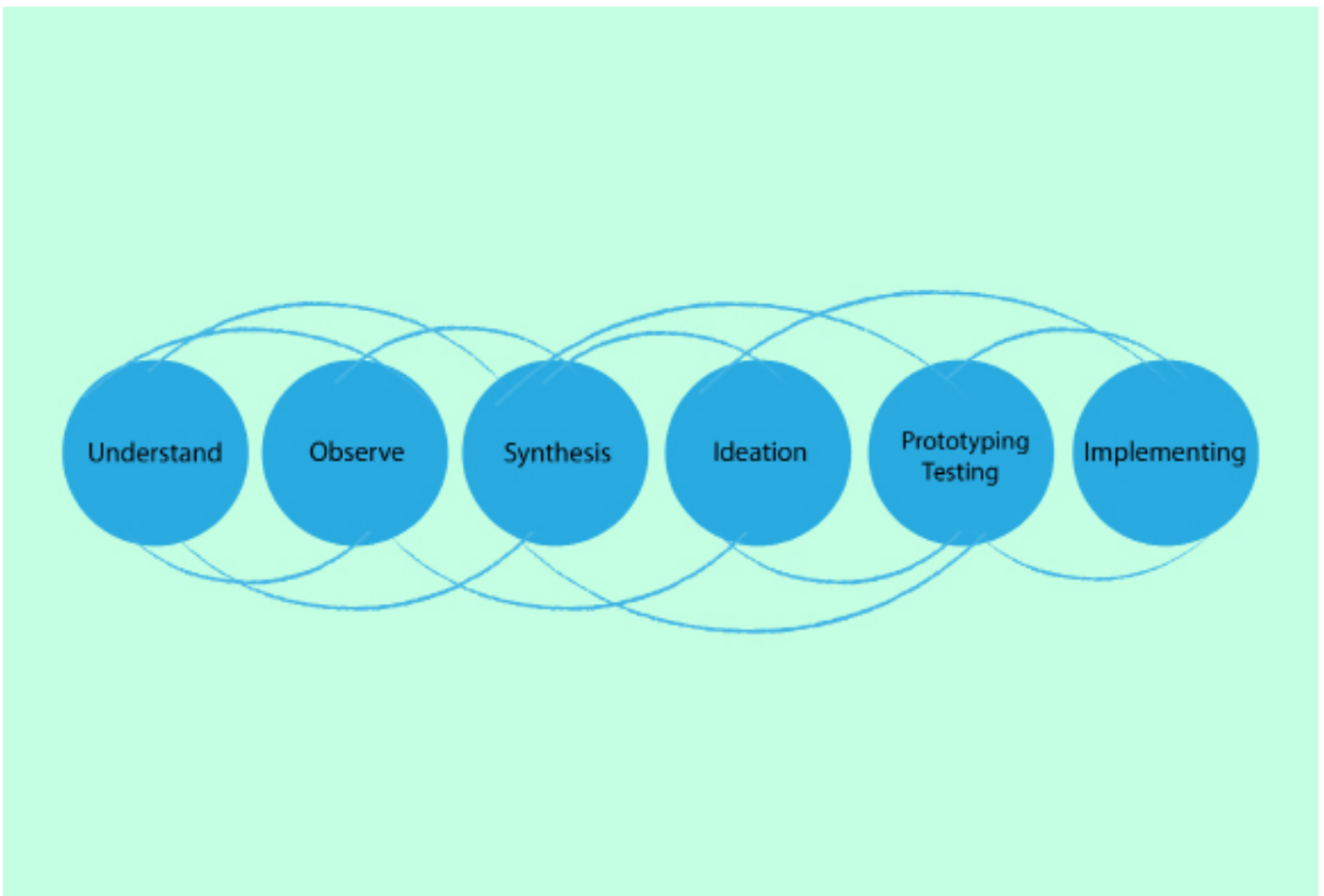
Before we get into the alternative processes for using mockups, we'd like to list some general advice to keep in mind when creating mockups, regardless of how you implement them. **Bima Arafah, Designer & Front-End Engineer at Nesia**, believes in a KISS (Keep it Simple Stupid) approach [when it comes to design processes involving mockups](). Generally speaking, it helps to avoid overly fancy visually effects like high contrast font types, and keep it simple by using existing palettes like [Adobe Kuler](). Here's 5 more tips to keep even high-fidelity mockups grounded in simplicity:

- **Design for the project's needs** — If stakeholders have trouble articulating their needs and expectations, ask them to collaborate. You should start the discussion around images, logos, and visual hierarchy well before your mockup is done (but don't let them have the final say, otherwise that's [design by committee]()).

- **Don't skip wireframing** — The wireframe is your guideline to your mockup. It allows you to answer broader questions about layout and content so that they don't distract you from visual decisions. Like this [Priority Guide](), a wireframe is naturally collaborative since the concepts are easily digestible.

- **Focus on a centralized concept** — Sketching and wireframing is great for early concepting, but you need to have a clear direction by mockup time. As the old Native American adage goes, "If you chase two rabbits, you will lose them both."

- **Check out rival sites** — In general, it's a helpful habit to be aware of what your competitors are doing, but in the wireframing and mockup phases it's especially helpful for thinking outside the company. A [quick heuristic review]() shows where competitors succeed — and how you can improve your own site based on their mistakes.

- **Don't neglect mockup presentation** — One of the main functions of a mockup is its assistance in stakeholder presentation. If some elements don't come across in the mockup, include a few notes to explain their functions.

As we explained in *The Guide to UX Design Process & Documentation*, the design process is as diverse as the companies using them. With these guidelines in mind, let's take a look at two opposite ways mockups fit into the design process.

II. WIREFRAMING → MOCKUPS → DEVELOPMENT

This design method skips UX prototyping and leaps straight from mockups into development. Proponents of this method claim that prototyping is useless, as the sooner development starts, the better. In this case, the mockup gets added importance, as it is the most prominent "blueprint" referred to throughout development.
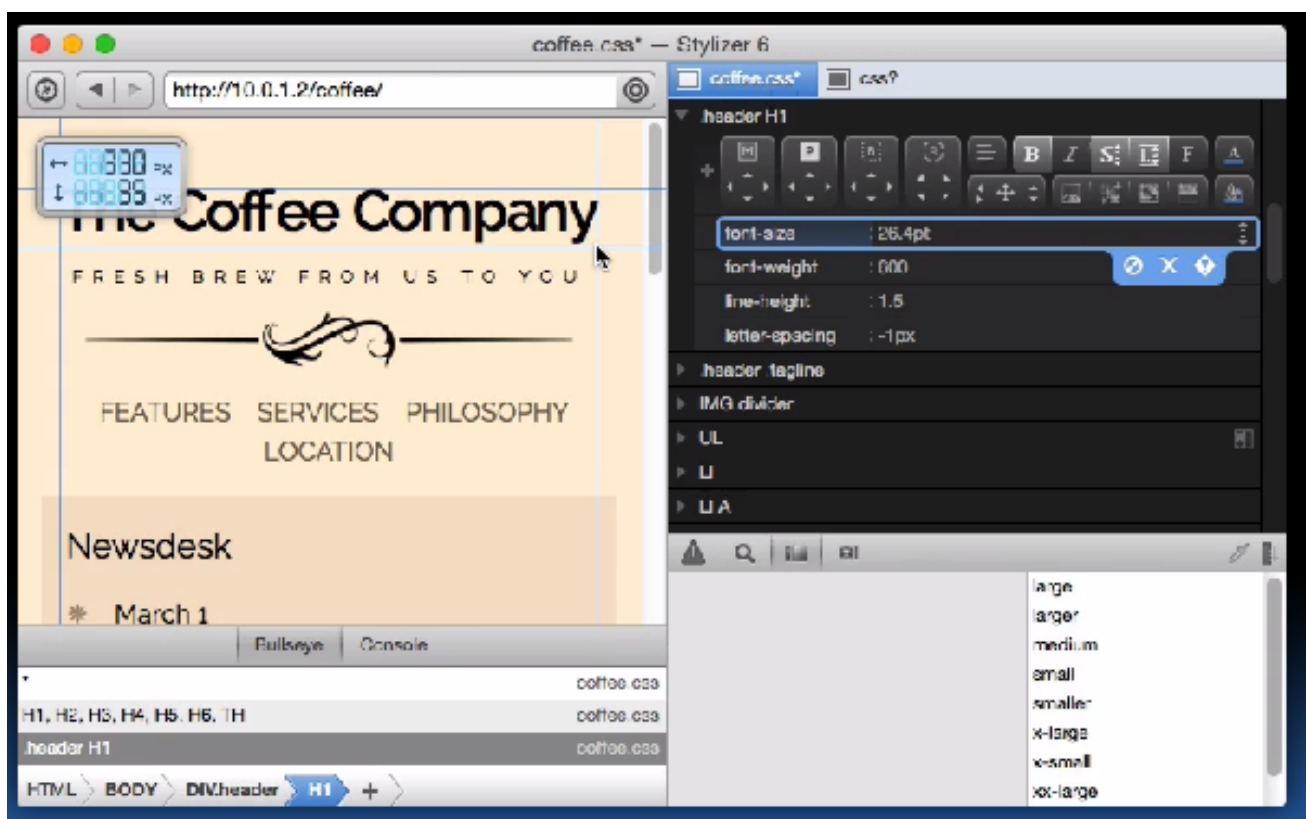

Photo credit: http://www.skybound.ca/

Steven Bradley of Vanseo Design is a follower of this method, and [explains his personal process in a post for his site](). However, it's important to note that even he, a professional designer, confuses the terminology and that his hi-fi wireframes and design comps are technically mockups. (For this reason, we included that disclaimer about nomenclature at the beginning of the e-book.) Here's the overall process:

- **Sketching/Wireframing** — The first step of the process is sketching out the basic ideas and starting elements, plus a few rough design elements. Next comes wireframing, where these rough ideas are fleshed out and specified. Essentially, these early phases don't stray from the process described in *The Guide to Wireframing*.

- **Hi-fi Mockups** — Next comes a high-fidelity mockup, meant to be as "pixel perfect" as possible. Keep in mind this is the final outline before the development process, so most details and decisions should be finalized and the final version should be run by the client.

- **Development** — With the preliminary phases out of the way, it's time to start building the actual site. Bradley starts with a single HTML file that includes CSS and Javascript. When visuals don't code properly, extra time is spent redesigning until the final product is workable. At that stage, the file is sent to a tool like WordPress, where further revisions are made if needed.

While we don't necessarily advise skipping the prototyping phase (since it can limit creativity), we understand its value in certain situations. However, even in such situations, we would recommend at least a rapid prototyping phase over none whatsoever. **Smashing Magazine** posted a [great piece]() on how to conduct a highly efficient 3-step prototyping phase for situations like these.

Speaking of the importance of prototyping, let's take a look at the a design process that combines mockups and hi-fidelity prototypes. This is actually the process we follow at UXPin.

## II. WIREFRAME → LO-FI PROTOTYPE → HI-FI MOCKUP/PROTOTYPE

Quite the opposite of the last method, this process places extra emphasis on the prototyping phase, initiating it as early as possible. This is the method that most closely relates to the coded mockup we discussed in Chapter 3. This early integration with functionality works well with designers who know their code, or with a project that involves especially complex technicalities.
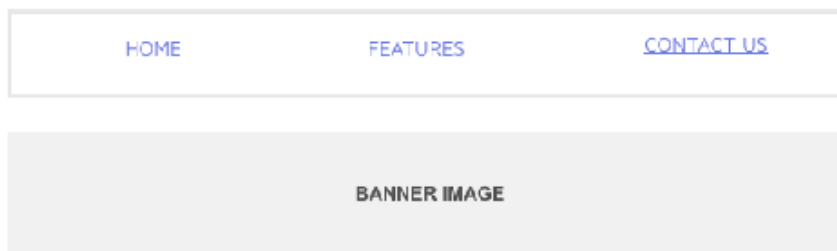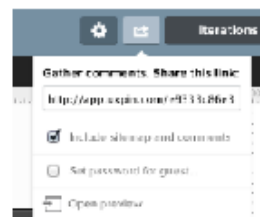
Photo credit: uxpin.com

In the previous e-book *User Testing & Design*, we discuss this method at length; however, here we'll give a brief overview:

- **Wireframing** — This phase starts out as always — basic ideas, basic structuring.

- **Lo-fi Prototype** — The important factor here is "lo-fi," as you don't want to bite off more than you can design this early on. At this stage, you focus more on interaction and functionality. For example, [when we reworked Yelp's website](#), we simply added interactions to our wireframe in UXPin, [as you can see on this live version](#).
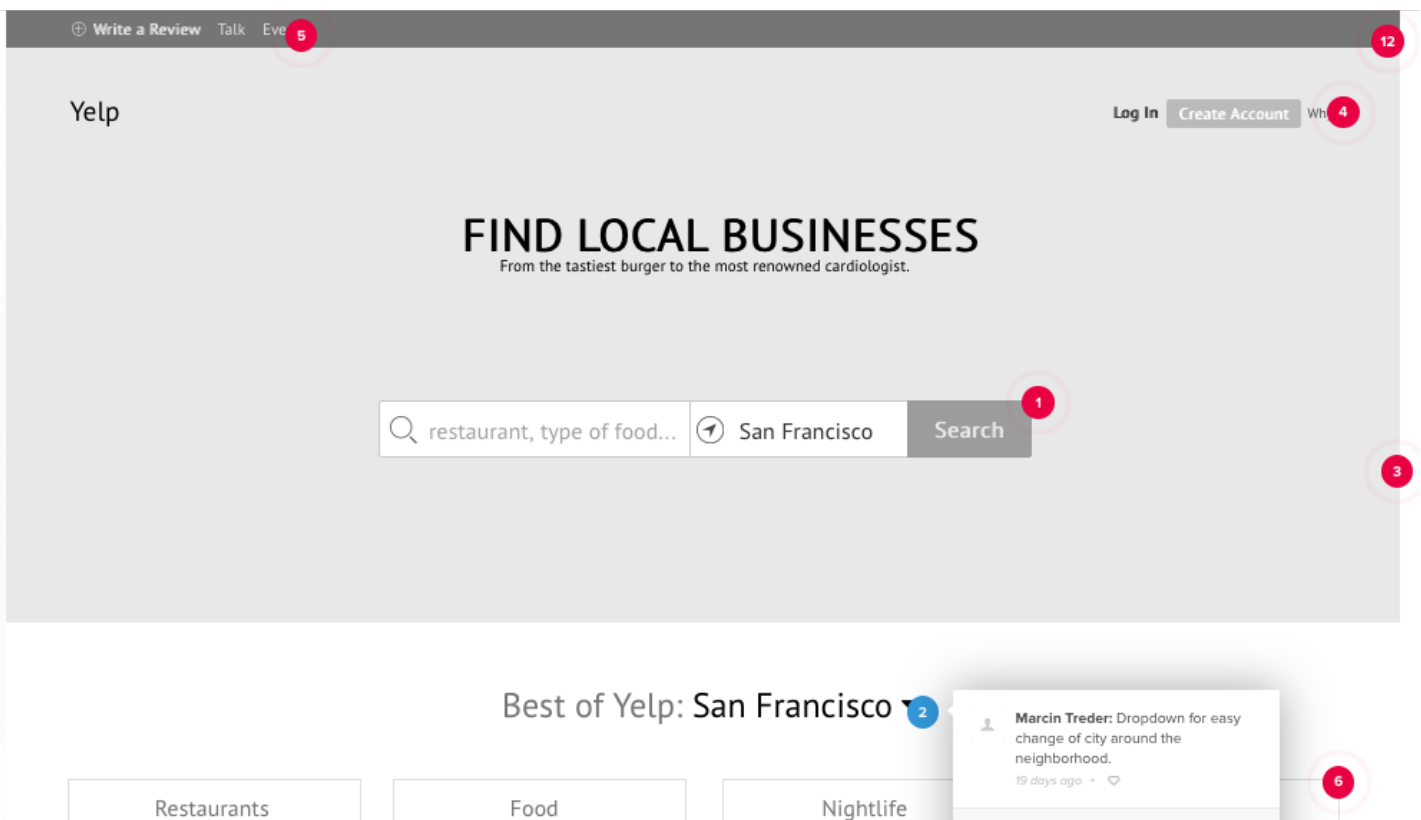
source: *UXPin Low-Fidelity Yelp Design*

- **Hi-fi Mockup/Prototype** — When it comes time to focusing on visuals (the main purpose of mockups), you can just inject those details into the existing lo-fi prototype. As the visuals develop, add more and more interactive elements as well, merging the mockup with the lo-fi prototype to create a hi-fi prototype. Whatever visuals you can't create in your prototyping tool can be built in Photoshop or Sketch, then can be imported and [dragged-and-dropped into UXPin](#) or redrawn in other tools.

- **Usability Testing and Iterating** — One of the big advantages of this method is that functionality can be tested, refined, and perfected early on when it's still (relatively) easy. By conducting usability tests between each stage of design, you can start addressing problems before they're even problems.

If you're curious about a variation of this process, **Bhavin Parikh, Founder and CEO of Magoosh**, explains [how to turn web requirements into mockups](#). Like we described before, he also emphasizes thinking about interaction design before working on the visual design (although he incorrectly uses the terms mockup and prototype interchangeably).

Really, though, there are any number of methods and strategies for the design process, from obsessive and meticulous planning to jumping head-first into development. While we've discussed several approaches, feel free to mix-and-match depending on your own needs, strengths, and weaknesses.

Chop Up Your Mockup for Lean Style Guides

We've spent a lot of time talking about how beneficial mockups are for visual design and client relations. But there is a third, lesser-but-no-less-important use for mockups, and that's in helping to compile a style guide. It makes sense when you think about it — the mockup is visual documentation of the product's style, so you can cut and paste elements to create a fast style guide.

In the e-book *Web UI Design Best Practices*, we discuss style guides and how to create them, listing out expert suggestions on what to include and how to format them.

Color palette

Yelp's primary palette contains red, blue, and warm greys.

When designing for Yelp, you are not limited to these colors, but they should be prominent in any consumer facing design.

$yelpy-red #c41200	$blue #3b65a7
$grey-10 #ebebe1	$grey-5 #f5f5f1

Color

Backgrounds and borders

$background-color #f5f5f1	$medium-background-color #e5e5e1
$border-color #cccccc8	$light-border-color #e5e5e1
$ad-border-color #ebc074	$highlighted-background-color #f01319 $highlighted-border-color #cbd6a7

Photo credits: http://www.yelp.com/styleguide#graphical_elements

To highlight the most relevant points, a style guide is simply a formalized compilation of the look and feel for your site, so that designers and developers have a quick reference guide. For some professional examples, the UX Bookmark lists some useful online samples including Apple, Android, and even Heineken. Style guides contain information like the proper layout standards, margin sizes, which fonts are used where and what size, branding rules, and so on. Because these are choices commonly made while building a mockup, the correlation between the two is easy to see.

When you make these stylistic decisions, you can simultaneously record them with quick notes in a style guide and some screenshots of the mockup. This will save you the trouble of building one later since you won't have to go digging up the information. Since we move pretty quickly in our design sprints, this is exactly how we build and update our style guides at UXPin.

For even more convenience, Medialoot offers a free UI Style Guide template — fill out what works and tweak the rest.

Takeaway

No matter where they appear in the design process, how much fidelity they have, or how much time you spend on them, mockups are quite helpful as visual documentation, quick spec sheets for developers, and a canvas for exploring visual details.

> *"Treat mockups as a medium for conversation, not something tossed to developers."*

TWEET THIS

Often ignored (or confused with wireframes), mockups can be overlooked because they're seen as extra work. But as we've discussed in this e-book, the key to minimizing double duty is collaboration. You'll get much more value from a mockup if you treat it as a medium to explore the technical feasibility of design rather than just a deliverable handed off to developers.
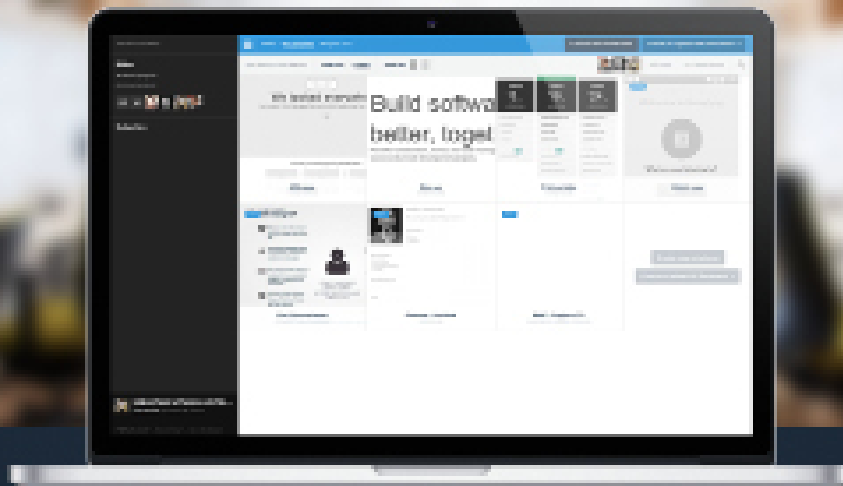

Photo credits: uxpin.com

Remember that design requires content and form: wireframes outline the content, and mockups add finesse to the form. If nothing else, mockups at least help you better avoid these dreaded words from clients and stakeholders: "I'll know it when I see it."

✔ Complete prototyping framework for web, mobile, and wearables

✔ Collaboration and feedback for any team size

✔ Lo-fi to hi-fi design in a single tool

✔ Integration with Photoshop and Sketch

UXPin

www.uxpin.com