Dave Pagurek van Mossel

University of Waterloo Software Engineering, class of 2019

Work

Software Engineering Intern at **Google** Mountain View, California, May-Aug 2017

- Implemented live person detection and tracking for Android Things smart cameras using Tensorflow and OpenCV
- Developed a framework to use machine learning inference from TensorFlow models as a trigger for home automation rules on Android Things

Software Engineering Intern at **Remind** San Francisco, California, Sept-Dec 2016

- Designed and implemented a REST API for district management, efficiently querying the graph of districts, schools, and users
- Developed features for backend Ruby and Go payments services, plus accompanying client work in React and Redux

Software Developer Intern at **Athos** Redwood City, California, Jan-Apr 2016

- Created a C++ library for defining signal processing pipelines by parsing a JSON-based language definition into a tree of filters, allowing variable scoping and mapping over lists
- Developed infrastructure and UI features in Objective C and Swift to allow users to run through athletic training plans and receive a score calculated from garment sensor data

Software Developer Intern at **Shopify** Ottawa, Canada, May-Aug 2015

- Developed Ruby scripts to transform and load sales data from sharded MySQL databases into central MySQL and Postgres databases on Amazon Web Services
- Introduced new language constructs in the Shopify Query Language parser allowing granular querying of data in Go and Ruby

Projects

Raytracer, 2016-present

A 3D raytracing renderer written in Swift

- Implemented soft shadows, depth of field blur, refraction, subsurface scattering, and motion blur sampled probabilistically over time
- Explained the math and logic of raytracing and how subsurface scattering works on my blog for others to learn from

Frontier, 2016

A procedurally generated 3D landscape art project

- Designed landscape components as recursive fractals and rendered them in Processing and OpenGL

Scala compiler, made for CS241E, 2015

A tool written in Scala to compile a subset of Scala into MIPS instructions

- Parsed input into an AST for the Scala grammar to compile
- Implemented closures, tail recursion, type checking, and garbage collection

Open-source contributions, 2015-present

- Contributed bug fixes and features to **Radiant Player**, a Facebook Messenger **Mac client** and **CLI**, **Vim Auto-Pairs**, and **Emerald language**

About

Programmer, web developer, digital artist, and Regular Expression enthusiast

- dave@davepagurek.com
- davepagurek.com
- github.com/davepagurek
- (613) 875-4951

Skills

- Demonstrated expertise with **Ruby** and functional **JavaScript**
- Professional experience with **C++11, Java, Go, Swift, C, SQL, Git,** and **Unix**
- Highly proficient with CSS layout models and experience writing performant animations
- Passion for creative approaches to visual and algorithmic design problems

Awards

- First place in Waterloo EngHack, both fall and winter 2015
- University of Waterloo President's Scholarship, 2014
- Top 25% distinction on the Canadian Computing Competition, senior division, 2013 - 2014
- Jerry Dermer Memorial Prize in Engineering, 2014
- Ottawa-Carleton District School Board Silver Medal, 2010-2014

Leadership

- Founder and Organizer, **TerribleHack I - VI**, a hackathon for programming for its own sake rather than for a practical purpose, 2015-16
- Organizer, **Tech Retreat**, a hackathon for high school students, 2015-16