How To Conduct Own Google Ventures Design Sprint

TO MENTOR 150 STARTUPS IN ITS PORTFOLIO, GOOGLE VENTURES DEVELOPED A FIVE-DAY DESIGN PROCESS. HERE, THE METHOD’S MASTERMIND, JAKE KNAPP, DETAILS HOW YOUR COMPANY CAN RUN ITS OWN SPRINT TO SOLVE A BIG PROBLEM.

At Google Ventures, we do product design work with startups all the time. Since we want to move fast and they want to move fast, we’ve optimized a process that gets us predictably good results in five days or less. We call it a product design sprint, and it’s great for getting unstuck or accelerating projects that are already in motion.

I’ve planned and run over 40 of these sprints, first with teams at Google and now with startups in the Google Ventures portfolio. To give you an idea of what one looks like, here’s a project we did with CustomMade:

Over the next several posts, I’ll be sharing a DIY guide for running your own design sprint.

**Before the sprint: Prepare**   
Get the people and things you need.

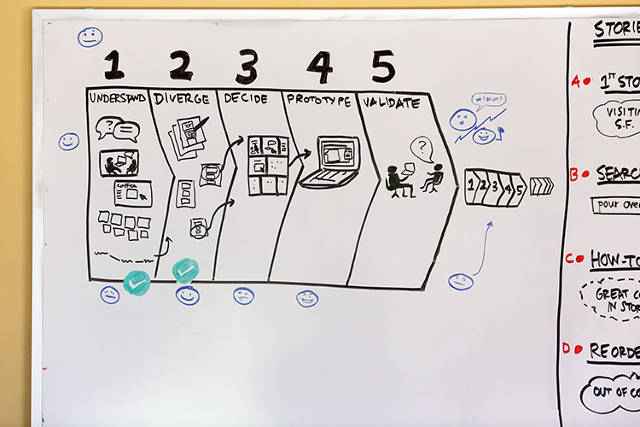
**Day 1: Understand**  
Dig into the design problem through research, competitive review, and strategy exercises.

**Day 2: Diverge**   
Rapidly develop as many solutions as possible.

**Day 3: Decide**  
Choose the best ideas and hammer out a user story.

**Day 4: Prototype**   
Build something quick and dirty that can be shown to users.

**Day 5: Validate**  
Show the prototype to real humans (in other words, people outside your company) and learn what works and what doesn’t work.



If you think you’ve heard of this model before, well, you’re right. It’s based on the [design thinking](http://en.wikipedia.org/wiki/Design_thinking) structure championed by [Ideo](http://www.ideo.com/" \t "_blank) and [Stanford’s d.school](http://dschool.stanford.edu/). However, I’ve experimented and tweaked the process a bunch over the last few years. The version I’m going to share works especially well for startups.

**WHAT DOESN’T WORK ABOUT BRAINSTORMING**

I’m a big-time process nerd. Several years ago, I started experimenting with product design processes at Google. At first, I ran group brainstorming workshops inspired by [the Ideo approach](http://blogs.hbr.org/cs/2012/09/the_secret_phrase_top_innovato.html). Group brainstorming, where everyone shouts out ideas, is a lot of fun. At the end of a workshop we’d be tired, in good spirits, and the proud owners of a big pile of sticky notes. But the new ideas we came up with didn’t go anywhere. It’s not that we were coming up with dumb ideas--most of the ideas were actually pretty good. Yet still, better ideas were coming from somewhere else. But where?

In my experience, the most successful ideas tended to come from individuals, not groups. The ideas took some individual heads-down work time to develop, too. I ran a lot of workshops before I realized this, so if it seemed obvious to you from the beginning, I hope you’ll cut me some slack.

To make matters worse, my workshops were choosing the winning ideas by consensus. But consensus doesn’t always pick the bold ideas, the unique ideas, or the ideas with design integrity. Consensus tends to compromise.

There were a lot of good things going on in the workshops: focusing a team on one project, considering a range of ideas instead of just one, working on paper, and more. But I decided my method was fundamentally flawed. I was getting good-but-not-great ideas through group brainstorming, then choosing winners by consensus. I knew that wasn’t working, but at first, I wasn’t sure what to do about it.



**THE MAGIC OF CONSTRAINTS**

One day I noticed something about my own design projects. The best work happened in short bursts, when I was under a deadline. One example was Gmail Priority Inbox, where we spent four weeks experimenting with different design prototypes. There were hundreds of internal dogfood users signed up to try a new experiment each week, so I had to move fast. By the end of the four weeks, I’d figured out which things worked, and saved months of noodling.

Another example was the project that became Google+ Hangouts. It started as a side project with two Googlers in the Stockholm office. I was visiting for only two days, so I designed as fast as I could. By the end, we had a working prototype that we could start using for our team’s meetings.

In both of these cases, I worked far more efficiently than I ever did in my normal day-to-day routine or in any of my brainstorm workshops. So what was different? I had time to focus and develop ideas on my own, not shouting and pitching the way I would in a group brainstorm.

But I also didn’t have too much time. I couldn’t afford to overthink things or get caught up in urgent but less important issues, the way I often did on normal workdays. And the people I needed to help me--engineers and product managers--were also focused on the project.

There was something magical about a tight time constraint combined with individual work, prototyping, and quick user feedback.

**ADAPTING IDEO-STYLE WORKSHOPS TO WORK AT GOOGLE**

I decided to try linking an Ideo-style “how might we” workshop to a few days of uninterrupted design time to execute the best ideas. In that very first sprint, designer Jason Cornwell roughed out the idea for the [Gmail people widget](http://gmailblog.blogspot.com/2011/05/introducing-people-widget.html). I knew I was on the right track.

I focused full-time on running design sprints with various teams at Google. I switched from group ideas to individual ideas and gave people more time to develop those ideas before getting feedback. I tried a bunch of critique and decision-making exercises that didn’t rely on consensus and chose a handful that worked best.

I got a lot of practice: I’d jump from team to team within Google, for a few days or a week at a time, leading sprints for projects like Chrome, Ads, Commerce, Apps, Search, and Google X. It was exciting. The designs were launching, and lots of teams started to run the process on their own.

**10X FASTER: RUNNING DESIGN SPRINTS AT STARTUPS**

When I joined Google Ventures, I thought I had sprints all figured out. But I quickly realized I had a lot to learn. The process had to be changed to reflect the differences between a large company like Google and the startups in our portfolio. For example, at Google, it was easy to get three or four designers together for a few days. At a startup, they might be lucky to have one. So we needed design and critique processes that engineers and CEOs could do as easily as designers.

Startups want to get their product out there quickly and learn what’s working, but it’s costly to launch--you have to write more code, fix more bugs, and handle more issues than with a prototype. So we compressed the sprint cycle even further to get companies faster feedback. I ditched polished Photoshop mockups in favor of quick-and-dirty [Keynote prototypes](http://keynotopia.com/guides/). Michael Margolis tied in his [lightning-fast research techniques](http://www.designstaff.org/articles/recruiting-how-to-find-great-participants-for-your-user-study-2012-02-22.html)to deliver us next-day feedback.

We’re still learning, but we’ve run enough sprints to be confident the process works well.

Stay tuned to this series, and please give me your thoughts along the way--I’m always looking for more tricks to improve what we do. What processes do you use to get good design results? What helps your company move faster?

[**Read the second post here.**](http://www.fastcodesign.com/1672889/from-google-ventures-the-6-ingredients-you-need-to-run-a-design-sprint)

The 6 Ingredients You Need To Run A Design Sprint

GOOGLE VENTURES’ JAKE KNAPP LISTS THE PEOPLE AND THINGS YOU’LL WANT ON HAND TO START TACKLING A BIG DESIGN CHALLENGE.

*[Editor’s note: This is the second post in a seven-part guide on how to conduct your own Google Ventures’ five-day design sprint.]*

At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the second in a series of seven posts on running your own design sprint.

Now that you know what design sprints are good for, you’ll need a few important ingredients to make yours successful. Start with a big, important problem; pitch it to your team; and schedule a user study before you even start. Get the right people and the right supplies in a room and you’re on your way to a successful design sprint.



**1. PICK A BIG FIGHT**

The first thing you need is an important design problem, and if you work at a startup, chances are good you probably have one lying around the office. Maybe more than one. It might be something big, like defining your product for the first time, or a big redesign or new feature. Or it might be something detailed, like improving conversion on a single user action. It just has to be really important to the company, and it has to be something you’re struggling to start or to make progress on--otherwise it can be difficult to get the other people you’ll need involved.

As long as it’s an important problem, it’s perfect for a design sprint. It’s OK if you don’t feel ready to start on it yet. No matter how overwhelming or ambiguous, you’ll be able to cut a big swath through the jungle of possible solutions.

**2. GET THE RIGHT PEOPLE**

The ideal sprint team is between four and eight people, but you can get by with more or fewer than that. Just make sure you have at least one:

• **Designer:** If your startup doesn’t have a designer yet, try to bring in a ringer.

•**CEO:** At a small startup, the CEO is the key decision-maker and needs to participate in order to get an actionable solution out of the sprint. At a bigger company, you’ll still need buy-in and it’s best to include the CEO, but if they can’t be there the whole time, you can bring them in at key decision-making moments.

• **Product manager:** The PM (or whoever is filling this role) will likely need to implement the solution that comes out of the sprint.

• **User expert:** The person on the team who has the most direct contact with customers often has great input, and can be the lead on user testing.

It’s also great to include:  
• Engineer  
• Marketer  
• Anybody else who’s interested

**3. SCHEDULE THE USER STUDY BEFORE YOU HAVE ANYTHING TO TEST**

Once you know when you’re going to do the sprint, recruit users and schedule the user studies for the last day of the sprint. This is a bit terrifying: you haven’t even started to talk about the problem, let alone design solutions, and people--outsiders!--are going to come in and need to be shown something. This hard deadline, even though it’s artificial, will help you move faster and make tough decisions to focus your work throughout the sprint.

**4. FIND A FACILITATOR**

Pick someone to be the facilitator of the sprint. The facilitator is going to be responsible for managing the sprint and moving things along. They need to be confident leading a meeting, including synthesizing discussions and telling people it’s time to stop talking. They also need to be comfortable with not getting to participate as much in the actual design work, because facilitating is a lot of work. Since you’re the one reading this, you may be a good candidate--but it’s always easiest if the facilitator is an outsider. See if you can get a friend from another company to help out.

**5. PUT IT ON THE CALENDAR**

Clear everybody’s schedule for five consecutive days. It’s also very important to have a dedicated room for the duration of the sprint, usually a conference room with lots of whiteboards.  
  
Much of the magic in design sprints comes from the sense of urgency. By their very nature, startups always feel time-constrained; the short, focused time of the sprint adds another constraint.

**6. GATHER THE INGREDIENTS**



Luckily, you don’t need anything fancy to run a sprint. Here’s everything I use:

• **Sticky notes:** I like the yellow [3×5 size](http://www.amazon.com/dp/B00006JN7U).  
• **Drawing pens:** Any standard black or blue pen is probably fine. I like[these](http://www.amazon.com/dp/B002S5387W), or you can get [these](http://www.amazon.com/dp/B0007OEE2E) for extra credit.  
• **Whiteboards:** If your war room doesn’t have a lot of whiteboard space in it, find another war room or some rolling whiteboards, or heck, get some[IdeaPaint](http://www.ideapaint.com/) and get busy.  
• Whiteboard markers: I like to use these instead of Sharpies because they’re so versatile. Buy some [good ones](http://www.amazon.com/dp/B000N35G94) and be sure to have enough[black markers](http://www.amazon.com/dp/B0019DEBS4) for each person in the sprint.  
• **Dot stickers:** for voting. You want something small with uniform color.[Post-It brand dots](http://www.amazon.com/dp/B002M3SBM2) are great.  
• **8.5 × 11 blank copy paper:** Nothing special, just have a pack of this on hand.  
• **Time Timer Clock:** Optional, but totally awesome, [see here](http://timetimer.com/). I guarantee you’ll find it useful during the sprint, and probably during regular meetings afterward.  
• **Snacks:** You’ll need caffeine and food handy. Trail mix, bananas, and dark chocolate covered raisins have proved especially popular at our sprints, although it is possible that it’s just me eating all of it.  
• **Sticky stuff:** You’ll need to stick your drawings and storyboards on the wall. This [removable gummy material](http://www.amazon.com/dp/B0000AQODM) is inexpensive and works great, with less fuss than tape.

OK, the stage is set. Now it’s time to start the sprint.

Stay tuned for tomorrow’s post, in which we’ll explore the exercises we use on the first day of a sprint.

The First Step In A Design: Build Team Understanding

The product design sprint: understand (day 1)

BEFORE EMBARKING ON A DESIGN SPRINT, MAKE SURE THE PLAYERS INVOLVED COME TO A COMMON UNDERSTANDING, WRITES GOOGLE VENTURES’ JAKE KNAPP.

*At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the third in a*[*series of seven posts*](http://www.designstaff.org/articles/product-design-sprint-2012-10-02.html)*on running your own design sprint.*

Now that you know when to get your team together for a sprint and how to set one up, it’s time to tackle the first day of the sprint: understanding.

Chances are that everyone involved in the sprint has different perspectives on the problem — and different information that might be helpful. The goal of the first day is to encourage everyone to share what they already know and develop a common understanding with the rest of the group. By starting at the beginning (even if some people are already familiar with the problem), it nudges the group into a beginner’s mindset and leads to fresh solutions. (This is where an outside facilitator can come in handy: since they’re truly new to the problem, their questions can keep the group in a beginner’s mindset.)

**Use these exercises to help build understanding**

We use the exercises below to get a ton of information on the table and quickly build understanding. You can do them in any order you want — and even omit the ones you don’t find valuable. Try capping each presentation or discussion at 10 minutes. This will keep the day moving and help everyone pay attention.

During the exercises, everyone in the sprint should be jotting down questions on sticky notes. We use the [“how might we”](http://blogs.hbr.org/cs/2012/09/the_secret_phrase_top_innovato.html) format to capture opportunities that might be interesting to explore. For example, “How might we build trust?” or “How might we figure out the user’s style?” Often, these end up being extremely useful in the next steps of the sprint.

**Business opportunity** — The CEO or product leader should walk the sprint team through the business opportunity and market.

**Lightning demos** — Look at competitors’ products. It can also be helpful to look at non-competitive products that solve a similar kind of problem in a different market.

**Lay it out** — Print out all the important screens in your product, lay it out, and walk through it as a user would.

**Success metrics** — How will you measure the success of this design? Now’s a great time to talk about success metrics. We like to use[Kerry Rodden’s HEART framework](http://www.designstaff.org/articles/how-to-choose-the-right-ux-metrics-for-your-product-2012-03-27.html).

**Existing research** — If you have user research for your product, that’s awesome, and you should be sure to go over it. If not, you should talk about whatever data you do know about your customers.

**Team interviews** — Knowledge about the problem is usually distributed across the company. We’ve found it very useful to go around interviewing people at the company who have specific expertise, whether that’s engineering or sales or customer service. (Customer service people often have incredibly valuable information about the problem.)

**Analytics** — Look at any data you have on feature usage, where customers drop off your site, conversion rates, etc.

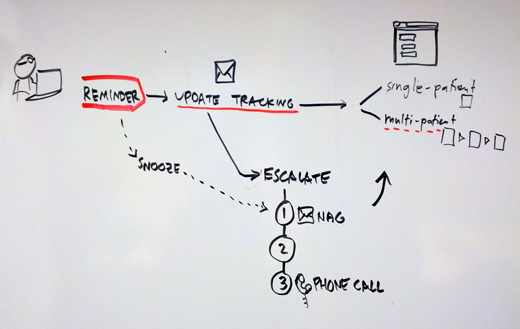
**Feeling overwhelmed? Gather data before the sprint**

If you’re reading this and feeling like the exercises will be tough because you don’t have enough data, don’t worry! That just means you should do some work before the sprint to quickly gather fresh data.

It could be as simple as scheduling [a few user studies](http://www.designstaff.org/articles/research-guide-2012-07-17.html) or deploying a [very short survey](http://www.designstaff.org/articles/microsurveys-2012-02-15.html) with questions related to the problem you’re going to tackle in the sprint. You can also do team interviews ahead of time — this is especially useful if you’ve got a lot of people to talk to.

**Sketch the most important user story**

As a group, use your common understanding to collaboratively map out the user story that’s important in this sprint. The facilitator (or another volunteer) should stand at the whiteboard and sketch the flow. This doesn’t have to be fancy to be useful. Here’s an example:



How do you know which user story is most important? It depends on the problem you are solving in the sprint. For example:

* Helping people understand and get started with your product — you probably want to focus on the experience of a user encountering your product for the first time.
* Creating a new product concept — you probably want to look into the future and imagine the value proposition and core features for an engaged user.
* Improving conversion rate from a landing page — you probably want to understand why people land on your page and what their goals are.

This step can be difficult and time-consuming, but it’s critical! Getting a visual map on the wall (like the one above) is invaluable for grounding the discussion and keeping everyone on the same page.

**Now it’s time to focus**

You’ve got a rapidly approaching user study and there’s a good chance you can’t prototype everything you uncovered, so you’ll need to choose which ideas to move forward. Use your story diagram on the whiteboard to frame this discussion.

At the end of the sprint, when you show your prototype to users, what do you hope to learn? What do you need to design and prototype in order to learn those things? What you decide here will set your course for the rest of the sprint.

The facilitator is responsible for keeping these discussions moving quickly (remember the [timer](http://timetimer.com/)). It’s also important to ask seemingly[obvious questions](http://www.youtube.com/watch?feature=player_detailpage&v=TQFAgUWxnlA#t=10s) because the answers usually further everybody’s understanding.

Now that you’ve fleshed out a common understanding of the problem and started to define which part of it you’re going to tackle in this sprint, it’s time to move on to the next step: rapidly developing as many solutions as possible.

We’ll cover that in the next post, so stay tuned!

The 8 Steps To Creating A Great Storyboard

GOOGLE VENTURES’ JAKE KNAPP TAKES US THROUGH THE LIGHTNING-FAST PROCESS HE EMPLOYS TO GET A TEAM’S IDEAS DOWN ON PAPER

The product design sprint: diverge (day 2)

*At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the fourth in a*[*series of seven posts*](http://www.designstaff.org/articles/product-design-sprint-2012-10-02.html)*on running your own design sprint.*

In the first two days of the sprint, we’ve learned about the problem, shared a lot of knowledge, and chosen the challenge we want to tackle in this sprint. It’s time to start cranking out solutions. Expect this step to take between two hours and all day.

I call this step “diverge” because when everyone (from the CEO to the marketing manager) is cranking out quick sketches, we tend to get a lot of ideas — and different kinds of ideas. Remember in the Legend of Zelda how the map would light up rooms you had visited as you explored the dungeon? That’s what you’re doing on Day 2: illuminating all of the possible paths.

Although you’re going to be generating ideas, don’t think of this as brainstorming — at least not the everybody-is-shouting kind of brainstorming. Instead, everyone in the sprint will be working quietly and individually, often around the same table. The exercises outlined below force you to get ideas out of your head and onto paper, without getting stuck feeling like they have to be finished or perfect.

**Dust off those old ideas**

In my experience, some of the best ideas that come out of sprints were usually around before the sprint started. It’s not that they were bad ideas; they just hadn’t gotten enough love yet. The sprint gives you a chance to put all solutions on a level playing field. If you don’t bring out your pre-existing ideas, you do yourself a disservice.

Because new ideas are so shiny and fresh, the facilitator needs to remind everyone to think old first. There’s no need to be embarrassed of that solution you thought of five months ago while eating a burrito or taking a shower.

**Paper first**

One problem with business-as-usual-design is that companies get in the habit of going straight to high fidelity mockups. In a design sprint, we start designing on paper for a number of reasons:

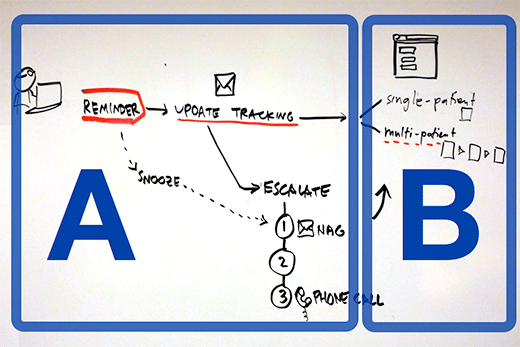
* It’s faster
* Everyone can contribute (not just designers)
* Nobody gets too attached to the ideas that are generated because they’re so quick and rough. We purposefully use [thick markers](http://www.amazon.com/dp/B000WGWJL0/) to make sure nothing gets too precious.
* Did I mention it’s faster?

Run the series of exercises below to guide everyone from note-taking through sketching and sharing. See my [earlier post](http://www.designstaff.org/articles/product-design-sprint-2-2012-10-09.html) for an exact list of the materials you need. I use my trusty [Time Timer](http://timetimer.com/) so everyone can see how much time is left in each exercise.

When I’m facilitating a sprint, I like to remind everybody that we’re not going to share any of the materials until we make storyboards — that’s step five of the cycle — and they’ll have plenty of time to polish those up. I want to make sure everybody feels loose and knows they’re actually going to have plenty of time to work, even though we’re keeping time as we go.

**1. Choose part of the problem**

In Day 1 you drew a [user story diagram](http://www.designstaff.org/articles/product-design-sprint-day-1-understand-2012-10-16.html#sketch_the_most_important_user_story). Look at it together as a team. If the user story is more than two steps long (and it probably is) you’re going to need to divide it up before you start sketching. This is as simple as finding natural chunks in the story and drawing a box around them, like this:



Now decide which part to focus on first. It usually makes sense to have everybody in the sprint focus on the same part of the problem at once. If you take that approach, you’ll do one cycle for each part of the problem, with everybody collaborating on each part as you go.

You can also divide and conquer — everybody picks a piece of the story they’re interested in and works on that. This way is usually faster, although it introduces the risk that people don’t think about the user story holistically. If you have more than two or three chunks in your story, you might have to divide and conquer, or perhaps decide you’re going to focus on a smaller part of the problem for this sprint.

Either way, the facilitator ensures that everybody knows which piece of the user story they’re focusing on before you continue.

**2. Take notes (5 minutes)**

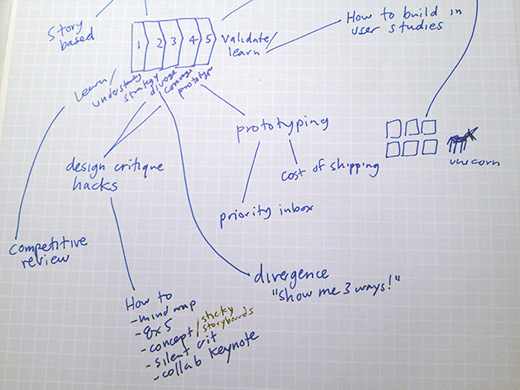
At this point in the sprint, the whiteboards and walls are probably covered in diagrams, notes, and [“how might we” sticky notes](http://www.designstaff.org/articles/product-design-sprint-day-1-understand-2012-10-16.html#use_these_exercises_to_help_build_understanding). This is your chance to reload that stuff into your brain. Everyone takes a piece of paper and jots down anything they think is useful.

**3. Mind map (10–15 minutes)**

Now you’re going to add all the other ideas that are in your head, mix them with the notes you just took, and loosely organize them on paper. The mind map is going to be your “cheat sheet” you can use when you’re sketching UI ideas.

If you’re not familiar with mind mapping already, I often describe it as writing down everything in your head with no specific formatting; or quiet individual brainstorming. You can write words and connect them or not, you can draw pictures or not — you basically can’t do it wrong. The important thing is that everyone is getting every solution, old and new, out of their head and onto paper at very low fidelity.

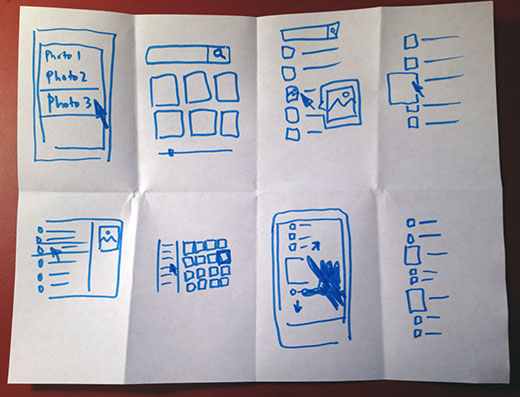
Here’s an example:



**4. Crazy Eights (5 minutes)**

Everybody folds a blank sheet of paper in half four times, then unfolds it, so they get eight panels. Then you have 5 minutes total to draw eight sketches, one in each panel. Yes, you did the math correctly, that’s about 40 seconds per sketch, which is crazy… but it’s a great way to crank out variations of ideas quickly. And since these aren’t shared with the group, there’s no need to worry about making them pretty.

Since you only have 40 seconds for each drawing, you’ll need to turn off the self-editing and just get your ideas on paper. Crazy Eights will also help loosen up your creative muscles and make you more productive in subsequent sketching exercises. If you get stuck, try repeating an earlier sketch with a small variation — this type of exploration is useful and it keeps you moving.



For best results, do two rounds of Crazy Eights. On the second round, everyone will have the hang of it. You’re scraping the bottom of the barrel, which makes it more painful to come up with new ideas, but often this is where the most interesting solutions come from.

Now you may be thinking I’m a bit of a hypocrite: earlier in this post I said old ideas are best, and now I’m asking you to come up with new ideas. Don’t get me wrong, it’s OK to fill out your Crazy Eights sheets with old ideas. But new ones are good too — just because old ideas tend to be stronger doesn’t mean they always win.

Pro tip: Get the [Bit Timer app](http://www.bittimerapp.com/) for your iPhone and set it to 30-second work periods and 10-second rest periods for eight reps so you don’t have to time it yourself. The rest period alarm acts as your 10-second warning to wind down your current sketch.

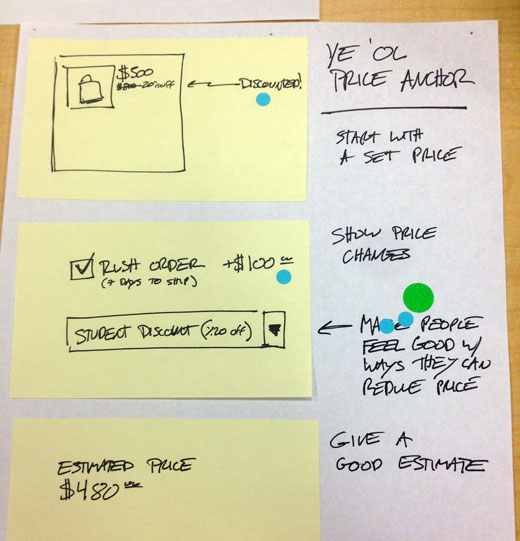
(Crazy Eights are based on the [685 exercise](http://www.gogamestorm.com/?p=688) introduced to us by Brynn Evans.)

**5. Storyboard (10–20 minutes)**

Now we’re going to make that user story diagram more concrete, and we’re going to make something that will be shared anonymously and critiqued by the group. The goal is to take the ideas we’ve generated so far and sketch an actual UI showing how a user would move through this part of the story — where they click, what info they enter, what they think, etc.

Start with a blank sheet of paper, and put 3 sticky notes on it. Each sticky note is one frame in the storyboard. It’s kind of like a comic book that you’re going to fill in.

Look back at your mind map and your Crazy Eights and find the best ideas. Chances are, you’re itching to illustrate at least one of them in more detail. Now you’re ready to rock. I ask everybody to draw UI in the three frames of their storyboard showing a progression: first this, then that, then that.



There are three important storyboard rules:

* Make it stand alone — Just like a real product, your drawing has to make sense by itself, without you there to pitch it. In the next steps, people will be looking at these, but you won’t have a chance to talk about your idea until the end.
* Keep it anonymous — Don’t write your name on your drawing. You’ll want all ideas to start on a level playing field and it can be distracting to know which one was drawn by the CEO.
* Give it a name — Come up with a catchy title for your idea. That makes it easier to discuss and compare later.

When you finish the storyboards, hang them on the wall with some[sticky stuff](http://www.amazon.com/dp/B0000AQODM). Pro tip: Hang them side by side (like an art museum) so people won’t have to crowd in too tight to see them.

**6. Silent critique (5–10 minutes)**

Give everybody a bunch of [dot stickers](http://www.amazon.com/dp/B002M3SBM2). Then, without speaking, everybody looks at the different storyboards and puts a sticker on every idea or part of an idea they like. There are no limits to how many stickers you can use, and I don’t even prevent people who want to brazenly vote for their own ideas. By the end, you’ve got a kind of heat map, and some ideas are already standing out.

**7. Three-minute critiques (3 minutes per idea)**

Next, everybody gathers around the storyboards one at a time. First, people talk about what they liked, then we ask the person who drew it if we missed anything important. Usually the best, most popular ideas are the ones people can understand without an explanation, so the author of the storyboard often doesn’t have anything else to add. This process works far better than letting people explain their ideas first — which almost always uses up a lot of time.



Sometimes I like to do this step on a projector, especially if there are a lot of ideas to get through. I’ll take photos of each storyboard on my phone, upload them to Dropbox, put them in a Keynote file, then make notes about parts we like with outlines and text labels as we go through on the projector. This is easier for everyone to see, and you have a digital artifact of the ideas for later. The downside is the setup: count on 15 extra minutes to capture and upload photos.

**8. Super vote (5 minutes)**

Once we’ve looked at all the ideas, everybody gets one or two “special” stickers (which can be the same dot stickers from before with a pen mark on them). These are “super votes” for the ideas you think are the very best. Between the original heat map and these super votes, it’s very easy to see which are the strongest concepts.

The super votes offer a unique way to tweak the process to reflect the decision-making structure of your team or company. Does your CEO make all final decisions about the product? If that’s the case, be honest about it, and give her three super votes and everybody else one. Or maybe it’s a UX director or maybe a tandem of product and design who call the shots. The simple rule is to give the deciders extra votes.

By default, this process will be a meritocracy, but that’s not always the way companies work and, frankly, consensus can lead to poor design decisions. The last thing you want are decisions that the deciders don’t truly support. On some teams, these may be unwritten rules, so don’t be surprised if it feels a bit awkward to bring it up — in the long run, you’ll be glad you did.

**Repeat**

Now it’s back to the first step to start the whole cycle over again. (Don’t worry, it gets easier with every repetition.) If you split up the user story last time, it may be time to move on to another chunk. Often when I’m running a sprint I’ll realize at this point that our scope was too large, and we should just double down and keep working on the same section. Either way, the end of a cycle is a good time to take a few minutes and carefully decide where to focus next.

Expect a team to be able to do this cycle two or three times in a day before getting burned out. Throw in plenty of breaks and snacks to keep the troops moving.

Stay tuned for the next episode, where we’ll talk about how to decide which pieces of these storyboards go into your prototype.

How To Decide What Ideas To Prototype

INSTEAD OF DEFAULTING TO GROUPTHINK, WRITES GOOGLE VENTURES’ JAKE KNAPP, MAKE SURE YOU HAVE THE BUY-IN OF THE HIGHER-UPS--OR YOU MIGHT REGRET IT LATER.

The product design sprint: decide (day 3)

*At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the fifth in a series of seven posts on running your own design sprint.*

At this point in a design sprint, you’ve got a lot of ideas down on paper. You’ve explored the problem, generated a ton of solutions, and looked around at how other companies are solving similar problems.

It’s awesome to have a lot of ideas. It’s a great feeling. But I’ve got bad news: you can’t build and test everything. And even if you could, it wouldn’t be very useful, because you’d have too much information to sift through. So you’ve got tough decisions to make: which solutions will you pursue and which will you put on ice?

Today we’ll look at how to decide which solutions to flesh out, and how you’ll fit them together into something you can rapidly test with users to learn what’s working and what isn’t.

**Combat the group effect**

The decision-making process is hard, and this is one place where working as a group can become a liability. Companies and teams have a natural way that they make decisions — but in a sprint, the group effect can cause decision-makers to behave more democratically than they do in real life. Once the sprint is over and that rosy democratic feeling wears off, you can be left with something that doesn’t have true support from the deciders.

To combat this effect, the facilitator often has to draw out the decision-maker to give their honest, true opinion. You’d be surprised at how often this reluctant decision-maker is the CEO. No way, really? Yes way. In the sprint, people are out of their comfort zones, and even CEOs can begin to behave in non-standard ways.

One method is giving “super votes” to the deciders during design critiques, which you did in day 2. But most of the time, there are no special techniques. You just have to be blunt.

As facilitator, you should be upfront with the team if you sense you’ve got a case of groupthink. Let everyone know that you need more assertive participation from the deciders. Additionally, you should have the words “make the call, Sally” on the tip of your tongue throughout day 3. (I’m assuming your CEO’s name is Sally.) Don’t worry about being a sycophant. If you aren’t conscientious about bringing the decider in now, you’ll have a problem later.

**Search for conflicts**

The first thing I like to do in this phase of a sprint is comb through storyboards from the previous day looking for conflicts. A conflict is a place where there are two or more different approaches to solving the same problem. Conflicting approaches are super helpful because they illuminate the choices for your product.

For example, let’s say you’re designing your home page, where you explain your product to potential customers. Maybe one person’s storyboard uses a video, and another uses diagrams on a long scrolling page, and a third uses a single image of the product. Great, you’ve found a conflict! Every time you find one, write it down. I like to put the topic and solutions on sticky notes, like this:



Each conflict is like a little gold mine. In business-as-usual design, designers often end up picking one approach and going straight to high resolution. When I was working on products in-house, I’d often get so caught up in that one solution that I wouldn’t even have time to think about how else it could be done.

So one of the best things about the product design sprint is that it allows you to map out those decision points, and perhaps even to explore a few conflicting ideas in parallel instead of immediately committing to a safe choice.

**Best shot or battle royale?**

You have two basic options for what kind of user study you’re going to run at the end of your sprint. You can prototype several different approaches and test them against each other (the “battle royale”) or you can go with a single prototype (the “best shot”).

The advantage of the “best shot” approach is that you can put a lot more work into that one prototype, or just get it done faster. If you’re testing only one solution, the user study is less complex, and it gives you more time to see what the users say about your competitors’ products (or just interview users, which is always surprisingly valuable for teams).

The “battle royale” works well for newer spaces where there really aren’t many conventions, and you need to figure which one is going to work best for the user. The disadvantage is that it takes more time, and your testers may run out of patience before you get all of the information you’d like to have from them. You may have to bring in more participants and run more studies.

On the upside, the results of a “battle royale” can be very surprising — when working with startups, I’ve often seen a dark-horse design turn out to be the strongest in user studies. When that happens, we thank our lucky stars we didn’t “best shot” it, or we never would have known.

You may also do some kind of hybrid. Occasionally, if you choose the “best shot” approach, you’ll get into testing and find that something’s really not working in your prototype, and you need to go back and have a “battle royale” over that specific feature.

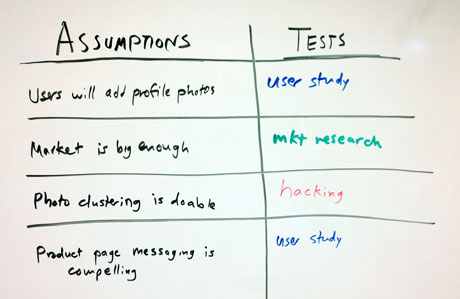
So how do you know which to pick? Start with a gut check: if everyone is excited about one option, you may be ready for a “best shot”. But if it feels more like you’re sitting there and scratching your heads about what to do next — or else you want to throttle each other because those fools *just won’t agree with you* — well, you may need a “battle royale.”

**Test your assumptions**

What else should you test in your user study? Listing out your underlying assumptions is a good way to revisit the big picture, especially when you’ve been heads down in a sprint for a few days.

Some of those assumptions might be about the users (example: “users are willing to upload a profile photo”), some about the business (“the designer-with-glasses-and-beard market is large enough to support our product”), some about technology (“we can automatically cluster profile photos by beard shape”), and maybe even some about messaging (“people will still find beard jokes to be amusing, even for the third time in a single paragraph”).

I can tell you that last assumption is false right now, but for most others, you’re going to need some kind of research. Guess what? You can test a bunch of them by showing a prototype to users.



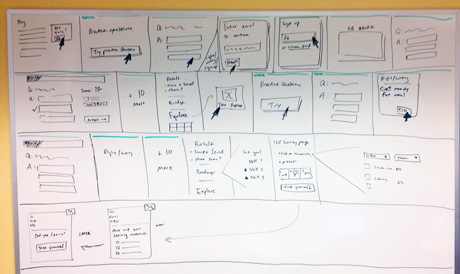
For example, if you have a big assumption that users will be comfortable sharing private data in your product, you may want to pick the most aggressive sharing defaults you can think of to prototype. When you show it to users, you’ll find out pretty quickly whether your assumption was correct.

Try to come up with a way to test all your assumptions, either in the user study or in some other parallel task that can start right away (e.g. ask the engineers to spend a few hours hacking at that beard-clustering algorithm). If you can’t test every assumption now, keep a list for next time. Untested assumptions are like takeout containers in your fridge: if you leave them for very long, things get nasty.

OK, you’ve picked which conflicts to explore and you’ve decided which assumptions to test. Congratulations — you’re ready to script your prototype.

**Whiteboard the user story**

Now we’re going to make a storyboard that shows exactly how the user will step through your prototype, click by click. This storyboard will become the spec for building the prototype. This is an activity that the group does together — it’s actually the last group step before you break for prototyping.



Start by drawing a big grid on the whiteboard — each cell should be about as large as two sheets of copy paper, and for most sprints, you’ll cover one or two whole whiteboards with your grid. The idea is to draw [a comic book that tells a story](http://www.amazon.com/Understanding-Comics-Invisible-Scott-McCloud/dp/006097625X) starting when the user opens the prototype and ending when they complete all necessary tasks.

In each comic book frame, you’ll draw a single action — whether it’s a pointer clicking on a button, text being entered, or a stick figure user doing something in real life. You don’t have to worry about layout or design in great detail, but you do have to think through every action that takes place in the story.

Drawing the storyboard is hard work, and you’ll want to facilitate carefully. Get one person to draw, but don’t make them figure everything out on their own. The group should be engaged and discussing what happens next and giving that brave soul holding the whiteboard marker as much help as possible.

When you begin drawing, imagine you’re framing the prototype for your user study participant. How will they get to your product? What will they be trying to do when they get there? That’ll help you figure out whether the first frame of your comic book is an email or a Google search or an advertisement or the App Store or whatever — and hopefully the story will flow easily from there, following the[outline you laid out in day 1](http://www.designstaff.org/articles/product-design-sprint-day-1-understand-2012-10-16.html#sketch-the-most-important-user-story).

**Keep the gloves off**

As you storyboard, there will be lots of small decisions to make that didn’t come up earlier in the day. That’s expected, since you’re working at a finer level of detail now. The facilitator has to work hard here to not let people be too nice. You don’t want design by committee. If there’s a good argument going, don’t try to find middle ground or make people agree. Help the team place a bet on one of the opposing solutions and keep the other in your back pocket if it fails. Call on the CEO to make a tough call when needed. If both solutions are viable, you may want to opt for a “battle royale” — just don’t use it as an excuse to avoid decisions.

When you’re finished with the user story, take a moment to pat yourselves on the back and eat some chocolate because it’s probably been a pretty epic task. You’ve given form to everything you want your user study participant to experience, and you’re ready to turn that story into a higher-resolution mockup.

In the next post, we’ll move on to prototyping. It’s time for the Fellowship of the Sprint to break up, at least temporarily, as everybody puts on their headphones and cranks out a crafty imitation of a real product. Stay tuned.

A Lightning-Fast Way To Make A Digital Prototype

A STEP-BY-STEP GUIDE TO TURNING A STORYBOARD INTO A PROTOTYPE FOR USER TESTING, FROM GOOGLE VENTURES’ JAKE KNAPP

The product design sprint: prototype (day 4)

*At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the sixth in a series of seven posts on running your own design sprint.*

On day 2 you drew concept sketches. On day 3, you made a plan and a storyboard for your prototype. Now it’s day 4 and the clock is ticking. You’re going to create a real-looking version of yesterday’s storyboard and show it to users tomorrow.

This part of the sprint is super exciting for me as a designer. Thanks to the storyboard, I know exactly what to do, and I also have a crazy deadline to get it done. It’s finally time to to open my laptop, put on my headphones, and start moving pixels.

But wait a second… what should this prototype look like? Are we going to have to code anything?

**What your prototype should look like**

Quite simply, a prototype is anything a person can look at and respond to. A prototype doesn’t usually have to be very complex in order to learn what you need to know.

**Make it minimally real**  
You’ll probably be amazed at how much real feedback a user can give you on a slide deck of mockups that aren’t even pixel-perfect.

They can tell you what they understand about your product — and what they don’t. They’ll tell you what they expect things to do (“I’d click here because I’d want to see a list of your customers…”), and when they get confused.

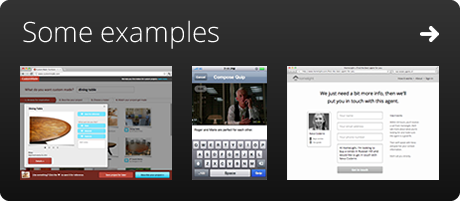
You’ll also learn things that metrics alone can’t tell you, in particular why users do the things they do, rather than just what they do. And you’ll learn much faster than if you had waited to build something.

**Write real text**  
While it can be tempting to use “lorem ipsum” when you’re building a quick prototype, don’t do it — always write real text for your prototype.

Why? First, user interfaces are mostly text. Punting on the text might save you time, but it won’t get you closer to solving your problem. Plus, in design sprints we’re often figuring out how to explain things and testing whether users understand. Lorem ipsum skips all of that. When you use dummy text, you avoid tough decisions and limit how much you can learn. (37signals has a [great essay on using real words](http://gettingreal.37signals.com/ch11_Use_Real_Words.php) in their book Getting Real.)

**Keynote versus code**  
Occasionally you’ll need to write some code for your prototype. This can be really valuable when you’re testing assumptions relating to data quality or personal content like email. But nine times out of ten we find that Keynote is sufficient. More on this later.

**Here are some examples**  
[We](http://www.googleventures.com/hands-on#design-studio) have created dozens of minimally real prototypes for the startups in [our portfolio](http://www.googleventures.com/companies). Here are a few examples, covering the spectrum from grayscale and low-fidelity to nearly pixel-perfect.

[](https://www.dropbox.com/sh/b5le4kch8m3eor8/aZ6qZcUBTk/Design%20Staff%20Prototypes)

**Keynote is the world’s best prototyping tool**

If you’re building a product for desktop, mobile, or iPad, the fastest and easiest prototyping tool you can use is [Keynote](http://www.apple.com/iwork/keynote/). I’ve spent my career working with fancy tools like Photoshop and Illustrator and Fireworks. (OK, Fireworks might not be fancy, but it’s almost certainly a tool.)

Here’s why Keynote wins for prototyping:

* It’s fast.
* It’s easy to make things look pretty good…
* But it’s impossible to make things look perfect, so you don’t get too precious.
* Anybody can quickly learn to use it; not just designers.
* The slideshow format is a natural fit for story-based design.
* It only costs $20.
* The animated transitions are a lightning-fast way to make your prototype look way more real than it really is.
* If it’s a mobile prototype, you can export a PDF and open it on your device with reasonable results.
* When you’re done, you not only have a prototype, you have a presentation, like, for free!
* After your user study, when you learn all of the problems in your design, it’ll take minutes to make changes instead of hours or days.

(“Wait,” you say, “I use a PC.” Wow, you just blew my mind. Almost everybody I meet who works at a startup has a Mac, but true, there are exceptions. If I stereotyped you, I am deeply sorry. Let me make it up to you by pointing you to Henry Tsai’s excellent post on[building PowerPoint prototypes](http://www.designstaff.org/articles/prototype-clickable-mockups-powerpoint-2012-11-06.html).)

The final Keynote (or PowerPoint) prototype might be linear or it might have a few links that allow the user to click around. It might be a grayscale wireframe or it might be somewhat polished. Either way, these make for good prototypes. They’re going to be realistic enough that your user study participants will forget they aren’t clicking on a real product.

**Keynotopia templates make it even faster**  
Here’s our favorite secret prototyping weapon: [Keynotopia](http://keynotopia.com/themes/). It’s a template with buttons, menus, and all kinds of plug-and-play elements that you can just drop into your prototype without any design skills at all. This makes it easier for CEOs and marketing directors to dive in and make something to communicate what they’re thinking. And it’s available for PowerPoint too, if that’s how you roll.

**All right already, I bought Keynote, now what?**  
You made a wise choice, my friend, and the prototypes are going to start flowing like milk and honey.

**What to do in day 4**

Here’s a rough schedule for your day of prototyping. Expect to spend about an hour in the morning to review and make your plan.



**Divide and conquer**  
What if you have too much to design and not enough time? Work together, baby.

Take a look at your team. How many people can help in Keynote? Chances are good that almost everyone can — in our sprints, we frequently have engineers, PMs, and CEOs cranking out good work in Keynote. Remember that everybody’s work doesn’t have to be genius level. A designer can always clean it up afterwards, but it’s usually faster to clean up a slide after someone else has put the building blocks in place.

Now take a look at yesterday’s [storyboard](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html#whiteboard-the-user-story) — or, if you’re doing a “[battle royale](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html#best-shot-or-battle-royale),” storyboard*s* plural. Break it into chunks and assign them out. It’s probably obvious, but it’s helpful to think about where different people have expertise, or where it’s best to spend your designer’s time, or who moves the fastest.

You probably want to divvy up the best of the storyboard sketches on paper, too. Those are often the blueprint for many of the mockups you’ll have to make, and they can save you a bunch of time.

Assign one person to be the stitcher. The stitcher’s job is to take everybody else’s work and put it into one cohesive flow. If you only have one designer, or if one person is really fast at Keynote, she might be a good candidate. If you’re doing a “battle royale,” you may decide to have one stitcher for each competing version.

Breaking up the design work and stitching it back together probably doesn’t sound like the way great design is done. And it’s not. Realistically you’ll get better results if you’ve focused enough that your designers can do all the work here. But if you need it, dividing and conquering is an incredibly fast way to build a good-enough-to-learn-from prototype.

**Build an asset library**  
Another way to increase efficiency is to take a few moments at the beginning of the day to build a template slide deck. Include anything that everyone will need — screenshots, user avatars, logos, formatted text; whatever you think might help. And don’t forget to include a browser bar at the top for realism. You don’t want to go through 99 slides at the end, scooting each one down to make room.

**Use a timer to maintain focus**  
Once again, the [Time Timer](http://amzn.com/B000J5OFW0) is your friend — although if you don’t have one, I guess some other kind of clock might work. We’ll often say something like: “Let’s work heads-down for the next hour and then we’ll take a break.” Timeboxing can make the project feel less daunting, and reduce the impulse to check your email and burn up valuable make time.

**Appoint an email sheriff**  
It’s sometimes helpful to appoint someone — usually the facilitator — as the “email sheriff.” This person’s job is to publicly shame anyone who is checking their email (or surfing the web or whatever time-wasting tactic you prefer). It’s usually enough to just make a big deal about saying that someone is going to do this… the threat alone practically makes the job unnecessary.

**Lightning critique**  
If you have a bunch of people working in parallel, it might be useful to do a quick critique midday. It’s useful to ensure consistency and get outside eyes on your design. But critiques can eat up a lot of time if you aren’t careful. Just like on day 2, you’re going to want to limit the time spent talking about each design. I recommend 5 minutes.

You’re also going to want to prevent people from trying to redesign somebody else’s work — they might have a great idea, but there just isn’t time in this part of the sprint. Instead, use the lightning critique as an opportunity to raise questions and criticisms, but make it clear that the person who designed those components is going to be responsible for figuring out the solution, not the group.

**Review with an outsider**  
Schedule 30 minutes with someone who is *not* doing design work today. It might be your user researcher if you’re lucky enough to have one, or it might be an executive — seems like everybody’s got one or two of those hanging around. The outside eyes will help prevent you from going too far down any groupthink rabbit holes. Just make sure to do this early enough in the day that you have plenty of time to respond to the feedback afterward!

**Pointers, text, and other final touches**  
If you’re creating a linear prototype with no links, which is the very fastest thing to do, it’s a good idea to draw a mouse pointer and add extra slides showing where the user would click, and extra slides when text is entered. It takes only a little extra time but makes the user study much more realistic.

Other crucial details:

* Check for consistency and typos, especially if you’ve got some made up user data. Don’t let it start out as Sally and end up as Suzy. That stuff is distracting in a user study.
* Try to make any content current and relevant. If you’re running the study in Seattle, and you need to show a newspaper, show the Seattle Times, not the Milwaukee Journal Sentinel.
* If you get stuck, remember what you’re trying to learn — don’t waste 30 minutes tweaking a button style if you’re doing a study about whether people understand the value proposition.

**When Keynote isn’t enough**

Nine times out of ten, you can learn everything you need to know in a user study with a click-through of mockups.

But sometimes you can’t avoid it. When you’re testing big assumptions involving data quality, or when users need to interact with their own stuff (email, docs, contacts, etc) to really understand the product, there’s no substitute for a code prototype. In this case, get someone with engineering chops to help.

Just make sure everybody, especially the engineer, knows that this code will be throwaway. That’s really important: The. Code. Is. THROWAWAY. We gotta move fast, so don’t get attached. Like the Keynote prototype, it just has to look sort of real, it doesn’t have to be anywhere near perfect.

**It’s OK if you’re not satisfied, Michelangelo**

It’s better to be done with something good enough than to be half-finished with a masterpiece. Remember that the goal is to learn from the user study tomorrow, not to have everything perfectly figured out and finished.

Stay tuned for the next — and final! — installment of the series, Day 5: Validate.

And in the meantime, what tricks do you use for super rapid prototyping? Do you have a tool that you swear is better than Keynote? I’d love to argue with you about it (respectfully, of course) in the comments!

Got A Bright Idea? Test It With A Rapid-Fire User Study

IN THE LAST PART OF A SEVEN-POST SERIES ON RUNNING A FIVE-DAY DESIGN SPRINT, GOOGLE VENTURES’ JAKE KNAPP DISCUSSES HOW TO GET FEEDBACK FROM POTENTIAL USERS.

The product design sprint: validate (day 5)

*At the Google Ventures Design Studio, we have a five-day process for taking a product or feature from design through prototyping and testing. We call it a product design sprint. This is the final in a series of seven posts on running your own design sprint.*

As day 5 of the product design sprint dawns and the team files into the war room, there’s a certain something in the air. Is it the chemical scent of the whiteboard markers? Maybe. Is it coffee breath? Yeah, almost for sure — in fact, you might want to hand out some gum.

But there’s also something else: anticipation. Today, we test our prototype and learn which ideas worked, which didn’t, and what to do next.

Today you’re going to be running a user study, showing your prototype to 4–6 real humans. By “real humans,” I mean people who don’t work at your company — more specifically, people who you’d like to have as users. [Learn how to recruit great participants for your study here](http://www.designstaff.org/articles/recruiting-how-to-find-great-participants-for-your-user-study-2012-02-22.html).

**Wait, what if I don’t know how to run a user study?**

Bad news: This post won’t teach you how to talk to users. This post is for the people watching the study, not for the poor soul conducting the interviews.

Good news: If you’re the interviewer, you can get some great advice from our [research guide](http://www.designstaff.org/articles/research-guide-2012-07-17.html). Be sure to check out Michael Margolis’ articles on [interviewing tips](http://www.designstaff.org/articles/get-better-data-from-user-studies-interviewing-tips-2012-03-07.html) and [body-language hacks](http://www.designstaff.org/articles/body-language-2013-01-24.html), and his [video about running user studies](http://www.youtube.com/watch?v=WpzmOH0hrEM) from the Google Ventures Startup Lab.

And you might as well read this post, too, so you know what’s going on in that observation room. I went to all the trouble of typing the whole thing up, so it’s kind of the least you could do.

**Before the first session: List your key questions**

Start at the end. When the day is over, you’ll want a concrete list of ways to improve your prototype. Talking about your game plan before the study starts will help you get there.

The interviewer and the observers should make a list of the key questions for the day. Here are a few tips:

* Review your [conflicts](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html#search-for-conflicts) and [assumptions](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html#test-your-assumptions). Those assumptions you flagged as testable with a user study? Now’s the time!
* Are you testing multiple prototypes in a [battle royale](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html#best-shot-or-battle-royale)? If so, be sure the interviewer understands the differences between the two versions, so they can ask the right questions.
* Consider showing participants some real products for comparison — they’re like [free prototypes](http://www.designstaff.org/articles/free-prototypes-competitive-research-2012-11-29.html)!
* What else do you want to see through your users’ eyes? Today’s study is not a usability test — you have a chance to find out how your users understand your product, what competitors or substitutes they use, and more. Think beyond the prototype and you are guaranteed to get some unexpected insights.

**Set up the observation room**

Everybody who participated in the sprint should be in the room. There’s no substitute for watching real humans use your product, and this is a golden opportunity to do it!

I like to reserve two rooms for the day: one for the user interviews, and another where the sprint team can watch live video of the sessions and take notes together. It’s kind of like a Battlestar Galactica marathon, only instead of comparing guesses on who’s a Cylon, you’ll be comparing guesses on which parts of your design are going to go up in flames. It’s actually pretty fun.

**Test the A/V ahead of time**

Setting up live video for observation shouldn’t be too hard — at the[Google Ventures Design Studio](http://www.googleventures.com/hands-on#design-studio), we’ve had success with WebEx, GoToMeeting, Apple Airplay, and Google Hangouts.

Just be sure to test your video before you start the study — ideally the night before and the morning of — because nothing’s worse than putting in all this work and not getting to watch the outcome. OK, there are a few worse things. But you see what I mean.

A few more A/V tips:

* A [conferencing mic](http://www.amazon.com/dp/B001TGTDFM/ref=cm_sw_su_dp) can provide a huge improvement in sound quality.
* Don’t forget to mute the audio in the observation room!
* Quickly re-test the A/V between each session. Seriously, stuff gets wacky.

**Don’t diss the user**

If you see people struggle to understand the prototype, keep in mind that we’re testing your design — not the participant. If they don’t get it, it’s not because they’re dumb. It’s because you haven’t nailed the design yet.

Make it clear in the observation room that it’s not OK to diss the participant. It’s tough to wade through a prototype while people are watching. You owe the participants your respect.

**Every observer takes notes**

Everybody should take notes on things they see during the interviews: good, bad, and other. Insist on paper note-taking — it’s best to keep laptops closed, lest you lose your fellow observers to email.

**Designate a court reporter**

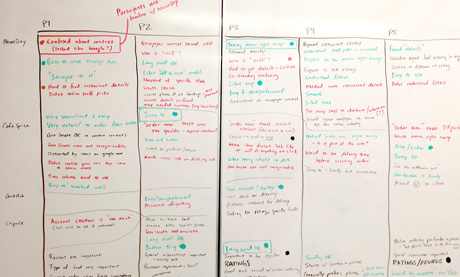
One person can use a laptop in the observation room, but they have a tough job: the court reporter. They’re responsible for typing a word-for-word transcript of the interview in real time. Assign a different court reporter for each session — it’s cruel and unusual to make the same person do it all day.

It’s kind of a pain, but these notes become an incredibly valuable reference after the study, and a text document is a heck of a lot faster to scan for quotes and reactions than a video recording. Often times, we don’t record the study and rely on these notes instead.

**Make a scoreboard**

Clear one big whiteboard to collect the group’s notes. Make a column for each participant and a row for each part of the interview (e.g. background, first prototype, second prototype, etc).

At the end of each session, write down the highlights from everybody’s notes — you can double check any questions against the transcript. I like to color code: green for things that went well, red for problems, and black for everything else. It makes it easier to find patterns at the end of the day.



**Observing humans: the emotional roller coaster**

I’ve never been in mission control when one of those rover things lands on Mars (NASA isn’t returning my calls). But I imagine it’s pretty similar to the atmosphere in the observation room of a user study. There’s tension and excitement and nobody wants to be the one who designed the doohickey that blows up. Here’s what you might expect to feel throughout the day.

**First session: “We’re geniuses!” or “We’re idiots!”**

The first user may love your prototype. Or they may hate it. Either way, take a deep breath. People are different, and I offer you an ironclad guarantee that not everyone will react in exactly the same way.

Regardless of how that first interview goes, resist the urge to make changes to your prototype. Unless it’s something simple like a typo or broken link, you risk “fixing” something that the next four users would have actually liked.

**Sessions 2–4: “Oh, this is complicated…”**

It’s not uncommon for the second and third participants to have dramatically different feedback, which means you’re going to feel a little confused. Just sit tight and keep taking notes. It’s still too soon to tell.

Actually, that’s not entirely true. Sometimes you know for sure that part of your prototype is rotten after two or three interviews, and watching more people suffer through it is punishment for all involved. If everyone agrees that something is way off, talk to the interviewer in between sessions and ask them to guide people around that part of the prototype.

**Studies 5-6: “There’s a pattern!”**

After the final interviews it’s easy to see the big patterns, but it’s worth double checking the notes. Go to the scoreboard and look for things that showed up two or more times. Mark good stuff with a big green dot, and bad stuff with red.

Now make two lists on the whiteboard: “things that work” and “problems to solve.” These are your top-line findings. The CEO or decider for the project should bless that list before you leave the room.

The sprint is complete. Take a deep breath.

**How to start your next sprint**

The vast majority of these studies ends with mixed results. Some of your solutions work, and some don’t. The outcome usually falls in one of three buckets:

**A. Most stuff worked**

This is pretty uncommon for the first sprint on a project, but if it happens to you, everyone on the team is probably on the same page about the fixes and tweaks you need to make.

*What to do next:* Tune your existing prototype and keep going. Try starting your next sprint at [step 3 (decide)](http://www.designstaff.org/articles/product-design-sprint-day-3-decide-2012-11-20.html).

**B. Some big questions**

The most common outcome after a user study is a mixed bag: a few hits, a few tweaks, and a couple of real head-scratchers. Fortunately Keynote prototypes are easy to change, and as long as some parts of your design are solid, you can probably build on what you have.

*What to do next:* You can move fast on the tweaks, but you’ll want to come up with multiple solutions for the bigger problems. Start your next sprint at [step 2 (diverge)](http://www.designstaff.org/articles/product-design-sprint-day-2-diverge-2012-10-26.html).

**C. Everything exploded**

I’ve seen a lot of of my designs go up in flames. It’s OK. You learned that something didn’t work, and it only took you a few hours to build it in Keynote. This is great progress and — relative to building and launching for real — very cheap progress at that. Think what would have happened if you’d spent weeks or months implementing this solution!

*What to do next:* Start your next sprint back at the drawing board with [step 1 (understand)](http://www.designstaff.org/articles/product-design-sprint-day-1-understand-2012-10-16.html). (Hint: the results of this study are perfect fodder for [reviewing and building understanding as a group](http://www.designstaff.org/articles/product-design-sprint-day-1-understand-2012-10-16.html#use-these-exercises-to-help-build-understanding).)

**The next sprint will be easier**

You may feel tired. Actually, if you don’t feel tired, you probably didn’t do the sprint properly. It’s an intense week. But you’ve built up a tremendous head of steam — even if everything exploded, you now have a much better understanding of the problem and possible solutions.

At the end of a sprint, CEOs often tell us they appreciate that they have a clear list of what to do next. Now that you’ve built up all that knowledge and momentum — not to mention a prototype — you’ll find you can do the next sprint more quickly, or with less effort, as long as you do it right away. Don’t let more than another workday go by before you jump back into it. If this problem is important, you’ve got to bear down and finish it off before people get distracted.

Thanks for reading through this series! We keep experimenting and we keep learning new things about our sprint process at the Google Ventures Design Studio. We’ll continue to share what we learn here at Design Staff.

I’d love to hear about your own experiences with product design sprints, or other processes that work well for you. How do you hack your process to get better results? What’s working? What still keeps you up at night?