

# RICHARD ZHANG

[r29zhang@uwaterloo.ca](mailto:r29zhang@uwaterloo.ca) | [linkedin.com/in/rz2004](https://linkedin.com/in/rz2004) | [github.com/notzree](https://github.com/notzree) | [richard-zhang.ca](https://richard-zhang.ca)

## EDUCATION

### University of Waterloo

Waterloo, ON

BASc, Systems Design Engineering 3.85 / 4.00 GPA

2022 – 2027

- Relevant Courses: Data Structures and Algorithms, Digital Computation (C++), Engineering Prototyping

## SKILLS

**Languages:** Go, Python, Rust, TypeScript, Bash, C#

**Frameworks:** gRPC, Chi, Express.js, Next.js, .NET, Django

**Tools:** Git, Docker, K8s, REST, AWS (ECS, RDS), PostgreSQL, Redis, Pulumi, REST

## PROFESSIONAL EXPERIENCE

### Blendable

Waterloo, ON

Software Engineering Intern

May 2024 – Aug. 2024

- Shipped internal Python observability APIs for insurance transfers, **saved \$100/month** by replacing accounting SaaS and decreased observability report generation times by **63%** through reduced API calls
- Refactored invoice and billing services with **extensible classes, abstractions, and functional options** to easily onboard 5 new report variants and consolidate document generation logic for over **80+ report variants**
- Overhauled ACH payments API and database schema to support Managing General Agencies (MGAs) using **Django** to enable the onboarding of backlogged **insurance plans totalling \$1.9M+**
- Increased code coverage by implementing **Python unit tests** for critical invoice components ensuring reliable behaviour

### Jitto (CDL 2023/2024)

Toronto, ON

Fullstack Engineering Intern

Aug. 2023 – Dec. 2023

- Led REST API and infrastructure migration from **DynamoDB** to **RDS**, reducing database size by **9%** through **database normalization** and enabling multi-supplier product support through better relational modelling
- Developed **Python Lambda function** automating weekly ingestion of PDF pricing data, improving the accuracy of produce pricing algorithm by **reducing data staleness a hour to 2 minutes**
- Automated phone call orders by using Twillio and LangChain to build AI services to converse and derive order details

### EnergyIntell

Toronto, ON

Software Engineer Intern

Jan. 2023 – Apr. 2023

- Architected ETL pipeline moving **50+ million rows/year** of historical electrical data to cold storage using .NET with Entity Framework, lowering archival job time from **2 hours to under 10 minutes** and reduced database volume by **20%**
- Modernized energy data SOAP web service into REST API to support interoperability using .NET and LINQ, reducing fetch speeds by **21%** and payload size by **40%**

## PROJECTS

Wikigraph - Measure things with Wikipedia | Go, Rust, MySQL, Docker

[Github](#)

- Engineered **Rust-based** Wikipedia link parser **compressing 92 GB of XML Data into a 1.27 GB** Binary graph format, achieving a **6 hour** faster runtime compared to alternative parsers (Wikicrush)
- Achieved a **35x improvement** in path finding speeds from **1.2s to 3ms** across multiple tests by implementing concurrent, bi-directional BFS using **goroutines and concurrency primitives**

Uprank (beta) - Freelancer Analytics SaaS | Go, Python, gRPC, Pulumi, AWS(SQS, ECS)

[Demo](#)

- Implemented **concurrent** job ranking pipeline with **Go**, using **Python gRPC functions** with Pandas and Pinecone to compute freelancer data features

Fly.io Gossip Glomers | Go, RPC, Distributed Systems

[Github](#)

- Designed and implemented a fault-tolerant, efficient **distributed broadcasting system** on Maelstrom, beating both latency and message-per-op benchmarks by **78% and 22%** respectively