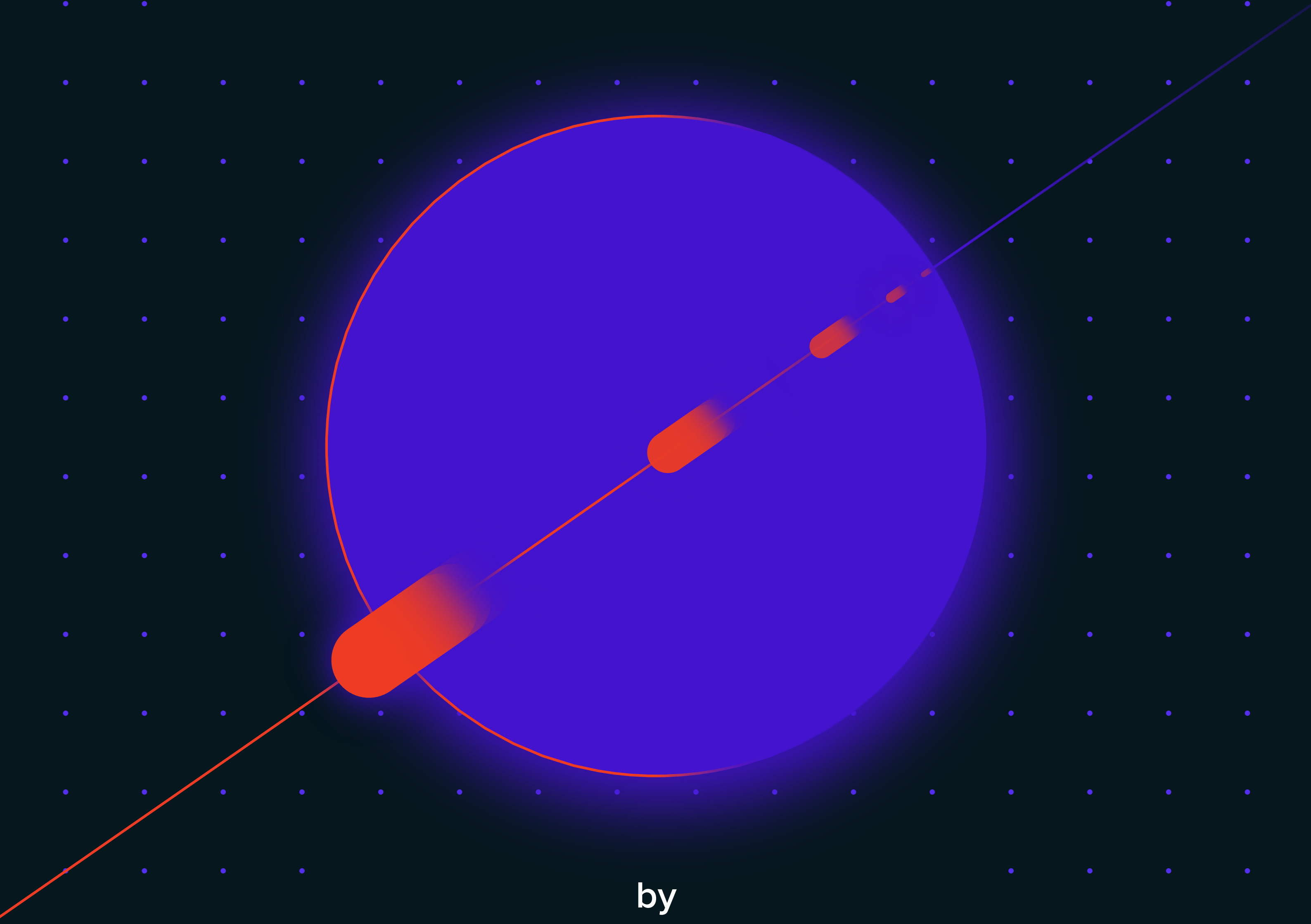
UXPin

The Guide to Agile UX Design Sprints

A Playbook for Product Teams



by
PwC Digital Designer
Alex Gamble

UXPin

The Guide to Agile UX Design Sprints

A Playbook for Product Teams

by
PwC Digital Designer
Alex Gamble

Index

Part 1: Set Up to Succeed

Part 2: Executing UX Opportunities

Authors

Alex Gamble is a product designer at Price Waterhouse Coopers New Zealand. He has helped a variety of businesses, from small start-ups to big corporates, develop user-centred products. Alex's goal is to bring forward a lean product revolution.
Follow him on twitter or email him on alex.x.gamble@nz.pwc.com

Alex would like to dedicate his work to his Mother, Elizabeth Gamble, who he owes everything to.

Introduction

In today’s world of Lean and Agile product development, we need to speed up the traditional UCD process without totally compromising user research.

Lean design has already been touched on in Jeff Gothelf’s excellent book *Lean UX*, and the Google Venture’s Design Sprints process (explained in their *Sprint* book). These processes are great frameworks that influenced my own approach as a product designer at PwC Digital.

For example, Lean UX is an easily-understood business strategy that translates well to the product team and executive stakeholders. On the other hand, the Google Venture’s Design Sprints is an excellent timeline references for Scrum masters who haven’t built up years of UX experience.

I’ll describe a process that isn’t exactly “agile with a capital A,” but a hybrid process that balances the efficiency of Agile, the due diligence of UCD, and the strategy of Lean UX. I’ve refined the approach through 50+ product design sprints.

The most efficient design processes don't follow a single discipline, but rather adapt the best parts for the team and company. While I'll refer to the process as "Agile UCD" or "Agile Design Sprints", follow the spirit rather than the letter of Agile.

The hybrid process focuses on:

- Starting with user research
- Designing around prioritized "How Might We" questions
- Collaboration between the whole team (designers, developers, marketers, etc.)
- Using minimal documentation and UXPin prototypes to drive decisions
- Focusing purely on outcomes and not outputs
- Fast iteration based on usability testing
- Quick retros to improve future sprints

If you have a good understanding of UX and Lean/Agile principles, you can always strike the right balance between user-focused design and fast timelines.

In this guide, I'll explain a flexible process to ensure successful product design amidst difficult circumstances. We'll explore the strategies and activities for an Agile design process without losing sight of the overall vision.

I’ve also included occasional pointers if you happen to run your design sprint with UXPin as your “base of operations”.

Let’s get started!

Agile UCD in the Wild: The Two Parts of the Design Sprint

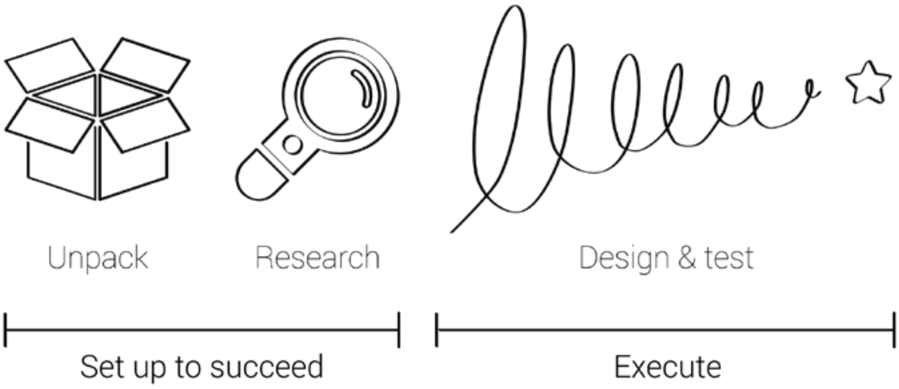
You'll find plenty of information covering the theory of user-centered design, but not as much for executing its processes step-by-step in an Agile environment.

Now that we've covered the pre-requisite information, the rest of the book will cover how to practice user-centered design in a fast-paced environment. I've written each step based on my team's process at PwC Digital.

We can break down the overall design sprint into two parts: 'Set up to succeed' and 'Execution'. Each section is comprised of multiple steps that you should follow in sequence.

- During "**Set up to succeed**", your product team unpacks their thinking about the opportunity. You start to align the team's thoughts. You'll also generate a list of assumptions to validate during user research. You'll end this section ready to design and iterate solutions.

- Next, you'll " **Execute the opportunities**" identified in the previous stage. You'll draw from the group's broad experience to design prototypes. You'll also test the prototypes with at least 5 real users. The 'Execute' stage is iterative – the more you repeat it, the better your designs become. In fact, I've never been on a project where we haven't had to change at least 20 different things the first time we tested.



Your design sprint team should consist of people with diverse skills. For example:

- Developers
- UX Designers
- Visual Designers
- Marketers
- Product Managers

Try to limit your team to a maximum of 10 people and a minimum of 5. Too many and you'll have too many opinions, too few and you'll miss key insights.

Once you've assembled the team, you can complete both parts of the UCD design sprint in as short as a week (e.g. Google Ventures 5-day sprint) or stretch it out to 2-3 weeks. The length of the sprint all depends upon the complexity of the design problem.

Of course, if you must meet a certain timeline, you'll need to focus on only the highest risk design questions (more on that later). At the end of a week-long design sprint, it's totally possible to deliver an MVP with two rounds of iteration if you hyper-focus on just two design questions.

Designer Pro Tip

We adapted practices from Extreme Programming, Agile, and Lean Startup (which I think is an evolution of both of those ideas). We didn't follow 'agile with a capital A', but just treated it as another inspiration for our hybrid process.

Jeff Veen, Design Partner at True Ventures

In my experience, a ~1 week sprint may look something like the following, with 6-8 hours per day for the activities:

- **Day 1**: Unpack your assumptions (Set up to succeed)
- **Day 2**: Conduct user interviews, analyze results, update assumptions (Set up to succeed)

- **Day 3**: Prioritize for 2-3 design questions, start divergent design activities and move to convergent design activities (Executing opportunities)

- **Day 4**: Prototype a focused solution (Executing opportunities)

- **Day 5**: Test the solution and iterate (Executing opportunities)

- **Day 6 and beyond**: Iterating designs, prototyping and testing (Executing opportunities), sprint retros

And a more complex ~2 week sprint may look something like this:

- **Day 1**: Unpack your assumptions (Set up to succeed)

- **Day 2-3**: Conduct user interviews, analyze results, update assumptions (Set up to succeed)

- **Day 4**: Prioritize for 4-5 design questions and start divergent design activities (Executing opportunities)

- **Day 5**: Move to convergent design activities (Executing opportunities)

- **Day 6-7**: Prototype a focused solution (Executing opportunities)

- **Day 8**: Test the solution and iterate (Executing opportunities)

- **Day 9 and beyond**: Iterating designs, prototyping and testing (Executing opportunities), sprint retros

Keep in mind that once you've completed the initial “Set-up to Succeed” stages, you can iterate between designing, prototyping, and testing solutions in as little as a day.

You’ll know your designs are ready to move from the design sprint and into production when testing participants start asking when they can get their hands on them.

For the sake of simplicity, I’ve written the guide as if we were running a sprint lasting 1.5-2 weeks. Think of the days as guidelines, however. Feel free to adapt the sprint schedule to your project needs.

Part 1: Set Up to Succeed

Before the Sprint: Schedule Your User Interviews & Usability Tests

Since the success of a design sprint relies on validated insights, you're generally better off overpreparing and overscheduling when it comes to research.

Aside from a few specific situations, you should start scheduling user interviews (day 2-3) and usability tests (day 8) with at least 5 users before the sprint starts. Unless you're running guerilla research, start recruiting at least 1-3 weeks before the sprint starts.

Let's explore how to seek out participants and offer incentives.

Guerilla Research

Guerilla research is effective in a time crunch for mass-market consumer products.

For example, in one of my projects for a major beverage company, we spoke with people on the streets of Sydney for 5-10 minutes at a time in exchange for free soft drinks and snacks.

You don't need to schedule guerilla research in advance. Once you've finished unpacking assumptions (discussed in the next chapter), you can hit the streets to validate assumptions. As you speak to people, emphasize that you aren't trying to sell them anything – you just want to speak with them for a few minutes.

Keep in mind, however, that guerilla research won't work as well for products with more specific user groups (e.g. enterprise platforms).

Paid Advertising

If you don't already have a user base, you can recruit participants by advertising on Facebook, Google Adsense, or something similar. Your ad should stand out visually and clearly state the incentive. Link the ad to a Google Form that asks a few questions to identify their persona type and collect their contact details.

For example, form questions such as “How many hours a day do you watch TV episodes or movies?” will help identify heavy users.

Finish up the form by asking for their contact details so you can follow up.

Emailing Current and Potential Users

If you’re working on an enterprise product, start emailing people who fall into your relevant customer segments.

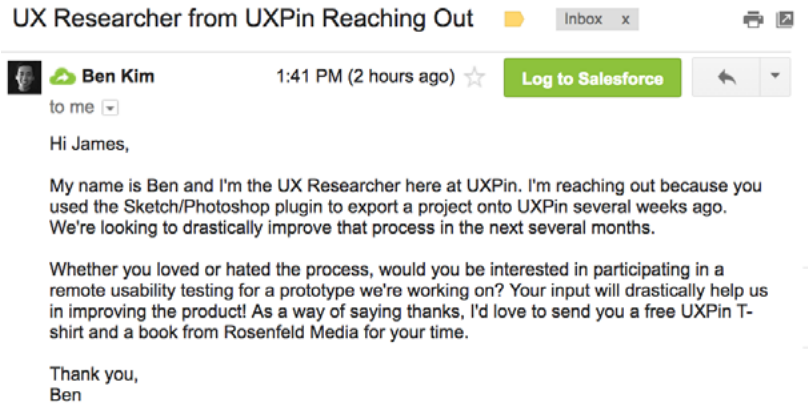
Outside of current users, you can use LinkedIn to find potential users. Before sending out the email, use the free Rapportive Chrome extension to validate email addresses. To follow up, you can then install the Streak Chrome extension to check when they’ve read your email.

Keep your email brief, and emphasize the following points:

- Their feedback is completely confidential and won’t be used outside of the company

- You're seeking their expertise (always helps to make people feel important)
- Exactly how much time you need (no more than 1 hour for user interviews and usability tests)

Try to contact three times as many people as you'd like to interview.



Hi James,

My name is Ben and I'm the UX Researcher here at UXPin. I'm reaching out because you used the Sketch/Photoshop plugin to export a project onto UXPin several weeks ago. We're looking to drastically improve that process in the next several months.

Whether you loved or hated the process, would you be interested in participating in a remote usability testing for a prototype we're working on? Your input will drastically help us in improving the product! As a way of saying thanks, I'd love to send you a free UXPin T-shirt and a book from Rosenfeld Media for your time.

Thank you,
Ben

Designer Pro Tip

Segment interview participants based on your research goal. For example, when we were studying how to improve integrations with Photoshop and Sketch, we generated an internal list of everyone who downloaded the plugin in the past 2 weeks (roughly 70 people). We then emailed them to schedule interviews, which lead to 15 user interviews.

Ben Kim, User Researcher at UXPin

Providing User Incentives

People will require some sort of incentive to participate in your research. The value of incentives should increase with the amount of time required.

At PwC Digital, we typically pay people with an $80 voucher for an hour-long session. If you struggle to recruit people, consider increasing your incentive. Of course, for recruiting current users, generally a swag bag is enough as a show of appreciation.

Conclusion

Regardless of your method, don't recruit participants who are likely to bias your results.

Recruit more people than you need. If you do end up with more users than you can accommodate, keep in touch with them so they can provide you ongoing feedback via email.

Once you’ve scheduled your interview and testing participants, you’ve set up your validation against the assumptions you’ll reveal on the first day of the sprint.

Day 1: Unpack Everyone’s Thoughts

Once you’ve scheduled your user research, begin the Agile UCD sprint by airing the project team’s assumptions out in the open.

You’ll identify what’s known and what requires further exploration.

Consider the outputs from the unpacking stage as a living record of what the group knows and where they’re heading. You will continually revisit the living record as you progress.

The unpacking step consists of focusing on outcomes, assumptive personas, and ‘How might we questions’. Set aside a full day workshop with the whole project team for all the activities.

Focusing On Outcomes vs. Outputs

We start by borrowing a core principle from Lean UX.

An output is a product or feature you think you should create – for example, an app. If you first focus on outputs, you're forcing your product in a direction you assume your users want to go.

Unless you’ve done prior research, you won't be considering any user needs before coming up with a solution.

Outcomes are specific goals like “Increase sales by 5% for new users without affecting retention rate for current users”. By focusing on outcomes first, you allow user needs to dictate the processes and technology.

- Begin your ‘Unpacking’ workshop by encouraging team members to write on Post-It notes all the outcomes they want to achieve through this project. Each team member should do this individually for 5 minutes.
- When time’s up, each team member sticks their Post-It Notes on a wall. Encourage each person to read each one out loud to the group as they’re stuck up. The group can help start clustering similar Post-Its together.
- Once all outcomes are presented, the product team identifies the top three outcomes. For example, the top outcomes for a foreign exchange trading platform might be: decrease anxiety for users

> when they're completing high value trades, give professional traders enough information for informative decisions, and decrease the amount of calls to our help centre by 15%.

To prevent design by committee, the core product team must lead the discussion and decide based on all the input.

Assumption Generation

Now it's time to examine your outcomes from different angles. Write all the questions on big pieces of paper and stick them on the wall.

Answer all questions that apply to your project. You can slightly reword them if necessary.

Questions – User Assumptions

- Who is the user?
- Where does our product fit into their life?
- What problems does our product solve?
- When and how is our product used?
- What features are important?
- How should our product look and behave?

Questions – Business Assumptions

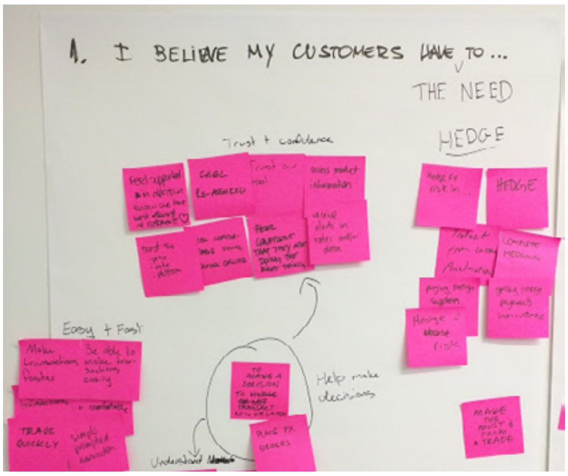
- I believe that my customers have a need to?
- These needs can be solved with?
- My initial customers are?

- The #1 value a customer wants to get out of my service is?
- The customer can also get these additional benefits?
- I will acquire the majority of my customers through?
- I will make money by?
- My primary competition in the market will be?
- We will beat them due to?
- My biggest product risk is?
- We will solve this through?
- What other assumptions do we have that, if proven false, will cause our business/project to fail?

Allow people 3-5 minutes per question to write as many answers as possible on post-it notes. Remind them to write one answer per post-it note.

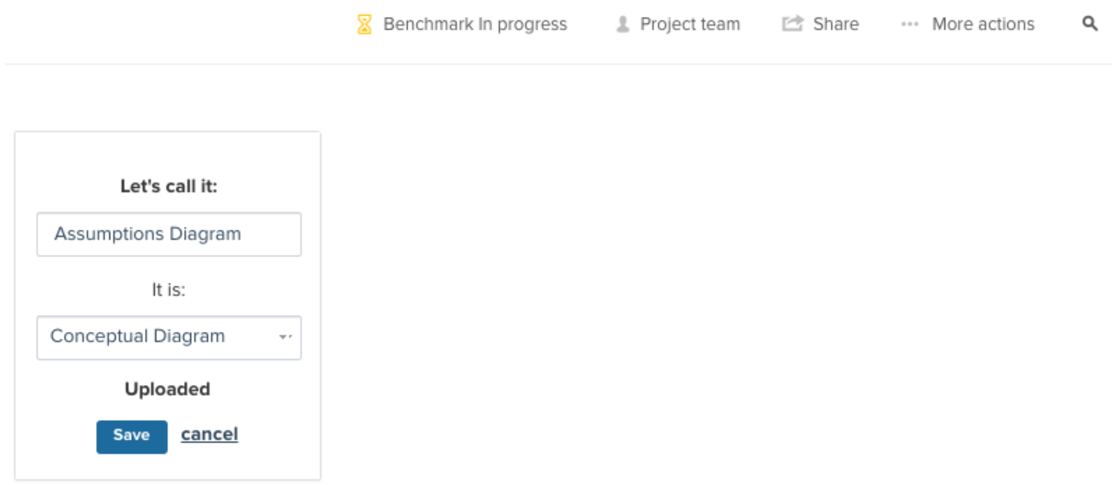
When time’s up, everyone sticks their post-it note answers under the question. As this happens, the facilitator can cluster similar answers together.

You’ll end up with something like this:



After answering all the questions, you’ve laid out enough raw material to create assumptive personas. Photograph the wall for future reference.

If you’re a UXPin user, you can take it a step further and drag the photo into your project as the first piece of documentation for your sprint team (Select “Conceptual Diagram” as the file type for easy tracking).



Let's call it: Assumptions Diagram

It is: Conceptual Diagram

Uploaded

Assumptive Personas

Assumptions in design are only bad if you never validate them. Luckily, assumptive personas are just an early iteration that we’ll verify with the scheduled interviews.

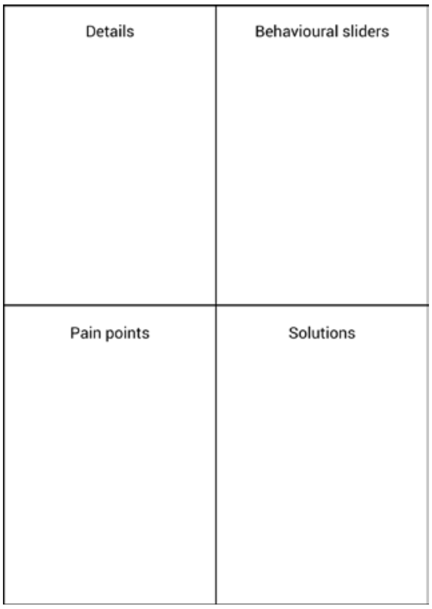
First, let’s clear up a misconception: personas aren't separated by things like age, gender, or geographical location. Useful personas are always separated by *behaviors*.

For example, a group of people who exercise three times a day and eat low-carb diets could be considered a persona, even if they were different ages and genders.

This contrasts with the way marketing teams define customer segments. These are based on factors like age, gender, location, and income. Customer segments often find their way into design. However, it doesn't matter what age you are, you can still be passionate about exercising three times a week and eating a low-carb diet.

When you nail the experience, your primary persona will start to promote your product to other persona groups. Only then do you build features that meet other persona needs, which will attract more users.

You start creating your assumptive personas by filling out the template below. You want to create an assumptive persona for each user type that you previously identified in the "Assumption Generation" activity.



Details	Behavioural sliders
Pain points	Solutions

1. **Details**

   Give your persona a categorical name. For example, 'Casual gym goers'.

   Note down a few characteristics of the people who belong to this persona and draw a picture of what this group may look like.

2. **Behavioral sliders**

   Behavioral sliders help you quantify subjective behaviors. They are measured on a scale and give you a visual, at-a-glance way to differentiate and understand the personas.

   A few sliders for gym goer personas could be:

   - Number of times per week this person visits the gym (scale of 0 to 14)
   - Where this persona sits on a scale of only focusing on cardio to only focusing on weights
   - Number of supplements this persona takes per week (from 0 to 20 servings)
   - Strictness of the persona's daily diet (from 0 to 350 grams of carbohydrates)

   Behavioral sliders don't need to be numerical values. For example, what does the person value on a scale of emotional well-being to physical fitness?

You should end up with 4 to 8 sliders. Any less and they’ll be too broad, any more and they’ll be too specific.

If you’ve done prior research like user interviews, base your sliders on what you know about user behaviors. If not, take your best guess. Again, you’ll validate these during the scheduled interviews.

3. **Problems**

   Simply list a few problems experienced by the persona. Examples for a ‘casual gym goer’ could be:

   - They find it hard to make time to go to the gym
   - They only visit the gym during peak hours, which means waiting for the machines
   - They often forget the routine recommended by personal trainers.

   If you’ve already done research on this persona, include the problems you’ve discovered. If not, again take your best guess.

4. **Pain relievers**

   Simply list a few things you could do to relieve this persona’s problems. Examples for our ‘casual gym goer’ could be:

   - Incentivize them to visit the gym outside peak hour with lower prices
   - Get more machines

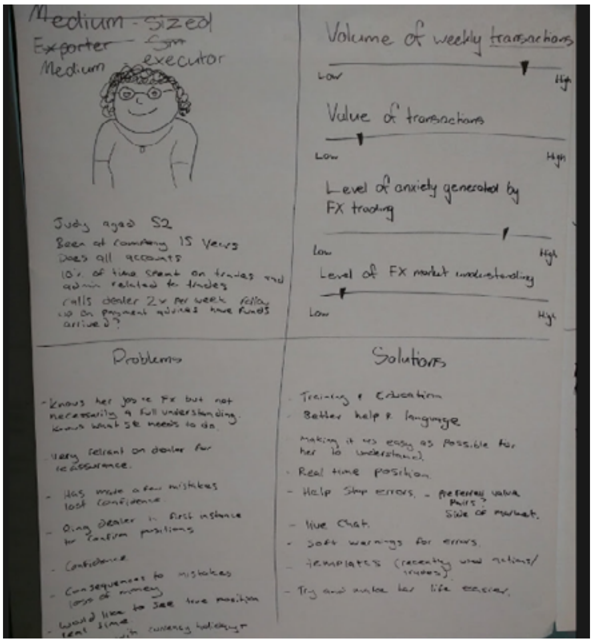
- Build an app that that provides video instructions for exercises recommended by trainers

Don’t feel committed to any of these pain relievers. The exercise is only meant for you to start thinking about how to help users.

If you’ve already done research, these relievers should directly relate to a pain experienced by your persona.

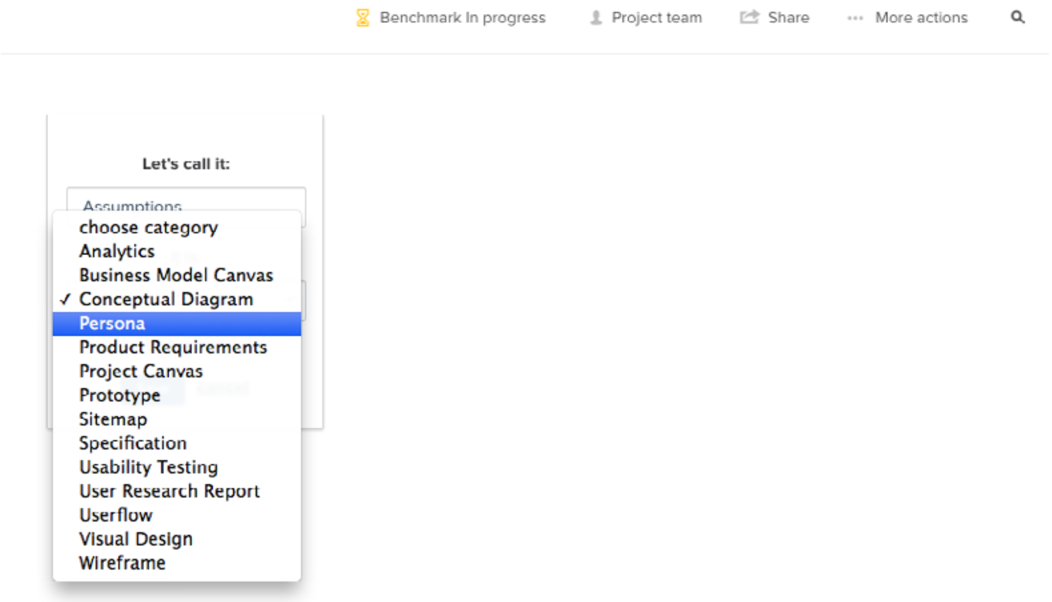
5. The complete assumptive persona

Your assumptive personas will end up looking like this:


~~Medium-sized~~
~~Exporter~~ ~~SM~~
Medium executor
Judy aged 52
Been at company 15 years
Does all accounts
10% of time spent on trades and admin related to trades
calls dealer 2x per week follow up on payment queries have funds arrived?
Volume of weekly transactions
Low
High
Value of transactions
Low
High
Level of anxiety generated by FX trading
Low
High
Level of FX market understanding
Low
High
Problems
- Knows her job re FX but not necessarily a full understanding. Knows what she needs to do.
- Very reliant on dealer for reassurance.
- Has made a few mistakes lost confidence.
- Ring dealer in first instance to confirm positions
- Confidence
- Consequences to mistakes loss of money
- Would like to see true position real time with currency holdings
Solutions
- Training & Education
- Better help & language
- Making it as easy as possible for her to understand.
- Real time position
- Help stop errors. - preferred value pairs? Side of market.
- live Chat.
- soft warnings for errors.
- templates (recently used actions/trades)
- Try and make her life easier.

If you haven’t based your personas on research, consider them assumptive. Don’t worry though, because we’ll validate these assumptions during the user research that comes next.

If you’re a UXPin user, snap photos of the personas and drag them into your project folder. Select “Persona” as the file type for easy reference later on. Now, you can quickly compare the assumptive persona to the assumptions diagram from the earlier exercise.

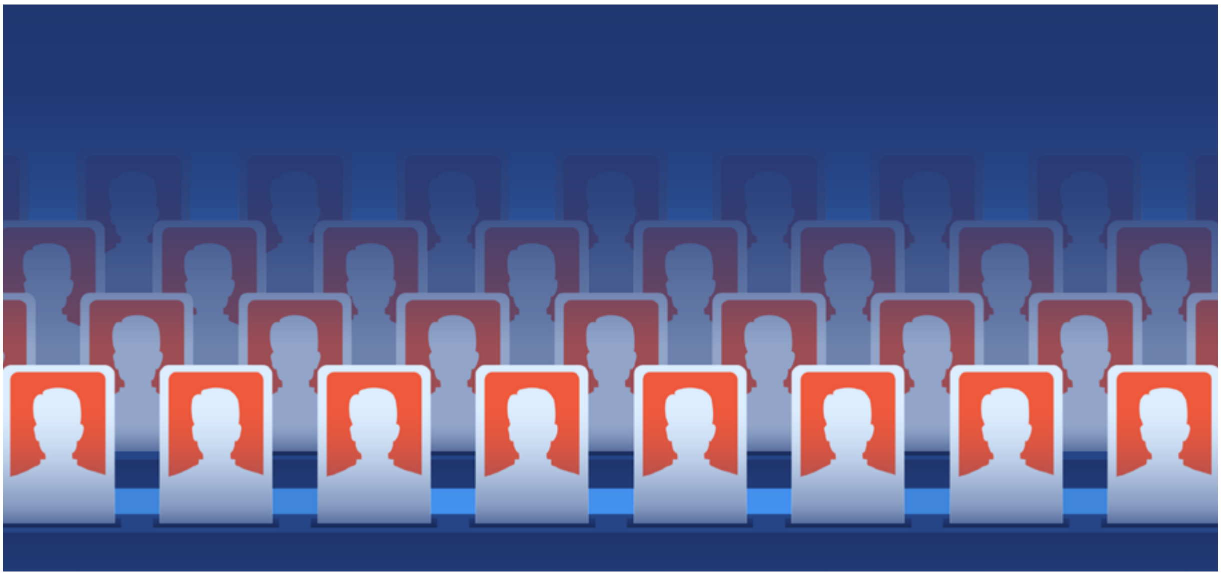


Let's call it: Assumptions

Day 2-3: User Research Through Interviews

Now you validate your assumptions and identify your persona's pains and gains. Both Lean UX and traditional UCD advocate user interviews for fast insights.

During your scheduled user interviews, ask users about the whole scope of the experience, not just the particular area you're interested in.



For example, if you want to create a competitor to Netflix, don't just talk to people about how they currently use Netflix. Talk to them about other services they use – legal or illegal. Where do they watch episodes or movies? What device they use? Do they download or stream? Why?

Exploring your topic with a wide scope allows you to discover what works and what doesn't.

You can also ask questions to validate assumptions from the unpack stage. For example, you may assume that people illegally download TV episodes because they don't want to pay. However, after talking to participants, you find that people download episodes illegally because Netflix doesn't offer the episodes they want.

You've now learned that availability, instead of price, might be the primary concern of users.

Now let's explore the power of quantitative research and how to run a successful user interview.

Validating Quantitative Research

Quantitative data only shows you the "what", but not the "why". User interviews help you dive deeper into the motivations behind the data. Quantitative research, like surveys and web analytics generated in Kissmetrics, uses a large amount of data to achieve statistical confidence.

For example, we could survey 300 users and discover that 30% responded that they do cardio workouts at our gym. Since we examined data from 300 users, we can be confident that cardio workouts aren't very popular.

The discovery, however, isn't very helpful for deciding *how* to redesign the layout of gym equipment. You could assume that people just aren't interested in cardio and replace cardio equipment with weights. But what if that's not the real reason why the usage rate is so low?

User interviews could shed light on the true motivation behind the 30% cardio workout rate. After talking to people and understanding their behaviors, emotions and motivators, you might discover that people avoid cardio because they feel self-conscious about struggling and sweating in front of strangers.

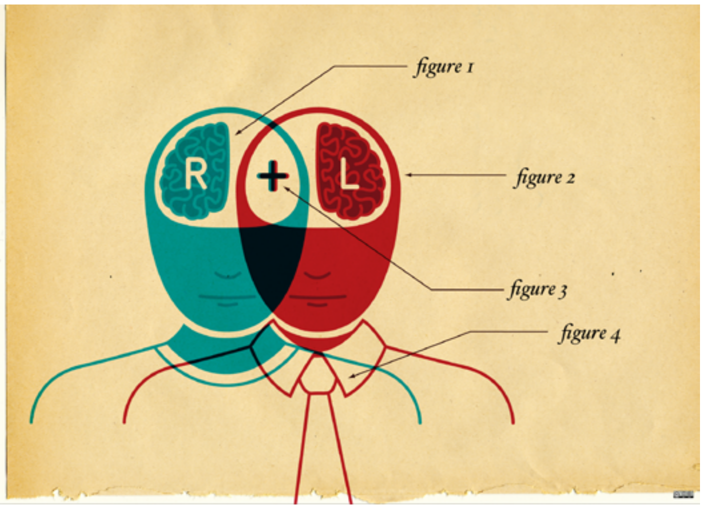


Photo credit: "Is the traditional business world at war with creativity?" Opensource.com.Creative Commons 2.0.

By validating the quantitative data with qualitative research, you've uncovered the true problem to solve: self-consciousness, not lack of demand. Now, instead of replacing cardio equipment with weights, you could create a dimly-lit exercise bike class that doesn't place users in the spotlight.

Without using qualitative research to identify behaviours, emotions and motivators, you certainly would have made a bad design decision.

The reverse process also works. You can also validate qualitative findings with quantitative questions. For example, when interviewing 20 people about the reasons behind visiting the gym, you might see a huge fluctuation in the days they visit. But since you only talked to 20 people, you can't be confident that behavior reflects reality.

To raise your confidence level, you can dig deeper with a single-question survey sent to 300 people:

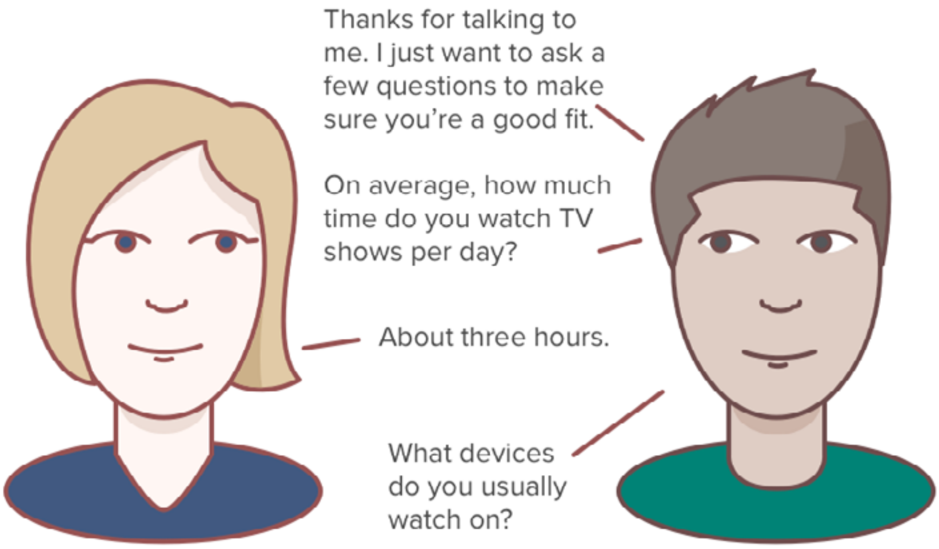
”What day of the week do you go to the gym?”

Quantitative research, like surveys, raises the statistical confidence level. Qualitative research then reveals the emotions and motivations for the aggregate data.

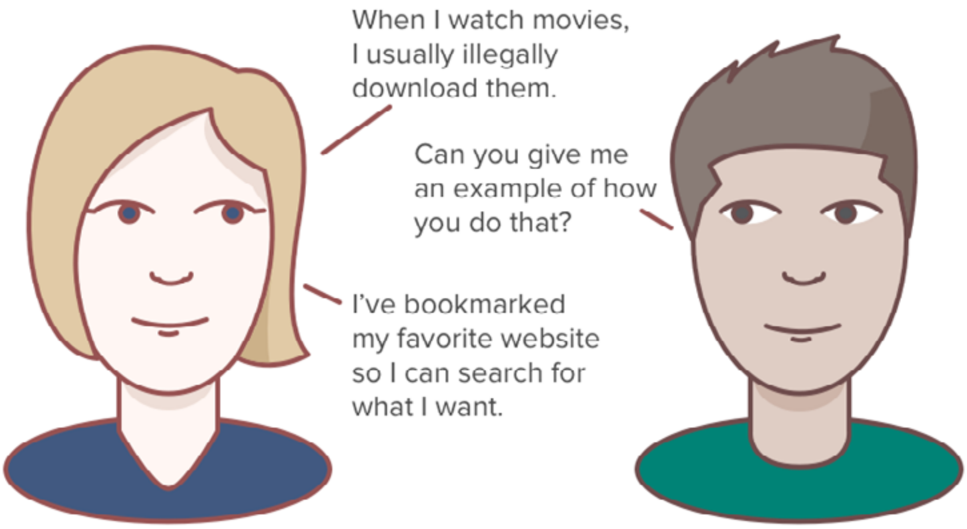
How to Conduct User Interviews

Start your user interview with an introduction of who you are and how the interview will work.

It's a good idea to ask participants questions about themselves early so you can add context to their answers.


Thanks for talking to me. I just want to ask a few questions to make sure you're a good fit.
On average, how much time do you watch TV shows per day?
About three hours.
What devices do you usually watch on?

Next, dive into the questions that speak directly to your research goals.


When I watch movies, I usually illegally download them.
Can you give me an example of how you do that?
I've bookmarked my favorite website so I can search for what I want.

Interviews work well by beginning with a broad question for each topic before narrowing down to the specifics. The following techniques will help you do this.

1. **Start broad**

   Begin with a question that broadly covers the topic. For example:

   "Tell me about the TV episodes and movies you watch."

   This way you'll allow the participant to lead the conversation and talk about what's top of mind for them.

   Starting broad lets the interview flow more like a conversation than a formal set of questions. This allows the participant to relax which leads to more honest and truthful answers.

Designer Pro Tip

If you're talking more than the user, you're doing it wrong. Don't try to fill in moments of silence in the conversation. Let the user finish up their answer to your broad question, then wait a few moments after they're done. They'll usually fill in the silence with an unexpected insight.

Ben Kim, User Researcher at UXPin

2. Ask about sequence

Another broad question is to ask about a sequence of actions. For example:

“Take me through what you did the last time you watched TV episodes or movies online.”

These questions help you understand the natural task flow preferred by users. The results will help you later to design intuitive flows in your prototypes.

3. Make your question easier

Of course, broad questions can sometimes be difficult to answer. If someone is struggling to answer a broad question, make it easier. For example, guide their thinking by further asking:

”What was the last TV show or movie you watched on online?“

Once the participant has answered that question, you can then ask:

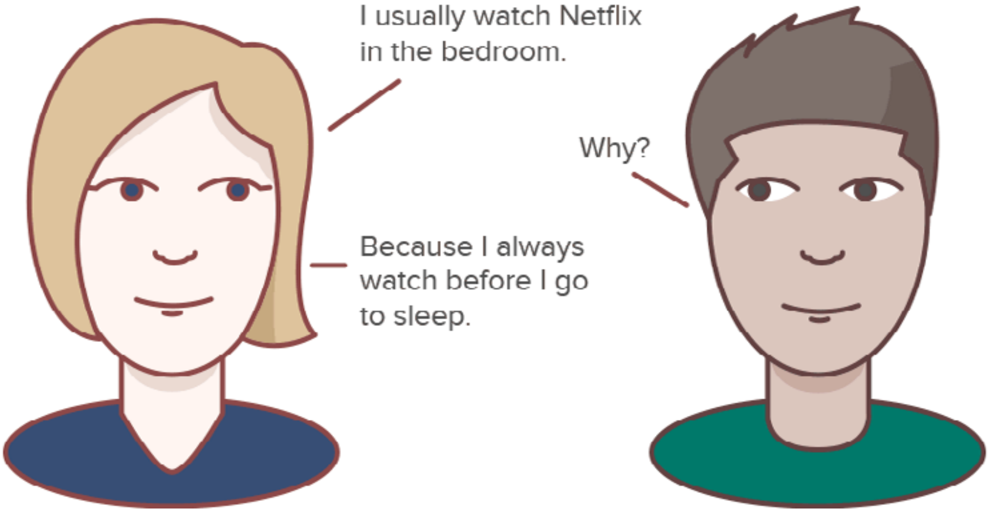
”What did you watch the time before that?

Then ask:

”What else do you usually watch?”

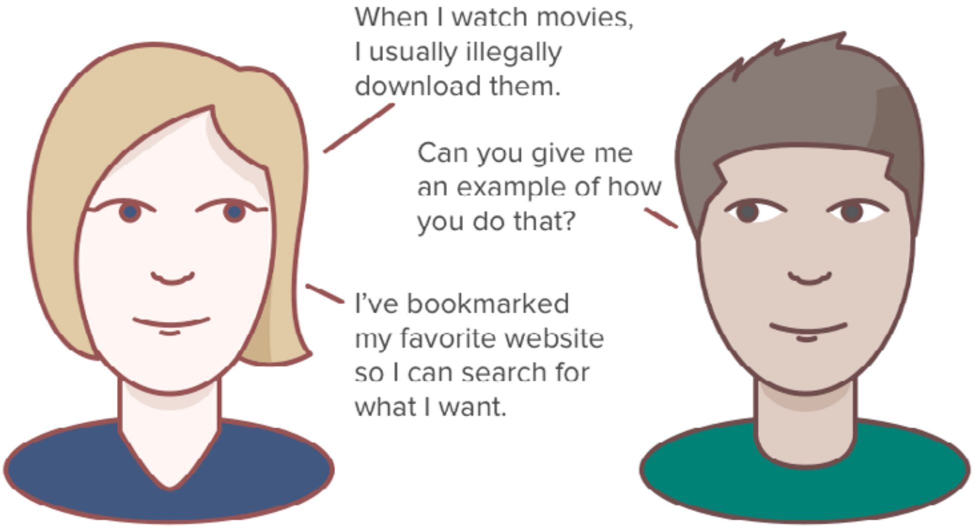
4. Ask why

If participants say something, always ask why.


I usually watch Netflix in the bedroom.
Why?
Because I always watch before I go to sleep.

5. Ask for examples

Ask for examples when you want more clarity about a particular topic or sequence of actions.


When I watch movies, I usually illegally download them.
Can you give me an example of how you do that?
I've bookmarked my favorite website so I can search for what I want.

6. Repeat the participant's language

If a participant says:

"I like that"

Then ask:

"What do you like about that?"

Or, if they say:

"That sucks!"

Ask:

"Why does that suck?"

Repeating language removes the possibility of biasing a participant's response.

For example, if a participant says "That's cool" and the facilitator asks "What do you like about that?," then the participant's response will be biased towards responding to why they like it, not why they think it's cool.

7. **Don't lead your participant**
   It's very easy to ask leading questions.

   For example, if a participant is talking about how much they watch reality TV four days a week, don't ask "Is that because you want more excitement in your life?"

   The participant didn't mention anything about seeking excitement, so it's wrong to assume so. Instead, you should ask "Why do you watch reality TV four days per week?"

Designer Pro Tip

Towards the end of a user interview, I'll ask a few quantitative questions to provide a comparative measure between interviewees. For example, I might say 'I find this feature easy to use. Can you rank this statement from a 1 (strongly disagree) to a 10 (strongly agree)?' We can compare the scores against the qualitative responses to understand which feature ideas to prioritize. If I talk to 15 people and the average is a 3, we know we might need to tweak our product roadmap.

Ben Kim, User Researcher at UXPin

Optional: Running User Surveys

If you’re on a time crunch, you can skip this step.

If the entire sprint spans 3 weeks instead of 7 days, however, you can run a user survey after your user interview to further validate the analytics and interview results.

The next few steps will help you create a survey you can trust.

1. **Ask for personal details last**
   Participants can be put off if you ask for personal details early in a survey. It’s better to ask these questions at the end when the participant has built trust.

2. **Only show participants questions that relate to them**
   Use question logic (available in most tools like Surveymonkey) to show appropriate questions based on prior answers.

   For example, if a participant answers that they don’t have Netflix, don’t show them any more questions about Netflix.

3. **Ask closed questions**

   Open questions are hard to analyze in survey format. Instead, create closed questions by reframing answers from user interviews.

   For example, during your interviews about online media consumption, you get varying answers to the question 'How many TVs do you have in your household?' To clarify the average number of TVs per household, you decide to turn this interview question into a survey question. If 300 people respond to your survey, you'll be a lot more confident about the average amount of TVs in a household.

   Here's some of my favorite resources for running user surveys:

   - Most Useful Questions for User Surveys
   - Writing Surveys That Work
   - Better User Research Through Surveys

Day 2-3: Updating User Personas

Once you've analyzed data and interviewed your users, it's time to hunt for patterns in the following areas:

- **Pains** – Where your participants described frustration.
- **Gains** – Which activities currently work, but could be improved.
- **Observations** – Any patterns you witnessed in the analytics, or behaviors/mindsets users described that are interesting, but aren't pains or gains. For example, a person mentioned they always downloads movies on a Sunday.

Review your findings and pull out each pain, gain and observation. For me, the best approach is to write findings on colored post-it notes:

- Pink for pains
- Green for gains
- Blue for observations.

Write with a thick-tipped Sharpie so each point is easy to see. You'll also force yourself to keep findings concise.

Assign each research participant a number. Write this on the post-it note for each finding. It will help trace your findings back to specific participants and build personas around similar behavioral traits.

As you write these findings, stick them on a wall. You'll begin to see similar findings from different participants. Start clustering these findings together. You'll end up with something like this:

Purchasing
Buy / Sell direction
Want to know how long deal is current for
Rate update during 30 sec hold
Editing after currency change
Divisions
Live rates

Update Your Assumptions

After analyzing your findings, you need to update your assumptions based on the newly validated research.

- Based on your initial Assumption Generation activity, make a list of all the assumptions proven correct (along with supporting evidence) on pieces of flipchart paper and stick them on a wall.

- Make a second list of all the assumptions proven false. Write what was proven false (along with supporting evidence) on flipchart paper and stick these near the true assumptions list.

- Make a third list of new considerations. For example, a team may have assumed that a heavy consumer of TV episodes and movies watches alone on their laptop. But through user interviews, you learned these people always watch on their TV with other people.

Once you've tracked everything up for the team to see, update your assumptive personas to reflect reality. If you're using UXPin, you can add a new persona titled "Validated Persona" to your project.

Day 2-3: Transforming Insights Into Design Questions

Now you can create ‘How might we’ questions based on the validated findings from user research. We’ll applying traditional design thinking with a bit of Agile prioritization.

HMW questions phrase opportunities in an open way that defer judgement. Words like ‘can’ and ‘should’ imply judgement by focusing on feasibility. Even a subtle difference in phrasing can stunt how opportunities grow within the product team.

Instead, substituting ‘can’ and ‘should’ with ‘*might*’ creates an open environment for exploring ideas. You’ll learn more by keeping an open mind, even if you discover that an opportunity isn’t worth perusing.

Developing HMW Questions

You want to create HMW questions that aren't too broad, yet aren't too narrow. Sound difficult? I'll explain below.

Let's say you're working on a online redesign project for a bank. Across 6 user interviews, you noticed comments like "I don't like to use the site since I prefer talking to a real person".

A question like "HMW redesign the online banking experience?" is too broad here. On the other hand, a question like "HMW we design an online banking app with conversational feedback messages?" is too prescriptive.

Instead, a well-balanced HMW question might be "HMW redesign online banking to feel more personable?".

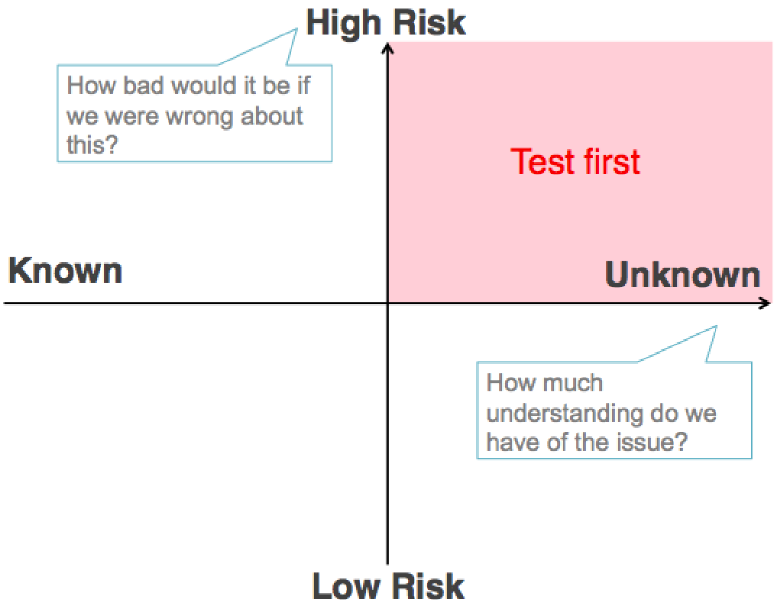
Of course, sometimes broad HMW questions can work if you're working on a large-scale project with ample time and budget. Usually, however, you'll want to aim for a balanced HMW question.

Generate as many HMW questions as you feel necessary to offer value to users based on your research insights. Don't worry – we'll prioritize them in the next part.

Prioritizing HMW Questions

Prioritize HMW questions by the degree of risk and amount of current knowledge. The highest priority HMW questions pose the highest risk and suffer from the lowest amount of current knowledge.

You can prioritize your HMW questions by plotting them on this graph.


High Risk
How bad would it be if we were wrong about this?
Test first
Known
Unknown
How much understanding do we have of the issue?
Low Risk

- **High Risk / Unknown** – Focus the remainder your sprint towards these questions. Spend time in divergent and convergent design to explore solutions.
- **High Risk / Known** – Depending on your timeline, it might also be worth exploring a few questions in this quadrant. If you're pressed for time, only focus on the questions above.
- **Low Risk / Unknown** – You could tackle in this sprint, or assign to backlog.

- **Low Risk/ Known** – You could tackle in this sprint, or assign to backlog.

The length of your sprint determines the number of HMW questions you can answer. For a 7-day sprint, you might be able to focus on 2-3 HMW questions.

Even if you can't tackle all the HMW questions in the current sprint, you can still plan them for Sprint 2, 3, etc. In fact, you might even indirectly answer lower-priority questions during the exploration of divergent and convergent design.

You'll validate your HMW question's priority again during usability testing in 'Stage 2: Executing UX Opportunities'. You may find that they aren't as much of a risk when you've created and tested a solution.

Part 2: Executing UX Opportunities

Day 4-5: Co-Design Sessions

Co-design sessions help eliminate inefficient conversations.

When you arrive at a shared understanding of the problem space (the “why” behind the “how”), you minimize the risk of misguided implementation. Designers and developers start thinking beyond the handoff mentality.

Co-design sessions have two steps:

- Divergent design
- Convergent design.

Each HMW question will go through its own diverge and converge process. Again, you’ll want to repeat both of the above processes for each HMW question. Start with the riskiest questions you identified in the previous section.

Remember that co-designs need to be a safe environment for ideas. All ideas, no matter how crazy, are welcome.

Idea-killing phrases like “that never works,” “we don’t do that here,” or “we’ve tried that already” are common and can easily ruin idea-finding environments. Ideas will always be easier to find if they’re not shot down on sight.

As you might imagine, the team will sketch a lot during the co-design sessions. Some people shy away from this because they don’t think they’re creative or just haven’t done it in a while. However, everyone is capable of creativity.

The only difference between “creatives” and others is more attitude and experience rather than nature.

Divergent Design

Diverging is when you create as many ideas as possible that aim to answer each HMW question. We don’t know what ideas have value until we explore them all – so the aim is to come up with as many as possible. Then you review each idea and narrow down to the best ones.

At this point, the goal is quantity, not quality.

1. **Warmup Scoping Session**
   Begin by discussing the rough feature set.

You prepare your team for the ensuing collaborative sketching session by helping them first list all the ideas before sketching them.

- Give the group 5 minutes to write down on Post-It notes any design features that could possibly answer your HMW question.
- Tell the group not to describe the aesthetics or functionalities in great detail. You just want to think about general functionalities.
- Stick these post-its onto a wall and cluster similar features as they're being stuck up. You'll begin to see areas that need to be co-designed. In a hypothetical high-budget project, a broad HMW question like 'How might we increase customer engagement?' may generate post-its like gamification, ease of use, and social media features. In the divergent sketching exercise, you'll actually start exploring the specificities (e.g. which steps of the user flow should we gamify).

No matter what you're designing, someone else in the world has probably done something similar. Use these similar products as an initial source of inspiration for your own solutions.

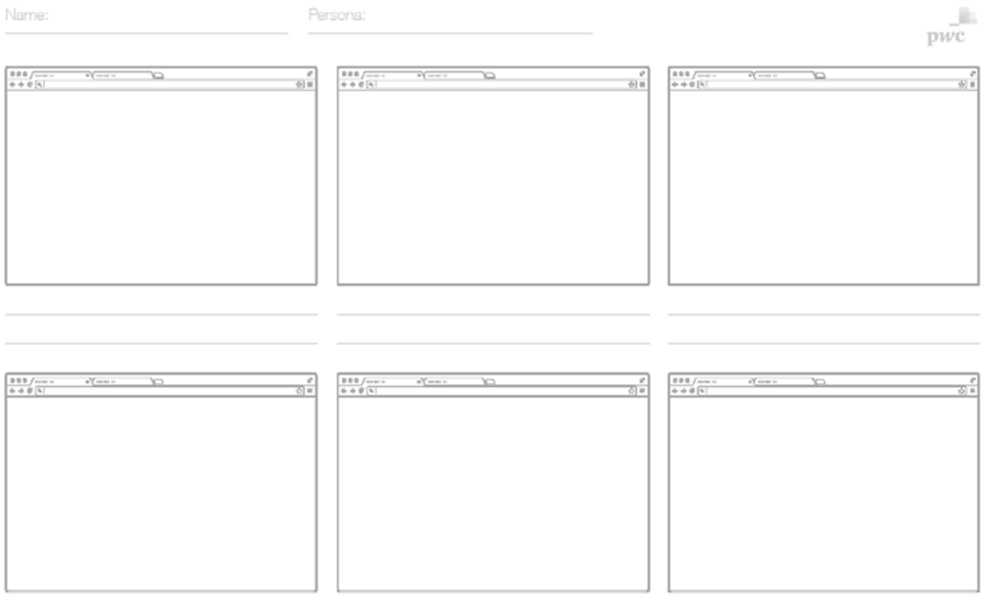
- Encourage each team member to look for something that may help achieve your outcomes.
- Print these examples out and present them before the co-design session. This will help create better designs.

Don’t cull any ideas yet (that comes in the next session). You’re only helping to facilitate idea generation and categorization.

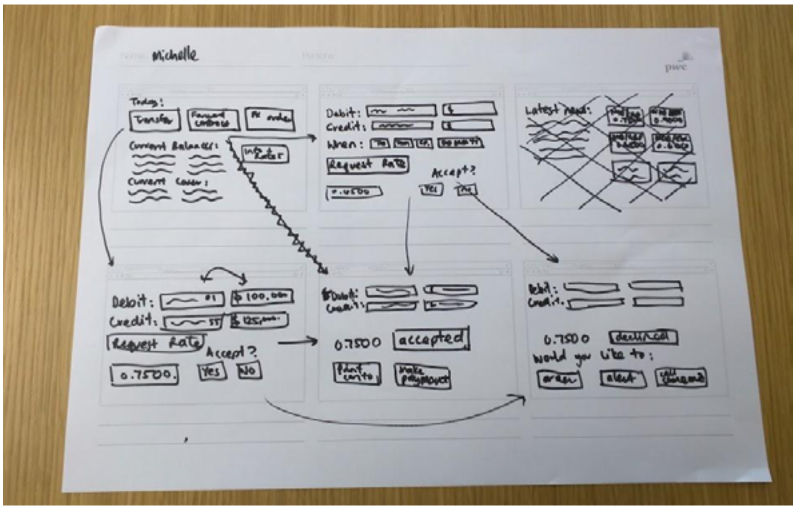
Your results here act as the guideline for the sketching session, since not everyone is a visual thinker who can start from scratch.

2. **Collaborative Sketching Session**

Now that you’ve generated the broad feature ideas, the collaborative sketching session starts with a ‘6-up’. A ‘6-up’ is an A-3 piece of paper that you divide into six boxes – one box for each idea.



Set aside 5 minutes for each team member draws as many solutions to the HMW question as possible.



Once time is up, all the 6-ups are stuck on a wall. Each team member describes their solutions to the group. While each team member presents their 6-up, the rest of the group writes feedback on post-it notes.

Format the feedback in two ways:

- ‘I like’ – What you like about the solutions.
- ‘I wonder’ – How you might improve the solutions.

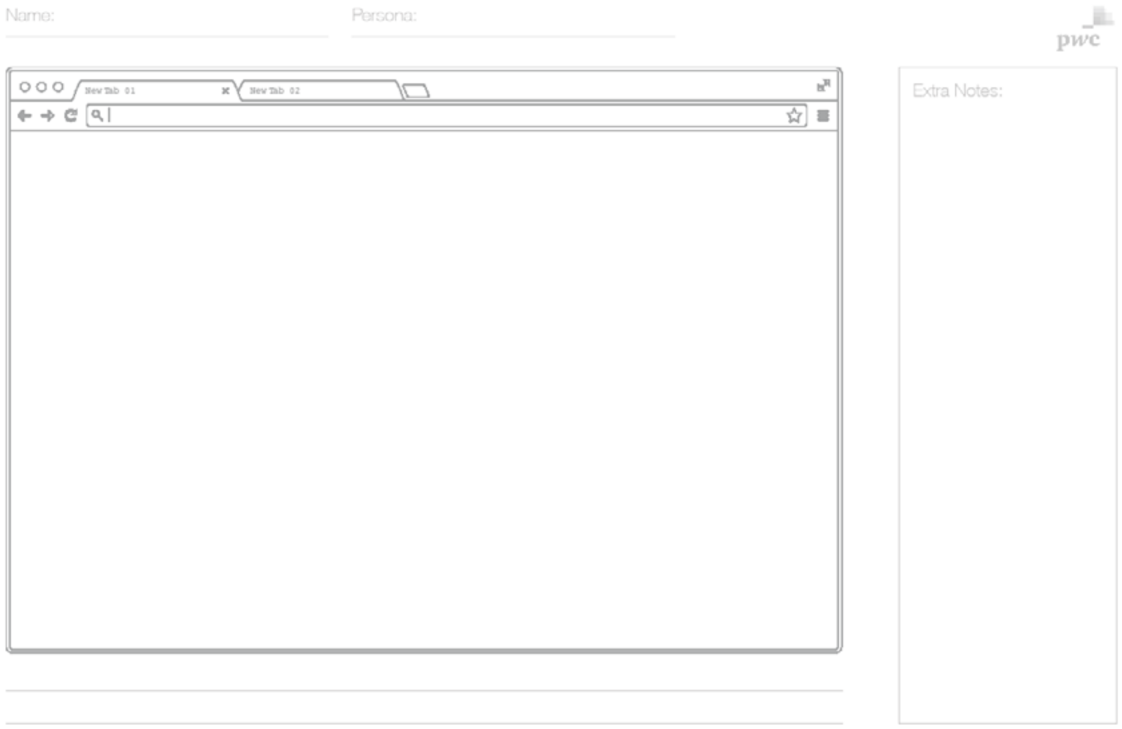
The idea is to write as much feedback as possible – one piece of feedback on each Post-It note. Make sure each person sticks the feedback on the relevant area of the 6-up.

Convergent Design

Once everyone has presented their 6-ups, they look over the feedback and consolidate the best designs into one solution each.

Remember that no idea belongs to anyone. You'll get to the best ideas if you steal and build on each other's.

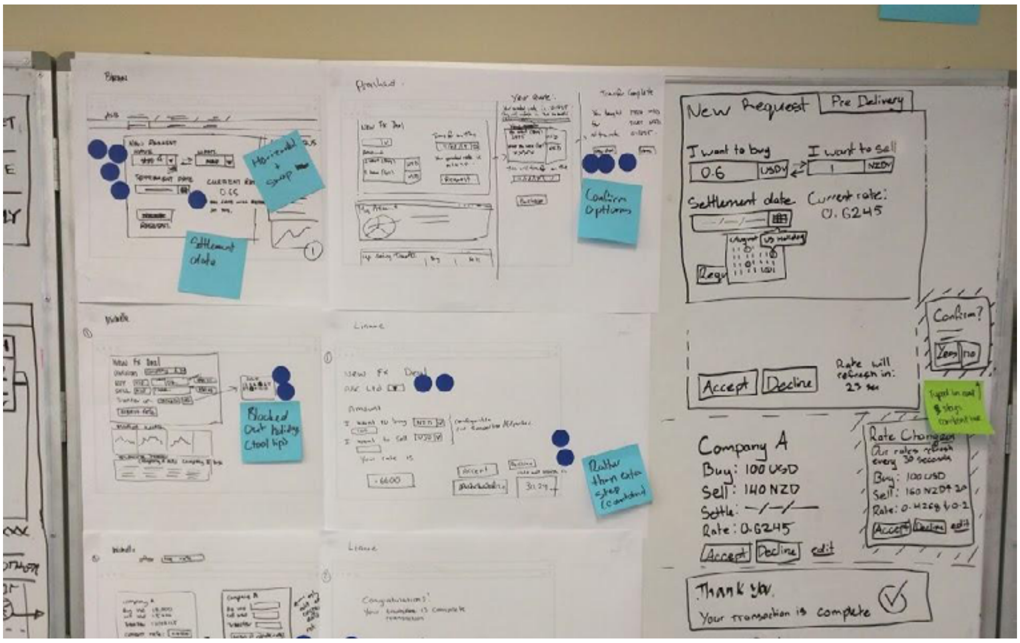
Give the group 5 minutes to draw their one best solution. Do this on a '1-up'. A 1-up is an A-3 sized template of the device, or situation, for which you're designing.



After 5 minutes, stick the 1-ups on the wall next to the 6-ups.

Each team member then presents their 1-up back to the group, explaining their rationale behind the refined solution.

Instead of giving feedback, the team now votes. They can either vote for their favorite idea as a whole or features of an idea. Give each team member three votes in the form of sticky dots.

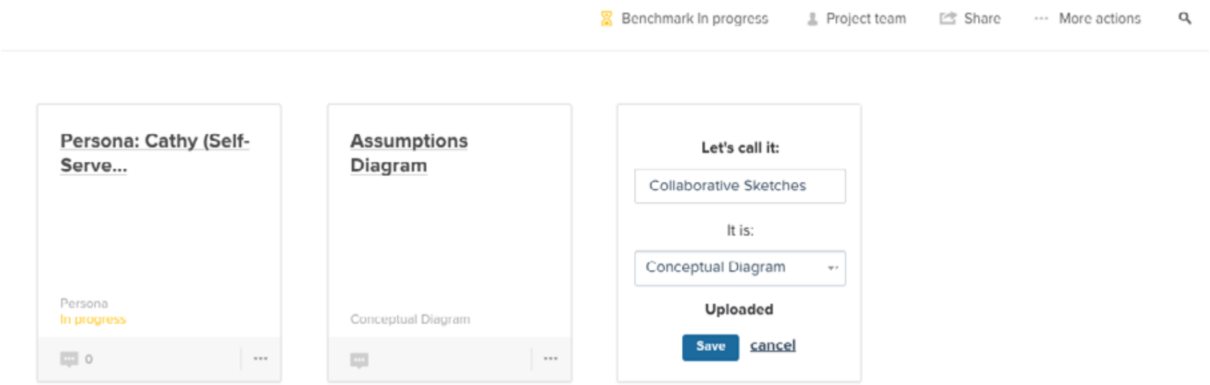


If a unanimous winner arises, you can take that design into the prototyping phase.

If votes are split, you can combine features from various solutions into a prototype. To avoid a Frankendesign, the designers and product manager will make this judgment call.

If competing solutions have a lot of votes, you can prototype both and A/B test them to discover what performs best during usability testing. Repeat this process of diverging and converging for the rest of the HMW questions. Make sure you save all the sketches.

If you're a UXPin user, consider photographing the sketches and dragging into your project (they'll help guide your prototyping and save you from adding to a growing pile of paper). Your team can also comment on the sketches in case an interesting idea strikes them after the session.



Persona: Cathy (Self-Serve...

Persona

In progress

0

Assumptions Diagram

Conceptual Diagram

Let's call it:

Collaborative Sketches

It is:

Conceptual Diagram

Uploaded

Day 6-7: Prototyping the Solution

Once you've passed through the divergent and convergent stages for your HMW questions, you'll have a pared-down list of focused features that are worth prototyping as a cohesive design.

When prototyping, you want to exert as little effort as possible. Remember the key lesson of Lean and Agile UX: focus on building an MVP.

In my experience, coding is generally out of the question because it's hard to iterate quickly in a design sprint. Not all designers can prototype in code as quickly as they can prototype on paper or in a digital tool.

Digital Prototypes

For digital projects, get your one or two most experienced designers to distill the co-designed solutions into a prototype–or if you're planning on A/B testing, two prototypes.

Let’s explore the process for prototyping quickly after your co-design sessions.

1. **List out the steps of your important user flows**
   Before you begin prototyping, think about the tasks you'll ask your users to complete when testing. This is so you know the exact scope of the prototype.

   For example, imagine we are prototyping a dieting app and the HMW question we're trying to answer is “How might we engage users to frequently use our app?” The winning solution might use gamification by rewarding users with badges when they achieve tasks–for example, submitting their first meal, sticking under their allotted calories for the day, losing 5 lbs etc.

   If you just prototyped exactly what was co-designed, you might end up with screens for submitting a meal and the presentation of a badge. However, this isn't what would realistically happen.

   Instead you need to think about the full journey of testing the scenario. For example:

   - The user takes out their phone
   - Opens the app
   - Navigates to the 'add a meal' section
   - Clicks add a meal
   - Finds the food they ate for breakfast

- Hits submit
- Then receives a badge for their first meal submitted

Then, to really test the experience you should:

- Ask users to repeat the process for lunch, dinner, and any snacks they've had throughout the day
- Then present them with another badge for keeping under their allotted calories for the day

This way your testing session will be more accurate because it will represent a realistic scenario. Create a list of all these steps so you know the scope of what your prototype will cover.

2. **Sketch the steps in your user flows**

Now we'll pull out the highly voted 1-ups from our co-design session.

If votes are split on different features of a 1-up:

- Take the highly voted elements and figure out how they'll work best together. You'll be avoiding Frankendesign because the highly voted elements shouldn't be similar. If they are, then two prototypes should be created to A/B test the solutions.

If votes are split on similar 1-ups:

- Come up with an A and B solution. Make sure you're clear about what elements you are creating a competition between. Set them up so the test is fair. For example, build the same steps leading up to the A/B designs so a user isn't more fatigued when they reach one compared to the other.

If there's a clear winner from co-design:

- Still sketch out the steps of the user flow to decide exactly how you'll create it before you prototype.

You may notice that the prototype doesn’t cover everything. For example, the navigation may not have been covered. Don’t worry– the purpose of co-designing was to figure out how to answer your HMW questions (the core functionality). Your prototypers can fill in the gaps by creating the missing UI elements.

Next, sketch on a piece of paper or whiteboard each screen of the prototyped solution.

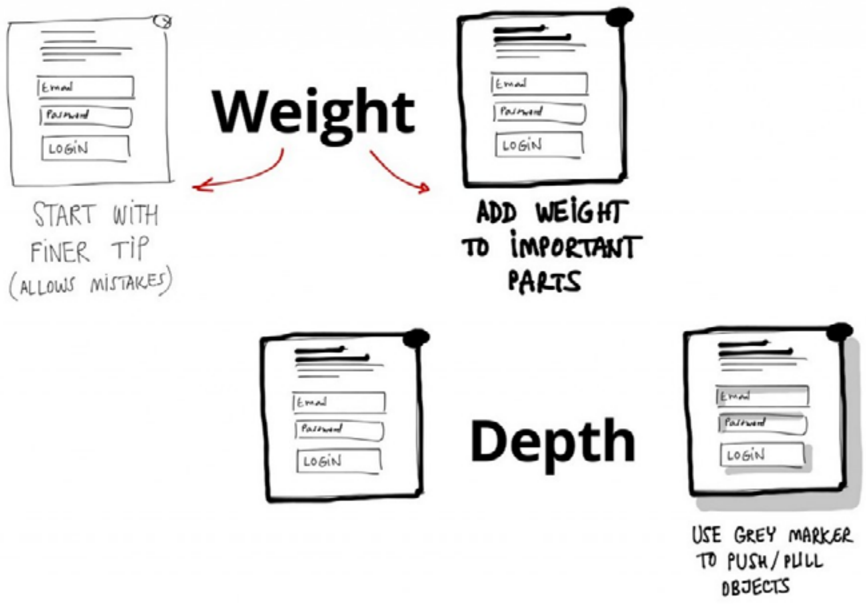

Weight
START WITH FINER TIP (ALLOWS MISTAKES)
ADD WEIGHT TO IMPORTANT PARTS
Depth
USE GREY MARKER TO PUSH/PULL OBJECTS
Email
Password
LOGIN

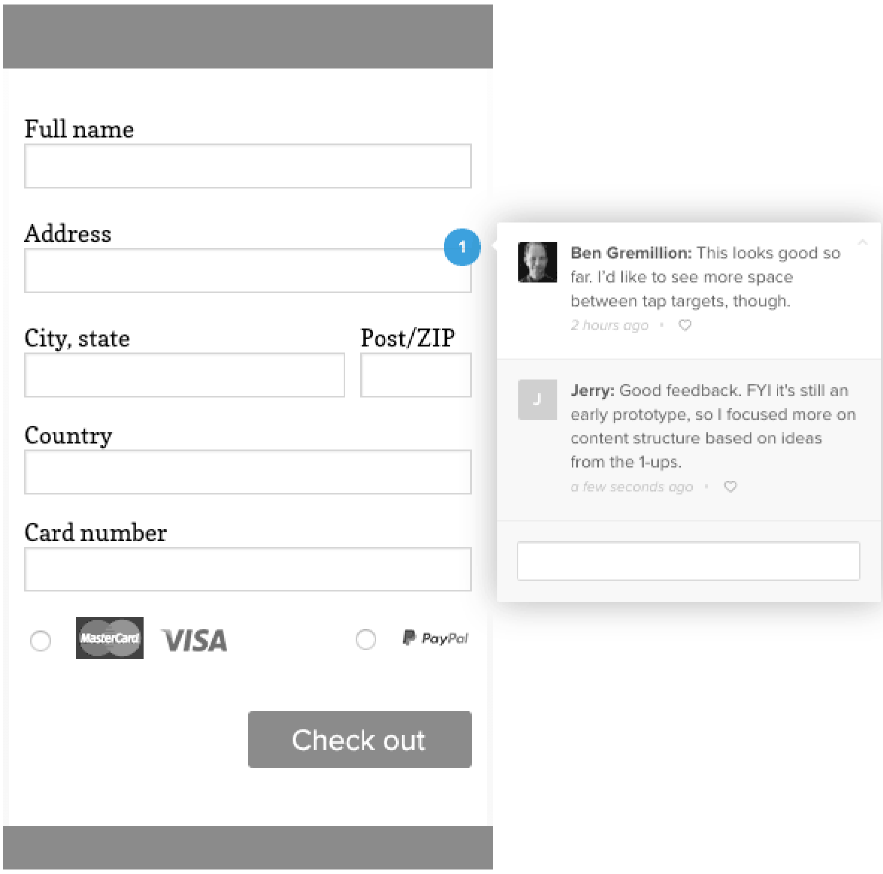
Photo credit: *Alecsandru Grigoriu*

It's vitally important not to stray too far away from what the group co-designed. If you do, you'll be undermining what the group collectively decided, which defeats the purpose of co-design.

Once you've decided how the solutions should be prototyped, you'll be ready to create them in your rapid prototyping tool of choice.

Try to digitally create the prototype with as little effort as possible. For example, only prototype the screens that are needed. If you're not asking your participant to look at the 'About Us' page, then don't prototype it.

If you’ve created your prototype in UXPin, you can now share it with stakeholders and clients for feedback.



Ben Gremillion: This looks good so far. I'd like to see more space between tap targets, though.
2 hours ago • ♡

Jerry: Good feedback. FYI it's still an early prototype, so I focused more on content structure based on ideas from the 1-ups.
a few seconds ago • ♡

Prototype Fidelity

Fidelity refers to the level of realism in a prototype.

For example, a low fidelity prototype could be your designs drawn on paper. A high fidelity prototype could be a prototype that looks and interacts like a real website.

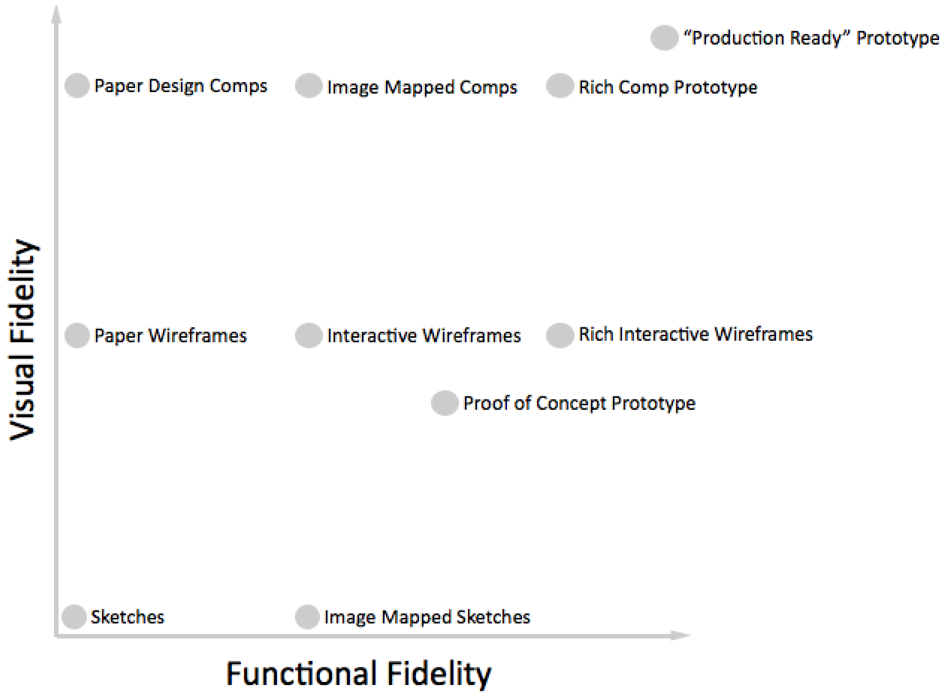


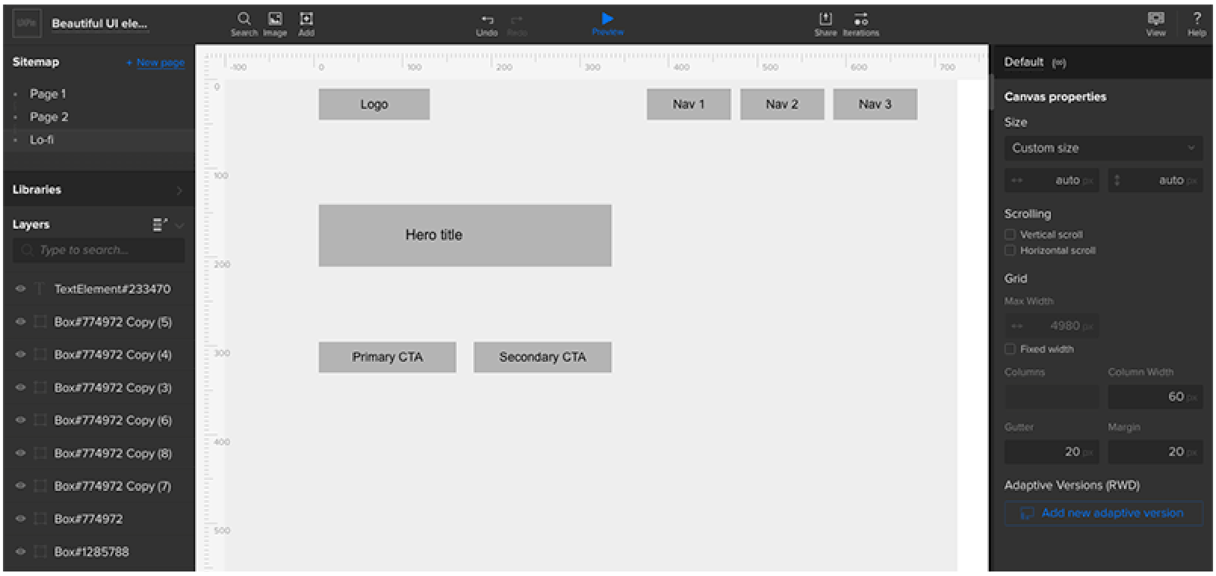
Photo credit: Fred Beecher via Boxes & Arrows. Used with permission.

The higher fidelity you go, the more you'll learn. For example, you'll learn about what colors work best for attracting attention and what users think about your product's look and feel.

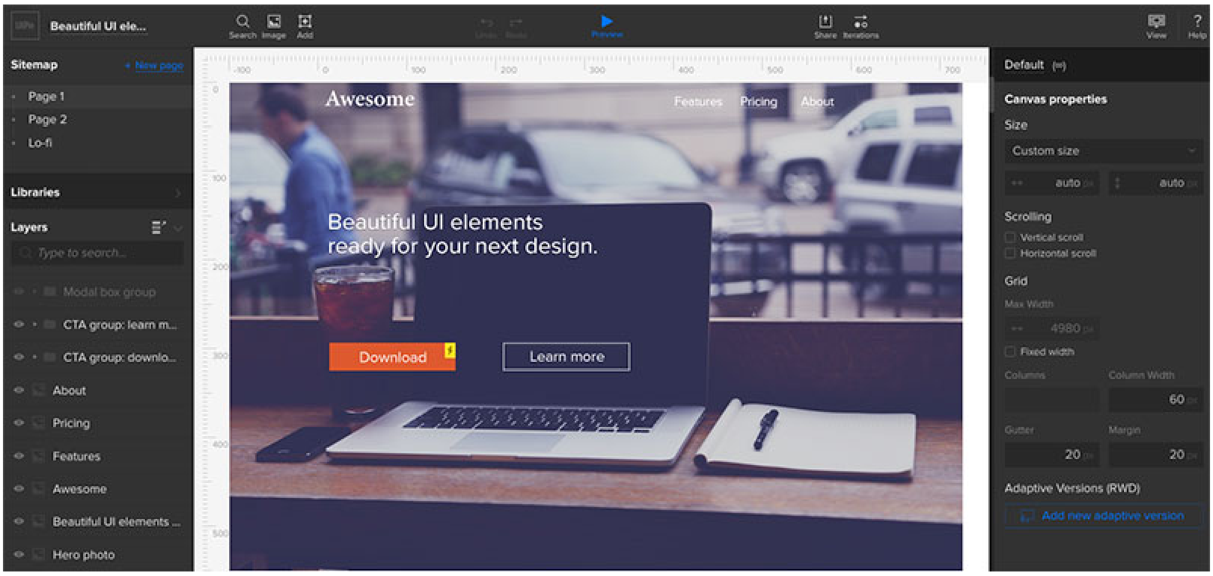
If you have an existing style guide, then it might be easier to go high fidelity because it takes away most of the decisions you have to make. So if it's easy to go high fidelity, then you'll learn more if you do. If you're under time pressure, then it makes sense to go low or mid fidelity–you'll still learn if your solutions answer most of your HMW questions.

For low or mid-fidelity prototypes, focus on the 20% of features that deliver 80% of the value for your HMW question.

Low visual fidelity/ low functional fidelity prototype



High visual fidelity/ high functional fidelity prototype



Physical Prototypes

Physical prototypes follow the same process as digital prototypes except you're physically creating your designs rather than using a prototyping tool.

Make sure you use lightweight materials that can be easily altered. For example, if you're prototyping a parking machine, don't use wood or metal. Instead, use cardboard because it's more portable and easier to modify.

Once again, make sure to balance effort and fidelity. Go too low fidelity and your user testing sessions won't be realistic. Go too high and you'll take too much time.

For example, we once tested a product's packaging by drawing on a box with colored pencils. When we tested the "prototype" with users, they couldn't take it seriously. For the next round, we created images in Adobe Illustrator, printed them on glossy paper and stuck them on high-quality cardboard. This process took the same amount of time as coloring in the box, but made the next usability test more insightful.

Day 8: Usability Testing

Usability testing is the undeniable moment of truth. Now that you've created your prototypes, it's time to get the verdict from users.

We've never uncovered less than 20 key findings the first time we test a prototype, so prepare yourself for serious feedback.

Defining the Testing Roles

You'll want to invite at least one other person to fulfill the two roles in a user testing session.

1. **Facilitator**
   The facilitator is the person who gives the participant tasks and asks them questions. It's the facilitator's job is to uncover exactly what the participant is thinking while making the test feel realistic.

   The designer usually acts as the facilitator.

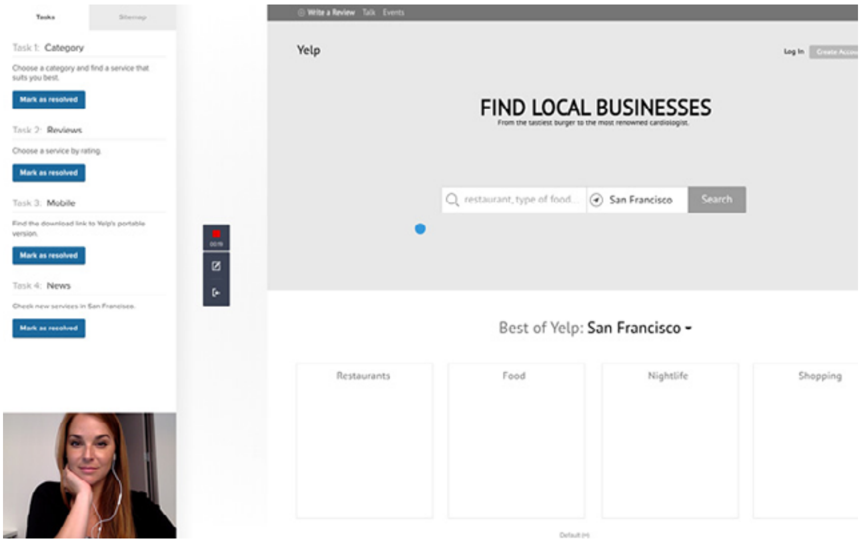
2. **Scribe**

   The scribe records precise notes of user behavior during the testing session. Choose someone who can type fast, since you want to capture feedback verbatim (instead of paraphrasing).

   Don't underestimate this role – if you don't document your findings well, you won't remember what happened and why. The best scribes are people who can type in a quickly to capture everything.

   It's important for other members of the team to observe the testing sessions. This helps them understand firsthand what works and what doesn't, which is more effective than trying to sell them on what happened.

   If what you're testing is digital, you can try multiple apps that record all the user and all their interactions on an Android, iPhone, or laptop screen (e.g. Lookback, UserTesting, even within UXPin itself). An alternative is to use a smartphone, GoPro, or other type of camera to film the session.



Task 1: Category
Choose a category and find a service that suits you best.

Task 2: Reviews
Choose a service by rating.

Task 3: Mobile
Find the download link to Yelp's portable version.

Task 4: News
Check new services in San Francisco.

If you record the session, the scribe isn’t mandatory.

If WiFi is available on-location, you can also stream the test live with tools like Google Hangouts, X-Split, Skype or something similar.

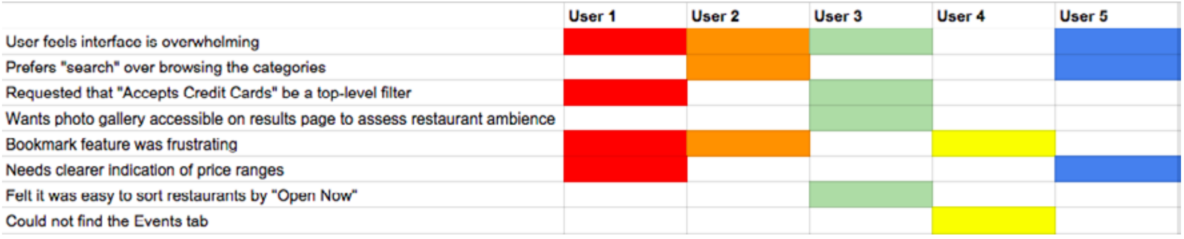
Whether you watch a live or recorded session, try to watch the session as a team. Encourage the team to write their own findings on post-it notes for each test participant:

- Green for areas of positive feedback.
- Pink for negative.
- Blue for personal observations.

Write findings with a thick-tipped sharpie so it's easy to see. More importantly, it forces team members to keep findings concise.

As you write these finding, stick them on a wall for a quick affinity diagram. You'll begin to see many similar findings from different participants. It's a good idea to cluster these findings together. You'll end up with something like this:

If your team can't all join in the same room, you can replicate the exercise with a Rainbow Spreadsheet suggested by former Google UX researcher Tomer Sharon.

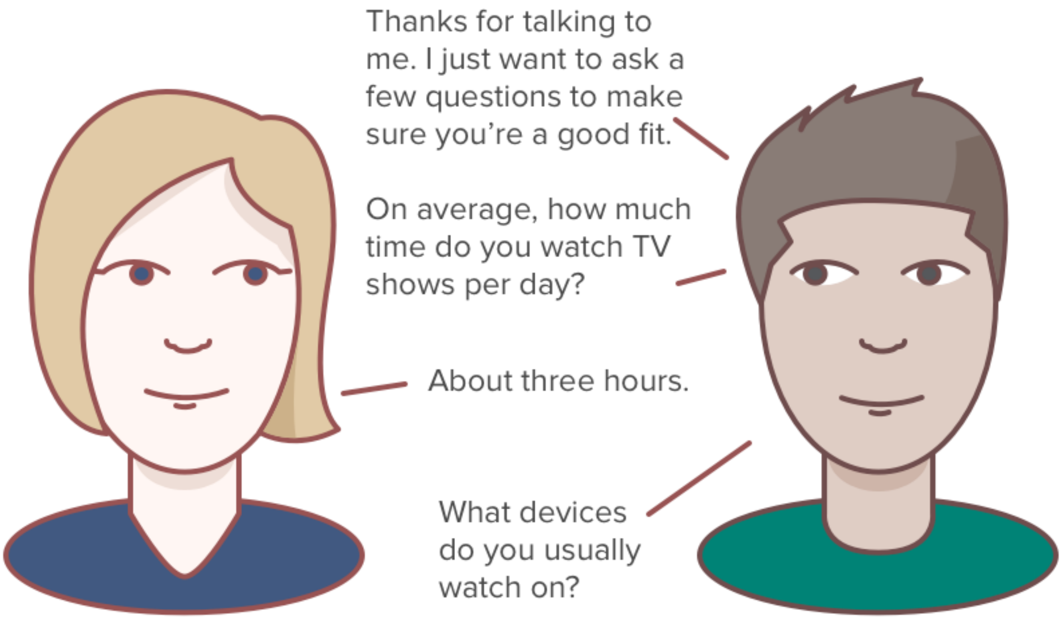


	User 1	User 2	User 3	User 4	User 5
User feels interface is overwhelming	🟥	🟧	🟩		🟦
Prefers "search" over browsing the categories		🟧	🟩		🟦
Requested that "Accepts Credit Cards" be a top-level filter	🟥		🟩		
Wants photo gallery accessible on results page to assess restaurant ambience			🟩		
Bookmark feature was frustrating	🟥	🟧		🟨	
Needs clearer indication of price ranges	🟥				🟦
Felt it was easy to sort restaurants by "Open Now"			🟩		
Could not find the Events tab				🟨	

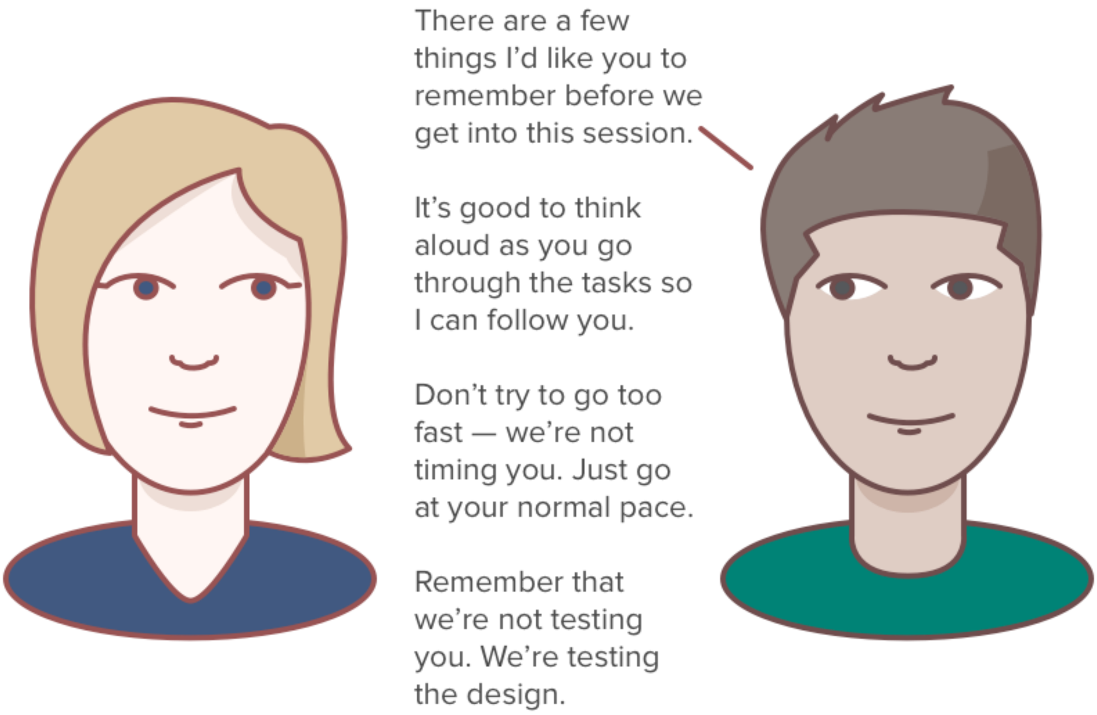
How to Conduct a Moderated Usability Test

Begin with an introduction.

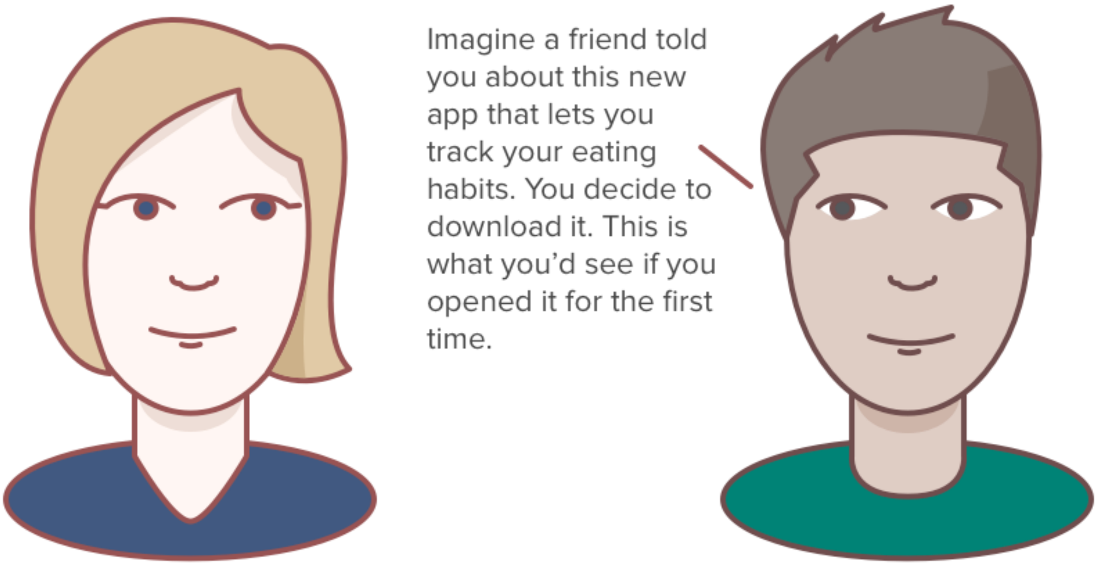
Ask the participant a few questions about themselves to again verify that they fit the right persona group.


Thanks for talking to me. I just want to ask a few questions to make sure you're a good fit.
On average, how much time do you watch TV shows per day?
About three hours.
What devices do you usually watch on?

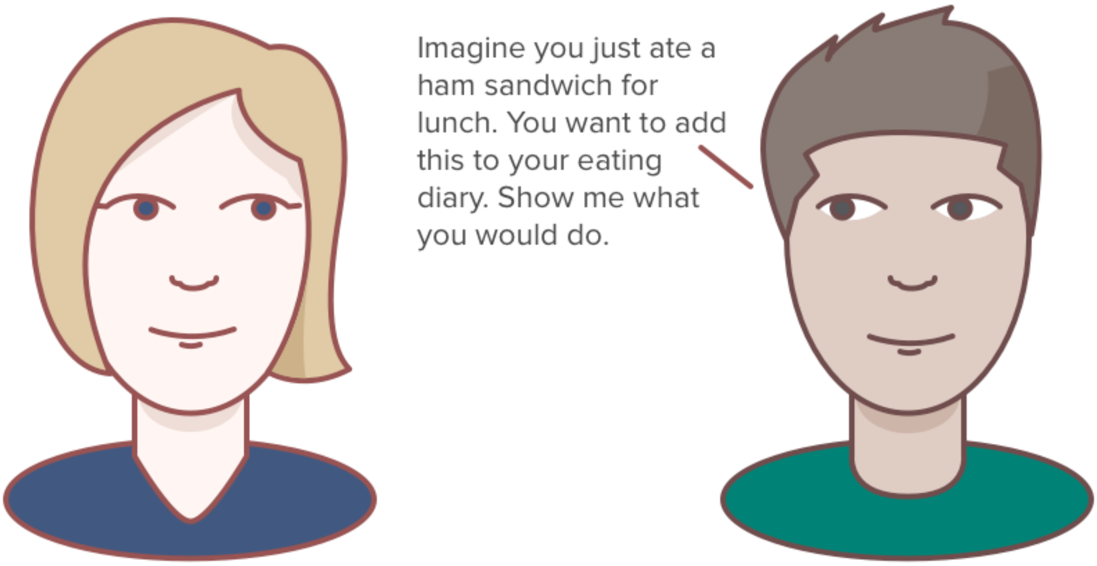
Next, remind them about the principles of user testing.


There are a few things I'd like you to remember before we get into this session.
It's good to think aloud as you go through the tasks so I can follow you.
Don't try to go too fast — we're not timing you. Just go at your normal pace.
Remember that we're not testing you. We're testing the design.

Start the session by setting the scenario. Give the participant a realistic story so they can imagine they're doing this for real. For example, you could say:


Imagine a friend told you about this new app that lets you track your eating habits. You decide to download it. This is what you'd see if you opened it for the first time.

They're now ready for their first task. This could be:


Imagine you just ate a ham sandwich for lunch. You want to add this to your eating diary. Show me what you would do.

Each task should cover an area of your product. Keep going until you've covered everything you want to test.

Finish up by asking three closing questions that aim to discover what the participant thought of the product as a whole, for example:

- "What was one thing you liked the most?"
- "What was one thing that you like the least?"
- "How would you describe this product to a friend?"

Encouraging Insightful Feedback

Throughout the session, prompt your participant to give you feedback by doing the following.

1. **The probe**

   Regularly ask:

   - “What are you thinking now?”
   - “What would you do now?”
   - “What would you do if you were doing this [in the situation you'd normally do this. For example, at home]?”

2. **The echo**

   If the participant says something like:

   "Oh, that's weird."

   The facilitator should say:

   "What's weird?"

   This helps you clearly figure out what the participant is talking about without making assumptions.

3. **The boomerang**

   Participants will ask the facilitator questions or state something to themselves.

It's good to understand what the participant is thinking or would do if the facilitator wasn't present. This is where you throw the question or statement back to them like a boomerang.

"Do I need to register?"

"What do you think?"

4. **The why**

If participants say or do something, ask why.

It will help you understand exactly what's going on which you can use to make design decisions. It's also good to repeat the participant's language to remove the possibility of biasing a participant's response.

"That's cool"

"Why is that cool?"

Following up after the test

After your session concludes, don't forget to email the participant to reiterate the following:

- Details on fulfilling the incentive (e.g. how and when you'll send them a gift card, or swag bag)
- How much you appreciate their insightful feedback

- Assurance that their feedback is completely confidential and won't be shared outside the company

Designer Pro Tip

Follow through is critical because you want to build relationships with your users. You want to leave a positive lasting impression in case you need to contact those users for feedback or testing again. That rapport also facilitates unexpected insights since they might start to email you unprompted product feedback on a regular basis.

Ben Kim, User Researcher at UXPin

Day 9 & Beyond: Iteration

After talking to all your participants, analyzing the sessions with the group and clustering post-it note findings, you should be ready to iterate your designs.

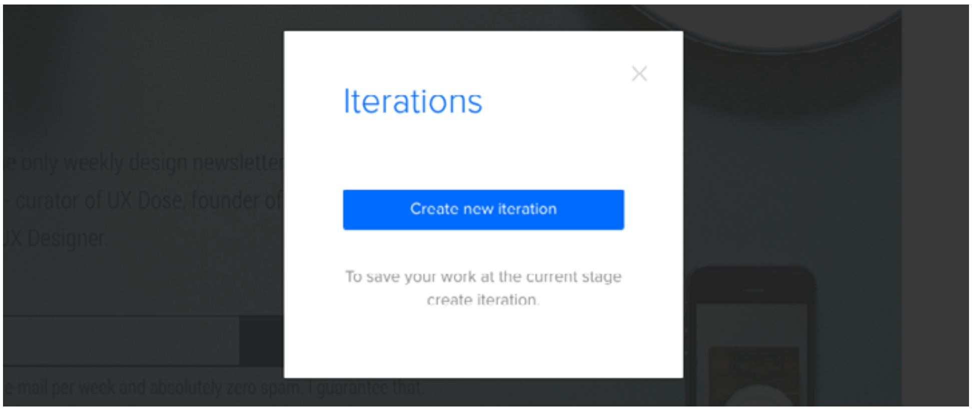
Now you jump straight back into the co-design stage. This second co-design session is similar to the first, except you're iterating your designs based on your user testing findings.

Follow the same process of diverging and converging for each area of your product that needs iteration.

Then test your product again to validate the changes you made to your product. No doubt there will be more findings that will need to be iterated again in the co-design phase.

For future design sprints, you can start tackling the lower priority HMW questions from your backlog. Pass those through the same divergent, convergent, and testing sessions.

If you’re a UXPin user, iteration is pretty straightforward: just click the “Iterations” icon in the top right corner to start on the next version of your prototype while automatically saving the previous one.


Iterations
Create new iteration
e only weekly design newslette
curator of UX Dose, founder of
X Designer.
e mail per week and absolutely zero spam. I guarantee that.

To save your work at the current stage create iteration.

Keep iterating back to the co-design phrase until your participants begin to ask questions like:

- “When is this going to be released?”
- “How much will it cost?”
- “Can I have it now?”

When you reach this stage, your designs are ready to implement in code.

Your work, however, isn’t over. As many have mentioned before: design is a team sport. Review each build with your developer, and show the implemented design to the same team for feedback.

While I’ve presented each step in this guide as part of a 9-day framework, you’re free to adapt the process as best as it fits your needs. As long as you complete the activities, you’ll be making much smarter product decisions.

After Your Sprint: The Retro

The retro is a classic Agile exercise in which the whole team gathers around to review learnings. The learning sessions are incredibly important for iterating your own design process.

The format is quite simple and informal. Follow the Start-Stop-Keep format with 3 key questions for team members:

- What should we start doing that we haven't already?
- What did we try that we should stop doing?
- What worked so well that we should keep doing it?

You can also review any relevant metrics for features that shipped. Remember to keep this meeting brief (15-30 minutes).

Of course, for large product teams, you can't ask everyone the same 3 questions above. The meeting would never end.

Instead, appoint key people as "Investigators". Prior to the retro, allow the appointed members a day to review the recent sprint on their own. Encourage them to share insights with other team members prior to the retro. Treat the actual retro as a “meeting of the minds”.

Designer Pro Tip

In a retro, you don’t look for blame or credit, but simply ‘What did we learn from what we just did?’

Jeff Veen, Design Partner at True Ventures

In the next and final section, I’ll explain case studies for Agile UCD so you can see how it adapts to different timelines and projects.

3 Case Studies: How Design Sprints Adapt to Different Scenarios

Agile UCD easily adapts to various situations at various speeds. A few factors always affect the speed of a project: commitment from the project team, how an organization operates, and the availability of key people.

Let's explore some real case studies from the 50+ design sprints I've run at PwC Digital. Due to NDAs, I won't be able to dive into every detail, but I will highlight the key lessons.

Multinational Soft Drink Corporation: 1-Week Agile Design Sprint

1. **Problem**

   The vending machine industry has been in decline ever since 2007 where it generated $23.21 billion–as of 2011 it was generating only $18.96 billion. We can attribute the downturn to greater convenience of purchasing products elsewhere. For example, purchasing the combination of food and a drink at a corner store, café, or restaurant.

2. Goals

With this opportunity, the Australian office of an international soft drink corporation wanted to create the best experience of purchasing their products. Not only did they want to make it more convenient, they also wanted to engage customers and learn from them.

3. Set Up to Succeed

Our team consisted of eight fully committed team members; three vending experts, two UX designers, two marketers, and one product owner. Because of this, we were able to move at a very fast pace.

We dedicated day one to 'Set up to succeed'. During this phase we created a few assumptions we wanted to validate:

- Students would be our target persona
- No one carried around coins to pay at a vending machine
- People didn't purchase products from vending machines because they're unreliable–for example, they often take your money without giving you your purchase

We unpacked all our thinking between 9am and 3pm then went out to validate our assumptions by conducting guerrilla research around the streets of Sydney.

We split into three groups to interview people. In each group, we had one dedicated facilitator who was comfortable with approaching and interviewing participants. The others acted as scribes.

We incentivized participants with soft drinks and ended up talking to 12 people in total.

We finished the day be relaying our results back to the team. We discovered that:

- Not many people could remember the last time they purchased something from a vending machine
- They would usually purchase the company's drinks from a corner store or with food from a Café or restaurant
- Some people wouldn't purchase the drinks because they thought the bottles were bad for the environment
- Some people didn't purchase the drinks products simply because they didn't like them

4. Execution

We started the second day by creating 3 HMW questions that would help us achieve our outcomes:

- How might we create a clear understanding of our solution's benefits and functionality?
- How might we create the easiest way to purchase soft drinks from vending machines?

- How might we engage people to create a habit of using our solution?

We decided to split our execution days into two parts. The first was to co-design solutions that will take into account our validated assumptions and outcomes. The second was to go out and test these solutions.

In order to do this, we co-designed and prototyped in parallel. Once we finished co-designing one HMW question, our UX designer turned it into a prototype that could be tested with users. While he was doing this, we moved onto co-designing the next HMW question.

Our prototyper finished up the prototyping when the rest of the group had lunch.

We were then able to test our solutions. We did this by splitting up into three groups and guerrilla testing in the afternoon.

Our goal was to talk to at least 5 people each. We gave ourselves two hours. We finished up the day by summarizing our findings and communicating them to the rest of the group.

We repeated the this process of co-designing in the morning and testing in the afternoon for the next 7 days.

5. **Results**

   We ended up validating our product with 82 people.

   We discovered that:

   - Our solution should be an app because they can be used for transactions and engagement
   - We had two personas that would use our product
     People who drank at least 2-3 soft drinks a week and wanted the easiest way to pay

     People who drank at least 2-3 soft drinks a week, wanted the easiest way to pay and wanted to be engaged with the app
   - These personas included people of all ages, which supported the fact that personas are based on behaviour

   By the end of the 8 days, our test participants were ask us questions like "when will this be released?", "this is awesome", and "I would definitely use this".

A Major New Zealand Bank: 1-Week Agile Design Sprint

1. **Problem**

   A major bank in New Zealand hadn't updated their foreign exchange portal in over five years. The portal didn't provide experienced traders with enough informative trading data.

2. **Goals**

   Create a simple but professional, humanized process for trading currency.

3. **Set Up to Succeed**

   This project began by running the 'Set Up to Succeed' phase on day one.

   We discovered that our solution needed to strike the right balance between our two personas: confident traders and not-so-confident traders.

   Our confident traders consisted of people like CFOs who only used the our service to actually make a trade–they got most of the data to make a trade elsewhere.

   Our not-so-confident traders consisted of small business accountants who didn't have much formal training and would often become quite anxious about making the right trade.

   We assumed that our traders wanted:

   - A fast and easy service
   - Help making decisions
   - To have trust and confidence in what they're doing

We produced two main HMW questions from this phase:

- How might we allow non confident traders to be confident when making trades?
- How might we provide our confident traders with enough information to make smart trades?

4. Execution

Our client team was very busy, so could only commit two days a week during the execution phase.

We made sure that these two days consisted of a full day of testing and a full day co-designing solutions to what we discovered during testing. The rest of the time our two UX designers prototyped the solutions.

5. Results

After iterating over three weeks and testing with 15 users, we discovered that our confident traders did want a easy service, but not too easy.

This is because they wanted a certain level of complexity in order for them to make educated decisions on when to make the right trade. This persona also didn't want too much help making decisions. Confident traders felt that if they received recommendations on what trades to make instead of having the raw data, they wouldn't be able to get as much of an advantage on the rest of the market.

Both our personas felt uncomfortable when the service was too fast–especially when trading large amounts of money. They gained more confidence when the service took time to 'process' the data.

By the end of the project, we were getting comments like; "Wow, that was easy", "This is so much better than the current version", and "When is this going to be released?"

Fonterra: 3-Month Agile Design Sprint

1. Problem

Fonterra's farmers weren't getting the right information from the company to improve their milk quality and production. This information is vital because milk can easily become infected by bad bacteria which can result in tankers having to dump hundreds of liters of milk.

2. Goals

Our goal was to provide dairy farmers with accurate information about their cows and milk from the milking shed to the paddock.

3. Set Up to Succeed

We begun the 'Set Up to Succeed' phase by creating assumptions. A few key ones included:

- Farmers don’t use smartphones because their jobs require a lot of 'hands on' work.

- We questioned if a smartphones would even work on a farm due to the lack of cellular reception.
- We would need to create a large obvious design to cater for farmers' large and rugged hands.

It became clear that we needed to immediately go talk to some dairy farmers to learn more about their situation.

After visiting a bunch of farms we discovered that:

- A lot of farmers are actually quite tech savvy; they have the latest smartphones and know how to use them
- Farms also have very good cellular reception (97% of New Zealanders can get 3G data)

4. Execution

We then co-designed solutions to explore what data farmers needed and when.

This was followed by testing on local farms and remote testing through Skype to test with other farmers around New Zealand. This project wasn't able to move as fast as the other case studies because of the geographical distance between farms which impacted our ability to testing.

We eventually transitioned from running through the Agile UCD process to developing our smartphone app solution. We continued to design and test solutions throughout development.

5. **Results**
   When the app was released, it only received positive feedback. A few quotes from farmers included; “Bloody brilliant”, “Awesome!”, and “This is the best thing Fonterra has ever done.”

Conclusion

Agile design sprints are an adaptable process that works in almost any situation from large corporations to small start-ups.

The process is also multipurpose–it can help design anything from websites and apps to parking machines, store layouts or even career paths.

If you have a good understanding of the whole process, you’ll be able to adapt it to any situation at any speed.

Then, it’s only a matter of time before you’ve arrived at the best possible solution for the right problem.

HBO & UXPin Case Study

Designers Increase Creative Time, Shrink Review Cycles

The Challenge

When UX designer Tracy Dendy moved from a small e-commerce company to HBO, she quickly learned one key thing about the entertainment behemoth: Although the industry giant has millions of microsites with their own branding and messaging, they all ultimately speak the same “language” of HBO’s style.

At HBO, Dendy works with various teams of developers, product managers and designers depending on what microsite she is working on. Design reviews were static before Dendy arrived – wikis or wireframes that were vague at best.

With an established international brand and reputation for high-quality creative, vague was not cutting it for the UX team.

The Solution

Dendy’s HBO team now uses UXPin as its de facto collaboration tool, with “let’s UXPin it” emerging as the new shorthand for meaningful reviews. Developers love the fact that designers no longer just pass off sketch files but instead create with developers as a team rather than cogs in a machine.

> *I came from a small company where, as the only designer, I alone was responsible for re-doing the user experience of the entire site," Dendy said. "I wanted a lot of animation and interaction, and after a review of many tools, I chose UXPin because it helped me quickly and easily build what I wanted, no crazy drop-down files needed like in Proto.io. I brought UXPin with me to HBO, and my team has loved it. When you're trying to be creative and still fit an overall style guide, it's impossible to do any kind of meaningful collaboration around static designs. That's how bad animations are created.*

Perhaps the biggest reason Dendy loves UXPin is that it has freed her from code, a language she is not fluent in and which bogs down her design process. With UXPin, she is free to dream and build entire prototypes. In fact, Dendy often "practices" with UXPin when brainstorming new designs.

The Results

- **Increased productivity for designers and developers alike**, with no one needing to learn the other's skills
- **Enhanced collaboration**, with "let's UXPin it" as the new shorthand for getting meaningful products done fast

- **Unbridled creativity**, with a tool that thinks like a designer, HBO UX designer Bundy can now build what she dreams rather than what fits into limited drop-down menu choices

Want UXPin to help your team? Check out our Enterprise Plan.

UXPin

ENTERPRISE

✓ **Create and Collaborate.**
Translate requirements into product features that resonate with customers.

✓ **Simplify your Process.**
Centralize projects and people into one clear workflow.

✓ **Empower your Team.**
Guide creativity with a common design language.

Take a look