

# The Founder Prompt Library

## 50 consulting-grade prompts to start a business without an MBA

*The free companion to How to Start a Business Without an MBA.*

**How to use these.** Each prompt is a starting point, not a magic button. Feed it *your* real inputs — customer quotes, numbers, notes — and treat the first answer as a draft to push back on, not a verdict. The founders who get the most out of AI have a conversation with it. And never accept a number, a law, or a claim without checking it.

---

### 1 • Mindset & your unfair advantage

**1. Founder Asset Map.** “Here’s an inventory of my skills, experience, network, interests, credibility, and problems I know intimately: [paste]. Act as a startup advisor. Find the non-obvious overlaps. Propose three business directions where I’d have an unfair advantage because of what I already have. For each, name the likely customer, the pain, and how fast I could test demand for free. Rank by ease of validation, not size of opportunity.”

**2. Constraint reframe.** “I have [time/money/skills] and these constraints: [list]. Instead of telling me what I lack, list five business models that turn each of these constraints into an advantage, with a one-line example of someone doing it.”

### 2 • Find a painful problem

**3. Pain pattern finder.** “Here are 40 reviews and forum posts about [topic]: [paste]. Extract every distinct complaint or unmet need. Group into themes. For each theme, estimate frequency, intensity, and whether it’s expensive to leave unsolved. Quote exact customer words. Rank themes by business potential and explain the ranking.”

**4. Alternatives & customer language.** “From the same posts, list the alternatives people currently use to cope (including ‘manual workarounds’ and ‘nothing’). For each, name what it’s worst at, in their words. Then give me the 15 exact phrases customers repeat most about this problem.”

**5. Interview script (The Mom Test style).** “I want to interview someone about [problem] without pitching my idea. Give me a 6-question script that asks only about their past behavior — the last time it happened, what they did, what it cost them, what they’ve tried — and never mentions my solution.”

**6. Interview synthesis.** “Here are notes from 8 customer interviews: [paste]. Pull out the recurring pain, the language they use, what they’ve already tried, and where they’d realistically pay. Flag anything that sounds like polite interest rather than real intent to buy.”

**7. Opportunity Thesis.** “Turn this raw research into a one-page opportunity thesis: [paste]. Fields: target customer (a specific person), the painful problem in their words, frequency/intensity, what it costs them, current duct-tape solution, tentative solution, price hypothesis, and where to reach 25 of these people.”

### 3 • Choose a niche you can win

**8. Niche comparison.** “Here are three niches I’m considering: [describe]. Score each on pain, budget, frequency, urgency, competition, ease of access, my ability to deliver, and automatability (1–5, one-line reason). Flag any with a fatal weakness. Recommend one for a solo founder with no capital, and explain why.”

**9. Wedge check.** “My candidate niche is [niche]. Tell me honestly: is it a good wedge, or is it so narrow it lacks volume, repeat business, or room to expand? Suggest a slightly wider or narrower version if mine is off, and describe the adjacent markets I could grow into later.”

**10. ICP in plain words.** “For [niche], describe my ideal first customer without corporate jargon: who feels the pain most, who pays fastest, who’s easiest to reach, who can decide alone, and what event triggers them to look for a solution.”

**11. Where they gather.** “List the specific places online and offline where [ICP] already gathers, complains, and asks for recommendations — subreddits, groups, forums, newsletters, events, marketplaces. For each, note what kind of post gets engagement there.”

### 4 • Design an offer people can buy

**12. Vague-to-sharp promise.** “My rough idea is: [idea]. Rewrite it as a sharp promise that names the exact person, the pain in their words, and a concrete before-and-after with a number. Give me five versions, from safe to bold.”

**13. Minimum viable offer.** “Build my minimum viable offer for [customer] with [problem]. Fill in all nine parts, one line each: customer, pain, transformation, mechanism, deliverable, proof, price, guarantee, next step. Flag any part I can’t answer yet as ‘needs work.’”

**14. Offer rewrites (bad → strong).** “Here’s my offer: [paste]. Show me three rewrites — outcome-focused, speed-focused, and risk-reversal-focused — and explain in one line why each might convert better. Then pick the strongest and say why.”

**15. The ‘so what?’ test.** “Read each line of my offer and challenge it as a skeptical buyer would, asking ‘so what?’ until every line ends in a concrete benefit the customer cares about. Rewrite the weak lines.”

**16. Pricing options.** “For this offer: [paste], propose three simple price structures — a single price, a two-tier, and an honest founder rate — with a one-line rationale each and the type of customer each attracts. Price on the value of solving the problem, not my costs.”

## 5 • Validate before you build

**17. Landing copy from real words.** “Here are 15 quotes from [customer] about [problem]: [paste]. Write a one-page landing that pre-sells [offer]: a headline naming the pain, three benefit bullets, a short ‘how it works,’ one honest line of proof, and one clear CTA. Use their exact phrases. No hype, no buzzwords, no exclamation marks.”

**18. Smoke-test plan.** “Design a 48-hour validation test for [offer] aimed at [customer]. Give me the cheapest experiment that produces a costly signal, the exact bar to pass (e.g., 3 of 10), where to run it, and the one metric that decides continue/pivot/kill.”

**19. Read the result.** “My test produced: [numbers/notes]. Tell me honestly whether that’s a real signal or wishful thinking, and if it failed, which single variable to change first — channel, promise, price, pain, customer, or mechanism — and why.”

**20. Objection radar.** “Here are the replies and objections from my test: [paste]. Group them, tell me which are real blockers vs polite excuses, and what each one tells me to fix in the offer, the price, or the targeting.”

## 6 • Build the smallest MVP

**21. Riskiest-assumption finder.** “Here’s my offer and customer: [paste]. Name the single riskiest assumption I should test first, and the lightest possible MVP to test it — prefer a concierge or no-code approach over building software.”

**22. MVP one-page spec.** “Write a one-page MVP spec: user, problem, core flow (start to outcome), inputs, outputs, the riskiest assumption, delivery route (concierge/manual/no-code/AI-coded), a success metric with a threshold, and what I’m deliberately NOT building yet.”

**23. Scope-creep guard.** “Here’s what I plan to build: [paste]. Cut it to the minimum that still tests my main hypothesis. List everything to remove from version one and why, and be blunt about anything that’s ego, not learning.”

**24. Weekly build sprint.** “Help me plan a 7-day build sprint for [MVP]. One hypothesis for the week, the smallest thing that tests it, who I’ll put it in front of by Thursday, and the single number that would change my mind by Friday.”

## 7 • Build it with no-code / AI

**25. Product build spec.** “Build [type of product]. Context: it’s for [user] who needs to [outcome]. User flow: 1)... 2)... 3).... Inputs: [list]. Outputs: [list]. Must NOT include: [extras]. Keep the design plain. Done when: [criteria]. Build the simplest version and tell me your assumptions.”

**26. Tool chooser.** “For an MVP that needs to [describe what it does], recommend the lowest-rung tool that works — landing page, form + spreadsheet, no-code app (Airtable/Softr/Glide/Bubble), or AI-assisted code (Cursor/Replit) — and warn me if payments or sensitive data mean I should test extra carefully.”

**27. Minimum QA checklist.** “I built [product] with no-code/AI. Give me a short QA checklist tailored to it: the full flow, payments, forms, confirmation emails, error cases, mobile view, and privacy. Flag the two places most likely to silently break.”

## 8 • Zero-budget sales

**28. Prospect list plan.** “For [ICP], give me a step-by-step plan to build a list of 50 real prospects for free, naming the exact sources (communities, LinkedIn searches, reviews, directories) and one identifying detail to capture for each so I can personalize.”

**29. Personalized outreach.** “My offer: [paste]. Prospect: [role, company, one public detail]. Write a cold message under 90 words that opens with the specific detail, names their likely problem, and ends with one low-friction ask. Plain and human. No flattery, no ‘I hope this finds you well.’”

**30. Five-touch sequence.** “Write a 5-touch outreach sequence over two weeks for [offer] to [ICP]: an opener, a value touch with no ask, a gentle nudge, a new-angle message, and a polite break-up. Keep each short, personal, and never pushy.”

**31. Follow-up after silence.** “A prospect went quiet after showing interest. Write three short, non-desperate follow-up options, each with a different angle, that make it easy for them to re-engage or to say no cleanly.”

**32. Objection responses.** “Here are the objections I keep hearing: [list]. For each, write an honest, non-manipulative response that respects the concern and reconnects it to the cost of their problem — not a trick to ‘overcome’ them.”

## 9 • Discovery calls & closing

**33. Discovery script.** “Give me a 7-question discovery script for [offer] and [customer] that uncovers their situation, the problem, its cost, what they’ve tried, urgency, budget, and the decision-maker — without being pushy. Remind me to listen twice as much as I talk.”

**34. Roleplay a skeptical buyer.** “Roleplay as a skeptical version of [customer] and run a discovery call with me, raising realistic objections one at a time so I can practice honest responses. Afterward, tell me where I talked too much or pitched too early.”

**35. Pilot-close proposal.** “Based on this conversation summary: [paste], draft a simple pilot-close: confirm the diagnosis in their words, propose a small paid pilot (scope, price, timeframe), reverse the risk, and name a start date.”

**36. Interests, not positions.** “A customer’s stated position is ‘[e.g., it’s too expensive]’. List the real interests that might sit behind it (budget timing, fear of choosing wrong, internal approval) and, for each, an option that meets the interest without simply discounting.”

## 10 • Content, authority & SEO

**37. One input, many pieces.** “Here’s a call transcript / list of objections: [paste]. Pull the 5 most useful insights for [audience]. For each, draft a short personal social post, one email paragraph, and one SEO article title with a three-line outline. Use the customer’s language.”

**38. Long-tail SEO finder.** “For [niche], list 20 specific, low-competition questions my customers type into Google when they’re close to buying. For each, give the searcher’s intent and a one-line angle that would answer it better than what’s out there.”

**39. Content-to-asset audit.** “Here are my last 10 content ideas: [paste]. For each, tell me its job (educate / capture demand / answer objection / show proof / start conversation) and cut any that don’t clearly do one.”

**40. Founder-led post.** “Turn this real learning from my week — ‘[insight]’ — into a short, honest, founder-voice post that shares what I learned and what didn’t work, with zero hype and a genuine question at the end.”

## 11 • Minimum viable brand & landing

**41. Landing copy that doesn’t sound like AI.** “Here are 15 exact phrases my customers used about [problem]: [paste]. Write landing copy: pain headline, promise, one-line mechanism, proof, offer, three objections answered, one CTA. Use their phrases. Plain,

specific, confident. Ban buzzwords and exclamation marks. Then give me three alternative headlines.”

**42. Proof without customers.** “I have no testimonials yet, but I have [pilot result / my own story / a demo / a guarantee]. Write the proof section of my landing page using only what I actually have, in a way that builds trust honestly.”

**43. Message before design.** “Draft my landing page as plain text following this structure: pain, promise, mechanism, proof, offer, top three objections, and one CTA. Design comes later — make the words carry it.”

## 12 • Finance for founders who hate finance

**44. Pocket CFO.** “Here’s a CSV of my sales and my monthly expenses: [paste]. Give me a simplified P&L, my runway (cash [amount] ÷ burn), gross margin per sale, the most dangerous number and why, and two cheap actions to improve it. Plain language — assume I hate finance.”

**45. Unit economics on a napkin.** “For a product priced at [price] with variable cost [cost], help me estimate gross margin, allowable CAC, a rough LTV if a customer stays [months], payback, and break-even. Tell me in one line whether the unit works.”

**46. Pricing stress test.** “Run three scenarios for my business at prices [A/B/C]: how each changes margin, break-even, and how many customers I’d need per month to survive. Recommend the one that balances survival and growth.”

## 13 • Legal & minimum viable bureaucracy

**47. Legal prep assistant.** “I’m starting a [type of business] in [country/region], serving [customers]. Act as a preparation assistant (not a lawyer). List the documents and registrations I likely need now vs can defer, draft a plain client agreement and scope-of-work, and give me the questions to ask a local professional. Flag anything that needs professional review.”

**48. Risk triage.** “Describe my business: [paste]. Tell me honestly whether it’s low, medium, or high legal/financial risk, what’s usually enough to start at that level, and when I should stop and get professional advice before launching.”

## 14 • Automate & run the machine

**49. Process-to-automation.** “Here are the repetitive tasks in my week: [list]. Rank them by how often I do them and how much I dread them. For the top one, design a simple

automation (trigger → actions → tools → what could break → how to test → upkeep).”

**50. Task-to-SOP.** “I’ll describe how I do [task] in messy detail: [describe]. Turn it into a clear standard operating procedure someone else could follow: numbered steps, tools used, what ‘done’ looks like, and common mistakes to avoid. Concise and practical.”

---

*Copy the constraint, not the company. Now go build.*