AI WORKFLOW FIELD GUIDES

AI Code Change Auditor Before They Ship

A practical guide to auditing AI-generated code changes before you trust, merge, or deploy them.

Figure: Operating loop visual: scope check, false-completion review, validation evidence, regression risk, and receipt continuity.

You are not buying generic AI advice. You are getting a field guide for checking AI-generated code work before it is trusted, merged, deployed, or called done.

Version: v0.1.0 premium guide
Product: AI Code Change Auditor
Primary outcome: catch fake-complete code changes before they ship.

AI WORKFLOW FIELD GUIDES

Before They Ship: AI Code Change Auditor

A premium field guide for verifying AI-generated code changes before the work is trusted, merged, deployed, or called done.

FIELD GUIDE PROMISE

Turn AI coding-agent claims into evidence-backed human review decisions before code is accepted as complete.

Apotheosys
Author: John Barros

USE BOUNDARY

Human review stays in the loop

This field guide translates the AI Code Change Auditor skill pack into a human-facing operating manual. It does not certify code, guarantee correctness, replace human code review, or authorize production deployment.

Use it to structure evidence review, scope checks, missing-test detection, false-completion triage, and receipt-backed decisions.

> **REQUIRED BOUNDARY**
>
> Do not mark a change PASS unless the requested outcome, changed files, validation evidence, and remaining risks have been inspected.

TABLE OF CONTENTS

How to use this guide

OPERATOR NOTE

Read chapters 1 through 6 first. After that, use the examples and templates as a working manual whenever an AI coding agent claims a task is done.

CHAPTER 01

The failure pattern this guide solves

The dangerous moment is not when an AI coding agent writes code. The dangerous moment is when the agent says the job is finished and nobody checks whether the changed files, validation evidence, and original request still line up.

The AI Code Change Auditor exists for that gap. It gives a buyer a repeatable way to convert a claimed completion into an evidence-backed review record.

Fake completion

The agent says tests pass, the UI works, or the bug is fixed, but the evidence is missing or incomplete.

Scope drift

The diff touches adjacent files, changes behavior outside the request, or rewrites more than the outcome required.

Silent test gap

The changed code path is not covered by the tests shown, or validation does not match the claim.

Unsupported done language

The final response sounds confident even though screenshots, logs, browser checks, migrations, or fixtures were never inspected.

BUYER OUTCOME

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

CHAPTER 02

Core doctrine: evidence before verdict

The skill does not reward confident summaries. It rewards inspected evidence. A change can be plausible and still fail the audit if the requested behavior was not verified.

Doctrine	What it prevents	How it shows up in the audit
No fake pass	Premature acceptance of unverified agent work	PASS is allowed only when the diff, scope, tests, and risks support it.
No scope blindness	Unrelated edits hidden inside a fix	Every changed file is mapped back to the requested outcome.
No silent test gap	Passing tests that miss the changed behavior	Missing validation is named explicitly.
No unsupported certainty	Treating inference as inspected fact	Facts, assumptions, estimates, and unavailable evidence are separated.

Doctrine	What it prevents	How it shows up in the audit
Receipt before done	Loose final answers with no review trail	Every audit ends with evidence, limitations, blockers, and status.

OPERATING RULE

If the audit cannot identify what was requested, what changed, what evidence was reviewed, and what remains unvalidated, the result cannot be PASS.

CHAPTER 03

Workflow architecture

The workflow moves from claim to receipt. It is intentionally narrow: it is not a full security audit, not a general code review replacement, and not a production certification system.

1. Restate request
Separate the original user goal from the agent's completion language.

2. Inventory evidence
List diffs, changed files, tests, logs, screenshots, and missing inputs.

3. Map scope
Classify each change as in scope, borderline, unrelated, or overreach.

4. Scan failure modes
Look for false completion, regressions, missing validation, and hidden assumptions.

5. Issue receipt
Produce status, blockers, limitations, and next actions.

Audit area	Question it answers	Blocking when missing?
Requested change map	What was actually requested?	Yes, if the original task cannot be determined.
Changed file inventory	Which files changed and why?	Yes, if changed files or diff are unavailable.
Validation review	What evidence supports the claim?	Often. Missing tests can block PASS.
Receipt	What can a human trust next?	Yes. The skill requires a final receipt.

CHAPTER 04

Inputs, evidence, and missing context

The audit can accept partial input, but incomplete evidence lowers confidence. The field guide treats missing evidence as a first-class finding, not as a formatting issue.

Input	Why it matters	If missing
Original request or ticket	Defines the scope and non-goals.	Restate that the audit is evidence-limited.
Agent claimed completion	Shows what must be tested against reality.	Review the diff only; do not infer promises.
Changed files or diff	The core evidence for scope and behavior.	Do not mark PASS.
Tests and output	Shows whether the changed path was validated.	Name the missing validation explicitly.
Screenshots, logs, CI, or browser checks	Support UI, integration, and runtime claims.	Classify as NEEDS_VALIDATION when relevant.
Known constraints and non-goals	Prevents broad rewrites from hiding in the patch.	Ask for constraints or mark assumption.

EVIDENCE CLASSIFICATION

Label each item as provided, observed, inferred, unavailable, or user-confirmed. Do not let inferred evidence become a passing claim.

CHAPTER 05

Step-by-step audit runbook

This is the working sequence to run every time an agent says a code task is complete. Keep the language direct and review-oriented.

01

State the request

Write the requested change in one or two sentences. Include constraints, files that should not change, and the expected user-visible outcome.

02

Separate claim from proof

Record what the agent says happened, then list only what was actually inspected.

03

Inventory changed files

Map each changed file to the requested outcome. Flag files that are unexplained, broad, generated, or unrelated.

04

Review scope compliance

Classify work as in scope, borderline, or overreach. A plausible fix can still fail if it changes the wrong surface.

05

Check false completion

Look for missing routes, schemas, imports, fixtures, migrations, UI states, browser checks, test assertions, or TODOs.

06

Scan regression risk

Group concrete risks by severity. Keep recommendations tied to the requested change.

07

Review validation

Name exact commands run, output reviewed, and missing checks. Do not use a generic 'tests pass' claim as sufficient evidence.

08

Issue the verdict

Use PASS, NEEDS_FIX, BLOCKED, or PRODUCT_REVIEW_REQUIRED. Finish with the fix verification receipt.

CHAPTER 06

Verdict system and receipt logic

The verdict is not a vibe. It is the final compression of the audit evidence. If the evidence does not support PASS, the verdict must say so.

Verdict	Use when	What the human does next
PASS	Scope is clean, evidence is adequate, and no blocking risks remain.	Proceed to normal human review or merge policy.
NEEDS_FIX	Concrete implementation, validation, or scope issues remain.	Fix named issues and rerun the audit.
BLOCKED	Required evidence or environment access is missing.	Provide diff, logs, tests, screenshots, or original request.

Verdict	Use when	What the human does next
PRODUCT_REVIEW_REQUIRED	Technical implementation appears plausible but needs product, UX, or business-rule acceptance.	Escalate to a human owner for acceptance.

Fix Verification Receipt

- Skill: AI Code Change Auditor
- Request reviewed:
- Evidence reviewed:
- Files reviewed:
- Tests reviewed:
- Tests missing:
- Overreach found:
- Blocking issues:
- Status: PASS | NEEDS_FIX | BLOCKED | PRODUCT_REVIEW_REQUIRED
- Limitation:

CHAPTER 07

Failure modes and troubleshooting

The skill is built around recurring agentic failure modes. These are not generic best practices. They are the failure patterns that make AI-

generated code dangerous to accept without verification.

Adjacent-file drift

The agent changes nearby files to make the task look complete. Check whether every changed file maps to the original request.

Happy-path only

The fix handles one obvious case but leaves edge cases, empty states, permissions, loading states, or failure paths broken.

Weak tests

Tests were added but do not assert the requested behavior, or they pass without touching the changed code path.

Forgotten integration point

A route, schema, migration, generated type, import, config entry, or fixture was omitted.

Removal disguised as repair

The agent removes behavior, validation, or UI instead of fixing the requested issue.

Polished final answer, unfinished diff

The response says done while TODOs, placeholders, dead code, skipped checks, or stale artifacts remain.

TROUBLESHOOTING STANCE

If the audit finds a concrete issue, do not broaden into unrelated refactor advice. Keep the next action tied to the requested change.

CHAPTER 08

Validation and test review

Validation review is where false completion usually breaks. The skill requires the reviewer to name what was run, what output was inspected, what was not covered, and what should be run next.

Claim	Evidence needed	Common failure
Bug fixed	A reproduction or targeted test showing the failing condition now passes.	Only broad test suite output is shown.
UI works	Screenshot, browser check, Playwright run, or manual verification note.	No mobile, interaction, or state check is shown.
Migration safe	Schema diff, data-shape review, rollback or fixture validation.	Migration file exists but downstream callers are not inspected.
Tests pass	Exact command, output, and coverage of the changed behavior.	The command is claimed but not shown.
Security-sensitive change safe	Permission, auth, secret, data exposure, and edge-case review.	General security assurance is implied without evidence.

NO HIDDEN SECURITY CLAIM

Flag security-sensitive changes when relevant, but do not represent this workflow as a complete security audit or compliance review.

CHAPTER 09

Prompt and agent instructions

The skill can be invoked as an agent instruction. The buyer should provide the original request, the claimed completion, the diff or changed files, and any validation evidence.

Copy-ready invocation

```
Use AI Code Change Auditor on this agent-generated patch.
Original request:
[paste request]

Claimed completion:
[paste agent summary]

Changed files or diff:
[paste diff or file list]

Validation evidence:
[paste tests, logs, screenshots, CI, or browser checks]

Tell me whether I can trust the result, what still needs fixing, and
produce a fix verification receipt.
```

INSTRUCTION BOUNDARY

The agent should audit the change. It should not deploy, publish, merge, delete files, collect secrets, or make hidden network calls unless separately authorized.

CHAPTER 10

Worked examples

Examples are where the field guide becomes operational. Each scenario shows how the same doctrine changes shape for bugs, UI patches, API routes, migrations, security-sensitive changes, scope drift, missing evidence, and release gates.

WORKED EXAMPLE 01

Agent Claimed Tests Pass Review Input

Sources: examples/agent-claimed-tests-pass-review-input.md | examples/agent-claimed-tests-pass-review-output.md

Scenario input

Agent Claimed Tests Pass Review Input

Scenario

The buyer wants to use AI Code Change Auditor for: agent claimed tests pass review.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.

- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Agent Claimed Tests Pass Review Output

Intake Summary

The requested scenario is `agent claimed tests pass review` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory

- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: agent claimed tests pass review
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 02

Api Route Regression Check Input

Sources: examples/api-route-regression-check-input.md | examples/api-route-regression-check-output.md

Scenario input

Api Route Regression Check Input

Scenario

The buyer wants to use AI Code Change Auditor for: API route regression check.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Api Route Regression Check Output

Intake Summary

The requested scenario is `API route regression check` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map

- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: API route regression check
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 03

Checkout Tax Bug Audit Input

Sources: examples/checkout-tax-bug-audit-input.md | examples/checkout-tax-bug-audit-output.md

Scenario input

Checkout Tax Bug Audit Input

Scenario

The buyer wants to use AI Code Change Auditor for: checkout tax bug audit.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Checkout Tax Bug Audit Output

Intake Summary

The requested scenario is `checkout tax bug audit` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: checkout tax bug audit
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 04

Database Migration Diff Review Input

Sources: examples/database-migration-diff-review-input.md | examples/database-migration-diff-review-output.md

Scenario input

Database Migration Diff Review Input

Scenario

The buyer wants to use AI Code Change Auditor for: database migration diff review.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Database Migration Diff Review Output

Intake Summary

The requested scenario is `database migration diff review` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: database migration diff review
- Evidence reviewed: buyer-provided context and stated constraints

- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 05

Missing Validation Triage Input

Sources: examples/missing-validation-triage-input.md | examples/missing-validation-triage-output.md

Scenario input

Missing Validation Triage Input

Scenario

The buyer wants to use AI Code Change Auditor for: missing validation triage.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:

- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Missing Validation Triage Output

Intake Summary

The requested scenario is `missing validation triage` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: missing validation triage

- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 06

Pull Request Acceptance Gate Input

Sources: examples/pull-request-acceptance-gate-input.md | examples/pull-request-acceptance-gate-output.md

Scenario input

Pull Request Acceptance Gate Input

Scenario

The buyer wants to use AI Code Change Auditor for: pull request acceptance gate.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:

- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Pull Request Acceptance Gate Output

Intake Summary

The requested scenario is `pull request acceptance gate` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor

- Scenario: pull request acceptance gate
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 07

React Ui Patch Review Input

Sources: examples/react-ui-patch-review-input.md | examples/react-ui-patch-review-output.md

Scenario input

React Ui Patch Review Input

Scenario

The buyer wants to use AI Code Change Auditor for: React UI patch review.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:

- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

React Ui Patch Review Output

Intake Summary

The requested scenario is `React UI patch review` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor

- Scenario: React UI patch review
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 08

Release-Blocking Fix Receipt Input

Sources: examples/release-blocking-fix-receipt-input.md | examples/release-blocking-fix-receipt-output.md

Scenario input

Release-Blocking Fix Receipt Input

Scenario

The buyer wants to use AI Code Change Auditor for: release-blocking fix receipt.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:

- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Release-Blocking Fix Receipt Output

Intake Summary

The requested scenario is `release-blocking fix receipt` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: release-blocking fix receipt
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 09

Scope Overreach Review Input

Sources: examples/scope-overreach-review-input.md | examples/scope-overreach-review-output.md

Scenario input

Scope Overreach Review Input

Scenario

The buyer wants to use AI Code Change Auditor for: scope overreach review.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.
- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Scope Overreach Review Output

Intake Summary

The requested scenario is `scope overreach review` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan
- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.

- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: scope overreach review
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

WORKED EXAMPLE 10

Security-Sensitive Auth Change Audit Input

Sources: examples/security-sensitive-auth-change-audit-input.md | examples/security-sensitive-auth-change-audit-output.md

Scenario input

Security-Sensitive Auth Change Audit Input

Scenario

The buyer wants to use AI Code Change Auditor for: security-sensitive auth change audit.

Buyer Context

- Target buyer: Claude Code users, Cursor users, Codex users, OpenCode users, indie hackers, agencies, AI-assisted developers, and technical operators.
- Buyer pain: AI coding agents often claim a change is complete when the diff still has scope drift, missing validation, weak tests, hidden regressions, or unsupported done language.

- Goal: Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Available Inputs

- Business or project context:
- Current workflow:
- Known constraints:
- Available evidence:
- Missing data:
- Human reviewer:

Requested Output

Produce the relevant premium workflow artifacts, quality gates, validation notes, safety boundaries, and receipt.

Audit output excerpt

Security-Sensitive Auth Change Audit Output

Intake Summary

The requested scenario is `security-sensitive auth change audit` for AI Code Change Auditor. The output must stay human-supervised and specific to the buyer context.

Recommended Artifacts

- Audit verdict
- Requested change map
- Claimed completion review
- Changed file inventory
- Scope compliance table
- False-completion check
- Regression risk scan

- Test coverage review

Quality Gate Notes

- Mark assumptions and missing evidence.
- Keep human review required before high-impact use.
- Do not claim guaranteed results.
- Include a receipt with status and limitations.

Example Receipt

- Skill: AI Code Change Auditor
- Scenario: security-sensitive auth change audit
- Evidence reviewed: buyer-provided context and stated constraints
- Missing inputs: to be confirmed by user
- Human review required: yes
- Status: READY_FOR_HUMAN_REVIEW

CHAPTER 11

Templates and checklists

Templates turn the audit from an opinion into a repeatable review system. Use them as full-width working records, not as cramped side-by-side cards.

TEMPLATE 01

Buyer Context Template

Source: templates/buyer-context-template.md

Buyer Context Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:

- Blockers:

TEMPLATE 02

Failure Mode Register Template

Source: templates/failure-mode-register-template.md

Failure Mode Register Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item	Detail	Source/Evidence	Risk	Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 03

Handoff Template

Source: templates/handoff-template.md

Handoff Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:

- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 04

Human Review Checklist

Source: templates/human-review-checklist.md

Human Review Checklist

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:

- Blockers:

TEMPLATE 05

Implementation Notes Template

Source: templates/implementation-notes-template.md

Implementation Notes Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item	Detail	Source/Evidence	Risk	Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 06

Input Inventory Template

Source: templates/input-inventory-template.md

Input Inventory Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:

- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 07

Marketplace Delivery Receipt Template

Source: templates/marketplace-delivery-receipt-template.md

Marketplace Delivery Receipt Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item	Detail	Source/Evidence	Risk	Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.

- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 08

Output Contract Template

Source: templates/output-contract-template.md

Output Contract Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item

Detail

Source/Evidence

Risk

Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 09

AI Code Change Audit

Source: templates/output-template.md

AI Code Change Audit

Audit verdict: `PASS | NEEDS_FIX | BLOCKED | PRODUCT_REVIEW_REQUIRED`

Requested change:

[Restate the original request in one or two sentences.]

Claimed completion:

[Separate the agent or developer claim from verified facts.]

Files reviewed:

- Reviewed:
- Not reviewed:
- Important files expected but missing:

Scope compliance:

- In scope:
- Borderline:
- Out of scope / overreach:

False-completion check:

[State whether the claimed completion is supported by inspected evidence. Name any missing proof.]

Failure-mode scan:

- High:
- Medium:
- Low:

Validation review:

- Tests or checks run:
- Output reviewed:
- Tests or checks missing:
- Suggested validation commands:

Recommended next actions:

1.

2.

3.

Fix Verification Receipt

- Skill: AI Code Change Auditor
- Request reviewed:
- Evidence reviewed:
- Files reviewed:
- Tests reviewed:
- Tests missing:
- Overreach found:
- Blocking issues:
- Status: PASS | NEEDS_FIX | BLOCKED | PRODUCT_REVIEW_REQUIRED
- Limitation:

TEMPLATE 10

Platform Fit Template

Source: templates/platform-fit-template.md

Platform Fit Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:

- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 11

Premium Intake Template

Source: templates/premium-intake-template.md

Premium Intake Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 12

Quality Gate Template

Source: templates/quality-gate-template.md

Quality Gate Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item	Detail	Source/Evidence

Risk

Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 13

Fix Verification Receipt

Source: templates/receipt-template.md

Fix Verification Receipt

- Skill: AI Code Change Auditor
- Date:
- Request reviewed:
- Claimed completion:
- Evidence reviewed:
- Files reviewed:
- Files not reviewed:
- Tests reviewed:
- Tests missing:
- Scope compliance:

- Overreach found:
- False-completion risk:
- Regression risk:
- Security-sensitive change:
- Blocking issues:
- Recommended next command:
- Status: PASS | NEEDS_FIX | BLOCKED | PRODUCT_REVIEW_REQUIRED
- Limitation:

TEMPLATE 14

Risk Register Template

Source: templates/risk-register-template.md

Risk Register Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:

- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 15

Scope Boundary Template

Source: templates/scope-boundary-template.md

Scope Boundary Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:

- Blockers:

TEMPLATE 16

Security Boundary Template

Source: templates/security-boundary-template.md

Security Boundary Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item	Detail	Source/Evidence	Risk	Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 17

Validation Plan Template

Source: templates/validation-plan-template.md

Validation Plan Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:

- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:
- Blockers:

TEMPLATE 18

Workflow Plan Template

Source: templates/workflow-plan-template.md

Workflow Plan Template

SKU

AI Code Change Auditor

Purpose

Use this template to produce one part of the premium workflow for this buyer promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Inputs

- Buyer goal:
- Available evidence:
- Missing inputs:
- Assumptions:
- Human reviewer:

Working Table

Item
Detail
Source/Evidence
Risk
Review Status

Quality Checks

- Specific to buyer context.
- Assumptions separated from evidence.
- Safety boundaries stated.
- Human review requirement included.
- Receipt or handoff note updated.

Output Notes

- Ready items:
- Items needing validation:

- Blockers:

CHAPTER 12

Security boundaries and human review

The skill is narrow by design. It helps buyers review code-change claims. It does not authorize credential collection, secret exposure, destructive operations, hidden network calls, or unapproved deployment.

Security Notes

Product

AI Code Change Auditor

Safety Boundary

This skill pack is human-supervised. It supports planning, scaffolding, review, validation, and receipts. It does not authorize hidden network calls, credential collection, secret exposure, unapproved publishing, unapproved deployment, destructive operations, or high-impact decisions without human review.

Prohibited Claims

- Guaranteed revenue, rankings, traffic, compliance, security, correctness, or performance.
- Fully autonomous operation.
- Replacement of employees, developers, staff, or expert review.
- No-human-review operation.
- Instant or magic outcomes.
- Unsupported production-readiness claims.

Required User Review

The buyer must verify inputs, permissions, source data, claims, tool access, platform rules, privacy requirements, and final outputs before use.

HUMAN REVIEW REQUIREMENT

The buyer must verify inputs, permissions, source data, claims, tool access, privacy requirements, and final outputs before use.

CHAPTER 13

Advanced usage and release gates

Once the buyer understands the audit pattern, the skill becomes a release gate. The same receipt can be used before merge, before deployment, during PR triage, after an agent handoff, or when a production bug fix needs independent verification.

Pull request gate

Require an audit receipt before accepting an AI-generated PR.

Release-blocking fix

Use the failure scan to determine whether a patch can ship or needs more validation.

Agent supervision

Prevent coding agents from declaring work complete based only on their own final summary.

Regression triage

Use changed-file mapping and test coverage review to identify risk before a fix spreads.

Product acceptance

Route technically plausible but ambiguous changes into PRODUCT_REVIEW_REQUIRED instead of forcing a false PASS.

CHAPTER 14

FAQ and final action plan

Frequently asked questions

Can this replace code review?

No. It structures post-agent verification and evidence review. It does not replace expert review.

Can the audit pass without a diff?

No. Without changed files or equivalent evidence, the audit is evidence-limited and cannot be PASS.

What if tests pass?

Passing tests are only one evidence type. The reviewer still checks whether the tests cover the requested behavior.

What if the change is technically plausible?

Use PRODUCT_REVIEW_REQUIRED when the remaining decision is product, UX, business-rule, or human acceptance.

What is the final artifact?

A fix verification receipt that states evidence, limitations, blockers, and status.

Final action plan

01 **Choose the closest scenario**

Start with the example that matches the patch: bug fix, UI patch, API route, migration, security-sensitive change, or release gate.

02 **Fill the evidence inventory**

List what was provided and what is missing before judging the implementation.

03 **Map changed files to the request**

Do not let unexplained files disappear inside the review.

04 **Run validation review**

Name what was run and what still needs to be run.

05 **Issue a receipt**

Use the final status honestly. A blocked receipt is better than a fake pass.

APPENDIX

Appendix A: actual skill structure

Folder or file	Role in the field guide
SKILL.md	Agent-operable workflow, output contract, quality gates, and receipt rules.
README.md	Buyer outcome, use cases, files, usage sequence, and safety limits.
examples/	Worked scenarios that show how the workflow behaves in real review situations.
templates/	Reusable records for intake, scope, validation, risk, handoff, output, and receipt work.
references/	Rubrics and policies that protect quality, positioning, human review, and no-guarantee boundaries.
security-notes.md	Human-supervised security boundaries and prohibited claims.

APPENDIX

Appendix B: reference rubrics

Buyer Promise Clarity Guide

Source: references/buyer-promise-clarity-guide.md

Buyer Promise Clarity Guide

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Example Quality Rubric

Source: references/example-quality-rubric.md

Example Quality Rubric

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Failure Mode Guide

Source: references/failure-mode-guide.md

Failure Mode Guide

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Human-Supervised Boundary Policy

Source: references/human-supervised-boundary-policy.md

Human-Supervised Boundary Policy

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Marketplace Positioning Guide

Source: references/marketplace-positioning-guide.md

Marketplace Positioning Guide

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

No Guarantee Claims Policy

Source: references/no-guarantee-claims-policy.md

No Guarantee Claims Policy

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Premium Readiness Rubric

Source: references/premium-readiness-rubric.md

Premium Readiness Rubric

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Quality Gate Rubric

Source: references/quality-gate-rubric.md

Quality Gate Rubric

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Receipt Quality Rubric

Source: references/receipt-quality-rubric.md

Receipt Quality Rubric

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

Security And Privacy Policy

Source: references/security-and-privacy-policy.md

Security And Privacy Policy

Applies To

AI Code Change Auditor

Review Standard

The output should be buyer-specific, evidence-aware, human-supervised, and bounded by the product promise:

Audit AI-generated code changes, diffs, pull requests, and claimed completions with scope checks, evidence review, validation analysis, risk findings, and a fix verification receipt.

Pass Criteria

- The buyer pain is addressed directly.
- The output uses clear sections and decision points.
- Claims are supported or labeled as assumptions.
- Security and privacy boundaries are explicit.
- Human review is required for high-impact actions.
- The final receipt states status and limitations.

Fail Criteria

- Generic advice that could apply to any buyer.
- Unsupported guarantees or production-readiness claims.
- Hidden automation, unapproved tool use, or implied credential access.
- Missing quality gates, validation plan, or receipt.

Approved Positioning

Use strong buyer language where accurate. Do not claim guaranteed outcomes, full autonomy, staff replacement, or no-review operation.

APPENDIX

Appendix C: source map

The V2 guide mapped the following source files from the skill pack.

- CHECKSUMS.txt
- MARKETPLACE_LISTING_COPY.md
- MARKETPLACE_PROMISE_REVIEW.md
- OVERCLAIM_AUDIT.md
- PACKAGE_MANIFEST.md
- README.md
- RELEASE_NOTES.md
- SKILL.md
- SKILL_BUILD_RECEIPT.md
- SKU_PATCH_RECEIPT.md
- START_HERE.md
- examples/agent-claimed-tests-pass-review-input.md
- examples/agent-claimed-tests-pass-review-output.md
- examples/api-route-regression-check-input.md
- examples/api-route-regression-check-output.md
- examples/checkout-tax-bug-audit-input.md
- examples/checkout-tax-bug-audit-output.md
- examples/database-migration-diff-review-input.md
- examples/database-migration-diff-review-output.md
- examples/input.md
- examples/missing-validation-triage-input.md
- examples/missing-validation-triage-output.md
- examples/output.md
- examples/pull-request-acceptance-gate-input.md
- examples/pull-request-acceptance-gate-output.md
- examples/react-ui-patch-review-input.md

- examples/react-ui-patch-review-output.md
- examples/release-blocking-fix-receipt-input.md
- examples/release-blocking-fix-receipt-output.md
- examples/scope-overreach-review-input.md
- examples/scope-overreach-review-output.md
- examples/security-sensitive-auth-change-audit-input.md
- examples/security-sensitive-auth-change-audit-output.md
- references/buyer-promise-clarity-guide.md
- references/example-quality-rubric.md
- references/failure-mode-guide.md
- references/human-supervised-boundary-policy.md
- references/marketplace-positioning-guide.md
- references/no-guarantee-claims-policy.md
- references/premium-readiness-rubric.md
- references/quality-gate-rubric.md
- references/receipt-quality-rubric.md
- references/security-and-privacy-policy.md
- security-notes.md
- submission-notes.md
- templates/buyer-context-template.md
- templates/failure-mode-register-template.md
- templates/handoff-template.md
- templates/human-review-checklist.md
- templates/implementation-notes-template.md
- templates/input-inventory-template.md
- templates/marketplace-delivery-receipt-template.md
- templates/output-contract-template.md
- templates/output-template.md
- templates/platform-fit-template.md
- templates/premium-intake-template.md

- templates/quality-gate-template.md
- templates/receipt-template.md
- templates/risk-register-template.md
- templates/scope-boundary-template.md
- templates/security-boundary-template.md
- templates/validation-plan-template.md
- templates/workflow-plan-template.md