

Derek M Frank

Software Engineer

Email: resume@derekmfrank.com

GitHub: [@defrank](https://github.com/defrank)

Website: <https://derekmfrank.com>

Location: [West Los Angeles, CA](#)



SUMMARY

Detail-oriented software engineer lead with experience in analysis, design, development, testing, and implementation of microservice architectures. While I do have a full-stack background, my focus is in distributed systems and data engineering.

At a more personal level, I am passionate of my work, excited about technology, always looking to take initiative, and hold myself to high standards. The best part of my career choice is the constant learning.

My outside interests include: raising my dogs, lifting weights, Jiu-Jitsu, reading comics/manga, listening to audiobooks, snowboarding, and learning new things.

SKILLS

Languages

Expert/Advanced:

Python, Javascript (Vanilla)

Proficient:

C, C++, Java, Common Lisp, SQL, Bash, Zsh

Prior experience:

Haskell, Prolog, Scheme, OCaml, Perl, Lua

Libraries/Frameworks

Expert/Advanced:

Pytest, Falcon, Flask, BottlePy, Django, Dramatiq, JQuery

Proficient:

Apache Beam, redis-py, PyMongo, uMongo, Selenium, PhantomJS

Technologies

Expert/Advanced:

GCP, Git, Docker, Kubernetes, HTML5, CSS3, JSON, XML, REST

Proficient:

AWS, Helm, Apache Beam, Redis, PostgreSQL, MySQL, SQLite, MongoDB, ArangoDB, Unicorn, Nginx, Apache

Operating Systems

Expert/Advanced:

Linux, macOS, BSD/Unix, Android

Proficient:

iOS, Windows

EDUCATION

Bachelor of Science Degree, Computer Science

[University of California, Santa Cruz](#)

2008 - 2012 | Santa Cruz, CA

EXPERIENCE

Software Engineer Lead

FanAI

Full-time | 2017 - Present | Santa Monica, CA

Lead duties are focused on meeting the backend's product roadmap. This includes sprint planning, unblocking teammates, improving rolling velocity, mentoring new hires, and documenting best practices and processes.

Pioneered the microservices architecture that powers the product API and data ETL.

Designed, implemented, and deployed data pipelines in Google Kubernetes Engine of which the highest impact was an ETL to ingest about one terabyte of Twitter followers daily.

Improved distributed systems stability and scalability by replacing big data services with Google products such as RabbitMQ with Pub/Sub, MongoDB with Big Query, etc.

Worked with Distributed Actor Framework, Dramatiq (like Celery), to asynchronously process jobs communicated over a message broker.

Google Cloud Platform Kubernetes Helm Docker

Docker Compose Python 3.8 Pytest Falcon

Hug Dramatiq Apache Beam Redis PostgreSQL

ArangoDB MongoDB RabbitMQ Git Linux

Embedded Systems Test Engineer

Medtronic MiniMed

Contract | 2013 - 2014 | Northridge, CA

Designed, developed, validated, and documented iOS test automation framework and API in Javascript, Python, and Qt4.

Designed, developed, tested, and validated tools using Python to automate testing and interacting with Diabetes-related embedded devices.

Analyzed requirements, designed test cases, created and debugged automated test scripts, and documented work per Medical Standard IEC62304.

Wrote intended use validation test plans based on product requirements, executed validation tests, and documented results to ensure product meets intended use.

Python 2.7 Javascript

Programmer

Star Labs LTD - Readz

Full-time | 2014 - 2017 | Woodland Hills, CA

Add features and fix bugs for both front-end and back-end of the product. This includes, but is not limited to: Shopify integration, custom copy/paste of elements, analytics tracking via Google-defined sessions.

Full ownership of internal build server used by QA, which behaved like a continuous integration server.

Full ownership of Akamai integration and automation to sync Readz and client domains via Akamai's Property Manager API.

Responsible for full releases of new software on AWS via ELB, EC2, RDS, and S3.

Improved pre-release bug detection of customer content by building an application with PhantomJS2 to compare production content with the current development version. This reduced QA time spent verifying hundreds of customer websites every sprint (i.e., two weeks).

Python 2.7 Django 1.8 Javascript HTML5 CSS3

MySQL Apache 2 Git AWS Akamai BottlePy

SQLite 3 PhantomJS Bash Linux