

KOMOL NATH ROY

+1 (437) 876 3995 • rax.komol@gmail.com • [linkedin/knroy](https://www.linkedin.com/in/knroy) • Windsor, ON

Summary

Dynamic full-stack engineering professional with 8+ years of experience in solving complex technical challenges. Expertise in system design, process optimization, and cross-functional collaboration, complemented by hands-on experience in multi-tenant SaaS patterns and microservice-based architecture design. Driven by analytical insight and innovative thinking to consistently deliver efficient, high-quality solutions that exceed project goals.

Skills

Frontend: JavaScript, TypeScript, Angular, ReactJs

Backend: GoLang, GoFiber, Python, Django, FastAPI, Python RQ & ARG, DjangoRQ, Django REST Framework, NodeJs, ASP .NET

DBMS: MySQL, MongoDB, Aws DynamoDB, PostgreSQL

Tools: Git, Jira, Slack, Redis, ValKey, ElasticSearch, RabbitMQ, Docker, Kubernetes, Google Cloud CI/CD, GCP Cloud Storage,

Google Kubernetes Engine

AI: LLM Integration, Agents & Workflow Automation with n8n

Experience

PwC Middle East (PricewaterhouseCoopers)

Dubai, UAE

Full Stack Software Engineer | Manager | Tax (Service Provider)

01/2022 - Present (Hybrid)

- Led the design and development of a scalable **E-Invoicing Generating Solution (EGS)** using Spring Boot, NodeJS, and Django REST Framework. Ensured seamless integration with retry mechanism to maintain highest possible accuracy and pass rate for submitted E-Invoices with XML & XSD Validation.
- Successfully implemented a microservices architecture, dividing complex functionalities into modular, independent services. This approach not only enhanced system maintainability but also paved the way for future scalability and feature expansion.
- Developed a PDF Engine with a focus on precision and user experience. Achieved pixel-perfect rendering of invoices, significantly improving client satisfaction and brand perception.
- Implemented robust Continuous Integration and Continuous Delivery (CI/CD) pipelines. This not only ensured faster time-to-market for features but also enhanced overall software quality.
- Integrated Redis for caching, resulting in a huge improvement in data retrieval times.
- Designed and Developed the following Microservices - Storage, Mail, Notification (WebSocket), UAM (User Access Management), UI Language Manager, EIS (E-Invoicing Business Service), PDF Engine Service
- **Technology:** TypeScript, Angular 15, Python, Django, Django REST Framework, Java Spring Boot, Gradle, NodeJS, PuppeteerJS, Redis, MongoDB, Nginx, Docker, GCP(Bucket Storage, Cloud Build & Triggers, Kubernetes Engine, Compute Engine & LB)

Secure Link Services Ltd (SELISE)

Dhaka, Bangladesh

Full Stack Software Engineer

08/2020 - 05/2022

- Spearheaded the design and development of a comprehensive workflow management system for the Swiss insurance, pension, and broker market.
- Led the frontend initiatives while actively contributing to backend development using .NET, ensuring robust integration between client-side features and server-side logic.
- Engineered a solution that streamlined workflow tracking and efficient client data importation/management.
- Successfully integrated the portal as a Microsoft Teams app, enhancing in-team collaboration and operational efficiency.
- **Technology:** TypeScript, Angular, Service Worker, Microsoft Teams, Azure Active Directory, .NET Core, Docker

Metropolitan University, Bangladesh

Sylhet, Bangladesh

Lecturer

10/2018 - 07/2020

- Delivered engaging lectures in Structured Programming, Data Structures, Algorithms, Microprocessors & Assembly Language, Relational Database Management Systems, and Natural Language Processing, blending theoretical foundations with practical applications.
- Designed and updated comprehensive course materials, lab exercises, and project-based assignments to facilitate hands-on learning and real-world problem solving.
- Supervised and mentored final year student projects, guiding research, technical development, and innovation to achieve high-quality project outcomes.
- Introduced interactive teaching methodologies and industry-relevant case studies, significantly enhancing student engagement and learning effectiveness.
- Worked closely with academic peers to integrate emerging technologies and industry trends into the curriculum, ensuring that course content remained current and competitive.

Secure Link Services Ltd (SELISE)

Dhaka, Bangladesh

Software Engineer

09/2017 - 10/2018

- Developed critical frontend features and platform applications on a generic framework using Angular and TypeScript, enhancing scalability and reusability.
- Engineered a complete single-page responsive web solution, ensuring a seamless user experience across devices.
- Integrated robust data management capabilities with MongoDB and REST APIs, while employing Git for version control in an Agile Scrum environment.
- Proactively researched emerging technologies and industry trends to continually refine and update the technical stack.
- **Technology:** JavaScript, AngularJS, Angular 2, TypeScript, MongoDB, Git, Agile, Scrum, REST API

Windmill Infotech Ltd.

Software Engineer

- Developed and deployed custom Android applications for diverse clients, ensuring high-quality user experiences and successful launches on the Google Play Store.
- Optimized existing Android apps by fine-tuning code and performance parameters, resulting in faster load times and improved overall efficiency.
- Implemented robust unit tests to maintain code quality, while collaborating closely with backend engineers to integrate APIs and streamline end-to-end functionality.
- **Technology:** Java, XML, Android, REST API

Dhaka, Bangladesh

04/2017 - 09/2017

Education

Metropolitan University, Bangladesh

Bachelor of Science in Computer Science and Engineering - January 2013 - December 2016

- Solving programming problems in different online judges
- Codeforces: cse.komol (rating - 1329)
- UVA: Solved 346 problems (handle: cse.komol)
- Participated ICPC Dhaka Regional 3 times

Personal Projects

Django-PDS

<https://github.com/knroy/django-pds>

- Faster REST API implementation with Django and MongoDB
- Row (Document) Level Security in-built
- Generic CRUD endpoints implementation
- SQL like query from from frontend with row level security

Celery RMQ

<https://github.com/knroy/celery-rmq>

- Ease implementation of Celery and RabbitMQ
- Producer and Consumer Message Broking
- Standalone and not limited to specific framework, Can be integrated with django and flask

Vortex

(Framework-as-a-Service Architecture with GraphQL enabled for dynamic schema with Row Level Security)

- Built **Vortex**, a FastAPI **Framework-as-a-Service** that standardizes microservice bootstrapping (routers, docs, health, permissions), middleware, and integrations, cutting new-service setup from weeks to hours.
- Implemented a **multi-tenant** request lifecycle: origin/host-based tenant resolution, **JWT validation per-tenant**, automatic tenant-scoped Mongo access, and **tenant-aware cache** keys.
- Delivered an **AWS-style RBAC engine** (roles, policies, bindings, resource URNs, wildcard actions) with FastAPI dependency hooks and **cache-backed PDP** for low-latency auth decisions.
- Created a **shared middleware stack** (auth, tenant validator, idempotency, rate limiting, audit logging, trace IDs, security headers, CORS/GZip) that's auto-wired for every service.
- Built an **ARQ background worker** harness that mirrors HTTP startup (**Mongo/Redis/RabbitMQ/Elasticsearch/cache**) and propagates AuthContext so **async jobs** remain tenant-aware.
- **Designed a centralized logging + audit pipeline**: structured logs enriched with tenant/user/trace metadata, queued via ARQ, persisted per-tenant in Mongo, and indexed in Elasticsearch for admin search.
- Established a **standard microservice layout** (<service>/routers | handlers | requests | responses | schemas | repositories | services | docs) with docs/cookbooks, improving onboarding and consistency.
- Added orchestrator patterns for service-to-service calls using shared auth/tenant rules, reducing cross-service glue code and integration churn.
- **Quantified and documented time/cost savings** (2-4+ developer weeks per service) versus building foundations from scratch, supporting adoption across teams.
- Ensured operational hygiene: cache bootstrap for tenants/sites/sensitive fields, cache invalidation strategies for RBAC changes, and guidance on tenancy/auth/observability best practices.
- Covering IAM/RBAC, control-plane + tenancy, schema/GQL (PDS), storage, mail + notifications, payments/subscriptions, orchestration, centralized logging/search, and cache management.