



MVP Planning Blueprint



MVP Planning Blueprint

Critical steps every founder must take before writing the first line of code to ensure product-market fit

sadhyarshi.com

Introduction

Most startups fail not because they couldn't build — but because they built the **wrong** product. This blueprint is your pre-flight checklist: eight battle-tested steps ensuring every line of code you eventually write is backed by evidence, empathy, and a clear path to market.

The Three Phases

Phase 1 — Discover (Steps 1–3) Understand the problem and the people who have it.

Phase 2 — Validate (Steps 4–6) Test your assumptions before building anything.

Phase 3 — Define (Steps 7–8) Scope your MVP tightly and set success metrics.

Step 01 — Identify a Real, Painful Problem

Great products are born from genuine pain, not clever ideas. Find a problem people desperately want solved — then confirm it obsessively before writing a word of code.

Action: Run Problem Safari Interviews

- Talk to 20+ potential users — in person or via video call
- Ask open-ended questions: "Walk me through your last experience with X"
- Never pitch your idea during this phase — just listen
- Look for recurring frustrations, expensive workarounds, or manual hacks

Tool: The Problem Validation Matrix Rate every problem across four dimensions:

- Frequency — How often does it occur? (Daily / Weekly / Rarely)
- Intensity — How much pain does it cause? (Rate 1–10)



- Budget — Are people already spending money trying to fix it?
- Willingness to switch — Would they change behaviour for a better solution?

Insight: Only pursue problems that score HIGH on at least two of the four dimensions. A rare, mild inconvenience is a feature request — not a business.

Step 02 — Define Your Ideal Customer Profile (ICP)

Not everyone is your customer. Trying to serve everyone is the fastest path to serving no one. Get ruthlessly specific.

Action: Build a Single-Page ICP Document

- Demographics — Age, role, industry, company size (B2B) or lifestyle (B2C)
- Psychographics — Goals, fears, motivations, and daily habits
- Current behaviour — What tools or workarounds are they using right now?
- Trigger events — What situation pushes them to actively seek a solution?

Rule: The Mom Test Filter

- Named after Rob Fitzpatrick's book — test every customer assumption ruthlessly
 - If your ICP is "anyone who needs X", it is too broad — narrow until it hurts
 - Your first 10 customers should all look nearly identical
-

Step 03 — Map the Competitive Landscape

Understanding what already exists helps you find the gap — and avoid building something the market already has at scale.

Action: Competitive Intelligence Sprint

- List all direct competitors (same solution) and indirect ones (same problem, different approach)
- Create a 2×2 matrix using the two axes your ICP cares about most
- Sign up for and genuinely use the top 3 competing products for a full week
- Interview users of competitor products — ask what they wish was different

Insight: Having no competitors is a red flag, not a green light — it often means no market exists. Look for crowded spaces with unhappy users. That is your opening.



Step 04 — Formulate and Prioritise Your Assumptions

Every product idea is a bundle of assumptions. Make them explicit now so you can test them before they cost you six months of development time.

Action: Build the Assumption Backlog

- Write down every assumption your business relies on aim for 20 or more
 - Examples: "Users will pay ₹499/month", "Onboarding takes under 5 minutes", "SMBs will trust a new vendor"
 - Sort by Risk × Learnability test the riskiest and easiest-to-test ones first
 - Each assumption becomes an experiment with a clear pass/fail success criterion
-

Step 05 — Validate with No-Code Experiments

You do not need a product to start learning. The fastest validation is to simulate the outcome without building anything.

Technique: Smoke Test Landing Page

- Build a one-page site describing your product and its core value proposition
- Add a "Get Early Access" CTA and measure sign-up conversion rate
- Target: 5–10% conversion from cold traffic is a strong signal of genuine interest
- Tools: Carrd, Webflow, or a simple page on Vercel live in under a day

Technique: Concierge MVP and Wizard of Oz

- Concierge — Manually deliver the outcome your product will automate, no software needed
- Wizard of Oz Build a polished facade; a human does the work behind the scenes
- Charge real money from Day 1 willingness to pay is the strongest validation signal
- Aim for 5+ paying customers before writing any production code

Insight: If you cannot get 5 people to pay for a manual version of your product, a polished app will not fix the underlying problem.

Step 06 — Conduct Problem-Solution Fit Interviews

After your no-code experiments surface interest, go deeper to confirm you have the right solution not just the right problem statement.



Framework: The Sean Ellis 40% Test

- Ask early users: "How would you feel if you could no longer use this product?"
- Target: 40% or more answering "Very disappointed" — below that, iterate first
- This is your product-market fit compass — reach 40% before scaling anything

Action: Solution Interview Protocol

- Show prototypes or mockups, not the product — Figma or pen-and-paper both work
 - Ask: "Does this solve the problem you described?" and "What would stop you from using it?"
 - Record sessions with consent — watch for hesitation, confusion, and polite non-answers
-

Step 07 — Define the Ruthlessly Scoped MVP

An MVP is not a beta product. It is the smallest possible thing that delivers one core value to one specific user — and absolutely nothing else.

Framework: MoSCoW Prioritisation

- Must Have — The single feature without which the product fails to deliver its core promise
- Should Have — Important, but the MVP still functions without it
- Could Have — Nice-to-have, move to the backlog without guilt
- Won't Have (v1) — Write this list explicitly and protect it fiercely from scope creep

Rule: The One-Sentence MVP Test [Product] helps [ICP] do [core job] so they can [desired outcome]. If your MVP cannot be described in one clean sentence, it is already too big. Every feature you add multiplies build time and dilutes your learning signal.

Insight: The first version of Instagram was a location check-in app. YouTube started as a video dating site. Scope does not define destiny — early validation does.

Step 08 — Define Success Metrics Before You Build

If you do not know what success looks like before launch, you will rationalise any result after launch. Set your metrics first, then open your IDE.

Framework: North Star Metric + Guard Rails

- North Star Metric (NSM) — The one number that best captures value delivered to users



- Example NSMs: Weekly Active Users, Tasks Completed, Revenue per User, NPS score
- Guard Rails — 2–3 metrics that must not decline while you optimise for the NSM
- Set a 60-day post-launch target for each metric before writing a single line of code

Action: Pre-Mortem Exercise

- Imagine it is 6 months from now and your product has completely failed write down exactly why
 - Identify the top 3 failure modes and build early-warning metrics for each
 - This single 30-minute exercise prevents more failures than any feature list ever will
-

Final Word

Validate first. Build second. Succeed always.

The founders who win are not the ones who code the fastest they are the ones who learn the fastest. This blueprint is your permission to slow down before the sprint, so your sprint actually takes you somewhere worth going.