Test your startup idea in Claude

Here's how to set up startup-skill, my free set of Claude skills from the video, and run your first idea through it: a board of eight advisors, your real competitors, 100 simulated buyers, where your customers will come from, the numbers, and a business plan with one verdict at the end. You won't write any code. With 20 buyers, my first run took about 20 minutes from idea to verdict. That was before I added the competitors and marketing steps, so give it longer.

What you need

- **A paid Claude plan.** startup-skill is free, but it runs in Claude Code, which comes with Pro, Max, Ultra, Team and Enterprise. It isn't in the free plan.
- **A Mac or a Windows computer.**
- **Python 3.8 or newer.** startup-skill's small tools use it, and nothing else. Not sure you have it?

Claude checks during the install.

- **Your idea in a few lines:** what it is, who it's for, what you'd sell and at what price.
- **Rough numbers, for steps 7 and 8:** how you'd find customers and how many hours a month you can give it, what one sale or one customer costs you to serve, your monthly costs (rent, software, and what you'll pay yourself), and what you'd spend before opening. Guesses are fine. Every guess is marked as one, in a list of numbers to replace with real quotes.

Using Codex instead? The steps are at the end of this guide.

---

1. Open Claude Code, no terminal

1. Download the Claude desktop app from claude.ai/download, install it and sign in.
2. Click **Code** at the top of the sidebar.
3. Make a new folder for your idea, for example "Startup test" in your Documents. Everything startup-skill writes goes in a folder called `startup` inside it.

4 In the row under the message box, choose **Local**, then click the folder control, choose **Open folder...** and pick that folder.

5 In the same row, check the mode says **Manual**. In Manual, Claude asks before it changes a file or runs anything.

---

2. Install startup-skill

1 Paste this into the message box and press Enter:

```
https://github.com/psabbagh/startup-skill

install skill
```

2 Claude reads the page and copies the seven skills into your personal skills folder, so they work in every project. It asks before each step: read what it wants to run, then allow it. It ends by checking your Python version.

3 If Claude says Python is missing, or older than 3.8, paste this:

```
Show me how to install Python 3, one step
at a time.
```

4 Type `/startup` in the message box. Seven skills show up in the list: `/startup-skill`, `/startup-board`, `/startup-competitors`, `/startup-buyers`, `/startup-marketing`, `/startup-numbers` and `/startup-plan`. If they don't, start a new session and look again.

5 Switch the mode to **Accept edits**. startup-skill saves a lot of files (one per buyer alone), and in Manual you'd approve each one. Claude still asks before it searches the web and before it runs its small tools: allow them.

---

3. Give it your idea

Type `/startup-skill`, then your idea, then `--quick`:

```
/startup-skill [What it is, in one sentence.] It's for [who, as specific as you can]. I'd sell [what] for [price]. [How customers find it: a city or online, and the channel.] [What you bring: money, time, skills, anything already decided.] --quick
```

`--quick` (two hyphens) runs 20 buyers instead of 100: a faster, cheaper first look. Step 10 shows how

to run the full 100.

No idea handy? Try the example from startup-skill's own page:

```
/startup-skill a booking app for hair salons,
$29 a month --quick
```

Claude tells you the six steps it's about to run, then goes through them in order: the board, the competitors, the buyers, the marketing, the numbers, the plan. It stops to ask you for anything only you know.

---

4. The board

If the name, the customer, the product or the price is missing, Claude asks you (three questions at most). Then it writes your idea down in `startup/idea.md`.

Eight advisors read it, each a separate AI applying one published playbook:

- **Offer:** Alex Hormozi's *$100M Offers*
- **Leads:** Alex Hormozi's *$100M Leads*
- **Monopoly:** Peter Thiel's *Zero to One*

- **Marketing:** Seth Godin's *This Is Marketing* and *Purple Cow*
- **Leverage:** Naval Ravikant's *How to Get Rich* and *The Almanack of Naval Ravikant*
- **AI-native:** Tobi Lütke's 2025 memo to Shopify on using AI
- **Traction:** Lenny Rachitsky's newsletter
- **Startup:** Paul Graham's essays, like *How to Get Startup Ideas* and *Do Things That Don't Scale*

They apply those ideas. They aren't those people and never speak for them. Each one scores your idea from 1 to 10, names the three things most likely to kill it, names the one assumption it all rests on and the cheapest way to test it, and votes: **fund**, **fund if** (with conditions), or **pass**.

Then the chair writes `startup/board.md`: the vote and the average score, the risks more than one advisor raised, where they disagreed, and every condition as a checklist that says which step will test

it. It also answers five questions I wish someone had asked me on day one:

1. What one problem does it fix, and do these customers already spend time or money on it?
2. Are you about to build the expensive version first?
3. How do your first ten customers hear about it, and what does each one cost to win?
4. How many habits does a customer have to change to get the value?
5. Does the demand come from many customers, or from one or two big ones?

It ends with the strongest version of your business the board can see. That may not be the business you pitched. If it's different enough, Claude asks whether you want to change your idea before going on. A pass is a result, not a failure.

---

5. The competitors

Claude searches the web the way a buyer would and finds who already solves this problem. It aims for five to ten: the direct competitors, the other products

people use for the same job, and the do-it-yourself way, which is often the real competitor.

This is the one step built on real-world evidence instead of simulation. Every competitor, price and complaint links to the page it came from, with the date it was read. If Claude can't link it, it doesn't go in. Expect Claude to ask before it searches and before it opens each new site. Allow them. If it offers not to ask again, that's fine too.

Read `startup/competitors.md`:

- What buyers already pay to solve this today. Money already spent on the problem is the strongest sign they'd pay for a better answer.
- A table of who's out there: what each one charges, what that price includes, their own headline and their rating.
- Where each one sits on two things your buyer cares about, like price and effort.
- What their own customers complain about, most common first, with links.
- **The gap:** what nobody does well that you could. If there's no gap, it says so.

- **The copycat risk:** who could copy you fastest, and how.

All of this feeds the next steps. The buyers weigh your offer against these real alternatives and prices, the marketing step uses the gap as your position, and the numbers step checks your price against theirs.

---

6. The buyers

1. **Check your customer.** Claude describes your target customer in a few lines: the groups, what each already spends on this kind of problem, and what they do about it today, using what the competitors step found. Correct anything that's wrong before it goes on. A wrong customer makes the whole panel wrong.
2. **Check your pitch.** It writes the three to six lines a buyer would see: what you sell, the price, how to get it, and the first step you'd offer, like a free trial, a money-back guarantee or a founding-member price. Keep only a first step you'd really give. Every buyer sees exactly this.

3 **Let it run.** Each buyer is a separate AI playing one person drawn from your customer, with their own income, budget, shopping habits, what they use today and how much the problem really bothers them. Nobody tells a buyer what to object to. Claude runs them ten at a time. My 20 took under three minutes.

Read `startup/buyers/results.md`:

- How many would buy, how many said probably, and how many passed. A buy means they'd pay now or, if your first step is a free trial, start it today and expect to pay after it.
- Who buys and who doesn't: by customer group, by how much the problem bothers them, by what they use today, by how they shop, and by income.
- Why the others don't buy, in their own words, and what they'll do instead.
- What would change their mind.
- The price range they'd accept, and where your price sits in it.

Claude also points out the doubts nobody expected: the ones buyers raised that weren't on your list.

That’s often the most useful line in the whole panel.

If few would buy, Claude can turn their objections into a better offer. It sorts what they asked for into proof, risk, price, effort or a missing answer, suggests a change for the top two or three, and asks whether you’d really do each one. Then the same buyers see the new pitch, and you get a before and after. Two rounds at most, because after that you’re teaching the pitch to please simulated buyers. A re-test uses as much of your plan as the first run. If the main reason is “no need” or “happy with what I use today”, Claude says so: that’s the market, not the pitch.

Read it honestly. These are simulated buyers, not customers. They’re good at finding the doubts, the groups and the words you hadn’t thought of, and bad at predicting real sales. AI leans agreeable, so treat the buy rate as the best case. Before you spend real money, check the big calls with real people. And never use what the buyers said as testimonials.

7. The marketing

Most forecasts break on one line: “and then customers arrive”. This step swaps that line for a plan you can count.

1. **Your first audience.** Claude takes the group that bought most in the panel, names it as narrowly as it can, and offers two or three one-line positions, with the one it recommends.
2. **Your channels.** It picks two or three ways to reach them, from people you know, direct outreach, content, paid ads, partners and referrals, and marketplaces and listings. It also says which ones it’s leaving out and why.
3. **What you’ll really do.** For each channel it asks what you’ll actually do each month: how many people you reach, the share that makes it through each step from reached to paid, what you spend and your hours. Every rate needs a source: your own results, a public benchmark with a link, or an estimate marked as one.

A small tool turns that into new customers per month, the cost to win one, and your hours. It warns you

when a channel converts better than the buyers' best case, when the plan takes more hours than a full-time job, or when month 12 brings almost no new customers.

Read `startup/marketing.md`:

- Your first audience and your one-line position, with the evidence.
- The channels, and the ones you're not using.
- The first 30 days, week by week: how you win the first customers by hand.
- Five messages, each answering one of the buyers' top objections.
- The forecast, and the three numbers to watch every week.

If no buyer would buy, Claude says so and offers to improve the pitch first, because no channel fixes an offer nobody wants.

8. The numbers

Claude asks you, in one message, for what only you know: your price, what you'll pay yourself, and any

rent or salaries you've been quoted. It checks your price against your competitors', works out whether you sell one-off or by subscription (customers who pay every month, and some who cancel), and fills in your new customers and what each costs to win from your marketing plan. It shows you the arithmetic behind your sales in month 1 and month 12. If your plan needs more buyers than the panel said would pay, it tells you.

Then it writes `startup/numbers.md`. At the top, a note from the CFO in plain words:

- What each sale or customer leaves you after its own costs, and what a customer is worth next to what it costs to win them.
- The one number that moves the result most.
- The cash you need before it pays for itself, and when that money comes back.
- Three ways to improve it, each with the number it changes.

Below the note: break-even, 24 months month by month, what happens if your price drops 10%, sales

fall 20% or costs rise 15%, and every estimate you should replace with a real quote.

9. The verdict and your plan

Last, the plan tool works out the verdict from your numbers and your buyers. Nobody writes it by hand.

- **Profitable:** every check passes. Each sale, or each customer each month, leaves money after its own costs. For a subscription, a customer brings in more than it costs to win them. By month 12, you make money every month. The money you put in comes back within 24 months. Break-even fits what you can handle, if you gave a limit. At least 20% of the buyers would buy (pay now, or start your free trial and expect to pay after it). The buyers saw the same price the numbers use. And if you made a marketing plan, the numbers use its forecast.
- **Not yet:** at least one check fails, and it shows you the number that failed.
- **Incomplete:** the numbers or the buyers are missing.

It also warns you when a Profitable plan wouldn't survive 20% fewer sales.

In your `startup` folder:

- `plan.md`: the full business plan. The verdict and every check at the top, then the key numbers, the summary, the board, the competition, the buyers, how customers will find it, the numbers, the real-world test, and what isn't done yet.
- `summary.md`: half a page. The verdict and the three numbers behind it, where the board and the buyers agreed and disagreed, the biggest risk, what it takes to start, and the one thing to do this week. When the verdict is Not yet, it starts with what has to change.
- `real-test.md`: the cheapest way to find out whether real people pay, before the big money goes out, like a landing page with pre-orders, three paying pilot clients, or a founding-member offer. It sets the success line before you start (for example, 30 paid pre-orders in 14 days at $29) and gives you five questions to ask ten real customers.

Claude ends with the verdict in one line, the biggest problem, and the one thing to do this week. My first test, an AI receptionist for plumbers and electricians, came back Not yet: none of the 20 buyers would pay now. That's what it's for.

---

10. Go further

- **Run the full 100.** Make a new folder, open it in Claude Code (step 1), and run your idea again without `--quick`. That's 100 separate AI runs, so it takes longer and uses a large share of your plan's usage. Claude may offer to run the buyers on a smaller, cheaper model to save usage. That's how I ran my 100: they took about 11 minutes, and the whole run, before I added the competitors and marketing steps, about 26.
- **Run one step on its own:** `/startup-board`, `/startup-competitors`, `/startup-buyers`, `/startup-marketing`, `/startup-numbers` or `/startup-plan`. Each one reads what the others wrote in your `startup` folder.

Stay safe

- startup-skill is free and open source. Its five small tools are plain Python and never connect to the internet. The competitors step is the only one that goes online: Claude searches public pages, and asks you first.
- Stay in Manual while you install, and read what Claude wants to run before you allow it.
- The plan is only as good as the numbers you give it. Get real quotes, and have an accountant check the money, tax and legal parts. startup-skill isn’t financial, legal or tax advice, and says so.

Using Codex instead of Claude Code

startup-skill works in Codex too: its GitHub page has Codex install steps. You’ll need a ChatGPT account, the ChatGPT desktop app and Python 3.8 or newer. Codex loads all seven skills and starts the main one

when you type its name, but I haven't run a whole idea through it there, so treat this as a first try.

1. Download the ChatGPT desktop app from chatgpt.com/download, install it and sign in.
2. Choose **Codex** from the ChatGPT dropdown, then open a new folder for your idea.
3. Leave the permissions control below the message box on **Ask for approval** while you install. Codex asks before it uses the internet or changes files outside your folder: read each request, then allow it.
4. Paste this into the message box and press Enter:

   ```
   https://github.com/psabbagh/startup-skill

   Install these skills for me in Codex,
   following the "For Codex: how to install"
   steps in the README.
   ```

   It copies the seven skills into Codex's skills folder and checks your Python version. If Python is missing or older than 3.8, ask it to show you how to install Python 3, one step at a time.

5 Start a new chat in the same folder, so the skills load.

6 In the permissions control, choose **Approve for me**. startup-skill saves a lot of files, and Codex still asks about anything it sees as unsafe.

Then follow steps 3 to 10 above, typing `$` where the guide says `/`:

```
$startup-skill a booking app for hair salons,
$29 a month --quick
```

Always type the name with `$`. Asking in plain words won't start the main skill, the board or the buyers. If the competitors step says it can't search the web, open **Settings**, then **Configuration**, and set **Web search** to **Live**.

Checked against startup-skill and Anthropic's documentation on 8 November 2026. The Codex steps were added on 10 November 2026 from startup-skill's own page and the ChatGPT app. Claude changes fast: if a menu has moved, look for the same words.