

Samuel Wolfson

206.601.5615 / wolfson@uw.edu
samwolfson.com / github.com/terabyte128

Summary

Education: University of Washington, BS/MS Computer Science (expected graduation June 2019 (BS), 2020 (MS); 3.90 GPA)
Languages: Java, Ruby, JavaScript, HTML/CSS, Python, LaTeX, C, Golang.
Frameworks: Ruby on Rails, ExpressJS, Sequelize, Django, gRPC.
Environments: Git, UNIX/Linux command line, SQL (MySQL, PostgreSQL), Docker.

Relevant Coursework

Computer Programming I/II, Hardware/Software Interface, Foundations of Computing I/II, Data Structures & Parallelism, Programming Languages, Software Design & Implementation, Systems Programming, Operating Systems, Databases, Distributed Systems, Compilers.

Projects & Activities

Processing to P5.js Compiler — Summer 2018 (Java, JavaParser)

- Modified a Java parser and AST builder library to build an AST for the Processing programming language (processing.org), and wrote visitors to modify the AST and output code in the the JavaScript version of the language, P5.js (p5js.org).
- This compiler will be used by students in an introductory programming class (CSE 120, for which I have worked as a TA) so they can run their Processing code on the web without manually converting it.

Quiz Website — Summer 2018 (Ruby on Rails, PostgreSQL, HTML/CSS, JS/jQuery, JSON)

- Developed a website for students to take online quizzes and get immediate feedback about their knowledge.
- Includes support for regular expression-based answers, and specific hints based on which incorrect answer a student gives.
- Designed to be extensible, with the ability to add more question types later on based on JSON schemas.

Proof Verifications — Summer 2017 (NodeJS, ExpressJS, Sequelize, HTML/CSS, JS/jQuery)

- Worked with a UW CSE faculty member on his research, and developed a website where students can verify correctness of proofs by annotating them and selecting templates for common errors.
 - This helps students to better understand how and why they make mistakes in their own proofs.
-

Work and Volunteer Experience

Arista Networks — Software Engineering Intern, Summer 2018

- Implemented code to automatically power off servers in our test environment when they are not actively in use, leading to a measurable decrease in energy usage.
- Designed a strategy to move management code for test servers out of local user workspaces and into containerized microservices, using Docker to run the services and gRPC to facilitate communication between the clients and the services.
- This design allows the underlying implementation of the service to change without breaking outdated workspaces, as long as the interface to the service (provided by gRPC) remains consistent.
- Wrote an implementation of a service (in Golang) using my design that controls the power state of our test servers.

UW CSE Teaching Assistant (CSE 351: Hardware/Software Interface, CSE 120: Computer Science Principles), 2017-18

- Led sections and labs as well as two office hours, where I answered students' questions and explained new & novel concepts.
- Helped to develop new course materials, worksheets and staff tools; created and presented lectures.

Progressive Tech (Seattle, WA) – Computer Repair Technician & Internal Support Specialist, 2009-Present

- Developed scripts for network booting of hardware diagnostic tools. (PXE/iPXE, DHCP networking, Linux.)
- Created web-based software tools for streamlining and standardizing repair quotes. (HTML/CSS, JS/jQuery.)
- Prototyped a new order and customer management system. (Ruby on Rails, PostgreSQL.)

Canoe Island French Camp (Canoe Island, WA) – Maintenance Volunteer, Summer 2014

- Worked in a small group to maintain and improve camp facilities.
-

References: available upon request.