

Derrick Lee

CONTACT INFORMATION	Cupertino, CA dlee3@scu.edu — 408-823-7288 — github.com/drkleee3 — linkedin.com/in/drkleee — drkleee.me
EDUCATION	B.S., Computer Science Santa Clara University June 2020 (expected) Relevant Coursework: <ul style="list-style-type: none">- Object Oriented Programming (C++)- Data Structures (C++)- Operating Systems (C/Rust)- Theory of Automata and Formal Languages (in progress)
EXPERIENCE	Open Source Developer January 2018 - Present <ul style="list-style-type: none">- Fixed incomplete features, improved and clarified code examples and documentation for serenity, a Rust library for interacting with the chat and VoIP platform Discord's API.- Implemented bug fixes and security fixes to open source Node.js chat bot LuckyBot.
RELEVANT SKILLS	Languages <ul style="list-style-type: none">- Rust, C, C++, JavaScript, TypeScript, Python, HTML5, CSS, ARM Assembly, SQL, Bash Related Technologies <ul style="list-style-type: none">- Git, Node.js, GraphQL, PostgreSQL, React, Koa, Apollo, Express, Next.js, Vue.js
PROJECTS	sushii-bot (14,000+ lines of code) December 2017 – Present <ul style="list-style-type: none">- Chat bot for Discord with a ranking system, activity tracker, moderation tools and more with over 64,000 total users.- Written in Rust with a PostgreSQL database, diesel-rs, and connection pooling with r2d2-diesel.- Paired a website with user leaderboards and statistics made with Next.js, React, Koa, Apollo server and client for GraphQL endpoints, and Join Monster for batch data fetching. Operating System Simulations (5,000+ lines of code) April 2018 – June 2018 <ul style="list-style-type: none">- Runs sequential and random disk reads with C, determines time differences and possible causes based on both physical and OS aspects. Programs and set up executed with Bash scripts.- Multi threaded simulation written in Rust of different memory page replacement algorithms with given page requests and a range of memory sizes. Data visualized with plots made in R.- Benchmarks in Rust to determine the overhead of synchronization primitives (Mutex) and lock contention. vlive-rs (1,000+ lines of code) April 2018 – May 2018 <ul style="list-style-type: none">- Rust library for livestreaming platform VLive's API to fetch channel and video data.- Implemented on the asynchronous Hyper client with Futures as well as Reqwest for a synchronous API. mp3-fft July 2017 <ul style="list-style-type: none">- Web application that takes mp3 files and recommends 100+ headphones based on price, form factor, and music sound signature (bass-heavy, neutral, mid-forward, v-shaped).- Uses the Fourier Transform and Welch's method to classify sound signature.- Written in Python with a Flask microframework back-end along with Vue.js for front-end.