AI · TEMPLATE

AI Governance for Regulated Products: A *Practical* Framework

KAANSYSTEMS.COM/LIBRARY/AI-GOVERNANCE-FOR-REGULATED-PRODUCTS · SEPTEMBER 16, 2026

ABOUT THIS TEMPLATE

AI governance for a regulated product isn't an ethics board — it's the controls that keep your AI features inside the HIPAA or SOC 2 boundary you already built. The framework, the evidence, and the buyer test.

THE TEMPLATE

An AI feature governance checklist. Run it before every AI feature that could touch regulated data, and re-run it per feature and per vendor — the BAA you signed for the summarization feature doesn't cover the new vendor you're using for embeddings. Score each item yes / no / not-applicable; any `no` on a required item blocks the ship until it's resolved.

Data flow (map it first)

- [ ] Traced every field that enters the prompt; confirmed it's the minimum necessary
- [ ] Confirmed no PHI/PII in URL parameters, feature flags, or analytics events for this feature
- [ ] Traced the response path; confirmed output isn't written anywhere out of scope
- [ ] Data-flow map recorded and attached to the feature

Vendor + agreements

- [ ] Model vendor scored on the vendor-risk rubric before access was granted
- [ ] BAA signed with the vendor (or PHI is de-identified before it leaves our systems)
- [ ] DPA in place where personal data is involved
- [ ] Vendor added to the public sub-processor list
- [ ] Data-use terms reviewed: confirmed no training on our inputs, opt-out verified in writing

- [ ] Vendor's retention of our data documented and acceptable

Logging + retention

- [ ] Prompts + responses are NOT logged as content to general application logs
- [ ] Traces are content-free (ids, versions, hashes, flags, decision)
- [ ] Anything retained lives in a scoped, encrypted, in-boundary store with production-grade access controls
- [ ] Retention period set deliberately, not left at the vendor default

Output safety + oversight

- [ ] Consequential outputs have a human in the decision, not just a notification
- [ ] Low-confidence / flagged outputs are surfaced and logged
- [ ] Users can tell the content was AI-generated
- [ ] A documented plan exists for when the model is confidently wrong

Versioning + change management

- [ ] Model version is a pinned dated snapshot, recorded per request (no moving aliases)
- [ ] Prompt templates live in version control and change via reviewed PR
- [ ] A model or prompt change triggers an eval run with a promotion threshold

Evidence + ongoing

- [ ] Every control above maps to a produced artifact (agreement, map, trace, review record, eval history)
- [ ] Human-review coverage is queryable: consequential outputs with no linked review = 0 rows
- [ ] This feature is in scope for the next access review + evidence pipeline
- [ ] A re-review triggers when the vendor changes terms or we change the data flow

The checklist takes about an hour per feature. The alternative — an undisclosed sub-processor, an out-of-scope PHI log, and an unanswerable "prove it" during a customer's diligence call — takes a lot longer and usually costs the deal.

WHAT GOES INTO AN AI GOVERNANCE FRAMEWORK?

Five controls, and each maps to a risk above. This is the whole framework; everything else is elaboration.

01 **Data-flow mapping.** For every AI feature, trace every field that enters the prompt, where it came from, where the response is written, and where any of it is logged. You can't govern a data flow you haven't

drawn. The map is also the artifact that answers half of a buyer's questions before they ask.

02 **Signed BAAs / DPAs with model vendors.** No PHI into a prompt without a Business Associate Agreement; no personal data into one without the DPA your privacy commitments require. This is the gate that stops most teams cold because it's the one usually skipped. "We'll get the BAA later" is how regulated data ends up somewhere it can never be recalled from.

03 **Retention and no-training terms.** Confirm in writing that the vendor does not train on your inputs, set the vendor's retention of your data to something you can defend, and re-verify on a schedule because AI vendors change terms far more often than infrastructure vendors do.

04 **Human-in-the-loop on consequential output.** The more consequential the output, the more the human has to be in the decision, not merely notified of it. Draft, don't decide; surface, don't act. This is your best defense against the confidently-wrong answer you didn't catch in testing, and it's the control buyers probe hardest.

05 **Model and prompt version tracking.** Pin dated model snapshots rather than moving aliases, keep prompt templates in version control behind reviewed pull requests, and record which versions were live per request. This is what makes "what was running on March 14" answerable and folds AI behavior into your existing change-management control.

The framework earns its keep only when each control leaves a trace. Governance you can't produce evidence for is a story, and stories fail Type II audits. Map every control to the artifact that proves it:

RISK AREA	CONTROL	EVIDENCE
Model vendor as sub-processor	Vet and score the vendor; sign a BAA/DPA; list on the public sub-processor page	Executed agreement; sub-processor page entry; vendor risk score on file
PHI/PII in the prompt	Data-flow map; minimum-necessary field review; de-identify where a BAA isn't possible	Data-flow diagram; field-level review sign-off per feature
Training on your inputs	Contractual no-training term; retention set deliberately; quarterly re-verification	Vendor-terms excerpt; "last verified" date; calendar cadence
Prompt / output logging	Content-free traces; scoped encrypted store for anything retained; deliberate retention	Trace schema; retention config; a log sample showing shape, not content
Model / prompt drift	Pinned dated snapshots; prompt templates in git via reviewed PR; eval on every change	Per-request model id; git log of the template; eval history keyed by version
Confidently-wrong output	Human-in-the-loop on consequential output; low-confidence flagging	Human-review records linked to a trace id; a coverage query returning zero unreviewed
Model version change	Treat an upgrade as a change with an approval and an eval run	Merged-PR approval; before/after eval scores

Read the evidence column and notice what it is: the same access, change-management, and logging evidence your SOC 2 or HIPAA program already produces, pointed at a new component. That's the load-bearing idea. You are not standing up a parallel compliance regime for AI — you are extending the one you have.