

MVP scoping workshop



MOONCASCADE



Contents

Let's start with a definition	3
When should you do MVP scoping?	4
Who will you need?	5
Decision maker	6
Technical expert	6
Designer/usability expert	6
Business stakeholders	7
The cherry on top: a customer expert	7
What will you get out of this workshop?	8
How will you achieve these outcomes?	9
Agenda	10
Day One	12
Business overview	12
Product overview	13
Customer overview	13
User flow schema	14
Day Two	17
User Stories	17
Prioritizing	20
Wireframes	23
After the workshop	27
The Memo	27
Time and money estimates	27
Good luck!	29



Let's start with a definition

If you're reading about MVP scoping right now, you probably already know that MVP stands for Minimum Viable Product. You probably have an idea of what an MVP is, too. But let's define the term anyway, as we'll be moving away from the definition by Eric Ries that most people are familiar with. He defines a minimum viable product as a

“ version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort.”

In Ries' definition, MVPs are all about maximizing learning and minimizing effort. The great thing is that you can accomplish this without building an actual product. And that should always be the first step you take when testing the waters for something completely new. In fact, if you have an idea for something completely new in mind, stop reading this article and have a read through this one instead: [Discover the 4 types of Minimum Viable Product](#). It'll give you some great ideas for testing your product without actually building it!

In this article, we'll be focusing on the part where you're already convinced that you should invest in building something real. Once your market validation has been done, for example, or if you're doing something like building a sub-product for an existing business. Our definition will be the Wikipedia one:

“ A minimum viable product (MVP) is a product with just enough features to satisfy early customers, and to provide feedback for future product development.”



When should you do MVP scoping?

In any project, the bulk of your investment will go to developing the product. Developing as in actual coding done by smart, cool programmer dudes and chicks. Those programmers aren't cheap! Even if you're founding a startup and happen to code yourself, you probably know that your alternative cost is pretty huge. (If you're a solid programmer, come work for Mooncascade instead and we'll pay you handsomely!) So you want to make sure that time spent on coding is as efficient as possible for your investment to be worth it.

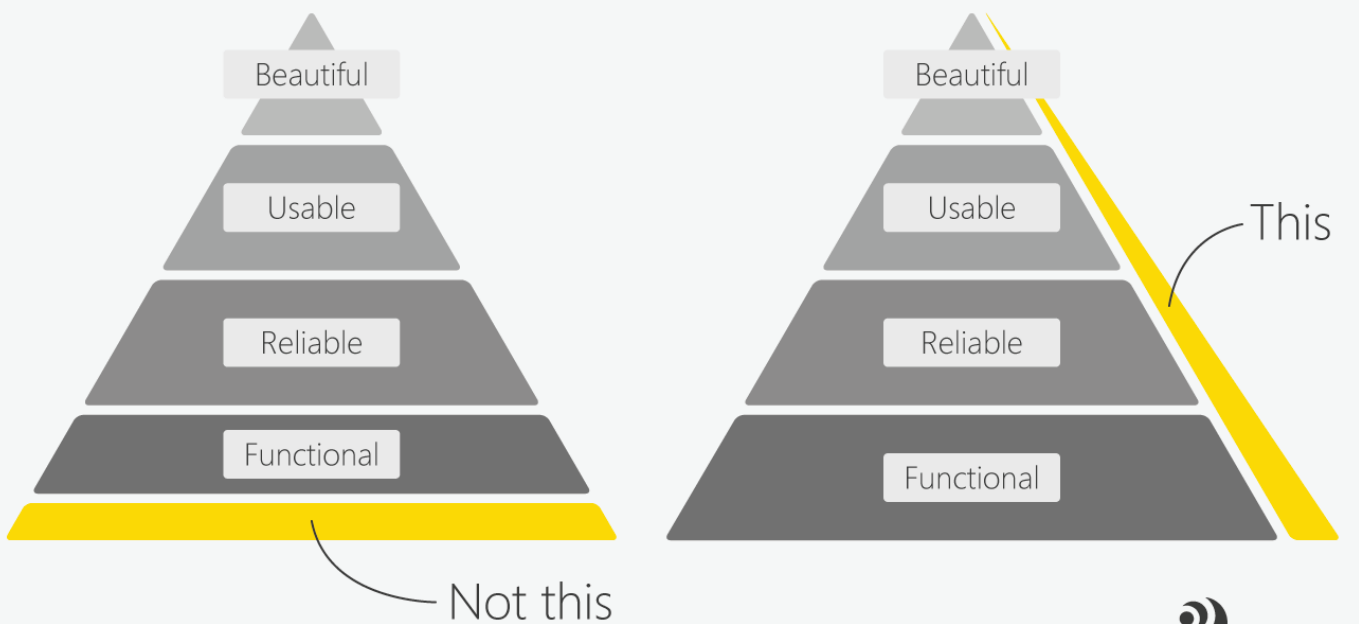
This can only be done if you've given some serious thought to MVP scoping beforehand. That said, you want to be agile, too. Avoid taking the route of those government agencies who spend months analyzing every inch of their product and producing dozens of pages of documentation, only to realize later on that they should've been more flexible from the start.

What we'd suggest instead is a two-day intensive MVP workshop with your product's most important stakeholders. This article should give you a rough overview of what's covered during a typical workshop, but if you feel that you need an experienced facilitator to get the most out of the process, Mooncascade does offer this as a [service](#) as well.

Who will you need?

Our goal in an MVP workshop is to find the slice of the product that will generate value for you without any extras added on top. It should be a version of your product that's usable and looks nice without making you waste effort on crazy animations or pixel-perfect design. Its architecture should be something you can build on and as automated as possible, but don't over-engineer by trying to accommodate millions of users just yet.

Minimum Viable Product (MVP)



To reach this goal, you'll need technical people, usability experts, and business stakeholders.

Decision maker

Make sure you have the person who owns the budget on board. This is the moment to ask for their time, even if they object to participating for two whole days. Having them fully involved now will significantly reduce the amount of emails, confirmations, and miscommunications your team has to face later, and it'll help you kick-start your project as quickly as possible.

Technical expert

You'll also need someone who has a solid understanding of any systems you currently use and the know-how to expand your technical setup in the future. MVP workshops aren't technical—we won't decide which programming language or architecture you should use. But it always surprises business people just how much technical nuances can affect the way a product works. It's essential to have someone behind the table who can object to a seemingly brilliant idea because they know it'll be needlessly expensive to build.

Designer/usability expert

Invite someone who isn't just competent in usability but who can also draw out your first wireframes. You can talk about user stories and personas all you want but visual prototypes are what makes your product's requirements understandable to all parties involved. We don't need high fidelity mocks just yet, just a user flow that can be visualized on a screen or a storyboard. In my experience, this is the only way to make sure everyone is on the same page.

During MVP workshops organised by Mooncascade, we usually bring on the designer ourselves. If the designer is just discovering the project as

they join the workshop, it creates the added benefit of having someone look at the problem from a customer's perspective.

Business stakeholders

You'll know who this is in your team. They're usually product managers, CEOs, sales directors—whoever follows up on your KPIs and sales numbers. They're the person who knows how your product will be monetized and what its targets are. Usually these people know business requirements well and can help with questions like: should we offer a five euro credit to first-time users, etc.

The cherry on top: a customer expert

If you can, bring in a customer expert. This is usually a person from your customer support or customer service team who knows your customer's everyday pains. If the product isn't on the market yet, it could be someone from your most relevant customer segment or someone who's done the research and knows your customers well. We need a champion for the end users! Often, what the business wants to sell and what the customer wants to buy aren't the same thing. Getting a business-oriented and customer-oriented person in the room together can make magic happen.

If you get one person per role, that's a total of five people. You might see some roles overlap but it's more likely that you'll have several people covering one role. Aim for a maximum of seven people around the table. Anything more will just kill your team's productivity.

And don't forget that you'll most likely have an eighth person in the room as well. That's the facilitator. This person isn't a participant, nor a person with an opinion, but someone who makes sure that value is de-

livered during the workshop's two days. It's possible to participate and facilitate at the same time, but it isn't easy. The facilitator needs to be really good at cutting pointless discussions short, making objective conclusions, creating a valuable memo, pushing for decisions, and ensuring that everything on the agenda is covered in time.

In Mooncascade's workshops, we provide a facilitator. You'll see that having a neutral person lead the meetings really helps enable productivity.

What will you get out of this workshop?

This workshop has three major outcomes:

- **A feature backlog for the first phase of development**

This list of features serves two needs:

- It's detailed enough that the items on your list can be groomed into development tasks.
- It's high level enough that it can be turned into a roadmap for your MVP's scope, which can in turn be presented to important stakeholders and investors.

- **Wireframes/storyboard**

I'm a big advocate of visualizing the feature backlog. This has two main benefits:

- 1) Better communication—it's so much easier to explain your features if you have something to show. Wireframes or a storyboard ensure that those attending the workshop understand the product in the same way. Plus, these can be used in a presentation for your stakeholders after the meeting.

2) Saved money: the user flow's design has a big impact on the price of development. The technical person in your group may zone out while you're discussing blue skies and fluffy unicorns, but they'll recognise a monster in disguise from the wire-frames. Spotting issues early allows you to find solutions right then and there.

- **An estimate for the budget and timeline**

Budget—the unsexy part of creating a new product. The moment you've finished painting a picture of the most awesome product ever, your investors will ask you how much it's going to cost them, how much it's going to make, and whether it can be done by quarter three or not. Talk about a turn-off.

But don't worry—once you've completed the MVP workshop, you'll be well-equipped to answer questions like these.

How will you achieve these outcomes?

After pages of why, what, when, and who, we've finally arrived at the how. I'll suggest we follow the rough agenda that Mooncascade's MVP workshops typically use. I'll go into detail for each topic on the agenda as much as possible, but in the end, it's all about your unique situation, your unique group, and your unique product. Before any of our workshops, we work with our customers to make sure the workshop will serve their needs and fine-tune the agenda accordingly. As long as the workshop produces its three main outcomes, your "how" can be flexible.



Agenda

Day One

- Business overview
- Product overview
- Customer overview
- User flow schema

Day Two

- User Stories
- Prioritizing
- Wireframes

After the workshop

- The Memo
- Time and money estimates
- Remember!

Day One



Day One

The first three sessions will include overviews of your business, product, and customer. They share some overlapping subjects and at times may seem similar. To generate unique value from each session, the facilitator will need to structure your discussions and be inflexible about going off-topic.

Business overview

Always start with a business overview. This is a session dedicated to discussing where you're playing and who the other players are. I'm talking industry-level here.

- **What sector is your company in? What is your target geographical area? What is your sector's current situation in the world? How about in your target area?**
 - If relevant: What is your current situation on the market? Think market share, revenue trends, etc.
- **Who are the main competitors in your sector and how do they differ?**
 - Compare competitors both in your geographical area and in the world.
 - Make sure to cover competitors in the industry, not competitors for a specific product.
- **What is your main source of revenue?**
 - Make sure you're talking about company-level revenue, not about the product specifically.
- **What are future trends in your sector?**
- **What are the biggest risks/constraints for that business?**

It's likely that everyone in the room already knows the answers to these questions. Still, I think it's a good idea to start the discussion here, to remind everyone of the bigger picture.

A good way to bring freshness to this discussion is to have an external expert call in for half an hour and share their knowledge about the industry and its trends. Having someone from outside the team point out important nuances will often spark an exciting conversation in the workshop.

Product overview

Now we'll move on to the actual product you're planning to build.

- Which pain is your product solving?
- Who has that pain? How many people have it?
- How is your product going to solve that pain?
- What is your revenue model?
- What are similar products on the market? How are you going to be better?

It's important to have a thorough discussion here. If anyone is left wondering "is this worth it?" after this session, you may need to take time to do more market validation before moving onto the development phase.

Customer overview

In this session, we'll zoom in on the end-user. We briefly discussed who has the pain you'll be solving in the product overview, but this time we'll go much deeper. Describe the different types of users and different types of pain you'll be dealing with. If relevant for your product, describe how the end-user and the person paying for the product are different.

This discussion should lead to writing down three to five main end-user scenarios. I don't want to call them personas, because creating a proper persona should start with tons of interviews and quantitative data, which you most likely won't have at this stage. (If you already have a proper set of personas, dedicate this session to making sure you understand them).

A user scenario could consist of:

- User type (e.g., B2B or B2C, personal use or buying for someone, demographic group if relevant, etc.)
- User-specific pain
- User's willingness to pay
- User's reason for not using the competitor's offer or why they may be trying many options

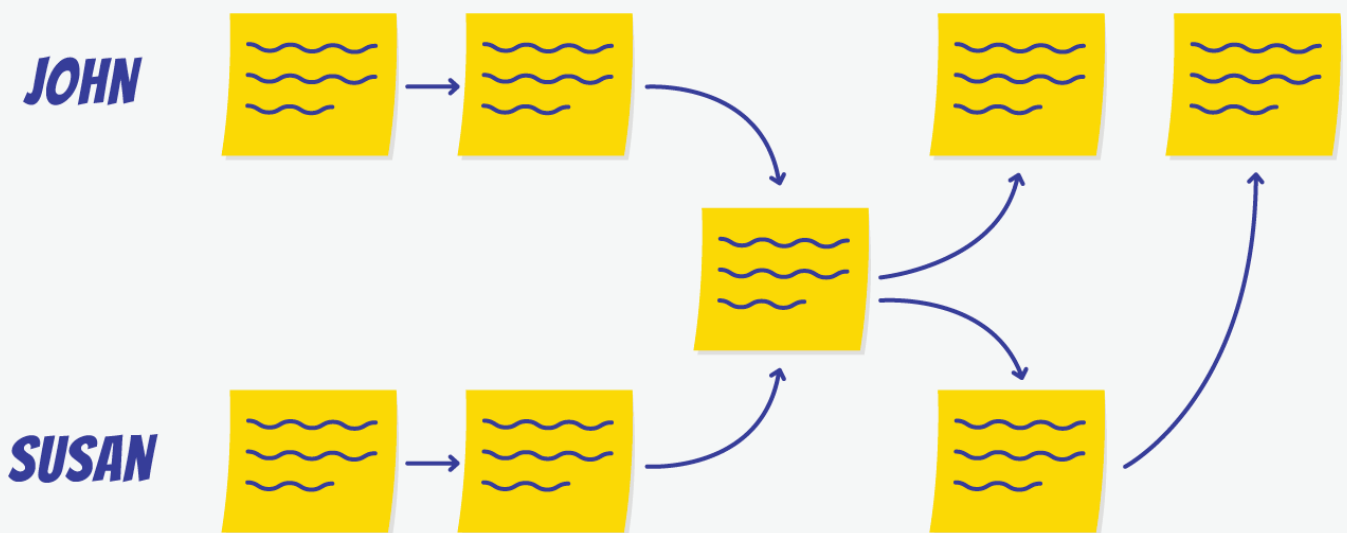
User flow schema

Create a map of your user flow with everyone present. It should be a simple diagram—if there are more actors in play than the end-user, list them on the left-hand side of the page. Then write down the very first and very last step of the flow. E.g., from the user opening the homepage to the user successfully making a purchase. Now map out the road to getting there, keeping all the important lessons learned during previous sessions in mind. Try not to use more than ten steps total, to keep this high level enough.

This should give you an idea of your product's basic requirements. Any important action that needs to be taken outside of your designed product should appear at this point. The journey's significant decision points

should also be clear. Keeping this process within ten steps will ensure that each step counts. Take your time with this exercise, as you'll regularly refer back to this schema over the course of the workshop.

As this exercise is largely inspired by the map teams make in a design sprint, here's a good exercise for creating one: [The Design Sprint Note-n-Map](#).



Day Two

Day Two

User Stories

Now that you have the core of your product mapped out in front of your eyes, it's time to get into some detail work.

How to write an agile user story



- 1 Define your **end user**
Who will be using your product?



- 2 Specify what **they want**
What solution are you offering?



- 3 Describe **the benefit**
What will your user gain from using your product?



- 4 Add **acceptance criteria**
What determines this story as "done"?

Top tips to create a good user story

- Develop a persona profile to visualise your end user
- Always write from your end user's perspective
- Avoid adding technical details too early
- Try not to add too many acceptance criteria
- Keep stories brief, breaking them up if you need to
- Make sure they meet your "definition of done"



MOONCASCADE

First of all, present the basic user story template to your workshop's participants.

The most common way to write a user story is: as a <user role>, I want <goal>, so that <benefit>, followed by the acceptance criteria that need to be met.

- 1) **User role** helps differentiate the different types of users (e.g., B2B vs B2C, admin user vs. end-user, etc.)
- 2) **Goal** should give you the gist of what's being done.
- 3) Writing down **the benefit** helps ensure everyone understands the “why” behind this user story.
- 4) **Acceptance criteria** is where you'll list the story's rules. If the proposed solution follows all of the rules, then you can accept it.

E.g., as an end-user, I want to login, so that I can see my personal information.

Acceptance criteria:

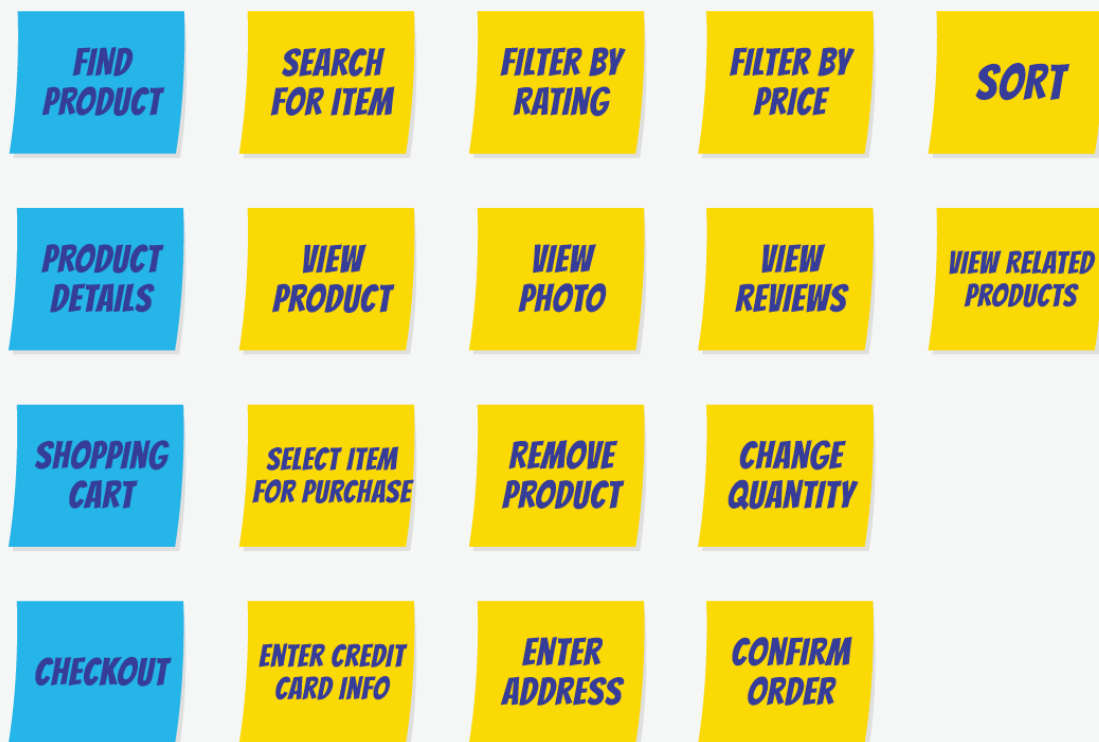
- At least two login methods
- User account locked in case of too many failed attempts
- User account details stored safely

Now, time to work alone. Assign each participant a step from the user flow schema. If you have many steps, assign two steps per person. If your group is big, assign more than one person to the most important steps.

Have each participant write down every user story they can think of on a big post-it note. The most basic happy day scenarios will be done within the first ten minutes. Give them twenty more minutes! This will ensure you get some more in-depth scenarios as well. Encourage them to think

of all the ways the step could go wrong and have them write it down. Our goal is to get as many stories as possible!

After thirty minutes of alone time, it's time to get back into some group work. The next step is user story mapping.



Write down each step from your user flow schema on a different-colored post-it. This will be the story map's epic, and stories will be stacked in columns underneath it. When the story map is done, moving left to right should take you in chronological order through the steps that need to be accomplished. If you see that the user stories under one epic are independent parts of the flow, make sure to have a new column for each step.

Have everyone introduce their user stories and stick their post-it under a relevant epic on the map. Make time for other participants to suggest stories for this epic. Be sure that the discussion is not spent criticizing someone's work. Instead, write down each different opinion as a new story.

Once every story has been mapped out, you can almost smell the workshop's first tangible outcomes.

Prioritizing

Let's take our participants on a trip to MoSCoW!

On this trip, we'll prioritize the user stories we've made and pretty much create the MVP's feature backlog.

The best way to do this is to put together a matrix on a whiteboard, mapping out the importance and effort of each of your stories. Have "essential" and "nice to have" on the tips of the vertical axis and "low effort" and "high effort" on the tips of the horizontal axis. Place each of your user story post-its in the corresponding quadrant.



Reveal the titles of each axis: must have, should have, could have and won't have. Explain what each one means.

Mo

MUST HAVES

The most vital things you can't live without.

S

SHOULD HAVES

Things you consider as important, but not vital.

Co

COULD HAVES

Things that are nice to have.

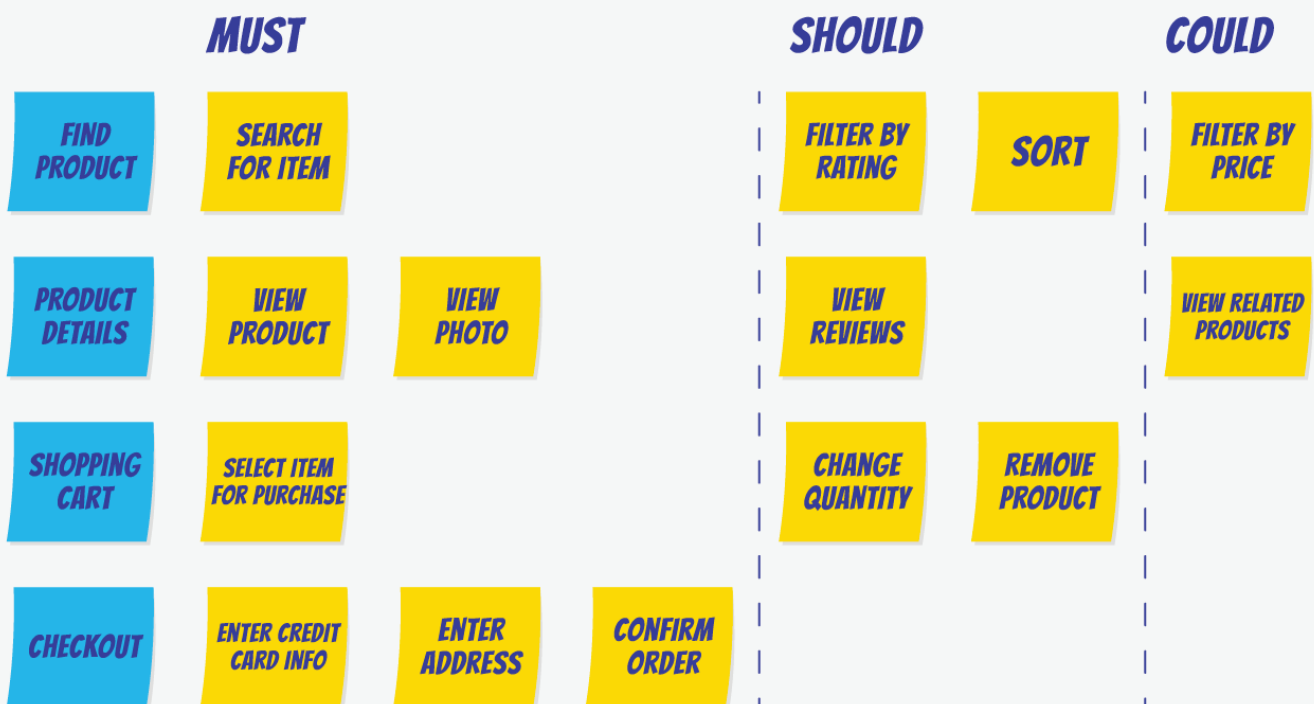
W

WON'T HAVES

Things that provide little to no value you can give up on.

Now divide your user story map into three levels (must, should, and could) and start putting your user stories back onto the map, in that order. Discuss this all the way through—some stories may look like they don't have a place in the matrix but will be must-haves for legal reasons. Some should-haves can end up being must-haves and some must-haves can end up being won't-haves.

PRIORITIZE



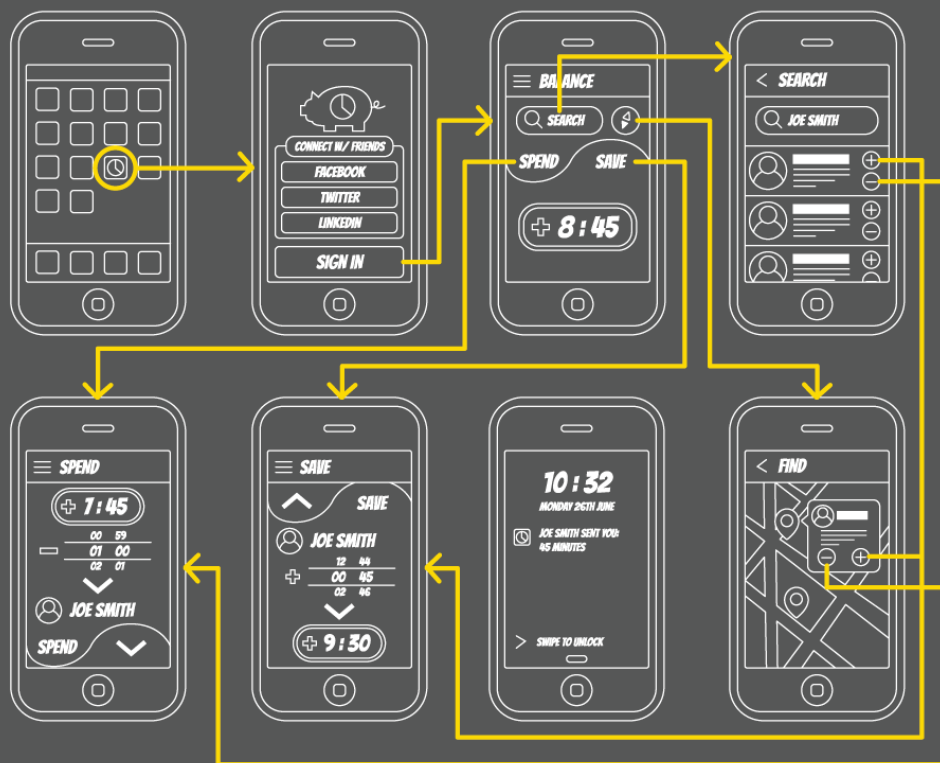
After this session, everyone should have a pretty clear idea of what the MVP will look like, because the MVP will only consist of must-haves.

Wireframes

Everyone should now have a pretty good idea of what the MVP will look like. The only problem is that they're all visualizing different things! It's time to draw things out, to give everyone a unified, visual understanding of the project.

We'll start this session by giving everyone a break, except for the designer/UX expert, who was assigned the task of drawing the product at the start of the workshop. Talk it through with them to decide how much time they need, but don't give them more than an hour. This will push them to get to the very core of the product and not spend time on details like the exact position or color shade of a button.

Let them choose their tools. I've seen designers create pretty impressive digital mock-ups within this timeframe and others stick to basic wireframes on a whiteboard. Just remind them to use a method that allows for making changes during the discussion.



Once the allocated time has run out, meet with the entire group to discuss the wireframe.

Hot tip! The wireframe needs to be narrated. Don't let the designer narrate! They'll be able to explain their way out of confusing moments, but the wireframe needs to be crystal clear without explanations. Instead, choose the most passive or least talkative person in the group to do this. You'll kill two birds with one stone: you'll ensure that this person is included in the discussion and you'll confirm your wireframe can be understood without explanations.

The design should be updated immediately whenever the narrator gets confused or misunderstands it. Make sure to add important comments and details to the wireframe. You don't want the loudest person in the room to be the only one criticizing it. You should trust the designer's UX expertise and you should validate any changes you make with the entire group.

The wireframes will look ugly after this session because of all the comments and quick changes that have been added to them. That's okay, they'll be updated and made presentable as soon as the memo happens.

You've now reached two out of three outcomes! Unfortunately, time has run out and the workshop is over. What about those budget and timeline estimates!?

After the workshop



After the workshop

It's important to tie up the workshop's loose ends once these two intense days have come to a close.

The Memo

Capturing the important takeaways from these two days should be done during the workshop and in front of everyone. During the business session, write your key points down on a whiteboard. Create user personas on big posters. Draw out the user flow schema and make sure it's visible on the wall the entire time. This is helpful to keep the workshop as productive as possible.

After the workshop, all of this should be digitized so that it can be referred back to. Take photos of your whiteboards and posters and create a written memo with illustrations. This will be tedious and time-consuming. Suck it up! This needs to happen or the workshop will lose half of its value—human memory is short and subjective.

In Mooncascade's workshops, this is the facilitator's job. Once you've re-written 70+ user stories yourself, you'll truly appreciate having someone else do this.

Send the memo out to everyone!

Time and money estimates

Like I said at the start, the bulk of your investment will go to developing the product. That's why the technical team should be responsible for estimating the budget after the workshop has ended. Let them take time to do this once they've rested and are far from all the business

people putting pressure on them. This is something that Mooncascade's technical team will do if you have us organize the workshop for you.

Have the technical team go through the wireframes and the must-have user stories. Have them estimate each story and write down any technical stories that need to happen in the background. Each team will usually have their preferred way of doing this. T-shirt estimations, story points, whatever works for them.

The important thing to note is that it's possible to assign time and price dimensions to each estimate. Include buffers for any possible unknowns and don't forget time spent on meetings, communication, testing, translations, marketing, and so on. Include it all in the estimate and you'll walk away with a rough budget and rough timeline for the entire project.

Remember!

This estimate is a gut feeling! Don't trust it completely! If you need to allocate an actual budget, add 30% just to be safe.

In agile development, your project's scope will change because your needs will change. It's also impossible to foresee all the dependencies, integrations, and blockers you're bound to encounter. So don't blame the tech team if their estimate isn't 100% accurate. Then why do this at all, you may ask? Because you need to. You need to have some kind of understanding of how much your idea will cost and how much time it will take to realize. It's impossible to plan your time and activities otherwise. Just make sure that the promises you make have some flexibility.



Good luck!

There you have it! Two intense days spent with the stakeholders, some hard work after it, and your MVP is ready to be developed. I hope that your product is an unicorn and wish you all the strength needed for your business to be agile and adaptive.

Au revoir!

If you have any questions or you would like to learn how to book a MVP scoping workshop for your business, [get in touch](#). We'll reply within one business day and organize a call to learn more about your needs.

About the author

[Karin Sarv](#) is a Product Manager at Mooncascade. Helping our clients to build digital products, her playground is at the intersection of technology, business and UX. Karin is certain that caring for the end-user helps to make great decisions about tech and business as well.

About us

[Mooncascade](#) is a design and product development company that's focused on building high-impact products with and for its clients. We have built over 200 products in the past 10 years for companies like TransferWise, Monese, Merck, Swedbank, Bolt and others. We specialise in fintech, mobility and smart cities to help companies bring tomorrow's technologies to the market today.



MOONCASCADE

www.mooncascade.com