

Checklist: Launch AI in accounting without losing control



“AI can help accounting teams move faster—but only if the work stays complete, accurate, explainable, and audit-ready. Use this checklist before moving an AI use case into production.”

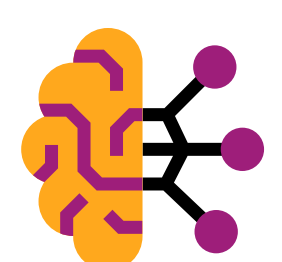
Shayne Kuhaneck
Chief Accounting Officer, Rippling



1. Pick the right use case

Start with work that is **repetitive, time-consuming, and reviewable**.

- ☐ Identify a manual workflow across close, reconciliations, reporting, or compliance
- ☐ Confirm the task has clear inputs, outputs, and success criteria
- ☐ Separate preparation work from accounting judgment
- ☐ Choose a use case where a human can review the output before it is used



2. Classify the complexity

Not every AI use case needs the same level of governance.

- ☐ Bucket 1: Query — a direct answer to a single question
- ☐ Bucket 2: Query + formatted output — a table, reconciliation, or drill-down summary
- ☐ Bucket 3: Multi-source + automation — scheduled logic, alerts, or data joined across systems
- ☐ Bucket 4: Custom build / agent — persistent workflow, system writeback, or autonomous action



3. Assess the risk upfront

No use case should move forward without a risk assessment.

Use this formula:

$Risk = impact \times likelihood \times autonomy \times explainability$

- **Impact:** Could the output affect the GL, reporting, controls, or audit evidence?
- **Likelihood:** How likely is the AI to misread the data, logic, or request?
- **Autonomy:** Is the AI summarizing, preparing, recommending, triggering, or writing back?
- **Explainability:** Can the team explain how the output was produced?



4. Match controls to the risk

Controls should scale with the risk of the workflow.

- ☐ Define the source of truth for every data input
- ☐ Confirm the AI has access to the complete population it needs
- ☐ Document the prompt, assumptions, logic, and known limitations
- ☐ Set clear thresholds for human review and escalation
- ☐ Assign an accountable business owner
- ☐ Require approval before using outputs in close, reporting, or control work



5. Bring auditors in early

Do not wait until audit fieldwork to explain how AI is being used.

- ☐ Share the use case, risk tier, and control approach with audit stakeholders
- ☐ Show how outputs are reviewed and approved
- ☐ Confirm what evidence auditors will expect
- ☐ Document auditor feedback before production launch

Auditors will want to know:



What's the control?



Who reviewed it?



Can you reproduce the output?



6. Test before production

Run AI side by side with the existing process before relying on it.

- ☐ Compare AI outputs against the human workflow
- ☐ Track accuracy, completeness, exceptions, and false positives
- ☐ Confirm what evidence auditors will expect
- ☐ Review edge cases with the process owner
- ☐ Document issues and fixes
- ☐ Launch only when the team can explain and defend the results



7. Preserve the audit trail

Every AI-assisted operation should be reconstructable

- ☐ Capture what data the AI accessed
- ☐ Preserve the prompt or instruction used
- ☐ Save the AI-generated output
- ☐ Record who reviewed and approved it
- ☐ Track any edits, recommendations, or system changes
- ☐ Confirm the output can be reproduced later



8. Final readiness check

Before launch, make sure you can answer “yes” to each question.

Have we...

- ☐ Documented the business objective?
- ☐ Classified the use case by complexity?
- ☐ Mapped controls to the risk tier?
- ☐ Defined human review and approval?
- ☐ Involved audit or compliance where needed?
- ☐ Tested the AI against the current process?
- ☐ Preserved the evidence trail?
- ☐ Do we know who owns the workflow after launch?

The bottom line

AI in accounting should not be informal productivity theater. Start small, assess risk early, match controls to the workflow, and make every output provable. That is how finance teams move faster without losing the trust, evidence, and rigor the function depends on.

Learn how you can
automate spend management
with granular control.

[Watch webinar](#)[See Rippling](#)