

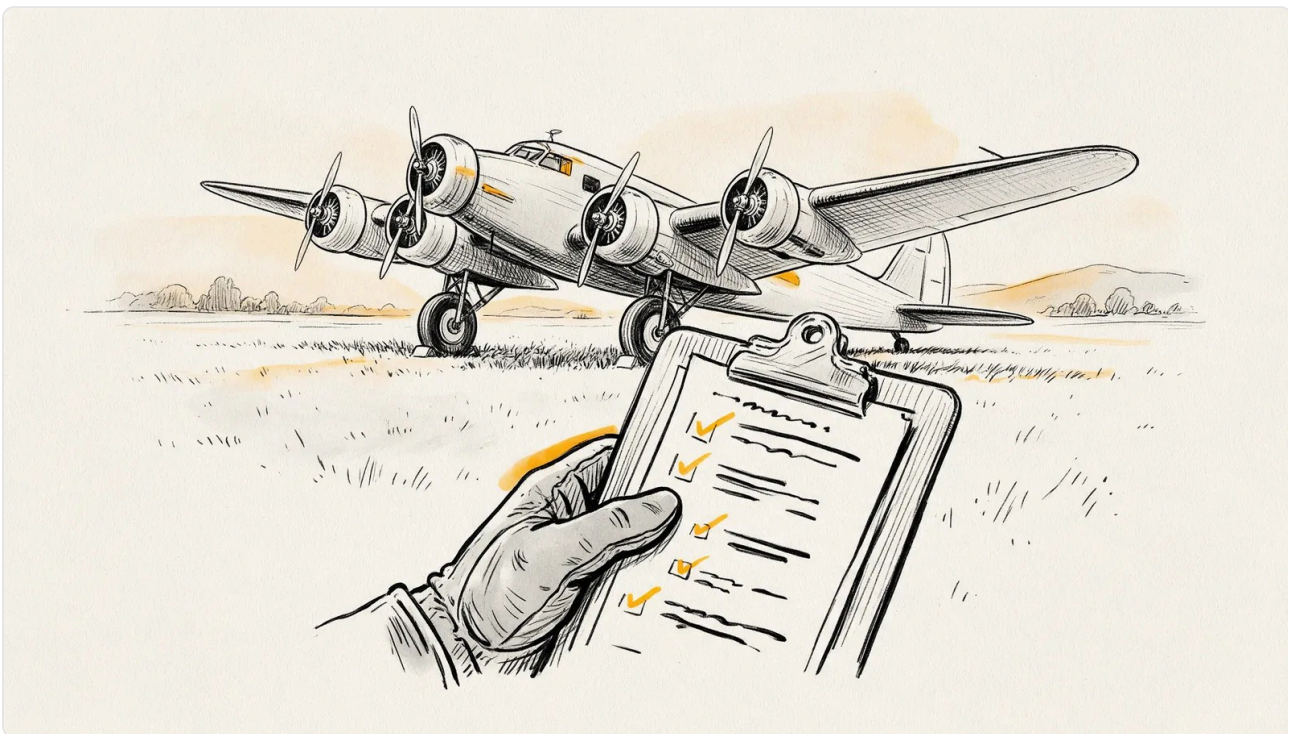
Six Questions Before the First Line of Code

Before the first line of code, the first hire and the first share of equity, six questions need an answer - who pays, for what, how much, and whether the maths fits the money you plan to raise. One page, with a test for each.

pitchera.ai · 1 October 2026 · 13 min read · pitchera.ai/blog/six-questions-before-the-first-line-of-code

In October 1935 Boeing flew its new four-engine bomber, the Model 299, for the US Army at Wright Field in Ohio. It climbed steeply, stalled and crashed. The pilot was the Army's chief of flight testing. Nothing was wrong with the aircraft: the gust lock, which holds the control surfaces still while the plane is parked, had not been released.

A newspaper called it too much airplane for one man to fly. The test pilots took a different view. The most experienced pilot they had was flying it, and he had missed one step. So they wrote the steps down on a card, to be read before every take-off. Pilots still use one today, on routes they have flown a thousand times.



A few lines on a card, read every time, even by the best pilot on the field.

Startups have no card. Most founding teams start with the part they know how to do, which is usually building. Six months later there is a prototype, a small team, a cap table with four names on it and the first angel money in the bank. There is also a question nobody in the company can answer in one line: who pays, how much, and why now?

They get there in the end. Most of them get there in an investor meeting, when the round depends on the answer.

So here is the card. Six questions, in this order, before the first line of code, the first hire and the first share of equity.

Why those three moments?

Because they are the three early decisions that are hard to take back.

Code is the easiest of the three to undo, and it is still weeks of work. A product built for "SMEs" has a sign-up flow, a pricing page and a feature list designed for nobody in particular. When you find the one buyer who cares, a good part of it gets rebuilt.

A first hire is harder. You hire for the company you think you are. If you think you sell to enterprises, you hire someone who sells to enterprises. If it turns out you sell to small clinics that pay by card, you now have a very good person in the wrong job and a difficult conversation in your diary.

Equity is the hardest. Shares split between co-founders in the first week are split on enthusiasm, before anyone knows what the company is or who will do what. Vesting softens the damage. Knowing what you are building before you split avoids most of it.

Why builders skip it

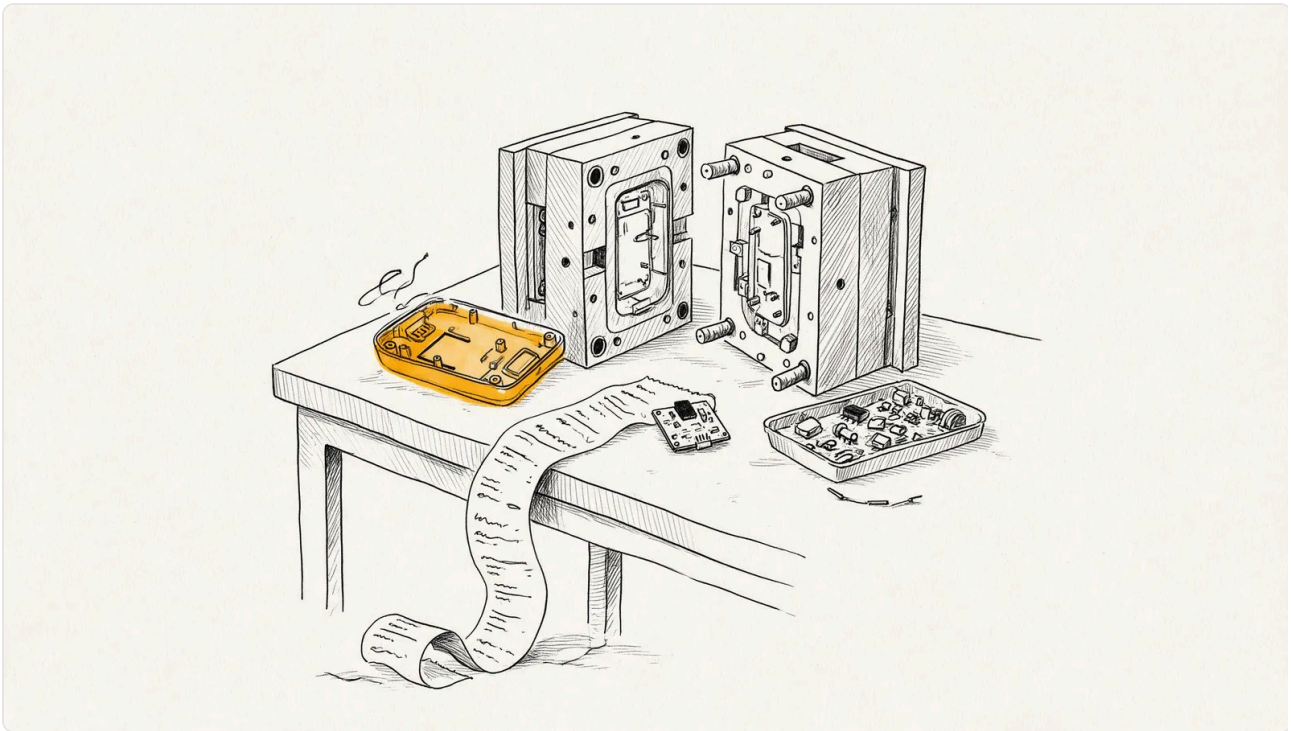
Because building is the fun part, and it feels like progress. Most technical founders, and we count ourselves among them, fall for the technology first and meet the customer later. A working demo gets applause at a meetup. A customer call more often ends with "interesting, send me something". It is easy to guess which of the two gets done more often.



The prototype does answer a question, just not the one that matters early. It proves you can build the thing, which customers and investors mostly assumed anyway. Whether anyone will pay for it is a question no amount of engineering answers from inside the workshop.

Hardware makes all of this more expensive. In software, a wrong guess about the customer costs a few sprints. In hardware it ends up inside things you have already paid for: the sensor you picked, the enclosure, the bill of materials, the mould, the certification you started for a market you may never sell in. A different customer can mean a different size, price point or power budget, and each of those sends you back to the bench.

Hardware teams are also usually teams of engineers, so the pull towards the workshop is stronger, and the first prototype tends to arrive well before the first serious sales conversation.



In software you rewrite the code. In hardware you pay for the mould again.

So build, but spend a few of the first weeks on the six questions, while changing your mind still costs a phone call rather than a new mould.

1. Who is your customer?

Not a sector. One person you could phone tomorrow, with a job title and a budget they control.

A segment belongs on a market slide. Almost everything you do next depends on a person instead: what the product does first, where the next ten customers come from, what they will pay, who signs the invoice. "Restaurants" tells you none of that. "The operations director of a group with ten to thirty sites, who reports food cost to the owners every month" tells you most of it.

If two people use the product and a third one pays, name the third. The users matter, but they do not sign.

THE TEST

Can you name five of them you have actually talked to? People, not companies.

2. What problem do you solve?

The one they already pay for today, in money, hours or risk.

The fastest way in is to ask what they do about it now. Every real problem already has a workaround: a spreadsheet, an intern, a supplier, a weekly meeting, or a decision to live with it. The workaround is your real competitor, and what it costs them sets the ceiling on what you can charge.

If the answer is "nothing, they just put up with it", slow down. Occasionally that is an opening. More often the problem does not hurt enough to spend money on.

THE TEST

What do they do about it now, and what does that cost them?

3. Why you?

Better, faster, cheaper or safer than the workaround from question two, and by how much.

The "how much" gets skipped. Ten per cent better rarely makes anyone switch, because switching has its own price: training, migration, a contract to cancel, a manager to convince. People move when the gain is big enough to explain to their boss in one sentence.

Customers and investors both assume you can build it. What they want to hear is what the customer gets that the workaround cannot give them.

THE TEST

Finish the sentence "Unlike ..., we ..." without using the word "AI".

4. What will they pay?

Most early prices come out of a spreadsheet, and nobody who would pay them has heard them. Take a competitor's price, knock a bit off, multiply by the market. The result looks sensible and has never been said out loud.

So say it out loud, early, before the product is finished, to someone who matches question one, and watch their face. A wince means the budget is tight. A shrug is worse: they were never going to buy. "Is that all?" means you guessed low. Along the way you find out whether a budget exists, whose budget it is and what they compare you with, which is often not what you expected.



THE TEST

Have you said a number out loud to a customer and watched their face?

5. How does the money move?

Who pays, how often, through whom. Subscription, per seat, per transaction, per outcome, licence, hardware plus service. Pick one to start with.

It sounds like a detail. The same product at the same price can be a good business or a poor one depending on how the money moves. Paid a year upfront, it funds your growth. Paid monthly by card, it loses a few per cent a month to failed payments and easy cancellations. Sold through a distributor, a third of it stays with the distributor and you never meet the person using it. Sold to a hospital, it waits for a procurement cycle nobody put in the plan.

Questions one to four decide this one. Before you know who signs and what they will pay, you are not choosing a model so much as picking one that sounds familiar.

THE TEST

Can you explain it in one sentence to someone outside tech?

6. Does the maths work, and for whom?

Two questions in one. How much can this earn, and by when? And does that path fit the money you plan to raise?

The second half is the one nobody explains on day one, so here is the short version. A venture fund has a fixed life, usually about ten years. Most of its investments will return little or nothing, so the few that work have to make up for all the others. That is why funds ask of every deal whether it could, on its own, return the whole fund.

Rough numbers. A 50 million euro fund that still owns ten per cent of you at exit gets its whole fund back only if you sell for 500 million, and the fund's own investors expect about three times their money, not just their money back.

A company turning over 20 million a year might sell for somewhere between 60 and 150 million, depending on margins and growth. That is a very good outcome for the founders and a rounding error for that fund.



A fund is planted for the one tree. Everything else in the field is the cost of finding it.

So the honest answer has two branches. If your plan can plausibly reach the outcome a fund needs, raise from funds. If it builds a good business, build it, and finance it with customers' money, angels, grants, revenue-based financing or a bank, whichever fits.

Both are good outcomes. Pretending one is the other is where it goes wrong: you take money that needs a 500 million exit for a company that will never be one, and spend the next five years on someone else's clock.

THE TEST

Write down the exit your target investor needs and the one your own plan reaches. Are they in the same league?

Why the order matters

Questions one to four are inputs. Five and six are what you work out from them.

Most founders go the other way. They start with a business model they like (subscription, because investors like recurring revenue), a market size from an industry report and a financial model with a price in it. Then they go looking for customers who fit the model. The spreadsheet has answers to questions one to four, but they were typed in, not found out.

The price does not match anything a customer has said. The customer is a segment. The model assumes a sales cycle nobody has tested.

In the full deck reviews we ran this year, four reports in five flagged the same gap: nothing on what one customer costs to win or what that customer is worth. Most of those companies had already built a product and hired a team. The last questions on this list were still open.

A worked example

Here is a made-up but typical case. Call the team Fenwick. They build a smart scale for commercial kitchen bins: it weighs what gets thrown away and tells the kitchen which dishes waste the most. Two founders, an engineer and a former chef. They started where engineers start, with the hardware, and had a working prototype in five months.



Fenwick, roughly. The scale was the easy part.

Their first answers were the usual ones:

- **Customer:** restaurants.
- **Problem:** food waste.
- **Why us:** AI-powered insights.
- **Price:** 99 euros a month, because a competitor charged 120.
- **Model:** SaaS.
- **Market:** every restaurant in Europe.

Then they went through the six questions properly. It took about six weeks, mostly on the phone.

The customer turned out not to be "restaurants" but the operations director of a group with ten or more sites, who already reported food cost to the owners every month and was judged on it. Independent restaurants loved the demo and did not buy. The problem was not waste in general but the gap between the food cost the group budgeted and the one it actually hit. Their workaround was a monthly stock count and a lot of educated guessing.

The "why us" became: unlike a monthly stock count, the scale shows which dish, on which shift, at which site, every day. The word AI left the deck.

Most operations directors winced at 99 euros per site, then said the price mattered less if the scale paid for itself within a quarter, so later pricing conversations were about savings rather than software. Two groups still said no on price.

The money moved as a one-off hardware fee plus about 100 euros a site a month, billed once a year to the group, not to each kitchen.

Then question six. There were about 2,000 groups of that size in their first three markets, averaging fifteen sites each. Even with a third of them, Fenwick was a business turning over 10 to 15 million a year: a good company, and nowhere near what a seed fund needs.

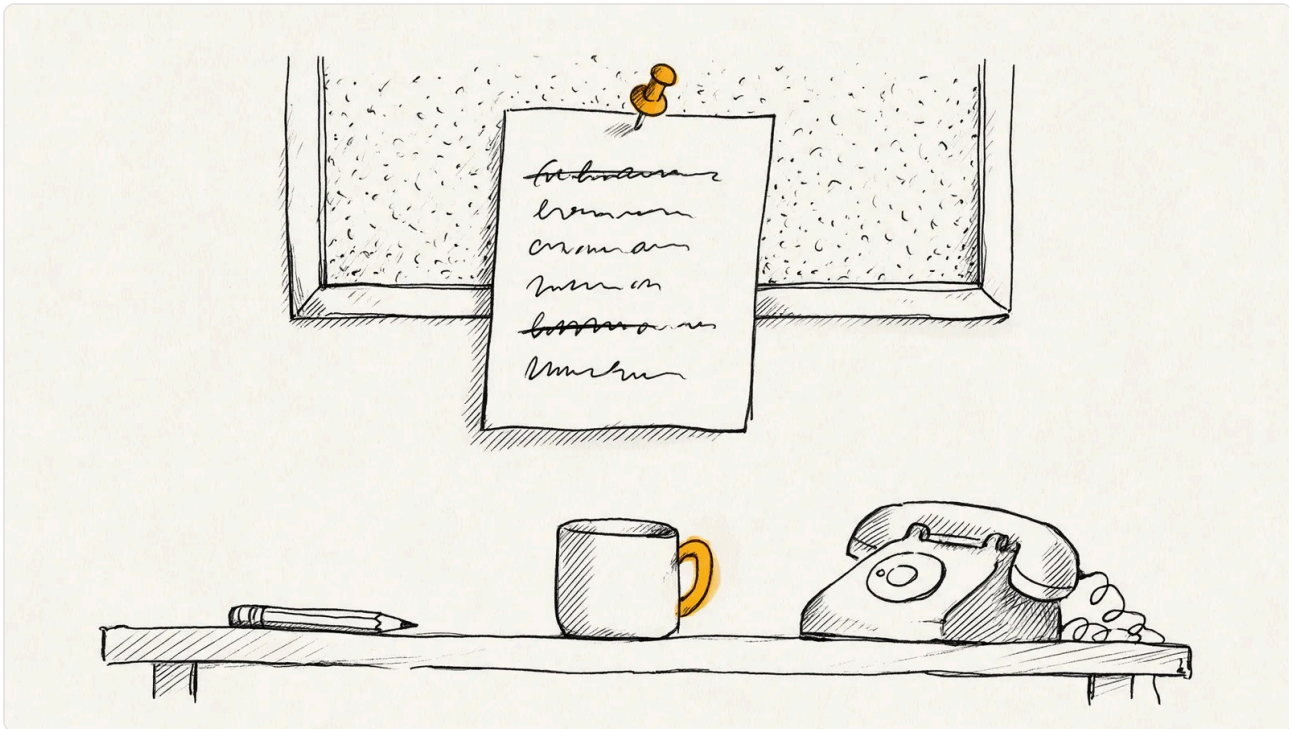
They had been preparing a pitch for funds. They raised a smaller round from angels who knew hospitality instead, and agreed payment terms with their manufacturer so customers paid for the hardware before the invoice came due.

The prototype barely changed. The next hire did: an account manager who had sold to restaurant groups, not a second hardware engineer. So did the next feature, a monthly report for the owners, which only came up because the operations directors kept asking for it. And so did the people they raised from.

The one-page version

Six lines, one sentence each. If you cannot finish a line, that is the one to work on this week.

1. Our customer is ... (job title, company size, who signs).
2. Today they deal with it by ..., which costs them ...
3. Unlike ..., we ...
4. They pay ... per ..., and we have said that number to ... of them.
5. ... pays us ..., every ..., through ...
6. That earns ... a year by ..., so we will raise from ... / build with ...



Pin it somewhere you see it. Rewrite a line every time a customer says something that breaks it. The first version will be wrong in at least three places, and that is fine. The point is to find the wrong ones before anything expensive depends on them.

Where this comes from

The "four in five" figure comes from 40 full readiness reports analysed between April and September 2026, one per company, with our own test uploads removed; unit economics was the gap those reports flagged most often. The longer story behind that number is in [Does This Make Money on Every Customer?](#). Fenwick is invented, and its numbers are illustrative. The fund arithmetic is deliberately rough: funds differ in size, ownership and how many winners they need, but the shape holds. The Model 299 crash is well documented; its place as the origin of the pilot's checklist is the version popularised by Atul Gawande in *The Checklist Manifesto*. Boeing kept the design, and it became the B-17.

Next

Next in the series: how to run the five customer conversations from question one, and what to ask in them.

If you already have a deck, the [free deck review](#) reads it the way an investor would and returns the three questions you are most likely to be asked. If one of them lands on a line you could not finish above, start there. No payment, no call.



Which three questions would your deck draw? The free deck review runs the same analysis on your own deck and returns the three an investor is most likely to ask. No payment, no sales call.

Scan the code, or go to pitchera.ai/deck-review