John Le

San Jose, CA || Berkeley, CA

☎ 669-285-7651 ✉ johnle@berkeley.edu johnthanhle.github.io github.com/johnthanhle linkedin.com/in/johnle-cs/

Education

University of California, Berkeley

Bachelor of Arts in Computer Science

Relevant Coursework: Algorithms, Artificial Intelligence, Computer Architecture, Computer Programs, Computer Security, Database Systems, Data Structures, Data Science Foundations, Information Devices & Systems I & II, Discrete Mathematics & Probability Theory, Operating Systems, Data Science Principles & Techniques, Machine Learning, Network Architecture, Probability for Data Science

Work Experience

Rippling **February 2023 – Present**

Software Engineer *San Francisco, CA*

- Work on the Risk Engineering team

Rippling **May 2022 – August 2022**

Software Engineer Intern *San Francisco, CA*

- Worked on the Apps Platform team that integrates third party applications and software onto Rippling to allow users to seamlessly interact with them on a single platform
- Primarily focused on OAuth 2.0 authorization protocol integration for third party applications along with optimizing aggregation of third party application data/metrics to develop scalable data import infrastructure
- Exposed critical customer-facing issues where new app integration features are disabled for thousands of users due to lack of access token scopes and created infrastructure that notifies users and fetches updated token scopes to enable the new integration features
- Launched a new initiative to standardize app integrations to the Rippling platform to engender uniformity and ensure critical features are always usable to customers

Amazon **May 2021 – August 2021**

Software Development Engineer Intern *Seattle, WA*

- Worked in the Profit Intelligence Organization that keeps track of profitability across all Amazon Marketplace shipments through the calculation of various metrics using streaming data pipelines that utilizes services like AWS KDA and AWS Redshift
- Created a full stack application using AWS Lambda, AWS API Gateway, and other internal tools that translates various metrics and business rules used in the streaming data pipelines from Amazon Ion format to human readable text and visualizes it on a user interface built with React and Amazon frameworks so that users can view a breakdown of various business components with their metrics without having to understand the codebase

Shopstack (YC W20) **January 2021 – February 2021**

Software Engineer Intern *Remote*

- Interned at YCombinator startup Shopstack (previously called Stryve and Circle), which has since rebranded as Quill, developing and testing mobile applications using GraphQL, Google APIs, MERN stack, and other technologies and frameworks
- Developed User Interfaces with React Native for iOS and Android and build additional backend logic with GraphQL and MongoDB integration to handle both synchronous and asynchronous requests

Projects

Badminton Projects | *Python, JavaScript, HTML/CSS, Express.js, Node.js, React*

- Developed various full-stack web applications for use by Cal Badminton and other collegiate badminton clubs
- Created a badminton queuing system that allows players to sign up on a queue during open gym sessions and sends notifications when it is their turn with built-in admin page to manage players
- Designed a tournament schedule web application that uses Cal Badminton's own tournament match scheduling algorithm and maps it the correct match metadata (event, time, players, scores) on Tournament Software to display on a searchable and filterable grid. Currently used by UC Berkeley and UC Davis for their bi-annual badminton tournaments

Pintos Operating System | *C*

- Designed and implemented Pintos to support various features of a basic operating system
- Features included spawning and waiting on processes along with various other system calls, user program multithreading, fair scheduling and priority scheduling with support for priority donation, and an extensible filesystem that implements directories/subdirectories and resizable files using the Berkeley Fast File System design

Technical Skills

Programming Languages: Java, C, Python, SQL, Scheme, JavaScript, TypeScript, HTML/CSS, Golang, RISC-V Assembly
Tools: Git, Flask, React/React Native, NumPy, Node.js, Express.js, MongoDB, GraphQL, LaTeX, Heroku