Edmund Loo

http://edmundloo.github.io/
https://www.linkedin.com/in/edmundloo/
(626) 423-9065 • edmundloo@outlook.com

EDUCATION

Bachelor of Science in Computer Science and Engineering **Expected December 2016**
University of California, Irvine **GPA: 3.2/4.0**

TECHNICAL QUALITIES

- Fluent in C++, Python, and PHP; Experience with JavaScript, React, Java, Flask, C, C#, OpenGL, Qt Libraries, Arduino
- Experience working with Linux, Mercurial, Perforce, Git, SQL and NoSQL databases, and large scale systems
- Knowledge in machine learning, algorithms, data structures, object oriented programming, networking, operating systems, web engineering, software engineering practices, signal processing, logic design, and embedded programming
- Strong enthusiasm and capabilities in adapting, learning, problem solving, research, and software development

CAREER HISTORY

Facebook, ***Software Engineering Intern***, *Menlo Park Headquarters* **June 2016 – September 2016**

- Worked on the Facebook Privacy and Trust Team to help users understand and control who can access their information and content and to make Facebook a product that people can trust and feel good about
- Heavy projects across parts of the Facebook tech stack including PHP, JavaScript, React, and Mercurial

Xumo, ***Software Engineering Intern***, *Irvine Headquarters* **February 2016 – May 2016**

- Pushed code fixing major bugs in a widely used caption conversion script within the first week of the internship
- Constructed MySQL queries in a plugin for Nagios monitoring software to monitor critical columns in a MySQL database
- Built a program that fixed a bug with an open source JavaScript framework, FFmpeg, where HLS durations are often incorrect; this program utilized a multitude of methods to calculate the duration of an HLS streams to find the most accurate result
- Implemented a Python Flask web application that allowed editing of a server-sided configuration file from the client, which includes strict form verification, query for configuration file format from a MySQL database, JSON generation for transporting the configuration file, and cross-application communication APIs

Arista Networks, ***Software Engineering Intern***, *Santa Clara Headquarters* **June 2015 – September 2015**

- Developed software for multi-chassis link aggregation (MLAG) on Arista's Linux-based Extensible Operating System
- Optimized a commonly called MLAG failure recovery process to be significantly faster for shipping on future Arista switches
- Adapted to Arista's development environment which uses Perforce, a full Linux command line interface, and many Arista tools

Computer Learning Center, ***Computer Science Tutor***, *Pasadena City College* **January 2014 – August 2014**

- Pioneered the Computer Science Tutoring Program as a facilitator and a first generation tutor
- Amplified progress and improvement in the Computer Science Department by collaborating with faculty and other students through localized educational research and peer surveying
- Generated success and retention in computer science courses, improvement was seen in over 90% of students and drop rates decreased from an average of 40% in some classes to under 10%

PROJECTS

CalPlug, ***Back-End, Networking, Machine Learning, and Embedded Systems*** **October 2014 – March 2016**

- Assembled functions and controls in C++ for an Arduino and an Adafruit Wi-Fi shield that allowed the Arduino to find and connect to a Wi-Fi network and continuously send several bytes of data to a local Python server through a TCP socket
- Built a local networking server in Python that constantly listens for data from a TCP socket connected to the Arduino, parses and creates a JSON object from the data, and sends the JSON object to both a local storage file and a web-hosted MongoDB
- Implemented entire back-end including APIs to access and modify information, all communication, and all application logic
- Designed and implemented a linear regression machine learning model that takes power and energy readings as data points for training and then identifies newly plugged in devices based on data points previously used to train the device

UAV Forge Networking Team, ***TCP Networking in a Graph*** **October 2014 – January 2015**

- Team-based, ongoing, research and design project to fabricate a military specification unmanned aerial vehicle from scratch
- Conducted the C++ programming members within the networking team by researching for and directing the members in the right direction, discussing specifications with team leads, and translating required specifications into implementation details
- Designed a mesh network simulation using a graph structure where edges were implemented as TCP sockets and nodes as I/O devices, the implementation included a routing table and the ability for the routing table to regenerate if a node goes down