AI implementation partner checklist

LLM Agents & RAG

Prepared by the RankGlobal Research Team · Reviewed 27 October 2026

Select LLM/RAG developers around grounded answers or safely completed tasks. Require source-access tests, separate retrieval and response evaluation, controlled tool integrations and observable task costs. A suitable partner explains when agent planning is useful, what can be reused, and who maintains source freshness, permissions, evaluations and external model or search dependencies after launch.

WHO THIS BUYER GUIDE IS FOR

Organizations building internal knowledge assistants, document copilots or agents connected to business applications.

Fill the fields and checkboxes in a compatible PDF reader, then save a separate copy for each candidate. You can also print this guide. No information is submitted.

Project / AI workflow

Candidate AI partner

Decision owner

Review date

01 AI service hiring and delivery checks

- [ ] Define supported questions, allowed actions and refusal boundaries.
- [ ] Inspect comparable production RAG or agent evidence and failure traces.
- [ ] Test source permissions, update propagation and business-tool access.
- [ ] Estimate tokens, embeddings, retrieval, storage and review costs per successful task.
- [ ] Validate reusable ingestion and agent components against your sources.
- [ ] Confirm code, index configuration, evaluation data and supplier-account ownership.
- [ ] Train users on evidence checking and source owners on approved updates.
- [ ] Accept evaluation suites, deletion procedures, runbooks and maintenance responsibilities.

Open the current LLM Agents & RAG provider directory

Read the full guides: AI use cases & scope · Costs & ROI · Choosing a partner · Build your brief

AI module cost context & offline quote notes

Illustrative upfront engineering budgets: the topic-specific effort assumptions below are multiplied by Upwork's published $35–$60/hour marketplace band. These are planning examples, not surveyed module prices, provider quotations or enterprise fees. Tools, hardware, tax and ongoing support are extra; supplier minimums may exceed these examples.

Cited knowledge-answer pilot · $1,960–$6,720

56–112 assumed engineering hours. Use one prepared document collection and reviewed questions to demonstrate retrieval, source links, abstention and basic role filters.

Version-aware document comparison workflow · $5,040–$17,280

144–288 assumed engineering hours. Connect one authorized file route to version selection, evidence-linked differences and a responsible reviewer interface.

Bounded case-action agent release · $12,320–$43,200

352–720 assumed engineering hours. Deploy one authenticated case workflow with typed tools, approval controls, confirmed writes, dependency-failure handling and trace logs.

Source: Upwork: AI engineer cost guide

Optional offline quotation notes: record actual supplier figures in one currency and identify included scope, usage limits and exclusions.

Discovery & readiness · one-time	
Build & integration · one-time	
Evaluation & deployment · one-time	
Training & handover · one-time	
Third-party tools, models & APIs · monthly	
Infrastructure & monitoring · monthly	
Maintenance & support · monthly	

Usage assumptions, exclusions and questions to resolve

Add the one-time rows for initial investment. Annualize monthly rows for operating cost; include variable usage and human review. Avoid counting the same managed-service charge both as a platform fee and a direct API expense.

Compare AI project scope

Qualitative scope inclusion, not measured cost or complexity. Core = explicitly scope the work; Conditional = confirm missing inputs; Defer = outside this narrowly bounded scenario.

Workstream	Grounded answer trial	Permission-aware assistant	Operated knowled ge-and-action service
Question and action scope	Core	Core	Core
Live integration · Document and knowledge repositories	Defer	Core	Core
Live integration · Search, vector or retrieval service	Conditional	Conditional	Core
Evidence and action evaluation	Core	Core	Core
Live support · Operated knowledge-and-action service	Conditional	Core	Core

Grounded answer trial

Evaluate a bounded question set against an approved source snapshot; keep tools and live writes disabled.

Boundary: Use approved samples covering question and action scope. Live document and knowledge repositories access is excluded unless the acceptance test requires it; the named pilot reviewer retains operating responsibility.

Permission-aware assistant

Connect maintained sources and retrieval, then add only explicitly authorized tool actions with confirmation.

Boundary: Include the agreed document and knowledge repositories connection. Confirm whether search, vector or retrieval service access is needed for this release; additional sites, departments or channels need a separate scope decision.

Operated knowledge-and-action service

Maintain permissions, indexes, evaluations and bounded tool integrations across the agreed user groups.

Boundary: List the assets, volumes and users covered by operated knowledge-and-action service. Agree monitoring, handover and ongoing support for question and action scope before expanding coverage.

02 Do we need a RAG assistant, an AI agent, or a simpler workflow for our business task?

Choose the architecture from the task rather than the label. A source-grounded answering service needs reliable retrieval, current permissions and tests for unsupported questions; a tool-using agent also needs bounded actions, confirmation rules and recovery. Give the partner real questions, source changes and failed-task examples, then request separate retrieval, answer and action evaluations. Compare the complete successful and unsuccessful task cost, including indexing, model loops and review. Require source-update ownership, observability and a transferable application around any reusable agent framework.

Question and action scope

Ask the AI team to inspect: Real user questions, unanswerable examples and the exact actions the application may perform.

Estimation impact: Broad promises of general assistance expand evaluation and tool-control scope.

Source condition and access

Ask the AI team to inspect: Representative documents, repository APIs, ownership and permission patterns.

Estimation impact: Unstructured files, inconsistent metadata and complex entitlements increase ingestion work.

Quality and operating envelope

Ask the AI team to inspect: Expected usage, response requirements and consequences of unsupported answers.

Estimation impact: More demanding evaluation, lower latency or strict deployment constraints change architecture and recurring cost.

03 Systems, tools and AI integration boundaries

Document and knowledge repositories

AI service requirement: Ingest approved content while preserving source identity and access.

Scope check: Check incremental updates, deletions, version history and connector limits.

Search, vector or retrieval service

AI service requirement: Retrieve and rank relevant authorized evidence.

Scope check: Specify index configuration, filtering and reindexing responsibilities before choosing a product.

CRM, ticketing or application tools

AI service requirement: Perform bounded actions with validated arguments and confirmed outcomes.

Scope check: Test identity, approval steps, retries and termination after a failed or denied action.

04 External software and operating costs

Language-model inference

Charging unit or cost driver: Input/output tokens or other provider units across complete task paths.

Ask before approving the quote: Include retries, planning loops, long contexts and evaluation traffic in the scenario.

Embeddings and indexing

Charging unit or cost driver: Content preparation, embedding calls and reprocessing when sources change.

Ask before approving the quote: Estimate update and deletion workloads, not only the initial corpus.

Retrieval and application infrastructure

Charging unit or cost driver: Search capacity, vector storage, reranking, hosting and data transfer.

Ask before approving the quote: Separate reserved capacity from usage and identify regional or scaling assumptions.

Observability and quality maintenance

Charging unit or cost driver: Traces, evaluation services, retained test data and human review.

Ask before approving the quote: Assign review ownership and include unsuccessful tasks when assessing unit cost.

05 Reusable components, licensing and ownership

Ingestion and retrieval accelerators

Check connector fit, permission enforcement, index portability and the right to maintain the implementation.

Agent frameworks and tool wrappers

Retest action schemas, stop conditions and failure handling; reusable orchestration is not validated business authority.

06 Staff training, documentation and support

Training and adoption

- Train users to inspect evidence, recognize unsupported answers and escalate an incorrect action with its request trace.
- Teach source owners to publish approved changes and operators to run evaluation cases before expanding tool access.

Documents and technical assets

- Provide the source inventory, ingestion rules, permission model and deletion procedure.
- Deliver prompts, tool schemas, model settings, deployment code and evaluation datasets with clear ownership.
- Document cost limits, incident triage, model changes, rollback and human escalation.

Maintenance and support

- Monitor unsupported answers, denied actions, stalled loops and unusual cost per completed task.
- Maintain source connectors, permission propagation and regression evaluations when models or business tools change.

07 From business requirement to working AI service

LLM agent and RAG development services: hire for grounded answers and controlled actions

LLM and RAG implementation services should start with the work a user needs to complete. Retrieval helps when an answer depends on your approved content; tool execution is needed when the assistant must act in another system. These require different evidence, permissions and failure handling. Ask the partner to justify each capability against

the task rather than proposing an agent framework first.

The estimate should cover document preparation, access-aware ingestion, retrieval evaluation, response testing, integrations and operational support. Reusing an ingestion connector or agent skeleton can reduce setup effort, but your source permissions, task limits and acceptance questions still require custom validation. Include model calls, embeddings, search, storage and human review in the operating budget, together with source-update and deletion maintenance.

08 How an LLM/RAG partner should implement a grounded application

Validate retrieval and tool execution separately so a convincing answer cannot conceal a missing source or failed action.

1. Bound the answer or task

Collect real questions and required business outcomes. Identify unanswerable cases, restricted material and actions that must remain with a person.

Accountable team: AI product analyst and workflow owner

Data / system: Question set, task map and approved source inventory

2. Prove source and tool access

Inspect source APIs and permissions; test authorized system functions outside the model. Establish document identities, update behavior and action schemas.

Accountable team: Retrieval and integration engineers

Data / system: Content repositories, identity service and system sandboxes

3. Evaluate the components and complete journey

Measure retrieval support, grounded response quality and successful tool execution. Include stale sources, conflicting evidence, denied access and repeated action attempts.

Accountable team: LLM evaluator and domain reviewers

Data / system: Versioned evaluation set, traces and transaction records

4. Release with observable limits

Set cost and action boundaries, escalation and rollback. Train source owners and users, then hand over ingestion, evaluation and incident procedures.

Accountable team: Application owner and operations engineer

Data / system: Monitoring, model configuration and deployment repository

09 AI use cases and evaluation requirements

1. Internal knowledge assistant

Retrieve approved procedures and answer employee questions with accessible source references.

Inputs: Versioned documents, realistic questions and role permissions.

AI partner deliverables: A retrieval pipeline, cited answers and an evaluation set of answerable and unanswerable questions.

Measure the result: Relevant-source retrieval, supported claims and access-isolation failures.

Failure to test: The answer cites a passage that contains similar words but does not support the conclusion.

2. Document comparison assistant

Compare specified versions or records and present differences for a responsible reviewer.

Inputs: Authorized files, version identities and reviewed comparison examples.

AI partner deliverables: Evidence-linked differences with unresolved ambiguities visible.

Measure the result: Missed material differences and reviewer correction effort.

Failure to test: The system combines passages from different versions without indicating which governs.

3. Service agent with bounded actions

Look up an authorized case, propose a permitted step and execute it only within validated rules.

Inputs: Case records, tool schemas, access policies and escalation routes.

AI partner deliverables: A tool-controlled agent with confirmation, failure handling and traceability.

Measure the result: Verified task success, incorrect writes and recovery from dependency failures.

Failure to test: A tool instruction from retrieved content is treated as authorization.

4. Analyst research workspace

Retrieve and assemble relevant evidence across an approved collection while keeping source attribution inspectable.

Inputs: Curated sources, query examples and evidence-review criteria.

AI partner deliverables: A research interface with source navigation and reproducible traces.

Measure the result: Useful evidence coverage and unsupported assertions requiring correction.

Failure to test: A polished narrative hides the absence of contradictory or missing evidence.

10 Specialist team and relevant deployment evidence

Retrieval and search specialist

Diagnoses parsing, chunking, query and ranking failures independently of answer generation.

Evidence to request: A retrieval evaluation showing relevant passages retrieved and missed.

LLM application engineer

Designs structured responses, abstention, traces and reproducible task evaluations.

Evidence to request: Failure analysis separating source, prompt and model behavior.

Identity and security engineer

Enforces document and tool access beyond model instructions.

Evidence to request: Tests with users who should see different documents and actions.

Backend integration engineer

Makes agent actions valid, duplicate-safe and observable in real systems.

Evidence to request: A failed-tool demonstration with correct reported outcome and recovery.

11 AI implementation milestones and acceptance gates

Question and collection study

Assemble genuine user questions, source passages, missing-answer cases and permission scenarios.

Acceptance gate: The content owner agrees which collection is authoritative and sufficiently complete.

Retrieval baseline

Evaluate parsing and search before adding agent planning or elaborate prompts.

Acceptance gate: Relevant evidence reaches the response stage consistently on the held-out questions.

Grounded response and tool tests

Test answer support, incorrect premises, malicious content and unauthorized tool requests.

Acceptance gate: The system abstains, restricts or escalates where evidence or authority is absent.

Observed workflow release

Introduce a defined user group and inspect traces, failures, costs and content updates.

Acceptance gate: Operators can identify the source and action versions behind a disputed result.

12 Project scope and quotation questions

Collection preparation

Scanned files, complex tables, duplicate versions and missing permissions increase ingestion work.

Ask in the quotation: Does the quote include parsing, cleanup, access metadata and content-change handling?

Retrieval design

Embeddings, indexing, reranking and freshness requirements affect both build and operation.

Ask in the quotation: How will the team prove that the additional search complexity improves the target questions?

Inference and agent steps

Long context, multiple model calls, repeated tool attempts and evaluations influence per-task cost.

Ask in the quotation: What is measured cost per completed task, including failed attempts and review?

Integration assurance

Each writable tool adds validation, identity, recovery and testing responsibilities.

Ask in the quotation: Which tools are read-only, which may write and which require approval?

13 Hiring and comparing AI implementation partners

Comparable grounded applications

Ask: Show a deployed answer or task backed by inspectable evidence.

Useful evidence: Approved sources or confirmed actions linked to user-request traces.

Warning sign: The demonstration relies on public facts unrelated to the buyer's material.

Candidate notes

Retrieval and permission evaluation

Ask: How do you distinguish a retrieval miss from a generation error?

Useful evidence: Component results and tests using inaccessible, missing and conflicting documents.

Warning sign: A single chatbot satisfaction score replaces source-level evaluation.

Candidate notes

Agent necessity and action control

Ask: Which tasks benefit from multi-step planning, and what limits the actions?

Useful evidence: A simpler baseline comparison, scoped tool permissions and tested stop conditions.

Warning sign: More agents are presented as evidence of a better solution.

Candidate notes

Cost and source lifecycle ownership

Ask: Can we inspect consumption and remove a source completely?

Useful evidence: Per-task traces, access updates, deletion tests and exportable configuration.

Warning sign: The partner cannot explain index freshness or failed-loop costs.

Candidate notes

14 Demonstrations, trial terms and hiring commitments

Ask whether a no-cost demonstration, sandbox or representative-sample evaluation is available. A free consultation does not imply free implementation; custom pilots are often paid. Agree workflow scope, what is free, usage limits, when billing starts, data handling, acceptance evidence, output ownership and the exit path before supplying data.

Relevant demonstration: A retrieval pipeline, cited answers and an evaluation set of answerable and unanswerable questions.

Failure to introduce: The answer cites a passage that contains similar words but does not support the conclusion.

15 Practical AI buying tips

- Ask a question whose answer is deliberately absent from the source collection.
- Demonstrate a permission change and a deleted document, not only initial indexing.
- Evaluate cost per verified completed task rather than cost per model call alone.

16 Questions buyers ask about AI implementation

When does a business assistant need an agent instead of retrieval alone?

Use retrieval when the task is to find and explain permitted information. Consider tool use or planning only where completing the task requires additional controlled steps. Ask the developer to justify each action and loop, define stopping conditions and compare a simpler fixed workflow before adding agent behavior that expands cost and evaluation scope.

How can RAG respect different employees' document permissions?

The application must enforce source access during retrieval and avoid exposing restricted information through generated answers or retained context. Ask the team to test the same question under users with different permissions, including removed access. A source citation alone does not prove the underlying retrieval or conversation storage honors authorization.

What should happen when a RAG assistant cannot find a supported answer?

The assistant should identify missing evidence or request clarification rather than fill the gap with a plausible claim. Include unanswerable and conflicting-source examples in evaluation. Show staff enough retrieved context and source version information to investigate the failure, and assign an owner to correct the source or the retrieval behavior.

Why is the model token price not the total cost of a RAG or agent task?

A task can trigger retrieval, embeddings, reranking, multiple model calls, tool requests and retries before producing an accepted result. Add indexing, storage, application hosting and quality review. Ask the supplier to show complete task traces and cost assumptions for failures and long-context cases, not only one model response.

Which RAG maintenance costs recur when our knowledge changes?

Include source synchronization, reindexing, deletion handling, permission updates and regression evaluation. Content changes may create new answer failures even when the application code is unchanged. Agree which source connectors and update volumes are included, who reviews unresolved questions and how model or retrieval-service changes are tested before release.

How should we compare the ROI of an internal knowledge assistant?

Measure useful, supported answers and the total time staff spend finding and verifying information against the existing method. Include abandoned questions, correction effort and remaining specialist review. Subtract application, model, indexing and maintenance costs; answer volume by itself does not demonstrate that employees can complete their work more effectively.

What should an LLM or RAG developer demonstrate with our own material?

Provide an unseen question, a restricted document and an unsupported request. Inspect retrieved evidence, the answer's claim support and the access behavior. If the assistant can act, also test a failed tool call and stopping condition. These tests distinguish application engineering from a polished prompt connected to a convenient document sample.

Which reusable agent components can reduce development effort?

Connectors, retrieval modules, evaluation harnesses and review interfaces may be reusable if they fit your access and workflow requirements. Ask which pieces are pre-existing, licensed or custom-built and what must be retested on your data. Confirm modification, export and support rights before treating an accelerator as a durable saving.

What handover should we require from an agent development partner?

Require source-connector configuration, prompt and tool definitions, evaluation cases, deployment instructions and cost observability. Document source ownership, permissions, action boundaries and failure escalation. Have your team update a source and investigate a failed task during handover so maintenance does not depend on undocumented knowledge held by the original developer.

17 Primary sources and editorial scope

1. Azure RAG solution design and evaluation

Primary architecture guidance for the RAG pipeline.

https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/rag/rag-solution-design-and-evaluation-guide

2. Microsoft: End-to-end RAG evaluation

Primary background distinguishing retrieved grounding information from the language-model response stage.

https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/rag/rag-llm-evaluation-phase

3. Azure agentic RAG architecture

Primary description of dynamic retrieval planning and its complexity tradeoffs.

https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/rag/rag-agentic

Original AI-service buying guidance. Verify current interfaces, licences, data rights and provider evidence. No supplier certification, project quotation or guaranteed outcome is implied.