Jason Sun's Resume

jason.sun@uwaterloo.ca	sunapi386.ca	+1 (519) 500-2969	Compiled on 08/07/2015

About Me

- *University of Waterloo* Computer Science undergraduate, graduating August 2015. Previous exchange student at *EPFL (Swiss Federal Institute of Technology at Lausanne).*
- Very driven and motivated with many hobbies (visit my blogs at *blog.sunapi386.ca*). Especially interested computer science, which was discovered after trying two different majors: physics, and business.
- Over eight years of experience using unix: Ubuntu, Arch Linux, OS X, and unix utilities like grep, less, find, fdisk, and etc.
- Experienced working in startup environments, at Velocity Residence and Velocity Garage. Enjoys technology conventions; ones I had attended were: *DEFCON (#22)*, *2600 Hope Number 9*, PennApps 2013, MHacks 2014, HackMIT 2014, and HackZurich 2014.
- Won hackathon prizes at:
  - *HackZurich 2014* at ETH Zürich, received Tamedia Digital Award (all-inclusive team trip to visit startups in Berlin, Germany) with *.GIFMeIt*: An iOS app that lets a user easily capture and share GIF images.
  - *PennApps 2013* at University of Pennsylvania, received Twilio's Communication Award ($500) with *Marmoset*: A chatbot to respond to your chosen Facebook friends without them knowing.

Work Experiences

- Software Engineer Intern *Shutterfly Inc.* in Redwood City, California

Develop functional load tests for distributed services. Design and implemented a distributed key value storage service, using technology like **Jersey** RESTful Web Services framework and Apache **Cassandra**.

- Software Developer at *Encircle Inc.* in Kitchener, Ontario

A startup in the *Velocity Garage*, worked in **java**, **coffee**, and **python**. Worked on the web and android application stacks.

- Undergrad Research Assistant at University of Waterloo in Waterloo, Ontario

Working with professor and experimented acquiring sound input from a NI myDAQ, a low-cost data acquisition device (DAQ), and performing data analysis on it using **Matlab**.

- Software Tools Developer Intern at RIM (BlackBerry) in Ottawa, Ontario

Built features to the GitLab open source project (**Ruby on Rails**). Developed a testing framework for testing website user interfaces, using the **Selenium** Java framework.

- Physics Teaching Assistant at *Wilfrid Laurier University* in Waterloo, Ontario

Developed a spectrometer reading program in python, using the **pySerial** library, and automate queries over serial port - previously you had to punch numbers on the machine.

Interesting Projects

- Real-time Operating Systems (CS 452): One project for the entire semester, which is to build a real-time microkernel and write user programs to control model trains on a track. The course is famously called the Trains course.
- Artificial Intelligence (CS 486): Worked on projects involving machine learning, learning probabilistic models, Bayesian networks, search and constraint satisfaction problems.
- Advanced Algorithms class (EPFL CS 450): A graduate course in algorithms, learned theoritical techniques and their applications to solve problems. Interesting techniques such as network flow, randomization, dynamic programming.
- Computer Graphics (EPFL CS 440): Half the course was about graphics rendering techniques, and the second half about animation techinques. Implmented rendering methods into a ray-tracer (called *Nori*), and explored animation modeling and some fluid mechanics.
- Intelligent Agent class (EPFL CS 430): Developed intelligent agents to pickup and deliver parcels in a simulated environment, with intelligent behaviours: reactive, deliberative, centralized, decentralized, and auctioning.
- Concurrency class (CS 343): Multithreaded quicksort, implementation of well known concurrency control mechanisms, such as monitors. Language is in ***uC++***, a concurrent dialect of C++, developed at University of Waterloo.
- Distributed Systems class (CS 454): Built a remote procedure call library in **C++** on top of TCP, both client and server side. Implemented the go-back-N reliable transmission protocol, over UDP using **Java**.
- Computer Security class (CS 458): Created exploits in **C** using techniques such as buffer overflow, and format strings. Implemented an intrusion detection program which parses output from tcpdump to detect spoofed packets, malicious hosts, and worms.
- Compilers class (CS 251): Built a MIPS compiler in C++, parsing a subset of C keywords and generating **MIPS assembly** code.
- Computer Architecture class (CS 450): Designed a pipelined CPU in **Verilog**, supporting 8 instructions for computer architecture class. This is sufficient to run machine code produced by the MIPS compiler, from the CS 251 compilers class.
- Personal project: Dotabuff-ripper tool is written in **Ruby** to aid the counter-hero picking in 5v5 Dota games. A scraper collects about hero winrates from Dotabuff and inserts into a ***Neo4j* graph based database**. The tool then suggests a list of potential counter-picks by simple algorithms. Meant to have a web interface, but the web component (written with **Ruby on Rails**) is only partially completed. Stopped working on this in 2013. Now (2015) working on this again for a course project for artificial intelligence class.