

The Five Ps

The 5-P Product Stage Model — Preparation to Profit

When building is cheap, only results are scarce. The Five Ps stage a product from Preparation to Profit, and answer the question the maturity models don't: not how good are you, but how far did the product get.

PBHQ Original

v1.0

Last updated August 2026

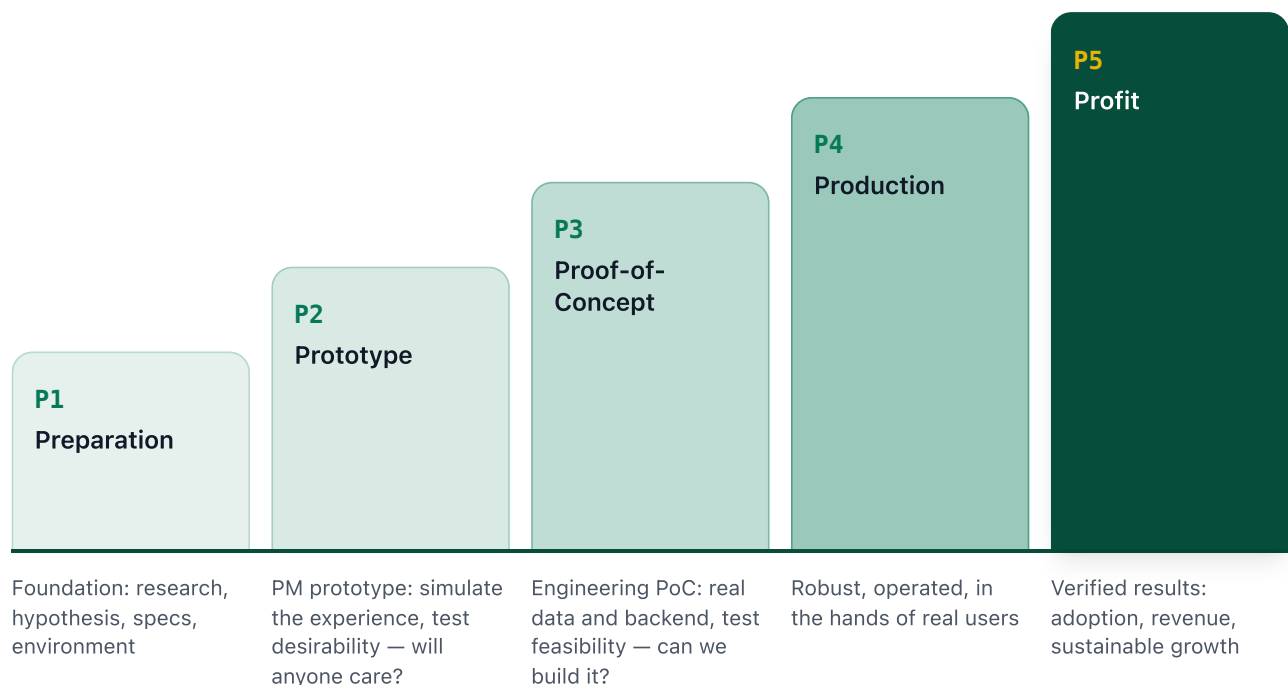
five-ps

product-stage

profit

traction

solo-founder



Minimally aim for Production. Ideally, Profit. ↗

There is a saying making the rounds as AI collapses the cost of software: *when building is cheap, review becomes important*. It is true, and it stops short. Review checks whether the thing was built right; judgment checks whether it was worth building. Both are necessary. Neither is sufficient.

When building is cheap, only results are scarce.

Results were always scarce — before AI and after it. A product lives or dies on whether anyone adopts it, pays for it, comes back to it, and you are always fighting for that attention against everything else competing for the same eyes. What AI changed is not the scarcity of results; it is the scarcity of everything else. Building was the hard part — the bottleneck, the moat — and now it is cheap. When the other scarcities fall away, the one that remains is the one that always mattered most.

So review and judgment are necessary, not sufficient — and sufficiency is the goal. When a prototype costs an afternoon, the prototype proves nothing; everyone has one. Go to the Lovable community and watch what gets celebrated: not builds, but revenue screenshots. When building is free, the demo is no longer the flex. The results are.

The Five Ps is the model for that. It stages a product from an idea to a business, and its whole purpose is to keep you moving past the stage where most products — and most builders — stop.

The Five Ps

The model has one job: name the stages so that stopping at an early one becomes a visible choice rather than a default.

#	Stage	What it means
P1	Preparation	Foundation — research, hypothesis, specs, environment. The work before the work.
P2	Prototype	The <i>PM prototype</i> — a backend-less click-through that simulates the experience and tests desirability: will anyone care?
P3	Proof-of-Concept	The <i>engineering PoC</i> — a spike against real data and a real backend that tests feasibility: can we build it?
P4	Production	Robust, operated, in the hands of real users. The first stage that is a real product.
P5	Profit	Verified results — adoption, revenue, sustainable growth. The point of the exercise.

Each stage is taller than the last because the stakes are. And the top bar is a destination, not a plateau: P5 is where the model points, the same way the maturity models point at their converged states.

The Product Lens: Why Prototype Comes Before Proof-of-Concept

An engineer reading the ladder may balk: doesn't proof-of-concept come *before* prototype? In engineering, yes — a PoC proves feasibility, and a functional prototype is the first real working model that follows it. The Five Ps orders them the other way on purpose, because it reads a product through a **product lens**, and "prototype" means two different things:

The **PM prototype** (P2) is behavioral — a click-through built in Figma, v0, or Claude with hardcoded data and no backend. Its only job is to simulate the experience and test **desirability**: *will anyone care?*

The **engineering proof-of-concept** (P3) is technical — a spike against real data and a real backend that tests **feasibility**: *can we build it?*

They test different risks, and a product-led model tests desirability first — deliberately. It takes hours to prompt a convincing frontend and weeks to stand up a stable backend, so you prove people want it before you prove you can build it. The PM prototype also *defines* the PoC: it shows engineering exactly what data the backend will have to handle. This is the same ordering the two loops enforce — the Product Loop asks "right thing?" before the Builder Loop asks "built right?" An engineering-led model would run P3 before P2; a model built for profit runs desirability first. **The order is the thesis.**

Why It Exists: Don't Stop at Prototype

This model predates ProductBuildersHQ. It was written by a solo founder with a specific message for product managers: **do not stop at prototype. Minimally aim for Production. Ideally, Profit.**

That message has only become more urgent. AI made prototypes nearly free, which means the P2 trap is now the default failure mode of AI-native building: an endless stream of impressive demos that never become products and never earn a dollar. A prototype that stops at P2 is a toy app — not an insult, a stage. The problem is staying there.

Read as stages, the Five Ps permit stopping; a proof-of-concept is a legitimate place to learn something and move on. Read as a commitment — the way it was meant — **P4 is the floor and P5**

is the goal. The model is normative on purpose.

Results Complete the Product Loop

The Five Ps is the product-stage view of a claim ProductBuildersHQ makes elsewhere. The **Product Loop & Builder Loop** model says a product needs two loops: one to build it right, one to decide it was worth building and verify it worked. The Five Ps is that second loop rendered as stages. P1 is the Product Loop's origination; P2–P3 are low-ceremony Builder Loop passes; P4 is the Builder Loop with real operations; **P5 is the Product Loop's verification half — Measure and Validate & Grow — made the destination.**

This is also where the model completes a distinction the frameworks draw carefully. Elsewhere we argue that taste and discernment — earned opinion and proven expertise — are the two inputs AI cannot supply. Results are the counterpart: the output only the market can supply. **Taste proposes; results dispose.** Judgment gets a product into the arena; results are the only thing that validates the judgment was right. A framework that stopped at judgment would be telling you to trust your own taste indefinitely. The Five Ps insists the market gets the last word.

It Interlocks With the Maturity Models

The Five Ps grades the *product*. The Product Builder Maturity Model grades the *person*; the Software Delivery Autonomy Levels grade the *process*. They are not three views of the same ladder — they are three different nouns. But they interlock, and the seam is visible in a shared word:

P2 “Prototype” is also PBMM Level 2, “Prototype Builder.” That is not a coincidence — it is the same insight discovered from two directions. Prototypes are what Prototype Builders produce, by definition. Which yields a theorem worth stating plainly:

The person's maturity caps the initiative's attainable stage.

A Level 2 builder produces Level 2 products. Reaching **P4 Production** requires the Level 3–4 skills of shipping and operating end to end. Sustaining **P5 Profit as a solo builder** requires climbing toward Level 5 *and* rising delivery autonomy — because one person cannot run production operations and growth experiments at the same time at low autonomy. This is why the **autonomy levels** matter to a business outcome that looks purely commercial: the \$1M — and eventually the

one-person \$1B — solo product is not a marketing goal, it is a capability threshold. The maturity models are the prerequisites for the Five Ps' upper rungs.

The Person and the Product

The cleanest way to hold the two models together:

The Product Builder Maturity Model is how you become a ProductBuilder. The Five Ps is what you aim your products at.

The person's ladder and the product's ladder. Together they restore the weight of the first word in the role: the **Product** in ProductBuilder means building *for profit* — for a result — not for the sake of building. In an era when building is the easy part, that is the whole point.

Minimally aim for Production. Ideally, Profit.