Startup Playbook

Written by Sam Altman · Illustrated by Gregory Koberger

INTRODUCTION 2

Part I: THE IDEA 4

Part II: A GREAT TEAM 8

Part III: A GREAT PRODUCT 10

Part IV: GREAT EXECUTION 13

Part IV: Execution - GROWTH 14

Part IV: Execution – FOCUS & INTENSITY 18

Part IV: Execution – JOBS OF THE CEO 21

Part IV: Execution – HIRING & MANAGING 25

Part IV: Execution – COMPETITORS 28

Part IV: Execution – MAKING MONEY 29

Part IV: Execution – FUNDRAISING 31

CLOSING THOUGHT 34

APPENDIX 35

Appendix 1: Projects and Companies 35

Appendix 2: How to hire 36

Appendix 3: Unit Economics 44

Appendix 4: Default Alive or Default Dead? 46

INTRODUCTION

We spend a lot of time advising startups. Though one-on-one advice will always be crucial, we thought it might help us scale Y Combinator if we could distill the most generalizable parts of this advice into a sort of playbook we could give YC and YC Fellowship companies.

Then we thought we should just give it to everyone.

This is meant for people new to the world of startups. Most of this will not be new to people who have read a lot of what YC partners have written—the goal is to get it into one place.

There may be a part II on how to scale a startup later—this mostly covers how to start one.

Part I: The Idea

Part II: A Great Team

Part III: A Great Product

Part IV: Great Execution

Closing Thought

Growth

Focus & Intensity

Jobs of the CEO

Hiring & Managing

Competitors

Making Money

Fundraising

Your goal as a startup is to make something users love. If you do that, then you have to figure out how to get a lot more users. But this first part is critical—think about the really successful companies of today. They all started with a product that their early users loved so much they told other people about it. If you fail to do this, you will fail. If you deceive yourself and think your users love your product when they don't, you will still fail.

The startup graveyard is littered with people who thought they could skip this step.

It's much better to first make a product a small number of users love than a product that a large number of users like. Even though the total amount of positive feeling is the same, it's much easier to get more users than to go from like to love.

A word of warning about choosing to start a startup: It sucks! One of the most consistent pieces of feedback we get from YC founders is it's harder than they could have ever imagined, because they didn't have a framework for the sort of work and intensity a startup entails. Joining an early-stage startup that's on a rocketship trajectory is usually a much better financial deal.

On the other hand, starting a startup is not in fact very risky to your career—if you're really good at technology, there will be job opportunities if you fail. Most people are very bad at evaluating risk. I personally think the riskier option is having an idea or project you're really passionate about and working at a safe, easy, unfulfilling job instead.

To have a successful startup, you need: a great idea (including a great market), a great team, a great product, and great execution.

For more

Growth Hacking stories

Click

collected by

Truong Nguyen

PART I

THE IDEA

Part I: THE IDEA

Home

One of the first things we ask YC companies is what they're building and why.

We look for clear, concise answers here. This is both to evaluate you as a founder and the idea itself. It's important to be able to think and communicate clearly as a founder—you'll need it for recruiting, raising money, selling, etc. Ideas in general need to be clear to spread, and complex ideas are almost always a sign of muddled thinking or a made up problem. If the idea does not really excite at least some people the first time they hear it, that's bad.

Another thing we ask is who desperately needs the product.

In the best case, you yourself are the target user. In the second best case, you understand the target user extremely well.

If a company already has users, we ask how many and how fast that number is growing. We try to figure out why it's not growing faster, and we especially try to figure out if users really love the product. Usually this means they're telling their friends to use the product without prompting from the company. We also ask if the company is generating revenue, and if not, why not.

If the company doesn't yet have users, we try to figure out the minimum thing to build first to test the hypothesis—i.e., if we work backwards from the perfect experience, we try to figure out what kernel to start with.

The way to test an idea is to either launch it and see what happens or try to sell it (e.g. try to get a letter of intent before you write a line of code.) The former works better for consumer ideas (users may tell you they will use it, but in practice it won't cut through the clutter) and the latter works better for enterprise ideas (if a company tells you they will buy something, then go build it.) Specifically, if you are an enterprise company, one of the first questions we'll ask you is if you have a letter of intent from a customer saying they'll buy what you're building. For most biotech and hard tech companies, the way to test an idea is to first talk to potential customers and then figure out the smallest subset of the technology you can build first.

It's important to let your idea evolve as you get feedback from users. And it's critical you understand your users really well—you need this to evaluate an idea, build a great product, and build a great company.

As mentioned earlier, startups are really hard. They take a very long time, and consistent intense effort. The founders and employees need to have a shared sense of mission to sustain them. So we ask why founders want to start this particular company.

We also ask how the company will one day be a monopoly. There are a lot of different terms for this, but we use Peter Thiel's. Obviously, we don't want your company to behave in an unethical way against competitors. Instead, we're looking for businesses that get more powerful with scale and that are difficult to copy.

Finally, we ask about the market. We ask how big it is today, how fast it's growing, and why it's going to be big in ten years. We try to understand why the

market is going to grow quickly, and why it's a good market for a startup to go after. We like it when major technological shifts are just starting that most people haven't realized yet—big companies are bad at addressing those. And somewhat counterintuitively, the best answer is going after a large part of a small market.

A few other thoughts on ideas:

We greatly prefer something new to something derivative. Most really big companies start with something fundamentally new (one acceptable definition of new is 10x better.) If there are ten other companies starting at the same time with the same plan, and it sounds a whole lot like something that already exists, we are skeptical.

One important counterintuitive reason for this is that it's easier to do something new and hard than something derivative and easy. People will want to help you and join you if it's the former; they will not if it's the latter.

The best ideas sound bad but are in fact good. So you don't need to be too secretive with your idea—if it's actually a good idea, it likely won't sound like it's worth stealing. Even if it does sound like it's worth stealing, there are at least a thousand times more people that have good ideas than people who are willing to do the kind of work it takes to turn a great idea into a great company. And if you tell people what you're doing, they might help.

Speaking of telling people your idea—while it's important the idea really excites some people the first time they hear it, almost everyone is going to tell you that your idea sucks. Maybe they are right. Maybe they are not good at evaluating startups, or maybe they are just jealous. Whatever the reason is, it will happen a lot, it will hurt, and even if you think you're not going to be affected by it, you still will be. The faster you can develop self-belief and not get dragged down too

much by haters, the better off you'll be. No matter how successful you are, the haters will never go away.

What if you don't have an idea but want to start a startup? Maybe you shouldn't. It's so much better if the idea comes first and the startup is the way to get the idea out into the world.

We once tried an experiment where we funded a bunch of promising founding teams with no ideas in the hopes they would land on a promising idea after we funded them.

All of them failed. I think part of the problem is that good founders tend to have lots of good ideas (too many, usually). But an even bigger problem is that once you have a startup you have to hurry to come up with an idea, and because it's already an official company the idea can't be too crazy. You end up with plausible sounding but derivative ideas. This is the danger of pivots.

So it's better not to try too actively to force yourself to come up with startup ideas. Instead, learn about a lot of different things. Practice noticing problems, things that seem inefficient, and major technological shifts. Work on projects you find interesting[1]. Go out of your way to hang around smart, interesting people. At some point, ideas will emerge.

[1] *See* Appendix 1

PART II

A GREAT TEAM

Part II: A GREAT TEAM

Home

Mediocre teams do not build great companies. One of the things we look at the most is the strength of the founders. When I used to do later-stage investing, I looked equally hard at the strength of the employees the founders hired.

What makes a great founder? The most important characteristics are ones like unstoppability, determination, formidability, and resourcefulness. Intelligence and passion also rank very highly. These are all much more important than experience and certainly “expertise with language X and framework Y”.

We have noticed the most successful founders are the sort of people who are low-stress to work with because you feel “he or she will get it done, no matter what it is.” Sometimes you can succeed through sheer force of will.

Good founders have a number of seemingly contradictory traits. One important example is rigidity and flexibility. You want to have strong beliefs about the core of the company and its mission, but still be very flexible and willing to learn new things when it comes to almost everything else.

The best founders are unusually responsive. This is an indicator of decisiveness, focus, intensity, and the ability to get things done.

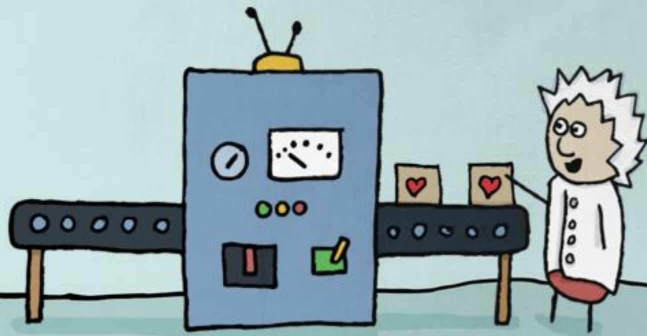
Founders that are hard to talk to are almost always bad. Communication is a very important skill for founders—in fact, I think this is the most important rarely-discussed founder skill.

Tech startups need at least one founder who can build the company's product or service, and at least one founder who is (or can become) good at sales and talking to users. This can be the same person.

Consider these criteria when you're choosing a cofounder—it's one of the most important decisions you'll make, and it's often done fairly randomly. You want someone you know well, not someone you just met at a cofounder dating thing. You can evaluate anyone you might work with better with more data, and you really don't want to get this one wrong. Also, at some point, the expected value of the startup is likely to dip below the X axis. If you have a pre-existing relationship with your cofounders, none of you will want to let the other down and you'll keep going. Cofounder breakups are one of the leading causes of death for early startups, and we see them happen very, very frequently in cases where the founders met for the express purpose of starting the company.

The best case, by far, is to have a good cofounder. The next best is to be a solo founder. The worse case, by far, is to have a bad cofounder. If things are not working out, you should part ways quickly.

A quick note on equity: the conversation about the equity split does not get easier with time—it's better to set it early on. Nearly equal is best, though perhaps in the case of two founders it's best to have one person with one extra share to prevent deadlocks when the cofounders have a fallout.



PART III

A GREAT PRODUCT

Part III: A GREAT PRODUCT

Home

Here is the secret to success: have a great product. This is the only thing all great companies have in common.

If you do not build a product users love you will eventually fail. Yet founders always look for some other trick. Startups are the point in your life when tricks stop working.

A great product is the only way to grow long-term. Eventually your company will get so big that all growth hacks stop working and you have to grow by people wanting to use your product. This is the most important thing to understand about super-successful companies. There is no other way. Think about all of the really successful technology companies—they all do this.

You want to build a "product improvement engine" in your company. You should talk to your users and watch them use your product, figure out what parts are sub-par, and then make your product better. Then do it again. This cycle should be the number one focus of the company, and it should drive everything else. If you improve your product 5% every week, it will really compound.

The faster the repeat rate of this cycle, the better the company usually turns out. During YC, we tell founders they should be building product and talking to users, and not much else besides eating, sleeping, exercising, and spending time with their loved ones.

To do this cycle right, you have to get very close to your users. Literally watch them use your product. Sit in their office if you can. Value both what they tell you and what they actually do. You should not put anyone between the founders and the users for as long as possible—that means the founders need to do sales, customer support, etc.

Understand your users as well as you possibly can. Really figure out what they need, where to find them, and what makes them tick.

"Do things that don't scale" has rightfully become a mantra for startups. You usually need to recruit initial users one at a time (Ben Silbermann used to approach strangers in coffee shops in Palo Alto and ask them to try Pinterest) and then build things they ask for. Many founders hate this part, and just want to announce their product in the press. But that almost never works. Recruit users manually, and make the product so good the users you recruit tell their friends.

You also need to break things into very small pieces, and iterate and adapt as you go. Don't try to plan too far out, and definitely don't batch everything into one big public release. You want to start with something very simple—as little surface area as possible—and launch it sooner than you'd think. In fact, simplicity is always good, and you should always keep your product and company as simple as possible.

Some common questions we ask startups having problems: Are users using your product more than once? Are your users fanatical about your product? Would your users be truly bummed if your company went away? Are your users

recommending you to other people without you asking them to do it? If you're a B2B company, do you have at least 10 paying customers?

If not, then that's often the underlying problem, and we tell companies to make their product better. I am skeptical about most excuses for why a company isn't growing—very often the real reason is that the product just isn't good enough.

When startups aren't sure what to do next with their product, or if their product isn't good enough, we send them to go talk to their users. This doesn't work in every case—it's definitely true that people would have asked Ford for faster horses—but it works surprisingly often. In fact, more generally, when there's a disagreement about anything in the company, talk to your users.

The best founders seem to care a little bit too much about product quality, even for seemingly unimportant details. But it seems to work. By the way, "product" includes all interactions a user has with the company. You need to offer great support, great sales interactions, etc.

Remember, if you haven't made a great product, nothing else will save you.

PART IV

GREAT EXECUTION

Part IV: GREAT EXECUTION

Home

Although it's necessary to build a great product, you're not done after that. You still have to turn it into a great company, and you have to do it yourself—the fantasy of hiring an "experienced manager" to do all this work is both extremely prevalent and a graveyard for failed companies. You cannot outsource the work to someone else for a long time.

This sounds obvious, but you have to make money. This would be a good time to start thinking about how that's going to work.

The only universal job description of a CEO is to make sure the company wins. You can do this as the founder even if you have a lot of flaws that would normally disqualify you as a CEO as long as you hire people that complement your own skills and let them do their jobs. That experienced CEO with a fancy MBA may not have the skill gaps you have, but he or she won't understand the users as well, won't have the same product instincts, and won't care as much.

Part IV: Execution - GROWTH

Home

Growth and momentum are the keys to great execution. Growth (as long as it is not "sell dollar bills for 90 cents" growth) solves all problems, and lack of growth is not solvable by anything but growth. If you're growing, it feels like you're winning, and people are happy. If you're growing, there are new roles and responsibilities all the time, and people feel like their careers are advancing. If you're not growing, it feels like you're losing, and people are unhappy and leave. If you're not growing, people just fight over responsibilities and blame.

Founders and employees that are burnt out nearly always work at startups without momentum. It's hard to overstate how demoralizing it is.

The prime directive of great execution is "Never lose momentum". But how do you do it?

The most important way is to make it your top priority. The company does what the CEO measures. It's valuable to have a single metric that the company optimizes, and it's worth time to figure out the right growth metric. If you care about growth, and you set the execution bar, the rest of the company will focus on it.

Here are a couple of examples.

The founders of Airbnb drew a forward-looking graph of the growth they wanted to hit. They posted this everywhere—on their fridge, above their desks, on their bathroom mirror. If they hit the number that week, great. If not, it was all they talked about.

Mark Zuckerberg once said that one of the most important innovations at Facebook was their establishment of a growth group when growth slowed. This group was (and perhaps still is) one of the most prestigious groups in the company—everyone knew how important it was.

Keep a list of what's blocking growth. Talk as a company about how you could grow faster. If you know what the limiters are, you'll naturally think about how to address them.

For anything you consider doing, ask yourself "Is this the best way to optimize growth?" For example, going to a conference is not usually the best way to optimize growth, unless you expect to sell a lot there.

Extreme internal transparency around metrics (and financials) is a good thing to do. For some reason, founders are always really scared of this. But it's great for keeping the whole company focused on growth. There seems to be a direct correlation between how focused on metrics employees at a company are and how well they're doing. If you hide the metrics, it's hard for people to focus on them.

Speaking of metrics, don't fool yourself with vanity metrics. The common mistake here is to focus on signups and ignore retention. But retention is as important to growth as new user acquisition.

It's also important to establish an internal cadence to keep momentum. You want to have a "drumbeat" of progress—new features, customers, hires,

revenue milestones, partnerships, etc that you can talk about internally and externally.

You should set aggressive but borderline achievable goals and review progress every month. Celebrate wins! Talk internally about strategy all the time, tell everyone what you're hearing from customers, etc. The more information you share internally—good and bad—the better you'll be.

There are a few traps that founders often fall into. One is that if the company is growing like crazy but everything seems incredibly broken and inefficient, everyone worries that things are going to come unraveled. In practice, this seems to happen rarely (Friendster is the most recent example of a startup dying because of technical debt that I can point to.) Counterintuitively, it turns out that it's good if you're growing fast but nothing is optimized—all you need to do is fix it to get more growth! My favorite investments are in companies that are growing really fast but incredibly un-optimized—they are deeply undervalued.

A related trap is thinking about problems too far in the future—i.e. "How are we going to do this at massive scale?" The answer is to figure it out when you get there. Far more startups die while debating this question than die because they didn't think about it enough. A good rule of thumb is to only think about how things will work at 10x your current scale. Most early-stage startups should put "Do things that don't scale" up on their wall and live by it. As an example, great startups always have great customer service in the early days, and bad startups worry about the impact on the unit economics and that it won't scale. But great customer service makes for passionate early users, and as the product gets better you need less support, because you'll know what customers commonly struggle with and improve the product/experience in those areas. (By the way, this is a really important example—have great customer support.)

There's a big catch to this—"Do things that don't scale" does not excuse you from having to eventually make money. It's ok to have bad unit economics in the early days, but you have to have a good reason for why the unit economics are going to work out later.

Another trap is getting demoralized because growth is bad in absolute numbers even though it's good on a percentage basis. Humans are very bad at intuition around exponential growth. Remind your team of this, and that all giant companies started growing from small numbers.

Some of the biggest traps are the things that founders believe will deliver growth but in practice almost never work and suck up a huge amount of time. Common examples are deals with other companies and the "big press launch". Beware of these and understand that they effectively never work. Instead get growth the same way all great companies have—by building a product users love, recruiting users manually first, and then testing lots of growth strategies (ads, referral programs, sales and marketing, etc.) and doing more of what works. Ask your customers where you can find more people like them.

Remember that sales and marketing are not bad words. Though neither will save you if you don't have a great product, they can both help accelerate growth substantially. If you're an enterprise company, it's likely a requirement that your company get good at these.

Don't be afraid of sales especially. At least one founder has to get good at asking people to use your product and give you money.

Alex Schultz gave a lecture on growth for consumer products that's well worth watching. For B2B products, I think the right answer is almost always to track revenue growth per month, and remember that the longer sales cycle means the first couple of months are going to look ugly (though sometimes selling to startups as initial customers can solve this problem).

PART IV: EXECUTION
FOCUS & INTENSITY

Part IV: Execution – FOCUS & INTENSITY

Home

If I had to distill my advice about how to operate down to only two words, I'd pick focus and intensity. These words seem to really apply to the best founders I know.

They are relentlessly focused on their product and growth. They don't try to do everything—in fact, they say no a lot (this is hard because the sort of people that start companies are the sort of people that like doing new things.)

As a general rule, don't let your company start doing the next thing until you've dominated the first thing. No great company I know of started doing multiple things at once—they start with a lot of conviction about one thing, and see it all the way through. You can do far fewer things than you think. A very, very common cause of startup death is doing too many of the wrong things. Prioritization is critical and hard. (Equally important to setting the company's priorities is setting your own tactical priorities. What I've found works best for me personally is a pen-and-paper list for each day with ~3 major tasks and ~30 minor ones, and an annual to-do list of overall goals.)

While great founders don't do many big projects, they do whatever they do very intensely. They get things done very quickly. They are decisive, which is hard when you're running a startup—you will get a lot of conflicting advice, both because there are multiple ways to do things and because there's a lot of bad advice out there. Great founders listen to all of the advice and then quickly make their own decisions.

Please note that this doesn't mean doing everything intensely—that's impossible. You have to pick the right things. As Paul Buchheit says, find ways to get 90% of the value with 10% of the effort. The market doesn't care how hard you work—it only cares if you do the right things.

It's very hard to be both obsessed with product quality and move very quickly. But it's one of the most obvious tells of a great founder.

I have never, not once, seen a slow-moving founder be really successful.

You are not different from other startups. You still have to stay focused and move fast. Companies building rockets and nuclear reactors still manage to do this. All failing companies have a pet explanation for why they are different and don't have to move fast.

When you find something that works, keep going. Don't get distracted and do something else. Don't take your foot off the gas.

Don't get caught up in early success—you didn't get off to a promising start by going to lots of networking events and speaking on lots of panels. Startup founders who start to have initial success have a choice of two paths: either they keep doing what they're doing, or they start spending a lot of time thinking about their "personal brand" and enjoying the status of being a founder.

It's hard to turn down the conferences and the press profiles—they feel good, and it's especially hard to watch other founders in your space get the attention.

But this won't last long. Eventually the press figures out who is actually winning, and if your company is a real success, you'll have more attention than you'll ever want. The extreme cases—early-stage founders with their own publicists—that one would think only exist in TV shows actually exist in real life, and they almost always fail.

Focus and intensity will win out in the long run. (Charlie Rose once said that things get done in the world through a combination of focus and personal connections, and that's always stuck with me.)

Part IV: Execution – JOBS OF THE CEO

Home

Earlier I mentioned that the only universal job description of the CEO is to make sure the company wins. Although that's true, I wanted to talk a little more specifically about how a CEO should spend his or her time.

A CEO has to 1) set the vision and strategy for the company, 2) evangelize the company to everyone, 3) hire and manage the team, especially in areas where you yourself have gaps 4) raise money and make sure the company does not run out of money, and 5) set the execution quality bar.

In addition to these, find whatever parts of the business you love the most, and stay engaged there.

As I mentioned at the beginning, it's an intense job. If you are successful, it will take over your life to a degree you cannot imagine—the company will be on your mind all the time. Extreme focus and extreme intensity means it's not the best choice for work-life balance. You can have one other big thing—your family, doing lots of triathlons, whatever—but probably not much more than that. You have to always be on, and there are a lot of decisions only you can make, no matter how good you get at delegation.

You should aim to be super responsive to your team and the outside world, always be clear on the strategy and priorities, show up to everything important, and execute quickly (especially when it comes to making decisions others are blocked on.) You should also adopt a “do whatever it takes” attitude—there will be plenty of unpleasant schleps. If the team sees you doing these things, they will do them too.

Managing your own psychology is both really hard and really important. It’s become cliché at this point, but it’s really true—the emotional highs and lows are very intense, and if you don’t figure out how to stay somewhat level through them, you’re going to struggle. Being a CEO is lonely. It’s important to have relationships with other CEOs you can call when everything is melting down (one of the important accidental discoveries of YC was a way for founders to have peers.)

A successful startup takes a very long time—certainly much longer than most founders think at the outset. You cannot treat it as an all-nighter. You have to eat well, sleep well, and exercise. You have to spend time with your family and friends. You also need to work in an area you’re actually passionate about—nothing else will sustain you for ten years.

Everything will feel broken all the time—the diversity and magnitude of the disasters will surprise you. Your job is to fix them with a smile on your face and reassure your team that it’ll all be ok. Usually things aren’t as bad as they seem, but sometimes they are in fact really bad. In any case, just keep going. Keep growing.

The CEO doesn’t get to make excuses. Lots of bad and unfair things are going to happen. But don’t let yourself say, and certainly not to the team, “if only we had more money” or “if only we had another engineer”. Either figure out a way to make that happen, or figure out what to do without it. People who let

themselves make a lot of excuses usually fail in general, and startup CEOs who do it almost always fail. Let yourself feel upset at the injustice for 1 minute, and then realize that it's up to you to figure out a solution. Strive for people to say "X just somehow always gets things done" when talking about you.

No first-time founder knows what he or she is doing. To the degree you understand that, and ask for help, you'll be better off. It's worth the time investment to learn to become a good leader and manager. The best way to do this is to find a mentor—reading books doesn't seem to work as well.

A surprising amount of our advice at YC is of the form "just ask them" or "just do it". First-time founders think there must be some secret for when you need something from someone or you want to do some new thing. But again, startups are where tricks stop working. Just be direct, be willing to ask for what you want, and don't be a jerk.

It's important that you distort reality for others but not yourself. You have to convince other people that your company is primed to be the most important startup of the decade, but you yourself should be paranoid about everything that could go wrong.

Be persistent. Most founders give up too quickly or move on to the next product too quickly. If things generally aren't going well, figure out what the root cause of the problem is and make sure you address that. A huge part of being a successful startup CEO is not giving up (although you don't want to be obstinate beyond all reason either—this is another apparent contradiction, and a hard judgment call to make.)

Be optimistic. Although it's possible that there is a great pessimistic CEO somewhere out in the world, I haven't met him or her yet. A belief that the future will be better, and that the company will play an important role in making the future better, is important for the CEO to have and to infect the rest of the

company with. This is easy in theory and hard in the practical reality of short-term challenges. Don't lose sight of the long-term vision, and trust that the day-to-day challenges will someday be forgotten and replaced by memories of the year-to-year progress.

Among your most important jobs are defining the mission and defining the values. This can feel a little hokey, but it's worth doing early on. Whatever you set at the beginning will usually still be in force years later, and as you grow, each new person needs to first buy in and then sell others on the mission and values of the company. So write your cultural values and mission down early.

Another cliché that I think is worth repeating: Building a company is somewhat like building a religion. If people don't connect what they're doing day-to-day with a higher purpose they care about, they will not do a great job. I think Airbnb has done the best job at this in the YC network, and I highly recommend taking a look at their cultural values.

One mistake that CEOs often make is to innovate in well-trodden areas of business instead of innovating in new products and solutions. For example, many founders think that they should spend their time discovering new ways to do HR, marketing, sales, financing, PR, etc. This is nearly always bad. Do what works in the well-established areas, and focus your creative energies on the product or service you're building.

Part IV: Execution – HIRING & MANAGING

Home

Hiring is one of your most important jobs and the key to building a great company (as opposed to a great product.)

My first piece of advice about hiring is don't do it. The most successful companies we've worked with at YC have waited a relatively long time to start hiring employees. Employees are expensive. Employees add organizational complexity and communication overhead. There are things you can say to your cofounders that you cannot say with employees in the room. Employees also add inertia—it gets exponentially harder to change direction with more people on the team. Resist the urge to derive your self-worth from your number of employees.

The best people have a lot of opportunities. They want to join rocketships. If you have nothing, it's hard to hire them. Once you're obviously winning, they'll want to come join you.

It's worth repeating that great people have a lot of options, and you need great people to build a great company. Be generous with equity, trust, and responsibility. Be willing to go after people you don't think you'll be able to get.

Remember that the kind of people you want to hire can start their own companies if they want.

When you are in recruiting mode (i.e., from when you get product-market fit to T-infinity), you should spend about 25% of your time on it. At least one founder, usually the CEO, needs to get great at recruiting. It's most CEOs' number one activity by time. Everyone says that CEOs should spend a lot of their time recruiting, but in practice, none but the best do. There's probably something to that.

Don't compromise on the quality of people you hire. Everyone knows this, and yet everyone compromises on this at some point during a desperate need. Everyone goes on to regret it, and it sometimes almost kills the company. Good and bad people are both infectious, and if you start with mediocre people, the average does not usually trend up. Companies that start off with mediocre early employees almost never recover. Trust your gut on people. If you have doubt, then the answer is no.

Do not hire chronically negative people. They do not fit what an early-stage startup needs—the rest of the world will be predicting your demise every day, and the company needs to be united internally in its belief to the contrary.

Value aptitude over experience for almost all roles. Look for raw intelligence and a track record of getting things done. Look for people you like—you'll be spending a lot of time together and often in tense situations. For people you don't already know, try to work on a project together before they join full-time.

Invest in becoming a good manager. This is hard for most founders, and it's definitely counterintuitive. But it's important to get good at this. Find mentors that can help you here. If you do not get good at this, you will lose employees quickly, and if you don't retain employees, you can be the best recruiter in the world and it still won't matter. Most of the principles on being a good manager

are well-covered, but the one that I never see discussed is "don't go into hero mode". Most first-time managers fall victim to this at some point and try to do everything themselves, and become unavailable to their staff. It usually ends in a meltdown. Resist all temptation to switch into this mode, and be willing to be late on projects to have a well-functioning team.

Speaking of managing, try hard to have everyone in the same office. For some reason, startups always compromise on this. But nearly all of the most successful startups started off all together. I think remote work can work well for larger companies but it has not been a recipe for massive success for startups.

Finally, fire quickly. Everyone knows this in principle and no one does it. But I feel I should say it anyway. Also, fire people who are toxic to the culture no matter how good they are at what they do. Culture is defined by who you hire, fire, and promote.

I wrote a blog post with more detail[2].

[2] *See* Appendix 2

Part IV: Execution – COMPETITORS

Home

A quick word about competitors: competitors are a startup ghost story. First-time founders think they are what kill 99% of startups. But 99% of startups die from suicide, not murder. Worry instead about all of your internal problems. If you fail, it will very likely be because you failed to make a great product and/or failed to make a great company.

99% of the time, you should ignore competitors. Especially ignore them when they raise a lot of money or make a lot of noise in the press. Do not worry about a competitor until they are beating you with a real, shipped product. Press releases are easier to write than code, which is easier still than making a great product. In the words of Henry Ford: "The competitor to be feared is one who never bothers about you at all, but goes on making his own business better all the time."

Every giant company has faced worse competitive threats than what you are facing now when they were small, and they all came out ok. There is always a counter-move.

PART IV: EXECUTION

MAKING MONEY

Part IV: Execution – MAKING MONEY

Home

Oh yes, making money. You need to figure out how to do that.

The short version of this is that you have to get people to pay you more money than it costs you to deliver your good/service. For some reason, people always forget to take into account the part about how much it costs to deliver it[3].

If you have a free product, don't plan to grow by buying users. That's really hard for ad-supported businesses. You need to make something people share with their friends.

If you have a paid product with less than a $1,000 customer lifetime value (LTV), you generally cannot afford sales. Experiment with different user acquisition methods like SEO/SEM, ads, mailings, etc., but try to repay your customer acquisition cost (CAC) in 3 months. If you have a paid product with more than a $1,000 LTV (net to you) you maybe can afford direct sales if your product is easy to sell. But unless your LTV is more like $5,000 or higher, it may not work. Try selling the product yourself first to learn what works. Hacking Sales is a useful book to read.

[3] *See* Appendix 3

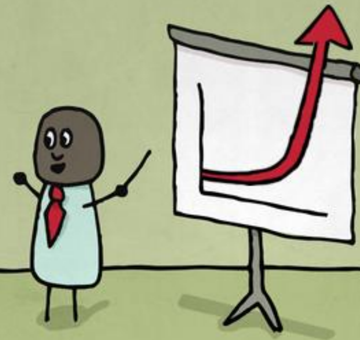
In any case, try to get to "ramen profitability"—i.e., make enough money so that the founders can live on ramen—as quickly as you can. When you get here, you control your own destiny and are no longer at the whims of investors and financial markets.

Watch your cash flow obsessively. Although it sounds unbelievable, we've seen founders run out of money without being aware it was happening a number of times (and read Paul Graham's essay[4]).

[4] *See* Appendix 4

PART IV: EXECUTION

FUNDRAISING



Part IV: Execution – FUNDRAISING

Home

Most startups raise money at some point.

You should raise money when you need it or when it's available on good terms. Be careful not to lose your sense of frugality or to start solving problems by throwing money at them. Not having enough money can be bad, but having too much money is almost always bad.

The secret to successfully raising money is to have a good company. All of the other stuff founders do to try to over-optimize the process probably only matters about 5% of the time. Investors are looking for companies that are going to be really successful whether or not they invest, but that can grow faster with outside capital. The "really successful" part is important—because investors' returns are dominated by the big successes, if an investor believes you have a 100% chance of creating a $10 million company but almost no chance of building a larger company, he/she will still probably not invest even at a very low valuation. Always explain why you could be a huge success.

Investors are driven by the dual fears of missing the next Google, and fear of losing money on something that in retrospect looks obviously stupid. (For the best companies, they fear both at the same time.)

It is a bad idea to try to raise money when your company isn't in good enough shape to attract capital. You will burn reputation and waste time.

Don't get demoralized if you struggle to raise money. Many of the best companies have struggled with this, because the best companies so often look bad at the beginning (and they nearly always look unfashionable.) When investors tell you no, believe the no but not the reason. And remember that anything but "yes" is a "no"—investors have a wonderful ability to say "no" in a way that sounds like "maybe yes".

It's really important to have fundraising conversations in parallel—don't go down a list of your favorite investors sequentially. The way to get investors to act is fear of other investors taking away their opportunity.

View fundraising as a necessary evil and something to get done as quickly as possible. Some founders fall in love with fundraising; this is always bad. It's best to have just one founder do it so the company doesn't grind to a halt.

Remember that most VCs don't know much about most industries. Metrics are always the most convincing.

It's beginning to change, but most investors (Y Combinator being a notable exception) unfortunately still require introductions from people you both know to take you seriously.

Insist on clean terms (complicated terms compound and get worse each round) but don't over-optimize, especially on valuation. Valuation is something quantitative to compete on, and so founders love to compete for the highest valuation. But intermediate valuations don't matter much.

The first check is the hardest to get, so focus your energies on getting that, which usually means focusing your attention on whoever loves you the most. Always have multiple plans, one of which is not raising anything, and be flexible

depending on interest—if you can put more money to good use, and it's available on reasonable terms, be open to taking it.

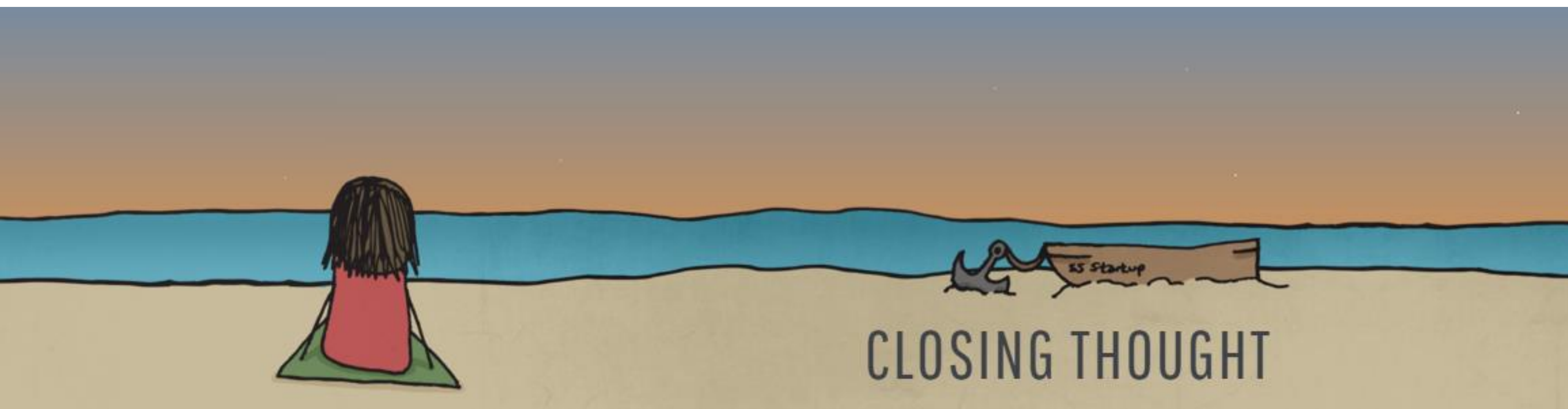
An important key to being good at pitching is to make your story as clear and easy to understand as possible. Of course, the most important key is to actually have a good company. There are lots of thoughts about what to include in a pitch, but at a minimum you need to have: mission, problem, product/service, business model, team, market and market growth rate, and financials.

Remember that the bar for each round of funding is much higher. If you got away with just being a compelling presenter for your seed round, don't be surprised when it doesn't work for your Series A.

Good investors really do add a lot of value. Bad investors detract a lot. Most investors fall in the middle and neither add nor detract. Investors that only invest a small amount usually don't do anything for you (i.e., beware party rounds).

Great board members are one of the best outside forcing functions for a company other than users, and outside forcing functions are worth more than most founders think. Be willing to accept a lower valuation to get a great board member who is willing to be very involved.

I think this essay by Paul Graham is the best thing out there on fundraising. As a tactical point, you will usually need to be a Delaware C Corporation to raise institutional capital, so it's best to just incorporate that way.



CLOSING THOUGHT

Home

Remember that at least a thousand people have every great idea. One of them actually becomes successful. The difference comes down to execution. It's a grind, and everyone wishes there were some other way to transform "idea" into "success", but no one has figured it out yet.

So all you need is a great idea, a great team, a great product, and great execution. So easy! ;)

Thanks to Paul Buchheit, Erica Carpenter, Brian Chesky, Adam D'Angelo, Paul Graham, Drew Houston, Justin Kan, Matt Krisiloff, Aaron Levie, Gabriel Leydon, Jessica Livingston, Dustin Moskovitz, David Rusenko and Colleen Taylor for contributing thoughts to this.

Source: http://playbook.samaltman.com/

APPENDIX

Appendix 1: Projects and Companies

Home

In the early days of my startup, I used to get slightly offended when people would refer to it as a "project". "How's your project going?" seemed like the asker didn't take us seriously, even though everything felt serious to us. I remember assuming this would stop after we announced a $5 million Series A; it didn't. I kept feeling like we'd know we made it when people started referring to us a company.

I now have the opposite belief. It's far better to be thought of—and to think of yourself—as a project than a company for as long as possible.

Companies sound serious. When you start thinking of yourself as a company, you start acting like one. You worry more about pretend work involving things like lawyers, conferences, and finance stuff, and less about building product, because that's what people who run companies are supposed to do. This is, of course, the kiss of death for promising ideas.

Projects have very low expectations, which is great. Projects also usually mean less people and less money, so you get the good parts of both flexibility and focus. Companies have high expectations—and the more money out of the gate and the more press, the worse off they are (think Color and Clinkle, for example).

Worst of all, you won't work on slightly crazy ideas—this is a company, not a hobby, and you need to do something that sounds like a good, respectable idea. There is a limit to what most people are willing to work on for something called a company that does not exist if it's just a project. The risk of seeming stupid when something is just a project is almost zero, and no one cares if you fail. So you're much more likely to work on something good, instead of derivative but plausible-sounding crap.

When you're working on a project, you can experiment with ideas for a long time. When you have a company, the clock is ticking and people expect results. This gets to the danger with projects—a lot of people use them as an excuse to not work very hard. If you don't have the self-discipline to work hard without external pressure, projects can be a license to slack off.

The best companies start out with ideas that don't sound very good. They start out as projects, and in fact sometimes they sound so inconsequential the founders wouldn't let themselves work on them if they had to defend them as a company. Google and Yahoo started as grad students' projects. Facebook was a project Zuckerberg built while he was a sophomore in college. Twitter was a side project that started with a single engineer inside a company doing something totally different. Airbnb was a side project to make some money to afford rent. They all became companies later.

All of these were ideas that seemed bad but turned out to be good, and this is the magic formula for major success. But in the rush to claim a company, they could have been lost. The pressure from external (and internal) expectations is constant and subtle, and it often kills the magic ideas. Great companies often start as projects.

Source: http://blog.samaltman.com/projects-and-companies

Appendix 2: How to hire

Home

After startups raise money, their next biggest problem becomes hiring. It turns out it's both really hard and really important to hire good people; in fact, it's probably the most important thing a founder does.

If you don't hire very well, you will not be successful—companies are a product of the team the founders build. There is no way you can build an important company by yourself. It's easy to delude yourself into thinking that you can manage a mediocre hire into doing good work.

Here is some advice about hiring:

***Spend more time on it.**

The vast majority of founders don't spend nearly enough time hiring. After you figure out your vision and get product-market fit, you should probably be spending between a third and a half of your time hiring. It sounds crazy, and there will always be a ton of other work, but it's the highest-leverage thing you can do, and great companies always, always have great people.

You can't outsource this—you need to be spending time identifying people, getting potential candidates to want to work at your company, and meeting every person that comes to interview. Keith Rabois believes the CEO/founders should interview every candidate until the company is at least 500 employees.

***In the beginning, get your hands dirty.**

Speaking of spending time, you should spend the time to learn a role before you hire for it. If you don't understand it, it's very hard to get the right person. The classic example of this is a hacker-CEO deciding to hire a VP of Sales because he doesn't want to get his hands dirty. This does not work. He needs to do it himself first and learn it in detail. Then after that, he should lean on his board or investors to give him opinions on his final few candidates.

***Look for smart, effective people.**

There are always specifics of what you need in a particular role, but smart and effective have got to be table stakes. It's amazing how often people are willing to forgo these requirements; predictably, those hires don't work out in the early days of a startup (they may never work).

Fortunately, these are easy to look for.

Talk to the candidates about what they've done. Ask them about their most impressive projects and biggest wins. Specifically, ask them about how they spend their time during an average day, and what they got done in the last month. Go deep in a specific area and ask about what the candidate actually did—it's easy to take credit for a

successful project. Ask them how they would solve a problem you are having related to the role they are interviewing for.

That, combined with the right questions when you check references, will usually give you a good feel about effectiveness. And usually you can gauge intelligence by the end of an hour-long conversation. If you don't learn anything in the interview, that's bad. If you are bored in the interview, that's really bad. A good interview should feel like a conversation, not questions and responses.

Remember that in a startup, anyone you hire is likely to be doing a new job in three to six months. Smart and effective people are adaptable.

***Have people audition for roles instead of interviewing for them.**

This is the most important tactical piece of advice I have. It is difficult to know what it's like working with someone after a few interviews; it is quite easy to know what it's like after working with them

Whenever possible (and it's almost always possible), have someone do a day or two of work with you before you hire her; you can do this at night or on the weekends. If you're interviewing a developer, have her write code for a real but non-critical project. For a PR person, have her write a press release and identify reporters to pitch it to. Just have the person sign a contractor agreement and pay them for this work like a normal contractor.

You'll get a much, much better sense of what it's like working with this person and how good she is at the role than you can ever get in just an interview. And she'll get a feel for what working at your company is like.

***Focus on the right ways to source candidates.**

Basically, this boils down to "use your personal networks more". By at least a 10x margin, the best candidate sources I've ever seen are friends and friends of friends. Even if you don't think you can get these people, go after the best ones relentlessly. If it works out 5% of the time, it's still well worth it.

All the best startups I know manage to hire like this for much longer than one would think possible. Most bad startups make excuses for not doing this.

When you hire someone, as soon as you're sure she's a star you should sit her down and wring out of her the names of everyone that you should try to hire. You may have to work pretty hard at this.

Often, to get great people, you have to poach. They're never looking for jobs, so don't limit your recruiting to people that are looking for jobs. A difficult question is what you should do about poaching from acquaintances—I don't have a great answer for this. A friend says, "Poaching is the titty twister of Silicon Valley relationships".

Technical recruiters are pretty bad. The job boards are generally worse. Conferences can be good. Hosting interesting tech talks can be good for technical hiring. University recruiting works well once you're reasonably established.

Don't limit your search to candidates in your area. This is especially true if you're in the bay area; lots of people want to move here.

View candidate sourcing as a long-term investment—you may spend time now with someone that you don't even talk to about a job for a year or more.

Use you investors and their networks to find candidates. In your investor update emails, let them know what kind of people you need to hire.

As a side note, if I were going to jump into the mosh pit of people starting recruiting startups, I would try to make it look as much like personal network hiring as possible, since that's what seems to work. I'd love a service that would let me see how everyone in my company was connected to a candidate, and be able to search personal networks of people in the company (LinkedIn is probably good at this for hiring sales people but not very good at this for developers).

***Have a mission, and don't be surprised at how much selling you'll have to do.**

You need a mission in order to hire well. In addition to wanting to work with a great team, candidates need to believe in your mission—i.e., why is this job more important

than any of the others they could take? Having a mission that gets people excited is probably the best thing you can do to get a great team on board before you have runaway traction.

As a founder, you'll assume that everyone will be as excited about your company as you are. In reality, no one will. You need to spend a lot of time getting candidates excited about your mission.

If you have a good mission and you're good at selling it, you'll be able to get slightly overqualified people—although, in a fast-growing startup, they'll end up in a role that they feel not quite ready for quickly anyway.

You should use your board and your investors to help you close candidates.

Once you decide you want someone, switch into closing mode. The person that the new hire will report to (and ideally also the CEO) should be doing everything possible to close the candidate, and talking to her about once a day.

***Hire people you like.**

At Stripe, I believe they call this the Sunday test—would you be likely to come into the office on a Sunday because you want to hang out with this person? Liking the people you work with is pretty important to the right kind of company culture. Only a few times have I ever seen a scenario where I didn't like an otherwise very good candidate. I only made the hire once, and it was a mistake.

That being said, remember you want at least some diversity of thought. There are some attributes where you want uniformity—integrity, intelligence, etc.—and there are some where you want coverage of the entire range.

***Have a set of cultural values you hire for.**

Spend a lot of time figuring out what you want your cultural values to be (there are some good examples on the Internet). Make sure the whole company knows what they are and buys into them. Anyone you hire should be a cultural fit.

Andrew Mason says “Values are a decision making framework that empower individuals to make the decision that you, the founder, would make, in situations where there are conflicting interests (e.g. growth vs. customer satisfaction)”.

Treat your values as articles of faith. Screen candidates for these values and be willing to let an otherwise good candidate go if he is not a cultural fit. Diversity of opinions and certain characteristics (e.g. you want nerds and athletes both on the team) is good; diversity of values in a startup is bad.

There are some people that are so set in their ways they will never get behind your values; you will probably end up firing them.

As a side note, avoid remote employees in the early days. As a culture is still gelling, it’s important to have everyone in the same building.

***Don’t compromise.**

In the grind of a startup, you’ll always need someone yesterday and it’s easy to hire someone that is not quite smart enough or a good enough culture fit because you really need a specific job done. Especially in the early days, never compromise. A single bad hire left unfixed for long can kill a company. It’s better to lose a deal or be late on a product or whatever than to hire someone mediocre.

Great people attract other great people; as soon as you get a mediocre person in the building, this entire phenomenon can unwind.

***Be generous with compensation packages, but mostly with equity.**

You should be very frugal with nearly everything in a startup. Compensation for great people is an exception.

Where you want to be generous, though, is with equity. Ideally, you end up paying people slightly below to roughly market salaries but with a very generous equity package. “Experienced” people often have higher personal burn rates and sometimes you’ll need to pay them more, but remember that great companies are not usually

created by experienced people (with the exception of a few roles where it really matters a lot.)

I am sure I will get flamed for saying this, but it's the right strategy—if you want an above-market salary, go work at a big company with no equity upside.

Ideally, you want to pay people just enough they don't stress about cash flow. Equity is harder, but a good rule for the first 20 hires seems to be about double what your investors suggest. For a company on a good but not absolute breakout trajectory, some rough numbers I've seen are about 1.5% for the first engineer and about 0.25% for the twentieth. But the variance is huge.

Incidentally, a very successful YC company has a flat salary for effectively all of their engineers, and it seems to work well. It's lower than what these people could get elsewhere, but clearly they enjoy the work and believe the stock is going to be worth a lot. The sorts of people that will take this deal are the kind of people you want in a startup. And unless something goes really wrong, at this point, these engineers are going to make way more money than that would have taking higher salaries elsewhere—not to mention how much better their work environment has been.

You will likely have to negotiate a little bit. Learn how to do this. In general, materially breaking your compensation structure to get someone is a bad idea—word gets out and everyone will be upset.

***Watch out for red flags and trust your gut.**

There are a few things in the interviewing/negotiating process that you should watch out for because they usually mean that person will not be successful in a startup. A focus on title is an example; a focus on things like "how many reports will be in my organization" is an even worse example. You'll develop a feel for these sorts of issues very quickly; don't brush them off.

If you have a difficult-to-articulate desire to pass, pass.

***Always be recruiting.**

Unfortunately, recruiting usually doesn't work as a transactional activity. You have to view it as something you always do, not something you start when you need to fill a role immediately. There's a fair amount of unpredictability in the process; if you find someone great for a role you won't need for two months, you should still hire her now.

***Fire fast.**

I have never met a newbie founder that fires fast enough; I have also never met a founder who doesn't learn this lesson after a few years.

You will not get 100% of your hires right. When it's obviously not working, it's unlikely to start working. It's better for everyone involved to part ways quickly, instead of hanging on to unrealistic dreams that it's going to get better. This is especially true for the person you have to let go—if they're just at your company for a couple of months, it's a non-issue in future interviews. And everyone else at the company is probably aware that the person is not working out before you are.

Having to fire people is one of the worst things a founder has to do, but you have to just get it over with and trust that it will work out better than dragging things out.

***Put a little bit of rigor around the hiring process.**

Make everyone on your team commit to a hire/no hire decision for everyone they meet, and write up their thoughts. If you get it wrong, this is useful to look back at later. It's good to have a brief in-person discussion with the entire interviewing team after a candidate leaves.

Have someone take the candidate out to lunch or dinner. Insist that everyone is on time and prepared for interviews/auditions. Make sure every candidate leaves with a positive impression of your company.

Be organized—one person should coordinate the entire interview process, make sure every topic you want to cover gets covered, convene people for the discussion after all interviews are done, etc. Also, have a consistent framework for how you decide whether or not to hire—do you need unanimous consent?

Remember that despite being great at what they do your team may not be great interviewers. It's important to teach people how to interview.

***Don't hire.**

Many founders hire just because it seems like a cool thing to do, and people always ask how many employees you have. Companies generally work better when they are smaller. It's always worth spending time to think about the least amount of projects/work you can feasibly do, and then having as small a team as possible to do it.

Don't hire for the sake of hiring. Hire because there is no other way to do what you want to do.

Good luck. Hiring is very hard but very important work. And don't forget that after you hire people, you need to keep them. Remember to check in with people, be a good manager, have regular all hands meetings, make sure people are happy and challenged, etc. Always keep a sense of momentum at your company—that's important to retaining talent. Give people new roles every six months or so. And of course, continue to focus on bringing talented people into the company—that alone will make other good people want to stay.

Always be identifying and promoting new talent. This is not as sexy as thinking about new problems to solve, but it will make you successful.

Thanks to Patrick Collison, Andrew Mason, Keith Rabois, Geoff Ralston, and Nick Sivo for reading drafts of this.

Source: http://blog.samaltman.com/how-to-hire

Appendix 3: Unit Economics

[Home]

Commentators are looking hard for what's wrong with startups in Silicon Valley. First they talked about valuations being too high. Then they talked about valuations not

really meaning anything. Then they talked about companies staying private too long. Then they talked about burn rates.

But something does feel off, though it's been hard to precisely identify.

I think the answer is unit economics. One of the jokes that came out of the 2000 bubble was "we lose a little money on every customer, but we make it up on volume". This was then out of fashion for a long time as Google and Facebook hit their stride.

There are now more businesses than I ever remember before that struggle to explain how their unit economics are ever going to make sense. It usually requires an explanation on the order of infinite retention ("yes, our sales and marketing costs are really high and our annual profit margins per user are thin, but we're going to keep the customer forever"), a massive reduction in costs ("we're going to replace all our human labor with robots"), a claim that eventually the company can stop buying users ("we acquire users for more than they're worth for now just to get the flywheel spinning"), or something even less plausible.

This is particularly common in startups that don't pass the Peter Thiel monopoly test—these startups seem to have to spend every available dollar on user acquisition, and if they raise prices, customers defect to a similar service.

Most great companies historically have had good unit economics soon after they began monetizing, even if the company as a whole lost money for a long period of time.

Silicon Valley has always been willing to invest in money-losing companies that may eventually make lots of money. That's great. I have never seen Silicon Valley so willing to invest in companies that have well-understood financials showing they will probably always lose money. Low-margin businesses have never been more fashionable here than they are right now.

Companies that have raised lots of money are at particular risk. It's so tempting to paper over a problem with the business by spending more money instead of fixing the product or service.

Burn rates by themselves are not scary. Burn rates are scary when you scale the business up and the model doesn't look any better. Burn rates are also scary when runway is short (i.e., burning $2M a month with $100M in the bank is fine; burning $1M a month with $3M in the bank is really bad) even if the unit economics look great.

The good news is that if you're aware of this you can avoid the trap. If there's no other way to operate in your space, maybe it's a bad business. The low-margin, hyper-competitive world is not the only place to be. Companies always have an explanation about how they're going to fix unit economics, so you really have to go out of your way not to delude yourself.

If you hold yourself to the standard of making a product that is so good people spontaneously recommend it to their friends, and you have an easy-to-understand business model where you make more than you spend on each user, and it gets better not worse as you get bigger, you may not look like some of hottest companies of today, but you'll look a lot like Google and Facebook.

Source: http://blog.samaltman.com/unit-economics

Appendix 4: Default Alive or Default Dead?

Home

Paul Graham

October 2015

When I talk to a startup that's been operating for more than 8 or 9 months, the first thing I want to know is almost always the same. Assuming their expenses remain constant and their revenue growth is what it's been over the last several months, do they make it to profitability on the money they have left? Or to put it more dramatically, by default do they live or die?

The startling thing is how often the founders themselves don't know. Half the founders I talk to don't know whether they're default alive or default dead.

If you're among that number, Trevor Blackwell has made a handy calculator you can use to find out.

The reason I want to know first whether a startup is default alive or default dead is that

the rest of the conversation depends on the answer. If the company is default alive, we can talk about ambitious new things they could do. If it's default dead, we probably need to talk about how to save it. We know the current trajectory ends badly. How can they get off that trajectory?

Why do so few founders know whether they're default alive or default dead? Mainly, I think, because they're not used to asking that. It's not a question that makes sense to ask early on, any more than it makes sense to ask a 3 year old how he plans to support himself. But as the company grows older the question switches from meaningless to critical. That kind of switch often takes people by surprise.

I propose the following solution: instead of starting to ask too late whether you're default alive or default dead, start asking too early. It's hard to say precisely when the question switches polarity. But it's probably not that dangerous to start worrying too early that you're default dead, whereas it's very dangerous to start worrying too late.

The reason is a phenomenon I wrote about earlier: the fatal pinch. The fatal pinch is default dead + slow growth + not enough time to fix it. And the way founders end up in it is by not realizing that's where they're headed.

There is another reason founders don't ask themselves whether they're default alive or default dead: they assume it will be easy to raise more money. But that assumption is often false, and worse still, the more you depend on it, the falser it becomes.

Maybe it will help to separate facts from hopes. Instead of thinking of the future with vague optimism, explicitly separate the components. Say "We're default dead, but we're counting on investors to save us." Maybe as you say that it will set off the same alarms in your head that it does in mine. And if you set off the alarms sufficiently early, you may be able to avoid the fatal pinch.

It would be safe to be default dead if you could count on investors saving you. As a rule their interest is a function of growth. If you have steep revenue growth, say over 6x a year, you can start to count on investors being interested even if you're not profitable. [1] But investors are so fickle that you can never do more than start to count on it. Sometimes something about your business will spook investors even if your growth is great. So no matter how good your growth is, you can never safely treat fundraising as more than a plan A. You should always have a plan B as well: you should know (as in write down) precisely what you'll need to do to survive if you can't raise more money, and precisely when you'll have to switch to plan B if plan A isn't working.

In any case, growing fast versus operating cheaply is far from the sharp dichotomy many founders assume it to be. In practice there is surprisingly little connection between how much a startup spends and how fast it grows. When a startup grows fast it's usually because the product hits a nerve, in the sense of hitting some big need straight on. When a startup spends a lot it's usually because the product is expensive to develop or sell, or simply because they're wasteful.

If you're paying attention, you'll be asking at this point not just how to avoid the fatal

pinch, but how to avoid being default dead. That one is easy: don't hire too fast. Hiring too fast is by far the biggest killer of startups that raise money. [2]

Founders tell themselves they need to hire in order to grow. But most err on the side of overestimating this need rather than underestimating it. Why? Partly because there's so much work to be done. Naive founders think that if they can just hire enough people it somehow will be. Partly because successful startups have lots of employees, so it seems like that's what one does in order to be successful. In fact the large staffs of successful startups are probably more the effect of growth than the cause. And partly because when founders have slow growth they don't want to face what is usually the real reason: the product is not appealing enough.

Plus founders who've just raised money are often encouraged to overhire by the VCs who funded them. Kill-or-cure strategies are optimal for VCs because they're protected by the portfolio effect. VCs want to blow you up, in one sense of the phrase or the other. But as a founder your incentives are different. You want above all to survive. [3]

Here's a common way startups die. They make something moderately appealing and have decent initial growth. They raise their first round fairly easily because the founders seem smart and the idea sounds plausible. But because the product is only moderately appealing, growth is ok but not great. The founders convince themselves that hiring a bunch of people is the way to boost growth. Their investors agree. But (because the product is only moderately appealing) the growth never comes. Now they're rapidly running out of runway. They hope further investment will save them. But because they have high expenses and slow growth, they're now unappealing to investors. They're unable to raise more, and the company dies.

What the company should have done is address the fundamental problem: that the product is only moderately appealing. Hiring people is rarely the way to fix that. More often than not it makes it harder. At this early stage, the product needs to evolve more than to be "built out," and that's usually easier with fewer people. [4]

Asking whether you're default alive or default dead may save you from this. Maybe the alarm bells it sets off will counteract the forces that push you to overhire. Instead you'll be compelled to seek growth in other ways. For example, by doing things that don't scale, or by redesigning the product in the way only founders can. And for many if not most startups, these paths to growth will be the ones that actually work.

Airbnb waited 4 months after raising money at the end of Y Combinator before they hired their first employee. In the meantime the founders were terribly overworked. But they were overworked evolving Airbnb into the astonishingly successful organism it is now.

Notes

[1] Steep usage growth will also interest investors. Revenue will ultimately be a constant multiple of usage, so x% usage growth predicts x% revenue growth. But in practice investors discount merely predicted revenue, so if you're measuring usage you need a higher growth rate to impress investors.

[2] Startups that don't raise money are saved from hiring too fast because they can't afford to. But that doesn't mean you should avoid raising money in order to avoid this problem, any more than that total abstinence is the only way to avoid becoming an alcoholic.

[3] I would not be surprised if VCs' tendency to push founders to overhire is not even in their own interest. They don't know how many of the companies that get killed by overspending might have done well if they'd survived. My guess is a significant number.

[4] After reading a draft, Sam Altman wrote:

"I think you should make the hiring point more strongly. I think it's roughly correct to say that YC's most successful companies have never been the fastest to hire, and one of the marks of a great founder is being able to resist this urge."

Paul Buchheit adds:

"A related problem that I see a lot is premature scaling—founders take a small business that isn't really working (bad unit economics, typically) and then scale it up because they want impressive growth numbers. This is similar to over-hiring in that it makes the business much harder to fix once it's big, plus they are bleeding cash really fast."

Thanks to Sam Altman, Paul Buchheit, Joe Gebbia, Jessica Livingston, and Geoff Ralston for reading drafts of this.

Source: http://paulgraham.com/aord.html

For more

Growth Hacking stories

Click

collected by
Truong Nguyen