

Idea → Product: One-Pager Playbook

Version 2025-08-08

1) Ideation & Validation (3–10 days)

- Goal: Kill bad ideas fast; keep only the ones with signal.
- Activities: Problem statement, user & pain points, 10 interviews or landing-page test, quick market & competitor scan.
- Deliverables: 1-page Idea Brief, Assumptions to Test, Validation Results.
- Exit: $\geq 30\%$ strong problem resonance OR $\geq 5\text{--}10\%$ landing-page conversion.

2) Scope & Plan (2–5 days)

- Goal: Define a first release (MVP) with hard guardrails.
- Activities: MVP job-to-be-done; lightweight PRD; user & system diagrams; risks & timeboxed spikes.
- Deliverables: MVP PRD, MVP-only backlog, Risk Register, Spikes list.
- Exit: Scope fits 4–6 weeks, clear non-goals and success metric.

3) Design & Architecture (3–7 days)

- Goal: De-risk UX and tech before heavy coding.
- Activities: Wireframes validated with 3–5 users; select stack & libraries (+Plan B); repo/CI/testing scaffold; initial ADRs.
- Exit: `git clone` & make dev` runs; smoke tests & CI green.

4) Build in Iterations (1–2 weeks each)

- Goal: Ship small, tested slices; shrink risk continuously.
- Cadence: Plan → Build → Test (unit+integration+exploratory) → Review → Demo → Retro.
- DoD: Acceptance criteria met; tests passing; telemetry added; docs updated; behind a feature flag.

5) Review & Adjust (0.5–1 day/iteration)

- Rule: No mid-iteration scope changes except Sev-1.
- New ideas/libs → Spike with decision date & kill criteria; schedule next iteration.
- Deliverable: Updated backlog; ADRs for accepted changes.

6) Pre-Launch Testing (2–7 days)

- Goal: Confidence to ship.
- Run regression (automated + manual), UAT with 5–10 users, performance & security sanity, accessibility quick pass.
- Exit: 0 Sev-1/2; rollback plan tested; dashboards live.

7) Launch & Feedback

- Canary/staged rollout with flags; analytics & error tracking on.
- Deliverables: Metrics snapshot; bug/feedback triage.

8) Post-Launch Iteration

- Keep 1–2 week cycles; prioritize via RICE; reserve ~15% capacity for tech debt; monthly roadmap review.

Decision Aid: Pivot vs Continue

- Does the change unblock a critical MVP risk? If yes, Spike next sprint; if no, defer.

- Is current approach failing acceptance/performance targets? If yes, Spike → ADR → plan migration.
- Will the change push MVP beyond 6 weeks? If yes, defer unless existential.

Testing Strategy (CI Gates)

- Unit (fast, many), Integration (boundaries), Contract (services), E2E (few critical).
- CI: lint/format, tests, coverage gate ($\geq 70\%$ non-decreasing), build & scan, optional SAST/DAST.