

Operationalising responsible AI in UAE enterprises

A practical operating model for accountable deployment

Release an AI service only when a named owner can explain its purpose, show how it was tested, authorise its remaining risks and stop it safely. This paper translates ethical principles into engineering controls, review decisions and operating evidence for UAE enterprises.

It is written for CIOs, risk and compliance leaders, product owners and engineering teams deploying enterprise assistants or retrieval augmented generation (RAG). RAG supplies a model with retrieved documents at request time; the application still needs controls over the sources, permissions and resulting answer.

Five decisions to make before launch

- Assign a business owner with authority over scope, funding and residual risk. Keep independent testing separate from the build team.
- Classify the use case by the harm it could cause, the data it processes and the actions it can take. Reassess when these change.
- Test Arabic and English separately, including refusal, escalation and access-control failures. A blended accuracy score can hide a weak service.
- Enforce permissions outside the model and make human review an operational capability with time, evidence and an escalation path.
- Approve a versioned release with evidence, monitoring thresholds and a rehearsed rollback. Treat model and retrieval changes as releases.

What this paper provides

Pages 2 and 3 define governance, risk tiers and responsibilities. Pages 4 and 5 specify technical controls and evaluation. Pages 6 and 7 work through a fictional RAG service and its approval evidence. Page 8 sets out a 90 day adoption sequence and primary sources.

Scope and status. The methods, tiers and thresholds are CodeX71 proposals for tailoring to an organisation. They are not a certification, legal opinion or assurance that a system is free from harm. The worked example is fictional, not a customer result.

01 GOVERNANCE AND RISK

Separate obligations from principles

The UAE Charter for the Development and Use of Artificial Intelligence sets out principles including safety, privacy, bias management, transparency, human oversight and accountability. Use it to shape policy and service design; do not treat alignment with the Charter as proof of legal compliance. [1]

NIST AI RMF 1.0 is voluntary. Its Govern, Map, Measure and Manage functions provide a useful structure for risk work; the Generative AI Profile adds model-specific considerations. UNESCO's Recommendation provides an international ethics reference rather than a standalone enterprise licence or compliance certificate. [2][3][4]

Maintain an obligations register

Applicable legislation and regulatory requirements are a separate track. The UAE federal Personal Data Protection Law has a defined scope and exclusions, including certain government, health, banking and free-zone contexts. Identify the entity, sector, data and processing location before selecting the relevant regime. [5]

Have qualified counsel and the relevant control owners record each applicable requirement, the control that addresses it, its evidence and review date. Check sector and free-zone rules, contractual obligations and cross-border data conditions. A UAE-hosted model alone does not establish compliance or settle where support staff, logs and subprocessors process data.

An example internal risk classification

These are proposed enterprise tiers, not UAE statutory categories. Assign the highest applicable tier; record inherent risk before controls and residual risk afterwards.

Tier	Typical use and consequence	Proposed approval
1 Limited	Public-content drafting with trained review; no confidential data or automated action.	Business owner; basic quality and security checks.
2 Controlled	Internal knowledge assistant; incorrect advice or restricted-data leakage could affect operations.	Owner plus documented security, privacy and independent test reviews.
3 Material	Advice influencing safety, employment, credit, healthcare or access to essential services.	Specialist legal and domain review; strengthened oversight and accountable executive approval.
Stop	Unlawful use, unmanaged severe harm, unresolved access leakage or no workable human recourse.	Do not launch; redesign or withdraw.

A low-risk interface can become a material-risk service when it gains personal data, external users or write access. This paper does not recommend autonomous high-impact decisions about people. Meaningful human judgement and applicable professional duties remain necessary.

02 OPERATING MODEL

Make ownership visible across the lifecycle

Start with a short system record: purpose, users, affected people, excluded uses, data sources, model and hosting providers, tools, risk tier and accountable owner. Attach evidence to this record throughout the lifecycle. The diagram is CodeX71's proposed workflow, informed by the NIST functions. [2]

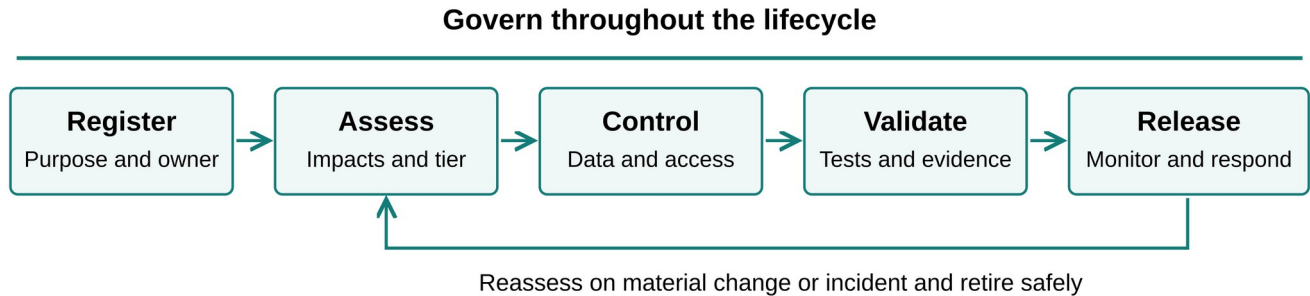


Figure 1 Proposed release lifecycle with governance and change review throughout

Assign one accountable role for each decision

RACI means Responsible, Accountable, Consulted and Informed. The matrix below is a starting point. A small organisation can combine roles, but the builder should not be the sole reviewer of a material-risk release.

Decision or activity	Owner	Build	Control	Test	Ops
Purpose and risk tier	A/R	C	C	C	I
Data and security controls	C	R	A	C	C
Independent evaluation	C	C	C	A/R	I
Release and residual risk	A	R	C	C	C
Incident containment	A	C	C	I	R
Material change or retirement	A	R	C	C	R

Owner is the business sponsor. Build is engineering. Control is the nominated security, privacy or risk lead for the decision. Test is the independent evaluator. Ops is the service operator. Required legal, privacy or security sign-offs remain mandatory under the organisation's policy; a RACI does not override them.

Give oversight enough authority to work

Reviewers need the source passage, its date, uncertainty and the proposed action, with enough time to disagree. They must be able to stop or correct the workflow without penalty. Test whether staff detect planted mistakes; a mandatory approval click alone does not establish meaningful oversight.

Disclose the AI-supported nature of the service and its limitations in language users understand. Provide a human route for complaints, corrections and contested outcomes. Include affected users in service testing and review whether the process is accessible to people with different language and digital skills.

03 DATA SECURITY AND SUPPLIERS

Build controls around the whole service

The model is only one component. Map the full route from source ingestion to retrieval, model request, answer, logs and downstream tools. Record trust boundaries and who can read or change each component. Apply the following proposed control design before live data enters the service.

Protect sources and retrieval

- Ingest only approved sources with an owner, version, effective date and retention rule. Quarantine unexpected files and record checksums or immutable versions. Remove retired documents from indexes and caches, then test that retrieval no longer returns them.
- Enforce document and tenant permissions before retrieval and recheck authorisation before returning content. Keep access metadata attached to chunks. Test cross-tenant and cross-role queries, including indirect hints and summaries.
- Classify and minimise personal data in prompts, embeddings, traces and feedback. Treat embeddings as potentially sensitive. Redact logs, restrict access and verify deletion across derived stores. Confirm lawful processing, retention and transfer conditions with the appropriate owner. [5]

Constrain prompt injection and agent actions

An attacker can place instructions in a document that the model later retrieves. Treat retrieved text and tool output as untrusted data. Separate it from system instructions, validate outputs and test encoded, multilingual and indirect attacks. Prompt wording and a second model used as a guardrail cannot provide a complete security boundary. [6]

Keep authorisation, schema checks and transaction rules in deterministic code. Use least-privilege service identities, tool allowlists, network egress restrictions and execution limits. Require a human confirmation for consequential actions. A read-only pilot is easier to contain, but can still leak data or mislead users. [6]

Require evidence from model and platform vendors

Question to resolve	Evidence to retain
Where can our data go?	Contracted regions, subprocessors, support access, retention and deletion terms; training-use settings.
What can change without approval?	Model version identifiers, update and deprecation notices, rollback options and regression-test commitments.
How are failures handled?	Security reports with scope and date, incident contacts, notification terms, service limits and exit/export plan.

Verify the configuration that is actually purchased. A provider's benchmark or certification does not establish that your retrieval permissions, Arabic service quality or human review process are adequate. Record unresolved gaps and who accepts or remediates them.

04 TESTING AND FAIRNESS

Evaluate the service in Arabic and English

Use a versioned test set drawn from the intended tasks, including questions the service must refuse or escalate. Keep evaluation items out of prompt tuning. Arabic evaluation should include Modern Standard Arabic and the dialects, transliteration and code-switching that the service claims to support. Validate source translations with bilingual domain reviewers.

Measure failures that matter to users

Measure	Definition and diagnostic use
Answer correctness	Correct, policy-consistent answers / eligible questions. Grade against current authoritative material, not plausible wording.
Supported claims	Claims supported by the cited source / factual claims reviewed. Check that the citation actually entails the claim.
Appropriate escalation	Correct escalations / cases requiring escalation. Also count unnecessary refusals on answerable questions.
Retrieval and access	Relevant-source recall at a chosen retrieval depth; unauthorised disclosures / access-control test attempts.
Service parity	Compare error, refusal and escalation rates for matched tasks by language and relevant user cohorts. Report sample counts.
Human performance	Reviewer detection of seeded errors, override quality, review time and backlog. Check whether oversight works under load.

Publish the denominator, test version and grading rubric beside each result. Show uncertainty intervals when making comparisons; small or uneven samples cannot establish equality. Record reviewer disagreements, adjudicate them and keep a challenge set for retesting. Use automated graders as aids, calibrated against human review.

Trace disparities to their cause

A higher Arabic error rate may originate in missing Arabic source material, document extraction, chunking, retrieval, translation or generation. Tag failures at each stage before changing the model. Match task difficulty when comparing languages; do not infer demographic fairness from language parity alone.

Assess other relevant harms with lawful, proportionate data: differential access, stereotyped responses, inaccessible interfaces or extra burden on a user group. Consult affected users. Avoid collecting sensitive attributes merely to fill a dashboard. There is no single fairness score that resolves every context or trade-off.

Agree release criteria before seeing results

Define a floor for each supported language, severe-failure blockers and what triggers investigation after launch. Include adversarial tests, stale or contradictory sources and safe failure during outages. NIST's Generative AI Profile advises checking source citations and cautions against generalising performance from narrow assessments. [3]

05 ILLUSTRATIVE APPLICATION

Work through a fictional RAG approval

Fictional scenario. Gulf Meridian Services, an invented UAE enterprise, wants an Arabic and English procurement-policy assistant. Staff can ask about approved purchasing procedures. The system cannot select suppliers, approve spend or access employee, payroll or customer records. All figures below are invented to demonstrate a decision process.

Set the boundary and the evidence

The owner assigns internal Tier 2 because incorrect guidance could cause unauthorised purchases. Only current, approved policy documents enter the index. Staff authenticate with enterprise identity; role-specific appendices are filtered before retrieval. The assistant cites a policy section and effective date, or sends the question to procurement.

The release package pins the model, prompt, embedding model, index snapshot and access-policy version. The team writes 120 matched questions per language, balanced across routine, ambiguous and out-of-scope cases, plus separate adversarial and access-control tests. A failure is an incorrect answer or inappropriate refusal/escalation under a pre-agreed rubric.

Illustrative test	First candidate	Revised candidate
English task failures	4 / 120 (3.3%)	3 / 120 (2.5%)
Arabic task failures	14 / 120 (11.7%)	5 / 120 (4.2%)
Observed language error gap	8.3 percentage points	1.7 percentage points
Unauthorised disclosures	0 / 40 attempts	0 / 40 attempts
Injection policy violations	3 / 40 attempts	0 / 40 attempts

Table 1 Synthetic results only. Rates are rounded; gaps are computed from unrounded values. Counts do not imply demonstrated production safety.

Use the results to make a bounded decision

The fictional team's preset criteria allow at most 5% task failures in each language, an observed gap no greater than 5 percentage points and zero observed access leaks or severe injection violations. These are example thresholds, not standards. The first candidate fails. Review finds that the Arabic index lacks current policy appendices and that retrieved instructions can override purchasing guidance.

The team repairs ingestion, tests source completeness, separates source text from instructions and strengthens output validation. It evaluates the revised candidate on a fresh equivalent holdout set and reruns the attack suite. The point estimates now meet the example criteria; uncertainty and limited coverage remain. Zero failures in 40 attempts is not proof of zero risk.

The owner can authorise a limited, read-only pilot after the full approval checklist passes. Staff review the cited policy before acting. Procurement samples answers daily, and any restricted-data disclosure pauses the service immediately. Expansion requires new evidence, not reuse of the pilot approval.

06 RELEASE EVIDENCE AND RESPONSE

Approve a release and prepare to stop it

Maintain one evidence package per released configuration. A ticket saying "approved" is insufficient if reviewers cannot recover the versions, tests and limitations behind the decision. The checklist below is CodeX71's proposed minimum for a controlled enterprise assistant.

Approval evidence checklist

- Purpose and scope: named owner, users, affected people, exclusions, risk tier, benefits to test and approved use boundaries.
- Obligations and data: applicable-regime assessment, lawful processing review, data flow, source rights, retention, transfers and supplier terms.
- Architecture and security: threat model, access-control tests, source provenance, secrets handling, tool permissions and injection test report.
- Evaluation: Arabic and English results, denominators, holdout method, subgroup limitations, severe failures, reviewer checks and open defects.
- Human oversight: trained reviewers, review capacity, source visibility, escalation route, user disclosure and correction or complaint process.
- Operations: monitoring thresholds, alerts, on-call ownership, fallback service, tested rollback and evidence-retention settings.
- Decision: pinned release manifest, required sign-offs, residual risks, exception owner, expiry or review date and next reassessment trigger.

Respond to an incident in a defined order

- 1 Contain. The operator disables the affected route, tool or model, revokes compromised access when appropriate and directs users to a human or verified static guidance. Pre-authorise emergency containment so staff need not wait for a committee.
- 2 Preserve and assess. Record times, versions, affected users and sources, while minimising exposed content in incident records. Protect relevant evidence. Security, privacy and legal owners determine impact, notification duties and applicable deadlines; this paper sets no universal legal deadline.
- 3 Recover. Restore a tested configuration only if its model, index, prompt and access policy remain compatible. Rollback must not resurrect deleted data or old permissions. If safe restoration is uncertain, keep the workflow disabled and use the fallback.
- 4 Learn and reapprove. Reproduce the failure, fix its root cause, expand the regression suite and revalidate both languages. Record what users were told and how harm or incorrect advice was corrected. A new release needs an explicit decision.

Monitor output quality, refusals, complaints, retrieval freshness, permission failures, latency, cost and reviewer workload. Set review frequency by risk and traffic. Reassess after model, source, audience, purpose or tool changes; retire the service when its owner or control evidence is no longer valid.

07 IMPLEMENTATION AND SOURCES

Adopt the method in 90 days

The sequence below is an example for a bounded internal assistant. A material-risk service may need longer; the calendar does not replace release criteria.

Days 1 to 15. Inventory AI use, choose one narrow pilot, name the owner and map applicable obligations. Record the human baseline, data boundaries and excluded uses.

Days 16 to 35. Approve sources and supplier terms. Implement identity and retrieval controls, the evidence store and a fallback. Agree evaluation rubrics and release thresholds.

Days 36 to 60. Run bilingual holdout and adversarial tests, close severe defects and rehearse human review and rollback. Produce the first approval package.

Days 61 to 90. Run a bounded pilot with active monitoring. Compare quality and review effort with the baseline. Approve expansion, remediate or stop on the evidence.

Define the next decision with CodeX71

A proposed CodeX71 workshop can scope one use case, its risk classification, control owners, bilingual evaluation plan and evidence required for a pilot decision. A managed-agent assurance review, including an AYNAI-scoped engagement where relevant, should state the systems, controls and limitations being assessed. It is not a compliance guarantee.

[Discuss a responsible AI workshop](#)

Primary sources

Source review date 1 October 2026. Numbered references support the framework and legal context; diagrams, matrices, checklists, thresholds and scenario are original implementation proposals. Verify current law and provider terms before a deployment decision.

[1] UAE Government. [UAE Charter for the Development and Use of AI](#). July 2024; official portal updated 3 July 2026.

[2] NIST. [AI Risk Management Framework 1.0](#). NIST AI 100-1, January 2023.

[3] NIST. [Generative Artificial Intelligence Profile](#). NIST AI 600-1, July 2024.

[4] UNESCO. [Recommendation on the Ethics of Artificial Intelligence](#). Adopted 23 November 2021.

[5] UAE Government. [Federal Decree Law No 45 of 2021 on Personal Data Protection](#). Article 2 scope; official legislation.

[6] OWASP. [LLM Prompt Injection Prevention Cheat Sheet](#). Current online guidance.

[7] NIST. [AI Risk Management Framework programme page](#). AI RMF 1.0 revision in progress at review date.

Version note. This edition uses the published AI RMF 1.0 and 2024 Generative AI Profile. NIST reports that AI RMF 1.0 is being revised; review the programme page before adopting a later edition. [7]