**Jun (Chris) Pak**

430 Kelton Ave. Los Angeles, CA 90024 | 818.521.7654 | jpak1996@g.ucla.edu linkedin.com/in/chris-pak | github.com/jpak1996

Work Experience

**Systems Development Engineer Intern** - *Workday (Jun. 2017 – present)*

* Developed custom Slack application integrating with ServiceNow to create simpler user experience in filling out request items/service tickets in IT (Node.js, Python, RabbitMQ)
* Built functionalities to validate and blacklist IP addresses through Palo Alto Networks while logging the records in a custom XML Database (C#, XML)
* Wrote wrapper classes for PagerDuty to be utilized in an API Proxy (Java, Mulesoft ESB)
* Accumulated knowledge: API, web services, REST, JSON/XML

**IT Student Consultant** - *UCLA Luskin IT Help Desk (Jan. 2017 – present)*

* Assisted graduate students with remote login, network issues, and email
* Managed Windows/Mac hardware and software installation for lab users
* Helped troubleshoot professors’ laptops before/during class presentations to ensure a smooth user experience

**Content Editor** - *UCLA Course Reader Solutions (Sept. 2016 – Jan. 2017)*

* Managed the digital database of 14,000+ course readers and kept record of old/new requisitions
* Assisted walk-in requisitions from over 300 UCLA professors in a fast-paced work environment
* Edited/proofread readings to be uploaded for online purchase adhering to professors’ deadlines
* Contacted publishers such as Harvard Business and Penguin Random House for copyright permissions

Education

**University of California, Los Angeles** *Expected Graduation: Dec. 2018*

* Bachelor’s in Computer Science, Class of 2018
* 3.22 GPA across core Computer Science classes
* Member of ACM, a community dedicated to running hacking events that taught students how to use JavaScript to build a spell-checking program, and how to build their own website
* Actively help students by participating in online Piazza discussions regarding current class projects and homework

Relevant Coursework

**CS 31**: C/C++, pointers, arrays, linked lists:

* Built a quick word-search program using vectors, later optimized by instead using a binary search tree

**CS 32**: Algorithms and data structures:

* Coded a fully functional game by implementing operator overloading and interpreting keyboard inputs for a simple user interface

**CS 33**: Assembly language, I/O programming, computer architecture and memory management:

* Used GDB to successfully stack-smash an open-source webserver

**CS 35 Lab**: Open-source software tools (Git, Emacs/Vim, Python) and concurrent/parallel code:

* Parallelized a simple ray-tracing program to speed it up by a factor of ~10
* Used GNU Privacy Guard to defend against remote login from another team using asymmetric cryptography

**CS 111**: Operating systems, memory virtualization, scheduling/synchronization

* Improved shell scripting commands by implementing optional flags and file modes using multiple parent/child processes

**CS 118**: Computer networks, including server-side/client-side interactions through TCP/UDP connections

* Built a simple router to redirect raw Ethernet frames as well as ARP/IPv4 packets with a fully functional routing table