

RECOVERABLE CHANGE CHECKLIST

Git Checkpoint and Restoration Checklist

Create a recoverable baseline, preserve failure evidence, and verify restoration without prescribing destructive commands.

Create a recoverable baseline, preserve failure evidence, and verify restoration without prescribing destructive commands. Use one identified candidate, system, operation, or decision per copy so the record remains reviewable.

USE THIS RESOURCE WHEN

- Use it before risky edits, cleanup, migration, or failure recovery.
- 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

Create a recoverable baseline, preserve failure evidence, and verify restoration without prescribing destructive commands. 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

Repository / branch / target paths

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

Current status / build / runtime baseline

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

Work and evidence that must be retained

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. Branch, commit, status, and explicit target paths recorded.

Evidence / notes

2. Tracked, staged, committed, untracked, and generated changes classified.

Evidence / notes

3. Reproduction steps and failure evidence saved separately.

Evidence / notes

4. Repository-approved handling confirmed for every class.

Evidence / notes

5. Intended work preserved under local policy.

Evidence / notes

Candidate evaluation — continued

6. Restoration operation previewed and narrowly scoped.

Evidence / notes

7. Untracked work protected before cleanup.

Evidence / notes

8. Restoration follows dependency order and approved procedures.

Evidence / notes

9. Final status and diff inspected.

Evidence / notes

10. Project relaunched or rebuilt from restored state.

Evidence / notes

11. Baseline checks rerun and preserved behavior confirmed.

Evidence / notes

12. Final commit, evidence, and unresolved files recorded.

Evidence / notes

Accountable decision

Restoration result

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

Remaining files / risks / unknowns

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

Reviewer / decision / date

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