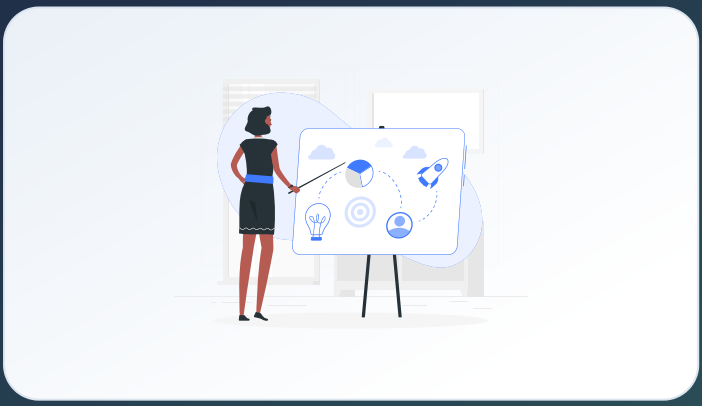
Co-funded by
the European Union

MODULE 2 · FREELANCER-TO-STARTUP TOOLKIT

Test It for Real

Lean start-up & digital innovation, test a real service cheaply, read the evidence honestly, and use everyday AI to punch above your weight.





Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them. · Erasmus+ KA210-YOU · Project 2025-1-SI02-KA210-YOU-000365490

MODULE TWO

Test It for Real

02

Will anyone actually pay for it? Replace guessing with cheap, fast tests, before you sink weeks into the wrong idea.

Module 1 helped you map your business and define an offer. Module 2 asks the harder, more honest question, **will anyone actually pay for it?** The mindset shift is everything: from "build it and hope" to **"run a small experiment and find out."**



By the end, a participant can

- Run the Build–Measure–Learn loop on a real offer, cheaply and fast.
- Design a service MVP and do customer discovery without pitching.
- Tell strong evidence from weak praise, and use everyday AI responsibly.

~3 hrs facilitated | 3 facilitator-led + 2 self-directed | Localised · PT · LT · SI | Youthpass · Entrepreneurship + digital

CONCEPTS IN THIS MODULE

1 · The Build–Measure–Learn loop
2 · Search before you scale
3 · The smallest version worth testing (MVP)
4 · Customer discovery
5 · Reading the signals (validation)
6 · Testing without a big-city market
7 · Where AI actually helps
8 · Everyday digital & the growth funnel
9 · Work in small, fast cycles
10 · The one-page experiment card

ACTIVITIES & RESOURCES

F1 · Discovery Conversations, 75 min
F2 · Strip It to the MVP, 60 min
F3 · AI in Practice, 60 min
S1 · Run a Cheap Test, 45 min
S2 · Your Experiment Card, 30 min
+ Case study · resources · templates

THE ENGINE

The Build–Measure–Learn loop

Most freelancers build first and hope, pouring weeks into a service, a site or an offer before anyone confirms it's wanted. The Build–Measure–Learn loop flips that: it treats every idea as a testable guess and gives you the fastest way to find out if you're right, before you've spent your savings finding out you're not.



WHAT IT IS

The loop has three steps. **Build** the smallest thing that can test one specific assumption. **Measure** what real people actually do with it, not what they say they'd do. **Learn** from that evidence and decide whether to persevere, tweak, or pivot. Then go round again. The unit of progress isn't features shipped, it's assumptions tested.

WHY IT MATTERS FOR YOU

Time and money are a freelancer's scarcest resources, and building the wrong thing burns both. The loop is how you fail cheaply and early instead of expensively and late. Each lap converts a risky guess ("people will pay for this") into evidence, so by the time you invest properly, you're building on facts, not hope. The faster you loop, the faster you learn what's true.

THE COMMON MISTAKE

Building too much before measuring, so the "test" is really a launch, and treating opinions as evidence. "That's a great idea, I'd totally use it" costs someone nothing to say. Measure behaviour, a deposit, a sign-up, a click, not enthusiasm. And test one assumption at a time, or you won't know what the result actually proved.

✎ Do this now

Name the one assumption that would kill your idea if it's false. Then design the smallest test you could run this week to find out, and put a date on it.

Facilitator note

Push participants to state their riskiest assumption as a sentence that could be proven false. If it can't be wrong, it can't be tested, help them make it specific and falsifiable.

Turn your best guess into evidence, cheaply and fast, then let the evidence decide your next move.

THE ENGINE

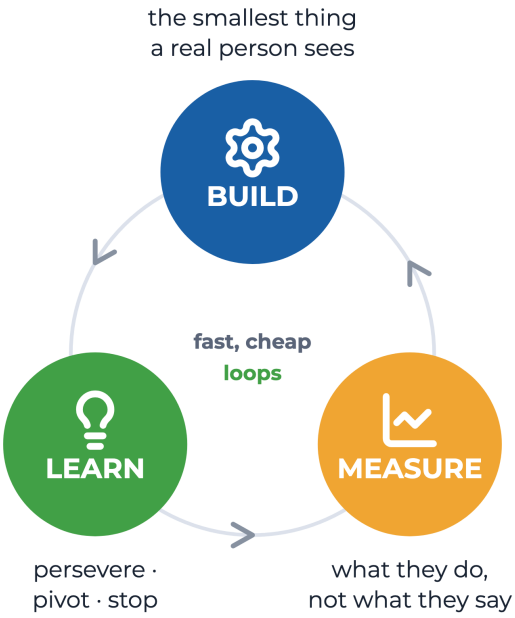
The Build–Measure–Learn loop

HOW IT WORKS

Pick your single riskiest assumption, the one that sinks the whole idea if it's wrong. Build the smallest possible test of just that, a message, a mock-up, a pre-sale. Put it in front of real people and watch what they do. Then decide: does the evidence say keep going, adjust, or change direction? Resist adding anything the test doesn't need.

The Build–Measure–Learn Loop

Turn a guess into evidence as fast and cheaply as you can, then go again.



Speed beats size, a small loop run weekly out-learns a big launch.

A WORKED EXAMPLE

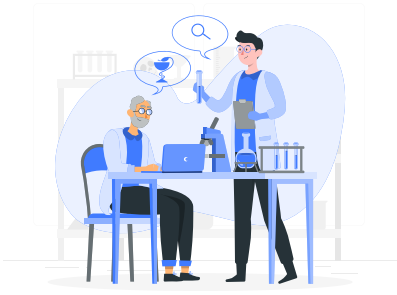
Léa wants to launch a meal-prep service. Instead of renting a kitchen, she loops: she posts one photo and a price in a local group (build), counts how many actually pay a deposit, not just “like” (measure), and learns people want it, but on Sundays, not weekdays (learn). One weekend, real evidence, no kitchen rented.

Don't build and hope. Build the smallest test, measure what people do, learn, and loop again, fast. Speed of learning is the real competitive advantage.

TWO MODES

Search before you scale

The most expensive freelance mistakes come from doing the right thing at the wrong time, specifically, scaling a business you haven't yet proven. Customer Development names two distinct modes, search and execution, and its core rule is simple: don't switch from one to the other until the evidence says you've earned it.



WHAT IT IS

Every venture starts in **search**: you don't yet know who exactly wants this, what they'll pay, or how to reach them, so the job is to find out through cheap experiments, conversations and pivots. Once you have real evidence, paying customers, a repeatable way to reach them, you shift to **execution**: doubling down, systematising, and scaling what works. They are genuinely different jobs with different rules.

WHY IT MATTERS FOR YOU

Searching and executing reward opposite behaviours. Search rewards curiosity, cheap tests and changing your mind; execution rewards focus, consistency and commitment. Scale, hire, spend on ads, commit to stock, while you're still searching, and you lock in a guess before you know it's right, the classic way freelancers burn their savings. Knowing your mode tells you whether to experiment or to commit.

THE COMMON MISTAKE

Scaling too early, quitting your job, building the full product, spending on ads, before you have proof, and, less obviously, staying in search mode forever because committing is scary. Both are failures of timing. Let evidence, not fear or excitement, tell you when to switch modes.

✎ Do this now

Decide which mode you're actually in today. If you're searching, list the next three cheap experiments. If you're executing, name the one thing that's working and how you'll do more of it.

Facilitator note

Have participants declare their mode and defend it with evidence. Most who claim to be “executing” are still searching, that realisation, made safely in the room, saves real money.

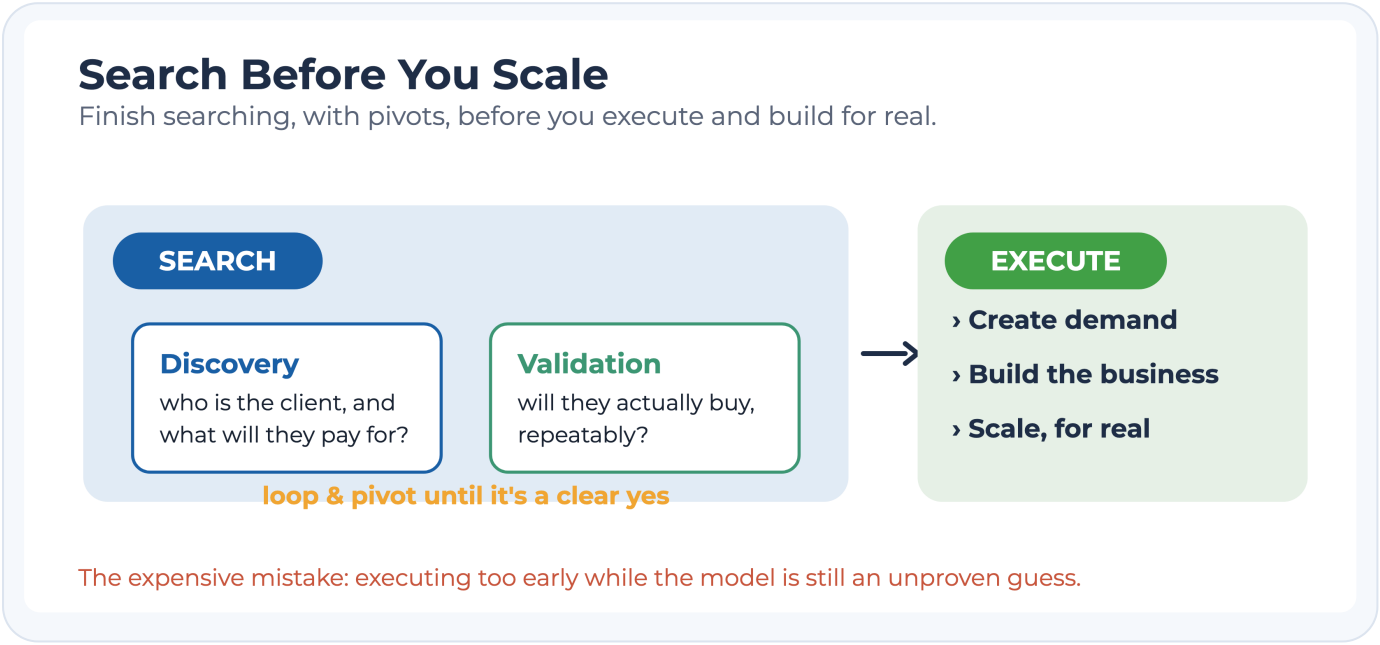
Finish searching, with pivots and dead ends, before you execute. Never pour fuel on a guess.

TWO MODES

Search before you scale

HOW IT WORKS

Ask honestly: do I have evidence that a specific group will pay for this, and a repeatable way to reach them? If not, you're in **search**, so keep experiments small, cheap and reversible, and expect pivots. If yes, you're ready to **execute**, so stop second-guessing and pour your energy into scaling what works. Don't spend like you're executing while you're still searching.



A WORKED EXAMPLE

Tiago wants to sell a fitness app. In search mode he runs a €0 test: a free WhatsApp group and a simple plan. Only when fifteen people ask to pay does he know he's found something, then he switches to execution and builds the app. Had he built first, he'd have scaled a guess.

Search and scale are different jobs. Stay in search, messy and cheap, until the evidence is undeniable, then commit fully to execution. Getting the order right is half the battle.

START SMALL

The smallest version worth testing

"Minimum Viable Product" is the most misused phrase in this module. It doesn't mean a rough, buggy version of your grand plan. It means the smallest experiment that answers one question: will real people take a real action that proves they want this? Get it right and you save months; get it wrong and you build a beautiful thing nobody asked for.



WHAT IT IS

An MVP is the least you can make or do to test your riskiest assumption with real people. For a freelancer it's usually not software at all: a one-page site with a "buy" button, a manual, done-by-hand version of the service, a single pre-order. It's **viable** (it delivers just enough value to test the response) and **minimum** (it contains nothing that doesn't serve the test).

WHY IT MATTERS FOR YOU

Building the full thing first is a bet you can't afford: you spend your scarcest resources before you know anyone wants it. The MVP inverts the risk, tiny investment, real answer. It also teaches you what to build: how people react to the smallest version reveals what actually matters to them, so if you do build more, you build the right things. Learning, not launching, is the point.

THE COMMON MISTAKE

Building a "minimum" version that's still far too big, or one so stripped-back it delivers no real value and tests nothing. The MVP must be genuinely small *and* valuable enough that a real response means something. And "people said they liked it" is not a pass, only real action, paying, signing up, showing up, counts.

✎ Do this now

Describe your idea, then design the smallest version you could put in front of real people this week, ideally done by hand. Name the single action that would prove they want it.

Facilitator note

Challenge every "MVP" that still takes weeks to build. Ask: what could you test by Friday, with no code? The best MVPs in the room are usually a conversation, a page, or a pre-sale.

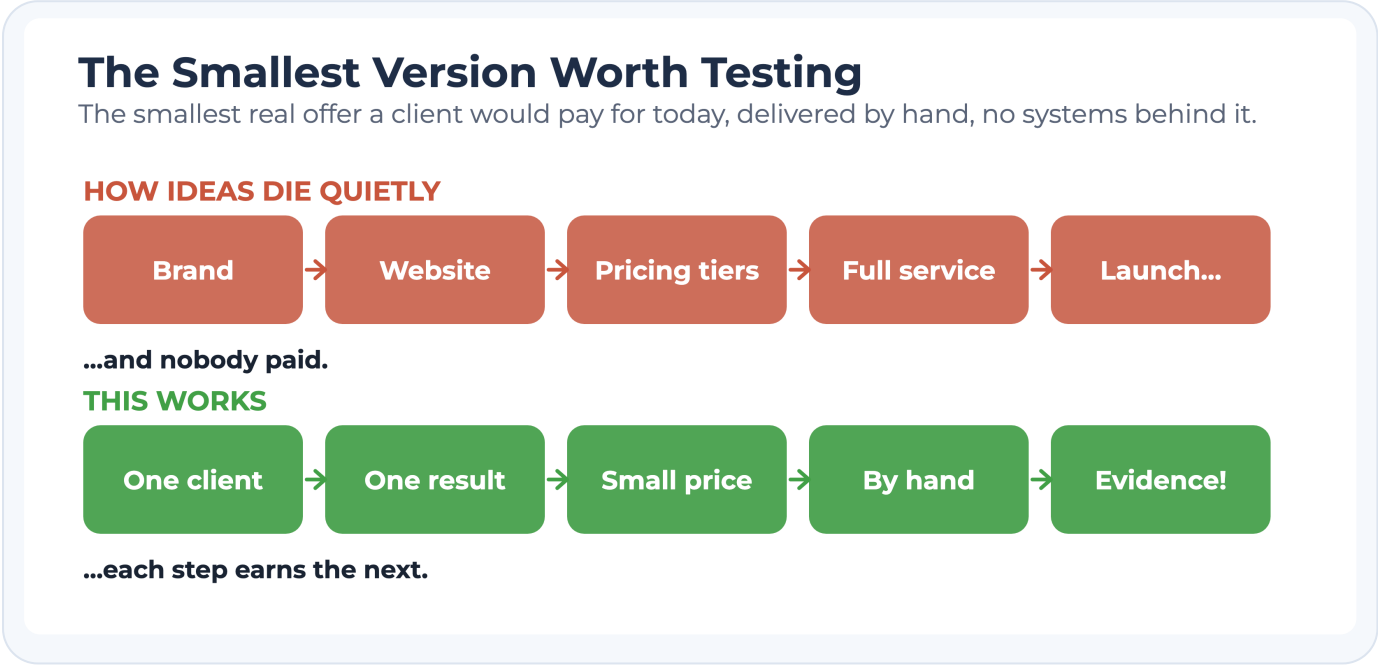
Don't build the whole studio. Sell one real result, to one real client, by hand, today.

START SMALL

The smallest version worth testing

HOW IT WORKS

Write down the value you claim to offer and the one assumption you're least sure of. Design the smallest way to test it with real behaviour, and strip out everything else. Prefer manual over automated, a page over a product, a pre-sale over a build. Put it in front of real people and measure what they do.



A WORKED EXAMPLE

Marta dreams of an online course. Her MVP isn't a course, it's a single live workshop she sells to ten people for a small fee. If they pay and love it, she's proven demand and learned which lessons matter, then she records the course. If they don't, she's out one afternoon, not three months.

Don't build the whole thing and hope. Build the smallest test of the one thing that matters, deliver it by hand if you can, and let real behaviour tell you what to do next.

THE HIGHEST-RETURN HABIT

Customer discovery, then iterate

You cannot think your way to what customers want, you have to ask them, carefully. Customer discovery is the disciplined habit of learning about your customers' real problems before, and while, you build. Done well it's the highest-return activity a freelancer can do; done badly, as a disguised sales pitch, it produces flattering lies that lead you off a cliff.



WHAT IT IS

Customer discovery is a small number of honest, structured conversations with the people you hope to serve, aimed at understanding their world: the job they're trying to do, the problems in their way, what they've already tried, and what those problems cost them. It is emphatically *not* a pitch. You're a curious researcher, not a salesperson, the goal is truth, not a yes.

WHY IT MATTERS FOR YOU

Guessing what customers want is the single biggest cause of wasted effort. Discovery replaces guessing with evidence for almost no money, an hour of conversations can save months of building the wrong thing. It also hands you the exact words your customers use, which later become the most persuasive words in your marketing. Everything downstream, the offer, the price, the pitch, gets easier when it's built on what people actually said.

THE COMMON MISTAKE

Turning the conversation into a pitch (“so, would you use something like this?”), which trains people to be polite instead of honest. Also asking about the future (“would you pay for X?”) instead of the past (“what did you do last time, and what did it cost?”). People are unreliable about hypotheticals but honest about history, mine the history.

✎ Do this now

List three real people who fit your target customer. Message one today and ask for fifteen minutes to hear about their experience with the problem, no pitch. Then just listen and take notes.

Facilitator note

Role-play a bad interview (all pitch) and a good one (all curiosity) so the difference is obvious. Then have participants draft three past-focused questions and test them on each other before going to real people.

Talk to people about their problem, not your idea. A few honest conversations beat months of guessing.

THE HIGHEST-RETURN HABIT

Customer discovery, then iterate

HOW IT WORKS

Pick a handful of real people who fit your target. Ask about their **past and present**, not your future product: "Tell me about the last time you dealt with X." "What did you try? What did it cost you?" Listen far more than you talk, and never describe your idea until the end, if at all. Look especially for problems people have already paid, in time or money, to solve.



A WORKED EXAMPLE

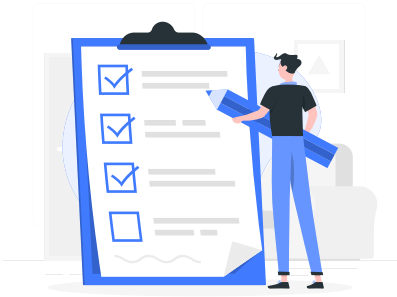
Before building a booking tool for hairdressers, Ivan interviews eight of them. He never mentions his app. He just asks how they handle appointments now, and discovers their real pain isn't booking, it's no-shows. He'd have built the wrong product; instead he builds a deposit feature they immediately want.

The cheapest, most valuable thing you can do is ask, about their problem, not your idea, and listen. A few honest conversations will out-teach months of clever guessing.

CLIMB THE PROOF LADDER

The validation ladder

People say “we validated it” to mean anything from “my friends liked the idea” to “forty strangers pre-paid.” Those are wildly different levels of certainty. The validation ladder makes the difference explicit: each rung is a stronger form of proof, and the discipline is to climb high enough, before you bet, that the evidence actually justifies the risk you're taking.



WHAT IT IS

The ladder runs from weak to strong signals. Near the bottom: attention and opinions, likes, “great idea”, a click. In the middle: small commitments, an email address, time booked, a waitlist spot. Near the top: real money and repetition, a pre-payment, a purchase, a repeat purchase, a referral. Each rung costs the customer more, and is therefore worth more as evidence.

WHY IT MATTERS FOR YOU

Not all evidence is equal, and treating a low rung as a high one is how people confidently launch things nobody buys. Likes and encouragement are nearly free to give, so they prove almost nothing; parting with money is costly, so it proves a lot. The ladder lets you calibrate: it tells you how strong your evidence really is, and whether it's strong enough for the size of the leap you're about to take.

THE COMMON MISTAKE

Confusing a low rung for a high one, treating likes, encouragement or “I'd definitely buy that” as proof, then committing real money on the strength of it. The opposite error exists too: demanding top-rung proof for a trivial, cheap experiment. Match the rung to the risk, no higher, no lower.

✎ Do this now

Write down your next big decision and what it risks. Then name the highest rung of proof you currently have, and design one test to climb at least one rung higher before you commit.

Facilitator note

Have participants place their current evidence on the ladder honestly, most sit lower than they think. Then set each a “one rung higher” challenge: turn a like into a sign-up, a sign-up into a pre-sale.

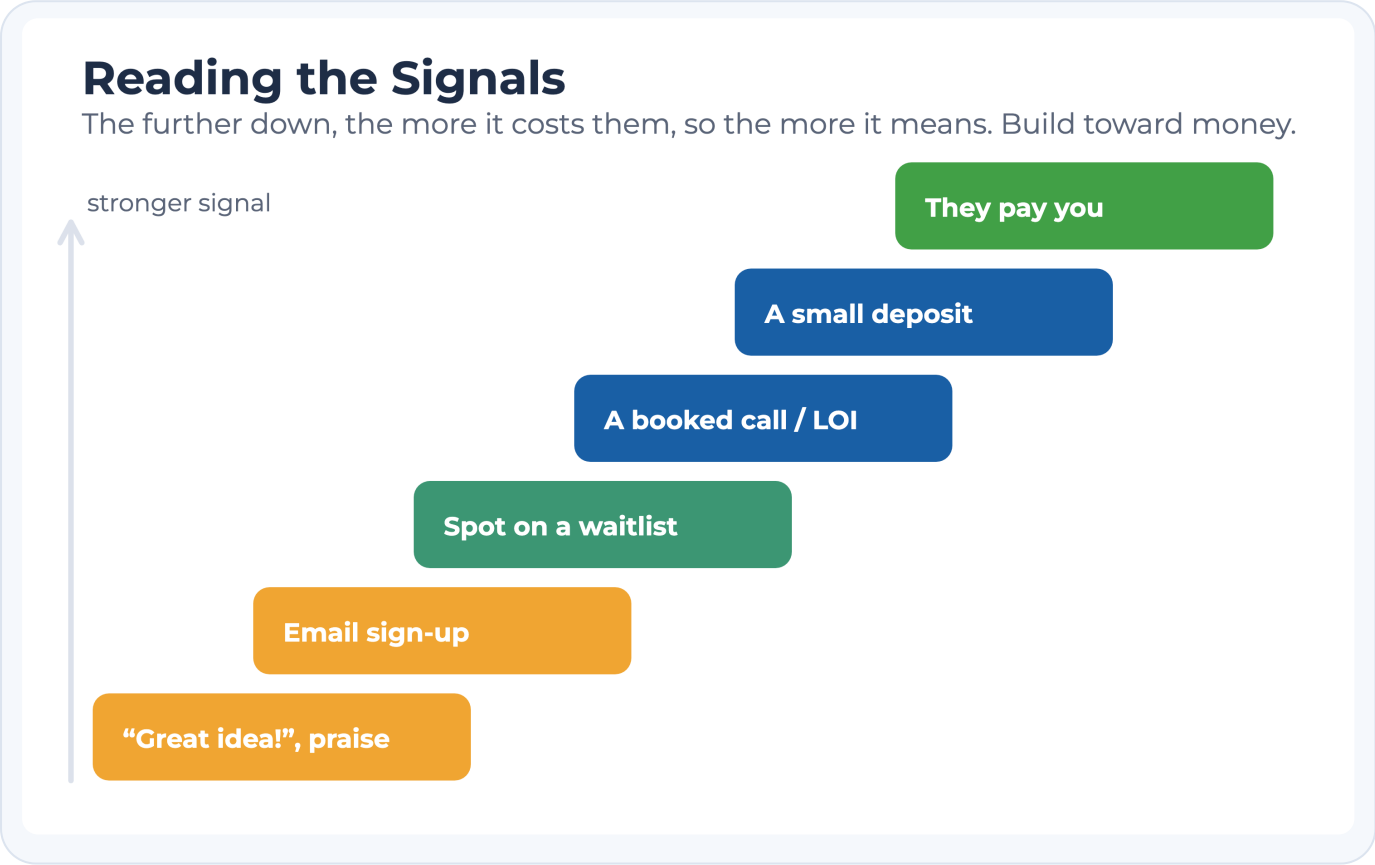
Validation is a ladder, not a yes. Climb from likes to sign-ups to payments, and match how high you climb to how much you're about to risk.

CLIMB THE PROOF LADDER

The validation ladder

HOW IT WORKS

Name the decision you're about to make and how much it risks. Then ask which rung of proof it deserves, and design a test to reach it. Chasing a bigger bet? Don't settle for sign-ups; ask for a pre-payment. The key move is always to push people one rung higher, from saying to doing, from doing to paying, because each step up sharply increases how much you can trust the result.



A WORKED EXAMPLE

Sara has 300 likes on her craft-kit idea, exciting, but low on the ladder. Before ordering stock she climbs: she opens pre-orders. Only twelve of the 300 actually pay. That gap, likes versus payments, saves her from buying inventory for 300 customers she didn't have. She adjusts and starts small.

Not all evidence is equal. Climb from words to actions to money, and only bet as big as your rung on the ladder allows.

REACH PEOPLE ANYWHERE

Reaching people, city or not

A quiet assumption runs through a lot of business advice: that you're in a city, with a network and a scene. Millions of young freelancers aren't, and that's fine. Reaching customers isn't about where you live; it's about choosing channels that fit your customers and your context, and today those range from the community hall down the road to a global platform on your phone.



WHAT IT IS

A channel is simply how you reach and are found by customers. They fall into two families. **Local and human**: word of mouth, community and religious groups, local markets, noticeboards, partnerships with nearby businesses, family and neighbour networks. **Online and borderless**: marketplaces, social platforms, freelance sites, local buy-and-sell groups, that let you serve clients regardless of distance. Most freelancers do best with a small mix of both.

WHY IT MATTERS FOR YOU

Choosing the wrong channel wastes effort on people who'll never buy, while the right one puts you in front of ready customers cheaply. For freelancers outside major cities this matters doubly: local channels turn a small, trusted community into reliable early clients, and online channels remove the geographic ceiling entirely. You don't need everyone, you need to be reliably visible to the specific people who want what you do.

THE COMMON MISTAKE

Copying the channels of famous founders, big-city networking, paid ads, a huge following, that don't fit your customers or context, and ignoring the humble local channels that actually work. Also spreading across every platform at once, better to own one or two channels your customers really use than to be faintly present everywhere.

✎ Do this now

Name where your ideal customers already gather, one local place and one online. Commit to showing up in just those two, consistently, for the next month, and ignore the rest.

Facilitator note

Map the room's channels on a local–online line. Participants outside cities often undervalue their local, trust-based channels, surface those, and pair each person with one realistic online channel too.

Reach people through the channels they already use, local networks or global platforms, not the ones that merely sound impressive.

REACH PEOPLE ANYWHERE

Reaching people, city or not

HOW IT WORKS

Start from your customer, not the channel: where do they already spend time, ask for recommendations, and look for services like yours? Pick one or two channels that match, one local, one online is a strong start, and show up there consistently rather than spreading yourself thin everywhere. Lean on trust: in smaller communities a personal recommendation outperforms any advert.



A WORKED EXAMPLE

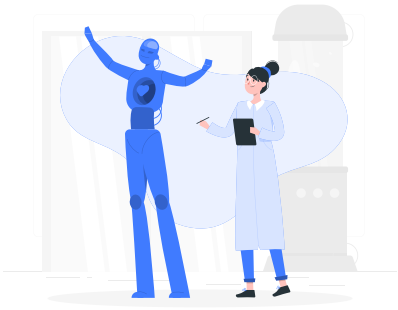
Nadia runs a tailoring service in a small town. She skips glossy ads and uses two channels: the local women's WhatsApp group (word of mouth, trusted) and a national marketplace app (distance-free). The first fills her week locally; the second brings custom orders from cities she's never visited.

You don't need a city or a crowd, you need the two or three channels your specific customers already use. Choose those, show up consistently, and let distance stop being an excuse.

YOUR UNFAIR HELPER

Where AI actually helps a freelancer

AI is neither magic nor a threat to fear, for a freelancer it's a fast, tireless assistant that's brilliant at starting things and terrible at being trusted. Used well it removes the two biggest drags on a one-person business: the blank page and the endless admin. Used badly, it fills your work with confident nonsense and quietly leaks your clients' data. The skill is knowing exactly which jobs to hand it.



WHAT IT IS

For a freelancer, AI is genuinely useful at four jobs. **Draft**: first versions of proposals, posts, replies, outlines, ideas to react to. **Summarise**: turn a long document, thread or call into the key points. **Research**: gather, compare and organise information to a starting point. **Automate**: handle repetitive admin, tidy notes, reformat, sort. In all four it produces a *starting point*, never a finished, trusted result.

WHY IT MATTERS FOR YOU

The blank page and the admin pile are where solo freelancers lose hours and momentum. AI collapses both: a draft in seconds beats a morning staring at nothing, and automated busywork frees you for the work only you can do. But its fluency is a trap, it states wrong things as confidently as right ones, so its value is entirely in *speed*, never in *trust*. It multiplies your output only if you keep the judgement.

THE COMMON MISTAKE

Two dangerous habits. Publishing AI output unread, so its confident errors become your mistakes and its blandness becomes your voice. And pasting confidential client material, contracts, personal data, into tools you don't control. Treat AI as a junior assistant whose work you always check and own, never as an authority.

✎ Do this now

Pick one recurring task that eats your time, drafting replies, summarising briefs, tidying notes, and try AI on just that this week. Judge the result honestly, and keep it only if you'd sign your name to it.

Facilitator note

Have participants try one real task with and without AI and compare. The lesson lands on its own: faster start, but useless until a human checks, cuts and rewrites it. Reinforce the two hard rules on facts and privacy.

Let AI take the first draft and the admin, you keep judgement, taste and the final decision.

YOUR UNFAIR HELPER

Where AI actually helps a freelancer

HOW IT WORKS

Give it a clear instruction, who it's for, the goal, the tone, the format, and let it produce options fast. Then do the part it can't: check every fact, cut what's generic, and rewrite it in your own voice until it sounds like you, not a machine. Reach for it on the four jobs above; keep it away from anything needing real judgement, taste, or confidential client information.



A WORKED EXAMPLE

Bruno, a freelance marketer, uses AI to draft ten subject-line options in seconds (draft), condense a client's twenty-page brief into one page (summarise), and list competitors to check (research). He then chooses, verifies and rewrites everything himself. He's twice as fast, and still unmistakably him.

AI removes the blank page and the busywork, not the thinking. Let it draft, summarise, research and automate, then check it, own it, and keep the judgement for yourself.

FIND THE LEAK

The growth funnel

Freelancers usually think about growth as “I need more clients”, so they chase more attention. But attention is only the top of a five-stage funnel, and if a lower stage leaks, more attention just drains out faster. The AARRR funnel (sometimes called “pirate metrics”) shows you the whole journey, so you can spot where you're actually losing people and fix the stage that matters.



WHAT IT IS

Five stages, top to bottom. **Acquisition**: people discover you. **Activation**: they have a first good experience. **Retention**: they come back. **Referral**: they tell others. **Revenue**: they pay, and pay again. Every stage leaks, only some of the people at one stage reach the next, and the shape of your leaks tells you exactly where your growth is stuck.

WHY IT MATTERS FOR YOU

Treating growth as a single number (“more clients”) hides where the real problem is. One freelancer's bottleneck is that nobody finds them (acquisition); another's is that people try them once and never return (retention). The same effort, more marketing, helps the first and completely wastes the second. Seeing the funnel lets you diagnose instead of guess, and put your limited energy where it actually moves the needle.

THE COMMON MISTAKE

Optimising the wrong stage, usually pouring effort into getting more attention when the real leak is further down (people try you once and vanish, or never refer). Also ignoring retention and referral entirely, chasing new clients while quietly losing the ones you have. Diagnose first; the loudest stage is rarely the leakiest.

✎ Do this now

Draw your five funnel stages and estimate the numbers. Circle the single biggest drop-off, that's your bottleneck, and name one change to fix it this month, before you chase any new leads.

Facilitator note

Have participants guess their leakiest stage, then estimate the numbers, the two often disagree. The gap between assumed and actual bottleneck is the insight; direct their next effort at the real one.

Don't chase more leads, find the leakiest stage of your funnel and fix that one first.

FIND THE LEAK

The growth funnel

HOW IT WORKS

Sketch your five stages and put a rough number on each, out of everyone who finds you, how many try, return, refer, pay? The stage with the biggest drop-off is your bottleneck. Fix that one before anything else. And remember the loop: delighting customers so they refer others quietly refills the top of the funnel, often the cheapest growth there is.

The Growth Funnel (AARRR)

Where do clients drop off? Track five stages, fix the leakiest one first.



Also called "Pirate Metrics." For a freelancer: don't chase more leads if the leak is retention.

A WORKED EXAMPLE

Ana gets plenty of enquiries (strong acquisition) but few become clients, a big leak at activation. More marketing would be wasted. She fixes the leak instead: a clearer first reply and a simple starter offer. Same number of enquiries, twice the clients, because she fixed the stage that was actually broken.

Growth isn't more leads, it's a less leaky funnel. Find the stage that loses the most people, fix that one, and let happy customers refill the top.

SHIP SMALL, SHIP OFTEN

Work in small cycles

Perfectionism is a freelancer's quiet killer: you wait until it's “ready”, and “ready” never comes, so nothing ships and nothing is learned. Working in short, agile cycles is the antidote. Instead of one long march to a big launch, you take small, repeated loops, each ending in something real, so progress and learning happen continuously, not all-or-nothing at the end.



WHAT IT IS

An agile cycle is a short, repeating loop: **plan** a small next step, **do** it, **review** what happened and what you learned, then **adjust** and go again. Each loop is short (days or a week, not months) and ends with something concrete, delivered, shown, or tested. The rhythm matters more than the size of any single step; you're trading one big bet for many small, correctable ones.

WHY IT MATTERS FOR YOU

Long, hidden projects accumulate risk: you can be wrong for months without knowing it, and by the time you reveal the work, correcting course is expensive. Short cycles surface reality early and often, every loop is a chance to catch a wrong turn cheaply. For a solo freelancer juggling client work, they also make big goals manageable: a scary project becomes a series of small, finishable steps, each with visible progress that keeps you moving.

THE COMMON MISTAKE

Planning everything in detail before starting (and never starting), or running cycles so long they're just mini-waterfalls that hide problems for weeks. The other trap is skipping the *review*, doing lots but never pausing to learn, so you move fast in possibly the wrong direction. The learning step is what makes it agile, not just busy.

✎ Do this now

Take one stalled project and slice off the smallest piece you can finish and show this week. Plan only that, do it, and book a fifteen-minute review with yourself for Friday to decide the next loop.

Facilitator note

Have each participant define a one-week cycle for a real goal, then commit to a review point. Emphasise ending every cycle with something visible, that's what turns motion into progress and feedback.

Work in small cycles, plan, do, review, adjust, so you're always shipping and learning, never disappearing for months.

SHIP SMALL, SHIP OFTEN

Work in small cycles

HOW IT WORKS

Take your goal and slice off the smallest useful piece you could finish this week. Plan just that, do it, then honestly review, what worked, what you learned, what's next, and adjust the plan before the next loop. Keep each cycle short and always end it with something real you can look at. Resist planning the whole thing in detail up front; plan the next loop, not the next year.



A WORKED EXAMPLE

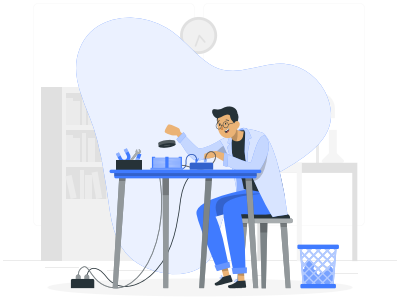
Hugo is building a portfolio site and keeps stalling on “getting it perfect.” He switches to weekly cycles: week one, one page live; week two, three projects added; week three, a contact form. Each week something real ships and gets feedback. In a month it's live and improving, instead of endlessly “almost ready.”

Stop waiting for perfect. Plan a little, ship a little, review, adjust, and repeat, small loops keep you learning and moving while others are still getting ready.

TEST, DON'T GUESS

Run it like an experiment

Most freelancers make big calls, on offers, prices, channels, on gut feeling, then defend them because changing course feels like failure. Running decisions as experiments removes the ego: instead of betting your belief, you design a small test with a result you agree to trust in advance. It's the discipline that turns every uncertain choice into cheap, honest learning rather than an expensive guess you're now committed to.



WHAT IT IS

An experiment is a structured test of one assumption. The tool is an **experiment card**: you write the **assumption** ("I believe X"), the **test** (the smallest way to check it), the **metric** (what you'll measure), and the **pass/fail line** (the result that means yes, and the one that means no), all *before* you run it. Then you run it, record what actually happened, and act on the outcome, not on your hopes.

WHY IT MATTERS FOR YOU

Deciding the success criteria beforehand is what makes an experiment honest. Run a test without a pre-agreed line and you'll bend any result into confirmation of what you already wanted, learning nothing. The card forces you to commit to the truth in advance, so the outcome can genuinely change your mind. It also makes decisions cheap and reversible: a small, timed test with a clear rule is far safer than a gut-feel commitment you can't easily undo.

THE COMMON MISTAKE

Running a "test" with no pre-set pass/fail line, so you interpret the result to suit yourself, or making the experiment so big and slow it's really just the launch. The whole value is in deciding the rule first and keeping the test small. And a result that says "no" is a win too: it saved you from a costly mistake, don't explain it away.

✎ Do this now

Take one decision you're unsure about and fill an experiment card: assumption, smallest test, metric, and the pass/fail line, set the line before you run it. Then run it this week and honour the result.

Facilitator note

Have participants write a pass/fail line *before* designing the test, and hold them to it. The discipline of committing to a rule in advance is the entire lesson, celebrate honest "fails" as money saved.

Turn decisions into experiments: write the assumption, the test, and the pass/fail line before you run it, then let the result decide.

TEST, DON'T GUESS

Run it like an experiment

HOW IT WORKS

Take a decision you're unsure about and fill a card: state the assumption plainly, design the smallest, fastest test of it, choose one clear metric, and, crucially, write the number or outcome that would count as a pass, and the one that would count as a fail, before you start. Run it, record the real result, and do what you promised the result would mean, even if it's not what you hoped.

The One-Page Experiment Card

Fill it in before you test; let the number, not your hopes, decide what happens next.

EXPERIMENT CARD — one test · one number

1. **WE BELIEVE**
   [client] has [problem] and will pay for [our small offer]
2. **WE'LL TEST IT BY**
   the cheapest real test we can run
3. **WE'RE RIGHT IF**
   a clear number, e.g. 3 of 5 say yes, or 2 pay a deposit
4. **THEN WE'LL**
   persevere · pivot · stop

A WORKED EXAMPLE

Rita assumes clients will pay twenty percent more for a “priority” option. Instead of just raising prices, she cards it: test = offer it to the next ten enquiries; metric = how many choose it; pass = three or more. Four do. She's not guessing anymore, she has evidence, and a pre-agreed rule she trusts.

Don't gamble on a hunch, run a small experiment with the pass/fail line set in advance, and let the result, not your ego, make the call.

FACILITATOR-LED

F1 • Discovery Conversations

75 min

Role-play & planning

BASED ON

Sub-chapter 4 · Customer discovery & iteration.

BY THE END, PARTICIPANTS CAN

Run a problem-first interview without pitching, and plan five real discovery conversations to hold this week.

Step by step

1. **Frame it** · 10 min
   Explain the goal: understand the problem, don't sell. Money follows understanding.
2. **Demo** · 10 min
   Role-play a pitchy conversation, then a listening one. Ask the group to spot the difference.
3. **Practice in pairs** · 20 min
   A plays a potential client; B interviews using the prompt cards, questions only, no pitching. Then swap.
4. **Debrief** · 10 min
   What was hardest? (Usually: resisting the urge to pitch.)
5. **Plan real conversations** · 20 min
   Each person lists five real people in their niche and how they'll reach them.
6. **Commit** · 5 min
   Everyone commits to holding at least one real conversation before the next session.

THE FRAMEWORK

Talk about their problem, not your idea, questions only in round one.

→ See the full diagram

MATERIALS

Printed interview prompt cards ("What's the hardest part of...?"); a planning sheet for listing five people.

FACILITATOR PREP

Print prompt cards and planning sheets. Read sub-chapter 4. Be ready to role-play both a bad (pitchy) and a good (listening) conversation yourself.

Facilitator tips

The pitch reflex is universal, enforce "questions only" in round one. For shy learners, give a full written script. For migrant and Roma youth, conversations can happen in their community and first language, their networks are a real discovery channel, not a limitation.

→ What can I do next?

Hold one real discovery conversation this week and write down the exact words the person used about their problem.

Reflection & Youthpass

What did I hear from a real person that I could not have guessed alone? **Youthpass:** Communication + entrepreneurship, "I can learn from a real customer instead of guessing."

★ Expert extension

These conversations are the Discovery stage. Don't jump to selling or scaling until Discovery and Validation are done, and be ready to pivot if the evidence tells you to.

FACILITATOR-LED

F2 • Strip It to the MVP

60 min

Hands-on exercise

BASED ON

Sub-chapter 3 · Minimum Viable Product.

BY THE END, PARTICIPANTS CAN

Cut a 'dream service' down to the smallest payable version and define one real client they could offer it to this month.

Step by step

1. **Frame it** · 5 min
   An MVP is the smallest thing a client pays for today, delivered by hand, no systems yet.
2. **Brain-dump** · 10 min
   Write out your full 'dream service' with everything you imagine it including.
3. **Strike it out** · 15 min
   Cross out everything not needed to deliver one valuable result to one client this month.
4. **Define the MVP** · 15 min
   Write: the one result, the one client, the one (even small) price, delivered by hand.
5. **Pair check** · 10 min
   A partner asks: "could someone pay for just this, today?" Adjust until the answer is yes.
6. **Close** · 5 min
   Each person states their MVP in one sentence.

THE FRAMEWORK

The smallest thing a client pays for today, by hand, no systems.

→ See the full diagram

MATERIALS

Paper and pens; the "not this / this" MVP handout for reference.

FACILITATOR PREP

Print the MVP handout. Read sub-chapter 3. Have a relatable example of an over-built idea ready (e.g. someone who spent months on a brand before a single sale).

Facilitator tips

Learners over-build out of perfectionism. Celebrate the shortest list. For creatives, the urge to make it beautiful is strong, remind them: deliver by hand, no brand or website yet. Reassure NEET/first-time learners that 'by hand, small price' is exactly right.

→ What can I do next?

Offer your MVP to one real person this week, not a friend being polite, a genuine potential client.

Reflection & Youthpass

What was I about to build that I didn't actually need? **Youthpass:** Entrepreneurship, "I can find the smallest valuable version of my idea."

FACILITATOR-LED

F3 • AI in Practice

60 min

Demo & hands-on

BASED ON

Sub-chapter 7 · AI tools for freelancers.

BY THE END, PARTICIPANTS CAN

Use AI for a content draft, a customer-insight summary and a small automation, and apply the responsible-use rules.

Step by step

1. **Frame it** · 5 min
   AI is a confident intern: fast and helpful, but you're the boss who checks the work.
2. **Demo 'Create'** · 10 min
   Together, draft a client email or post from a short prompt, then critique and edit it as a group.
3. **Demo 'Understand'** · 10 min
   Paste the sample reviews and have AI summarise them into a simple client persona.
4. **Demo 'Automate'** · 10 min
   Create a reusable reply template or tidy a block of messy notes.
5. **Responsible-use rules** · 10 min
   Never send what you haven't read · no private client data · check facts · keep your own voice.
6. **Hands-on + close** · 15 min
   Each learner tries one task on their own offer, then shares one thing AI saved time on.

THE FRAMEWORK

Create · Understand · Automate · Learn, you always check the work.

→ See the full diagram

MATERIALS

A device or phone per learner with a free AI chat tool; the AI-tools map; a short block of sample customer reviews to summarise.

FACILITATOR PREP

Make sure everyone can access a free AI tool. Prepare and test the prompts yourself beforehand. Have the sample reviews ready to paste.

Facilitator tips

For low digital confidence, pair learners with a more confident peer and go slowly. Stress privacy around client data, make this rule non-negotiable. For non-native speakers, AI can help draft in a second language, but it must be checked, not trusted blindly.

→ What can I do next?

Use AI for one real task this week, a draft, a summary or an automation, and edit it into your own voice before using it.

Reflection & Youthpass

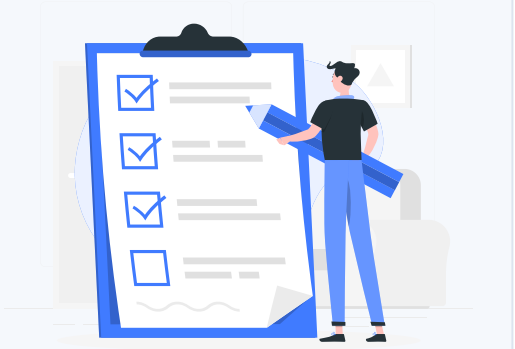
Where did AI genuinely help, and where did my own judgement matter most? **Youthpass:** Digital competence, "I can use AI tools practically and responsibly in my work."

SELF-DIRECTED · ~45 min + run time · Mini-mission

S1 • Run a Cheap Test

Not all 'yes' is equal. Chase the signals that cost the other person something, a deposit, a booking, a payment.

1. **Pick one assumption**
   e.g. "people will pay €X for Y".
2. **Choose the cheapest real test**
   offer it to five people · a small pre-sale · a simple message · a basic landing note.
3. **Decide your signal in advance**
   what exactly counts as a 'yes'? (money > committed time > email > praise)
4. **Run it and record**
   write down what people actually did, not what they politely said.



→ Next

Based on the signal, decide: persevere, pivot or stop. Then plan your next small loop.

✎ Your workings

Reflection & Youthpass

Did I get real commitment, or just praise? Is my success number honest, or did I set it low to protect my idea?

SELF-DIRECTED · ~30 min · Planning exercise

S2 • Your Experiment Card

Before you test, write it down. Afterwards, the card, not your hopes, tells you what to do next.

1. **We believe...**
   [client] has [problem] and will pay for [our small offer].
2. **We'll test it by...**
   the cheapest real test you can run.
3. **We're right if...**
   a clear number: e.g. 3 of 5 say yes, or 2 pay a deposit in a week.
4. **Then we'll...**
   persevere · pivot · stop, and add a date to run it by.



→ Next

Print the experiment card, run the test by your date, then fill in what happened and make the call.

✎ Your workings

Reflection & Youthpass

Did I get real commitment, or just praise? Is my success number honest, or did I set it low to protect my idea? **Youthpass:** "I can test an idea cheaply, read the evidence honestly, and decide what to do next."

CASE STUDY

Buffer: two pages before a product

Joel Gascoigne had an idea for a tool to schedule social-media posts. The obvious move was to spend months building it. Instead, in 2010, he did something far cheaper, he **tested whether anyone actually wanted it, before writing the real product**.

He built a tiny two-page website. The first page explained the idea and had a **"Plans & Pricing"** button. Anyone who clicked it, signalling they might *pay*, landed on a second page that said, honestly, "not quite ready yet," with a box to leave their email. That click was his signal: not praise, but a real step toward money.

Enough people clicked through and left their emails that Joel knew the demand was real. Only **then** did he build a minimum version and go back to those exact people. He had his first paying customers within weeks, and Buffer grew into a company used by millions, still without ever building on a guess.

Joel Gascoigne / Buffer. Sources: buffer.com · failory.com. Facilitator tip: pair it with a **local** freelancer's cheap test from your own community.



THE MODULE 2 IDEAS IN THIS STORY

- **Test before you build (MVP) →**
- **Reading the signal, a click toward paying →**
- **Build–Measure–Learn →**
- **The experiment card →**

What you can borrow, at any size

1. **Fake the offer first.** A one-page site or a simple message can test demand before you build anything.
2. **Make the signal cost something.** A pricing click, a deposit or an email beats a polite "great idea".
3. **Only build after the yes.** Go back to the exact people who signalled, they are your first customers.

He didn't build first and hope. He tested demand with two pages, then built only what people had already reached for.

GO DEEPER

Resource library

Everything to go further with Module 2. The **printable templates are built into this booklet** (next pages) and also at shift.elnlab.eu/resources.

SHIFT printable templates

- Experiment Card · S2 ↗
- Discovery Interview Guide · F1 ↗
- "Not this / this" MVP handout · F2 ↗
- AI-tools map · F3 ↗

Frameworks & further reading

- The Lean Startup, Eric Ries
- Customer Development, Steve Blank
- "Idea to paying customers in 7 weeks", Buffer
- The Mom Test, Rob Fitzpatrick (discovery)

Everyday digital tools (free tiers)

- Book, calendly / cal.com
- Get paid, sumup / wise / revolut
- Simple site, carrd.co / a Google form
- Talk to customers, WhatsApp Business

Free AI tools by job

- Create drafts, a free AI chat tool
- Understand, summarise reviews & feedback
- Automate, templates, tidy notes
- ⚠ Never paste private client data

Watch & listen (free)

- The Lean Startup, Eric Ries (talk)
- Steve Blank, Customer Development
- Buffer, the validation story

Case study

The "test before you build" story on the previous page, Buffer by Joel Gascoigne, pairs well with a local freelancer who ran a cheap test in your own community.

PRINTABLE TEMPLATE · 1 OF 4

Experiment Card

The tool for self-directed activity **S2 · Your Experiment Card**. Print several, you'll run many tests, not one.

Experiment Card

SHIFT · print & fill

Fill it in BEFORE you test. Print several, you'll run many experiments.

1 · We believe that...
which client · which problem · what they'll pay for

2 · We'll test it by...
the cheapest real test you can run

3 · We're right if...
a clear number decided in advance (e.g. 3 of 5 say yes)

4 · Then we'll... persevere · pivot · stop
circle one, based on the number

Run the test by: ___ / ___ / ___ Result: _____

1. **Decide the number first**, the pass/fail bar, before you test. It stops wishful thinking.
2. **Keep it small, cheap and real**, the fastest honest signal, not a perfect launch.
3. **Chase money over praise**, money > committed time > email > a polite "great idea".

Before you test, write it down. After you test, the card, not your hopes, decides what happens next.

PRINTABLE TEMPLATE · 2 OF 4

Discovery Interview Guide

The material for facilitator activity **F1 · Discovery Conversations**. Ask about their problem, questions only, no pitch.

Discovery Interview Guide

SHIFT · print & fill

Ask about their problem, not your idea. Questions only in round one; make no offer.

Prompt questions (ask, then be quiet and listen)

1. **"What's the hardest part of ...?"**
2. **"Tell me about the last time you dealt with this."**
3. **"What have you tried so far? How did it go?"**
4. **"How much time / money does this cost you?"**

Five real people I'll talk to this week, and how I'll reach them

1. _____
2. _____
3. _____
4. _____
5. _____

1. **Ask, then be quiet.** The client's real words about their problem are the gold.
2. **No offer in round one.** Resist the pitch reflex, you're collecting evidence, not selling.
3. **Five real people** beat a hundred guesses. Plan who, and how you'll reach them.

Talk to people about their problem, not your idea, and write down the exact words they use.

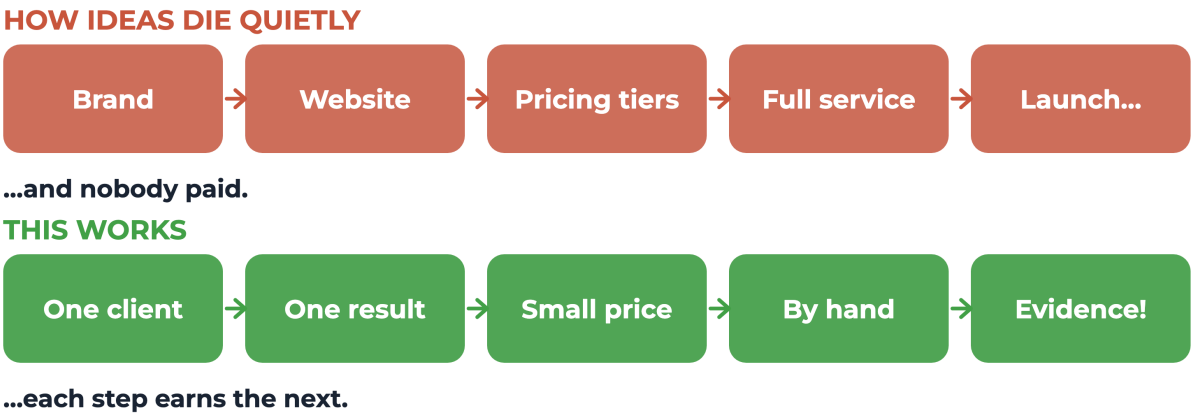
PRINTABLE HANDOUT · 3 OF 4

The MVP Handout, “not this / this”

A keep-able reference for **F2 · Strip It to the MVP**. Print it and pin it up while you cut your dream service down.

The Smallest Version Worth Testing

The smallest real offer a client would pay for today, delivered by hand, no systems behind it.



1. **Shortest list wins.** You're finding the single most valuable slice, not a smaller dream.
2. **Delivered by hand.** No brand, no website, no systems, those are earned later.
3. **One real, even small, price.** If someone would pay for just this, today, it's an MVP.

Don't build the whole studio. Sell one real result, to one real client, by hand, today.

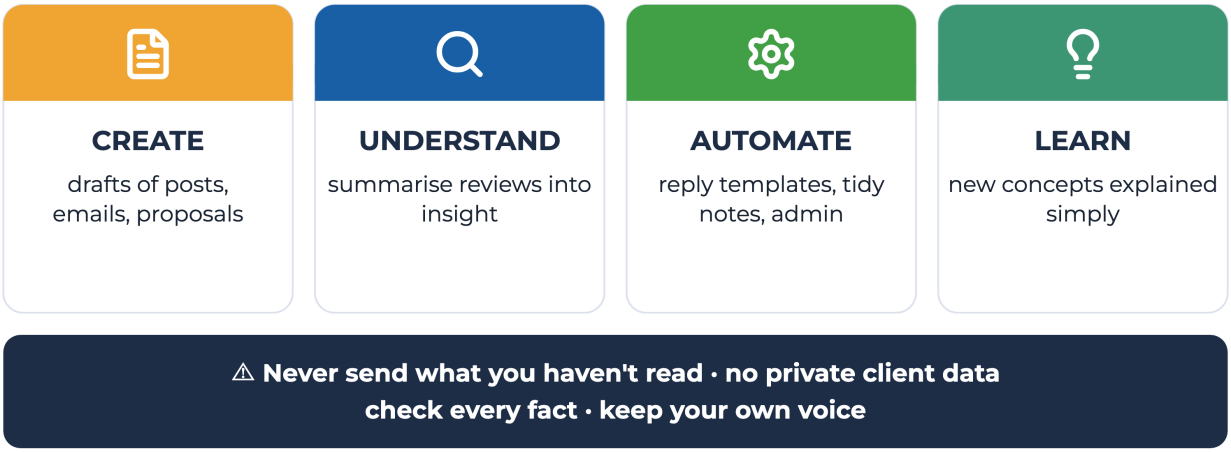
PRINTABLE HANDOUT · 4 OF 4

The AI-Tools Map

A keep-able reference for **F3 · AI in Practice**. Four jobs to delegate to AI, with the rules that keep you in charge.

Where AI Actually Helps a Freelancer

Delegate the draft, the admin and the research, keep judgement and the final call.



⚠ **Never send what you haven't read · no private client data**
check every fact · keep your own voice

1. **AI drafts, you decide.** Read and edit everything before it leaves your hands.
2. **No private client data** in tools you don't control, this rule is non-negotiable.
3. **Keep your own voice.** Use AI to start, then make it sound like you.

AI is a confident intern, fast and helpful, but you are the boss who checks the work.