

# Jerry Lung

jzlung@gmail.com | (661) 678-3966 | linkedin.com/in/jerrylung | github.com/jzlung

JavaScript, React, HTML, CSS, Java, Python, Unix, Git & SVN

## EDUCATION

---

**University of California, Berkeley** B.A. Computer Science, May 2015

## WORK EXPERIENCE

---

### **IXL Learning**

Aug 2015 – Present

#### Software Engineer

- Lead development and coordinated cross-team (PM, QA) communication on release-blocking **React** UI components in recent milestones to successful deliveries
- Mentored and trained over 30 new hires in their first front-end **React** projects
- Redesigned our Front-End Engineer hiring process to grow our team to 3x its size
- Modernized our technology stack by upgrading our **JavaScript** build (**Gulp + NodeJS**) to **ES6** aka **ES2015**
- Developed a **JavaScript** controller integral to the release of multiple new drag-and-drop components
- Adapted quickly to resolve over 40 bug cases within 2 months to help my team release on schedule

### **NICE Systems**

June 2014 – Aug 2014

#### Software Engineering Intern

- Boosted team's workflow by developing internal infrastructure tools to monitor application health levels and automate tasks (**Python, Bash**)
- Shipped a **Java** developer tool to assist with dependency management when updating software
- Adapted to an **Agile test-driven** environment using industry technologies such as **Jenkins**

### **UC Berkeley Electrical Engineering & Computer Science**

Aug 2013 – May 2015

#### CS162, CS61C Course Reader, CS61B Lab Assistant

- Developed for the grading infrastructure to manage hundreds of grades (**see Octobear below**)
- Facilitated students' understanding of Operating Systems, Data Structures, Algorithms
- Enhanced skills in understanding others' code through debugging C code in a class of 500 students

## PROJECTS

---

### **Octobear** (Sp. 2015) – UC Berkeley EECS Instructional Staff

- Team developer for new Linux grading infrastructure for EECS courses.
- Contributed core algorithm for automatically parsing grade spreadsheets and entering into database.
- Shipped interactive **Python** scripts for grade entering and syncing to **MongoDB** database, **Flask** webapp interface for students to monitor **Jenkins** autograder build queue and grades, **CRON** tasks.

### **Music Machine** (HackJam 2014) - [musicmachine.herokuapp.com](http://musicmachine.herokuapp.com)

Project lead and developer on MEAN-stack web application for users to quickly query and download music (**Express.js, Node.js, MongoDB**).