Alvin Ko

(925) 913-0305 / ko.alvin.1748@gmail.com

OBJECTIVE: Graduate student at San Jose State University seeking position of Software Engineering. Areas of experience include working full stack web development at Cisco, software QA testing at Sequent Software, and practical projects in the master's program

EDUCATION

MS Software Engineering — Graduation: Summer 2018
San Jose State University – San Jose, CA — GPA: 3.2

BS Computer Science — Graduation: December 2015
San Jose State University – San Jose, CA

TECHNICAL SKILLS: Python, Java, C, C++, MySQL, Web Programming languages (HTML, CSS, PHP, JavaScript, jQuery), LAMP/WAMP, Selenium Testing Suite, JUnit, REST, verbal and technical writing skills

RELEVANT COURSEWORK: Enterprise Distributed Systems, Web & Data Mining, Software Quality Assurance, Cloud Technologies, Database Management Systems, Data Structures and Algorithms, Programming Paradigms, Object Oriented Programming, Operating Systems, Virtualization

EXPERIENCE

Software Engineer — October 2015 – February 2017
Cisco Systems, Inc., San Jose, CA

- Managed an internal Bootstrap-framework website that provided information on the Cisco UCS server product line to assist technical marketing engineers
- Integrated knowledge of server-side web development and design to improve the layout, usability, and structure of the internal website built on Joomla and WordPress
- Provided guidance and assistance to an intern who worked on a data analyzing project based on aggregated data from the internal website

Software Developer Engineer in Test Intern — July 2015 – August 2015
Sequent Software, San Jose, CA

- Participated in fortnightly Agile end-of-sprint meetings to review assignments and daily scrum calls to actively communicate "what I did, what do I plan to do, and what got in my way" for advice from senior level engineers
- Utilized Selenium in a Java program to validate and test websites on a CI server. Primarily ran front-end headless and GUI tests
- Incorporated issue tracking software, Jira, into workflow by documenting and tracking bugs that I discovered when testing programs

NOTABLE PROJECTS

Python RocksDB Replication, SJSU November 2017

- This class project is based off of the Pinterest Rocksplicator, a real-time C++ replicator, but with 2 key differences; this class project used a GRPC Python server instead of Thrift server and did not have a cluster management feature
- Utilized an Ubuntu Docker image to execute this project due to issues running RocksDB (project requirement) on macOS. Docker allowed underlying host, macOS, to be abstracted from RocksDB which prevented dependency issues
- Running Docker allowed the TA to streamline the grading process due to a uniform project environment. This also prevented errors in the grading process.
- Incorporated gRPC ('g' Remote Procedure Call) to emulate HTTP commands, GET and POST, on 1 master server process and 3 slave server processes on 4 separate terminal windows. When the master server is being written to, the slave databases will begin replication asynchronously

MLP Machine Learning Analysis on SF Housing Prices, SJSU May 2017

- Built a housing price prediction model that takes various factors (neighborhood school ranks, public spending proportions, population, crime rates, and number of parks) into consideration
- Ran a multilayer perceptron (MLP) machine learning model to analyze the SF dataset. MLP examined the information we collected and let us know the margin of error in how well it can predict housing prices based on parameters above. Results indicated that housing prices are not influenced by the parameters above.
- Generated decision trees showed that potential real estate buyers tend to look at school ranks and safety above all else.
- Our approach to consolidate different datasets using zip codes as the unifying factor was not optimal. Thus, future models should aim to use a better parameter (i.e. street addresses) to perform this analysis

Amazon Review Classification (kNN algorithm), SJSU March 2017

- Implemented a k-Nearest Neighbor Classifier to predict the sentiment for a large set of baby product reviews. A practical application in e-commerce applications is to infer sentiment (or polarity) from free form review text submitted for a range of products
- Provided testing and training data was stemmed and generated into a sparse matrix. This matrix was scaled with Inverse Document Frequency (IDF) and normalized so that it could compute cosine similarity of the testing and training matrix
- The nearest neighbor based on k-parameter is found and exported into a text file to be compared to the actual sentiment for accuracy rating

Server Backup, Cisco Systems, Inc. January 2016

- Deployed a bash script containing rsync and mysqldump commands into a Cron job to automate daily backups on the production server.
- Rsync backed up the files on the server while mysqldump backed up the server's database files. The bash script utilized SSH passwordless credentials every time the backup job occurred. The backups are zipped for convenience and stored on another remote server
- This project was a major upgrade for my team because, previously, the production server was manually backed up on a weekly basis by another engineer. By automating the production server's backups, I reduced the margin of error from this tedious and routine process.

EVENTS: Microsoft App Madness Challenge, Corona SDK Demo, IT Cloud Computing Conference