Tristan Hume

Student Developer - University of Waterloo Computer Science

🌐 thume.ca
⬛ github.com/trishume
✉ tristan@thume.ca

Work

Google
Software Engineering Intern, Summer 2017
- Worked on an open-source, heavily documented Conflict-free Replicated Data Type for text, refining it for fast and seamless editing and merging.
- Implemented multi-device text synchronization for the Fuchsia operating system including reliable merging of changes after editing offline across devices.

Jane Street Capital
Developer Intern, Fall 2016
- Wrote a js_of_ocaml-based debugging and exploration tool that reinterprets a rule matching language to find all possible paths given a set of constraints on the input.
- Implemented parsing and processing code for a low latency binary UDP market data feed with zero-allocation OCaml. Learned a lot about finance in the process.
- Developed a custom tree diffing algorithm based on minimizing a cost model using dynamic programming accelerated with an A* search for large trees.

UWaterloo HCI Lab
Undergraduate Researcher, Winter 2016
- Designed and developed a hands-free mouse alternative that combines the speed of an eye tracker and the accuracy of head tracking using MAGIC.
- Developed high accuracy low-latency audio recognition algorithms for using various mouth noises (e.g lip popping) to perform actions like clicking.
- Combined knowledge from hundreds of HCI and eye tracking papers to develop an enjoyable to use system with speed and accuracy similar to a trackpad.

Shopify (Shipping Team)
Developer Intern, Summer 2015
Helped develop Shopify Shipping. I fixed production disruptions, implemented package tracking and owned the development of the unified fulfillment and label purchase form now used by thousands of merchants every day.

Developer Internships During High School
1 month at Shopify in 2014, 2 months at Shopify in 2013, 2 months at The Eclipse Foundation in 2012, 3 weeks at Halogen in 2012.

About

I'm an enthusiastic developer who has spent the last 10 years building dozens of projects using a large variety of languages and technologies. I'm also building a solid academic groundwork through my studies as a CS student, research work, and spending lots of my spare time reading. I'm a 3rd year student with a GPA of 88% and 94% in-major.

Open Source

I've created dozens of open source projects with over 20,000 combined users (300,000 if you count web apps), all of which you can find on my Github page.

I was also the first contributor and long time top contributor to Spacemacs, a now quite popular configuration package for Emacs.

I once went on a 201 day long Github streak.

Languages Used

My strongest languages are Ruby, Rust and C++. I've written over 10,000 lines of code in each of these languages.

I also enjoy learning new languages: I've done projects in 22 different languages including Javascript, Haskell, D, Go, and OCaml.

Selected Projects

Dayder
A Rust web app for finding spurious correlations in 390,000 time series data sets. I wrote custom optimized DOM, JS Canvas rendering, caching, correlation and binary serialization code for instantly responding to queries.

The Open Turing Compiler
An LLVM based compiler for Turing as well as a Qt-based IDE and a simple drawing library.

PolyType
I built a working keyboard I designed in AutoCAD and put together with laser cut acrylic layers, Cherry MX switches, lots of soldering, and an ARM microcontroller.

SmartGaze
I reverse engineered my Eye Tribe tracker's USB protocol by scripting LLDB to capture their USB messages and implemented a glint and iris tracker on the raw image feed suitable for high accuracy eye tracking.

StashLine
An IOS app for long term personal finance simulation with 7000 users. Has a custom built UI that instantly updates a visualization of your entire life's financial future while you manipulate inputs.

Syntect
A high quality syntax highlighting library in Rust based on Sublime Text 3's grammar and theme formats. Relentlessly optimized for high performance and excellent highlighting quality. Well-documented, tested and fully-featured.