AMIR FIROUZMANESH, PhD

Senior Software Engineer — AI Integration · Distributed Systems

Dubai, UAE • amir@AandZ.tech • linkedin.com/in/amir-firouzmanesh • github.com/amirhf
Portfolio: AandZ.tech

PROFESSIONAL SUMMARY

Senior software engineer (PhD) and founder of AandZ.tech, an independent AI & cloud engineering practice based in the UAE. Proven track record of shipping high-impact features, including Editor AI for Windows Photos (+20% MAU) and reducing support tickets by over 50% at Property Finder. Now helps teams design and build AI-powered, event-driven systems by blending hands-on coding with clear architecture and data-driven decisions. Open to freelance, contract, and remote roles in AI integration and distributed systems.

TECHNICAL SKILLS

- **Languages:** Go, Python, Java, JavaScript/TypeScript, SQL.
- **AI/ML:** PyTorch, LLM APIs (OpenAI, Anthropic, Gemini), LangChain, ONNX Runtime.
- **Cloud & Infra:** AWS (Kinesis, Glue, Athena, Step Functions, RDS, S3), Azure, Docker, Kubernetes.
- **Data/Storage:** PostgreSQL, MySQL, SQL Server, MongoDB, Qdrant, pgvector.
- **Messaging/Streaming:** RabbitMQ, Kinesis, Kafka/Redpanda.
- **Observability:** Grafana, Application Insights, Prometheus, Jaeger.
- **Architecture:** Microservices, Event-Driven, CQRS, DR/HA, Idempotency, Transactional Outbox, CI/CD.

PROFESSIONAL EXPERIENCE

AandZ.tech — **Founder & Principal Engineer**
Dubai, UAE — *2025 – Present*

- Founded an independent engineering practice focused on generative AI integration, event-driven architectures, and cloud-native platforms for startups and digital businesses.
- Designed and built open-source reference systems acting as blueprints for client work: Credit Ledger, AI Image Search, and Learning Path Designer.
- Standardized reusable infrastructure (Docker Compose, migrations, seeders, observability) to enable clients to quickly demo, extend, or self-host systems.
- **Stack:** Go, Python, TypeScript, FastAPI, Next.js, Qdrant/pgvector, Postgres, Docker, Kafka/Redpanda.

Property Finder — **Senior Software Engineer (B2B Subscriptions/Credits)**
Dubai, UAE — *2025*

- Cut support tickets by **50%+** on listing publish/credits flows by clarifying ownership rules, fixing root causes of failures, and simplifying error paths.
- Led the move from a single codebase to microservices for subscriptions/credits, defining simple contracts between teams to ensure safer and faster releases.
- Implemented clear dashboards and alerts, improving on-call confidence and enabling the team to resolve incidents faster.
- **Stack:** Go, TypeScript, AWS, MongoDB/Postgres, RabbitMQ, Docker, Grafana, Application Insights, CI/CD.

Microsoft — **Software Engineer (Windows Photos & Designer AI)**
Canada — *2021 – 2025*

- Launched Editor AI features (background/object selection, smart erase, upscaling) driving a **+20%** increase in monthly active users for Windows Photos.
- Made AI fast and dependable by combining on-device models with a cloud fallback when needed—so most users get instant results and everyone gets consistent quality.
- Improved performance of heavy edits by reworking how images are processed (tiling/batching, careful memory use), cutting the P95 time-to-result for users.
- Stabilized releases by reducing flaky tests and adding quality gates tied to real user metrics; fewer rollbacks and more confident shipping.
- Worked closely with design/PM to instrument adoption and retention, then used those insights to prioritize fixes that mattered most to users.
- **Stack:** TypeScript, C#, Python, ONNX Runtime, Azure AI endpoints, Docker, Windows APIs, Telemetry, CI/CD.

Amazon Web Services (AWS) — **Software Engineer (RDS SQL Server)**
Canada — *2019 – 2021*

- Expanded RDS SQL Server capabilities by enabling heterogeneous linked servers and contributing to SSAS/SSIS support, facilitating easier migrations and analytics.

- Improved seller reporting pipelines (Kinesis → Glue → Athena) to ensure reliable data delivery even with noisy upstream sources.
- **Stack:** Java, AWS, SQL Server, PostgreSQL, IaC, Observability, CI/CD.

DataGardens (CenturyLink) **Software Engineer (DR/HA)**
Canada *2014 – 2018*

- Reduced onboarding time from 1+ hours to ∼10 minutes by automating deployment and creating comprehensive runbooks.
- Added health checks and safe failover steps to ensure predictable production cutovers.
- **Stack:** Python, Go, Java, RabbitMQ, Ansible, AWS/Azure/GCP, Linux.

SELECTED PROJECTS & REFERENCE IMPLEMENTATIONS

Credit Ledger (Go + Postgres + Event-Driven) github.com/amirhf/creditLedger

- Reference implementation of a double-entry ledger for subscription platforms featuring transactional outbox, idempotent consumers, and replayable CQRS projections.
- *Stack:* Go 1.22+, Postgres, Kafka/Redpanda, CQRS, Docker.

Retail AI: Polyglot Search & Feature Router github.com/amirhf/imageSearch

- Architected a high-concurrency **Retail Discovery Engine** using **Go** for search orchestration and **Python** for ML inference; implemented **Hybrid Search** (Qdrant) to combine dense vector recall with sparse keyword precision.
- Designed the **AI Feature Router** pattern, a "circuit breaker for intelligence" that reduces cloud costs by ∼90% by dynamically routing requests to local quantized models (BLIP/SigLIP) before falling back to premium APIs.
- Achieved sub-100ms p95 latency via **gRPC** connection pooling and Qdrant index optimization, utilizing **OpenTelemetry** for distributed tracing across the polyglot stack.
- *Stack:* Go, Python (FastAPI), Qdrant, Redis, OpenTelemetry (Jaeger), Docker.

Learning Path Designer (RAG + Agent Orchestration) github.com/amirhf/learningPathDesigner

- RAG and agents demo that assembles grounded learning paths and quizzes, utilizing a Go gateway and Python services for content generation.
- *Stack:* Next.js, Go, Python, Qdrant, Postgres.

EDUCATION

PhD, Computer Science University of Alberta

Research focused on computer vision and image processing; completed coursework and a research project in distributed systems. Several peer-reviewed papers have been published in computer vision and image processing (see Google Scholar: https://scholar.google.com/citations?hl=en&user=8XrRyZMAAAAJ)