4-2050 Vine Street
Vancouver, BC, V6K 3K1
+1 (778) 980-4817
ruperthorlick@gmail.com
www.github.com/ruhatch

Rupert Horlick

I am a Computer Scientist and Software Engineer with experience in functional programming and its mathematical foundations. I would like to bring the Computer Science perspective on distributed systems of information to new fields including, but not limited to, biology, politics, and sociology.

Experience

Sep 2017 – Present
Lead Software Developer, *Money&Co*.
The company had accrued large amount of technical debt and brought me on to rebuild from scratch. Built new infrastructure using Nix for reproducible builds. Used Haskell for the server, the frontend, and the type-safe API for communicating between the two. Used GHCJS to compile the frontend targeting the web with Haskell. Had complete control over development, design decisions, and my time.

Mar – Sep 2017
Software Development Engineer, *Myrtle Software*.
Started part-time during my Masters degree and moved to full-time in June. Worked on compiling neural networks to FPGAs, using modern tools such as Haskell and Nix. Joined as one of three developers and became a leading member as the team grew. Was offered a senior position, including responsibility for the team's productivity.

Summer 2016
Research Intern, *Microsoft Research Cambridge*.
Worked under Simon Peyton-Jones on a project adding functional features to Excel. Prototyped probability distributions in cells. Our demos were presented to managers in Redmond, convincing them to use our research in the next version of Excel.

Education

2016 – 2017
MEng Computer Science, *University of Cambridge*, Distinction.
Thesis: Formalised the theory of Generalised Species in Homotopy Type Theory (HoTT) using Agda. Worked closely with leading researcher in the field, who offered me a research position to continue our work.
Modules: Category Theory, Multicore Semantics & Programming, Advanced Functional Programming, Distributed Games & Strategies, Interactive Formal Verification.

2013 – 2016
BA (Hons) Computer Science, *University of Cambridge*, 1 (81%, Rank: 8/81).
Implemented Path ORAM, a cryptographic primitive, in OCaml on MirageOS to perform search over encrypted documents. Analysed the performance and security properties in a dissertation. Presented ideas to Microsoft Research Cambridge.

2007 – 2012
High School, *St. Paul's School*.

A-Level	Computing — A*, Maths — A*, Further Maths — A*, Physics — A*
AS-Level	Chemistry — A
GCSE	10 A*s, 1 A

Other

Programming Languages
- Isabelle, Agda, Prolog
- Haskell, OCaml, Nix, Java

Tools
- Unix, git, NixOps
- Octave, LaTeX