

# Ricky Luo

Canadian & US Citizen | [rickyluo7@gmail.com](mailto:rickyluo7@gmail.com) | [linkedin.com/in/rickyl21](https://linkedin.com/in/rickyl21) | [github.com/rickyl21](https://github.com/rickyl21)

## EDUCATION

### University of British Columbia

Bachelor of Science in Computer Science

Expected Graduation: Apr 2026

Vancouver, BC

- Courses: Data Structures & Algorithms, Relational Databases, Computer Networking, Software Engineering

## WORK EXPERIENCE

### Boeing

Jan 2024 - Jun 2024

Software Engineer Intern

Vancouver, BC

- Built an end-to-end dashboard in React serving **75 clients**, integrated with a PostgreSQL back-end via SQLAlchemy.
- Reduced load time for approximately **10,000 flight data points** by optimizing API routing using Python and Flask.
- Optimized data operations speed by **37%** through refactoring execution calls in Google Cloud Workflows.
- Implemented scalable microservices architecture, improving data processing by **30%** using Django and REST APIs.
- Spearheaded Databricks integration by converting **80+** worker outputs to Delta tables with Pandas and PySpark.

### Bank of Montreal (BMO)

Jan 2023 - Aug 2023

Software Engineer Intern

Toronto, ON

- Automated AWS deployment approvals, saving **~\$55k per month** via a Python tooling solution and GitHub Actions.
- Accelerated policy creation time by **33 times**, creating an automatic parsing API with Babel, Next.js, and Node.js.
- Decreased steps needed to view AWS details by **79%** by developing a React dashboard, using AWS SDK calls.
- Monitored a Jenkins pipeline **supporting 24+ teams** by setting up performance indicators in AWS CloudWatch.
- Enhanced code quality through Jest tests (**83% coverage**) and ensured code safety by refactoring to TypeScript.

## PROJECTS

### Simple Domain Name System Client | Java, java.net, TCP, UDP

- Created a DNS resolver, achieving resolution speeds comparable to dig and supporting 5 query types using Java.
- Built to parse and handle 512 byte messages for both individual and iterative DNS queries using DatagramSockets.

### League of Legends Discord Bot | discord.py, Python, HTTPX, DynamoDB

- Developed a Discord bot used in 3 servers with 370+ users that uses Riot API with HTTPX to query user data.
- Set up DynamoDB back-end to store 40+ users' information and query history to reduce REST API dependency.

### PowerPoint to Notes Converter | MongoDB, Express.js, React, Node.js, Next.js, Python, TypeScript

- Prototyped a full-stack application that uses OpenAI API to convert .pptx and Google Slides into simplified notes.
- Designed the front-end using React and Tailwind with user authentication using Auth0 to ensure 100% security.

### Pokémon Quiz Web App | JavaScript, React, Node.js, Axios, Material-UI, HTML, CSS

- Engineered a fully responsive quiz web app with a React front-end, Node.js back-end, and Material-UI for styling.
- Integrated RESTful APIs, improving load times to 1 second by leveraging Axios to dynamically fetch and display data.

### Receipt Tracker Web App | Python, JavaScript, Flask, GCP, HTML, CSS

- Created a receipt-scanning app that won 6ixSideHacks using Python, Google Cloud Vision API, and Google Sheets API.

## TECHNICAL SKILLS

**Languages:** Java, Python, C, C++, JavaScript, TypeScript, HTML, CSS, Bash, SQL, Go (Golang), Kotlin

**Frameworks/Libraries:** React.js, Next.js, Express.js, Tailwind CSS, Jest, JUnit, Flask, Django

**Technologies:** Node.js, AWS, GCP (Google Cloud Platform), Git, Unix, NoSQL, Postman, Docker, GraphQL, Redis, Azure

**Concepts:** Test-Driven Development, Agile, Distributed Systems, Object-Oriented Programming, CI/CD, Machine Learning