

How to Turn Visitors Into Customers

Landing Pages: Room 3, where the worldview you built finally cashes out into a customer

Contents

The third room	01
The brief is the bottleneck	02
Taste opens the door — evidence converts	03
Score the page	04
Measure the room, not the likes	05
A loop that learns	06
Resources	07



CHAPTER 1

The third room

Your content fills the first two rooms. The landing page is the third — the path, and the one asset most founders write once and never revisit.

CHAPTER 1

The third room

Every product lives in three rooms. **Room 1** is the worldview — the shift in the world that makes what you do necessary. **Room 2** is the stakes — who wins and who loses as that shift plays out. Your content lives in those two rooms: your posts, your videos, your newsletter walk a stranger through the worldview and earn the belief.

Room 3 is the path. It's where someone who already bought the worldview walks in to *act* — and Room 3 has an address. It's a URL. It's your landing page.

You can't open cold traffic in Room 3. A cold visitor who lands on “Start your free trial” and a feature list has been dropped into Room 3 with no Room 1 and no Room 2 — no worldview, no stakes, just an ask. It reads as marketing, which means it reads as noise. The page only works for someone the first two rooms already warmed up.

The landing page is the most-leveraged, least-iterated asset in the whole funnel. Founders pour months into the product and the content, then write the page in an afternoon and never touch it again. That's backwards. And it gets punished at scale:

Traffic exposes a broken page louder. More traffic does not fix a page that doesn't convert. It just sends more people through the leak. Fix the page first; then turn up the volume.

This playbook walks Room 3 the way the rest of the series walks the journey — write a **brief**, understand why **evidence beats taste**, **score** the page, **measure** the right thing, and run it as a **loop that learns** — and shows where LEANSpark feeds the page the context that converts, scores it, and puts a real **experiment** around it, with a goal the page can fail.

Audit your three rooms

The job: find out which room your funnel is missing before you touch the page.

▸ IN LEANSPARK — YOU TYPE

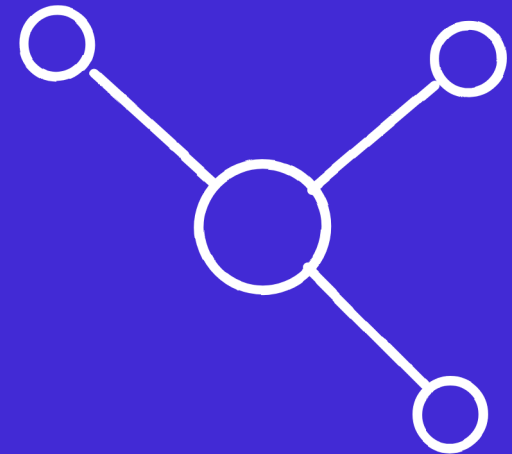
Audit my three rooms

Walks your funnel the way a stranger would — where the worldview gets built (Room 1), where the stakes get personal (Room 2), and what your page asks for (Room 3) — and flags every place you're asking a cold visitor for a Room 3 action.

RETURNS A room-by-room read of your funnel with the missing room named, so you know whether the page is the problem or the path to it is.

▸ VIA MCP — PROMPT ANY AGENT

"Here are my funnel links — homepage, content, landing page: [paste]. Read `business_model://current`, then map each link to the three rooms — worldview, stakes, path — and flag anywhere a cold visitor is asked for a Room 3 action before Rooms 1 and 2 have done their work."



CHAPTER 2

The brief is the bottleneck

Five questions, answered from validated evidence. A promise you can prove false in ninety days. Receipts, not adjectives.

CHAPTER 2

The brief is the bottleneck

A landing page is easy to *write* and hard to get *right*, and the difference is almost never the design. It's the brief. Before a word of copy, answer five questions — and answer them from evidence, not from your own head:

1. **Who is this for?** — a validated segment, not “everyone.”
2. **What problem are they here to solve?** — in their language, from your interviews.
3. **What promise does this page make?** — specific enough to be proven wrong.
4. **What proof makes the promise believable?** — receipts, not adjectives.
5. **What is the single CTA?** — one. Not three, not “or.”

Make the promise falsifiable. A falsifiable promise is one where, ninety days out, you can say plainly “it held” or “it broke.” “*Grow your business faster*” is not falsifiable — you can spin any quarter into either answer. That's not a promise, it's a vibe. “*Go from ten customers to fifty in ninety days*” is falsifiable. Either you did or you didn't.

And once the promise is specific, the proof has to match its shape. If the promise is “your team stops missing release deadlines,” the proof is receipts *on release deadlines* — a named customer, the releases-per-quarter before, the number after. Adjectives don't convert. Receipts do.

Draft the page from validated context

The job: turn the five-question brief into a first draft that starts from evidence, not a blank page.

► IN LEANSPARK — YOU TYPE

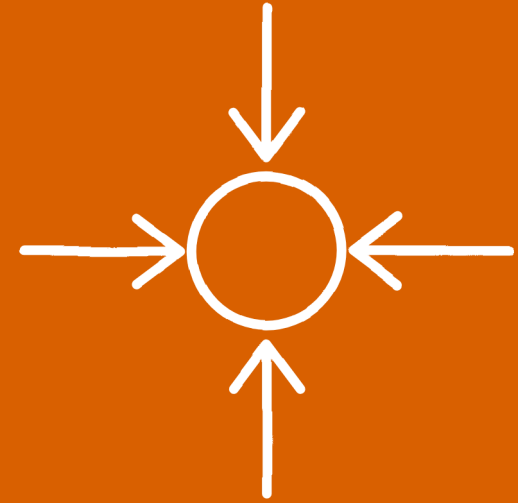
Draft my landing page

Drafts a starting page from your brief — hero, problem, proof, one CTA — in your own voice and pulled from your canvas, your interviews, and your proof, so the copy starts from validated context instead of a blank page.

RETURNS A page draft (hero / problem / solution / proof / CTA) in your voice, built around one falsifiable promise. You wire it up and ship it.

► VIA MCP — PROMPT ANY AGENT

"Pull my voice profile, offer, competitive landscape, segments, and recent interviews over MCP — and my design system, so the page wears my actual brand. Draft a landing page for my committed segment around a single falsifiable promise and one CTA. Pull the problem language from real interviews — quote the validated pain, don't invent it — and match the proof to the shape of the promise."



CHAPTER 3

Taste opens the door — evidence converts

Brand buys you a few points. Validated context buys the rest. One context engine feeds two surfaces.

CHAPTER 3

Taste opens the door — evidence converts

Here's the demo that settles the argument. I took a generic landing page — clean hero, “trusted by teams worldwide,” a few feature cards, a “Get Started” button — and ran it through a seven-dimension scorecard out of 100. It scored **43**. Poor.

Then I changed *nothing but the brand*. Same copy, word for word — real fonts, real color, my actual design system. It looked *gorgeous*, like a real company. New score: **46**. Three points. All that taste bought three points.

Then I changed the *copy* — a validated segment, the customer's own language, real proof matched to the promise — and kept the brand. **93**. Excellent.

The forty-seven-point gap is the whole lesson. Brand bought 3 points. The copy bought 47 more. “Pretty AI slop” is more dangerous than ugly slop, because it *looks* legit — so you stop looking for the real problem. Taste gets you in the door. Evidence is what converts. Structure and polish are table stakes; the context is the moat.

And here's where it compounds. The context that makes the page convert — your voice, your interviews, your proof, your positioning — is the *same* context that powers your content in Rooms 1 and 2. You don't maintain two sources of truth. You maintain one, and it feeds both surfaces.

One context engine, two surfaces. Your content is Rooms 1 and 2. Your landing page is Room 3. Both run off the same validated voice, the same library of interviews and proof, the same positioning. That shared engine is what LEANSpark is — and it's why the page and the posts sound like one company telling one true story.

Find your taste points and your evidence points

The job: find out how many of your page's points come from taste and how many from evidence.

► IN LEANSPARK — YOU TYPE

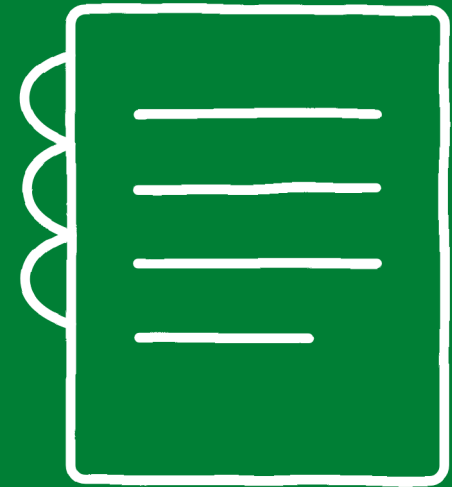
Where does my page get its points?

Reads your page and marks every line any company in your category could ship (taste and structure) versus lines only your validated evidence could produce (context) — each generic line is a slot where language from your real interviews should go.

RETURNS A taste-versus-evidence read of the page with the generic lines named, and a proposed replacement for each pulled from your interview language.

► VIA MCP — PROMPT ANY AGENT

"Here's my landing page: [paste the URL or the copy]. Mark every line that any company in my category could ship versus lines only my validated evidence could produce. For each generic line, propose the replacement — read my segments and interviews over MCP and quote the validated pain, don't invent it."



CHAPTER 4

Score the page

Seven dimensions, a hundred points, a gate on each. Stop guessing whether the page is good — measure it.

CHAPTER 4

Score the page

You can argue about a landing page forever. A scorecard ends the argument. The **Landing Page Assessment** reads your live page and scores seven dimensions, each out of ten, each with a gate it has to clear:

1. **Clarity & Positioning** — is it instantly obvious what this does and who it's for?
2. **Targeting** — is the audience specific, and does the language match it?
3. **Problem & Insight** — is the problem articulated with a real, earned insight?
4. **Unique Value** — is the value proposition differentiated and specific?
5. **Proof & Credibility** — are there outcomes, authority, and social proof?
6. **Offer & Objections** — is the CTA clear, the friction low, the objections handled?
7. **Story & Flow** — does the sequence carry a reader from top to bottom?

The seven roll up to a score out of 100, with bands — *excellent* (85+), *good* (70–84), *needs work* (50–69), *poor* (under 50). The gate matters as much as the total: a page can look fine and still fail on the one dimension that's quietly costing you the conversion.

A real example: the LEANSpark homepage scored 77. “Good” — all seven cleared the gate. But two dimensions only just did: **Targeting** (the segment was “founders,” too broad) and **Proof** (one anecdote, no stacked outcomes). That's the value of the gate — it doesn't let a strong average hide the two weak rooms. The fix list writes itself: name the segment, stack the proof.

Assess your live page

The job: replace the argument about whether the page is good with a number.

► IN LEANSPARK — YOU TYPE

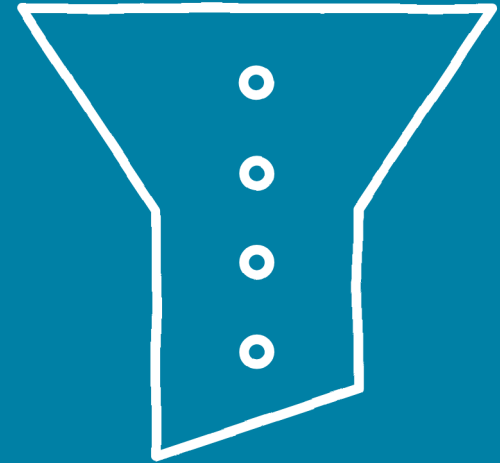
Assess my landing page

Fetches your live page and scores it across the seven dimensions, flags every one below the gate, and names the single dimension to fix first — so you spend your next cycle on the thing that's actually costing you conversion.

RETURNS A 0–100 score with a per-dimension breakdown and the weakest dimension named. You decide what to change.

► VIA MCP — PROMPT ANY AGENT

"Here's my landing page: [paste the URL or the copy]. Score it on the seven dimensions — clarity, targeting, problem, unique value, proof, offer, story — gate each at 7/10, and name the one dimension to fix first."



CHAPTER 5

Measure the room, not the likes

Five metrics, in order. The only one that ties the page to the business is qualified signup.

CHAPTER 5

Measure the room, not the likes

A score tells you if the page is *built* well. Live metrics tell you if it *works*. Watch five, in order — each one isolates a different failure:

1. **Sessions** — did anyone arrive? (Below ~500, don't read the signal too hard.)
2. **Above-the-fold dwell** — did they read the hero? (Three seconds is a floor; twelve means the hero landed.)
3. **Scroll depth** — did they keep going? (Where the curve drops tells you which room failed.)
4. **CTA click** — did they want what you offered? (Isolates page quality from whatever comes after.)
5. **Qualified signup** — did the *right* person sign up?

Qualified signup is the only metric that ties the page to the business. A page can convert 5% to signup and 0% to *qualified* signup — it's converting the wrong visitor, which is worse than a 1% page that only converts the right one. Vanity conversion feels like progress and isn't. Before you ship, wire the page with tracking tags so every signup carries where it came from; a number you can't trace back to a source can't teach you anything.

Wire the five metrics

The job: instrument the page so every number can be traced to a source.

► IN LEANSPARK — YOU TYPE

Set up my page metrics

Turns the five-metric ladder into a concrete tracking plan for your page — what to instrument for sessions, hero dwell, scroll depth, CTA click, and qualified signup — with your definition of “qualified” written down and a source tag on every path in.

RETURNS A tracking checklist you can implement in an afternoon, a one-sentence definition of qualified signup for your segment, and a baseline table to fill in before you change anything.

► VIA MCP — PROMPT ANY AGENT

"Read `business_model://current` . Define “qualified signup” for my committed segment in one sentence, then give me the tracking plan for the five metrics on my page — including the source tags each inbound path needs — and a baseline table to fill before I start iterating."



CHAPTER 6

A loop that learns

Change one variable per cycle. Plan, do, check, act. A loop that learns beats a line that runs.

CHAPTER 6

A loop that learns

A landing page is never finished; it's iterated. The trap is iterating without learning. If you change the hero copy, the image, the problem framing, the proof, and the button color all at once, and the new version does better — you've learned exactly nothing. You can't tell which change moved it.

One variable per cycle. Change one thing. Measure. Then change the next. It feels slow and it's the fastest way to actually learn what moves your page, because every result points at a single cause.

The loop is **PDCA** — Plan, Do, Check, Act — the same learning cycle Deming carried to Japan in 1950 and Toyota built its production system on. On a landing page it reads cleanly:

- **Plan** — form a hypothesis specific enough to be wrong. “If I name the segment instead of saying ‘founders,’ Targeting moves from 7 to 9.” The brief *is* the plan.
- **Do** — build the smallest version of that one change. Not the whole page.
- **Check** — score it, and read the metric you predicted would move. Honestly.
- **Act** — keep what worked, discard what didn't, update the brief, run the next cycle.

After a few cycles you'll have a verdict: kill it, keep iterating, or scale it by turning up the traffic you held back until the page

was ready.

The Check step now comes to you. In LEANSpark, a page ships inside a **conversion experiment** with a structured goal — a target rate the page has to hit, a visitor floor below which there's no verdict, and a time window. Recording the page starts the clock. Every three days the system posts a check card: *report your numbers*. Your AI assistant pulls the real analytics and reports them back over the same MCP connection, and LEANSpark scores them against the goal — on goal, off track, or not enough traffic yet. The loop you used to run on willpower now runs on a schedule. One honesty rule is enforced: real counts for real periods. The system doesn't take estimates.

The agent proposes. The founder decides. Your AI assistant can stage an experiment for you — template, hypothesis, goal — over MCP. What it *can't* do is start one. A staged experiment sits in your sprint until you review it and press start, because the moment an agent can silently start and stop experiments, you've automated lying to yourself. Drafts and proposals are the agent's job. Decisions stay yours.

Run one clean cycle

The job: turn the weakest dimension into a single-variable experiment.

▸ IN LEANSPARK — YOU TYPE

Help me iterate my page

Takes the weakest dimension from your last score, turns it into a one-variable hypothesis, drafts just that change in your voice, and re-scores after — so each cycle teaches you exactly one thing.

RETURNS A single-variable change, a before-and-after score, and a keep / kill / scale call. You ship the change and read the live metric.

▸ VIA MCP — PROMPT ANY AGENT

"My page scores [X], weakest on [dimension]. Propose one — only one — variable to change, write the new version in my voice, and tell me which metric should move if the hypothesis holds."

Put the page on the clock

The job: ship the page inside an experiment with a goal it can fail.

▸ IN LEANSPARK — YOU TYPE

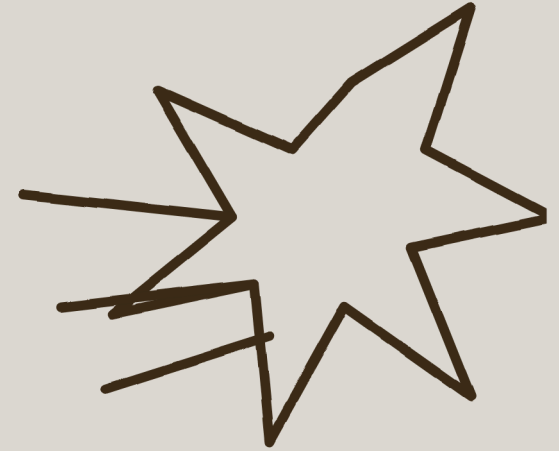
Stage my conversion experiment

Stages a Landing Page Conversion Test in your current sprint — hypothesis, target rate, visitor floor, time window — and files your page into it. The experiment lands as a proposal: nothing runs until you review it and press start.

RETURNS A staged experiment with the goal attached and the page recorded against it. You start it; the check clock starts when the page is recorded; every three days the system asks for real numbers.

▸ VIA MCP — PROMPT ANY AGENT

"Stage a landing page conversion experiment for this page over MCP, with my hypothesis and a target rate worth beating — then file the page into it with record_artifact. Tell me what it returned, including who has to start it. When the check cards arrive, pull my real analytics and report them with report_metric — real counts for real periods only."



CHAPTER 7

Resources

Score and iterate your page on LEANSpark. Go deeper on the method.

CHAPTER 7

Resources

Run your page on LEANSpark

- **Score it** — the Landing Page Assessment reads your live page and scores all seven dimensions, gates each one, and names what to fix first. This is the scorecard the whole loop runs on.
- **Draft from context** — generate a starting page from your canvas, voice profile, interviews, proof, and design system, so the copy begins from validated evidence rather than a blank page — and wears your actual brand.
- **Put it on the clock** — stage a Landing Page Conversion Test with a structured goal (target rate, visitor floor, time window). Recording the page starts the check clock; every three days the system asks for real numbers and scores them against the goal.
- **The context engine** — your **Voice Profile** and **Library** (interviews, proof, past content) are the same ones that power your content. One source of truth, two surfaces.
- **Pull it anywhere** — via the MCP, any AI assistant can read your voice, offer, competitive landscape, segments, interviews, and design system — and write back: file the

page into an experiment, report its real metrics, even stage the next experiment for your review.

Where the line falls. LEANSpark feeds the page validated context, scores it, runs the check schedule, and keeps the verdict honest — that’s the shipped part. What stays yours is the *deciding*: you write the brief, you start the experiment, you make the kill / iterate / scale call. The agent proposes. The founder decides.

Score your page at leanspark.ai.

Go deeper on the method

- **The Content Pipeline** (No. 8 in this series) — the companion to this playbook: it fills Rooms 1 and 2 off the *same* context engine that converts in Room 3.
- *Running Lean* and **The AI-Native Founder’s Playbook** (No. 1) — where positioning, segments, and proof come from in the first place.
- **How to Talk to Customers** (No. 3) and **How to Sell Before You Build** (No. 5) — the discovery and the offer that give the page its language and its proof.

The page at a glance

STAGE	WHAT YOU'RE DOING
Brief	Five questions, answered from evidence. One falsifiable promise, one CTA.
Evidence	Brand buys a few points; validated context buys the rest. Receipts, not adjectives.
Score	Seven dimensions out of 100, a gate on each. Fix the weakest first.
Measure	Five metrics, in order. Qualified signup is the only one that counts.
Experiment	A goal the page can fail: target rate, visitor floor, time window. The clock starts when the page is recorded.
Loop	One variable per cycle. Plan, do, check, act. Kill, iterate, or scale — you make the call.



How to Turn Visitors Into Customers is part of The AI-Native Playbooks, a series from LEANSTACK / Ash Maurya. LEANSpark feeds your page the context that converts, scores it, and runs the conversion experiment around it.