How Conversation-First SaaS Avoids the Biggest Product-Building Mistakes

Practical Steps to Validate Your SaaS Idea Before *and* After You Build

By: Faridah Giwa-Bello

TL;DR

Building SaaS without talking to users first is costly. Whether you built first or second, the smartest move is to **pause, validate, and listen**. Early conversations reveal real pain points, reduce wasted features, and set your product on a path to growth. Small iterations based on user feedback beat assumptions every time.

Building a SaaS product without talking to users first is like designing a three-story building without ever visiting the site. Only later do you realize the ground can't support it. This is the real cost of skipping SaaS validation.

Many SaaS founders jump straight to code and features, assuming their perception of the problem is enough.

As [one startup analysis](#) notes, founders '*identify problems in the market, think up potential solutions and jump right into building a product*'. They often skip customer validation because '*building a product is heads-down, tangible work*' they can control, while talking to users is not.

But months later, beta tests return negative reviews, or launch results in flat metrics. That's when they discover their product doesn't actually meet real user needs. The same mistake behind [42% of startup failures](#).

You don't need a study to see it, just scroll through any founder community. Every week, there's someone rebuilding their product after realizing they never talked to users in the first place.
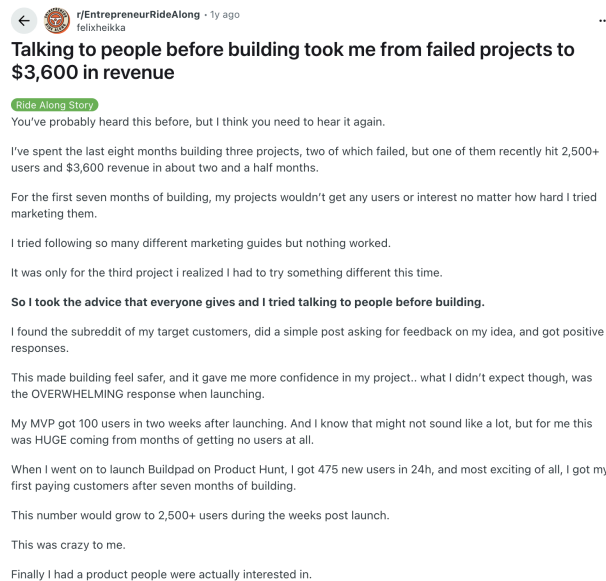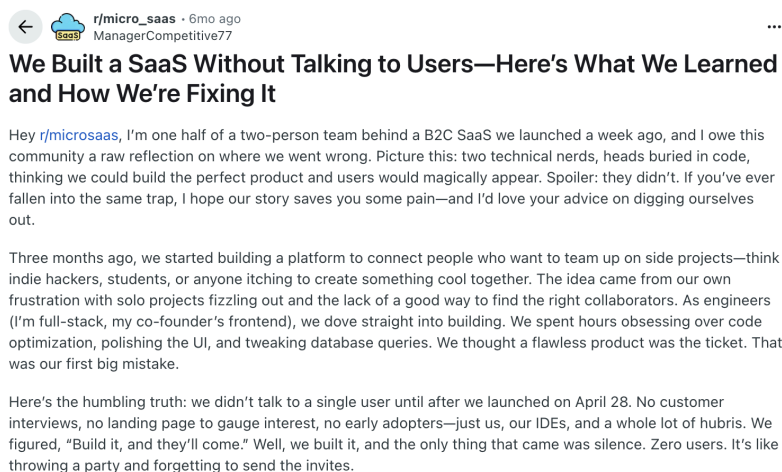
View more of my work [here](#).

Here's what real founders say on public forums when they reflect on coding-first mistakes:

- "*I built my first SaaS for 2 months. Zero customers. … It was only for the third project that I realised I had to try something different this time. So I … tried talking to people before building.*" ([Reddit r/EntrepreneurRideAlong](#), December 2024)

r/EntrepreneurRideAlong · 1y ago
felixheikka

Talking to people before building took me from failed projects to $3,600 in revenue

`Ride Along Story`

You've probably heard this before, but I think you need to hear it again.

I've spent the last eight months building three projects, two of which failed, but one of them recently hit 2,500+ users and $3,600 revenue in about two and a half months.

For the first seven months of building, my projects wouldn't get any users or interest no matter how hard I tried marketing them.

I tried following so many different marketing guides but nothing worked.

It was only for the third project i realized I had to try something different this time.

So I took the advice that everyone gives and I tried talking to people before building.

I found the subreddit of my target customers, did a simple post asking for feedback on my idea, and got positive responses.

This made building feel safer, and it gave me more confidence in my project.. what I didn't expect though, was the OVERWHELMING response when launching.

My MVP got 100 users in two weeks after launching. And I know that might not sound like a lot, but for me this was HUGE coming from months of getting no users at all.

When I went on to launch Buildpad on Product Hunt, I got 475 new users in 24h, and most exciting of all, I got my first paying customers after seven months of building.

This number would grow to 2,500+ users during the weeks post launch.

This was crazy to me.

Finally I had a product people were actually interested in.

- "*We built a SaaS without talking to users — here's what we learned: heads buried in code, thinking users would show up. Spoiler: they didn't.*" ([Reddit r/micro_saas](#), May 2025)

r/micro_saas · 6mo ago
ManagerCompetitive77

We Built a SaaS Without Talking to Users—Here's What We Learned and How We're Fixing It

Hey r/microsaas, I'm one half of a two-person team behind a B2C SaaS we launched a week ago, and I owe this community a raw reflection on where we went wrong. Picture this: two technical nerds, heads buried in code, thinking we could build the perfect product and users would magically appear. Spoiler: they didn't. If you've ever fallen into the same trap, I hope our story saves you some pain—and I'd love your advice on digging ourselves out.

Three months ago, we started building a platform to connect people who want to team up on side projects—think indie hackers, students, or anyone itching to create something cool together. The idea came from our own frustration with solo projects fizzling out and the lack of a good way to find the right collaborators. As engineers (I'm full-stack, my co-founder's frontend), we dove straight into building. We spent hours obsessing over code optimization, polishing the UI, and tweaking database queries. We thought a flawless product was the ticket. That was our first big mistake.

Here's the humbling truth: we didn't talk to a single user until after we launched on April 28. No customer interviews, no landing page to gauge interest, no early adopters—just us, our IDEs, and a whole lot of hubris. We figured, "Build it, and they'll come." Well, we built it, and the only thing that came was silence. Zero users. It's like throwing a party and forgetting to send the invites.

View more of my work [here](#).

Coding-First Puts SaaS Founders at Risk

Building a product that no one wants not only drains your resources, it further impacts your product's future. Here's a look at how:

Delaying Validation and Breaking Feedback Loops

The longer you wait to talk to users, the longer it takes to discover you're off track. Early user conversations help you clarify the real pain point, identify how users describe it, and find out what they're already using.

According to a recent [ThinkUp article](#), startups that embrace regular feedback loops cut product failure rates by nearly 50%. Companies like [Buffer](#) report up to a 51% increase in actionable insights by using continuous feedback tools such as Slack or [Microsoft Teams](#).

Founders often delay validation, assuming they'll fix things after launch. But, [one study](#) found that it takes 2–3 times longer to find product-market fit than most founders expect (Failory, 2024; Startup Genome Project). That delay eventually drains cashflow, quietly killing the startup before the product ever fits the market.

Confusing Innovation With Invention

Innovation solves problems people feel; invention just creates something new. Likewise coding first makes you an inventor while having conversations first makes you a problem-solver. In SaaS, people don't pay for novelty, they pay for relief.

The [Startup Genome Project](#) found that founders overestimate the value of their intellectual property before product-market fit by 255%. This shows how coding first can mislead teams into prioritizing perceived innovation over real user needs.

Losing the Emotional Insight Behind User Behavior

Talking to users uncovers emotional triggers such as frustration, confusion, and unmet expectations. Code can't reveal that. Yet, those emotions are what great UX, onboarding, and retention strategies are built on.

In fact, [behavioural-trigger research](#) shows that SaaS companies using emotional insights can reduce churn by up to 40%. When personalization or emotional alignment misses the mark, [76% of users end up feeling frustrated](#). This is a big reason for early churn.

View more of my work [here](#).

Understanding the Ultimate Cost of Building Without Users

At this critical stage, energy and resources that could have gone into research and validation are wasted. Founders stare at stagnant metrics. Teams wonder if all the late nights and hard decisions were for nothing. Features that once felt smart now feel useless.

All too often, the reason the operation's beams finally snap is financial: [around 29% of startups fail after running out of funds](#). That unreinforced ground suddenly threatens to bring the entire building down.

Many SaaS companies don't recover and end up shutting down. Even pivots rarely work out, as many risk becoming another 'what happened to…' story. In micro-SaaS, [92% fail within three years](#), more in just 18–24 months.

Every week spent without talking to users is another risk, another beam bending under weight. It doesn't just mean building the wrong features, it could mean sinking the whole product.
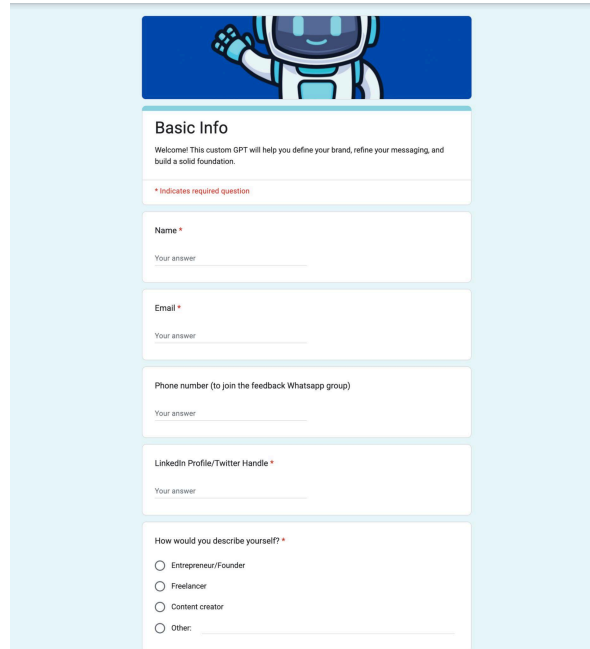
All these consequences show why building without talking to users can seriously hurt your product.

The good news? These risks and lost opportunities are avoidable. By starting with conversation, validation, and early feedback, you turn guesswork into insight and give your product a far stronger shot at success.

Leading Product-Building With Conversation

The best SaaS founders start with conversation, not code. They observe behavior, run small tests, and iterate with empathy. Slack, Notion, Canva, and many others didn't just launch features, they listened first, spotted friction, and built solutions that felt inevitable to users.

I learned that lesson firsthand while testing a beta sign-up form for a custom-GPT.

View more of my work [here](#).

Even though many people clicked on the call-to-action, few completed it. Like with many SaaS startups, users abandoning sign-up halfway is never a good sign and signals a deeper issue. That was the friction in my process.

To fix it, I asked a friend for unfiltered feedback on the signup form. Our candid conversation eventually led to me simplifying the sign-up to 4 fields, making the fifth optional. According to Formstack, removing just one form field can boost conversions by ~50%.

Within 2 days, my signups nearly doubled, and users trusted my product more because the experience was smooth and aligned with their motivations.

That experience taught me what countless SaaS founders eventually realize: users reveal the truth in what they do, not what they say.

Early friction costs SaaS companies users. Pendo shows that software products retain just ~39% of users after one month and ~30% after three months, emphasizing the importance of observing and iterating. Small, deliberate observations often reveal behavior patterns that assumptions or brainstorming alone would miss.

The best SaaS products aren't born from lines of code, they're shaped by the quiet discipline of listening, observing, and iterating.

View more of my work here.

Turning Coding-First Mistakes into Learning Wins

Even if you dove straight into building, it's not too late to course-correct and reinforce your building site. The goal now is simple: stop assuming, start listening.

Pause and Take Stock

You've spent months building, now it's time to be honest and identify which parts of your product truly solve real problems.

A founder on Indie Hackers shared that after six months of building features nobody needed, they decided to list every feature by utility, and scrapped the ones that weren't solving real problems. That honesty became the foundation for a successful pivot.

Talk to Real Users Immediately

Even if you've spent months coding, it's not too late to start listening.

Another founder shared on Indie Hackers that after talking to just 10 potential users, they discovered their messaging was off and a key feature wasn't needed. Those conversations reshaped their roadmap completely and doubled engagement.

Observe How People Actually Behave

Hearing users isn't enough, you have to see what they do.

A startup documented by Sprintwell noticed users getting stuck on the third step of onboarding. By fixing that one friction point, retention jumped to 32%, proving small observations can have a massive impact.

Iterate Quickly

Small, fast changes matter more than massive features.

On Reddit, one founder shared that after switching to rapid iterations based on user behavior, their signup completion improved from 34% to 52%. Time to first value also dropped from 8 to 3 minutes.

Validate Continuously

You can still test your assumptions before going further.

View more of my work here.

One founder on [Indie Hackers](#) built a landing page before building the full product and secured over 100 waitlist sign-ups in 30 days. This gave genuine feedback before he committed to months of work.

Keep Track of What You Learn

Documenting insights turns scattered observations into a roadmap for success.

One founder shared on [Reddit](#) that by tracking every conversation, observation, and tweak, they spotted recurring frustrations. Following these patterns doubled retention and prevented chasing useless features.

This kind of turnaround takes time, but it's worth every bit of effort. Because, now your building plan is back on track to creating a solid, three-story masterpiece.

Learning from Conversation-First SaaS

Here's a quick look at leading brands that champion conversation before they build:

Intercom

Before building their messaging platform, Intercom's founders actually talked to small business owners to understand their pain points. They ran experiments, watched how teams handled support, and iterated features based on what they saw.

The result? A product that clicked immediately with SMBs and became a go-to for customer communication.

Slack

Slack was supposed to be a video game. Along the way, the team noticed internal communication was a nightmare, so they talked to potential users before pivoting. They tested workflows, notifications, and integrations based on real feedback.

The outcome: A tool teams love, with millions of users and a $27.7B acquisition.

Canva

Canva's founders spoke to non-designers to figure out why design tools felt impossible to use. They launched a beta, watched people struggle, asked what worked and what didn't, and iterated fast.

View more of my work [here](#).

The result? An intuitive platform used by over 100 million people.

Duolingo

Before coding lessons, Duolingo's team tested with students to see why learning apps frustrated them. They prototyped exercises, ran A/B tests, and refined based on real user responses.

The outcome: a global language-learning leader with over 500 million downloads.

Notion

Notion's founders noticed knowledge workers juggling too many tools. They interviewed users, launched a simple MVP, observed behavior, and iterated on databases and customization.

The result? Millions of happy users and a top collaboration tool.

These five brands embody the listen–observe–iterate mindset, and serve as clear blueprints for founders who want to build products that truly improve users' lives.

Rethinking Your Approach Before You Build Again

Innovation alone isn't enough without proper context. Don't be the builder who spends months drawing plans and selecting materials, only to realize the foundation can't hold the weight. Every beam and wall could risk collapse. That's what building a product without talking to users feels like.

The solution is simple, but it takes discipline: start with conversation. Validate your assumptions early, observe real user behavior, iterate based on insight, and document everything you learn.

Even if you've coded first, it's never too late to pause, listen, and course-correct. When you build the product that fits the ground you're standing on, everything else falls into place.

So, talk to three users today and use what they tell you to validate your SaaS product.

View more of my work here.