STRETCHLEARN · WORKBOOK

A/B Testing & Experimentation — Workbook

This workbook turns the course into a real experiment you can run and defend. Each section mirrors one course module with hands-on exercises, fill-in worksheets, and checklists. Pick one page or flow of your own — a landing page, signup form, or checkout step — and carry it through every section. You will finish with a sized, documented A/B test, a scored backlog, and a results archive your team can reuse.

What Experimentation Is and Why Most Tests Lie

Internalise what a controlled test proves, audit your past testing for the seven traps, and confirm your traffic is high enough to test at all.

Exercise: Spot the Broken Test

For each scenario below, decide whether the test is valid or which specific trap it falls into. This builds the reflex of catching design flaws before you launch your own.

• A team ran version A in March and version B in April, then declared B the winner because April converted higher. Valid or which trap?

• A test crossed 95 percent significance on day 3 after 60 conversions and was stopped immediately. Valid or which trap?

• A test tracked 18 metrics and shipped the variant because time-on-page rose, though the primary signup metric was flat. Valid or which trap?

• Traffic arrived split 55/45 instead of the intended 50/50. What does this signal and what should you do?

Worksheet: Can I Even Test This? Traffic Gate

Before adding any idea to your backlog, fill this in for the page you intend to test. If you cannot clear the traffic floor, route the idea to qualitative research instead.

Page or flow to be tested

Weekly visitors to that page

Current weekly conversions on the primary action

Is the change reversible and incremental? (yes / no)

Does the outcome happen within days, not a long sales cycle? (yes / no)

Decision: A/B test it, or use qualitative research (session recordings, 5-user usability test, interviews)?

Checklist: Trap-Avoidance Pre-Flight

[ ] I will run A and B simultaneously, not one after the other.
[ ] I have named a single primary metric and will not switch it after launch.
[ ] I will set a sample size and end date before launching and not peek-and-stop.
[ ] I will run at least two full weekly cycles to defuse the novelty effect.
[ ] I will check the traffic split for sample ratio mismatch and pause if it is off.

The Statistics You Actually Need

Practise stating the five inputs out loud, compute a real sample size and runtime, and learn to read a confidence interval honestly.

Worksheet: State Your Five Inputs

Fill in the five numbers a sample-size calculator needs for your chosen page. Be honest about the MDE — set it to the smallest lift that would actually be worth shipping, not the smallest you can dream of.

Baseline conversion rate (current rate, e.g. 4%)

Minimum detectable effect / MDE (smallest worthwhile lift, e.g. 10% relative)

Statistical significance (conventionally 95%, alpha 0.05)

Statistical power (conventionally 80%)

Number of variations (e.g. 2 for a simple A/B)

Exercise: Calculate Sample Size and Runtime

Take your five inputs to a free calculator (Evan Miller's Awesome A/B Tools, Optimizely, or VWO) and record the answers. Then try a second scenario with a larger MDE and see how dramatically the cost drops.

• Calculator used, and required visitors per variation it returned.

• Total sample size across all variations, and your weekly traffic, then the runtime in weeks.

• Re-run with double the MDE (e.g. 20% instead of 10%): what is the new sample size and runtime?

• Final planned sample size and end date you will write into the test ticket (longer of: calculated sample, or two full weeks).

Worksheet: Read the Confidence Interval

For each result below, write whether you would ship, keep the control, or call it inconclusive, and why. The width and whether it crosses zero matter more than the headline number.

Result 1: +8%, 95% CI from +2% to +14% — decision and reason

Result 2: +8%, 95% CI from -3% to +19% — decision and reason

Result 3: +0.1%, 95% CI from +0.05% to +0.15%, highly significant — decision and reason (practical significance?)

Your own test's expected result range and what verdict each end of the interval would imply

Checklist: Statistics Sanity Check

[ ] I can say in one sentence what a p-value does and does not tell me.
[ ] I set power to at least 80% so a real effect is likely to be detected.
[ ] My MDE is the smallest lift worth shipping, not an unrealistically small one.
[ ] I will read the confidence interval, not just the point estimate and the significance flag.
[ ] I know whether a significant result is also practically worth the cost to ship.

Designing and Running a Clean Test

Turn an observation into a sharp hypothesis, score it against competing ideas, pick a tool, and run the experiment without contaminating it.

Worksheet: Write Your Hypothesis

Fill the template completely for your chosen test. Every blank must be concrete — real evidence, the exact change, one metric, an expected size, and the reasoning. If you cannot fill a blank, you are not ready to build.

Evidence we observed (from analytics, recordings, support, or usability testing)

The specific change we will make

The audience it applies to

The single primary metric it should move, and the expected direction and size

The reasoning — why we think this change causes that effect

We will know we are right when we see (measurable result)

Exercise: Score the Backlog with ICE

List at least five test ideas and score each 1-10 on Impact, Confidence, and Ease, with one sentence of justification per number. Average the three and sort descending to set your run order.

- Idea 1: Impact / Confidence / Ease scores, one-line reason for each, and the average.
- Idea 2 through 5: same scoring, each number justified in a sentence.
- Your ranked run order, top to bottom, after averaging.
- Which page or surface each top idea touches, to make sure two tests will not collide.

Worksheet: Pick Your Tool and Engine

Choose a testing platform and confirm how its statistics work, so you know whether peeking is allowed. Match the tool to who runs tests and what you change.

Who will run tests (marketer-led visual editor, or engineer-led feature flags)?

Chosen platform (e.g. GrowthBook, VWO, Optimizely, PostHog, Statsig, Convert)

Statistical engine: frequentist (fix sample, no peeking) or Bayesian/sequential (can monitor)?

Client-side (possible flicker) or server-side (no flicker, needs dev)?

Free tier or budget confirmed for this tool

Checklist: Clean-Run Launch Checklist

[ ] Both variants QA'd on major browsers and on mobile before launch.
[ ] Only one element changes in this simple A/B (multi-element changes saved for a powered multivariate test).
[ ] Sample ratio mismatch check planned — I will confirm the split is near 50/50 after launch.
[ ] No overlapping test runs on the same page at the same time.
[ ] Sample size and end date are written into the ticket, and I will not edit the test mid-flight.

Reading Results and Building a Program

Read your result in the right order, turn every outcome into an action and a documented learning, and stand up the rituals that make testing a habit.

Worksheet: Read the Result in Order

After your test reaches its planned finish, answer these in sequence. Do not jump to the verdict before confirming the test actually finished and the interval supports it.
Did you reach the planned sample size and runtime? (yes / no — if no, result is provisional)

Is the primary metric significant against your threshold or probability-to-beat?

Confidence interval range — does it cross zero?

Is the effect practically worth shipping given engineering, support, and risk?

Verdict: clear win / clear loss / inconclusive

Any segment finding to log as a future hypothesis (not as a result to ship)?

Exercise: Turn the Outcome Into the Next Move

Whatever the verdict, write the concrete next action and the learning it produced. This is where one test becomes a stream of compounding insight.

• If it won: roll out to 100%, plus how long you will monitor for novelty decay, plus the follow-up hypothesis it suggests.

• If it lost or was flat: what assumption was wrong, and what does that teach the next idea in this area?

• The one-sentence learning from this test, win or lose.

• The follow-up hypothesis you will add to the backlog because of this result.

Checklist: Document-the-Test Checklist

[ ] Hypothesis as originally written, with evidence and predicted effect, is recorded.
[ ] Screenshots of control and variant are attached.
[ ] Numbers logged: sample size, runtime, primary-metric result, confidence interval, verdict.
[ ] Decision and reasoning (including any practical-significance call) are written down.
[ ] Learning and follow-up hypothesis are captured in the shared archive.

Worksheet: Stand Up the Program

Define the rituals and health metrics that turn occasional tests into a sustainable cadence. Pick targets you can actually keep.

Sustainable cadence (e.g. 1-2 tests per surface per month)

Experiment review meeting — frequency and who attends

Where the results archive lives (Notion, Confluence, GrowthBook, shared sheet)

Named owner accountable for calendar, QA, and statistical rigour

North-star metric most tests ladder up to

Program health targets: velocity (tests/month), expected win rate (~20-33%), learning rate

Your Action Plan

1. Choose one page or flow of your own to carry through the whole plan, and confirm it clears the traffic floor.
2. Gather evidence from analytics, session recordings, or a 5-user usability test to ground at least one hypothesis.
3. Write the hypothesis using the full template — evidence, change, audience, metric, expected size, reasoning.
4. Score your backlog of ideas on ICE or PIE with a one-line justification per number, and set the run order.
5. State your five inputs and use a free calculator to get the required sample size and runtime.
6. Write the sample size and end date into the test ticket so you cannot move the finish line later.
7. Pick a tool, confirm whether its engine is frequentist or Bayesian, and QA both variants before launch.
8. Launch one clean A/B test, check for sample ratio mismatch, and do not peek-and-stop or edit mid-flight.
9. At the planned finish, read the result in order and reach an honest verdict on the primary metric.
10. Document the test in your archive with screenshots, numbers, decision, and the learning, then queue the follow-up hypothesis.