

Contractor Code Review Checklist

By Kris Drouet, Engineering Executive, in partnership with KORE1. Twenty-one checks in five groups, each with the proof to attach and who runs it.

PULL REQUEST	REPOSITORY	CONTRACT ENGINEER (AUTHOR)
REVIEWER	EMPLOYEE OWNER	DATE REVIEWED

HOW TO READ A LINE	 Gone must not be in the pull request	 Attached proof that must be on it	 Employee owner runs or signs it
--------------------	---	--	--

1. Secrets and credentials Nothing that opens a door is in the diff, the history or the fixtures. @@ 3 GONE, 2 ATTACHED @@

☐	 S1 GONE	No keys, tokens, passwords or private keys in the diff That includes test fixtures, sample config and a pasted curl command. If it opens something, it does not belong in a commit.	Secret scan passed on the pull request	☐ Any reviewer	INITIALS AND DATE
☐	 S2 GONE	No secrets anywhere in the branch history A secret deleted in the last commit is still in the one before it. Scan every commit on the branch, not only the final diff.	Scan covers all commits on the branch	☐ Any reviewer	INITIALS AND DATE
☐	 S3 GONE	No environment files or local config committed A .env file, a keystore or a downloaded credentials file is a leak waiting for a push. Sample files hold placeholders only.	File list checked against the ignore rules	☐ Any reviewer	INITIALS AND DATE
☐	 S4 ATTACHED	Any new secret is requested through the secrets manager The description names the secret, where it lives and which employee rotates it. The contractor never holds the production value.	Request or ticket linked in the description	☒ Employee owner	INITIALS AND DATE
☐	 S5 ATTACHED	A push protection bypass has a reason an employee has read By default anyone with write access can bypass the block by giving a reason, so someone other than the author reads the reason.	Bypass reason seen in the alert log	☒ Employee owner	INITIALS AND DATE

2. PII handling No real customer data in the code, the logs or the tests, and a clear path for the data that is meant to be there. @@ 3 GONE, 1 ATTACHED @@

☐	 P1 GONE	No real customer data in tests, fixtures, screenshots or logs Use synthetic records. A production export pasted into a test file puts real data in version control for good.	Fixtures and screenshots read, synthetic data only	☐ Any reviewer	INITIALS AND DATE
☐	 P2 ATTACHED	New fields that hold personal data are listed in the description Name the field, where it is stored, who can read it and how long it is kept.	Field list in the pull request description	☒ Employee owner	INITIALS AND DATE

☐	 P3 GONE	No personal data in log lines, error messages or analytics events Names, emails, account numbers and government IDs stay out. Log an internal ID instead.	Every new log line read in the diff	☐ Any reviewer	INITIALS AND DATE
☐	 P4 GONE	No new path for personal data to leave the system No new third-party call, SDK, export job or webhook that carries customer data unless the owner approved it.	New outbound calls listed and approved	☒ Employee owner	INITIALS AND DATE

3. Licensing of pasted and AI-generated code

Every block of code has a known origin and a license you can live with. @@ 2 GONE, 2 ATTACHED @@

☐	 L1 GONE	No pasted code of unknown origin A snippet from a forum, a gist or another repository names its source and license in the description, or it comes out.	Source and license written beside the snippet	☐ Any reviewer	INITIALS AND DATE
☐	 L2 ATTACHED	New dependencies pass the license check Each new package is on the approved list. Copyleft licenses such as GPL and AGPL go to counsel before they go into a closed-source product.	License scan result attached	☐ Any reviewer	INITIALS AND DATE
☐	 L3 ATTACHED	AI-generated code is disclosed, and the author owns every line The description says which parts a tool wrote, the tool is on the approved list, and the author can explain each line.	Disclosure line in the description	☐ Any reviewer	INITIALS AND DATE
☐	 L4 GONE	No license headers or notices removed or altered Vendored code keeps its license file. Copyright lines stay as the source had them.	Headers compared with the source	☐ Any reviewer	INITIALS AND DATE

4. Test evidence

The proof is a link anyone can follow, not a sentence anyone can type. @@ 1 GONE, 3 ATTACHED @@

☐	 T1 ATTACHED	Tests cover the changed behavior, including the failure path A test for the happy path proves only that the code works when nothing goes wrong.	New or updated tests in the diff	☐ Any reviewer	INITIALS AND DATE
☐	 T2 ATTACHED	CI is green on the final commit Not on an earlier one. A push after the green check reopens the question.	Link to the run for the head commit	☐ Any reviewer	INITIALS AND DATE
☐	 T3 GONE	No tests that call real production services or use real data Tests run against fakes, sandboxes or synthetic records.	Test setup read in the diff	☐ Any reviewer	INITIALS AND DATE
☐	 T4 ATTACHED	The author says how they tested by hand Environment, steps and result, in the description. Development or staging, never production.	Manual test note in the description	☐ Any reviewer	INITIALS AND DATE

5. Owner sign-off

An employee who owns the code approves it, after the last commit, and the record stays. @@ 1 GONE, 3 ATTACHED @@

☐	 01 ATTACHED	An employee who owns each touched path is named The owner is an employee with write access to the repository, named in CODEOWNERS or in the description.	CODEOWNERS entry or a named owner	Ⓢ Employee owner	INITIALS AND DATE
☐	 02 ATTACHED	The owner approves after the last commit Dismiss stale approvals, so a change pushed after the approval gets looked at again.	Approval dated after the head commit	Ⓢ Employee owner	INITIALS AND DATE
☐	 03 GONE	No self-approval, and no contractor approval counted as owner sign-off A contractor's review is welcome and useful. It just is not the signature.	Required reviews set to code owners	Ⓢ Employee owner	INITIALS AND DATE
☐	 04 ATTACHED	The merge record is kept where an auditor can find it The pull request, approvals, scan results and test run stay linked for as long as your policy says.	Retention written in the team's working agreement	Ⓢ Employee owner	INITIALS AND DATE

READY TO MERGE

Author

I ran these checks and attached the proof.
Signature and date.

Reviewer

I reviewed this change against the checklist.
Signature and date.

Employee owner

I own this code and I approve the merge.
Signature and date.

Read this too. 10 checks look for something that must be gone and 11 look for proof that must be attached. A checklist template, not legal advice or a security audit. Change the items to fit your stack and your policy, but keep a proof and an owner on every one. The reasoning behind each group is at kore1.com/contractor-code-review-checklist/. For the trust ramp and the branch settings, read kore1.com/reviewing-contractor-code/. For who owns the code a contractor writes, read kore1.com/contractor-ip-ownership/.