

Architecture & risk assessment

A practical path to a *safer pilot*.

Fieldnote: a fictional scheduling and document-assistant startup preparing to onboard its first three pilot organizations.

Fictional demonstration, not a client case study

All company details, artifacts, findings, and estimates in this report are illustrative. No real client, live application, source repository, or production system was reviewed. This sample shows the format and reasoning you can expect from an assessment.

Prepared for: Fieldnote / fictional scenario | **Sample edition:** October 17, 2026 / v1.1 | **Decision:** Controlled pilot, after remediation

Executive recommendation

Proceed with a limited pilot after enforcing tenant authorization, restricting AI actions, and defining document-handling controls. Complete a restore exercise and usage controls before expanding customer access.

This is an illustrative recommendation. It is not an assurance that any system is secure, compliant, or production-ready.

05 Prioritized findings Three high-priority and two medium-priority gaps, each tied to a specific next action.	30 Days to address the basics A suggested sequence for authorization, AI boundaries, recovery, and usage controls.	90 Days of direction A roadmap the founder and technical team can use to plan and discuss tradeoffs.
--	--	--

Operated by Kshathra Technologies Inc., Canada Fictional example / No live testing performed

Understand the system.

Bound the review.

The scenario assumes a six-person team building a multi-tenant SaaS application. Users upload internal documents, ask questions, and ask an AI assistant to prepare scheduling changes. Three organizations are ready to join a pilot.

Business objective

Validate whether document-assisted scheduling saves users time, while keeping each organization's information and actions separate.

Assumed evidence

- One architecture sketch and data-flow description.
- Illustrative search and AI-action handlers from two repositories.
- A sample logging configuration, backup schedule, and job configuration.

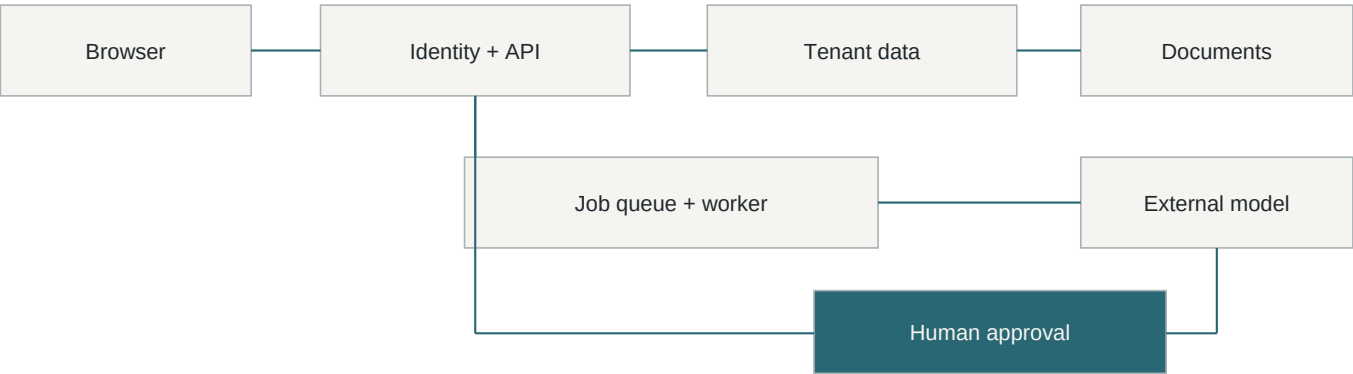
Review boundaries

One application and one deployment environment. Architecture and trust-boundary review, plus sampling of supplied material. Findings below are constructed for this demonstration.

Not performed

Penetration testing, live exploit attempts, exhaustive source review, production access, performance benchmarking, and legal or compliance assessment.

Illustrative architecture



The model is outside the authorization boundary. Document content and model output cannot grant new access or authorize a write. This diagram illustrates a proposed boundary; it does not represent a verified implementation.

Priority method: "High" means a gap could materially affect tenant confidentiality, authorized actions, or sensitive document handling. "Medium" means a gap could materially affect recovery, reliability, or operating costs. Ratings reflect the stated scenario; no exploitability or likelihood was verified.

Protect the boundary between *customers and actions*.

01 / Enforce tenant isolation in retrieval

Illustrative evidence: The scenario's search handler ranks documents by relevance, but its query does not include a mandatory server-side tenant condition. This is a constructed artifact, not observed source code.

Why it matters: A retrieval feature that relies on client-supplied filters could return another organization's records. Adding a model after retrieval does not resolve the underlying access boundary.

Recommendation: Derive tenant membership from the authenticated session, apply authorization before retrieval, and use the same policy for database queries, vector search, file access, and background jobs. Treat client filters as search preferences only.

Acceptance check: A user in Tenant A cannot retrieve Tenant B's documents by changing identifiers, search terms, document links, or job parameters. Add negative checks for direct API requests and worker execution.

Suggested owner: Backend lead | **Sequence / dependency:** Before pilot / define membership policy first

02 / Put explicit permission around AI actions

Illustrative evidence: The scenario's worker can apply schedule changes from model output without a separate approval step or a narrow capability allowlist.

Why it matters: Documents can contain misleading instructions, and model output can be incorrect. Neither should be sufficient authority to change someone's schedule or trigger an external action.

Recommendation: Begin with draft-only output. Validate structured arguments, allow only necessary tools, recheck user permissions at execution, and require a person to approve meaningful writes. Keep a reviewable record of the request, approval, and resulting action.

Acceptance check: Malicious document instructions and malformed model arguments cannot trigger a write. An unauthorized or unapproved action is rejected even when the model requests it.

Suggested owner: AI lead with backend lead | **Sequence / dependency:** Before pilot / build on Finding 01

Ratings apply only to the fictional scenario. Recommendations require implementation and verification.

Know where data goes.

Know how to recover it.

03 / Define document handling and retention

Illustrative evidence: The scenario's diagnostic logging records complete document prompts. No retention or deletion behavior is specified for logs, derived search records, or model-provider requests.

Why it matters: Copies of sensitive material can spread beyond the primary application. Unclear handling makes it difficult to answer customer questions about access, deletion, and approved external processing.

Recommendation: Map document copies and derived data, minimize content sent externally, redact routine logs, agree retention and deletion rules, and verify provider settings against the customer's requirements. Document decisions before using customer information.

Acceptance check: Routine operational logs contain no document bodies. A test deletion follows the agreed policy through storage, search records, and relevant logs. The team can explain provider handling and any stated exceptions.

Suggested owner: Technical lead and product lead | **Sequence / dependency:** Before pilot data upload / confirm requirements

04 / Demonstrate recovery with a restore exercise

Illustrative evidence: A backup schedule appears in the scenario, but there is no supplied restore record, documented recovery target, or verification of document/database consistency.

Why it matters: A scheduled backup is useful only if the team can recover the right information in a useful time. Separate document and database backups may need coordinated recovery.

Recommendation: Agree acceptable data loss and restoration time, restore an isolated backup, check record and document consistency, and write a short runbook with owners and escalation contacts. Restrict restored-data access.

Acceptance check: Record the actual elapsed restoration time, the recovered point in time, and a consistency checklist. Compare results with the agreed targets and create actions for any gap.

Suggested owner: Infrastructure lead | **Sequence / dependency:** Before broader rollout / isolated restore environment

No privacy or regulatory compliance determination is made. No backup or restore was executed for this sample.

Make the pilot measurable.

Keep the next step small.

05 / Bound usage, retries, and duplicate work

Illustrative evidence: The scenario's worker has no per-tenant usage cap, stable job identifier, or documented retry ceiling. Provider cost is visible only in a shared monthly total.

Recommendation: Introduce per-tenant quotas, timeouts, bounded retries, stable idempotency keys, and alerts. Track model requests and cost estimates by tenant without recording document bodies. Make exceeded limits understandable to the user.

Acceptance check: Duplicate submissions yield one durable result. A transient failure stops after the configured retry limit. A test tenant reaching its cap cannot generate unbounded work, and the team receives the intended alert.

Suggested owner: Backend lead | **Sequence / dependency:** Before broader rollout / agree pilot budgets

Suggested 90-day sequence

- DAYS 01-07 **Close the main trust boundaries** Address Findings 01-02. Define authorized document access and require approval for AI-driven writes.
- DAYS 08-14 **Agree data handling and prove recovery** Address Findings 03-04. Limit logs, document retention decisions, and run an isolated restore.
- DAYS 15-30 **Launch a bounded, observable pilot** Address Finding 05. Use quotas, review outcomes with pilot users, and track failure patterns.
- DAYS 31-90 **Improve from observed evidence** Build a task-specific evaluation set, refine operating documentation, and prioritize measured performance constraints.

What to measure

Unauthorized access attempts rejected; percentage of meaningful AI writes with explicit approval; deletion checks completed; restore time versus the agreed target; duplicate-job rate; AI cost per completed workflow; and pilot users' reported task time.

The dates indicate sequencing, not a delivery commitment or effort estimate. Staffing, implementation details, and third-party dependencies must be validated. No numerical improvement or business result is promised.

What an assessment could look like.

This fictional report shows one way to explain a system, its risks, and the next steps. Your own assessment would start with your decision, system boundaries, and available evidence. The proposal would confirm the questions, scope, access, deliverables, price, and timing.

Reviews are scoped to your system

We agree which applications, environments, repositories, documents, and interviews are useful. The report distinguishes evidence from assumptions and identifies areas that were not verified. Implementation and specialist testing can be scoped separately.

Three ways to start

Option	Price	What it is for
Initial session	Free / 30 minutes	Discuss your goals, timing, and whether I can help. No payment or obligation.
Hourly consulting	\$200 USD / hour	A focused discussion or review with an agreed scope. One-hour minimum; additional approved time in 15-minute increments at \$50.
Custom work	Contact me	An assessment, implementation, or ongoing support with a written scope and quote.

Share a short brief first, then choose a time for the free 30-minute initial session. Paid work begins after we agree the scope, price, and payment arrangements. Applicable taxes and agreed third-party costs are extra.

Useful starting documents

- Inquiry brief
- Proposal template
- Onboarding checklist
- Assessment structure

This sample excludes penetration testing, formal certification, exhaustive source review, legal advice, and compliance attestations. A real assessment is limited to its agreed scope and the evidence available at the time.

Behkish Advisory, operated by Kshathra Technologies Inc. behkish.nz@outlook.com