

# Patryk Rosłonek

Senior IT Project & Portfolio Manager / Delivery Lead  
AI-Augmented Delivery, Human Accountability  
Agile & Hybrid Leader

rosłonek.patryk@gmail.com  
+48 502 984 266  
linkedin.com/in/patryk-rosłonek-pm

AI is **production infrastructure** in my delivery work: it mines more material than a team could read and hands back the pattern, and it runs whole processes once the pipeline exists. What it does not carry is **consequence**. Steering 30+ enterprise products across 10 specialised teams, I build and operate the AI systems that absorb the routine load of the role. The 2026 bottleneck is no longer producing work but **verifying** it, and verification capacity is senior engineers, Tech-Leads and the PM. So I point AI at volume and keep judgement with the experienced: **every system below stops at a human approval gate**.

## AI-Augmented PM Operating System

*A delivery cockpit built on an agentic AI CLI (Claude Code + Claude Opus), wired into my toolchain and versioned in my own repository.*

- **Eight AI "skills" authored to date** - in-repo, reviewable agents that encode my craft for a named audience: weekly status, CEO and board portfolio summary, client-facing delivery statement, portfolio risk register, company-newsletter input, customer profiling, specification Q&A, team-improvement cycle.
- **Each one is a contract, not a prompt** - live context in, topics proposed for my sign-off, a draft in that audience's house style, then a **stop for human approval**.
- **Persistent memory layer** - a curated, file-based knowledge base the agent recalls across sessions (team roster, billing model, program structure, prior decisions), so context compounds instead of resetting every conversation.
- **Live tool integrations (MCP)** - Azure DevOps (work items, WIQL), Google Workspace and GitHub, so the agent reasons over real project data, not a description of it.

## Production Agentic Automation - self-hosted n8n

*Designed, hosted and operated end-to-end: own VM today, Kubernetes next.*

- **Migrated 18 Power Automate flows onto n8n** and consolidated them roughly 4:1. Re-architected from polling to **event-driven Azure DevOps Service Hooks** with field-level filters and loop protection, after diagnosing a long-standing duplicate-work-item defect from the flow definitions rather than from the symptoms.
- **RAG meeting-summary agent** - a Fireflies.ai transcript feeds a **Claude Opus agent grounded in a retrieval-augmented knowledge base**, producing a **bilingual PL/EN business summary** written straight to Google Docs and GitHub. **95 wrap-ups** to date, with KB caching, restart detection and a recovery path for API outages.
- **Status-archival agent** - every sent status parsed to clean Markdown and versioned by idempotent upsert: **72 weekly statuses** as a searchable project chronicle.
- **Team-availability pipeline** - headless automation pulling approved absences from a login-walled HR SaaS with no API, re-pointed when the vendor changed - **an access problem solved with engineering, not permissions**.

## Decision Intelligence - clients, forecasts, specs, audiences

*Where AI sharpened what I know about the business - and about who is reading.*

- **Probabilistic portfolio roadmap, shared with the client** - long requested, never kept current by hand, now it maintains itself: scope only from my approved weekly status, progress from Azure DevOps, dates from a **10,000-run Monte Carlo forecast** (P50/P85) on each topic's own pace. Commitment, forecast and "no data" are drawn apart: too little history means **no date, not a guess**. Behind company SSO, refreshed four times a day.
- **Customer profiling and business-need projection** - evidence-graded dossiers from public sources that map a client's or prospect's **business** needs now (0-6 months) and next (6-18 months): from conversion and churn to the AI gap versus competitors and continuity risk. Every need carries a metric, a source and a confidence grade; hypotheses stay labelled as hypotheses. A cross-account register keeps the account advised, not just served.
- **Grounded specification Q&A** - **845 requirement documents** indexed from the moderated shared drive, distilled on demand with staleness detection and answered **with citations** (module, section, source link), never from memory.
- **Audience profiling** - the same verified facts re-cut for board, client sign-off, newsletter and Tech-Leads, each at its own altitude and burden of proof.

### AI & Automation Stack:

Claude Code (agentic CLI) ·  
Claude Opus / Anthropic API ·  
agent & prompt engineering · RAG  
/ vector stores · Gemini  
embeddings · MCP integrations ·  
n8n (self-hosted) · GCP Compute  
/ Docker · Azure DevOps REST &  
Service Hooks · Power Automate  
· Monte Carlo forecasting ·  
OAuth2 / OIDC (Microsoft Entra  
ID) · Caddy · Playwright ·  
Fireflies.ai · Google Workspace  
API · Git / GitHub

### By the numbers:

- 8** AI "skills" authored in my delivery CLI, each with a human approval gate
- 18** Power Automate flows migrated onto self-hosted n8n and consolidated ~4:1
- 845** requirement documents indexed for grounded, cited spec Q&A
- 95** bilingual meeting wrap-ups produced by the RAG agent
- 72** weekly statuses auto-archived into a searchable chronicle
- 10k** Monte Carlo runs behind every roadmap date
- 30+** enterprise products across 10 specialised teams (~30 FTE)

### What I do not automate:

People decisions · development  
feedback · accountability · the  
confidential half of a team  
conversation · anything a client  
signs

## The Human Factor in the Loop

*Why I build this way - and why the systems are trusted enough to stay in daily use.*

- **Accountability does not delegate.** An agent can research, synthesise and execute; a named person decides and signs. Every artefact that leaves my desk - client statement, board summary, risk register, status - passes a human approval gate. Automation earns its place by being reviewable, not by being invisible.
- **I spend automation budget on protecting senior capacity, not on generating more work for it.** AI output is cheap; expert review is the scarce resource. The targets are synthesis, retrieval, reporting and orchestration - never the judgement calls that need an engineer's experience.
- **Senior engineering expertise is what makes AI safe to use, and it is becoming more valuable, not less.** Tech-Leads and senior developers are the layer that catches the plausible-looking answer, the misread requirement, the wrong edge case. My job is to keep that layer unclogged, credited and visible to the business.
- **Zero fabrication, by contract.** Missing data comes back as an explicit "none", never invented; sources are cited; hypotheses are labelled. Where a number matters to a client or the board, it traces back to a work item, a specification or a sent message. A roadmap topic without history gets no date, not a plausible one.

## People & Team Leadership

*10 specialised teams, ~30 FTE, four Tech-Leads plus analysis, design and QA leadership. None of this is an AI problem.*

- **A monthly two-level improvement cycle** I designed for the portfolio: a short per-team check with each lead, feeding a cross-team forum for the expensive problems that live in the seams between analysis, design, development, QA and deployment. Built **around** the engineers, not at them: veteran leads who reject ceremony vocabulary get a deliverable, not a ritual. It is also explicitly **not an evaluation instrument**, and that promise is what makes the input honest.
- **Half-yearly development feedback** for the developers in my portfolio, written for their line managers in three separated layers: evidence of real scope, the Tech-Lead's own words, then my PM view and recommendation. Grounding the first layer in delivery data **changed three of five assessments** - it surfaced infrastructure work that status reports hide by design, because its success criterion is that nothing happened.
- **Availability, capacity and continuity** - absence and capacity data pulled automatically into reporting; a structured hand-over brief covering five delivery areas during a C-level absence; blockers named with the person and the dependency, not hidden behind a RAG colour.

## Engineering & Data Practice

- **Auditable, idempotent pipelines** - dated paths, slugging, de-duplication, stable identifiers across editions and a diff against the previous version, so every run is reproducible and every change is explainable; forecasts are logged for calibration.
- **An AI-ready data foundation before the AI** - a curated corpus manifest with freshness tracking, a roster and a decision log as single sources of truth, anonymisation rules for internal reporting. Retrieval quality is a data-governance problem wearing a model's clothes.
- **Agent engineering** - role-anchored prompts with deterministic output contracts, mandatory retrieval before writing, domain dictionaries that correct recurring transcription errors, service-identity and loop-protection decisions taken explicitly rather than discovered in production.

## Governance & Internal Enablement

- **Delivery framework ownership** - a custom Azure DevOps framework for 10 development teams: work-item taxonomy, states, custom fields and the automation that keeps design and specification artefacts in step, including a template-tree clone that bootstraps each new client onboarding in one click.
- **I teach what I build.** Ran an internal workshop opening my environment to PM and analyst colleagues, and maintain the follow-up track they asked for: RAG and workflow mechanics, how to build an AI skill and how to put limits on it, querying delivery data for dependencies and bottlenecks. Organisational AI literacy of the kind the EU AI Act expects employers to be able to evidence.

### Signals I build against:

**Verification is the new bottleneck.** Senior engineers report spending more time reviewing AI output than building; reviewer capacity, not generation speed, is the scaling constraint.

State of AI Code Quality 2026

**AI agents cannot be accountable entities** - AI-generated recommendations still need human review and a human owner. Gartner, PMO guidance

**AI literacy is an employer duty** in the EU, enforceable from August 2026, and one organisation has to be able to demonstrate.

EU AI Act, Article 4

**Hiring shifted to proof of work** - what was built, for whom, with what measurable effect, over tool name-dropping.

LinkedIn / market reports 2026

### AI Certifications:

**AI in Business Development**  
Google & SGH

**AI in Business Development**  
Ideas Generator

**Introduction to Generative AI**  
Google

**Introduction to Responsible AI**  
Google

### Operating Principles:

**Human-in-the-loop** - AI drafts, the PM decides, a person signs.

**Zero fabrication** - unknown stays unknown.

**Protect the experts** - remove routine load, not judgement.

**Auditable & idempotent** by construction.

**Context that compounds** - memory over restarts.

**Own the platform** - self-hosted, documented, reversible.

**Build over wait** - engineer around blockers.