

# The roadmap to Product/Market Fit (PMF)... maybe

by Jason Cohen on December 24, 2023

This eight-step process brought WP Engine from an idea to a Unicorn. While there are other roads to Product/Market Fit, consider copying some of these ideas.



"What was initially thought to be a simple process is in fact an incredibly complicated, intricate, and complex system that I've codified and organized into a few easy-to-follow rules that are more difficult to implement than you'd think."

What is the formula for going from an initial product idea to Product/Market Fit—a company that is growing and sustainable, with customers that want to pay and want to stay.

(Assuming you even agree with that definition of Product/Market Fit.)

There isn't a fail-safe roadmap of course, but there is a progression that describes how WP Engine became a unicorn, supported by 17 years of articles from this site.

Here's what that progression looks like. Then we'll explore the other truth—that it often doesn't work like this after all.

1. **Personal Fit:** "Passion" is useful, but winning requires a personal edge.
2. **Market Fit:** Most good ideas aren't good businesses.
3. **Customer Fit:** Talk to customers before you waste months building the wrong thing.

4. **Build and ship the SLC** quickly: It's where the real learning happens.
5. **Marketing and Sales** more than writing code and tweaking design.
6. **Retention-driven Product** development; attend to your existing customers first.
7. **Prioritize** systematically, ruthlessly, strategically.
8. **Manage your psychology** on a journey of self-discovery and constant rejection.

---

## 1. Personal Fit: The inner fire, leveraging a personal edge to do what you were meant to do

A great idea or a great strategy that *you* can't execute well, isn't a good strategy *for you*.

You start with nothing—no product, no customers, no brand, no distribution, and compared to competitors, no money and no time.

To succeed, you need *something*. Something special. Something that gives you an edge, despite being woefully inadequate in every dimension. Is "passion" that something?

### Passion

“*Do what you love, and the money will follow.*”

—Motivational advisors

That is false, as evidenced by most artists, philosophy majors, and the 80% of startups that fail despite founders' genuine love and obsession.

Passion *is* required; it's just not enough. Passion is the motivator, especially in the dark times when your savings is depleted, competitors are beating you, as you suffer a constant bombardment of rejection from potential customers and potential employees and potential investors, as you question your self-worth and wonder whether the cause of your Impostor Syndrome is that you're actually an impostor.

“*You have to be burning with an idea, or a problem, or a wrong that you want to right. If you're not passionate enough from the start, you'll never stick it out.*”

—Steve Jobs

You need to articulate exactly what your passion is, so that you can use it as a filter for business concept. This is difficult; some people spend their whole life trying to figure it out. Jerry Colonna says “The purpose of life is to discover who you are, and then live fully into that.” It is not an exaggeration to say that a startup is who you are; that's why they call it “your baby.”

See the last section of this article on personal fulfillment for a list of questions that help you suss what your purpose and passions are.

Your goal is to **find the work you were meant to do**. Your calling. The work that you would do for free, but more—what you are *compelled* to do.



‘Listen, I’m fine with the robe, the sorcery, even mixing potions in the bathtub. But for god’s sake, Gary, would you *please* use a coaster?!’

Still, the existence of passion doesn’t imply the existence of a business model. (We’ll solve that in the next step.) Passion doesn’t give you a competitive edge, because all the other founders have passion, and larger companies have mountains of advantages that make “passion” look like a sling-shot attacking an aircraft carrier. You need an edge.

### Leverage

“Leverage” means yielding a huge output from a given input—accomplishing an inordinate amount of quantity or quality, with relatively little time and money. You generate leverage from the mixture of talents, taste, and experiences that you possess. To figure out how this works for you, see this article for many types and examples of leverage.

Leverage is good, but *unique* leverage is far better, because that’s your source of differentiation<sup>1</sup> from competitors and alternatives. Often this appears at the intersection of your peculiar above-average talents and experiences, which taken together are unique<sup>2</sup>. This uniqueness must then be coupled with a particular way that you are approaching this problem and solution, such that the combination of your leverage against your path is uniquely excellent, even for a small number of potential customers.

This leverage + path combination also forms the kernel of your strategy<sup>3</sup> as your competitive positioning. You need something that competitors lack, because your customers are going to compare you to competitors whether you like it or not, and you have to have an answer besides “we have this one minor feature that they don’t have” or “we’re \$10 cheaper.”

---

<sup>1</sup> “Different,” not “better.” Yes, you will be “better” according to some set of people, but definitionally that means you’re “worse” for another set. The latter might even be orders of magnitude more numerous than the former. That’s OK, and that’s why I don’t like using the word “better.” You should be distinctive, aligned with your calling and your strengths, so that the set of people who do find your brand of different to be “better” will flock to you, buy from you, and even love paying you for it.

<sup>2</sup> The quintessential exponent of this idea is Scott Adams, writer of the Dilbert cartoon, who points out that he is a decent illustrator but not a great one, a decent humorist but worse than any comedian on Netflix, and has held jobs in the tech sector, unlike almost any artist or comedian. It is in the intersection of being in the top 25% in each of these three “circles of competence” that he is unique, not because he is top 1% or “best in the world” at any one thing.

<sup>3</sup> Leveraging strengths into durable competitive advantage is one of the six characteristics of great strategy.

The key question that summarizes this intersection of “you” and “path” is:

**Why are you the perfect person to build this company?**

“*Be yourself. Everyone else is taken.*”

—Oscar Wilde

## 2. Market Fit: A working theory of why they will buy.

Once you inventory the inner world—the shapes of spaces where you were meant to play—you turn your gaze externally to figure out whether your idea works out in the market, in the world of customers, competitors,

trends, problems-to-solve, jobs-to-be-done, and products. These things you do not control, and thus must understand, conform to, but also exploit.

It’s right there in the name: “Product/Market Fit” means fitting into the market, not just building something that would be fun/interesting/edifying/curious/exciting/ego-enhancing. Those are all good reasons to have a side-project, but none is a reason why that side-project will become a profitable business, even at a scale that feeds a single person. That’s why most “indie hacker” startups and “AI is cool” startups fail—the genesis was “fun project,” not “plausible business.”

*“I had the problem myself, so I built a product to solve it.”*

This might be the most common origin story, tacitly concluding that “this must be a business because I would have been a customer.” Indeed, my startups all started this way. But your understanding of “the problem” as it pertains to you alone is much less likely to be a real business than you think. You are, in fact, not like your customer<sup>4</sup>.

---

<sup>4</sup> For a start, your customer isn’t quitting their day job to create a business.

Sometimes a passion project turns into a business anyway. That’s what happened to me at Smart Bear. More on that later. But that’s luck. The point of a framework like this one is to reduce your reliance on luck.

“The way to get startup ideas is not to try to think of startup ideas. It’s to look for problems, preferably problems you have yourself.

The very best startup ideas tend to have three things in common: they’re something the founders themselves want, that they themselves can build, and that few others realize are worth doing. Microsoft, Apple, Yahoo, Google, and Facebook all began this way.”

—Paul Graham, *How to Get Startup Ideas*

### A theory of why they will buy

You need a plausible theory of the customer, market, and business model. “Plausible,” because many ideas fall apart under honest scrutiny. “Theory” because of course it won’t be entirely correct, but it will be your working understanding of the world, which you modify as you learn.

Here’s exactly how to develop a specific theory of the market, customer, and your positioning. You’ll analyze these market characteristics:

1. **Plausible:** Do 10M people or 100k companies have the problem?
2. **Self-Aware:** Do they know & care they have the problem?
3. **Lucrative:** Do they have substantial budget to solve this problem?
4. **Liquid:** Are they willing and able to buy right now?
5. **Eager:** Do they want to buy from you, specifically?
6. **Enduring:** Will they still be paying(-it-forward) a year from now?

A lot of people read that article, then said they wish they had read it before they wasted two years of their life (and savings and heartache) building [insert name of failed startup]. Don’t be one of those people.

You might think it infeasible to answer these questions, because the research is impossible or the idea is so new that existing data and trends are irrelevant. However, as you’ll see in the article, you *can* do it using Fermi Estimation—a technique useful not only in market analysis but for ROI, probabilities, and decisions.



“Well, it’s not the worst I’ve seen.”

### Early strategy

Even better than having a great business model is to have a great strategy. Not a twenty-page document, but a one-pager that conforms to the guidelines in that article, explaining “how we will win.” It’s never too early to be asking yourself how to leverage your strengths to build products that are great and differentiated despite your weaknesses. It’s never too early to write down your assumptions and decisions, to ensure they are at least self-consistent if not self-reinforcing, so that when contrary evidence appears, you can notice that, and react methodically.

To generate tangible strategic ideas for your business model or strategy—explaining “how we will win”—consider creating a theory of where your product will fit in the customer’s Needs Stack, pick a few of these tactics for navigating the fact that the future is unpredictable, and decide how you will generate “Love” and “Utility” types of Willingness-to-Pay. All of these are fun, accessible, strategic, and they work in practice.

*“But often the first idea is wrong, so why do I need a specific idea with a specific business model?”*

Because walking in a random direction is not progress, and because the main way to discover the *right* direction, is to have a clear theory of the world, and notice when



the world contradicts it, so you can pivot into a better theory, and thus iterate into a genuinely great strategy and market-understanding.

That's what *actually* happened with Smart Bear, and Slack and WhatsApp and Flickr, as described and analyzed here. Only by having a specific, strong, clear idea, was it possible to notice what customers *really* wanted *instead*, which led each of those companies to pivot into the idea that became successful.

In all those cases, the idea was very personal to the founders—the personal-fit of passion for the solution and leveraging an edge to create something that resonated with early users. But it was only with a specific theory of business model and strategy, which was then negated by reality, and then an intentional pivot, to achieve Product/Market Fit.

### 3. Customer Fit: Find the ideal customers first.

There's only one source of truth for what customers will buy: Customers.

Not advisors, not experts, not analysts, not Twitter polls, not research, not past data, not even competitors' behavior. **You have to talk to customers.**



"So, tell me a little bit of what you think I want to hear about yourself."

Customers are fickle. You ask whether they'll pay \$100 if your product does \_\_\_\_\_ and they say "yes;" then you build it and they don't buy. So why talk to them at all?

Don't you need to put the actual product in front of them, and see what they actually do?

Customers can tell you what their lives are like, which is how you validate your business model and strategy from the previous section. Customers can tell you what they *won't* buy, which has happened to me repeatedly. You can discover that the average customer doesn't know the problem exists, or doesn't have a budget for it, or isn't prioritizing it. When they say "yes," it's a "maybe," but when they say "no," it's a "no," and you just saved yourself months or years of wasted time.

You can find out where they go to find products like this, so that you can advertise in the right place. You can find out what language they use to talk about the problem or solution, so you can copy that language in your advertising and social media home page and capture their attention. You can find out how their budgets work, so you can price and package and position accordingly. You can find out how they're addressing the problem today, so you know what you're selling against, whether that's a competitor, an alternative, or something they're doing by hand. You can find out what causes them to break out of their daily life and say as Bob Moesta puts it, "Today is the day I'm going to buy \_\_\_\_\_," so that you can try to be there when that event happens, or possibly even cause it.

It is easy to find examples of successful companies who never asked customers what they wanted. That happened to me at Smart Bear. But again that relies on luck, and most of the time, you don't get lucky. This is not a good way to gamble the next few years of your life.

**But you don't want to do this work.** You want to build the product because that's the fun part. Potential customers are hard to find and they don't want to talk to you. It's going to be like that when you have a product too, so if you can't do it now, you won't be able to do it later. Building the product first won't make it easier to find or talk to customers. Building the product first will,

however, ensure that you haven't actually built what people want, because you never found out what people want.



Justin ✓  
@JustinQuda

...

It's taken years for me to change my mindset from "I know best" to "what does the user want?"

I've been building a startup in my own echo chamber.

Today I did an 1 hr call with a Quda creator - it was awesome!

Talk to users - they won't bite. They want you to succeed too. 😊

### How to interview customers

Once you get someone on the phone, how do you interview them in a way that maximizes learning and leads to a specific theory that you can execute on, from your unique winning advantage to pricing?

### The customer validation system I've developed is the Iterative-Hypothesis customer development method.

This provides you with goals that you achieve using interviews, how to create the hypotheses that will drive your business model, and how to write questions that maximize learning. 15 years ago it invalidated a startup idea that I *thought* was good, and then validated the startup idea that became WP Engine—now a unicorn.

### How to get customers to talk to you

The way I got interviews for WP Engine is by using LinkedIn to find people who had the title and industry I was targeting (web developers in WordPress), and asked them for an hour of their time to chat about a new startup concept for whom they are the ideal customer. Furthermore I offered to pay any amount they wished for that hour, because I value their expertise and their time; I'm not asking for a donation, I'm genuinely interested in their expertise. Out of 50 requests, 40 agreed, and only one asked to be paid. (30 eventually became customers.)

Another technique is to buy ads for a landing page with an email sign-up form and web-chat, and talk to the people who arrive. Ask what they're looking for, what they

need it for, why are they looking for it right now, how they found you, and what else they're considering. When it goes well, ask them to hop on the phone and run the full method above.

Another technique: Go to an event where your target customers are, walk the halls, and interview everyone all day long. People like talking about themselves, and you get to hone your pitch with 100 practice sessions.

More techniques:



Christopher Gimmer ✓  
@cgimmer

Follow

...

About a month ago, I had an idea for a new SaaS app based on the pain I was experiencing with the market leader that completely overhauled their product for the worse.

Here's a few things I did to gain more confidence in moving forward:

1) Reached out to 10+ people in my network to see if they had pain points with [X] and did customer research interviews.

2) Sent DMs to random people on Twitter who were complaining about [x]. I told them we were thinking of building a competitor and if they'd be willing to hop on a call to share their current pain points. Some replied and agreed to hop on a call.

3) Read a bunch of G2 reviews of the incumbent to see if peoples' pain points matched mine.

4) Dug through many Reddit threads to see what the common pain points were.

By the end of all this, I was pretty confident that many other people were experiencing the same pain points as me and that there was a willingness to pay.

I then analyzed all the available marketing channels and put a plan together on how we'd get traction.

Going to be announcing the new product soon 🚀

### Find your ICP

Your goal is not only to validate your theory of the market, but to discover your ICP (Ideal Customer Profile). This is your "perfect customer"—a segment almost comically over-specified to be so perfect for your product that you are truly the best choice in the market, and they would be crazy not to buy. You will then aim all of your marketing messaging at this person: Website, advertisement, terminology, writing style, pricing. When the ICP

lands on the home page, it should be obvious in three seconds that this is perfect for them, that you “get” them, and they will be pulled through the process until they’re a customer.



**Bryan Smith** ✓  
@OrionSeven

...

The biggest mistakes I made in my first startup was planning an MVP far too large.

The root cause for that was I didn’t figure out my ideal customer profile.

Instead of building for everyone I should have built for someone specific which would have let me build far less.

You are scared that targeting only the ICP limits your potential market, but this is not what happens. In “[Selling to Carol](#),” I explain the mechanism and provide examples showing that for every ICP there are 10x more people who make similar buying decisions, and 100x more who take more convincing but ultimately also agree, and therefore you end up selling to a far wider market than you feared, yet earning attention and loyalty from having a clear, unique message. Your company, brand, and product will actually mean something.



“I’m looking for something in a small pond.”

Find and talk to the customers; they’re the only ones with the answers.

## 4. Build *and ship* the SLC.

Interviews give you a better model of the world, but the real learning doesn’t start until people are using—or more often, *not* using—a product.

SLC is my alternative to MVP:

**Simple**, because complex things can’t be built quickly, and you need to ship product to customers now.

**Lovable**, because crappy products are insulting to customers, and you didn’t start this company to make crappy products. The love overpowers the fact that it’s buggy and feature-poor. There are many wonderful, powerful, competitively-defensible forms of “Love.” Pick a few.

**Complete**, because products are supposed to accomplish a job. Customers want to use a v1 of something simple, not v0.1 of something broken.

Build something small (but also lovable and complete), and ship it to customers, because that’s when real “progress” begins. That’s how you learn the truth.

If you’re building for six months and a customer hasn’t used it, you’re not on the road to Product/Market Fit; indeed you’re not making progress on the business at all. What you’ve done is proved what you already knew—you can build software.

The learning starts only when customers are (trying to) use it.



**Brandon** ✈️ Flightcontrol ✓  
@flybayer

...

Yo, watching @LogRocket recordings of people using your new app is like the most painful thing in the world 🤔🏠

But wow, I cannot imagine living without it. People just won’t tell about all the little issues and confusions they have

5:42 PM · Nov 16, 2023 · 6,449 Views

## 5. Focus more on marketing and sales, less on product.

Your natural inclination is to do what you enjoy and understand, so you need to press yourself to do so much marketing and sales that it feels like “too much.”

Ask yourself: What can you do *today*, that will (a) get more people from the target market to come to the website or (b) convert more of those people to try the product or (c) convert more of those to buy the product.

The answer *might* be found in writing more code. Improving the new-user experience, for example, could increase the conversion rate from “try” to “buy.” But more often the answer isn’t inside the product; indeed several of those steps happen before the person has used the product at all. The code is actually your enemy.

You’ll advance the product anyway—I know that. And you should, but you don’t need to be told to do that. After all, it’s all you ever wanted to do, even at step one. That’s why you need to set an intention every day to make progress on one of those questions. That’s how you’re going to generate growth.



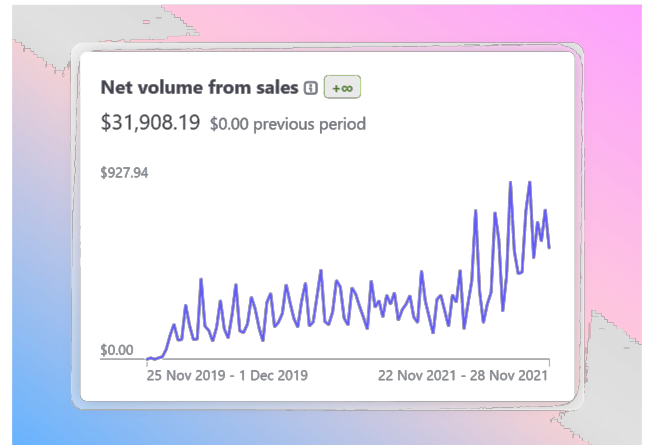
Pallyy's first 2 years were pretty rough, almost gave it in.

Rev almost covered costs, was using some savings.

Decided to go all-in on marketing before giving up.

Next 12 months it grew 10x.

Keep going.



10:08 PM · Feb 27, 2024 · 10.8K Views

26

6

156

29



## 6. Prioritize retention-based product work.

What does it mean, that one person out of billions cared enough about your problem-space to notice your advertisement or social media post, then felt compelled to follow the link, then was so intrigued by the website that they joined the 1-out-of-100 that tried the product, then liked what they saw enough to take out their wallet and pay, then started to use it... and then *cancelled*?!?

After all that—clear signals of desire, need, willingness to pay, and apparent “fit”—they decided “Nope, this is not what I want.”

What it means is: You made the right promise, but didn’t keep it.

Marketing is where you discover the promise the customer wants you to make, and retention<sup>5</sup> is where you discover whether you're fulfilling that promise.

---

<sup>5</sup> Actually, retention is a lagging, multi-factor indicator, which makes it useful as a high-level check, but not useful as an operational day-to-day metric. Here's a [complete system for practical metrics](#) that embraces this fact and points to the right set of metrics.



"No, you're thinking of Cupid. I'm the one-night-stand fairy."

Some people believe that low retention is fine because you can make up for it with growth. That's not true for two reasons:

1. Growth will start out good, but then will slow and halt sooner than you realize<sup>6</sup>.
2. If customers don't want to use the product, your "growth" is fake news. It's not Product/Market "Fit" if the "market" decides "not a fit" after actually using the product.

---

<sup>6</sup> See these articles with [data from Product/Market fit companies](#) and [the math behind growth grinding to a halt](#) at companies with low retention.

Therefore, when you do work on product features, build things that increase retention.

This starts by understanding why people cancel. Gaining this understanding is difficult, because once they leave they don't want to talk to you anymore. You might coerce

them by paying for interviews with Amazon gift cards or charitable donations. Some people will respond to an email (no harm in asking). Better is to look for signs that the customer isn't being successful (e.g. activity within the product) and reach out before it's too late, while they're still in the middle of their struggle, and therefore might be willing and able to articulate it.

Remember that your goal is not to discover what unsuccessful customers have in common, because successful customers often have those things in common as well. Rather, it is to find patterns in unsuccessful customers *that are not shared by successful customers*. Those are the attributes that lead to action.

For example, if "using feature X" means people are successful, maybe unsuccessful customers don't know about it (so you promote its usage), or maybe they can't use it because it's incompatible with their workflow (so add integrations or options). Sometimes, however, it's about the customers themselves, where tuning the ICP means you find more customers who are already a fit, perhaps targeting different industries or geographies, different stages or sizes of company, the title or attitude of the buyer, or the specific use-cases they are trying to solve for.

The [Talk/Walk workshop](#) might help, because it helps elucidate the difference between the promise and the execution. But probably, as usual, you just need to talk to customers.

## 7. Ruthlessly, methodically, systematically prioritize.



More than just “focus,” more than a generic admonition to “prioritize,” use a system that ensures you’re leveraging your most valuable, most limited asset (time) to maximize progress.

Do this by combining [Fairytale Planning](#) with the [Rocks, Pebbles, Sand](#) work-prioritization system:

### Identify the next milestone.

At all times, be crystal clear about the single most important thing to achieve.

*(Ex: Finding a good business model. Public launch. 20 paying customers. Getting trial conversion rates from 1% to 5%. Getting cancellation from 7%/mo to 4%/mo. Getting to \$10k MRR so you can quit your day job.)*

You have to be executing against the next milestone every day. Write it down and look at it every morning. Everything else is a distraction to be ignored, or a necessary evil that you should dispatch as quickly as possible, including delegation or accepting that it’s fine if it’s executed poorly or late.

### Identify the current obstacle to achieving the milestone.

At all times, be crystal clear why it is difficult or slow or expensive to achieve that milestone.

*(Ex: Can’t figure out how to leverage your strengths for something differentiated and desirable. Easy to get someone to use it once, but only 15% use it again. Got 17 customers from Product Hunt but now there’s no repeatable way to get more. Can’t get people to agree to be interviewed. SLC isn’t accepted by beta testers. Freemium users don’t have enough incentive to convert to paid.)*

You must face the difficult truth. Do not pick the obstacle that feels comfortable; pick the existential crisis that is uncomfortable exactly because it is critical and scary and true.

You have to be attacking or side-stepping the obstacle every day. Write it down and look at it every morning.

### Split work into Rocks, Pebbles, Sand.

Use this specific [Rocks, Pebbles, Sand framework](#) to prioritize and schedule different types of work with the correct philosophy and process appropriate for each type.

### Select the Rock that attacks the obstacle and milestone together.

Use [Binstack](#) to prioritize one Rock that addresses that most important milestone and attacks the obstacle. (You don’t have time for more than one.)

If you can’t think of one that’s good enough, don’t just proceed with a mediocre plan that will occupy the next three months; grab some friends—or better yet, customers!—and [brainstorm](#) a better one.

### Schedule Pebbles sparingly.

Because you have to focus on the Rock, and you have little time for anything else, use this [ROI framework](#) to prioritize just a few other activities that are valuable enough to justify spending your remaining time.

### Almost everything should be only “good enough”

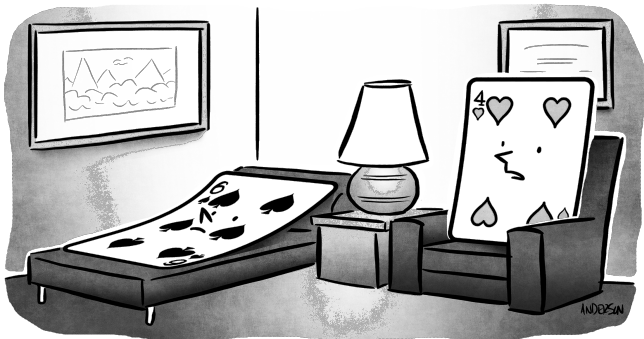
You have taste, you have craft, and part of the reason you’re doing all this is to express yourself. But almost everything you do will not determine whether the company is successful. (Repeat that last sentence until you fully internalize it.) Which means most things should only be “good enough.” Let it go.

Most metrics should be satisfied, not maximized. Pick KPIs accordingly. Design matters less than you think.

Having said all that, sometimes the very best aspects of a design are the little things, the obsessive things, the things you are compelled to make just so. Whether web design or UX design or a “perfect” feature which is only perfect because of the myriad corner cases that took a surprising amount of code and care to cover, but which means a new customer has an amazing experience. Pick your battles.

Don’t stray from the system. You don’t have the time.

## 8. Manage your psychology.



“So you’ll never be part of a royal flush. You know what?  
Me neither. And that’s OK.”

Throughout this a step-by-step roadmap, you face demanding psychological challenges.

- **Facing the truth**—not allowing “what you wish were true” to get in the way of finding out what is true, about your abilities, customer’s wishes, the shape of the market, why people leave.

- **Finding yourself**—coming face-to-face with who you are, and who you are not, figuring out what is actually important versus what is necessary or temporary, enjoying the journey especially when even a successful end result is often not what you hoped it would be.
- **Making clear decisions**—deciding what you’re not good at, and not trying to do it anyway; deciding your place in the market, and not trying to be all things to all people; deciding your ICP and having the fortitude to go all-in; deciding the current milestone, the main challenge, the one big thing you have to do, and sticking to it without getting distracted.
- **Constant rejection**—your ideas won’t be good enough, your customer interviews will negate key assumptions, 99% of website visitors won’t take an action, long sales calls will result in nothing, customers will buy but then leave, employees won’t join you, investors won’t invest.
- **Fluid change**—despite clear decisions, rejections often demand a change, which is difficult to accept, difficult to enact, difficult to admit when things are wrong, and difficult to decide whether some rejection should be ignored as an anomaly, or whether it’s indicative of a “learning” that must result in a change, difficult to tell the difference between the chaos that results in PMF and the one that results in failure.
- **Acting while uncertain**—being “all-in” even while knowing you must be wrong about some things, even while getting rejected and making changes, even while doing things you’ve never done while experts tell you what to do differently, even while feeling like an impostor who has no right to be here, even while knowing you have no idea what you are doing while it appears everyone else knows exactly what they’re doing. (Don’t worry, they really don’t.)

This might be the hardest thing you’ve ever done. It’s a gauntlet of pain, even if it’s also the most exciting thing you’ll do in your professional life. Many people stop be-



cause it's so hard. You can't blame them.

## But you're not going to do this, are you...

You have a fun feature idea so you're just building it.

You always wanted to learn Rust so you're just learning it.

AI is astounding and it's the future so you're building things with it, even though it's a technology and not a customer's problem that needs solving.

You've started your own company in part because you don't want to be shackled by "processes" and "frameworks," so you're just doing whatever you want.

Visual Studio is comfortable and Google Ad Manager is foreign and like Las Vegas is designed to separate you from your money, to say nothing of the discomfort of asking for the sale or getting someone on a call to talk about their workflows or get an earful about how your precious software actually sucks.

Uncomfortable. Scary. And you're lost, unskilled at those things, not making progress, not even *wanting* to. So you slip back to Visual Studio where you know what to do, and enjoy doing it, and you're good at it.

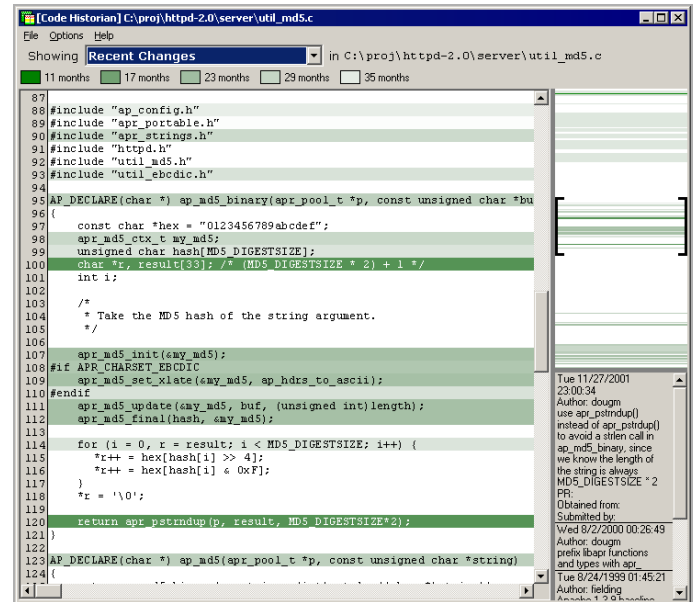
Then you post on Twitter about how you have \$1600 in MRR after 18 months but it's okay because perseverance is a virtue. When in fact you've fallen into the well-known trap of doing what you're good at, what you love, but not what the business needs done.

Perseverance *is* a virtue, if you're doing the right work, with the right goals. It's a vice if it means you're moving diligently in the wrong direction.

## Smart Bear only halfway followed the roadmap

But who am I to talk?

The company I founded in 2002, sold in 2007, and that gave me the online handle that persists to this day on this site, Smart Bear's first "product" Code Historian was a personal project, built because I want to learn .NET (we would later rewrite everything in Java), built because I thought there was value in the historical record encoded in version control (all products that operated under that assumption failed).



This doesn't make me look old, right? *Right???*

I didn't talk to customers ahead of time. I did talk to them a lot after they started paying, and learned that many didn't really want Code Historian at all, but rather they were abusing it to accomplish something else (in a story told here). As a result, I created Code Reviewer, which sort of hit the need in question, but the architecture was all wrong and the features were all hacks. But all this customer feedback gave me the confidence to rewrite everything from scratch (which everyone says you should never do), resulting in a product called Code Collaborator, which made millions of dollars.

Would I have found the idea for Code Collaborator by interviewing customers using the "Iterative-Hypothesis" method above? No, because I wouldn't have known what to ask them, and they wouldn't have known to tell me. I would have discovered that Code Historian wasn't a good



business idea... and that's it. Maybe I would have had a completely different and better idea. Maybe. But the fact is, the actual "path to success" was to create something I liked, just interesting enough to attract initial customers even though it's not a great business, and then pivot to Code Reviewer, and then pivot to Code Collaborator. The interviews would have stopped everything cold. Does that make "interviewing" the wrong road to Product/Market Fit?

I do think building and getting product into customers hands is far more valuable than conversations. Smart Bear was an example of (1) having a specific idea matching my passions, which then because of (4) building and shipping and (5) marketing and selling and (6) building for retention by talking to customers, lead to the right idea. So, it was part of the path, but not the full path.

So, while the path to success didn't honor every step in the roadmap, it still would have failed if I had violated those other steps—coding in isolation, coding instead of selling, coding instead of honestly figuring out what customers really wanted.

## WP Engine followed the roadmap and became a unicorn

That said, with WP Engine I traveled this roadmap exactly.

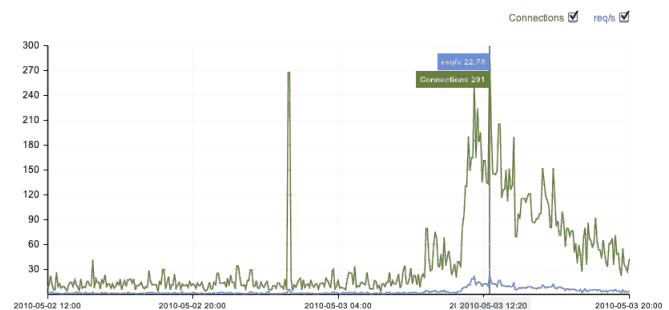
WP Engine wasn't the initial idea. The first idea was for a marketing analytics tool with features that Google Analytics lacked: A real-time event dashboard, a clever goal-setting-and-measuring system that worked retroactively, and the ability to capture web-form data even when it was only partially filled-in and never submitted, all with just one line of Javascript<sup>7</sup>. But I ran that Iterative-Hypothesis method and after 20 interviews I learned that the idea wasn't strong enough<sup>8</sup>. I was on the road; I (1) had a specific idea that (2) was clearly a great market and (3) vetted it with customers. Thanks to (3), I did not waste years on that project.

<sup>7</sup> These features are typical of analytics tools today, but in 2009 it was innovative.

<sup>8</sup> [This article gives the details of this invalidation](#), and how it was different from WP Engine, which the same method validated.

I then had the idea of a managed WordPress platform because I needed it myself—my blog would crash when I got to the top of Hackernews (a link-sharing site). I asked other bloggers what they use to keep WordPress fast and scalable; many said, "I don't know, but if you find it, tell me, because I need that too." I ran the Iterative-Hypothesis method again, and this time it was very clear what the need was, what people would pay, and how to find them. And the market was already huge (WordPress powered 11% of the largest 10 million websites) yet growing fast (in 2023 it's 43%). I knew WordPress because I was a user, and for nearly twenty years I've loved code optimization problems, so this was a problem I understood and am ideal to solve, where customers agree the problem exists, are already spending money on it, and the market is large and growing. That's steps (1), (2), and (3), but this time the concept was validated.

I built just enough software to fulfill the promises of speed and scale, building the first version in 30 hours. Good enough to not only stay alive and fast on a Hackernews day, but with almost no load on the server, which meant I was still far away from a capacity limit.

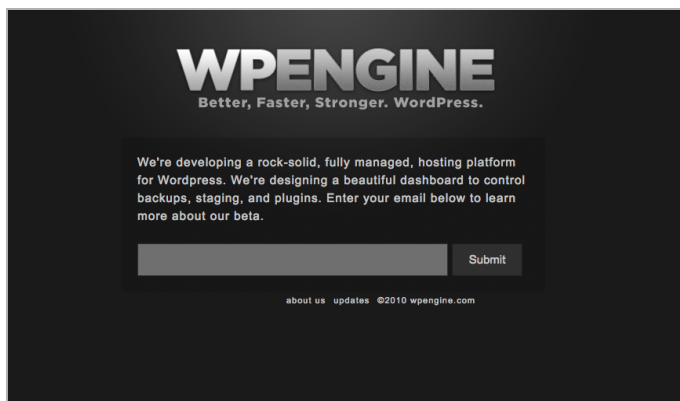


Hundreds of simultaneous connections on a Hackernews day in 2010

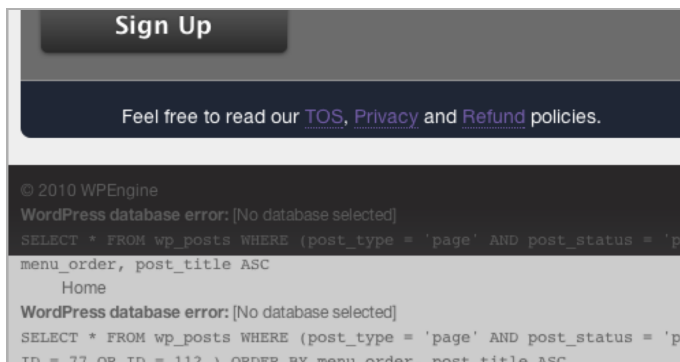
The first version didn't even have a customer dashboard. I took credit cards with the default web-form supplied by the online credit card service, and the rest was just

WordPress.

The first version of the website was... “spare,” shall we say?



The second version of the website was so bad, there were database errors shooting out the bottom. Database errors, on the home page, of the company that you're supposed to trust with your precious website.



The market need must be overwhelming, if customers walk through this wall of fire and say “yeah just take my money.”

So that's step (4), the SLC product that fulfills the job completely with the absolute minimum of “Product.”

We reached \$1M ARR after 18 months. The reason—besides having the right product with the right promise in the right market, which was elements (2) and (3)—is step (5) focus on marketing and sales.

Because I did marketing and sales before even having a product login screen, I was able to launch with 30 paying customers. Sure I worked on the product and worked

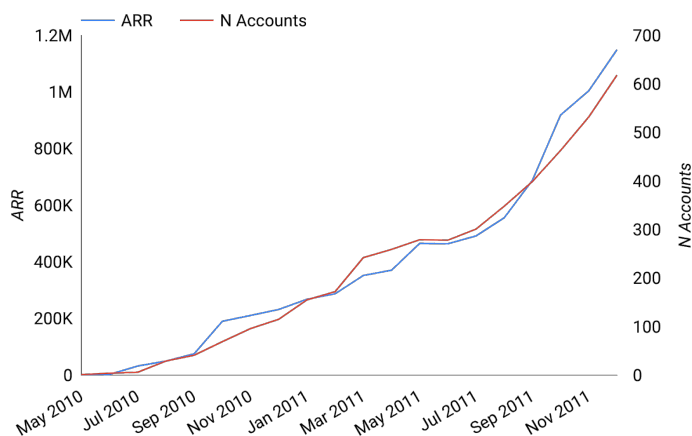
support tickets, but what I focused on was how to get customers:

**QUESTION FOR YOU:** Are there places we should be advertising? Sponsoring? Pushing our affiliate program?

**QUESTION FOR YOU:** Do you know other blogs (on any platform) that should be using us? Please make an introduction!

From an “advisory group” email update from 2010; my asks were for marketing and sales, not product and engineering.

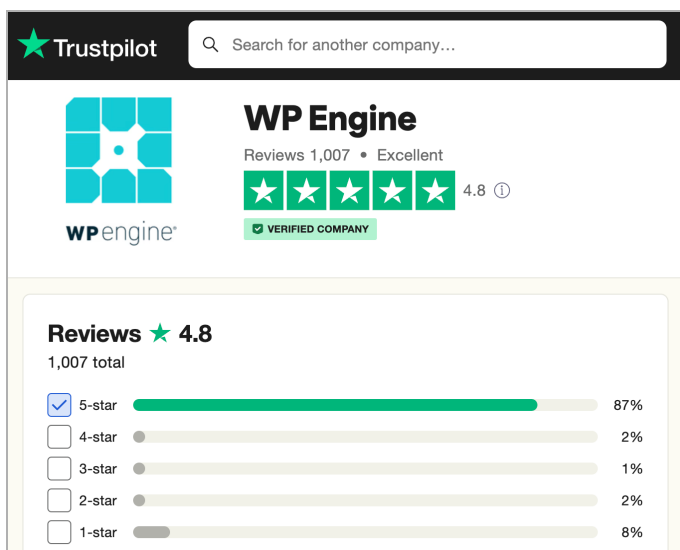
When you do marketing and sales, with a product that people actually need, in a large and growing market (all covered by the exercise under (2)), then you might have a nice growth curve:



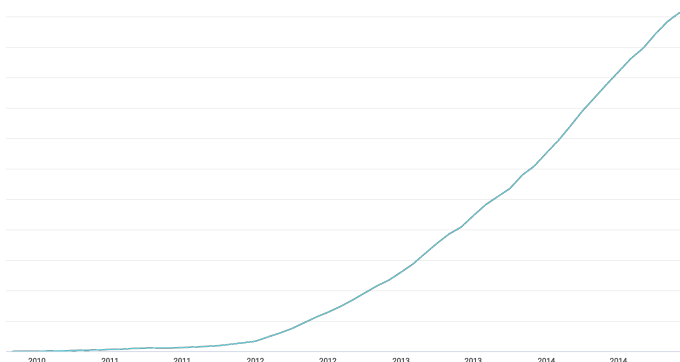
Hit \$1M ARR in first 18 months at WP Engine.  
But wait, the Product/Market Fit is still to come...

We've always spent a lot of time on (6) retention. Sometimes years would pass where we wouldn't add a new feature, because all our engineering time was spent on the challenges of scale and continuing to improve on the promise of speed and uptime. As a result, we've had best-in-industry customer retention for 14 years.

High retention due to happy customers folds back into growth through word-of-mouth referrals and a thousand top-of-class reviews:



All that, combined with the hard work, the craftsmanship, the large and growing market—did I mention the large and growing market?—and yes also the luck, meant that in early 2012 we hit Product/Market Fit, as further detailed here:



The previous chart's "up and to the right" growth from 2010-2012 looks almost flat compared to the sudden moment of Product/Market Fit, when it became hyper-growth, and never slowed.

This roadmap worked for WP Engine, and it worked for others. Consider this thread from [Simon Hoiberg](#), explaining why his second bootstrapped company succeeded ([FeedHive](#)) where his first failed. This roadmap echos within it—playing in a large, growing market, finding and focusing on an ICP, building only an SLC to start, and treating marketing and sales as more important than features and innovation:

**simonhoiberg**
1w ...

My first startup never made any money.

My second startup passed \$10K MRR in year 1, and is doing +\$1M ARR today, 3 years later.

What did I do differently?  
Let me share the lessons learned 📌

❤️ 💬 ↺️ ▼

4 replies · 80 likes

---

**simonhoiberg**
1w ...

→ I talked to users and listened instead of telling myself that I knew better.

→ I built in a very well-established market with verified demand instead of trying to innovate.

→ I consistently marketed my product - slow, but sustainably - instead of trying to "blow up" using the latest growth hacks.

❤️ 💬 ↺️ ▼

👤 2 replies · 21 likes

---

**simonhoiberg**
1w ...

→ I kept my product slim but effective instead of trying to solve everything for everyone. No one likes a "feature creature".

→ I put UI/UX on a pedestal and had a huge focus on \*how\* we solve a problem, instead of just solving it "somehow".

→ I focused on users who truly appreciated the unique way we're solving problems differently instead of trying to target everyone.

❤️ 💬 ↺️ ▼

16 likes

Do you *have* to walk this road like Simon discovered with FeedHive and I discovered at WP Engine, or can you stumble onto it halfway through like Smart Bear, or can you take a completely different route and just emerge at the finish line? Or can you just cherry-pick the things you like from this framework, and ignore the rest?

Yes to all the above. Everything is possible.

But not all roads are equally likely to result in success. We enjoy stories about outliers because exceptions are interesting. But exceptions are not repeatable, and therefore they may not be teaching you anything useful. I believe the roadmap above is repeatable. Not a guarantee of success, but a process that makes sense.

It's your life; you can take any road you please.

What are you going to do?

Many thanks to Gordon Daugherty, Rowan Udell, and Sathyanand S for

*contributing insights to early drafts.*

---

Printed from: *A Smart Bear*

<https://longform.asmartbear.com/product-market-fit-formula/>

© 2007-2024 Jason Cohen  @asmartbear