

AI OUTPUT REVIEW CHECKLIST

AI Change Acceptance and Rejection Checklist

Review AI output against its brief, repository evidence, ownership boundaries, regressions, and recovery readiness.

Review AI output against its brief, repository evidence, ownership boundaries, regressions, and recovery readiness. Use one identified candidate, system, operation, or decision per copy so the record remains reviewable.

USE THIS RESOURCE WHEN

- Use it before accepting any AI-authored change into a game project.
- Ownership, evidence, or decision authority must be explicit.
- Another person must be able to review or reproduce the result.



Free Academy resource

English • A4 edition • Fillable PDF • v1.0

[Resource page](#)

How to use it

Review AI output against its brief, repository evidence, ownership boundaries, regressions, and recovery readiness. Use one identified candidate, system, operation, or decision per copy so the record remains reviewable.

HOW TO USE IT

- 1 Identify the exact subject and baseline before filling the working sections.
- 2 Separate verified facts from assumptions and unresolved questions.
- 3 Link evidence to the candidate, build, version, or decision it actually supports.
- 4 Record an accountable decision, owner, and next trigger before closing the document.

EXPLORE THE ACADEMY

[Martinez AI Studios](#)

[Academy](#)

[170-lesson course](#)

[Resource page](#)

USE AND SHARING

Free for personal and educational use. Share the original Martinez AI Studios URL. No resale, commercial redistribution, or re-upload.

Candidate and baseline

Change / branch / commit / reviewer

Be specific. Name the source, evidence, owner, and decision where applicable.

Approved brief and protected surfaces

Be specific. Name the source, evidence, owner, and decision where applicable.

Baseline and required evidence

Be specific. Name the source, evidence, owner, and decision where applicable.

Candidate evaluation

For every item, mark the result and record evidence. An empty box is not a pass.

1. Every changed file and hunk maps to a brief clause.

Evidence / notes

2. Claims are labeled VERIFIED, ASSUMED, or UNSUPPORTED.

Evidence / notes

3. The correct owner performs each mutation.

Evidence / notes

4. Protected systems and invariants remain intact.

Evidence / notes

5. Unrelated edits, generated files, secrets, and hidden dependencies are identified.

Evidence / notes

Candidate evaluation — continued

6. Focused, boundary, regression, and supported-route checks pass.

Evidence / notes

7. Build/commit identity and observable evidence are recorded.

Evidence / notes

8. Ambiguity is resolved; plausible code is not treated as proof.

Evidence / notes

9. Restoration instructions and baseline are verified.

Evidence / notes

Accountable decision

Decision — accept / revise / reject

Be specific. Name the source, evidence, owner, and decision where applicable.

Open risks / unsupported claims

Be specific. Name the source, evidence, owner, and decision where applicable.

Decision owner / follow-up / date

Be specific. Name the source, evidence, owner, and decision where applicable.