

6

TEST

Groupon is one of the fastest-growing companies of all time. Its name comes from “group coupons,” an ingenious idea that has spawned an entire industry of social commerce imitators. However, it didn’t start out successful. When customers took Groupon up on its first deal, a whopping twenty people bought two-for-one pizza in a restaurant on the first floor of the company’s Chicago offices—hardly a world-changing event.

In fact, Groupon wasn’t originally meant to be about commerce at all. The founder, Andrew Mason, intended his company to become a “collective activism platform” called The Point. Its goal was to bring people together to solve problems they couldn’t solve on their own, such as fund-raising for a cause or boycotting a certain retailer. The Point’s early results were disappointing, however, and at the end of 2008 the founders decided to try something new. Although they still had grand ambitions, they were determined to keep the new product simple. They built a minimum viable product. Does this sound like a billion-dollar company to you? Mason tells the story:

We took a WordPress Blog and we skinned it to say Groupon and then every day we would do a new post. It was totally ghetto. We would sell T-shirts on the first version of Groupon. We’d say in the write-up, “This T-shirt will come in the color red, size large. If you want a different color or size, e-mail that to us.” We didn’t have a form to

add that stuff. It was just so cobbled together.

It was enough to prove the concept and show that it was something that people really liked. The actual coupon generation that we were doing was all FileMaker. We would run a script that would e-mail the coupon PDF to people. It got to the point where we'd sell 500 sushi coupons in a day, and we'd send 500 PDFs to people with Apple Mail at the same time. Really until July of the first year it was just a scrambling to grab the tiger by the tail. It was trying to catch up and reasonably piece together a product.¹

Handmade PDFs, a pizza coupon, and a simple blog were enough to launch Groupon into record-breaking success; it is on pace to become the fastest company in history to achieve \$1 billion in sales. It is revolutionizing the way local businesses find new customers, offering special deals to consumers in more than 375 cities worldwide.²



A minimum viable product (MVP) helps entrepreneurs start the process of learning as quickly as possible.³ It is not necessarily the smallest product imaginable, though; it is simply the fastest way to get through the Build-Measure-Learn feedback loop with the minimum amount of effort.

Contrary to traditional product development, which usually involves a long, thoughtful incubation period and strives for product perfection, the goal of the MVP is to begin the process of learning, not end it. Unlike a prototype or concept test, an MVP is designed not just to answer product design or technical questions. Its goal is to test fundamental business hypotheses.

WHY FIRST PRODUCTS AREN'T MEANT TO BE PERFECT

At IMVU, when we were raising money from venture investors, we

were embarrassed. First of all, our product was still buggy and low-quality. Second, although we were proud of our business results, they weren't exactly earth-shattering. The good news was that we were on a hockey-stick-shaped growth curve. The bad news was that the hockey stick went up to only about \$8,000 per month of revenue. These numbers were so low that we'd often have investors ask us, "What are the units on these charts? Are those numbers in thousands?" We'd have to reply, "No, sir, those are in ones."

However, those early results were extremely significant in predicting IMVU's future path. As you'll see in [Chapter 7](#), we were able to validate two of our leap-of-faith assumptions: IMVU was providing value for customers, and we had a working engine of growth. The gross numbers were small because we were selling the product to visionary early customers called early adopters. Before new products can be sold successfully to the mass market, they have to be sold to early adopters. These people are a special breed of customer. They accept—in fact prefer—an 80 percent solution; you don't need a perfect solution to capture their interest.⁴

Early technology adopters lined up around the block for Apple's original iPhone even though it lacked basic features such as copy and paste, 3G Internet speed, and support for corporate e-mail. Google's original search engine could answer queries about specialized topics such as Stanford University and the Linux operating system, but it would be years before it could "organize the world's information." However, this did not stop early adopters from singing its praises.

Early adopters use their imagination to fill in what a product is missing. They prefer that state of affairs, because what they care about above all is being the first to use or adopt a new product or technology. In consumer products, it's often the thrill of being the first one on the block to show off a new basketball shoe, music player, or cool phone. In enterprise products, it's often about gaining a competitive advantage by taking a risk with something new that competitors don't have yet. Early adopters are suspicious of something that is too polished: if it's ready for everyone to adopt.

how much advantage can one get by being early? As a result, additional features or polish beyond what early adopters demand is a form of wasted resources and time.

This is a hard truth for many entrepreneurs to accept. After all, the vision entrepreneurs keep in their heads is of a high-quality mainstream product that will change the world, not one used by a small niche of people who are willing to give it a shot before it's ready. That world-changing product is polished, slick, and ready for prime time. It wins awards at trade shows and, most of all, is something you can proudly show Mom and Dad. An early, buggy, incomplete product feels like an unacceptable compromise. How many of us were raised with the expectation that we would put our best work forward? As one manager put it to me recently, "I know for me, the MVP feels a little dangerous—in a good way—since I have always been such a perfectionist."

Minimum viable products range in complexity from extremely simple smoke tests (little more than an advertisement) to actual early prototypes complete with problems and missing features. Deciding exactly how complex an MVP needs to be cannot be done formulaically. It requires judgment. Luckily, this judgment is not difficult to develop: most entrepreneurs and product development people dramatically overestimate how many features are needed in an MVP. When in doubt, simplify.

For example, consider a service sold with a one-month free trial. Before a customer can use the service, he or she has to sign up for the trial. One obvious assumption, then, of the business model is that customers will sign up for a free trial once they have a certain amount of information about the service. A critical question to consider is whether customers will in fact sign up for the free trial given a certain number of promised features (the value hypothesis).

Somewhere in the business model, probably buried in a single cell in a spreadsheet, it specifies the "percentage of customers who see the free trial offer who then sign up." Maybe in our projections we say that this number should be 10 percent. If you think about it, this is a leap-of-faith question. It really should be represented in giant letters in a bold red font: WE ASSUME 10 PERCENT OF CUSTOMERS WILL SIGN UP.

Most entrepreneurs approach a question like this by building the product and then checking to see how customers react to it. I consider this to be exactly backward because it can lead to a lot of waste. First, if it turns out that we're building something nobody wants, the whole exercise will be an avoidable expense of time and money. If customers won't sign up for the free trial, they'll never get to experience the amazing features that await them. Even if they do sign up, there are many other opportunities for waste. For example, how many features do we really need to include to appeal to early adopters? Every extra feature is a form of waste, and if we delay the test for these extra features, it comes with a tremendous potential cost in terms of learning and cycle time.

The lesson of the MVP is that any additional work beyond what was required to start learning is waste, no matter how important it might have seemed at the time.

To demonstrate, I'll share several MVP techniques from actual Lean Startups. In each case, you'll witness entrepreneurs avoiding the temptation to overbuild and overpromise.

THE VIDEO MINIMUM VIABLE PRODUCT

Drew Houston is the CEO of Dropbox, a Silicon Valley company that makes an extremely easy-to-use file-sharing tool. Install its application, and a Dropbox folder appears on your computer desktop. Anything you drag into that folder is uploaded automatically to the Dropbox service and then instantly replicated across all your computers and devices.

The founding team was made up of engineers, as the product demanded significant technical expertise to build. It required, for example, integration with a variety of computer platforms and operating systems: Windows, Macintosh, iPhone, Android, and so on. Each of these implementations happens at a deep level of the system and requires specialized know-how to make the user experience exceptional. In fact, one of Dropbox's biggest competitive advantages is that the product works in such a seamless

way that the competition struggles to emulate it.

These are not the kind of people one would think of as marketing geniuses. In fact, none of them had ever worked in a marketing job. They had prominent venture capital backers and could have been expected to apply the standard engineering thinking to building the business: build it and they will come. But Dropbox did something different.

In parallel with their product development efforts, the founders wanted feedback from customers about what really mattered to them. In particular, Dropbox needed to test its leap-of-faith question: if we can provide a superior customer experience, will people give our product a try? They believed—rightly, as it turned out—that file synchronization was a problem that most people didn't know they had. Once you experience the solution, you can't imagine how you ever lived without it.

This is not the kind of entrepreneurial question you can ask or expect an answer to in a focus group. Customers often don't know what they want, and they often had a hard time understanding Dropbox when the concept was explained. Houston learned this the hard way when he tried to raise venture capital. In meeting after meeting, investors would explain that this “market space” was crowded with existing products, none of them had made very much money, and the problem wasn't a very important one. Drew would ask: “Have you personally tried those other products?” When they would say yes, he'd ask: “Did they work seamlessly for you?” The answer was almost always no. Yet in meeting after meeting, the venture capitalists could not imagine a world in line with Drew's vision. Drew, in contrast, believed that if the software “just worked like magic,” customers would flock to it.

The challenge was that it was impossible to demonstrate the working software in a prototype form. The product required that they overcome significant technical hurdles; it also had an online service component that required high reliability and availability. To avoid the risk of waking up after years of development with a product nobody wanted, Drew did something unexpectedly easy: he made a video.

The video is banal, a simple three-minute demonstration of the technology as it is meant to work, but it was targeted at a community of technology early adopters. Drew narrates the video personally, and as he's narrating, the viewer is watching his screen. As he describes the kinds of files he'd like to synchronize, the viewer can watch his mouse manipulate his computer. Of course, if you're paying attention, you start to notice that the files he's moving around are full of in-jokes and humorous references that were appreciated by this community of early adopters. Drew recounted, "It drove hundreds of thousands of people to the website. Our beta waiting list went from 5,000 people to 75,000 people literally overnight. It totally blew us away." Today, Dropbox is one of Silicon Valley's hottest companies, rumored to be worth more than \$1 billion.⁵

In this case, the video was the minimum viable product. The MVP validated Drew's leap-of-faith assumption that customers wanted the product he was developing not because they said so in a focus group or because of a hopeful analogy to another business, but because they actually signed up.

THE CONCIERGE MINIMUM VIABLE PRODUCT

Consider another kind of MVP technique: the concierge MVP. To understand how this technique works, meet Manuel Rosso, the CEO of an Austin, Texas-based startup called Food on the Table. Food on the Table creates weekly meal plans and grocery lists that are based on food you and your family enjoy, then hooks into your local grocery stores to find the best deals on the ingredients.

After you sign up for the site, you walk through a little setup in which you identify your main grocery store and check off the foods your family likes. Later, you can pick another nearby store if you want to compare prices. Next, you're presented with a list of items that are based on your preferences and asked: "What are you in the mood for this week?" Make your choices, select the number of meals you're ready to plan, and choose what you care about most

in terms of time, money, health, or variety. At this point, the site searches through recipes that match your needs, prices out the cost of the meal for you, and lets you print out your shopping list.⁶

Clearly, this is an elaborate service. Behind the scenes, a team of professional chefs devise recipes that take advantage of items that are on sale at local grocery stores around the country. Those recipes are matched via computer algorithm to each family's unique needs and preferences. Try to visualize the work involved: databases of almost every grocery store in the country must be maintained, including what's on sale at each one this week. Those groceries have to be matched to appropriate recipes and then appropriately customized, tagged, and sorted. If a recipe calls for broccoli rabe, is that the same ingredient as the broccoli on sale at the local market?

After reading that description, you might be surprised to learn that Food on the Table (FotT) began life with a single customer. Instead of supporting thousands of grocery stores around the country as it does today, FotT supported just one. How did the company choose which store to support? The founders didn't—until they had their first customer. Similarly, they began life with no recipes whatsoever—until their first customer was ready to begin her meal planning. In fact, the company served its first customer without building any software, without signing any business development partnerships, and without hiring any chefs.

Manuel, along with VP of product Steve Sanderson, went to local supermarkets and moms' groups in his hometown of Austin. Part of their mission was the typical observation of customers that is a part of design thinking and other ideation techniques. However, Manuel and his team were also on the hunt for something else: their first customer.

As they met potential customers in those settings, they would interview them the way any good market researcher would, but at the end of each interview they would attempt to make a sale. They'd describe the benefits of FotT, name a weekly subscription fee, and invite the customer to sign up. Most times they were rejected. After all, most people are not early adopters and will not

sign up for a new service sight unseen. But eventually someone did.

That one early adopter got the concierge treatment. Instead of interacting with the FotT product via impersonal software, she got a personal visit each week from the CEO of the company. He and the VP of product would review what was on sale at her preferred grocery store and carefully select recipes on the basis of her preferences, going so far as to learn her favorite recipes for items she regularly cooked for her family. Each week they would hand her—in person—a prepared packet containing a shopping list and relevant recipes, solicit her feedback, and make changes as necessary. Most important, each week they would collect a check for \$9.95.

Talk about inefficient! Measured according to traditional criteria, this is a terrible system, entirely nonscalable and a complete waste of time. The CEO and VP of product, instead of building their business, are engaged in the drudgery of solving just one customer's problem. Instead of marketing themselves to millions, they sold themselves to one. Worst of all, their efforts didn't appear to be leading to anything tangible. They had no product, no meaningful revenue, no databases of recipes, not even a lasting organization.

However, viewed through the lens of the Lean Startup, they were making monumental progress. Each week they were learning more and more about what was required to make their product a success. After a few weeks they were ready for another customer. Each customer they brought on made it easier to get the next one, because FotT could focus on the same grocery store, getting to know its products and the kinds of people who shopped there well. Each new customer got the concierge treatment: personal in-home visits, the works. But after a few more customers, the overhead of serving them one-on-one started to increase.

Only at the point where the founders were too busy to bring on additional customers did Manuel and his team start to invest in automation in the form of product development. Each iteration of their minimum viable product allowed them to save a little more time and serve a few more customers: delivering the recipes and shopping list via e-mail instead of via an in-home visit. starting to

parse lists of what was on sale automatically via software instead of by hand, even eventually taking credit card payments online instead of a handwritten check.

Before long, they had built a substantial service offering, first in the Austin area and eventually nationwide. But along the way, their product development team was always focused on scaling something that was working rather than trying to invent something that might work in the future. As a result, their development efforts involved far less waste than is typical for a venture of this kind.

It is important to contrast this with the case of a small business, in which it is routine to see the CEO, founder, president, and owner serving customers directly, one at a time. In a concierge MVP, this personalized service is not the product but a learning activity designed to test the leap-of-faith assumptions in the company's growth model. In fact, a common outcome of a concierge MVP is to invalidate the company's proposed growth model, making it clear that a different approach is needed. This can happen even if the initial MVP is profitable for the company. Without a formal growth model, many companies get caught in the trap of being satisfied with a small profitable business when a pivot (change in course or strategy) might lead to more significant growth. The only way to know is to have tested the growth model systematically with real customers.

PAY NO ATTENTION TO THE EIGHT PEOPLE BEHIND THE CURTAIN

Meet Max Ventilla and Damon Horowitz, technologists with a vision to build a new type of search software designed to answer the kinds of questions that befuddle state-of-the-art companies such as Google. Google befuddled? Think about it. Google and its peers excel at answering factual questions: What is the tallest mountain in the world? Who was the twenty-third president of the United States? But for more subjective questions, Google struggles. Ask, "What's a good place to go out for a drink after the ball game in my

city?” and the technology flails. What’s interesting about this class of queries is that they are relatively easy for a person to answer. Imagine being at a cocktail party surrounded by friends. How likely would you be to get a high-quality answer to your subjective question? You almost certainly would get one. Unlike factual queries, because these subjective questions have no single right answer, today’s technology struggles to answer them. Such questions depend on the person answering them, his or her personal experience, taste, and assessment of what you’re looking for.

To solve this problem, Max and Damon created a product called Aardvark. With their deep technical knowledge and industry experience, it would have been reasonable to expect them to dive in and start programming. Instead, they took six months to figure out what they should be building. But they didn’t spend that year at the whiteboard strategizing or engage in a lengthy market research project.

Instead, they built a series of functioning products, each designed to test a way of solving this problem for their customers. Each product was then offered to beta testers, whose behavior was used to validate or refute each specific hypothesis (see examples in sidebar).

The following list of projects are examples from Aardvark’s ideation period.⁷

Rekkit. A service to collect your ratings from across the web and give better recommendations to you.

Ninjapa. A way that you could open accounts in various applications through a single website and manage your data across multiple sites.

The Webb. A central number that you could call and talk to a person who could do anything for you that you could do online.

Web Macros. A way to record sequences of steps on websites so that you could repeat common actions, even across sites, and share “recipes” for how you accomplished online tasks.

Internet Button Company. A way to package steps taken on a website and smart form-fill functionality. People could encode buttons and share buttons à la social bookmarking.

Max and Damon had a vision that computers could be used to create a virtual personal assistant to which their customers could ask questions. Because the assistant was designed for subjective questions, the answers required human judgment. Thus, the early Aardvark experiments tried many variations on this theme, building a series of prototypes for ways customers could interact with the virtual assistant and get their questions answered. All the early prototypes failed to engage the customers.

As Max describes it, “We self-funded the company and released very cheap prototypes to test. What became Aardvark was the sixth prototype. Each prototype was a two- to four-week effort. We used humans to replicate the back end as much as possible. We invited one hundred to two hundred friends to try the prototypes and measured how many of them came back. The results were unambiguously negative until Aardvark.”

Because of the short time line, none of the prototypes involved advanced technology. Instead, they were MVPs designed to test a more important question: what would be required to get customers to engage with the product and tell their friends about it?

“Once we chose Aardvark,” Ventilla says. “we continued to run

with humans replicating pieces of the backend for nine months. We hired eight people to manage queries, classify conversations, etc. We actually raised our seed and series A rounds before the system was automated—the assumption was that the lines between humans and artificial intelligence would cross, and we at least proved that we were building stuff people would respond to.

“As we refined the product, we would bring in six to twelve people weekly to react to mockups, prototypes, or simulations that we were working on. It was a mix of existing users and people who never saw the product before. We had our engineers join for many of these sessions, both so that they could make modifications in real time, but also so we could all experience the pain of a user not knowing what to do.”⁸

The Aardvark product they settled on worked via instant messaging (IM). Customers could send Aardvark a question via IM, and Aardvark would get them an answer that was drawn from the customer’s social network: the system would seek out the customer’s friends and friends of friends and pose the question to them. Once it got a suitable answer, it would report back to the initial customer.

Of course, a product like that requires a very important algorithm: given a question about a certain topic, who is the best person in the customer’s social network to answer that question? For example, a question about restaurants in San Francisco shouldn’t be routed to someone in Seattle. More challenging still, a question about computer programming probably shouldn’t be routed to an art student.

Throughout their testing process, Max and Damon encountered many difficult technological problems like these. Each time, they emphatically refused to solve them at that early stage. Instead, they used Wizard of Oz testing to fake it. In a Wizard of Oz test, customers believe they are interacting with the actual product, but behind the scenes human beings are doing the work. Like the concierge MVP, this approach is incredibly inefficient. Imagine a service that allowed customers to ask questions of human

researchers—for free—and expect a real-time response. Such a service (at scale) would lose money, but it is easy to build on a micro scale. At that scale, it allowed Max and Damon to answer these all-important questions: If we can solve the tough technical problems behind this artificial intelligence product, will people use it? Will their use lead to the creation of a product that has real value?

It was this system that allowed Max and Damon to pivot over and over again, rejecting concepts that seemed promising but that would not have been viable. When they were ready to start scaling, they had a ready-made road map of what to build. The result: Aardvark was acquired for a reported \$50 million—by Google.⁹

THE ROLE OF QUALITY AND DESIGN IN AN MVP

One of the most vexing aspects of the minimum viable product is the challenge it poses to traditional notions of quality. The best professionals and craftspersons alike aspire to build quality products; it is a point of pride.

Modern production processes rely on high quality as a way to boost efficiency. They operate using W. Edwards Deming's famous dictum that the customer is the most important part of the production process. This means that we must focus our energies exclusively on producing outcomes that the customer perceives as valuable. Allowing sloppy work into our process inevitably leads to excessive variation. Variation in process yields products of varying quality in the eyes of the customer that at best require rework and at worst lead to a lost customer. Most modern business and engineering philosophies focus on producing high-quality experiences for customers as a primary principle; it is the foundation of Six Sigma, lean manufacturing, design thinking, extreme programming, and the software craftsmanship movement.

These discussions of quality presuppose that the company already knows what attributes of the product the customer will perceive as worthwhile. In a startup, this is a risky assumption to make. Often

we are not even sure who the customer is. Thus, for startups, I believe in the following quality principle:

If we do not know who the customer is, we do not know what quality is.

Even a “low-quality” MVP can act in service of building a great high-quality product. Yes, MVPs sometimes are perceived as low-quality by customers. If so, we should use this as an opportunity to learn what attributes customers care about. This is infinitely better than mere speculation or whiteboard strategizing, because it provides a solid empirical foundation on which to build future products.

Sometimes, however, customers react quite differently. Many famous products were released in a “low-quality” state, and customers loved them. Imagine if Craig Newmark, in the early days of Craigslist, had refused to publish his humble e-mail newsletter because it lacked sufficient high design. What if the founders of Groupon had felt “two pizzas for the price of one” was beneath them?

I have had many similar experiences. In the early days of IMVU, our avatars were locked in one place, unable to move around the screen. The reason? We were building an MVP and had not yet tackled the difficult task of creating the technology that would allow avatars to walk around the virtual environments they inhabit. In the video game industry, the standard is that 3D avatars should move fluidly as they walk, avoid obstacles in their path, and take an intelligent route toward their destination. Famous best-selling games such as Electronic Arts’ The Sims work on this principle. We didn’t want to ship a low-quality version of this feature, so we opted instead to ship with stationary avatars.

Feedback from the customers was very consistent: they wanted the ability to move their avatars around the environment. We took this as bad news because it meant we would have to spend considerable amounts of time and money on a high-quality solution similar to The Sims. But before we committed ourselves to that

path, we decided to try another MVP. We used a simple hack, which felt almost like cheating. We changed the product so that customers could click where they wanted their avatar to go, and the avatar would teleport there instantly. No walking, no obstacle avoidance. The avatar disappeared and then reappeared an instant later in the new place. We couldn't even afford fancy teleportation graphics or sound effects. We felt lame shipping this feature, but it was all we could afford.

You can imagine our surprise when we started to get positive customer feedback. We never asked about the movement feature directly (we were too embarrassed). But when asked to name the top things about IMVU they liked best, customers consistently listed avatar “teleportation” among the top three (unbelievably, they often specifically described it as “more advanced than The Sims”). This inexpensive compromise outperformed many features of the product we were most proud of, features that had taken much more time and money to produce.

Customers don't care how much time something takes to build. They care only if it serves their needs. Our customers preferred the quick teleportation feature because it allowed them to get where they wanted to go as fast as possible. In retrospect, this makes sense. Wouldn't we all like to get wherever we're going in an instant? No lines, no hours on a plane or sitting on the tarmac, no connections, no cabs or subways. Beam me up, Scotty. Our expensive “real-world” approach was beaten handily by a cool fantasy-world feature that cost much less but that our customers preferred.

So which version of the product is low-quality, again?

MVPs require the courage to put one's assumptions to the test. If customers react the way we expect, we can take that as confirmation that our assumptions are correct. If we release a poorly designed product and customers (even early adopters) cannot figure out how to use it, that will confirm our need to invest in superior design. But we must always ask: what if they don't care about design in the same way we do?

Thus, the Lean Startup method is not opposed to building high-

quality products, but only in service of the goal of winning over customers. We must be willing to set aside our traditional professional standards to start the process of validated learning as soon as possible. But once again, this does not mean operating in a sloppy or undisciplined way. (This is an important caveat. There is a category of quality problems that have the net effect of slowing down the Build-Measure-Learn feedback loop. Defects make it more difficult to evolve the product. They actually interfere with our ability to learn and so are dangerous to tolerate in any production process. We will consider methods for figuring out when to make investments in preventing these kinds of problems in [Part Three](#).)

As you consider building your own minimum viable product, let this simple rule suffice: remove any feature, process, or effort that does not contribute directly to the learning you seek.

SPEED BUMPS IN BUILDING AN MVP

Building an MVP is not without risks, both real and imagined. Both can derail a startup effort unless they are understood ahead of time. The most common speed bumps are legal issues, fears about competitors, branding risks, and the impact on morale.

For startups that rely on patent protection, there are special challenges with releasing an early product. In some jurisdictions, the window for filing a patent begins when the product is released to the general public, and depending on the way the MVP is structured, releasing it may start this clock. Even if your startup is not in one of those jurisdictions, you may want international patent protection and may wind up having to abide by these more stringent requirements. (In my opinion, issues like this are one of the many ways in which current patent law inhibits innovation and should be remedied as a matter of public policy.)

In many industries, patents are used primarily for defensive purposes, as a deterrent to hold competitors at bay. In such cases, the patent risks of an MVP are minor compared with the learning benefits. However, in industries in which a new scientific

breakthrough is at the heart of a company's competitive advantage, these risks need to be balanced more carefully. In all cases, entrepreneurs should seek legal counsel to ensure that they understand the risks fully.

Legal risks may be daunting, but you may be surprised to learn that the most common objection I have heard over the years to building an MVP is fear of competitors—especially large established companies—stealing a startup's ideas. If only it were so easy to have a good idea stolen! Part of the special challenge of being a startup is the near impossibility of having your idea, company, or product be noticed by anyone, let alone a competitor. In fact, I have often given entrepreneurs fearful of this issue the following assignment: take one of your ideas (one of your lesser insights, perhaps), find the name of the relevant product manager at an established company who has responsibility for that area, and try to get that company to steal your idea. Call them up, write them a memo, send them a press release—go ahead, try it. The truth is that most managers in most companies are already overwhelmed with good ideas. Their challenge lies in prioritization and execution, and it is those challenges that give a startup hope of surviving.¹⁰

If a competitor can outexecute a startup once the idea is known, the startup is doomed anyway. The reason to build a new team to pursue an idea is that you believe you can accelerate through the Build-Measure-Learn feedback loop faster than anyone else can. If that's true, it makes no difference what the competition knows. If it's not true, a startup has much bigger problems, and secrecy won't fix them. Sooner or later, a successful startup will face competition from fast followers. A head start is rarely large enough to matter, and time spent in stealth mode—away from customers—is unlikely to provide a head start. The only way to win is to learn faster than anyone else.

Many startups plan to invest in building a great brand, and an MVP can seem like a dangerous branding risk. Similarly, entrepreneurs in existing organizations often are constrained by the fear of damaging the parent company's established brand. In either

of these cases, there is an easy solution: launch the MVP under a different brand name. In addition, a long-term reputation is only at risk when companies engage in vocal launch activities such as PR and building hype. When a product fails to live up to those pronouncements, real long-term damage can happen to a corporate brand. But startups have the advantage of being obscure, having a pathetically small number of customers, and not having much exposure. Rather than lamenting them, use these advantages to experiment under the radar and then do a public marketing launch once the product has proved itself with real customers.¹¹

Finally, it helps to prepare for the fact that MVPs often result in bad news. Unlike traditional concept tests or prototypes, they are designed to speak to the full range of business questions, not just design or technical ones, and they often provide a needed dose of reality. In fact, piercing the reality distortion field is quite uncomfortable. Visionaries are especially afraid of a false negative: that customers will reject a flawed MVP that is too small or too limited. It is precisely this attitude that one sees when companies launch fully formed products without prior testing. They simply couldn't bear to test them in anything less than their full splendor. Yet there is wisdom in the visionary's fear. Teams steeped in traditional product development methods are trained to make go/kill decisions on a regular basis. That is the essence of the waterfall or stage-gate development model. If an MVP fails, teams are liable to give up hope and abandon the project altogether. But this is a solvable problem.

FROM THE MVP TO INNOVATION ACCOUNTING

The solution to this dilemma is a commitment to iteration. You have to commit to a locked-in agreement—ahead of time—that no matter what comes of testing the MVP, you will not give up hope. Successful entrepreneurs do not give up at the first sign of trouble, nor do they persevere the plane right into the ground. Instead, they possess a unique combination of perseverance and flexibility. The

MVP is just the first step on a journey of learning. Down that road—after many iterations—you may learn that some element of your product or strategy is flawed and decide it is time to make a change, which I call a pivot, to a different method for achieving your vision.

Startups are especially at risk when outside stakeholders and investors (especially corporate CFOs for internal projects) have a crisis of confidence. When the project was authorized or the investment made, the entrepreneur promised that the new product would be world-changing. Customers were supposed to flock to it in record numbers. Why are so few actually doing so?

In traditional management, a manager who promises to deliver something and fails to do so is in trouble. There are only two possible explanations: a failure of execution or a failure to plan appropriately. Both are equally inexcusable. Entrepreneurial managers face a difficult problem: because the plans and projections we make are full of uncertainty, how can we claim success when we inevitably fail to deliver what we promised? Put another way, how does the CFO or VC know that we're failing because we learned something critical and not because we were goofing off or misguided?

The solution to this problem resides at the heart of the Lean Startup model. We all need a disciplined, systematic approach to figuring out if we're making progress and discovering if we're actually achieving validated learning. I call this system innovation accounting, an alternative to traditional accounting designed specifically for startups. It is the subject of [Chapter 7](#).