

ORIGIN

AI, HONESTLY · THE PLAYBOOKS

Every setup from the series, written up so you can do it yourself.

Each episode of AI, Honestly shows one thing I actually run my business on. This is the how behind them: the steps, the prompts, and the skills you can copy and paste.

The after-call system

vidIQ + Claude

How Claude edits my videos

Train Claude to write like you

The skills starter kit

What a real build takes

Build your website with Claude

Zack Petrocca

Founder, Origin · The right hand most owners never had.

joinoriginos.com/ai-honestly

START HERE

Pick one. Put it to work this week.

You do not need all of them. You need the one that takes the most off your plate right now. Set it up, run it for a week, then come back for the next.

What is inside

- 1 **The after-call system.** The skill that grades the call, logs every promise, writes the follow-up and updates the CRM. *Episode 2, comment CLAUDE.*
- 2 **vidIQ + Claude.** The 30-second setup and the five prompts I use to pick every video before I shoot it. *Episode 3, comment VIDIQ.*
- 3 **How Claude edits my videos.** The honest setup behind the series, and the edit rules that keep every video on brand. *Episode 4, comment VIDEO.*
- 4 **Train Claude to write like you.** Your voice, your company and your rules in one place, read before it writes anything. *Episode 5, comment VOICE.*
- 5 **The skills starter kit.** The prompt that makes Claude interview you and write the skill, plus three to start with. *Episode 8, comment SKILL.*
- 6 **What a real build takes.** A buyer's guide to custom software: what AI made easy, the eight parts it did not, and how to tell if you are ready. *Episode 9, comment BUILD.*
- 7 **Build your website with Claude.** The honest process behind a 72-page site: the words first, one page at a time, and the phone test that makes it feel like you. *Episode 17, comment WEBSITE.*

What you need for most of these

A paid Claude plan (Pro or higher), so you can use connectors and skills. Some playbooks also use Claude Code on your computer. Each one says what else it needs.

The one rule under all of it: hand AI the task, never the judgment. It does the work. You decide what is true, what is good, and what goes out.

The after-call system

Granola takes the notes and it is connected to my Claude. When the call ends, I run one skill, and by the time I have hung up the follow-up is basically done.

TIME About 20 minutes

YOU NEED A paid Claude plan (Pro or higher) · Granola, or any notetaker that gives you a transcript · Your CRM and task manager (optional)

What you end up with

You hang up, type three words, and before your coffee is cold you have the call graded, the follow-up email drafted in your voice, every promise you made written down word for word, the CRM record current, and the tasks already in your list with the priorities set. You read it and you send it. That part stays yours.

Honest version: this runs on a note taker that was actually in the call. If nothing recorded it, the skill has nothing to read. Setting it up takes about half an hour, and most of that is connecting things.

What you need

- A paid Claude plan, so you can use connectors and skills.
- A note taker in your calls. I use Granola. Anything that gives you a transcript works.
- Your CRM and your task manager, connected if Claude supports them. Anything it cannot reach, the skill writes out for you to paste.
- Two or three follow-up emails you actually sent and were happy with.

What the skill does, in order

- 1 **Grades the call** like a sales psychologist: what I did well, what I missed, where they are in buying, and any pain point I did not dig into.
- 2 **Builds the to-do list** from what was actually said.
- 3 **Does what it can on its own:** the research, a first draft of any document, two times for the next meeting if we did not set one.
- 4 **Logs every promise I made**, word for word, so nothing I said I would send gets forgotten.
- 5 **Writes the follow-up email** and drafts any attachment I promised.
- 6 **Writes the CRM summary**, so their record is current as of this call.
- 7 **Puts every task in my task manager**, with the priority set.

I still read and send the email myself. It does the work. I make the calls.

1. Connect your note taker

Claude > Settings > Connectors > Granola > Connect

Test it before you build anything. Ask: "What were the main points from my last call?" If it cannot answer that, nothing after this step will work either, and you have found the problem in ten seconds instead of after your next call.

2. Connect your CRM and your task manager

Same place, if Claude supports them. Do not skip this step because your CRM is not on the list. A skill that writes the CRM summary out for you to paste is still faster than writing it yourself, and you can connect it properly later without touching the skill.

3. Create the skill

Copy this into a new skill, then change the parts in [brackets] to fit your business. The brackets are the whole job: this is my skill with my business taken out of it.

THE AFTER-CALL SKILL

Name: after-call

When to use: whenever I say "run after-call" or share notes or a transcript from a sales call.

Steps, in order:

1. Pull the notes and transcript for the call from Granola. If there is more than one recent call, ask me which one.
2. GRADE THE CALL. Act as a sales psychologist, not a cheerleader. Give me:
 - a score out of 10, and one line on why
 - two things I did well, quoting what I said
 - two things I could have done better, each with the line I could have used instead
 - where they are in buying: just curious, have a problem, want to solve it, ready to decide
 - any pain point they mentioned that I did not dig into
 - whether they said the problem and what it costs them in their own words
3. TO-DO LIST. Everything that needs to happen next, one line each, with who owns it.
4. DO WHAT YOU CAN. Anything on the list you can do now (research their company, draft a document), do it and show me. If no next meeting was set, suggest two times.
5. PROMISES. Every single thing I said I would send, do or find out, word for word.
6. FOLLOW-UP EMAIL. In my voice: [short, warm, direct, no filler]. Recap what they want, what I am sending, and the one next step. If I promised an attachment, draft it. Never send it.
7. CRM SUMMARY for [my CRM]: who they are, what they need, where they are in buying, the next step and when. Update the contact if it is connected; if not, write it out for me.
8. TASKS into [my task manager]: high if it is a promise or has a date, medium if it moves the deal, low for everything else.

Rules:

- Only use what was actually said on the call. Never invent a detail, a number, a price or a promise.
- If something is unclear, say so and ask me.
- Nothing goes to the client without me reading it first.

4. Teach it your email before you trust it

The email is the part people notice, so give it something real to copy instead of hoping. Paste two or three follow-ups you actually sent into the skill, under a heading that says what good looks like.

ADD THIS TO THE BOTTOM OF THE SKILL

What a good follow-up email looks like:

[paste a real one you sent]

[paste another]

Match the length, the opening line and the sign-off. Never open with "I hope this email finds you well" or "just circling back". One question per email, not three.

5. Run it after your next call

Hang up, open Claude, and type "run after-call". Read the grade first, before the email. The email saves you twenty minutes. The grade is the part that makes you better at the job.

6. The part everyone skips: correcting it out loud

The first few runs will be wrong in small ways. Say what is wrong in plain English and then put the fix in the skill. Real notes I have given mine:

- "Grade me harder on asking questions. I want the line I should have used, not a compliment."
- "My emails are half that length. Three to five sentences."
- "I never say reach out. I say call me or send it over."
- "You listed a promise I did not make. Only things I said word for word."
- "Stop guessing their budget. If they did not say a number, say they did not say a number."

Every note you give it, add to the skill. After about five calls the first draft stops needing edits, and the grade starts telling you the same thing twice, which is how you find out what you actually need to work on.

What goes wrong, and the fix

- It summarizes the wrong call. Name the call and the day. If two calls ran back to back, say which one.
- It invents a price, a date or a headcount. Put "never invent a number, a price or a date, ask me" in the rules and it stops doing it.
- The grade is all praise. Tell it to grade like someone who wants you to get better, and to quote the line you should have used instead. A 10 out of 10 on a call you lost is useless.
- The email sounds like a consultant. It has no example to copy. Paste two real ones in.
- Tasks land with no priority or no date. Tell it what high, medium and low mean in your business, in one line each.
- Nothing comes back at all. The transcript never synced. Open the note taker first and check the call is there.

Before you send

- Read the email out loud. If you would not say it, do not send it.
- Check every promise on the list is one you actually made.
- Check every number against what was said on the call, not against what sounds right.
- Send it yourself. Never give it permission to send.

The twenty minutes of admin is the small win. The real one is that you get graded on every call you take, by something that was in the room and does not care about your feelings.

PLAYBOOK · COMMENT VIDIQ

vidIQ + Claude

My first two scripts for this series were forecast at 17 to 700 views. Nobody was searching for them. Now I check every idea against real search numbers first. That is how I picked most of the videos in the series.

TIME About 30 seconds to connect

YOU NEED A paid Claude plan (Pro or higher) · A vidIQ account (some tools need a paid plan)

What you end up with

You stop guessing what to make. You ask a question in plain English and get back what people are actually searching for this month, whether anyone is covering it, what is working right now for channels your size, and a title built from the words people type instead of the words you like.

Honest version: this is a research tool, not a growth tool. It tells you what to make and what to call it. It does not make you good on camera and it does not write your hook. Those are still yours.

What you need

- A paid Claude plan, so you can use connectors.
- A vidIQ account. The free one answers most of what is below.
- A channel, or an honest idea of the one you are about to start.

1. Connect it

Claude > Settings > Connectors > vidIQ > Connect > Sign in

Test it in a new chat before you plan anything: "How many people search for [something in your world] on YouTube each month?" If a number comes back, you are connected.

2. The five prompts I actually use

FIND TOPICS WORTH MAKING

Find me topics in [your industry] with rising searches and low competition. Show me monthly search volume, the trend, and the competition for each.

CHECK AN IDEA BEFORE YOU SHOOT IT

I am planning a video called "[your title]". How many people search for this? Is there a version of this title more people are searching for?

FIND WHAT IS WORKING FOR SMALL CHANNELS

Find Shorts about [topic] from channels under 100,000 subscribers that got far more views than usual. What do the hooks have in common?

READ A VIDEO THAT WENT BIG

Pull the transcript of [video link]. Break down how the first 5 seconds hook people, and how the video is structured.

TITLE IT LIKE A SEARCH

Give me 5 titles for this video that match what people actually type into YouTube. Here is the script: [paste it].

3. Turn the answers into a week you can actually shoot

A list of topics is not a plan. This is the prompt that turns the research into things you can film in one sitting.

THE PLANNING PROMPT

Here are the topics you found. I can film for [two hours] this week.

Pick the [five] worth making, and for each one give me:

- the search people are typing, and the monthly volume
- the title, in their words
- the first line I say out loud, which is not the title
- the one point the video makes, in a sentence
- what I have to show on screen for it to be believable

Only pick topics I can speak to from experience: [one line on what you actually do]. If a topic has big volume but I have no real story for it, leave it out and say why.

4. Take one video that went big apart

This is worth more than any list of topics. Find a video in your world that did far better than the channel normally does, and study why instead of guessing.

THE TEARDOWN PROMPT

Pull the transcript of [video link] and the stats for that channel.

Tell me:

- how many views this got compared to what that channel usually gets
- the first 5 seconds, word for word, and what it is doing: a question, a number, a claim, a face
- how the video is built, section by section, with roughly how long each part runs
- where it would have lost me, if anywhere
- what I could steal about the structure without copying the content

Then write the version of the first 5 seconds that would work for [what you do], in my words, not theirs.

The rule I work by

The search sets the title. The hook on screen is the human line. "How to Connect vidIQ to Claude" is the title, because that is what people type. "176,000 people search this every month" is what you hear in the first second, because nobody says a title out loud.

What goes wrong, and the fix

- You chase the biggest number. Volume with a wall of established channels in front of it is worth less than a smaller number nobody is covering. Ask for both and read them together.
- It hands you topics you cannot speak to. Tell it what you actually do and give it permission to throw topics out.
- The titles come back as clickbait. Say it plainly: the title has to be true after they watch it. A title that beats the video costs you the next one.
- It quotes a number and you repeat it on camera. Search volumes are estimates. Say "about" or do not say it at all.
- Every idea starts sounding the same. Feed it your last ten videos and ask what you keep making, then ask for the one you have never made.

Before you shoot

- Say the first line out loud. If it sounds like a title, rewrite it.
- Check you have the proof. If the point needs a number or a screen, have it ready before you hit record.
- Check the title against the search, not against your taste.

Search numbers are a direction, not a promise. The research tells you what to point the camera at. It cannot tell you whether what you say next is worth hearing.

PLAYBOOK · COMMENT VIDEO

How Claude edits my videos

I film on my phone. Claude does the edit. The honest version: it is not one click in the Claude app. It runs in Claude Code with an open-source video tool called HyperFrames, and setting it up takes an afternoon. After that, each video takes me a few messages.

TIME An afternoon to set up

YOU NEED A paid Claude plan · Claude Code (the Code tab in the Claude desktop app) · Node.js on your computer

What you end up with

You film on your phone and drop the clips in a folder. Claude listens to every word, cuts the dead air, times the captions to your voice, puts your hook on screen, adds the graphics, and renders a vertical video ready to post. You watch it and say what is wrong until it is right. That part stays yours.

Honest version: this is not one click in the Claude app. It runs in Claude Code, on your computer, with a free video tool called HyperFrames. Setting it up takes an afternoon. After that a video is a few messages.

What you need

- A paid Claude plan, so you can use Claude Code.
- Claude Code: the Code tab in the Claude desktop app, or the terminal version from claude.com/claude-code.
- Your brand guide, if you have one, and 10 videos you wish yours looked like.

1. One prompt sets the whole thing up

Make an empty folder for your videos, open Claude Code in it, and paste this. It installs what it needs, builds the folders, and tells you if something is missing on your machine.

THE SETUP PROMPT

Set this folder up so you can edit my vertical talking-head videos.

Do all of this yourself and tell me in plain English when it is done:

1. Check I have Node.js. If I do not, walk me through installing it.
2. Install HyperFrames and confirm it runs.
3. Make these folders: footage (my clips straight off my phone), exports (finished videos), brand (my fonts, colors and rules).
4. Set a standing rule for yourself: phone video is often missing keyframes and stalls a render, so re-encode every clip into a clean copy before you edit it, and keep the audio at broadcast loudness.
5. Tell me exactly what to drop in each folder to get started.

Do not edit anything yet.

2. Teach it your taste

This is the part that makes the videos yours instead of generic, and it is a conversation, not a form. Put your brand guide in brand/ if you have one, then find 10 videos you wish yours looked like. They do not have to be ranked. What matters is that you say what you like about each one.

THE TASTE PROMPT

I am going to teach you how my videos should look.

In brand/ is my brand guide. Read it and pull out the fonts, the colors and anything about how we write.

I am also going to paste 10 links to videos I like. For each one I will say what I like about it: the captions, the pacing, the way the graphics come in, whatever it is.

Ask me questions until you understand the look. Push back if two things I said contradict each other.

Then write brand/edit-rules.md in my words: the frame, the captions, the hook, the graphics, the motion, and a Never list. Show it to me before you use it.

You end up with a file like this, in your language, not mine. Claude reads it before every edit, and you add a line to it every time you correct something.

WHAT BRAND/EDIT-RULES.MD ENDS UP LOOKING LIKE

How to edit my videos

The frame

- 1080 x 1920, vertical. Never crop my face out of the middle.
- Cut the silence before I start, and any pause longer than a quarter second.
- Only cut between sentences. Never stitch words inside a sentence.

Captions

- 1 to 4 words at a time, timed to how I actually speak.
- [your font], [your color]. Never over my face.
- One emphasis word per card, in [your accent color].

The hook

- On screen inside the first 1.5 seconds. One line, the way a person would say it.

Graphics

- When I name a tool, show its logo, not its name typed out.
- One style for every graphic: [describe it].
- Real numbers only. Never put anything on screen that is not true.

Motion

- Zoom only on the words I stress. Never on a timer.
- Something changes on screen at least every 2 seconds.

Never

- No stock footage. No fake screenshots. No music I did not pick.
- Nothing on screen that names a client.

3. Edit the first one

Drop a clip in footage and say this. Expect the first edit to be 80 percent right.

THE EDIT PROMPT

Edit the clip in footage/ into a vertical short.

Read brand/edit-rules.md first and follow it exactly.

The hook is: "[your first line]"

Then tell me three things: how much dead air you cut, how many captions there are, and anything you were unsure about. Do not post it anywhere. Put the file in exports/ and I will watch it.

4. The part everyone skips: the review loop

The first version is never the one. Say what is wrong in plain English and it fixes it. Real notes I have given it:

- "The text at the top is ugly. Mock up three other styles on my own frames and show me."
- "That zoom is off the beat. Only zoom on the word I stress."
- "You cut me off mid-sentence at 12 seconds. Only cut between sentences."

- "The caption is covering my mouth. Move it down."
- "It says clod. I said Claude. Fix it everywhere."

Every note you give it, add to your rules file. After a few videos it stops making that mistake, and the first version starts landing at 95 percent.

5. Then do a week at a time

THE BATCH PROMPT

Here are [number] clips in footage/. Edit all of them the same way, following brand/edit-rules.md.

For each one: the hook is the first sentence I say, unless I tell you otherwise.

When they are done, give me a list: file name, length, and anything you were unsure about. I will watch them before anything goes out.

What goes wrong, and the fix

- A render hangs partway. That is the keyframe problem. Remind it to re-encode every clip before editing.
- A word comes out wrong in the captions. Give it a fixes list: "clod" is "Claude", "AT" is "a T". It applies them every time.
- A file that looks finished is the old one. Have it check the file time and the last frames before it tells you a video is ready.
- It gets flowery. Tell it plainly: captions are what I said, word for word, not a rewrite.

Before you post

- Watch the whole thing once, end to end, with the sound on.
- Check the file was written after your last change.
- Check the last two seconds. That is where a bad render shows first.

None of this makes you a video editor. It makes the editing cheap enough that you post every day. You still have to know what good looks like. It does the work. You are the taste.

Train Claude to write like you

Out of the box, Claude is what everybody has, so it writes like everybody. Your clients can tell. The fix is not a better prompt. It is three things it does not have: your voice, your company and your rules.

TIME

About an hour

YOU NEED

A paid Claude plan (for Projects) · An hour to gather what you already have

What you end up with

One place you write from. You ask for an email, a post or a proposal and what comes back sounds like you on a good day, knows what you sell and what it costs, and never invents a fact about your business. You still read it before it goes anywhere, but you stop rewriting it from scratch.

Honest version: the quality of this is the quality of what you feed it. An afternoon of gathering real writing beats any prompt anyone will ever sell you. There is no shortcut, and the ones that promise one all sound the same.

What you need

- A paid Claude plan, so you can make a Project with its own knowledge.
- An hour to gather things you have already written. You are not writing anything new.
- The company documents you already have: what you sell, who it is for, what it costs.

1. Your voice

Collect real things you wrote and sent. Not your polished best, your real stuff: 10 to 15 emails, 10 to 15 texts, and a transcript or two of you on a call. The call transcript is the one people skip and it is the most valuable, because it is how you talk when you are not performing. Take out names, numbers and anything private. Save each group as its own document.

2. Your company

Gather what already exists. Do not write anything new for this. Vision, mission and purpose. What you sell, who it is for, what it costs. Your sales documents and how you answer the objections you hear every week. Your best posts and your website copy. Your brand guide.

3. Your rules

This is the short document that does most of the work. Write it as three lists, in your own words.

WHAT THE RULES DOCUMENT LOOKS LIKE

How I write

Do

- Lead with the point in the first line.
- Short sentences. One idea each.
- Say the real number or say nothing.
- Write like I am talking to one person I like.

Do not

- Do not open with a pleasantry.
- Do not ask three questions in one email.
- Do not explain something I have not been asked about.
- Do not use a word I would have to look up.

Never use

- "I hope this email finds you well"
- "circling back", "just checking in", "touching base"
- "leverage", "synergy", "utilize", "robust"
- [add yours here, the ones that make you cringe]

4. All of it in one place

Create a Claude Project called "[Your company] writing". Add every document to its knowledge, and paste this into its instructions.

PROJECT INSTRUCTIONS

You write for [company name]. Before you write anything, read:

- the voice doc that matches what you are writing (email, text or call)
- the company docs, for anything about what we do or sell
- the rules doc, every time

Write like me on my best day, never like a generic AI. Follow the do's and don'ts exactly. Never use a word from the never list. Never invent a fact, number, price or promise; if you do not know it, ask me. Everything you write is a draft. I read it before it goes anywhere.

5. Test it before you rely on it

Do not find out it is wrong on something that matters. Give it a job you have already done and compare.

THE TEST PROMPT

Write the email I would send to [a real situation you handled last month].

Then, separately, tell me which of my documents you used and which lines of the rules you were following.

I am going to compare this to the one I actually sent. Do not try to impress me. Try to sound like me.

Read them side by side. Where it is different, that difference is a line missing from your rules document, not a reason to give up on it.

6. Sharper every week

When something comes out wrong, do not just fix it and move on. Tell Claude why, in the words you would use, then add the line to your rules. Real corrections:

- "Too formal. I would never say reach out."
- "That is three questions. I ask one."
- "You made up a timeline. If I did not give you a date, do not use one."
- "That opening is a warm-up. Start at the second paragraph."
- "Nobody in my world calls it a solution. It is a system."

Every fix you write down makes every future draft better. Every fix you only say out loud gets made again next week.

What goes wrong, and the fix

- It sounds like a press release. You gave it your polished marketing copy and none of your real emails. Add the messy ones.
- It uses a phrase you hate. It is not in the never list yet. Add it, word for word.
- It invents a price or a timeline. Say it once in the instructions: if you do not know it, ask. It will ask.
- Everything comes out the same length. Tell it what you write long and what you write short, with an example of each.
- It is great at emails and bad at posts. Those are two voices. Give each its own document.

Before you send

- Read it out loud. That catches the AI rhythm faster than reading it on screen.
- Check every fact, number and price against something real.
- Cut the first sentence. It is usually a warm-up.

Mine goes one step further: each role in the business has its own skill, and each one reads the right documents before it writes. Start with one Project. That is most of the win, and it is one afternoon.

The skills starter kit

A skill is instructions Claude follows every single time, for one job. Think of it as your SOPs, finally used. The test: if you explain the same thing to Claude two times a week or more, make it a skill.

TIME About 15 minutes per skill

YOU NEED A paid Claude plan (Pro or higher)

What you end up with

A short set of instructions Claude reads before it does a job you do all the time, so you stop re-explaining yourself. You say four words and it runs the way you would run it, in the order you would run it, avoiding the mistakes you have already corrected once.

Honest version: a skill is not magic and it is not code. It is the written version of how you train a new hire. If you cannot explain the job out loud, the skill will not save you, and that is worth finding out.

What you need

- A paid Claude plan.
- One job you do at least weekly and always do the same way.
- One example of that job done well, that you can paste in.

Where skills live

Open Settings in Claude and find Skills (on some plans it sits under Capabilities). You can create one there, or simply ask Claude in any chat to build one with you.

1. Pick the right first skill

Not the most impressive one. The one you do most often and hate most. Three tests: you do it at least once a week, you do it roughly the same way every time, and you can tell in ten seconds whether the result is good. If a job fails the third test, it is not ready to be a skill.

2. The fastest way to write one: let Claude interview you

Open a new chat, turn on voice-to-text, and paste this. Answer out loud, like you are training a new hire, because you are. Talking gets you the real version; typing gets you the tidy one.

THE INTERVIEW PROMPT

I want to build a Claude skill for [the job, e.g. writing proposals]. Interview me one question at a time until you understand exactly how I do it: what triggers it, the steps in order, what good looks like, what I never want, and an example of one I was happy with. Then write the skill for me.

3. The template every good skill follows

THE SKILL TEMPLATE

Name: [short name]
When to use: [the words or situations that should trigger it]
Steps:
1. [first thing]
2. [next thing]
3. [...]
What good looks like: [paste an example you were happy with]
Never: [the mistakes you keep correcting]

Four parts, and the last two are the ones people leave out. A skill with no example guesses at your standard. A skill with no never list makes the same mistake every week.

4. Three to start with

EMAILS IN MY VOICE

Name: my-emails
When to use: whenever I ask you to write or reply to an email.
Steps:
1. Ask who it is to and what I want to happen, if I did not say.
2. Lead with the point in the first line.
3. Keep it to 3 to 5 short sentences unless I ask for more.
4. End with one clear next step.
What good looks like: [paste 2 or 3 emails you sent and liked]
Never: "I hope this email finds you well," "circling back," "just checking in," more than one question.

WEEKLY CONTENT PLAN

Name: content-plan
When to use: when I say "plan my content" or "what should I post this week."
Steps:
1. Ask what happened in the business this week: wins, client stories, questions people asked.
2. Turn the best three into post ideas, each with one clear point.
3. Write a hook for each that works in the first second.
4. Give me a simple posting schedule for the week.
What good looks like: [paste a post that did well]
Never: generic tips with no real example, anything I did not actually do.

FIRST DRAFT OF A PROPOSAL

Name: proposal-draft

When to use: when I share call notes and say "draft the proposal."

Steps:

1. Pull out their problem in their own words, and what it costs them.
2. Write the outcome they want before anything about us.
3. Lay out what we will do, in phases, with what they get at the end of each.
4. Leave the price as [PRICE] for me to fill in.

What good looks like: [paste a proposal that closed]

Never: invented numbers or timelines, jargon, more than two pages.

5. Test it on a job you have already done

Run the skill on last week's work, where you already know the answer, and compare. You are not checking whether it is impressive. You are checking whether it is you.

THE TEST PROMPT

Run [skill name] on this: [paste the real input from a job you already did].

Then tell me which step you were on when you made each choice, and anything in the skill that was unclear or that you had to guess at.

I have the real version. I am going to compare them.

The second half of that prompt is the useful half. Every place it had to guess is a line the skill is missing.

6. Correct it into the skill, not the chat

Fixing it in the chat fixes today. Fixing it in the skill fixes every week after. When you correct something, say why, then put the why in the Never list.

The never list is the most valuable part of the skill, and it is the only part you cannot write on day one. It gets written one mistake at a time.

What goes wrong, and the fix

- It never triggers. The "when to use" does not match the words you actually type. Put your real phrases in, including the lazy ones.
- It does two jobs badly. One job per skill. "Write emails and plan content" is two skills.
- It ignores a step. The steps are vague. "Research them" is not a step. "Look up their company site and their last three posts, and tell me what changed recently" is.
- The output is close but not yours. There is no example in it. One real example beats a page of instructions.
- It works once and then drifts. You have been correcting it in chat. Move the corrections into the skill.

Before you rely on it

- Run it on three real jobs, not one.
- Check it never invents a number, a name or a date. If it did once, it will again: put it in the Never list.
- Read the skill back as if you were the new hire. If a step needs you to already know something, write that down.

Skills are how a business stops living in one person's head. Every one you write is a job someone else could now do the way you do it, which is the same thing as being able to hire.

PLAYBOOK · COMMENT BUILD

What a real build takes

I am not a developer, and over 1,500 updates have gone into the software my company runs on, built with Claude. So here is the honest version. AI made writing code cheap. It did not make building software easy. This guide is what actually decides whether custom software works, drawn from our own build.

TIME Fifteen minutes to read

YOU NEED A process you are thinking about turning into software · Fifteen minutes for the readiness check

What AI changed, and what it did not

WHAT AI MADE EASY

- Writing the code
- A first version you can click through
- Screens that look finished
- Changes in minutes instead of weeks

WHAT IT DID NOT

- Knowing what to build
- Who is allowed to see what
- Moving years of your data in safely
- Making sure a change does not break what already worked

A demo that looks finished takes an afternoon now. Software a team can run a business on still takes judgment, testing and time. Most of the cost is in the part you cannot see.

The eight things underneath

Screens are what you see. These are what you are actually paying for, and every one of them came up in our own build.

01

Who can see what

Your salesperson should not see payroll. A client should see their project, not the next client's. Every screen, report and export has to respect that, and it has to be tested.

02

Your existing data

Years of spreadsheets, a CRM, email history. Moving it in without losing or duplicating a record is a project of its own.

03

The tools it talks to

Email, calendar, accounting, your CRM. Two-way connections fail quietly: a record updates in one place and not the other.

04

The ugly cases

The client with two addresses, the refund, the duplicate lead, the person who double-clicks. Software lives or dies on the cases nobody puts in a demo.

05

Security

Two-step sign-in, what happens when someone leaves, where passwords and keys are kept. The day you need it is the wrong day to add it.

06

Phones

Your team uses it in a car, a hallway, on a job site. If it only works on a laptop, it does not work.

07

When it breaks

Backups, alerts, and who fixes it at 9pm on a Friday. Everything breaks eventually. What matters is how fast you know and how fast you recover.

08

Every change after

Each update can quietly break something that worked last week. Checking for that is most of the ongoing work, and it never ends.

From our own build

Real things that happened while building Origin's software. AI wrote the code every time. It caught none of these.

THE FEATURE THAT VANISHED

One update quietly removed an entire feature, and every automatic check still passed, because the change took the feature and its tests out together. Nothing failed and nothing warned us. Now every change is checked against what already worked.

THE DOUBLE CHARGE

A button clicked twice in quick succession could start the same paid job twice. The fix was not a better button. It was a rule in the database that makes a second copy impossible.

THE FORM THAT SAID NO

A security check on our website quietly rejected every new signup. The pages looked fine. We caught it the same day because we watch the numbers, not just the screens.

THE FIRST SIGN-IN

On a brand new account, two setup steps could run at the same moment and create duplicate records, with an error on the very first screen a client saw. It only happened sometimes, which is the hardest kind of problem to find.

None of these show up in a demo. All of them show up in month two.

Are you ready for custom software?

Tick each one that is true today. Six or more, and a build is worth a serious conversation. Fewer, and the best first step is usually fixing the process, which is often faster and cheaper.

☐ I can describe the process step by step, and so can the person who does it.

☐ It happens every week, not every quarter.

- | | |
|---|--|
| ☐ We have tried off-the-shelf tools, and the gap is specific to how we work. | ☐ I know what it costs us today when this goes wrong. |
| ☐ I know who will own it inside the business once it exists. | ☐ The data it needs lives somewhere we can get to. |
| ☐ I can say in one sentence what done looks like. | ☐ We will test it on real work before we trust it. |

Seven questions to ask anyone who builds it

An agency, a freelancer, or someone on your team with Claude. The answers tell you whether they have done this before.

- 1 How will you move my existing data in, and how will I know nothing was lost?
- 2 How do you control who can see what, and how do you test it?
- 3 What happens when an update breaks something that used to work?
- 4 How do you test on a phone, with my real records?
- 5 When it goes down, how fast will I know, and who fixes it?
- 6 What does it cost to run and maintain each month after launch?
- 7 Can I change it later without you?

Bring a brief to the first conversation

Paste this into Claude before any first call. The one-page brief it writes saves hours in the meeting, including one with us.

THE BRIEF PROMPT

I am thinking about building software for my business to handle: [one sentence].

Interview me one question at a time: who will use it and on what device, how the work happens today step by step, where it breaks, what it costs when it goes wrong, what data it needs, and what done looks like.

Then write a one-page brief: who uses it, what it replaces, the steps it covers, what done looks like, what it must never get wrong, and what is out of scope. Plain language. Do not suggest technology.

PLAYBOOK · COMMENT WEBSITE

Build your website with Claude

I built my company's website with Claude. No designer, no developer. It is 72 pages now. The first version took one day, and it looked like every other AI website. Making it feel like us took weeks. This is the process, including the part that took the longest.

TIME A day for a first version. Weeks to make it yours.

YOU NEED A paid Claude plan · Claude Code (the Code tab in the Claude desktop app) · An hour to answer three questions about your business before anything gets built

What you end up with

A website that says what your business actually does, in your words, that you can change yourself any time. Not a template with your logo on it.

Honest version: the first version took me one day, and it looked like every other AI website. What made it work was the strategy underneath: who it is for, how the pages fit together, and every word on them.

1. Answer three questions before anything gets built

- 1 Who is it for?** One kind of customer, described the way you would describe them to a friend. Not "small businesses". Which ones, and what is going wrong for them.
- 2 What do you actually do?** In their words, not yours. What changes for them after working with you.
- 3 What should they do next?** One action. Book a call, take a quiz, buy the thing. Every page points there.

2. Start with this prompt

Open Claude Code in an empty folder and paste this. It writes the words with you before it designs anything, which is what keeps the site from looking generic.

THE FIRST PROMPT

I want to build a website for my business: [one sentence on what the business does].

Before you design or build anything, interview me one question at a time: who the site is for, what problem they have, what I actually do about it, what proof I have, and the one thing I want every visitor to do next.

Then write the words for the home page only, in plain language, the way I talk. Show me the words before you build the page.

3. Describe one page at a time

Never ask for the whole site. Describe each page like you are explaining it to a person who has never seen your business.

THE PAGE BRIEF

Build the [page name] page.

Who lands here: [who, and what they were looking for]

What they need to understand: [the two or three things, in order]

What they should do next: [the one action]

What it should feel like: [calm and expensive, friendly and quick, whatever is true for you]

Use only the words we agreed on. Never invent a number, a review or a client.

4. Review every page on your phone

Most of your visitors are on a phone, and most of what looks wrong only shows up there. Pull the page up, scroll it like a stranger, and say exactly what is wrong. Vague feedback gets vague fixes.

THE PHONE REVIEW

I am looking at the [page name] page on my phone. Here is what is wrong:

- [e.g. This looks cheap: the heading is too big and the buttons look like a template]
- [e.g. Too much text before I know what you do]
- [e.g. Move the proof up, it is buried]

Fix these, then tell me exactly what you changed.

What goes wrong, and the fix

- It looks like every other AI website. You skipped the words. Go back to the three questions.
- It invents a testimonial or a number. Put "never invent a number, a review or a client" in every brief.
- It looks great on your laptop and broken on a phone. Review on the phone first, every time.
- Every page says the same thing. Each page gets one job and one reader. Say which.

AI can build the website now. Knowing what you want it to say is still on you, and it is most of the work.

One new playbook a week, free.

Every playbook from the series lives at joinoriginos.com/ai-honestly, and new ones land there as the series goes on. The Origin Diagnostic names the part of your business to fix first, in about five minutes.

joinoriginos.com/start

© 2026 Origin Consulting Co. Free for your own business use. Do not resell or repackage. "Origin" and "Origin OS" are trademarks of Origin Consulting Co. Claude is a product of Anthropic; vidIQ, Granola and HyperFrames belong to their owners. Origin is not affiliated with or endorsed by any of them.