Ian Lai

ianlaiuw@gmail.com
https://ianlai.ca
linkedin.com/in/laiian
iangitscode

Education

University of Waterloo
Bachelor of Computer Science, Economics Minor - 4A Term

Skills

Programming
- HTML5 / CSS3
- Javascript ES6
- Typescript
- Angular 7
- NodeJS
- C++
- C
- Python 3
- BASH
- SQL

Development Tools
- Mercurial
- Git
- IntelliJ Idea
- Sublime Text
- Vi
- Linux

Interests


Dragonboat


Taekwondo


Musicals


Chess


Trombone


Piano

Employment

Visier, Software Developer — September 2017 to April 2018
- Developed methods to take information and present it in a meaningful manner
- Added the ability for an admin user to upload their company's logo and display it throughout the application
- Implemented a toggleable high contrast mode to aid with visibility on projectors
- Assisted with a complete visual overhaul of the entire application for HR Tech 2017
- Rigorously wrote unit tests for components and services created
- Gained extensive experience with Mercurial, Angular, RxJS, and NodeJS, and was exposed to Docker and Scala

Humber Institute of Technology, Math Learning Support — January 2017 to April 2017
- Supported students with their Math and Computer Programming courses
- Created a web application to organize and provide auditory reminders of class visits
- Conducted an internal analysis on the Math Centre's attendance and return rates
- Assisted in the migration of the Headstart program from an outdated platform to D2L

Projects

UWaterloo People Counter — 2018
⁕ Used UWaterloo public API to estimate the number of people in each building on campus
⁕ Based on the assumption that each person has a device connected to that building's wireless access point
⁕ Frontend written in Angular, backend written in Python, using Flask to connect to the UWaterloo API and Psycopg2 to interact with a PostgreSQL database

Not Cards Against Humanity — 2018
⁕ An online Cards Against Humanity clone created with Angular and NodeJS
⁕ Supported running multiple games simultaneously with an arbitrary amount of players in each game
⁕ Featured intuitive controls optimized for mobile play

OS/161 Kernel — 2018
⁕ A small toy kernel built in C
⁕ Developed on top of Harvard's OS/161 for the MIPS architecture
⁕ Implemented crucial kernel components such as synchronization primitives, system calls, support for multiple processes, TLBs, and page tables

Juggle — 2017
⁕ Implemented a basic physics engine to emulate the Facebook Messenger soccer juggling game
⁕ Leveraged CSS to minimize Javascript usage and achieve a smooth playing experience
⁕ Featured a game speed slider to modify difficulty

GraphHax — 2017
⁕ An in-browser graph visualization tool
⁕ Developed a toolbox interface to access manipulation functions
⁕ Built using Canvas, Javascript, and JQuery

Quadris — 2016
⁕ A Tetris command line based game created with C++
⁕ Featured customizable block shapes, multiple levels, and an expandable playing field as game options