🔳 AI Sprint Playbook (Lightweight, Defensible, Scalable)

Purpose

This playbook defines **how AI is used safely and effectively within a sprint**, without introducing bureaucracy or slowing delivery.

AI is treated as:

> a productivity tool operating under explicit constraints, with humans retaining ownership and accountability.

The AI Sprint Loop (One Page)

Every AI-assisted sprint follows this loop:

```
1. Scope the sprint (before work)
2. Execute with constraints
3. Review outcomes (after work)
4. Communicate results
5. Log decisions (only if needed)
```

No step is optional. No step is heavyweight.

1️⃣ Before the Sprint — Scope the Work

Artifact: **One-Page AI Sprint Scope Contract**

Why it exists

- Prevents AI overreach
- Makes safe behavior the default
- Clarifies ownership *before* work begins

Filled out when

- Sprint planning starts
- Before any AI execution

Key questions it answers

- What is AI allowed to do?
- What is AI forbidden from doing?
- Who owns outcomes?
- How risky is this sprint?

If something is not explicitly allowed → **AI must not do it**.

---

2 During the Sprint — Execute with Constraints

Execution Rules

- AI operates only within the declared role(s)
- AI never expands scope implicitly
- AI does not make irreversible decisions
- Humans review intermediate outputs as needed

Design Principles

- Prefer **vertical slices** over horizontal work
- Bound tasks by **single responsibilities**
- Keep blast radius small and reversible

If execution starts to feel ambiguous → **pause and rescope**.

---

3 After the Sprint — Review Outcomes

Artifact: **One-Page AI Review Report Checklist**

Why it exists

- Proves the sprint is defensible
- Separates "it works" from "it's safe"
- Makes risk visible

Completed when

- Sprint work is complete
- Before merge / release

Core questions it answers

- Was scope respected?
- Did behavior change?
- What validation was performed?
- What risk remains?
- Who approved the outcome?

If this checklist feels hard to complete → the sprint was likely underspecified.

---

4 Communicate — Translate Review into Trust

Artifact: **Stakeholder Update (Derived)**

This is **not a new document** — it is a translation of the review checklist.

Communication pattern

1. What changed
2. What did not change
3. How we know

Audience-aware

- Executives → outcomes + stability
- Engineering → safeguards + validation
- Compliance → traceability + control

Avoid AI agency language. Preserve accountability.

---

5 Log Decisions — Only When Necessary

Artifact: **One-Page Decision Log**

Used only when

- A decision is hard to reverse
- New assumptions are introduced
- Behavior meaningfully changes
- Future contributors might ask "why?"

Not used for

- refactors with no behavior change
- formatting
- tests
- documentation

Overuse kills signal. Use sparingly and intentionally.

Risk Management Rules (Always On)

AI Is Generally Allowed For

- bounded refactors
- test generation
- documentation
- debugging with known errors
- scoped data transformations

AI Requires Explicit Approval For

- metric definition
- data semantics
- system architecture

- production decision logic
- retraining policies

Heuristics to Remember

- If rollback is expensive → slow down
- If review requires re-learning the system → rescope
- If AI has touched the same component for 3 sprints → stabilize

---

Stabilization Sprints (When Needed)

Trigger a stabilization sprint when:

- confidence drops despite passing tests
- refactors outpace understanding
- assumptions feel implicit
- new engineers struggle to onboard

Stabilization sprints:

- add no features
- focus on documentation, invariants, validation
- allow AI only in review/analysis mode

The Core Governance Rules (Memorize These)

1. AI is a tool, never the owner
2. Scope must be explicit before execution
3. Behavior changes require extra validation
4. Risk accumulates across sprints, not just within them
5. Communication preserves trust, not transparency theater

What This Playbook Enables

With this in place, teams can:

- move fast without fear
- scale AI usage safely
- onboard new contributors easily
- satisfy governance without bureaucracy
- reuse the same system across projects

Including **Crypto Project Risk Radar**.

Final Note

This playbook works because it:

- constrains behavior, not people
- relies on artifacts, not meetings
- encodes judgment, not rules-for-rules' sake

Most organizations try to *govern AI*. You've built a system that **governs work that uses AI** — which is the only approach that actually scales.