Article for Module 5

A Minimum Viable Product is Not a Product, it’s a Process

A Minimum Viable Product is not a Product, It’s a Process

Author: Jim Brikman. **Source:** Taken from Y Combinator.

It’s the same story again and again. First, a team comes up with an idea

Next, they build a minimum viable product (MVP) as a proof of concept, spending a lot of time arguing about which features to include or exclude from the MVP. Finally, if the MVP works well, they plan on building the full, mature, stable product.

So what’s wrong with this picture? Why does it all go wrong for so many startups?

The problem is that these teams do not understand the point of an MVP. An MVP is not just a product with half of the features chopped out, or a way to get the product out the door a little earlier. In fact, the MVP doesn’t have to be a product at all. And it’s not something you build only once, and then consider the job done.

> | “You know that old saw about a plane flying from California to Hawaii being off course 99% of the time—but constantly correcting? The same is true of successful startups—except they may start out heading toward Alaska.” —-Evan Williams

An MVP is a process that you repeat over and over again: Identify your riskiest assumption, find the smallest possible experiment to test that assumption, and use the results of the experiment to course correct.

When you build a product, you make many assumptions. You assume you know what users are looking for, how the design should work, what marketing strategy to use, what architecture will work most efficiently, which monetization strategy will make it sustainable, and which laws and regulations you have to comply with. No matter how good you are, some of your assumptions will be wrong. The problem is, you don't know which ones.

In a post-mortem of more than 100 startups, CB Insights found that the number one cause of startup failure (42% of the time) was “no market need." Nearly half of these startups spent months or even years building a product before they found out that they were wrong in their most central assumption: that someone was interested in that product in the first place.

The only way to find that out—the only way to test your assumptions—is to pu product in front of real users as quickly as possible. And when you do, you will of find that you have to go back to the drawing board. In fact, you'll have to go back to the drawing board not just once, but over and over again.

This is not unique to product development. When you're writing a book or an essay, you have to produce many drafts and spend lots of time editing. And when you're writing code, you frequently have to refactor or even rewrite the code.[2] Every creative human endeavor requires an enormous amount of trial-and-error.

In a trial-and-error world, the one who can find errors the fastest wins. Some people call this philosophy "fail fast." At TripAdvisor, we called it "Speed Wins." Eric Ries called it Lean. Kent Beck and other programmers called it Agile. Whatever you call it, the point is to find out which of your assumptions are wrong by getting feedback on your product from real users as quickly as possible.

Whether you're building a product, writing code, or coming up with a marketing plan, you should always be asking yourself two questions:

- What is my riskiest assumption?
- What is the smallest experiment I can do to test this assumption?

MVP-as-a-process, in action

Let's go through an example.

You decide to build a product that allows restaurant owners to create a mobile app for their restaurants in just a few clicks. It'll have a simple drag and drop interface, a bunch of pre-built templates, an events calendar, newsletter, check-ins, photo galleries, real-time chat, integration with review sites, social networks, and Google Maps. And most importantly, it'll offer a way to make reservations, place take-out orders, and use coupons, from which you'll take a small cut as a way to monetize your product. This is going to be awesome!

You find a few friends to join you as co-founders and, if you're a typical startup team, you'll raise some money, lock yourselves in a room for 12 months, and try to build all these features. If you're slightly more savvy, you'll cut a few features that aren't essential for the first launch, so you'll be able to launch your "MVP" in 8 months instead of 12.

And in both cases, you're most likely going to fail.

Why? Well, consider how many assumptions you're making that could turn out to disastrously wrong:

- You spend months figuring out how to launch custom mobile apps for your clients, only to find out that what a restaurant owner really wants is a mobile-optimized website that's easy to find on Google.
- Or, after using all the latest technologies to build real-time chat, you find out that restaurant owners can barely deal with email and don't want to sit at a computer all day.
- Or, worst of all, you might find out that restaurant owners don't want the hassle of dealing with technology and maintaining mobile apps and have no interest in using your product in the first place.

Waiting months to discover these critical flaws is too long. At best, losing that much time is an enormous waste, and at worst, it'll put your company out of business. In the words of Peter Drucker, "There is surely nothing quite so useless as doing with great efficiency what should not be done at all."

Let's try the MVP-as-a-process approach and see if we can do better. We will build the product incrementally, at each stage asking:

- What is my riskiest assumption?
- What is the smallest experiment I can do to test this assumption?

At the very beginning, the riskiest assumption is probably that restaurant owners want to create mobile apps.

Therefore, the very first MVP could be a mockup of such a mobile app -- maybe even one you did on the back of a restaurant napkin (how fitting!) Go around to restaurant owners in your neighborhood and ask them what problems they have with technology. Do they have a mobile app already? If not, why not? Do they want one? How tech savvy are they? Do they understand the benefits? Show them your mockup. Find out if that would be a good solution to their problems.

You might find out there is not enough interest from restaurant owners to make this a viable business. That's a shame, but the good news is that all it cost you was a few hours of conversation instead of months of development. On the other hand, you might find out that restaurant owners are interested not in mobile apps, but in easy-to-create websites. That's progress!

But you're not done yet. Now you must repeat the process to build your next MVP.

- What is your riskiest assumption?

At this point, it's probably that the restaurant owners would actually be willing to pay for such a website. What's the smallest experiment to test this assumption? One idea for the next MVP may be to create static websites by hand for a few of the restaurant owners who expressed interest and see how they respond.3 Do they like it? Are they impressed that the website is already done? How much would they pay to have such a site launched today?

Perhaps, when faced with the immediate prospect of spending money, you'll find out the restaurant owners aren't actually that interested. Well, good thing you learned this from just a few days of work instead of wasting months of development.

Or perhaps you'll find out they are willing to pay. So you accept the payment for a few months of service by cash or check (so you don't waste lots of time building a billing system), launch their sites, and ask them to email you if they need to update any of the website info. Yes, this involves a lot of manual effort on your part. No, this won't scale if the number of customers grows. But when you're a tiny startup, do not be afraid to do things that don't scale. Scaling is a good problem to have, as it means you've built something worth scaling.

But in the meantime, you need to repeat the MVP process one more time.

- What is your riskiest assumption?

At this point it might be that your marketing strategy works. You can't go around in-person to every restaurant in the world. What is the smallest experiment you can do to test this assumption? Your MVP could be a landing page that describes what your product will do, shows off the restaurant websites you built by hand earlier, and lets visitors provide their email address if they are interested in hearing more when you launch. You can then buy a few hundred dollars of ads on Google, Facebook, Twitter, or LinkedIn to drive traffic to your landing page and see what happens.

If potential users won't even give you their email address, then they probably aren't going to give you any money either. It's much easier to discover this by tweaking some text and a few images on a landing page than by rewriting thousands of lines of code in the full product! The earlier you can find errors, the less time wasted on building the wrong thing.

This, in a nutshell, is the MVP process. Whether you're developing a product design, marketing plan, or writing code, always ask:

- What is my riskiest assumption?
- What is the smallest experiment I can do to test this assumption?