Domenico Nota

domenico.nota@mail.utoronto.ca | linkedin.com/in/domenico-nota-bb4b873ba | github.com/DomenicoNota | domeniconota.ca

Education

University of Toronto Toronto, ON

Bachelor of Science in Computer Science, Minor in Mathematics *September 2023 – May 2027*

- Relevant coursework: Data Structures and Algorithms, Operating Systems, Databases, Software Engineering, Object-Oriented Programming, Machine Learning

Experience

Software Engineer Intern January 2026 – Present

Bid Engine *Toronto, ON*

- Cut supplier catalog onboarding **6 hours** to **8 minutes** with a Python OCR pipeline parsing price sheets to SKUs
- Raised tender line-item extraction coverage from **71%** to **94%** by segmenting scoped work along CSI divisions
- Reduced agent p95 latency **820ms** to **490ms** with a Java Spring Boot MCP server exposing tender and pricing
- Lifted top-1 SKU match accuracy from **68%** to **91%** using pgvector hybrid search with a cross-encoder rerank
- Automated **87%** of tender estimates with a Python LangGraph agent routing sub-**80%** matches to human review
- Sustained **2,400** invitations per minute at p95 under **900ms** via Kafka fanout across Spring Boot services on AKS

Student Machine Learning Engineer January 2025 – April 2025

University of Toronto *Toronto, ON*

- Delivered a React and Flask stock trading simulator across **14** REST endpoints by owning the full stack solo
- Sustained **10,000+** market ticks per minute at **99.4%** write success by batching live ingestion into MongoDB
- Cut forecast refresh cost **87%** by scheduling windowed model jobs against cached MongoDB feature reads
- Raised deploy success rate to **96%** by containerizing services behind a GitHub Actions test and build pipeline

Web Developer Sep. 2024 – Dec. 2024

UofT.ai *Toronto, ON*

- Built a responsive web application using React and Chakra UI, improving usability for **700+** monthly users
- Implemented a mobile friendly interface with accessibility features to support cross device compatibility
- Deployed application on AWS EC2 with a CI/CD pipeline, enabling automated builds and reliable releases

Projects

LogStream Observability Platform | *Rust, Go, PostgreSQL, Docker, Kafka, React* GitHub

- Sustained **50MB/s** log ingestion at **99.9%** delivery via a Rust agent batching tailed events through Kafka
- Cut search p95 from **1.4s** to **120ms** with cursor pagination and partial indexes on a Go and PostgreSQL API
- Held log loss at **zero** across **12** failure drills using dead-letter queues, drop counters, and health checks

Fault Tolerant Order System | *Python, FastAPI, RabbitMQ, PostgreSQL, AWS(EC2, RDS, Cloud Watch)* GitHub

- Built a distributed microservices platform with **RabbitMQ** messaging for order, payment, inventory, and shipping
- Implemented **retry mechanisms** and **dead letter queues** achieving **99% recovery rate** in failure scenarios
- Deployed on **AWS EC2** with **Docker Compose** and **CloudWatch** monitoring for queue depth and latency

Distributed Search Engine (Mini Elasticsearch) | *Java, Spring Boot, Docker, JMH* GitHub

- Built a full text search engine with an **inverted index** and **TF-IDF** ranking to return relevant results efficiently
- Implemented sharding and replication with **consistent hashing** and Java concurrency to support parallel queries
- Benchmarked performance using **JMH** and **JMeter**, sustaining **5K+ queries/sec** under 200ms latency

Technical Skills

Languages: Python, JavaScript, TypeScript, C, Java, SQL, HTML, CSS
Technologies: React, Next.js, Node.js, Express, FastAPI, Spring Boot
Tools: AWS, GCP, Docker, Redis, Postman, Git, Nginx, Kafka
Concepts: Distributed Systems, Microservices, Event-Driven Architecture, REST APIs