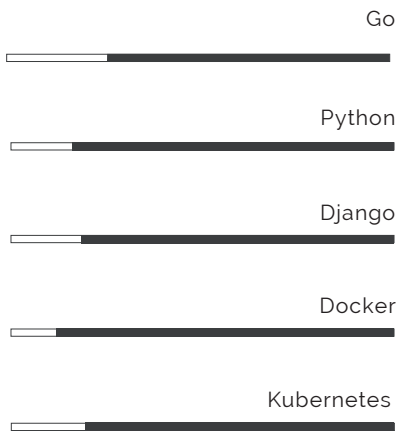


# JAKE CHONG

## SOFTWARE ENGINEER

### SKILLS



### EDUCATION

**B.S. EECS**  
UC Berkeley, Class of 2018  
Major: Electrical Engineering /  
Computer Science

### COURSES

- Data Structures
- Computer Architecture
- Networks
- Databases
- Operating Systems
- Machine Learning

### CONTACT

Email: [jakechong@berkeley.edu](mailto:jakechong@berkeley.edu)  
Phone: (213) 800-2288

Portfolio: [jakechong.github.io](https://jakechong.github.io)  
LinkedIn: [jaekyungchong](#)  
Github: [jakechong](#)

### EXPERIENCE

**RIDECCELL** AUG 2018 - Present  
Software Engineer

- Designed and developed a hardware abstraction layer to facilitate communication between Ridecell's backend and hardware vendor APIs, which allows Ridecell backend to not care about which vendor it communicates with
- Developed a state machine to keep track of rental state of vehicles and vehicle data
- Created multiple microservices from scratch

**RIDECCELL** JUN 2018 - AUG 2018  
Software Engineer Intern

- Wrote script on mock car server to simulate moving cars in QA environment
- Maintained monolith codebase and fixed bugs in production

**LOGITECH** JUN 2017 - AUG 2017  
Software Test Automation Engineer Intern

- Developed test suite using Python unittest framework to automate testing for mouse driver software
- Specifically tested the internal API, which includes methods to retrieve data about mouse settings

**CAL PERFORMANCES** JUN 2016 - SEP 2016  
Web Developer

- Maintained Cal Performances website, adding upcoming events and managing user accounts
- Worked with PHP to automate updates and created a JSON database to manage all events
- Used PHP server-side scripting to dynamically generate web pages of performances

### PROJECTS

**PARKPAL**

- Built a parking sensor with Arduino that informs any user if their parking space has been taken
- Created a web app in Flask that uses the Google Maps API to show the location of the parking spot
- Used data visualization with Highcharts to display times of the day the parking space is occupied