Jerome Rasky

jyrome.112@gmail.com
(650) 394-5091
Vancouver, Canada

Overview

Introduction

Former NASA Software Engineer with 5 years of professional work experience. Well-developed skills in systems programming and administration, front-end and back-end development, and project management.

Languages

Java • JavaScript • Go
C++17 • C11 • Rust
SQL • AMD64 Assembly

Frameworks

React • Redux • Webpack
Android • WordPress • Django

Tools

Linux • Git • Emacs
Bash • NodeJS • Mocha

Skills

Systems programming
Site relability
Backend development
Frontend development

Education

The University of British Columbia

Vancouver • 2014-Present

Bachelor of Science in Computer Science, graduating 2018

Coursework

Algorithms • Data Structures
Operating Systems

Experience

New Relic
San Francisco • Summer 2016

Frontend engineer working on the newly-launched Infrastructure product. Made significant contributes, including on the backend and on other teams, which enabled the team to meet critical milestones.

Arrayent
Redwood City • Summer 2015

Site reliability engineer managing a large deployment of production systems. Resolved issues in services with real-time availability requirements, and learned to effectively monitor large-scale infrastructure.

NASA Ames
Mountain View • 2011-2014

Full-stack developer, managing both frontend and backend codebases and production systems. Worked closely with highly-specialized users to deliver interfaces to meet their needs the most closely.

UBC CS Student Society
Vancouver • Fall 2016-Present

Vice President External, responsible for the yearly Technical Career fair. Practiced effective communication to coordinate a diverse team in the organization of a complex event.

Personal Projects

Half2
Algorithms • Data Processing • Filesystems

B-tree implementation optimized nearly to match the one in the language. Started out as a version control system, but shifted to focus more directly on data structures.

Kernel
Systems • Algorithms • Memory Management

OS kernel research project, focusing on some ideas I have about operating systems. Prompted by a desire to learn more about operating systems and hardware.

BIS2/FLX
Fuzzy Search • Optimized • Command-Line

Highly optimized fuzzy search for terminal history. Improves over the default search interface with better matching and a better UI. Has become something that I use every day as an integral part of my workflow.