INTRODUCTION TO AI SERIES · BOOK 2

2026 EDITION · INTERMEDIATE

ENTERPRISE TECHNICAL FIELD GUIDE

Enterprise AI Automation, RAG & Workflow Implementation

Move AI out of isolated chat windows and connect it directly to company documents, CRM/ERP systems, and operational workflows with strict RBAC, reliability patterns, and measurable ROI.

5 Diagrams	10 Blueprints	60 Days	15 Pages
VECTOR ARCHITECTURES	PROMPTS, SQL & JSON	8-WEEK PILOT PLAN	FREE DIRECT PDF

Hybrid RAG (BM25 + Vector) · n8n / Make vs Laravel & Python · WhatsApp · CRM · ERP · Retrieval-Layer RBAC · Human-in-the-Loop (HITL) · Golden Q&A Evals



Ferhat Balaban
Software Architect & Enterprise AI Trainer · 20+ Years Experience

ferhatbalaban.com

Open Access · Free for internal team sharing

GUIDE OVERVIEW & NAVIGATION

Table of Contents

Most organizations treat AI as a personal chat tool for the first six months. Real competitive advantage starts when AI securely reads internal SOPs, classifies inbound tickets, extracts invoice fields, and updates CRM/ERP systems with full auditability. This guide documents the field-tested architecture patterns we deploy with engineering and operations teams.

-62%
Average reduction in internal document lookup time after hybrid RAG rollout

≥ 95%
Target citation accuracy on a 40–60 case Golden Q&A suite before production

60 Days
From single-process data audit to measured pilot ROI and scale sign-off

01 **RAG Architecture: Assistants Over Company Documents** p. 03
Ingestion, chunking strategies, hybrid search (BM25 + vector), reranking, and citation-backed generation.

02 **n8n / Make vs Custom Engineering (Laravel / Python)** p. 05
Objective comparison across speed, RBAC, version control, auditability, cost at scale, and hybrid graduation.

03 **CRM, ERP, Email & WhatsApp Integration Patterns** p. 07
Lead qualification, support triage, invoice extraction, webhook idempotency, queues, and HITL gates.

04 **Data Hygiene, Access Control (RBAC) & Hallucination Prevention** p. 09
Document versioning, retrieval-layer RBAC filtering, low-temperature grounding, and Golden Q&A validation.

05 **60-Day Pilot Automation & RAG Roadmap** p. 11
An 8-week phased rollout from data audit and prototype to closed pilot, Go/No-Go gate, and ROI sign-off.

06 **Practical Appendices: Blueprints, Checklist, Scorecard & Glossary** p. 13
Production prompt & JSON schemas, 10-item launch checklist, 0–20 readiness scorecard, and glossary.

Who Should Read Book 2?

- **IT & Operations Leaders:** Decision matrices, security gates, and 60-day ROI governance.
- **Laravel / Python / n8n Engineers:** Production webhook, pgvector RBAC, PII redaction, and JSON patterns.
- **Product Owners:** How to graduate safely from a 2-week prototype to a hardened service.

How to Use This Guide

- **30-Min Executive Read:** Focus on the 5 architecture diagrams, case studies, and Chapter 5 roadmap.
- **Engineering Implementation:** Copy the SQL, PHP, Python, and JSON schemas directly into your repo.
- **Team Assessment:** Score your readiness (0–20) on page 15 before starting Week 1.

CHAPTER 01 · PRIVATE KNOWLEDGE ARCHITECTURE

01 · RAG Architecture: Assistants Over Company Documents

Why Retrieval-Augmented Generation beats fine-tuning for living business documents, and how to engineer hybrid retrieval, chunking, reranking, and strict source citations.

Enterprise documents change constantly: price lists are revised quarterly, HR leave policies are updated, and technical SOPs evolve with every product release. Baking this knowledge into model weights via **fine-tuning** is slow, expensive, and impossible to audit at the paragraph level. **RAG (Retrieval-Augmented Generation)** solves this by keeping the LLM stateless and injecting only the exact, authorized, up-to-date document excerpts at query time.

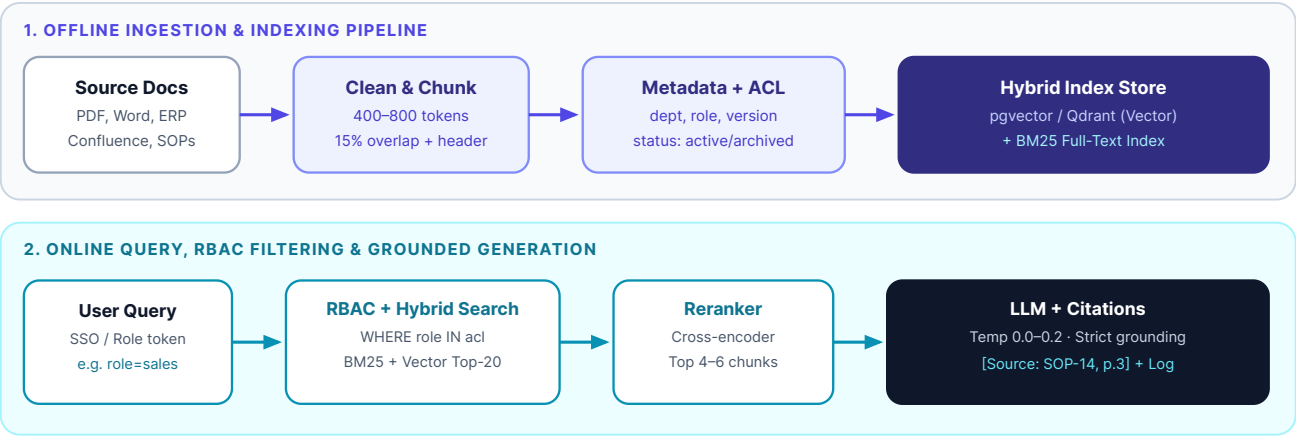


Figure 1.1: Two-stage Enterprise Hybrid RAG architecture (offline indexing + online RBAC retrieval). Ref: Production Pattern

1.1 RAG vs Fine-Tuning vs Long-Context Window

Criterion	Hybrid RAG (Recommended)	Fine-Tuning	Long-Context Stuffing
Freshness	Instant Re-index changed doc in seconds.	Slow Requires retraining cycle.	Manual Must re-send full doc every call.
Citations & Audit	Exact `[doc_id, page]` provenance per claim.	Cannot reliably cite source paragraphs.	Weak grounding across 100k+ tokens.
RBAC Security	Hard SQL/vector filter before LLM sees text.	All users share weights (leak risk).	Must filter docs before prompt assembly.
Cost at Scale	Low — sends only Top 4–6 relevant chunks.	High GPU training + dedicated hosting.	High token cost & latency per query.
Best Fit	SOPs, contracts, ERP manuals, support KBs.	Strict tone, domain syntax, or niche classification.	One-off analysis of 1–3 standalone PDFs.

1.2 Chunking Strategy: Where 70% of RAG Quality Is Won or Lost

Naive fixed-character splitting slices right through pricing tables, warranty clauses, and numbered steps. Production ingestion pipelines parse document structure first (headings, Markdown tables, list blocks) and prefix every chunk with its **hierarchical breadcrumb** (e.g., `[Doc: Return Policy v4.2 > Section 3: B2B Returns]`).

WEAK PATTERN · NAIVE SPLITTING

1,000-char blind split + vector-only search: Breaks table headers away from rows, misses exact SKU codes like `INV-2026-441`, and returns outdated 2024 PDFs alongside 2026 files.

PRODUCTION PATTERN · STRUCTURE + HYBRID

400–800 token semantic chunks + 15% overlap + BM25 + Vector + Reranker: Preserves tables as intact Markdown, matches exact product codes via BM25, and filters `status = 'active'`.

1.3 Hybrid Search + Cross-Encoder Reranking

Dense vector embeddings excel at semantic paraphrases (*"how do I cancel an annual seat?"* matching *"subscription termination procedure"*), but struggle with exact identifiers such as part numbers, tax codes, or contract IDs. Combining **BM25 lexical search** with **vector similarity** via Reciprocal Rank Fusion (RRF) pulls the top 20 candidates, and a lightweight **cross-encoder reranker** narrows them to the best 4–6 chunks before calling the LLM.

FIELD CASE STUDY #1 — **Industrial B2B Equipment Distributor · 140 Staff**

Technical Support & Spare-Parts Hybrid RAG Assistant

Challenge: Field technicians spent 18–25 minutes searching 1,400 PDF service manuals and warranty bulletins on SharePoint to verify torque specs and spare-part SKUs. Pure vector search missed exact part codes (`VALVE-44B`) 38% of the time.
Solution: Re-indexed manuals into structure-aware 600-token chunks in PostgreSQL (`pgvector` + `tsvector` BM25) with `status='active'` filtering and a cross-encoder reranker, enforcing clickable `[Manual, Page]` citations.

22 min → 2.4 min	96.4%	4.5 Weeks
Median spec lookup time	Verified citation accuracy	Time from audit to live pilot

3 COMMON RAG ANTI-PATTERNS TO AVOID

- **Dumping an entire SharePoint drive uncurated:** Indexing draft, duplicate, and expired files guarantees contradictory answers.
- **Sending 20 chunks to the LLM "just in case":** Dilutes attention ("lost in the middle"), raises cost 4x, and increases hallucination.
- **Skipping the refusal rule:** If top reranker score is below threshold (e.g. `< 0.35`), force the assistant to say *"Not found in authorized documents"*.

CHAPTER 02 · BUILD VS ORCHESTRATE

02 · n8n / Make vs Custom Engineering (Laravel / Python)

How to avoid both traps: over-engineering a simple workflow in custom code, or running mission-critical multi-tenant business logic inside brittle visual flows.

Visual automation platforms like **n8n** and **Make** allow a single engineer or ops specialist to ship a working AI workflow in 48 hours. However, once volume crosses thousands of events per month or compliance demands strict RBAC, automated unit tests, and Git code reviews, visual canvases become hard to maintain. The smartest organizations follow a **3-Stage Graduation Strategy**.

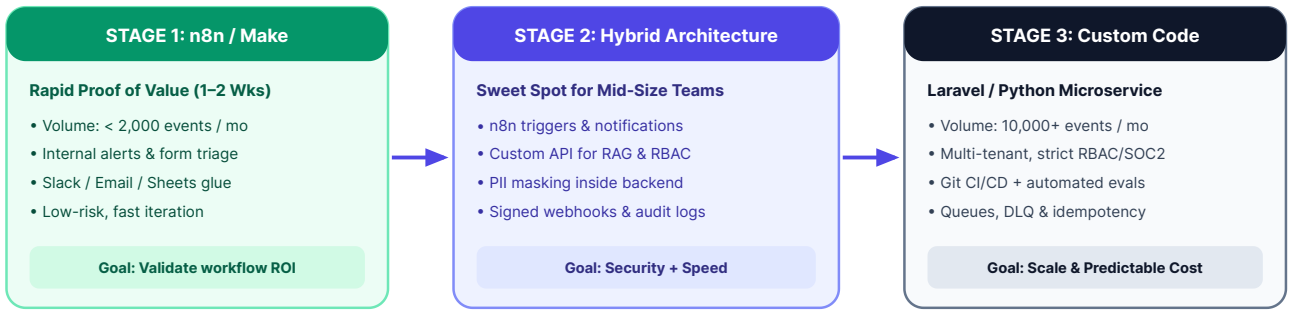


Figure 2.1: Graduating from visual prototyping (n8n/Make) to Hybrid and Custom Code without rewrites. Ref: Architecture Decision

2.1 Objective Architecture Decision Matrix

Dimension	n8n / Make	Hybrid (n8n + Custom API)	Custom (Laravel / Python)
Time to First Demo	1–3 Days Drag-and-drop nodes.	1–2 Weeks Core API + visual glue.	3–6 Weeks Full repo, CI/CD, infra.
RBAC & Privacy	Basic credential vaults; risky for per-user ACLs.	RBAC & PII redaction enforced inside backend API.	Full SSO, tenant isolation, row-level security.
Testing & Git	Hard to diff JSON flows or run CI evals.	Core logic unit-tested in Git; triggers in n8n.	100% Git PR review, PHPUnit/Pytest, CI evals.
Error Handling	Basic retry; complex dead-letter queues are clunky.	Queue/idempotency in backend; alerts in n8n.	Redis/SQS queues, exponential backoff, DLQ.
Cost at 50k Events/Mo	Make ops fees spike; self-hosted n8n needs tuning.	Predictable: LLM API + modest app server.	Lowest marginal cost per 1,000 events.

2.2 The 5 Triggers That Signal "Graduate to Custom Code"

Stay on n8n / Make When...

- Monthly volume is under **2,000–3,000 runs**.
- Failures are non-financial (internal Slack summaries, CRM note drafts, form routing).
- No per-user document permission filtering is needed.
- You are still validating whether the team actually uses the workflow.

Move to Laravel / Python When...

- You touch **invoices, ERP stock, customer payouts, or external emails**.
- Multiple roles or tenants require hard **RBAC isolation**.
- You need **idempotency**, rate limiting, and audit logs for ISO 27001 / SOC 2.
- More than 2 engineers maintain the logic concurrently.

2.3 Total Cost of Ownership (TCO) Snapshot — 10,000 Monthly Events

Cost Component	Make (Cloud SaaS)	Self-Hosted n8n + API	Custom Laravel / Python
Orchestration / Hosting	$180–$450 / mo (ops tiers)	$40–$80 / mo (VPS + Docker)	$50–$120 / mo (App + Worker)
LLM API (Mini + Frontier mix)	$60–$140 / mo	$45–$110 / mo (prompt cache)	$35–$90 / mo (model routing)
Engineering Maintenance	Low initially, high after 20+ flows	Balanced (4–6 hrs / mo)	Predictable via CI/CD tests
Audit & Compliance Readiness	Limited	Strong	Enterprise-Grade

ARCHITECT'S RULE OF THUMB

Never argue "n8n vs code" as a religious debate. Prototype the workflow in **n8n** during Weeks 1–2 to prove business value with real users. Keep the AI prompt, RBAC retrieval, and PII redaction behind a thin **Laravel or FastAPI endpoint**—that way you can replace or scale the trigger layer anytime without touching your core intelligence.

CHAPTER 03 · OPERATIONAL RELIABILITY

03 · CRM, ERP, Email & WhatsApp Integration Patterns

Connecting AI to live business channels without duplicate messages, timeout crashes, or unapproved customer replies: webhooks, queues, idempotency, and Human-in-the-Loop (HITL) gates.

Calling an LLM synchronously inside a webhook handler is the #1 cause of production incidents. WhatsApp Cloud API, Stripe, and CRM webhooks expect an HTTP `200/202` response within 2–5 seconds; if your LLM call takes 9 seconds, the provider retries the webhook and your bot sends the customer three identical replies. Every production integration must decouple **ingestion** from **AI execution**.

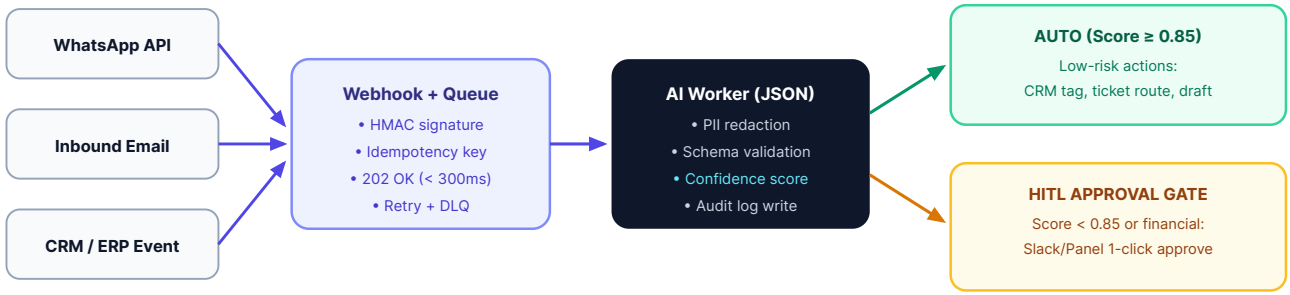


Figure 3.1: Event-driven integration architecture with HMAC verification, idempotency, queues, and HITL gates. Ref: Reliability Blueprint

3.1 Four High-ROI Operational Integration Patterns

Workflow	Trigger & Input	AI Structured Action	Safety Gate
Lead Qualification	Web form / WhatsApp inquiry	Extract budget, urgency, industry → score 1–100 + CRM tags	Auto-tag CRM; sales rep approves quote
Support Triage	Inbound Zendesk / Freshdesk email	Classify category, sentiment, SLA priority + draft RAG reply	Agent clicks "Send" on draft (HITL)
Invoice / PO Extract	PDF attachment to `ap@company`	Extract vendor, tax ID, line items, totals → strict JSON	Cross-check math; flag diff > $0.01
Meeting to ERP/CRM	Sales call transcript	Generate action items, next-step date, deal risk summary	Rep reviews 30-sec card before CRM sync

3.2 Production Webhook Idempotency Pattern (Laravel / PHP)

app/Http/Controllers/InboundWebhookController.php

```php
public function handle(Request $request): JsonResponse
{
    // 1. Verify HMAC signature before touching DB or queues
    abort_unless($this->verifyHmac($request), 401, 'Invalid signature');

    $eventId = (string) $request->input('event_id');

    // 2. Atomic idempotency guard (prevents duplicate AI jobs on provider retry)
    $acquired = Cache::add("wh:processed:{$eventId}", true, now()->addHours(48));
    if (! $acquired) {
        return response()->json(['status' => 'duplicate_ignored'], 200);
    }

    // 3. Persist raw payload & dispatch async worker immediately (< 150ms)
    ProcessInboundAiEvent::dispatch($eventId, $request->all())->onQueue('ai-triage');

    return response()->json(['status' => 'accepted'], 202);
}
```

3.3 Confidence Thresholds & Human-in-the-Loop (HITL) Routing

Confidence Band	Low-Risk Action (Tag / Draft)	Medium-Risk (Customer Reply)	High-Risk (ERP Write / Refund)
≥ 0.85 High	Execute automatically + log	Queue draft or auto-send for FAQ	**Always require 1-click HITL**
0.60–0.84 Med	Execute + flag for daily sample	Mandatory agent review in panel	Escalate to supervisor queue
< 0.60 Low	Fallback to `needs_triage`	No AI draft; route to human	Block write + log incident

FIELD CASE STUDY #2 — B2B Logistics & Customs Brokerage · 3,200 Tickets/Mo

Inbound Email & WhatsApp Triage with HITL Draft Approval

Challenge: Operations coordinators spent 3.5 hours daily reading shipment status emails, extracting container numbers, checking ERP status, and typing repetitive replies.
Solution: Deployed an async Laravel queue worker that verifies webhooks, extracts `container_no` via strict JSON schema, queries ERP read-only API, and posts a ready-to-send draft into the helpdesk. Coordinators approve 84% of drafts with zero edits.

14 min → 2.1 min	0 Duplicates	112 hrs / mo
First-response time	Across 18k+ webhook events	Verified ops time saved

CHAPTER 04 · TRUST, SECURITY & GOVERNANCE

04 · Data Hygiene, Access Control (RBAC) & Hallucination Prevention

Why prompt instructions alone never enforce security, how to filter permissions inside the vector database, and how a 40-case Golden Q&A suite catches drift before users do.

A single confidential leak—such as an intern asking the company assistant about executive compensation and receiving a paragraph from an HR PDF—can shut down an enterprise AI program overnight. Security and accuracy must be enforced in **four independent layers** so no single failure compromises trust.

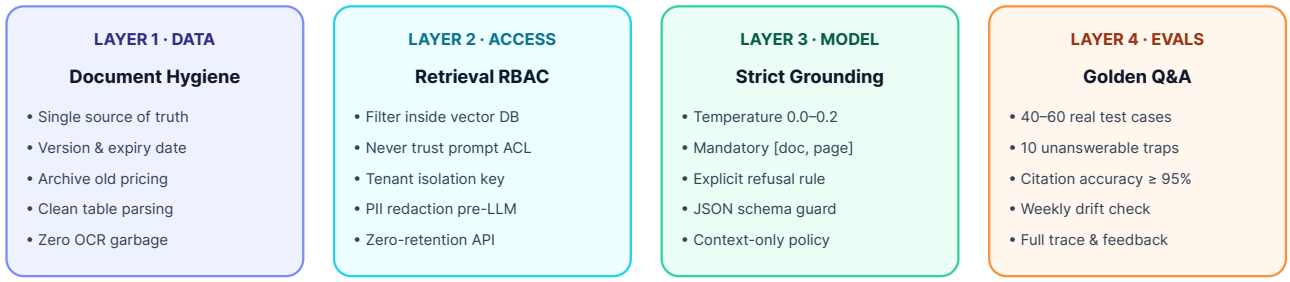


Figure 4.1: Four-layer defense-in-depth for Enterprise RAG (Data, Retrieval RBAC, Grounding, Evals). Ref: Security Architecture

4.1 Retrieval-Layer RBAC in PostgreSQL (`pgvector`)

Never ask the LLM *"Only answer if the user is in Finance."* Prompt-level access control is trivially bypassed. Instead, bind the authenticated user's roles and tenant ID directly into the SQL `WHERE` clause before similarity ranking occurs:

sql/hybrid_rbac_search.sql

```sql
SELECT c.id, c.doc_title, c.page_no, c.content,
       1 - (c.embedding <=> $1::vector) AS vec_score,
       ts_rank_cd(c.tsv, plainto_tsquery('english', $2)) AS bm25_score
FROM document_chunks c
WHERE c.tenant_id = $3
  AND c.status = 'active'
  AND (c.expires_at IS NULL OR c.expires_at > NOW())
  AND c.allowed_roles && $4::text[]   -- Hard RBAC intersection (e.g. ARRAY['sales','all'])
ORDER BY (0.65 * (1 - (c.embedding <=> $1::vector)) + 0.35 * ts_rank_cd(c.tsv, plainto_tsquery('english',
$2))) DESC
LIMIT 20;
```

4.2 Pre-LLM PII Sanitizer (Python)

Before sending support tickets or customer emails to an external LLM API (even under a zero-data-retention enterprise DPA), strip direct personal identifiers into reversible placeholders:

services/pii_sanitizer.py

```python
import re

PII_RULES = [
    ("EMAIL", re.compile(r"[A-Za-z0-9._%+-]+@[A-Za-z0-9·-]+\.[A-Za-z](2,)")),
    ("PHONE", re.compile(r"\+?\d[\d\s\-()](8, 16)\d")),
    ("IBAN",  re.compile(r"\b[A-Z]2\d2[A-Z0-9](11, 30)\b")),
    ("CARD",  re.compile(r"\b(?:\d[ -]*?)(13, 16)\b")),
]

def mask_pii(text: str) -> str:
    for label, pattern in PII_RULES:
        text = pattern.sub(f"[REDACTED_{label}]", text)
    return text
```

4.3 Building Your Golden Q&A Evaluation Suite

Never judge a RAG system by asking 3 ad-hoc questions in a demo. Assemble a version-controlled JSONL file of **40–60 real business questions**—including at least **10 unanswerable or out-of-scope traps**—and run it on every prompt, chunking, or model change:

Metric	Target Gate	How to Measure Automatically
Retrieval Recall@5	≥ 90%	Does the expected `doc_id + page_no` appear in the top 5 reranked chunks?
Citation Grounding	≥ 95%	Every factual claim in the response maps to a quoted passage in retrieved context.
Refusal Precision	100%	On the 10 unanswerable trap questions, the model refuses instead of guessing.
RBAC Leak Rate	0.0%	Zero restricted chunks returned when queried with a lower-privilege role token.

CHAPTER 05 · EXECUTION & ROI GOVERNANCE

05 · 60-Day Pilot Automation & RAG Roadmap

An 8-week execution plan designed to deliver one measurable production win before expanding budget, complete with a formal Week-6 Go/No-Go gate and ROI formula.

Enterprise AI pilots fail when they try to automate five departments simultaneously. Restrict your first 60 days to **one high-frequency workflow** with clear document ownership, a willing team lead, and a measurable baseline (minutes per ticket, lookup speed, or error rate).

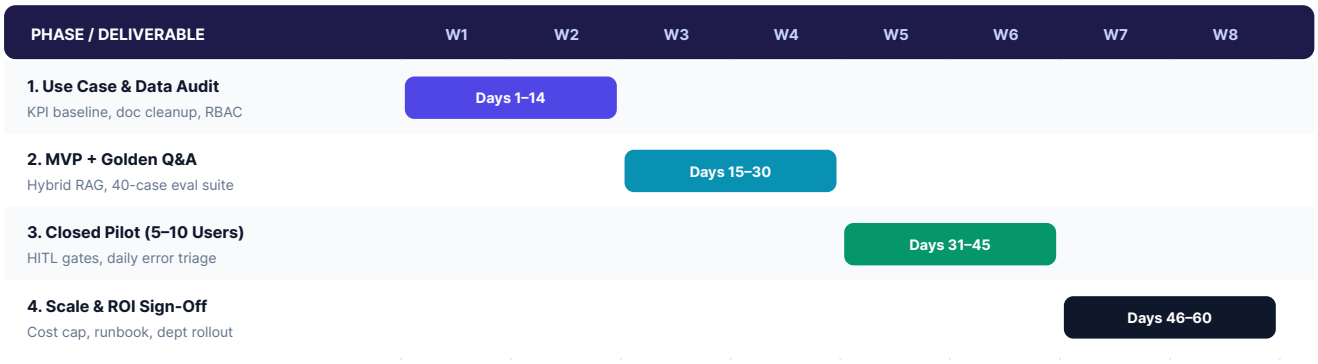


Figure 5.1: 8-week phased rollout from data hygiene to closed pilot and executive ROI sign-off. Ref: Delivery Plan

5.1 Phase-by-Phase Deliverables

Phase 1 · Weeks 1–2 (Audit & Scope)

- Select 1 workflow with ≥ 200 monthly repetitions.
- Record 2-week baseline metrics (time, error rate).
- Curate 50–150 authoritative docs; archive obsolete versions.
- Define RBAC matrix and sign zero-retention API DPA.

Phase 2 · Weeks 3–4 (MVP & Golden Q&A)

- Build hybrid ingestion pipeline + metadata schema.
- Assemble 40–60 Golden Q&A test cases with domain leads.
- Tune chunk size, reranker threshold, and refusal prompt.
- Pass ≥ 90% Recall@5 and 0% RBAC leak test.

Phase 3 · Weeks 5–6 (Closed Pilot)

- Onboard 5–10 power users with thumbs-up/down logging.
- Enforce HITL approval on all external or ERP actions.
- Run daily 15-minute error triage to fix missing docs.
- Execute the Week-6 Go/No-Go gate review.

Phase 4 · Weeks 7–8 (Scale & ROI)

- Set hard monthly API spend alerts and rate limits.
- Publish 1-page operations runbook & owner rotation.
- Present before/after ROI deck to leadership.
- Approve Department #2 using the proven blueprint.

5.2 Week-6 Go / No-Go Production Gate

Gate Criterion	Minimum Pass Bar	Action If Below Bar
Weekly Active Pilot Usage	≥ 70% of pilot users	Interview non-users; fix UX friction (move into Slack/CRM).
Zero-Edit Acceptance Rate	≥ 75% on drafted outputs	Tighten prompt constraints and clean contradictory SOPs.
Latency (P95)	< 4.5 seconds	Enable prompt caching, stream tokens, or route to faster model.
Unit Economics	ROI ≥ 3x monthly API + ops cost	Switch classification steps to mini models; cache recurring queries.

SIMPLE EXECUTIVE ROI FORMULA

`Monthly Net Value = (Hours Saved × Loaded Hourly Rate) + (Error Reduction Value) − (API + Infra + Maintenance Cost)`

Example: **90 hours/mo saved × $45/hr = $4,050/mo** against **$280/mo** API & hosting yields **$3,770/mo net value**—paying back a 60-day implementation in under 90 days.

5.3 Pilot Ownership Matrix & Risk Mitigation

Role	Core Responsibility During 60-Day Pilot	Primary Risk & Built-In Mitigation
Executive Sponsor	Approves scope, removes cross-team blockers, signs Week-6 Go/No-Go gate.	Scope creep → Lock pilot to 1 workflow until Day 60.
Domain / Ops Lead	Curates 50–150 source docs, writes 40 Golden Q&A cases, leads pilot users.	Stale SOPs → Enforce monthly doc owner & expiry date.
Lead Engineer	Builds hybrid RAG, RBAC filter, webhook queue, PII redaction, and logging.	Runaway API spend → Hard token/dollar circuit breaker.
Security / Legal	Reviews zero-retention DPA, validates RBAC leak test (0% leak bar).	Shadow AI leakage → Provide sanctioned SSO portal.

CHAPTER 06 · READY-TO-USE ENGINEERING KIT

06 · Practical Appendices: Blueprints, Checklist, Scorecard & Glossary

Copy-and-adapt production templates: grounded RAG system prompt, JSON classifier schema, audit log payload, 10-item launch checklist, and your 0–20 readiness scorecard.

6.1 Grounded RAG System Prompt Blueprint

prompts/rag_grounded_system.txt

```
You are an internal enterprise assistant for {company_name}.
RULES:
1. Answer ONLY using the provided <context> blocks. Never rely on outside memory.
2. Cite every factual statement inline using [{doc_title}, p.{page_no}].
3. If the answer is missing or ambiguous in <context>, reply verbatim:
   "This information is not available in your authorized company documents. Please contact {owner_team}."
4. Keep answers concise, structured with bullet points, and temperature ≤ 0.2.
```

6.2 Structured Support Triage & Audit Log JSON Schemas

schemas/triage_and_audit.json

```json
{
  "trace_id": "trc_20261124_89a1",
  "channel": "email",
  "category": "billing_dispute",
  "priority": "high",
  "confidence": 0.89,
  "requires_hitl": true,
  "citations": ["Billing-SOP-v3.pdf#p4"],
  "pii_redacted": true,
  "model": "gpt-4o-mini",
  "tokens": { "in": 1420, "out": 185 },
  "latency_ms": 1180
}
```

6.3 10-Item Production Pre-Launch Checklist

Security & Data Integrity

- ☐ Zero-data-retention enterprise API DPA signed
- ☐ RBAC filter verified inside SQL/vector query
- ☐ PII redaction active before external LLM calls
- ☐ Obsolete/draft PDFs marked `archived`
- ☐ HMAC verification + idempotency key on webhooks

Quality, Cost & Operations

- ☐ 40+ Golden Q&A suite passes ≥ 95% citation bar
- ☐ 10/10 unanswerable traps trigger clean refusal
- ☐ HITL gate enforced on all financial/external writes
- ☐ Monthly spend alert + circuit breaker configured
- ☐ Full audit trace logged (prompt, chunks, user, latency)

6.4 Enterprise AI Automation & RAG Readiness Scorecard (0–20 Points)

Score each dimension from **0 (Not Started)** to **4 (Production-Ready)** with your IT and operations leads before kicking off Week 1:

Dimension	What "4 Points (Ready)" Looks Like	Score (0–4)	Owner
1. Document Hygiene	Single source of truth, versioned PDFs, obsolete files archived.	_____ / 4	Ops / KM
2. Access & Privacy	Clear role matrix (RBAC), PII rules, approved enterprise API account.	_____ / 4	IT / Legal
3. Workflow Baseline	1 target process chosen with measured current time & volume KPI.	_____ / 4	Dept Lead
4. Engineering Readiness	Webhook/queue capability, Git CI/CD, or hybrid n8n + API setup.	_____ / 4	Tech Lead
5. Pilot Governance	5–10 pilot users assigned, HITL reviewers named, Week-6 gate scheduled.	_____ / 4	Sponsor

0–8 Points · Foundation First
Clean your top 50 documents, establish RBAC rules, and start with Book 1 team prompting habits before automation.

9–14 Points · Pilot Ready
Launch a scoped 60-day hybrid RAG or triage pilot with strict HITL gates and weekly Golden Q&A reviews.

15–20 Points · Scale Ready
Build reusable internal RAG/integration services and graduate toward production agents & MCP (Book 3).

6.5 Plain-English RAG & Automation Glossary

Term	Practical Definition
BM25	Classic keyword-ranking algorithm essential for exact SKUs, invoice numbers, and legal codes.
Chunking	Splitting long documents into 400–800 token segments with overlap and heading context for retrieval.
Embedding	Numeric vector capturing semantic meaning so paraphrases match even without shared keywords.
Golden Q&A	Versioned benchmark of 40–60 real questions + expected citations used to test RAG accuracy.
HITL	Human-in-the-Loop: requiring a 1-click human approval before high-risk or low-confidence actions execute.
Idempotency	Guaranteeing that receiving the same webhook event twice never creates duplicate replies or ERP records.
Reranker	A cross-encoder model that re-scores top-20 retrieved chunks to select the most relevant 4–6 passages.
RBAC	Role-Based Access Control enforced inside the database query so users only retrieve authorized documents.

ABOUT THE AUTHOR



Ferhat Balaban

Software Architect, Corporate AI & Software Trainer with 20+ years of production engineering experience across enterprise web systems, mobile apps, RAG pipelines, and AI automation. Helps engineering and operations teams turn AI pilots into auditable, ROI-positive systems.

CORPORATE TRAINING & ARCHITECTURE WORKSHOPS

Want to build your company's first RAG & AI Automation pilot with hands-on guidance?

Custom on-site and remote workshops for engineering, IT, and operations teams—tailored to your real documents, CRM/ERP stack, and security requirements.

Web: ferhatbalaban.com/en/training · Resources: ferhatbalaban.com/en/resources

Scan for Training Programs
ferhatbalaban.com/en/training

COMPLETE 3-BOOK SERIES · FREE DIRECT PDFS

BOOK 1 · BEGINNER

Introduction to AI for Businesses

Prompting framework, 5 departments, privacy rules & 30-day plan.

BOOK 2 · YOU ARE HERE

Enterprise AI Automation & RAG

Hybrid RAG, n8n vs code, CRM/ERP/WhatsApp, RBAC & 60-day pilot.

BOOK 3 · ADVANCED

Production AI Agents & MCP

Agent patterns, MCP servers, multi-model routing, evals & LLMOps.

© 2026 Ferhat Balaban · All rights reserved. Free to share internally with attribution.