

# PAVEL YAKOVLEV

(707) · 971 · 9336 ◇ yvpavel@gmail.com  
pauilyakovlev.com ◇ github.com/pauilyakovlev

## EDUCATION

---

**University of California, Santa Cruz**  
Bachelor's in Computer Science  
**Santa Rosa Junior College**

*September 2018 - June 2020*  
*September 2015 - May 2018*

## EXPERIENCE

---

**UCSC Long Marine Lab**  
*Software Engineer*

January 2019 - Present  
*Santa Cruz, CA*

- Developed React geospatial data analysis tool to be used by lab technicians.
- Led the development of data management process and workflow.
- Assisted in planning and development with software engineering best practices. (test-driven-development, Agile-Scrum process, version control)

**Lawrence Livermore National Lab**  
*Computing Intern*

June 2019 - September 2019  
*Livermore, CA*

- Worked on automated testing and software integration framework for large scientific facility.
- Worked on regression testing the software that runs the National Ignition Facility with JUnit and Jenkins.
- Development focused on industry-standard continuous integration and test-driven development practices.

**remote.it**  
*Software Engineering Intern*

May 2018 - June 2019  
*Palo Alto, CA*

- Built an administrative web dashboard with React.
- Worked with Terraform + Packer for AWS EC2 AMI deployment and testing.
- Created a library of user-end Python and Shell scripts for bulk device management.
- Worked remotely from September 2018 through the end of the internship.

## PROJECTS

---

**Pollen Planter (CruzHacks '20 Winner)**

[devpost.com/software/pollenplanter](https://devpost.com/software/pollenplanter)

- Mobile app that helps to support local pollinator(bees,butterflies) communities.
- Built with React Native + Firebase Firestore.
- Worked in team of 4 - took the 1st place prize for Best Earth Hack.

**NetGuard (Treehacks '19)**

[devpost.com/software/netguard](https://devpost.com/software/netguard)

- Alerts owner of IoT device of cyber attack and locks down the device.
- Written in Python, runs on Raspberry Pi, SMS alerts achieved with Twilio API.
- Worked with another team member to write the listener and API request component.