

# Luke LaFontaine

805 813-3131 luke.lafontaine@gmail.com www.lukelafontaine.com

## **EDUCATION**

California State University, Channel Islands (2012-2016)

- B.S., Computer Science; Minor: Mathematics; GPA: 3.4
- Related Coursework: Object-Oriented Programming, Data Structures, Operating Systems, Unix Programming, Compilers, Calculus II, Discrete Math, and Linear Algebra

## **EXPERIENCE**

Software Engineering Intern, NASA's Jet Propulsion Lab (05/2015 - Present)

- Designed and built a search autocomplete feature for JPL's internal search engine
- Built an alternate document ranking system for JPL's internal search engine
- Developed multiple features for Purator, a search curation application (see projects section)

Computer Science Tutor, California State University, Channel Islands (01/2015 - 05/2015)

- Tutored students in their lower division CS classes including Introduction to Programming, Object-Oriented Programming, and Data Structures & Program Design

Software Engineering Intern, NASA's Jet Propulsion Lab (06/2014 - 08/2014)

- Extended an internal file sharing application to support bulk uploads using Bash and Python
- Automated functional testing on a quality assurance application using Selenium and Python

## **LANGUAGES & TECHNOLOGIES**

- Skilled with Java, JavaScript, C, and Python
- Adept at collaborating using Git and working in an Agile environment
- Experience with Elasticsearch, Flask, Docker, Kubernetes, and AWS

## **PROJECTS**

Alternate Search Ranking (2016)

Designed and implemented an alternate ranking system for JPL's internal search engine. Adapts the ranking based on patterns in user activity gathered from click logs. It is written in Python and uses Neo4j.

Purator (2015)

Purator is a web application used for curating the internal search engine at JPL. Implemented features to improve the search experience by removing dead links, modifying the weight between keywords and URLs, and deleting sensitive search results. The back end is written using Node.js, mongoDB, and Elasticsearch. The front end utilizes Angular.js.

ciLisp (2015)

Wrote a Lisp interpreter that employs static scoping, a basic type system, variable declarations, conditional statements, a math library, and basic I/O. It is written in C.

## **ACTIVITIES & ACHIEVEMENTS**

- President of the CSUCI Computer Science Club
- Placed 2nd in the CSUCI Programming Guru Competition
- 8 out of 10 gold medals in academic decathlon (2012)
- Enrolled in public speaking class since the age of 6 (16 years)