

ALEXANDER HOANG

San Jose, CA — Davis, CA

☎ 408-480-0068 ✉ avhhoang@ucdavis.edu 👤 alexhoangs.github.io  linkedin.com/in/alexhoangs/  github.com/AlexHoangs

Education

University of California, Davis

Expected: June 2024

Bachelor of Science in Computer Science, Minor in Technology Management GPA: 3.50

Relevant Coursework: Algorithms, Discrete Math, Data Structures, Machine Dependent Programming, Algorithm Design and Analysis, Object Orientated Programming, Web Programming, Computer Architecture, Probability and Statistics, Operating Systems and System Programming, Bio-Informatics, and Combinatorics

In Progress: Computer Vision and Algorithm Design and Analysis

Technical Skills

Programming Languages: Python, C++, C, Kotlin, HTML/CSS, JavaScript, SQL/SQLite, RISC-V Assembly, System Verilog, x86 Assembly, LaTeX, R, Perl

Developer Tools: PyGame, Git, ReactJS, Node.js, Express.js, ModelSim, Quartus 2

Knowledge of: Object Orientated Programming, Android Development, Full Stack Development, Web Development

Personal: Excellent in Leadership, Collaboration, Problem-Solving, and Cross-Functional. Strong Communication and Interpersonal Skills, Passionate, Self-Motivated, Organized, and Adaptive Learner

Work Experience

Software Engineer Intern

June 2023 – September 2023

Intel Corporation

Santa Clara, CA

- Summer 2023

Lead Software Developer

January 2023 – Present

CodeLab

Davis, CA

- **Lead** team of developers/designers to **collaborate** with client, **producing** industry level tools to **gather user data**
- Utilized **React** and **Javascript** to **develop** key features on client's website, **increasing traction** by **10%**

Math and Robotics Intern

July 2022 – October 2022

C-STEM Robotics

Davis, CA

- **Designed** programming, math, and robotics exercises for the C-STEM curriculum using **C** and **C++**.
- **Collaborated** with team members to **develop, review, and debug** coding activities for their curriculum.

Front-End Development Intern

June 2022 – August 2022

The FarmLink Project

Davis, CA

- **Created** engaging and **responsive user interactive** web pages utilizing **HTML/CSS** for their website.
- **Analyzed** social media **platform data** and **collaborated** with analytics media-core team to increase user engagement.

Projects

User Thread Library | C

Private Repository

- **User-level thread library** for **linux** that provides an **interface** to **simulate multi-threading**.
- Implemented **User-Thread, Semaphore, and Preemption API's** to allow users to create, run, and **process threads concurrently** in a **round-robin** fashion while limiting the number of threads that are allowed to run **concurrently**
- Installed a **virtual timer** to **interrupt** and prevent the current thread from **starving other** threads by **yielding**

Local Food Truck Reviews | Kotlin, Android Studio

github.com/AlexHoangs/FoodTruckReviews

- Created user interactive **Android application** by fetching data from UC Davis food trucks **web API**.
- Implemented **authentication** with **back-end server** using **ID tokens** and **Google's login API**.
- Allowed **authenticated** users to **view/post** reviews about each food truck after logging into their Google account.
- Designed tab layout to switch between menu and reviews and **cached data** using **Room database** for offline use.

Discord Bot: Calorie Tracker | JavaScript, SQLite, Node.js

github.com/AlexHoangs/DiscordCalorieTracker

- Created a **personalized user interactive bot** to handle **commands** and keep track of user's calorie count.
- Utilized **SQLite** to store **parsed** user inputted data and keep track of what days their calorie goals were met.
- Implemented **commands** to allow user to input/view list of food/calories consumed throughout the day.
- Hosted on local **Linux server PC** to run **24/7** for **active users**.

Battleship | C++

github.com/AlexHoangs/BattleshipWithAIs

- Developed a **terminal-based** battleship game with multiple options on who to play against
- Incorporated option to play against another player or three different types of **AI's** including, hunt and destroy AI, cheating AI, or random AI by utilizing **classes, inheritance, and polymorphism**.
- **Allocates** board space and allows user to personalize ship placement on any **valid** coordinate on the board.
- Displays each players current board, updating each turn with both opponent's view and player's view.