

SIG AI TOOLKIT

Rovo for Jira: *The Quality Audit Cheat Sheet*

Copy and paste prompts to audit backlog health, story quality, acceptance criteria, dependencies, metrics, risks, and compliance. Built for every level, from the board to the ART to the portfolio.

Who it is for:

SCRUM MASTERS & POS

RTES & PROGRAM LEADERS

PORTFOLIO & PMO LEADERS

DEVELOPERS & TESTERS

[BRACKETS] = fill in your own details

Where Rovo lives in Jira

- **Rovo Chat.** Ask questions about your Jira work in plain language and get summaries, tables, and drafts back.
- **Agents in Jira.** Mention an agent in a work item comment, assign work to it, or add it to a workflow status.
- **Rovo search.** Find work items by describing them in plain language instead of writing JQL.
- **Skills and Studio.** Save a repeatable audit as a Rovo skill, or build a custom agent in Rovo Studio.

The three part prompt formula

1. **Scope.** Name the board, sprint, project, filter, fix version, PI, or issue key.
2. **Standard.** Paste your Definition of Ready, Definition of Done, or story template.
3. **Output.** Ask for a table, a score, a top five, or an executive summary.

Example: Audit the top 30 items in the [PAYMENTS TEAM] backlog against our definition of ready: [PASTE YOUR DoR]. Return a table with a red, yellow, or green score and a one line fix.

Agents worth knowing

- **Readiness Checker.** Checks a work item against your definition of ready and suggests fixes.
- **Work Item Planner.** Breaks epics and large work into smaller, actionable items.
- **Work Item Organizer.** Moves items into sprints, assigns them to epics, and updates them.
- **Jira Triage Agent.** Recommends next steps, field updates, duplicates, and links.
- **Jira Delivery Agent.** Spots delivery risks and turns activity into updates.
- **Jira Theme Analyzer.** Finds themes across work items or groups them into epics.
- **Progress Tracker.** Gives a real time view of in flight projects and priorities.
- **Bug Report Assistant.** Helps write clear bug reports with the details triage needs.

Quick audit scorecard: what good looks like

AREA	GREEN LOOKS LIKE	RED FLAGS
Backlog	Next two sprints refined, estimated, and linked to features.	Items with no AC or parent, or stale for 90+ days.
User stories	Clear user and outcome, small enough for one sprint.	Tasks dressed up as stories, or oversized work.
Acceptance criteria	Testable and clearly pass or fail.	Missing, vague, or untestable criteria.
Dependencies	Every dependency linked with a need by sprint.	Hallway dependencies, or providers landing late.
Metrics	Stable trend and low carryover.	Rising carryover and aging work in progress.
Status reporting	One set of Jira data feeds every level.	Reports rebuilt by hand every week.
Risks	Owned, ROAMed, and updated.	Risks that only live in comments.
Missing stories	Requirements trace to stories, NFRs included.	Specs with no matching stories.
Compliance	DoD evidence linked on every done item.	Done with no PR, test, or approval.

01 Backlog Quality

Is the backlog ready to plan from, or is it a junk drawer?

SM & PO Audit the top 30 ranked items in the [BOARD] backlog. For each one, flag a missing description, missing acceptance criteria, no estimate, no parent epic or feature, or no update in 60 days. Return a table with a red, yellow, or green score and a one line fix.

SM & PO Find duplicate or overlapping work items in the [PROJECT] backlog. Suggest which ones to merge or close and why.

SM & PO List backlog items older than 90 days that have never been in a sprint and have no recent comments. Recommend keep, rework, or close for each.

SM & PO Do we have two sprints of refined work? Compare the estimated, ready items at the top of [BOARD] to our average velocity of [X] points.

RTE Across every team on [ART NAME], score backlog health by team: the percent of the top 20 items that are estimated, have acceptance criteria, and link to a feature.

PMO Which epics in [PORTFOLIO PROJECT] have no child features or stories, and which have children but no owner?

03 Acceptance Criteria

If you cannot test it, you cannot call it done.

SM & PO List every story in [SPRINT or BOARD] that has no acceptance criteria, or fewer than [2] criteria.

SM & PO Rewrite the acceptance criteria on [ISSUE KEY] in Given, When, Then format. Keep each one testable and add edge cases we missed.

SM & PO Score the acceptance criteria on the stories in [SPRINT] for testability. Is each criterion measurable, unambiguous, and clearly pass or fail?

RTE For features planned in [PI], check that each one has a benefit hypothesis and acceptance criteria. List the ones that do not.

DEV & TEST From the acceptance criteria on [ISSUE KEY], draft test cases for the happy path, negative paths, and boundary conditions.

DEV & TEST Compare the acceptance criteria on [ISSUE KEY] to the linked Confluence page [PAGE]. Flag anything in the spec that the criteria do not cover.

02 User Story Quality

Clear stories make faster teams. Vague ones make rework.

SM & PO Review [ISSUE KEY] against the INVEST criteria. Score each letter from 1 to 5, explain any low score, and rewrite the story so it passes.

SM & PO @Readiness Checker check [ISSUE KEY] against our definition of ready: [PASTE YOUR DoR].

SM & PO List stories in [SPRINT] that read like tasks instead of user value, with no clear user or outcome. Suggest a rewrite for each.

SM & PO Find stories in [SPRINT] estimated above [8] points and propose a split into smaller stories that each still deliver value.

DEV & TEST Explain [ISSUE KEY] in plain language: what we are building, for whom, what is out of scope, and the open questions I should ask the PO before I start.

DEV & TEST Is anything in the description of [ISSUE KEY] vague or contradictory? List the questions a tester would ask.

04 Dependencies

Most missed dates start as a dependency nobody tracked.

SM & PO List every work item in [SPRINT] that is blocked by, or depends on, another team. Show the blocking item, its owner, its status, and its planned sprint.

SM & PO Which items in our sprint link to work in another project that has not started yet?

RTE Build a dependency table for [ART] in [PI]: requesting team, providing team, work item, needed by sprint, and current status. Highlight any where the provider lands after the need date.

RTE Which cross team dependencies have no linked work item, or a linked item with no sprint assigned?

PMO Which epics in [PORTFOLIO PROJECT] depend on the same shared platform, team, or vendor? Where is a single team becoming a bottleneck?

DEV & TEST What is linked to [ISSUE KEY]? Summarize anything that could block me or that my change might break.

Performance Metrics

05 Use Rovo to spot the trend and ask the right question. Confirm hard numbers in Jira reports before you share them.

SM & PO Summarize the last [5] sprints on [BOARD]: committed versus completed, carryover, and scope added after the sprint started. What is the trend?

SM & PO Which work items in [SPRINT] have sat in the same status for more than [3] days? Group them by status and suggest a next step for each.

SM & PO Compare cycle time for stories versus bugs on [BOARD] over the last [90] days and call out the outliers.

RTE For [ART], compare planned versus completed work by team for [PI]. Which teams consistently over commit, and what pattern do you see?

PMO Which epics in [PORTFOLIO PROJECT] have grown the most in scope since they started? Compare their child item counts over time.

DEV & TEST How much of my work last sprint was unplanned? List the items added after the sprint started.

Missing Stories and Gaps

08 What should be on the board that is not?

SM & PO Read feature [ISSUE KEY] and its acceptance criteria. What stories are missing? Consider error handling, logging, security, accessibility, data migration, documentation, and testing.

SM & PO Compare the requirements on the Confluence page [PAGE] to the stories under [EPIC]. List every requirement with no matching story.

RTE For features in [PI], list any with zero child stories, and any still open even though every child story is done.

RTE @Jira Theme Analyzer group the stories with no parent in [PROJECT] into themes and suggest the epic or feature each one belongs to.

PMO Which approved epics in [PORTFOLIO PROJECT] have no features planned for the next PI?

DEV & TEST What non functional work is missing from [EPIC]? Think performance, security review, monitoring, alerts, and runbooks.

Risk Summary

07 Pull the risks out of the comments and into the light.

SM & PO Scan the comments and descriptions in [SPRINT] for signals like blocked, waiting, unclear, delayed, or risk. Summarize the real risks and who owns each one.

SM & PO @Jira Delivery Agent what delivery risks do you see in [PROJECT or FIX VERSION]?

RTE Build a ROAM table for [ART] in [PI]: risk, impact, owner, and a proposed status of Resolved, Owned, Accepted, or Mitigated.

RTE Which risks raised at PI Planning are still open with no update in the last two weeks?

PMO Summarize the top 10 risks across [PROJECTS] in a one page view for [AUDIENCE]. Rank them by impact on our strategic goals.

DEV & TEST What is the riskiest change in [SPRINT] from a testing or rollback point of view, and why?

Compliance and Governance

09 Make the audit trail easy to prove, not painful to rebuild.

SM & PO List the done items in [SPRINT] that do not meet our definition of done: [PASTE YOUR DoD]. Show what is missing on each one.

SM & PO Find items moved to Done in [SPRINT] with no linked pull request, test evidence, or approval comment.

RTE Check that every item in [FIX VERSION] has these required fields completed: [FIELD LIST]. Return only the gaps.

PMO For epics labeled [REGULATED LABEL], list any missing approvals, sign offs, or evidence links. Format it as an audit table.

PMO Which epics in [PORTFOLIO PROJECT] changed priority or scope after approval, and is there a comment explaining why?

DEV & TEST Before I close [ISSUE KEY], check it against our definition of done and tell me what is missing.

BOARD LEVEL

SM & PO Give me a standup summary for **[BOARD]**: what moved since yesterday, what is stuck, and what is at risk of not finishing this sprint.

SM & PO What changed on **[BOARD]** since **[DATE]**? Only meaningful changes: status moves, new blockers, and scope added.

TEAM LEVEL

SM & PO Write a Sprint Review summary for **[SPRINT]**: the sprint goal, what we finished, what carried over and why, and what we will demo.

SM & PO Draft a weekly team status: overall green, yellow, or red, three highlights, top risks, and asks for leadership. Keep it under 150 words.

PROGRAM / ART LEVEL

RTE Create an ART status for **[ART]** in **[PI]**: progress toward PI objectives by team, features done versus planned, the top 5 risks, dependencies at risk, and decisions needed.

RTE Which features in **[PI]** are unlikely to finish, based on the remaining stories and the sprints left? Explain your reasoning.

RTE Draft talking points for this week's Coach Sync and PO Sync for **[ART]**.

PORTFOLIO LEVEL

PMO Summarize every epic in **[PORTFOLIO PROJECT]** for an executive audience: on track, at risk, or off track, with one sentence on why. No jargon.

PMO Which strategic themes or goals have the least active work tied to them right now?

Put your audit on autopilot

- **Save it as a skill.** In RoVo Chat, try: "Create a skill that runs our backlog quality audit on **[BOARD]** using these rules: **[YOUR RULES]**." RoVo asks clarifying questions and saves it for reuse.
- **Build a custom agent.** In RoVo Studio, give an agent your Definition of Ready, Definition of Done, and story template so every team audits against the same standard.
- **Put an agent in the workflow.** Add an agent to a workflow status, like Ready for Dev, so a readiness check runs automatically when an item arrives there.
- **Wire it into automation.** Use RoVo agents in Jira automation rules, like flagging items that look blocked based on their comments.

Pro tips for better answers

- **Name the scope.** RoVo works best with a board, sprint, project, filter, fix version, or issue key.
- **Paste your standards.** Your DoR, DoD, and story template beat RoVo's best guess every time.
- **Ask for a table.** Tables are easier to scan, sort, and paste into Confluence.
- **Trust, then verify.** Spot check numbers against Jira reports before you share them up the chain.
- **RoVo sees what you see.** Results follow your Jira permissions, so leaders may get a wider view than teams.
- **Share the chat.** Shared chats show your team how you got to the answer.

Want this running across your ARTs?

SIG helps PMO and program leaders turn AI into a working part of how teams plan, deliver, and report. We can set up your standards, skills, and agents so every team audits the same way.

stephensinsightgroup.com
info@stephensinsightgroup.com
(616) 724-0297

RoVo features, agent names, and availability vary by Atlassian Cloud plan and your admin's settings, and Atlassian updates them often. Agent names and features on this page reflect Atlassian's published documentation as of October 2026. AI can be wrong, so review every output before you act on it or share it.

Sources: Atlassian Support: Out of the box RoVo agents, <https://support.atlassian.com/rovo/docs/atlassian-agents/>; Atlassian: AI in Jira, <https://www.atlassian.com/software/jira/ai>; Atlassian Blog: Introducing agents in Jira (Feb 2026), <https://www.atlassian.com/blog/rovo/ai-agents-in-jira>; Atlassian Blog: New in RoVo Chat (Sep 2026), <https://www.atlassian.com/blog/company-news/rovo-chat-new-features-september-2026>; Atlassian Support: Use RoVo to search for work items, <https://support.atlassian.com/jira-software-cloud/docs/use-atlassian-intelligence-to-search-for-work-items/>.

© 2026 Stephens Insight Group LLC. RoVo, Jira, and Confluence are trademarks of Atlassian. This cheat sheet is not affiliated with or endorsed by Atlassian.