William Mak

Personal Data

Address	126 Majestic Drive, Markham Ontario	Email	william@wmak.io
Phone	(416)-953-7198	GitHub	wmak
Education	Honours BSc, University of Toronto, Computer Science		

Technical Skills

Languages	Python, Go, C, Shell, LaTeX, JavaScript
Software	Git, vim, Jenkins, Sublime, svn, TextMate
Operating Systems	**arch**linux, Debian, Fedora, Mac OS X, Windows
Frameworks	AngularJS, Django, Selenium, PhoneGap

Work Experience

Jun 2014 – Present

Programmer at University of Toronto

- Constructed a mobile application using JavaScript via PhoneGap and AngularJS
- Designed the user interface based on user input(ie. Director of The Hub).
- Wrote a Python script that used Twitter's API to create a blogpost for the Vice-Principal of Research, U of T Scarborough.

Sep 2012 – Dec 2013

QA Automation Engineer Kobo

- Engineered tests using Selenium Webdriver library based on the Page Object pattern.
- Experienced with utilizing the Saunter framework.
- Performed Exploratory Testing to identify and communicate defects to developers.
- Investigated failures with the system to diagnose the root cause of the issue and created defects reports on findings.
- Configured and maintained a continuous integration test suite using Jenkins.
- Participated in the Agile Scrum process.
- Critiqued and reviewed UX designs.

Personal Projects

- Created a web automation framework around selenium: selenate(github.com/wmak/selenate), with over 1000 downloads in the first 3 days of release.
- Designed an algorithm(wmak.io/t) using unicode that would be able to store Latitude and Longitude in 4 characters, accurate up to 7 decimal points.
- Participating in an open source project Hermes(github.com/hermes), a distributed unlimited redundant backup solution written in Go.
- Developed a golf swinging analysis program swingr(github.com/swingr) that through the use of OpenCV would track the head of a golf club giving a user a relative score against a "master" swing.
- Created an image analysis program iris(github.com/IrisDS) that could locate the relative positions of the capturing devices from one another using OpenCV and python.
- Developing a Go implementation of RaptorQ; "The world's most advanced forward error correction (FEC) code for data networks" go-raptor(github.com/hermes/go-raptor)