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

In Progress: Operating Systems and System Programming, Bioinformaticas, and Combinatorics

Technical Skills

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

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, and Problem-Solving. Strong Communication and Interpersonal Skills, Passionate, Self-Motivated, Organized, and Adaptive Learner

Work Experience

Math and Robotics Intern **July 2022 – Present**

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 consisting of Geometry, Algebra 1 and Algebra 2.

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.

Information Technology Support **March 2021 – July 2021**

Corovan *San Jose, CA*

- **Analyzed**, **troubleshooted** and **assembled 100+** company computers and electronics.

Projects

Choose the Best TikTok | *Node.js, SQLite, CSS, HTML* **github.com/AlexHoangs/ChooseTheBestTiktok**

- Produced **user interactive web application** allowing **300+** people to vote for their favorite video among list of 8 personalized user uploaded videos. Utilized **SQLite** to display list of video names and allow video addition or removal.
- **Parsed** video **URL** to preview video uploaded and display 2 videos side-by-side to allow users to compare and vote.
- Implemented **PageRank algorithm** to determine most popular video by utilizing **SQLite** to **store data** and votes.
- Employed **Node.js** and **Express.js API** to function as **backend** of **server** to **get/request data** with **JSON** objects

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.
- Prompts user to eat and input data about their recent meals to help reach their calorie goal for the day.
- 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.