

Muhammad Ali Qasmi

Senior AI Engineer • Senior Engineering Manager and Senior Product Engineer by scope of work; employer titles are under Experience

U.S. Permanent Resident • San Francisco, CA • [linkedin.com/in/qasmi](https://www.linkedin.com/in/qasmi) • muhammadaliqasmi+work@gmail.com • github.com/muhammadaliqasmi

SUMMARY

Engineering Manager and hands-on Applied AI engineer with 13+ years of experience building and scaling large-scale distributed systems, machine learning platforms, and production-grade agentic AI, including seven years leading engineering teams, three as Technical Team Lead and four as Engineering Manager, covering hiring, 1:1s, performance appraisals, promotions, and ownership of the team roadmap and quarterly OKRs for a revenue-critical platform.

Currently designing and delivering an AI agent platform on Azure AI Foundry: multi-agent workflows, four production MCP servers, a RAG platform, an LLM evaluation and safety harness, near real-time telemetry pipelines, and LLM-driven optimisation and reporting agents with full observability and audit trails. Proven record of measurable business impact, including ~\$6M in avoided operational labour cost, ~\$2M in annual operational savings, and 99.99% production SLA uptime. AI-native engineer using Claude Code, Claude Cowork, and Azure AI Foundry daily.

EXPERIENCE

Senior AI Engineer

Apr 2026–present

Qcells North America

Architecting and delivering an AI agent platform on Azure AI Foundry for autonomous infrastructure operations, and driving adoption of AI-native engineering practices through an agentic engineering framework adopted organisation-wide.

- Established an agentic engineering practice as a Claude Code plugin framework (role-specialised subagents, workflow skills, AI review gates) and a nine-skill automation library, adopted across polyglot teams.
- Designed multi-agent workflows of single-responsibility agents plus a three-mode trigger framework (schedule, KPI event, manual).
- Built and own four production MCP servers that expose platform capabilities to the agent runtime as callable tools.
- Delivered a RAG platform that converts technical documentation into machine-readable operating procedures, with an LLM evaluation and safety harness and human-in-the-loop governance.
- Built near real-time telemetry pipelines into the agent decision layer, and optimisation and reporting agents with full auditability.
- Shipped with production rigour: cloud-native agent runtime, CI/CD, Terraform, OpenTelemetry, audit trails.

AI Forward Deployed Engineer

Apr 2024–Mar 2026

Walmart Global Tech

Owned end-to-end delivery of large-scale cross-border commerce capabilities on a seller platform — mission-critical supply chain systems used by third-party sellers.

- Championed AI-native engineering by integrating Claude Code and Claude Cowork into daily delivery at enterprise scale.
- Partnered with Senior Manager, Director, and Senior Director stakeholders on technical strategy, architecture reviews, and delivery commitments.

- Led an international import port expansion from one port to six.
- Delivered outbound logistics support, dynamic document generation, new REST APIs, and Kafka-based shipment event processing with real-time pricing.
- Maintained 99.99% SLA uptime across seven production microservices with Grafana observability; stress-tested with Apache JMeter before deployment.
- Executed a JDK 17 to JDK 21 modernisation across services on Azure SQL and Azure-hosted Kubernetes with zero production downtime.

Senior ML Engineer

Jul 2022–Mar 2024

AT&T

Led technical delivery for network operations machine learning initiatives across a 21-month program supporting 70,000+ cellular sites nationwide.

- Owned platform architecture and end-to-end delivery for two workstreams, setting technical direction and carrying the delivery commitment for both.
- Proposed and built an MLflow-based MLOps platform for model training, versioning, deployment, and monitoring — presented to VP-level leadership as a shared enterprise capability.
- Designed and delivered an ML anomaly detection platform for remote identification of misconfigured cellular sites — forecast to cut field visits 18–20% annually.
- Built a geospatial disaster-preparedness platform using Elasticsearch geospatial indexing and weather forecast data to recommend portable site deployments ahead of storms and floods.
- Developed unsupervised-clustering pipelines over a decade of tower telemetry, wrapped as REST APIs for real-time site health prediction. Solutions projected at ~\$2M in annual operational savings.

Engineering Manager (HR title: Principal Software Engineer)

Jul 2019–Jun 2022

GfK (An NIQ Company)

Engineering Manager for a product-data platform integration and delivery program — people manager of a team of five frontend and backend engineers, owner of the team roadmap and quarterly OKRs, and holder of technical oversight of a 23-person program spanning engineering, taxonomy, and QA.

- Managed a team of five end to end: hiring, performance appraisals, org management, and promotion decisions from Junior to Senior and Staff levels.
- Held weekly 1:1s and mentored engineers across React, TypeScript, Java, and Spring Boot.
- Owned the team roadmap and defined quarterly OKRs aligned to platform revenue and quality goals.
- Shipped 20+ customer-facing API and Excel integrations and 10+ third-party marketplace integrations.
- Established RFC-based design processes and code review standards.
- Platform capabilities delivered under this program coincided with the platform reaching a \$28.1M revenue milestone.

Engineering Manager / Technical Team Lead (HR title: Staff Software Engineer)

Jun 2016–Jul 2019

GfK (An NIQ Company)

Technical Team Lead through June 2018, then Engineering Manager from July 2018. Led a cross-functional team of seven: three engineers, a business analyst, a UI/UX designer, and two QA.

- Scaled the data-extraction platform to eliminate the need for an estimated 400+ additional extraction staff between 2016–2022 — approximately \$6M in avoided operational labour cost.
- Enabled automated extraction of ~2.6M additional products that would have been infeasible manually.

- Proposed and built a rule-based template pre-fill engine (Apache UIMA Ruta) that automated taxonomy mapping for 25% of incoming product requests, reducing time-to-publish by ~20% for covered cases.
- Built event-driven services in Java, Spring Boot, and Apache Kafka with MySQL and MongoDB; contributed to React and TypeScript frontend delivery; handled AWS operations.
- Grew the platform to serve rich content for 20M+ products across 25,000+ eCommerce users in 30 languages and 200+ technology categories.

Technical Team Lead (HR title: Senior Software Engineer)

May 2015–Jun 2016

GfK (An NIQ Company)

Technical Team Lead of a four-engineer team on a platform modernisation, and a founding engineer of the data-extraction platform that underpinned catalogue growth and revenue for the following seven years.

- Designed and implemented a globally distributed real-time data crawling platform on 70 AWS EC2 Spot Instances, with MongoDB as the raw data store, autonomous task redistribution, and a fault-tolerant, self-healing architecture.
- Delivered a ~60% annual reduction in AWS compute run-rate by replacing on-demand instances with a dynamically scaled spot fleet of 10–70 nodes tied to business seasonality, with a stateful coordinator that re-queues work from terminated instances.

Earlier experience

Software Engineer — GfK (An NIQ Company)

Sep 2012–Mar 2013

Full-stack features for an internal operations management and workflow automation system (Java and Spring backend, AngularJS frontend).

Software Engineer — Interactive Group

Dec 2011–Sep 2012

Led R&D of an NLP-powered document search and management system (Java and Spring, Apache Solr indexing, OpenNLP pipelines).

Software Engineer — Folio3

Jun 2011–Dec 2011

Analysis, design, and implementation of a resource allocation system for a 300-employee organisation (ASP.NET MVC, Microsoft SQL Server).

EDUCATION

M.Sc. Computer Science — University of Bonn, Germany

2013–2015

B.Sc. Computer Science — FAST-NUCES (National University of Computer & Emerging Sciences), Pakistan

2007–2011

PUBLICATIONS

Data-Intensive Cloud Computing: Requirements, Expectations, Challenges, and Solutions

J. Shamsi, M. A. Khojaye, Muhammad Ali Qasmi. Journal of Grid Computing, Springer, April 2013.

DOI 10.1007/s10723-013-9255-6

71 citations, 2,072+ accesses (as of September 2026).

RDF Quality Extension for OpenRefine

Muhammad Ali Qasmi. M.Sc. thesis, University of Bonn, December 2014.

TECHNICAL RANGE

AI & Agentic Systems: agentic AI · multi-agent systems · agent orchestration · MCP servers · agent trigger frameworks · Azure AI Foundry · RAG and agentic RAG · context engineering · LLM evaluation · LLM-as-judge · guardrails · human-in-the-loop review · Claude Code plugins / subagents / skills · harness engineering

ML & MLOps: MLflow · scikit-learn · anomaly detection · model versioning and monitoring · ML pipelines

Cloud & Infrastructure: Azure · AWS · GCP · Kubernetes · Docker · Terraform · CI/CD · OpenTelemetry · Grafana · SLOs/SLIs

Data & Streaming: Apache Kafka · Elasticsearch · Redis · MongoDB · MySQL · Postgres · real-time streaming · geospatial indexing

Backend & Architecture: microservices · event-driven architecture · distributed systems · Spring Boot · FastAPI · REST APIs · system design

Languages: Python · Java · TypeScript · SQL · Bash

Leadership: engineering management · hiring · performance management · roadmap planning · OKRs · RFC process · ADRs · code review standards

CERTIFICATIONS

- Claude Code in Action — Anthropic, 2026
- Introduction to Claude Cowork — Anthropic, 2026
- Coursera course certificates (Google Cloud curriculum) — 2020, seven courses
 - Preparing for the Google Cloud Professional Cloud Architect Exam
 - Architecting with Google Kubernetes Engine: Foundations
 - Reliable Google Cloud Infrastructure: Design and Process
 - Elastic Google Cloud Infrastructure: Scaling and Automation
 - Essential Google Cloud Infrastructure: Foundation
 - Essential Google Cloud Infrastructure: Core Services
 - Google Cloud Platform Fundamentals: Core Infrastructure

WHAT I AUTOMATE

Twenty categories of engagement and 67 build blueprints I have designed: the problem, what I build, the stack, the gate, and what done looks like.

Agents and conversation

AI agents and assistants

Operations copilot with typed tools

Scheduled agent that watches and acts

Multi-agent research and drafting team

Conversational AI and chatbots

Support bot with a clean hand-off

Website assistant that qualifies and books

WhatsApp and SMS service bot

Voice AI

Inbound phone agent for appointments

After-call summariser that writes the CRM

Outbound reminders and confirmations

Internal knowledge systems

Ask-the-docs with citations and permissions

Slack answer agent that learns from resolved threads

Procedure generator from technical documentation

Business operations

Workflow automation

Approval workflows with an agent in the loop

Shared-inbox triage and drafting

Meeting-to-action pipeline

Finance and operations automation

Invoice capture with confidence routing

Bank reconciliation assistant

Month-end close checklist agent

Expense policy checker

Recruitment and HR automation

Structured screening against a rubric

Interview scheduling and prep packs

Onboarding concierge

Custom integrations and internal tools

MCP server generated from an OpenAPI contract

Internal admin tool with agent assist

Legacy system bridge

Revenue and growth

Sales and outreach automation

Inbound lead qualification and routing
Research-backed outbound drafts
Call notes to CRM hygiene

E-commerce automation

Catalogue enrichment from supplier data
Order exception desk
Returns and refunds agent under policy
Review and question responder

Content and marketing pipelines

Brand-voice content pipeline with review gates
Repurposing engine
Search content refresh agent

Data enrichment and research automation

Company and contact enrichment with provenance
Competitive monitoring digest
Tender, grant and RFP scanner
Literature and market scans

Data and oversight

Document and data processing

Schema-guided extraction service
Form and contract intake to systems of record
Long-document summaries with page citations

Data pipeline and ETL automation

Agent-assisted ingestion for new sources
Schema-drift sentinel
Data quality triage

AI analytics and reporting

Governed natural-language analytics
Automated weekly reporting agent
Anomaly to explanation

Compliance and monitoring automation

AI agent audit trail and evidence pack
Policy checks on communications and content
Control evidence collection
AI system register with risk tiers

Sector workflows

Legal tech automation

Contract review against a playbook
Obligation tracker
Client intake and conflict check assistant

Real estate automation

Lead qualification that books viewings
Listing content with compliance checks
Maintenance request triage for property managers
Lease abstraction

Education and training automation

Course builder from source material
Rubric-based grading assistant
Tutor grounded in the course
Training compliance tracker

Healthcare workflow automation

Patient intake and triage summaries
Prior authorisation packet assembly
Referral and fax processing
Clinic phone agent for scheduling

These are my designs, not client records; the production record is in Experience above. Every engagement starts with the process as it runs today, an agreed definition of done, and a measured result before anything is rolled out. Full blueprints are in the appendix below.

APPENDIX: BUILD BLUEPRINTS

Sixty-seven designs across the twenty categories above. Each states the problem, what I build, the stack, the evaluation gate and what done looks like. These are my designs, not client records.

AI agents and assistants

Agents and conversation

Operations copilot with typed tools

Problem

Teams want an assistant that acts inside their systems, not one that only talks about them.

Build

- 01 An MCP server over the internal APIs: every tool has a typed input and output schema and a permission tier (read, propose, execute).
- 02 A Claude Agent SDK loop with the tool set, a system prompt that states the authority model, and a session store.
- 03 A proposal queue: execute-tier actions wait for the named approver; read and propose run immediately.
- 04 An audit log per tool call (who asked, what ran, what changed) and a Slack front end.

Stack

Claude Agent SDK, MCP, Postgres, Slack Bolt, OpenTelemetry

Gate

Fifty gold tasks scored by an LLM-as-judge for correctness and for refusing out-of-scope asks; the execute tier switches on only after the pass.

Done

The assistant completes the team's top ten requests end to end, and every action is attributable.

Scheduled agent that watches and acts

Problem

Someone checks the same dashboards, queues or inventories every morning and raises the same tickets by hand.

Build

- 01 A three-mode trigger framework: schedule, KPI threshold, manual.
- 02 Single-responsibility agents, each with one question to answer and the tools it needs.
- 03 Proposals land in a review queue; a small class of low-risk actions is auto-approved under limits.
- 04 Durable execution so a long wait for approval survives restarts.

Stack

Temporal (or cron plus Postgres), Claude, MCP tools, Grafana

Gate

Thirty historical days through the pipeline: the agent must raise every real incident and no more than one false alarm per week.

Done

The morning check runs itself, and the owner reviews a short list instead of a dashboard.

Multi-agent research and drafting team

Problem

Reports take days because research, writing and checking are done by one person in sequence.

Build

- 01 Planner, researcher, writer and critic agents with a shared memory and a token budget each.
- 02 Sources are fetched through tools and stored with URLs; the writer may only cite stored sources.
- 03 The critic scores against a written bar and returns the draft until it passes.
- 04 Output is a document with inline citations and an appendix of what was searched.

Stack

Claude Agent SDK, web and document tools, pgvector, Markdown to Docx

Gate

Ten past reports regenerated and judged against the originals for coverage, accuracy and citation validity.

Done

A first draft with sources in under an hour, reviewed by the analyst before it leaves the team.

Conversational AI and chatbots

Agents and conversation

Support bot with a clean hand-off

Problem

Customers ask the same twenty questions, and the hard cases still need a person with full context.

Build

- 01 Grounded answers from the help centre and the order API, with a citation on every answer.
- 02 A refusal policy for out-of-scope topics and a confidence threshold for escalation.
- 03 Hand-off creates a ticket with a transcript summary, the customer's intent and what was already tried.
- 04 A review loop that turns corrected answers into new help-centre entries.

Stack

Claude, pgvector, Intercom or Zendesk APIs, Postgres

Gate

A gold set of three hundred real conversations: resolution accuracy, false-resolution rate and escalation precision, scored before launch and weekly after.

Done

The routine questions resolve themselves, and agents open escalations that already contain the context.

Website assistant that qualifies and books

Problem

Visitors leave with questions a sales person could have answered, and forms collect too little to act on.

Build

- 01 Answers from the site's own content, with a strict no-invention rule and a short tone guide.
- 02 A qualification flow (need, timeline, size) that stays conversational.
- 03 A booking tool on the calendar and a CRM write with the full transcript.
- 04 An opt-in for follow-up, logged with consent.

Stack

Claude, Next.js widget, Calendly API, HubSpot API

Gate

Fifty scripted visitor journeys judged for accuracy and for never promising what the site does not say.

Done

Qualified conversations become meetings on the calendar with notes already in the CRM.

WhatsApp and SMS service bot

Problem

Customers message on their phones, and replies depend on whoever is on shift.

Build

- 01 Twilio messaging in, with language detection and templated replies for regulated message types.
- 02 Order and appointment lookups through typed tools; no free-form access to customer data.
- 03 Quiet-hours and opt-out handling built in.
- 04 A hand-off to a human inbox with the thread attached.

Stack

Twilio, Claude, MCP tools, Postgres

Gate

Every supported intent has test conversations in each language; the bot must pass them all before a number goes live.

Done

Status questions answer themselves at any hour, in the customer's language.

Voice AI

Agents and conversation

Inbound phone agent for appointments

Problem

The phone rings more than the front desk can answer, and most calls are bookings and reschedules.

Build

- 01 A real-time pipeline: telephony in, streaming speech-to-text, the model, streaming text-to-speech, with barge-in.
- 02 A calendar tool with availability rules, and a transfer-to-human path with a spoken summary.
- 03 A latency budget per stage, measured on every call.
- 04 Call recording and transcripts stored with consent.

Stack

Twilio or Telnyx, LiveKit Agents or Pipecat, Deepgram, ElevenLabs or Cartesia, Claude

Gate

A scripted call set covering accents, interruptions and edge cases; booking accuracy and a response-time ceiling are both gates.

Done

Routine calls are booked without a person, and the rest reach one with the context already spoken.

After-call summariser that writes the CRM

Problem

Calls end and the notes never make it into the system.

Build

- 01 Recordings to transcripts, then to a structured note: outcome, commitments, next step, risks.
- 02 A CRM write that updates fields and creates the follow-up task.
- 03 A quality score per call against a coaching rubric, visible to the rep first.

Stack

Deepgram or Whisper, Claude with structured outputs, HubSpot or Salesforce API

Gate

One hundred calls with human-written notes as the reference; field accuracy and commitment recall are scored.

Done

Every call leaves a complete record and a task, with no typing.

Outbound reminders and confirmations

Problem

No-shows and missed renewals cost more than a reminder call would.

Build

- 01 Consented call lists with time windows and retry rules.
- 02 Scripted flows with both spoken and keypad responses, and a reschedule tool.
- 03 Opt-out and do-not-call handling, logged.
- 04 Results written back to the source system.

Stack

Twilio Voice, Pipecat, Cartesia, Claude, Postgres

Gate

A dry run against a test cohort; confirmation accuracy and complaint rate are reviewed before scaling.

Done

Confirmations arrive on time and the schedule holds.

Internal knowledge systems

Agents and conversation

Ask-the-docs with citations and permissions

Problem

The answer exists in Confluence, Notion or Drive, and nobody can find it.

Build

- 01 Connectors that respect source permissions, so a user only gets answers from what they can read.
- 02 Hybrid retrieval (keyword plus embeddings) with reranking, and a citation on every claim.
- 03 An 'I do not know' path when retrieval is weak, instead of a confident guess.
- 04 A feedback loop that turns bad answers into retrieval fixes.

Stack

Claude, pgvector, BM25, source APIs, Slack or web front end

Gate

A question set with known answers: retrieval recall, answer precision and citation validity, wired into CI.

Done

People ask once, get a sourced answer, and stop pinging the one person who knows.

Slack answer agent that learns from resolved threads

Problem

The same questions are answered in Slack every week and the answers scroll away.

Build

- 01 A miner that finds resolved threads and drafts FAQ entries, approved by the channel owner.
- 02 An in-channel agent that answers from the approved set with links to the original thread.
- 03 A weekly digest of what was asked and what has no answer yet.

Stack

Slack Bolt, Claude, Postgres, pgvector

Gate

Answer precision on a held-out month of threads; nothing unapproved can be cited.

Done

Institutional answers accumulate, and new joiners find them.

Procedure generator from technical documentation

Problem

Manuals describe systems; operators need step-by-step procedures that match them.

Build

- 01 Ingestion of manuals and specs into a versioned store.
- 02 Generation of procedures against a checklist schema: preconditions, steps, checks, rollback.
- 03 A validation gate that checks every step against the source, and the operations lead signs off before publish.
- 04 Versioning so a changed manual re-opens the procedures that depend on it.

Stack

Claude, pgvector, structured outputs, Git-backed store

Gate

A set of procedures reviewed by a domain expert; the gate must catch every seeded error.

Done

Operators follow procedures that are traceable to the documentation and reviewed by a person.

Workflow automation

Business operations

Approval workflows with an agent in the loop

Problem

Requests arrive by email and chat, get lost, and approvals take weeks.

Build

- 01 An intake form with an agent that completes missing fields and checks policy before routing.
- 02 Durable approval steps with reminders, timeouts and a named escalation owner.
- 03 A full audit trail: who asked, what the agent checked, who approved.
- 04 Reporting on cycle time by step.

Stack

Temporal, Claude, Postgres, Slack and email

Gate

Last quarter's requests rerun through the intake: routing accuracy and policy-check precision scored against the actual outcomes.

Done

Requests move on their own, and nothing waits in silence.

Shared-inbox triage and drafting

Problem

A shared inbox is read by three people and answered by none.

Build

- 01 Classification into intents with confidence, and extraction of the facts each intent needs.
- 02 Routing to the right queue, with a drafted reply for the common cases.
- 03 Low-confidence items go to the queue owner with the draft attached.
- 04 Metrics on time to first response by intent.

Stack

Gmail or Microsoft Graph API, Claude with structured outputs, Postgres

Gate

A labelled month of mail: classification accuracy, extraction accuracy and draft acceptance rate.

Done

Every message is routed within minutes and the easy ones are answered.

Meeting-to-action pipeline

Problem

Decisions are made in meetings and lost by the next one.

Build

- 01 Transcripts into decisions, actions, owners and dates, with the quote that supports each.
- 02 Owner confirmation in chat before anything is created.
- 03 Tickets created in the tracker with a link back to the moment in the recording.

Stack

Fathom or Granola or Zoom transcripts, Claude, Jira or Linear API, Slack

Gate

Twenty meetings with human minutes as reference: action recall and owner accuracy.

Done

Every meeting ends with tracked actions and no minute-taker.

Finance and operations automation

Business operations

Invoice capture with confidence routing

Problem

Invoices arrive in every layout and someone keys them in.

Build

- 01 Extraction to a fixed schema (vendor, lines, tax, totals) with a confidence per field.
- 02 Three-way matching against purchase orders and receipts.
- 03 Exceptions (new vendor, mismatch, duplicate risk, low confidence) go to a review queue; the rest post automatically.
- 04 Posting to the ledger with an audit trail.

Stack

Claude with structured outputs, QuickBooks or NetSuite API, Postgres

Gate

Five hundred historical invoices: field accuracy and the share routed to review are both tracked and gated.

Done

Most invoices post without a person, and the exceptions arrive with the reason attached.

Bank reconciliation assistant

Problem

Unmatched transactions pile up until month end.

Build

- 01 Deterministic matching first; the model handles partial amounts, timing differences and renamed payees.
- 02 Proposed journal entries with an explanation and the evidence.
- 03 The controller approves in one screen; approved patterns become rules.

Stack

Bank feeds, Claude, Postgres, ledger API

Gate

Against the accountant's final books for six months: match rate and false-match rate, both gated.

Done

The reconciliation is a short review rather than a day of work.

Month-end close checklist agent

Problem

Close runs on a spreadsheet, and the variance explanations are written at midnight.

Build

- 01 A task list per entity with owners, dependencies and evidence slots.
- 02 Agents that pull balances, compute variances and draft explanations from the underlying transactions.
- 03 Evidence attached to each task; sign-off recorded.

Stack

Claude, warehouse or ledger queries, Postgres, Slack

Gate

Variance drafts compared with the controller's final notes; precision gates the roll-out.

Done

Close status is visible at any hour, and the explanations arrive with the numbers.

Expense policy checker

Problem

Expense reports are approved without anyone checking them.

Build

- 01 Receipt extraction and matching to claims.
- 02 Policy as rules plus a model for the judgement calls, each flag with a reason.
- 03 Anomaly flags across employees and vendors.
- 04 A reviewer queue for flags only.

Stack

Claude, expense platform API, Postgres

Gate

A labelled set of past reports: flag precision and recall against known violations.

Done

Reviewers see the few reports that matter.

Recruitment and HR automation

Business operations

Structured screening against a rubric

Problem

Hundreds of applications, inconsistent first reads, and bias risk in both directions.

Build

- 01 A job-specific rubric written with the hiring manager; every score cites evidence from the application.
- 02 Identity fields removed before scoring; consistency checks across reviewers and reruns.
- 03 Scores and evidence go to the recruiter, who decides; the system never rejects.
- 04 ATS integration for intake and notes.

Stack

Greenhouse or Ashby API, Claude with structured outputs, Postgres

Gate

A past requisition with known outcomes: rubric agreement with the hiring panel, plus a parity check across groups.

Done

Recruiters start from an evidence table instead of a pile.

Interview scheduling and prep packs

Problem

Scheduling takes more emails than the interview takes minutes.

Build

- 01 Calendar coordination across panels with constraints.
- 02 A prep pack per interviewer: role, rubric, the questions they own, and the candidate summary.
- 03 Scorecard reminders and a consolidated debrief draft.

Stack

Google or Microsoft calendar APIs, ATS API, Claude

Gate

A month of real scheduling requests rerun: every constraint honoured, and prep packs rated usable by at least nine of ten interviewers.

Done

Interviews get scheduled and prepared without a coordinator in the loop.

Onboarding concierge

Problem

New joiners wait for access and ask the same questions in their first week.

Build

- 01 A day-by-day plan generated from role and team.
- 02 Access requests raised through tools, tracked to completion.
- 03 Questions answered from HR and team documentation with citations.

Stack

Claude, HRIS and IT ticketing APIs, pgvector, Slack

Gate

A dry run with the last five joiners' questions; answer accuracy and request completion time.

Done

Every new joiner has access and a plan before their first morning.

Custom integrations and internal tools

Business operations

MCP server generated from an OpenAPI contract

Problem

An internal API exists, and no agent can use it safely.

Build

- 01 Tools generated from the contract with typed inputs and outputs and the auth model.
- 02 Rate limits, read and write scopes, and redaction of sensitive fields.
- 03 A test per tool, run in CI against a sandbox.

Stack

MCP, OpenAPI, TypeScript or Python, CI

Gate

Every tool passes its contract tests; a judge checks that tool descriptions make agents pick the right one.

Done

Any agent in the company can use the API within its permissions.

Internal admin tool with agent assist

Problem

Support and ops staff run bulk changes through scripts and hope.

Build

- 01 A small admin app over the domain model.
- 02 An agent that turns a request into a bulk operation, shown as a preview and a diff.
- 03 Approval and execution with a rollback plan and an audit record.

Stack

Next.js, Postgres, Claude, MCP tools

Gate

A suite of operations with expected diffs; nothing executes without matching its preview.

Done

Bulk changes are safe, reviewable and reversible.

Legacy system bridge

Problem

Partners still send CSV, SFTP drops or EDI, and someone re-keys them.

Build

- 01 Ingestion from the legacy channel with schema mapping proposed by the model and approved once.
- 02 Validation, idempotent writes and replay for failures.
- 03 A dashboard of what arrived, what failed and why.

Stack

Python, Temporal, Postgres, target API, Claude for the first mapping

Gate

Historical files replayed against the known-good records; mapping accuracy and replay safety.

Done

The partner keeps their format and the data lands correctly anyway.

Sales and outreach automation

Revenue and growth

Inbound lead qualification and routing

Problem

Inbound leads wait, and the best ones get the same template as the rest.

Build

- 01 Enrichment from company and contact sources with provenance stored.
- 02 Scoring against the ideal customer profile with reasons.
- 03 Routing and assignment in the CRM, and a drafted first reply for the rep to send.
- 04 Suppression for existing customers and opted-out contacts.

Stack

HubSpot or Salesforce API, Clay or Apollo, Claude, Postgres

Gate

Last quarter's leads rerun: score agreement with closed outcomes, routing accuracy, reply acceptance by the reps.

Done

Every lead is enriched, scored and in the right hands within minutes.

Research-backed outbound drafts

Problem

Outbound is either generic or takes an hour per account.

Build

- 01 An account research agent with sources (site, filings, hiring, news) stored per claim.
- 02 Drafts in the team's voice, grounded only in stored facts, with the reason to reach out.
- 03 Human approval per sequence; nothing sends itself.
- 04 Compliance: consent basis, unsubscribe, regional rules.

Stack

Claude, web tools, sequencing platform API, CRM

Gate

Fifty accounts judged for factual accuracy and relevance; a draft with an unsupported claim fails.

Done

Reps send researched messages at template speed.

Call notes to CRM hygiene

Problem

The CRM is out of date because updating it is a chore.

Build

- 01 Transcripts to deal fields: stage, next step, risks, stakeholders.
- 02 Proposed updates shown to the rep for one-click acceptance.
- 03 Pipeline review digest built from the accepted updates.

Stack

Gong or Fathom API, Claude, CRM API

Gate

Field accuracy against rep-corrected records over a month.

Done

Forecast calls use data that is actually current.

E-commerce automation

Revenue and growth

Catalogue enrichment from supplier data

Problem

Supplier datasheets arrive in every format, and listings go live incomplete.

Build

- 01 Extraction of attributes to the category schema, with confidence.
- 02 Titles, descriptions and alt text in the brand's voice; taxonomy mapping with review for new categories.
- 03 A review queue for low confidence and new attributes; approved items publish.

Stack

Shopify Admin API, Claude with structured outputs, Postgres

Gate

Five hundred products with human-completed listings as reference; attribute accuracy and publish-without-edit rate.

Done

New products reach the store complete, and the team edits only exceptions.

Order exception desk

Problem

Late shipments, bad addresses and failed payments create tickets that need the same five actions.

Build

- 01 Exceptions detected from order and carrier events.
- 02 Classification and the allowed action per class (re-ship, address fix, refund, hold).
- 03 Customer notification drafted and sent after action; higher-value cases go to the operations lead.

Stack

Shopify or order system webhooks, carrier APIs, Claude, Temporal

Gate

A month of exceptions rerun: action correctness checked by the operations lead, customer-contact tone judged.

Done

Most exceptions resolve before the customer writes in.

Returns and refunds agent under policy

Problem

Return requests are judged inconsistently and slowly.

Build

- 01 Policy as code: windows, conditions, limits per customer and order value.
- 02 An agent that gathers the facts, applies the policy and explains the decision.
- 03 A refund tool with hard limits; beyond them, the support lead approves.

Stack

Shopify or returns platform API, Claude, Postgres

Gate

Historical returns replayed; decision agreement with the policy owner and zero over-limit refunds.

Done

Returns are decided in minutes, consistently, within policy.

Review and question responder

Problem

Product questions and reviews go unanswered, and defects are noticed late.

Build

- 01 Drafted answers from product data and policy, approved by the merchandiser at first, then by exception.
- 02 Defect and sizing signals extracted from reviews and sent to the product team weekly.

Stack

Marketplace and store APIs, Claude, Postgres

Gate

Answer accuracy on a labelled set; defect signals checked against known issues.

Done

Shoppers get answers, and the product team hears problems early.

Content and marketing pipelines

Revenue and growth

Brand-voice content pipeline with review gates

Problem

Content volume rises, and quality and voice drift with it.

Build

- 01 Brief to research to draft, with sources stored and cited.
- 02 Critics for voice, facts and search readiness (SEO and answer-engine structure), each with a written bar.
- 03 A CMS draft with metadata and schema markup, waiting for the editor.

Stack

Claude, web tools, CMS API, Postgres

Gate

Twenty drafts judged against the house style guide; a draft with an unsupported fact fails.

Done

Editors edit instead of write, and the voice stays the brand's.

Repurposing engine

Problem

One long piece should become ten, and it becomes none.

Build

- 01 Long-form in; social posts, newsletter sections, scripts and summaries out, each in its format rules.
- 02 A review step per channel owner.
- 03 Scheduling through the existing tools.

Stack

Claude, CMS and social APIs

Gate

Twenty source pieces, each channel's derivatives rated by its owner: a channel is automated only once four in five are accepted without edits.

Done

Every long piece ships with its derivatives.

Search content refresh agent

Problem

Pages decay quietly and nobody owns the refresh.

Build

- 01 Decay detection from search analytics.
- 02 Proposed updates as diffs, with the evidence (queries lost, competitors, stale facts).
- 03 Editor approval and publish, with before and after tracking.

Stack

Search Console API, Claude, CMS API

Gate

Proposed updates reviewed by an editor; precision of the decay signal measured over a quarter.

Done

The content library maintains itself under editorial control.

Data enrichment and research automation

Revenue and growth

Company and contact enrichment with provenance

Problem

Enriched data is bought, merged and trusted without knowing where it came from.

Build

- 01 Multi-source enrichment with a stored source and timestamp per field.
- 02 Deduplication and conflict resolution with confidence.
- 03 A refresh schedule and change alerts.

Stack

Claude, enrichment APIs, Postgres

Gate

A sample verified by hand; field accuracy by source, so weak sources are dropped.

Done

Every record can answer 'how do we know this'.

Competitive monitoring digest

Problem

Competitor changes are noticed when a customer mentions them.

Build

- 01 Watchers on pages, pricing, releases, job posts and filings.
- 02 Diffs summarised with citations; noise filtered by a relevance bar.
- 03 A weekly digest and an alert channel for big moves.

Stack

Web fetch and diff, Claude, Postgres, Slack

Gate

Three months backfilled; a product manager rates relevance to set the bar.

Done

The team sees changes the week they happen.

Tender, grant and RFP scanner

Problem

Opportunities are found late or not at all.

Build

- 01 Sources crawled on a schedule; eligibility rules applied; a summary with deadlines and requirements.
- 02 A tracker with owners and a go/no-go record.
- 03 Drafted question lists and outline for the ones pursued.

Stack

Web tools, Claude, Postgres

Gate

Past opportunities replayed: recall of the ones the team actually pursued.

Done

Nothing eligible is missed, and the decision is recorded.

Literature and market scans

Problem

A review of fifty papers or reports takes a month.

Build

- 01 Structured extraction per document against a schema agreed with the reader.
- 02 A comparison table and a synthesis with citations to page or section.
- 03 A gap list of what the sources do not say.

Stack

Claude, PDF tools, Postgres

Gate

Twenty documents extracted and compared with the subject expert's own table: field agreement per column before the full run.

Done

The reader starts from a table, not a stack.

Document and data processing

Data and oversight

Schema-guided extraction service

Problem

Each new document type becomes a new manual process.

Build

- 01 A service that takes a JSON schema and a document and returns the fields with confidence and locations.
- 02 A review interface that shows the source span for each field.
- 03 Corrections stored as examples that improve the next run.

Stack

Claude with structured outputs, PDF and OCR tools, Postgres, small review UI

Gate

Per document type, a labelled set; field accuracy and the share needing review are gated.

Done

A new document type is a schema and a day, not a project.

Form and contract intake to systems of record

Problem

Signed forms are scanned, emailed and re-typed.

Build

- 01 Classification of incoming documents, extraction, validation against business rules.
- 02 Writes to the system of record with the document attached and the extracted values stored.
- 03 Exceptions routed with the reason.

Stack

Email and scan intake, Claude, target system API, Postgres

Gate

A month of intake rerun: rule-violation detection and write accuracy against the records the team entered by hand.

Done

Documents become records on arrival.

Long-document summaries with page citations

Problem

Nobody reads the two-hundred-page report, and decisions rest on the summary someone wrote once.

Build

- 01 Chunking with structure awareness, section summaries, then a synthesis.
- 02 Every claim cites a page; a question tool for follow-ups.
- 03 Output in the reader's template.

Stack

Claude, PDF tools

Gate

Summaries checked by the reader against the source on a sample; an uncited claim fails.

Done

Long documents answer questions in minutes, with the page to check.

Data pipeline and ETL automation

Data and oversight

Agent-assisted ingestion for new sources

Problem

Every new data source is weeks of mapping, modelling and tests.

Build

- 01 The agent profiles the source, proposes a schema mapping, the dbt models and the tests.
- 02 The data engineer reviews the pull request; the agent fixes test failures.
- 03 Lineage and documentation generated from the result.

Stack

dbt, Dagster or Airflow, Claude, Git

Gate

Three past sources rebuilt from scratch and compared with production models.

Done

A new source is a reviewed pull request within days.

Schema-drift sentinel

Problem

A renamed column breaks a dashboard three days later.

Build

- 01 Detection of source changes on every run.
- 02 An impact map across models and dashboards, and a proposed fix as a pull request.
- 03 Owners notified with the blast radius.

Stack

Warehouse metadata, dbt, Git, Slack, Claude for the proposed fix

Gate

Seeded drift in a staging copy; every case must be caught and correctly mapped.

Done

Drift is a notification and a fix, not an outage.

Data quality triage

Problem

Quality checks fail and the alert goes to a channel nobody reads.

Build

- 01 Failed expectations trigger a root-cause agent that queries the data and the recent changes.
- 02 A diagnosis with evidence sent to the owner, with a suggested fix.
- 03 A weekly report on failure classes.

Stack

Great Expectations or dbt tests, Claude, warehouse, Slack

Gate

Past incidents replayed; diagnosis accuracy rated by the data team.

Done

Alerts arrive with a cause and an owner.

AI analytics and reporting

Data and oversight

Governed natural-language analytics

Problem

Text-to-SQL over raw tables gives confident wrong answers.

Build

- 01 A semantic layer with certified metrics, joins and access rules.
- 02 An agent that answers only within the modelled scope and says so when a question is outside it.
- 03 Query, result and explanation shown together; charts on request.

Stack

dbt Semantic Layer or Cube, Claude, warehouse, Slack or web

Gate

A question set with known answers per metric; execution accuracy and refusal accuracy both gated.

Done

Business users ask in words and get the number finance would sign.

Automated weekly reporting agent

Problem

The weekly report takes a day to assemble and is read for a minute.

Build

- 01 Metrics pulled by certified queries; narrative drafted only from those numbers.
- 02 Charts generated and the report sent where people read.
- 03 A changelog of what moved and why, with links to the queries.

Stack

Warehouse, certified queries, a charting library, Claude for the narrative, email or Slack

Gate

Four weeks compared with the analyst's own reports for numeric agreement and tone.

Done

The report arrives Monday morning without the analyst's Friday.

Anomaly to explanation

Problem

A metric moves and the investigation takes the afternoon.

Build

- 01 Anomaly detection on certified metrics.
- 02 A drill-down agent that tests the usual dimensions and recent changes, and reports hypotheses with evidence.
- 03 Findings posted to the owner with the queries attached.

Stack

Warehouse, Claude, Slack

Gate

Past incidents replayed; the real cause must appear in the top three hypotheses.

Done

The first hour of every investigation is already done.

Compliance and monitoring automation

Data and oversight

AI agent audit trail and evidence pack

Problem

An auditor asks what the agents were allowed to do and what they did, and the answer is in logs nobody can query.

Build

- 01 A structured decision record per agent action: inputs, tools, gates (hard or soft), approver, outcome.
- 02 Queryable storage and streaming to the monitoring platform.
- 03 An export that maps records to SOC 2 and EU AI Act record-keeping expectations.

Stack

OpenTelemetry, Postgres or a SIEM, Claude for summaries

Gate

An auditor's question list answered from the records alone.

Done

Every agent action can be reconstructed and attributed.

Policy checks on communications and content

Problem

Regulated statements slip into emails, posts and decks.

Build

- 01 Rules for the hard constraints, a model classifier for the judgement calls, each flag with the policy clause.
- 02 A review queue with override reasons recorded.
- 03 A monthly report of flags by team and policy.

Stack

Claude, email and content platform APIs, Postgres

Gate

A labelled corpus of past content; flag precision and recall by policy.

Done

Flagged content is held for review before it is sent, with the policy clause attached.

Control evidence collection

Problem

Audit season means weeks of screenshots and spreadsheets.

Build

- 01 Scheduled collectors across cloud, source control, identity and HR systems.
- 02 Evidence mapped to controls, with freshness and gaps shown.
- 03 An agent that drafts the control narrative from the evidence.

Stack

Cloud and SaaS APIs, Postgres, Claude for the narrative drafts

Gate

Each control's collected evidence compared with last audit's accepted evidence: a control is automated only when its collector matches on every item.

Done

Audit requests are answered from a live evidence store.

AI system register with risk tiers

Problem

Nobody can list the AI systems in use or where the EU AI Act places them.

Build

- 01 An inventory with purpose, data, model, vendor and owner per system.
- 02 Risk tiering against the Act's categories, with the reasoning recorded.
- 03 Obligations per tier tracked to evidence.

Stack

Postgres, Claude, a small web app

Gate

Counsel's tiering of the first twenty systems is the reference; the model's tier must match on every one before it tiers unassisted.

Done

Every AI system in use is listed, tiered and owned.

Legal tech automation

Sector workflows

Contract review against a playbook

Problem

Routine agreements take the same senior hours as the hard ones.

Build

- 01 Clause extraction and comparison with the firm's playbook positions and fallbacks.
- 02 Deviations ranked and redlines drafted in Word, with the playbook reason on each.
- 03 A lawyer accepts, edits or rejects; the system never finalises.

Stack

Claude, Word add-in or document API, Postgres

Gate

Fifty reviewed agreements: deviation recall and redline acceptance by the lawyers.

Done

Lawyers start from a marked-up draft and spend time on the contested terms.

Obligation tracker

Problem

Renewal dates and notice periods are found in the contract after they pass.

Build

- 01 Extraction of obligations, dates and conditions from executed agreements.
- 02 A calendar and alert system with owners.
- 03 A report of upcoming obligations by counterparty.

Stack

Claude with structured outputs, document store, calendar API

Gate

Extraction checked against a lawyer's abstract on a sample; a missed date fails the gate.

Done

Every obligation reaches its owner before its date.

Client intake and conflict check assistant

Problem

Intake is a form, a search and an engagement letter, done by hand each time.

Build

- 01 Structured intake through chat or form, with completeness checks.
- 02 A conflict search across the matter database with results for a person to clear.
- 03 An engagement letter drafted from the template and the intake.

Stack

Claude, matter management API, document templates

Gate

Past intakes replayed; conflict recall must be complete.

Done

New matters open in minutes with the checks done.

Real estate automation

Sector workflows

Lead qualification that books viewings

Problem

Enquiries arrive at all hours and the fast ones win.

Build

- 01 Chat and text replies within seconds, collecting budget, timeline and pre-approval status.
- 02 Viewings booked on the agent's calendar; the CRM updated with the conversation.
- 03 Hot leads escalated to the agent's phone.

Stack

Twilio, Claude, CRM API, calendar API

Gate

Scripted enquiries judged for accuracy and tone; booking accuracy gated.

Done

Every enquiry gets an answer and the good ones get a viewing.

Listing content with compliance checks

Problem

Listings are written late and reviewed for fair-housing wording by memory.

Build

- 01 Descriptions from property data and photos, in the brokerage's voice.
- 02 A compliance check against fair-housing and advertising rules, with the flagged phrase and the reason.
- 03 Formatting for the listing service.

Stack

Claude, property data feed, listing platform API

Gate

A labelled set of past listings; compliance recall must be complete.

Done

Listings go out faster and cleaner.

Maintenance request triage for property managers

Problem

Tenant requests arrive by text and photo and need a person to read every one.

Build

- 01 Category and urgency from text and images; emergency rules are deterministic.
- 02 Vendor dispatch by category with the tenant's availability.
- 03 Status updates to the tenant and a record for the owner.

Stack

Twilio or portal intake, Claude, vendor and property-management APIs

Gate

A year of requests rerun: urgency accuracy gated, and no emergency missed.

Done

Requests are dispatched in minutes and tenants hear back.

Lease abstraction

Problem

Key lease terms are re-read every time a question comes up.

Build

- 01 Extraction of parties, term, rent schedule, options, obligations and critical dates.
- 02 A portfolio table and alerts for upcoming dates.
- 03 Page-cited answers to questions about any lease.

Stack

Claude, document store, Postgres

Gate

Fifty abstracted leases with a specialist's abstracts as the reference: field accuracy per term, and no critical date missed.

Done

The portfolio is a table with the leases behind it.

Education and training automation

Sector workflows

Course builder from source material

Problem

Turning expert material into a course takes months.

Build

- 01 Outline, lessons, examples and quizzes generated from the source, each tied to a learning objective.
- 02 Item review by the subject expert; rejected items regenerated.
- 03 Export to the learning platform.

Stack

Claude, LMS API or SCORM and xAPI export

Gate

Objective coverage and item quality rated by the expert before publish.

Done

A course draft in days, reviewed by the expert.

Rubric-based grading assistant

Problem

Grading is slow and inconsistent across sections.

Build

- 01 Scores per rubric criterion with evidence quoted from the submission.
- 02 Instructor edits any score; consistency checks across the cohort.
- 03 Feedback drafted for the learner in the instructor's voice.

Stack

LMS API, Claude with structured outputs, Postgres

Gate

Agreement with instructor grades on a prior cohort; parity across groups checked.

Done

Instructors review scores instead of producing them.

Tutor grounded in the course

Problem

Learners get stuck at night and the forum answers come late.

Build

- 01 Answers only from the course material, with references to the lesson.
- 02 A Socratic mode that asks before it tells.
- 03 Misconceptions reported to the instructor weekly.

Stack

Claude, pgvector, LMS integration

Gate

A question set from past forums; answer accuracy and no-invention rate.

Done

Learners get help at any hour, within the course.

Training compliance tracker

Problem

Mandatory training is tracked in a spreadsheet and chased by email.

Build

- 01 Assignments by role, reminders, and completion evidence collected from the platform.
- 02 A manager view and an audit export.

Stack

LMS and HRIS APIs, Postgres, email

Gate

A quarter of completion records reconciled against the learning platform: any assignment or completion missing from the export fails.

Done

Compliance status is live and provable.

Healthcare workflow automation

Sector workflows

Patient intake and triage summaries

Problem

Intake forms are long, and the clinician reads them in the room.

Build

- 01 Structured intake through a form or chat, summarised for the clinician with flagged items.
- 02 No diagnosis or advice; the summary is organised, not interpreted.
- 03 Protected health information minimised and handled only through services under a business associate agreement.

Stack

Claude via a BAA-covered platform, FHIR-capable EHR API, encrypted storage

Gate

Summaries reviewed by clinicians on a sample; any omitted flagged item fails.

Done

The clinician walks in with the summary already read.

Prior authorisation packet assembly

Problem

Prior authorisations are assembled by hand against each payer's rules.

Build

- 01 Payer requirements as checklists; the chart read through FHIR for the evidence.
- 02 A missing-items list for the authorisation team and a drafted packet.
- 03 Status tracking and follow-up tasks.

Stack

FHIR APIs, Claude via a BAA-covered platform, Postgres, task queue

Gate

Past authorisations replayed; packet completeness checked by the authorisation team.

Done

Packets go out complete the first time.

Referral and fax processing

Problem

Referrals still arrive by fax and sit in a queue.

Build

- 01 Fax to text to structured referral: patient, reason, urgency, sender.
- 02 A task created in the EHR with the document attached; urgent ones flagged.
- 03 A daily report of what arrived and what is pending.

Stack

Fax gateway, OCR, Claude via a BAA-covered platform, EHR API

Gate

A month of referrals replayed; extraction accuracy and urgency recall.

Done

Referrals are tasks within minutes of arriving.

Clinic phone agent for scheduling

Problem

The phone line is the bottleneck, and most calls are scheduling.

Build

- 01 A voice agent for scheduling, rescheduling and directions, with warm transfer for anything clinical.
- 02 Identity verification steps before any appointment detail is read.
- 03 Audit of every call and recording retention per policy.

Stack

Telephony, LiveKit or Pipecat, speech vendors under BAA, Claude, scheduling API

Gate

Scripted calls including attempts to extract another patient's information; verification must hold.

Done

Scheduling calls are handled and clinical calls reach a person.

[Book a call](#) · muhammadaliqasmi+work@gmail.com

Updated 2026-10-03