



Every business idea starts with the vision. Some visions are smaller, like “I want to build a coffee house in the middle of Munich”. Others are bigger: “I want to build the world’s biggest search engine”. Well, independent of the scope of your vision, you will need to probably communicate it to others. And this is especially important in an innovative setting. You don’t know a lot about the context, about your idea, and you need to evoke the right images with others. So, how do you do that and what are pitfalls? Let’s look at a very simple example. Meet Max. Max is a visionary. Max envisions the biggest and best coffee house in Munich and he tells his friend Lisa about it.

So, what do you think Lisa has on her mind when he talks about his coffee house idea? Well, Lisa just recently came back from Vienna and her image is quite clear. She imagines an old style nineteenth century coffee house with waiters dressed up in suits. You get coffee and cake, you rest there for hours and obviously read free newspapers. However, Max recently came back from a trip from the United States and he enjoyed the coffee shop culture very much over there. So, in Silicon Valley he met with a lot of business partners and friends and he was amazed by the ease and the efficiency of this business model. And he thinks about selling espresso and bagels and donuts and brownies rather than coffee and cake. Thus, he imagines a hot spot for the local, creative and entrepreneurial scene.

What you were just about to witness is a classical communication problem. Especially in innovative settings, when the term is not quite clear and if you don’t know what you actually want to do, context is crucial.

Context is crucial because words themselves, equal if written or spoken, are ambiguous. Hence, what you can see is that, dependent on your experience and your preconditions, you have a certain image in mind when you hear a certain word. And this is what you keep in mind when you talk about your ideas: Context matters. Otherwise, this leads to miscommunication. So, how can you make sure everybody knows and understands the context? Well, the answer is simple. Build prototypes, enhance your words with things.

Do you know what a *Pokemon* is? Well, probably not, probably yes. Pictures help. This picture of a prototype clearly tells you what it is. Or at least what it could be. Prototypes do exactly that: they provide a holistic perspective. So, they help you think in systems. No product exists on its own. There is always influencing factors. Knowing every of these influencing factors in advance is very difficult. We have to experience it. Prototypes also help you to broaden your horizon. So, you can think out of the box or in completely different scenarios, industries or fields. You can put these prototypes anywhere you want and come up with new ideas. And finally, they help you validate the opportunity criteria. So, you know better what is feasible and what your customers want at a very early stage in your venture creation process.

Isn’t that good? Before you rush out of the building now testing your prototypes, one word of warning: What we call a prototype here is probably not what someone from the industry understands as a prototype.



So, for example if you talk to someone from a manufacturing company, a prototype for them is something that is close to serial production.

This is not where you are. In essence, if we're talking about social prototyping, and that is what the session is about, we are talking much more about the process of prototyping rather than the output. So, the social prototype can be something completely different at a later stage than, for example, the *MVP* [*Minimum Viable Product*] or your final product. And to clarify what we mean, let's look at three different types of prototypes and what we think they are.

When we talk about prototypes, we distinguish social prototypes, minimum viable products, and final products. So, a social prototype is something that is a mockup of the customer's problems. It helps you to clarify what customers mean, together with customers. It is not working in the solution domain yet. *MVP* is something more, it's the minimum feature set. It's something that proposes a solution and also wants to trigger a buying reflex with the customer. You want to figure out who is your customer and what are they willing to pay.

In the end, the final product is something where you want to figure out: Is my business really profitable? Well, next to if you can sell it, obviously. With the final product you know about production costs, you know about the bill of materials. You have a lot of professional knowledge that you don't have in the beginning. But, at the moment this shouldn't matter. Start small, start with a social prototype.

So, you may now be asking, how am I supposed to approach social prototyping? What I am supposed to do? What is the room supposed to look like? What do my participants have to bring? Well, we say all this does not matter. In the beginning the most important thing is that you start. Another idea is to just use papers, scissors, clay and all other material you can find at home. Piece something together. It doesn't need to look fancy or anything. It needs to communicate in your team or to a potential customer group. If you have a little more space, try to build a mood board. Cut out pictures from newspapers. Try to build a vision of the future, a vision of the system your product lives in.

Another idea is to use storyboards. You can draw a service process. You can draw a movie of how your world changes when your product is there. Be creative. In the end, what you also can do with paper, you can build physical prototypes. You can fold it to visualize navigation, you can fold it and visualize *UI* [*User Interface*]. Make a haptic feeling with very little expenditure. You can also visualize navigation and usability by combining all these methods. You can create 3D artefacts. You can create a navigation for your website with a few pieces of paper and a pencil. So, what keeps you from doing it? What is a minimum viable product or *MVP*?

Well, it's the minimum feature set you can sell. And it's a good tool to drastically reduce risk in your business creation process.

If you think back to school, you have always been told that something is only good once it is free of errors.



And you shouldn't make errors. And it should be perfect. But this is not really true in a startup environment, where resources are really scarce and time is crucial. The worst thing that can happen to you is premature optimization. So, sticking to something for a very long time, trying to optimize something that is not really needed or required. If you don't work against the right requirements, you may be running out of cash much quicker than you can actually spell the word premature optimization.

Let's imagine you want to build a mobility device. Your customer interviews, your observations and also your social prototypes have all yielded at a strong need for people to get from A to B. So, how do you build an *MVP* for that? Would you start trying to sell them cars, or would you start with something simpler? And we would strongly argue to you start with something simpler that takes less capital expenditure and helps you in a similar manner to figure out which needs are actually wanted by your customers?

So, you could think about starting with a skateboard and evaluate needs step by step. The need is that people want to be mobile, get from A to B. The question is: Will they also pay for this? For a skateboard probably not as much as for a car. And you could end up with the result that comfort and safety is much more important as a sales argument than just getting from A to B. When developing software and also hardware, it's important that you develop it iteratively.

This means, don't fall into the trap of premature optimization. But also think about what you can use that is readily available out there. Piece together existing things, existing source code, existing products, existing gadgets, stuff that you find on the street. Whatever you can use to showcase the function of your final product works for you.

In addition, quality is none of your concerns in the beginning. If it's free from error or working perfectly, it doesn't matter. And also, the costs. Your final costs, the costs for the products you sell are a function of the scalability of your products, not of the original resources you used for the prototype or for your *MVP*. Hence, how can you go about this? For software services, if you want to showcase functionality, even if you're not a programmer, just create a presentation, link to slides. Sometimes a good Excel template is good enough.

You don't need to be the world's best hacker to showcase the functionality of a UI. The same goes for hardware. Think about what you can do with the workshop of your dad. Or think about what 3D printing or a laser cutter can already do for you. There is no need to start with the heavy machinery, with a *CNC* [*Computer Numerical Control*] mill or a wood workshop. You can do that little by little, but only once you validated the features and that the quality is actually needed. Think about our example, from the skateboard to the car.

So, you've built something. How do you know you built the right thing?

Well, you can find out, but this is more of a research question. And we suggest to find out, you conduct experiments and define real quantifiable measures.



Why is that good? Well, because a lot of entrepreneurs fall into the trap of the confirmation bias. So, they will only look at things and see things they already think are true. And this is not what we want. You sometimes want to be disconfirmed. So, when specifying a quantifiable matrix with your team, you can be sure that you have agreed on a common threshold for when something is true. And this is a good way to discuss which features, which type of needs are really the most important ones. Yet, you built something, and you measured something.

However, learning happens only when the actions that you inferred from the results of your experiments actually lead to improvement in your business creation process. Hence, that you test the right hypothesis, find the right needs, or are able to do all that more quickly. A good way to speed up this process is prioritizing, obviously. So, how can you prioritize? There's two dimensions that we usually think are good for prioritizing features in this setting:

You can think about the ease of implementation of one of the features as one dimension. Thus, how quickly or with how little resources you can implement this feature, now or in the future. The other one is a topical fit. How much value or how much competitive edge does a certain feature provide to your business or to your customers. And if you put these features on a matrix, you will find out that you have certain features that you should have, something that you could have, but that there are certain features that are a must-have for your customers. Those are the ones that are very easy to implement, but create high value for your business in the end. That is what we have been talking about before when we said: Focus on the most important features.

So, what are actually smart ways to build an *MVP*? Well, an *MVP*, a *Minimum Viable Product*, something that you can sell, doesn't have to be a physical product or a working software in the end. Sometimes, other ways are also good to figure out what people are willing to pay for. What they are interested in buying. For example, think about crowd funding campaigns. You have a crowd funding campaign as something that may tell you about the willingness to pay, the amount people are willing to pay, and also what they are willing to pay for. Here are some other ideas for smart ways to building an *MVP*:

You can, for example, set up a simple web page and track the users' behavior. You can figure out what people are actually interested by who is going to your web site. You can put a shopping cart on the web site and put a price to your product. Do that, even though it doesn't exist. And the nice thing is, you can trace every little bit on that page. You can do A/B tests with which product color works better than others, which type of features are important. You can really test your value proposition and figure out, what is the most important thing that your customers want to hear in the conversation.

You can also observe your users. Hence, give them your final product, or your *MVP*, or something really sketchy in their hands to test. You observe them, you record them, you tape them, you give them a questionnaire in the end. You will get a lot of data that may help you improve the product that you are producing. Try to get this data at your hand. The worst thing that can happen is that your customers start a conversation with you about why the



product is not as good as they wanted it to be, why they don't understand the questionnaire. That is the best thing that can happen to you.

Probably a little more old-school method would be to just sell the product at a local retailer. Why not go to 10 retailers in your town, and try and ask them to put your product on their shelves. It's free for them. They are allowed to set the price. The only thing that you request from them is a prominent location within their store.

You can then come back after a few days and see what has been sold. Talk to the shop owner and get their feedback.

If you have something really good, you will get a call from the shop owner really quickly, asking you to deliver more.