Jerry Xu

jerryxu99.github.io/
linkedin.com/in/jerryxu-/

jerry.xu99@gmail.com
+1-604-367-2733
github.com/jerryxu99

Education

- **University of British Columbia** **GPA: 4.0**
  Combined Major in Computer Science and Business *Anticipated Graduation : 2024*
  - **Courses:** Systematic Program Design - 100%, Models of Computation - 100%, Software Construction - 98%, Data Structures & Algorithms - 96%, Operating Systems - 95%, AI - 95%, Distributed Systems - In Progress.
  - **Awards:** Ranked 2nd in Faculty of 700 students, Sauder School of Business Scholarship.
  - **Extracurricular:** UBC BizTech Software Developer, UBC BizTech Tri-Mentorship Program.

Skills

- **Languages**: OCaml, Kotlin, Java, C++, JavaScript, TypeScript, Python, Swift, HTML/CSS.
- **Frameworks and Tools**: gRPC, Node.js, AWS, Git, MongoDB, React, GraphQL, Docker, PostgreSQL, Jest.

Experience

- **DoorDash** Toronto, ON
  Software Engineer Intern *May 2022 - Aug 2022*
  - Architected a dynamic fee disclosure tooltip by working with gRPC protocol buffers **across multiple Kotlin micro-services,** releasing the feature to **over a million users** and relieving the risk of $250M in regulatory fines.
  - Designed and implemented the migration of Quebec's pricing system onto a new modular platform, reducing engineering time to experiment on pricing configurations **from one week to 30 minutes.**
  - **Exceeded my intern responsibilities,** assisting on-call engineers by taking initiative and leading the diagnosis and solution of severe pricing display issues.
- **Luscinia** Vancouver, BC
  Software Engineer (Part-Time) *May 2021 - Aug 2021*
  - Completed coding of Hiring Service prototype **50% ahead of schedule,** leveraging Next.js to **support over 200 applications** in the following hiring season with a responsive serverless backend.
  - **Developed a REST API with Flask** (Python) to parse user metrics and dynamically respond with personalized learning resources.
  - Implemented rapid lesson creation functionality in **MongoDB** and **reduced API latency by 40% using Redis** in memory cache.

Projects

- **Reddit Clone** *Jul 2021*
  - Successfully replicated Reddit using a **Typescript, Node.js, PostgreSQL, and GraphQL** backend, supported by a **Next.js** frontend with modular React hooks.
  - Designed **custom caching** to implement responsive GraphQL mutations and seamless pagination.
  - Supported **server-side rendering** using NextJS and streamlined deployment onto Heroku using **Docker images**.
  - Leveraged Redis to implement authentication using **session tokens** and forgot password tokens.
- **LiveliHood** *Jan 2021*
  - Developed a web app that allows users to help neighbours who are more vulnerable to COVID by completing tasks such as shopping in populated grocery stores.
  - Created a **REST API with Node.js** to manage personalized tasks for each user using **JWT** for authentication and MongoDB for persistent data store.
  - Built an interactive UI using **React** and integrated the Google Maps API to visualize locations of tasks.
  - **Automated the test pipeline by leveraging Jest and Supertest** to ensure a flawless production deployment.