AI QUALITY FRAMEWORK

Software you can trust. Process you can verify.

How Trinity Software Center safeguards quality, security and compliance in AI-assisted software development

Prepared for technical leads and CTOs evaluating AI-assisted development partners | Version 1.0 — June 2026

Why We Built This Framework

Trinity Software Center has used AI tools in our development workflow since 2024. The productivity gains are real — our Scrum teams deliver more within a sprint without adding headcount. The risks are real too: AI-generated code can look correct while hiding subtle bugs, security gaps, or architectural shortcuts that only surface once software is in production.

This framework sets out, in plain terms, exactly how we manage that risk — the tools we use, the review steps we follow, and the standards we hold ourselves to. It is written for the people who have to answer for the software once it ships.

At a Glance — Our Five AI Quality Commitments

Commitment	What it means in practice
1. 100% human review of AI-generated code	No AI output enters a client codebase without sign-off from a second human engineer.
2. Independent tools for writing and reviewing	A different AI tool reviews the code than the one that wrote it, to avoid a model approving its own mistakes.
3. Automated tests written in parallel	Unit tests and test scripts are built alongside AI-generated code, not bolted on afterwards.
4. Full transparency on AI tool use	You are told which AI tools were used on your project and what they contributed.
5. EU AI Act aligned classification	AI components we build are classified and documented against the Act's risk categories.

How AI Fits Into Our Development Process

Where AI helps: drafting boilerplate code, generating first-pass implementations, suggesting refactors, accelerating documentation, and supporting the analysis phase of legacy-software rescue projects.

Where humans stay firmly in charge: architecture decisions, security-sensitive logic, requirements interpretation, final code approval, and all client communication.

AI use is built into our Scrum process, not bolted onto it. Every sprint team includes a dedicated QA engineer. The Definition of Done for any story includes passing review and tests, regardless

dianavanderstelt@trinitysoftwarecenter.com | stephenofori@trinitysoftwarecenter.com | +316 15038099 | +233 557916646

of whether AI was involved in writing it. Sprint retrospectives are used to refine how each team uses AI tools as the technology — and our experience with it — evolves.

Commitment 1 — 100% Human Review

We reinforced our code review policy to 100% specifically in response to AI-assisted development: every commit — whether typed by a developer or drafted with AI assistance — is reviewed by a second engineer before it is merged. Senior engineers review the work of more junior colleagues, which is itself part of how we train the next generation of African software engineers. Review comments are logged and resolved, not silently waved through.

Commitment 2 — Avoiding "Self-Confirmation"

A model that wrote a piece of code is not a reliable judge of whether that code is correct — it tends to confirm its own assumptions. Where we use AI assistance in the review step itself, we deliberately use a different tool than the one that generated the code, alongside the human reviewer who has the final say. We have applied this in practice: when a substantial change was made to one of our care-sector platforms, the change was checked by a human reviewer and by a second AI tool chosen specifically because it had not written the original code, so it could flag anything the author — human or AI — might have missed.

Commitment 3 — Tests Written in Parallel

Automated unit tests and test scripts are written alongside AI-assisted development, in the same sprint, not deferred to a later "testing phase." This catches regressions early and gives us a concrete, repeatable way to verify that AI-generated code does what it claims — beyond a human reviewer’s read-through.

Commitment 4 — Transparency

On request, we document which AI tools contributed to your project and what they were used for. This is part of standard project handover documentation, alongside the architecture notes and security considerations we already provide on every project. We do not believe undisclosed AI use belongs in professional software delivery.

Commitment 5 — EU AI Act Aligned Classification

We classify the AI components we build against the EU AI Act’s risk-based categories, and document that classification as a standard part of project delivery:

Risk tier	Example	What we do
Prohibited	Social scoring, real-time biometric ID in public spaces	We do not build these systems.
High-risk	AI used in employment, credit, or healthcare decisions affecting individuals	Documented risk assessment, human oversight, quality datasets, logging for post-market monitoring; support for Fundamental Rights Impact Assessments where required.
Limited risk	Chatbots, AI-generated content shown to users	Clear disclosure that users are interacting with, or viewing, AI-generated output.
Minimal risk	AI-assisted form completion, internal	Standard QA process; light-touch

	automation	documentation.

From August 2026, high-risk AI system obligations under the EU AI Act take full effect. We apply this classification process to AI components in client projects today, ahead of that date, not after it.

Responsible AI Design Principles We Apply

Beyond the EU AI Act's legal categories, every AI-assisted feature we build is checked against six design principles:

- **Human-in-the-loop —** AI supports decisions; people remain accountable for them.
- **Purpose-driven —** AI is applied to a clearly defined problem, not used for its own sake.
- **Transparency —** users understand what a system does, and does not do.
- **Fairness —** we watch for and actively mitigate bias in AI-supported outcomes.
- **Privacy & security —** data protection is built in by design, not retrofitted.
- **Robustness —** systems are tested, monitored, and improved over time.

AI Governance at Trinity

We maintain an internal AI governance structure responsible for setting coding standards for AI-assisted development, evaluating new AI models and tools before they are approved for project use, running structured knowledge-sharing across our development teams, and reviewing our own AI policies as the technology and the regulatory landscape evolve. Current focus areas include the August 2026 EU AI Act high-risk obligations and the data protection implications of any move toward self-hosted, open-source language models.

Our broader information security management practices — covering GDPR, OWASP, NIS2-conscious design, and our ongoing work toward ISO/IEC 27001 alignment — sit alongside this AI governance structure and are described in our companion Security Questions Checklist.

What This Means For Your Project

Ask us, before you sign anything: which AI tools would be used on this project and for what; how review and testing would work, sprint by sprint; and how we would classify any AI feature in your system under the EU AI Act, and what that means for your own compliance obligations. We would rather answer these questions at the start of a relationship than have you discover the answers the hard way later.

Want to talk this through?

Book a free discovery call — we will walk you through our AI quality process in plain language, on your project specifically.

dianavanderstelt@trinitysoftwarecenter.com | +316 15038099 | +233 557916646