Khai Phan

Location: Clackamas, OR 97015 **Email:** khaiphn41@gmail.com
Website: khaiphan.dev **LinkedIn:** linkedin.com/in/khai-phan
GitHub: github.com/Esk3tit **Phone:** (503) 453-2396

SUMMARY

Results-driven Software Engineer with a strong background in embedded systems and a proven track record in designing, developing, and deploying comprehensive full-stack software solutions. Adept at collaborating with diverse cross-functional teams to deliver innovative applications and features that elevate user experience and promote cutting-edge technology.

SKILLS

- **Languages:** Python, C/C++, JavaScript, TypeScript, HTML/CSS, SQL
- **Technologies:** React, Node.js, AWS, Flask, Git/Github, MongoDB, Kubernetes, Docker, Jenkins, JIRA

EDUCATION

- **Bachelor of Science in Computer Science - GPA: 3.99/4.00, Oregon State University (OSU)** **Expected Winter 2024**
  - Dean's List/Honor Roll 2019 - Present

WORK EXPERIENCE

Software Engineering Intern, Garmin **March 2022 - September 2022**

- Designed and documented efficient test cases and requirements, resulting in a resolution of 83% of all failing/faulting tests for G2N navigator products through the use of strengthened tests and optimized Python & C macros.
- Constructed fuel buttons and implemented auto-saving and input functionality, enhancing user flows and the overall UX.
- Created new fuel onboard capabilities and flyover waypoints, while diagnosing and fixing issues such as excessive waypoints in flight plans, contributing to improved product performance.

PROJECTS

AIphone-but-better

- Transformed the frontend architecture by replacing Jinja2 with React and collaborated with team members to re-engineer the Flask backend into a robust API, ensuring seamless compatibility with the new frontend design.
- Employed Docker to containerize the project and enhance its maintainability and scalability, encompassing components such as Redis, Stable Diffusion AI image generator, Flask backend, and React frontend.
- Elevated the user experience by implementing threaded and background tasks to generate images, communicating with the app via Redis lists and real-time WebSockets (socket.io) notifications.
- Conducted both manual and automated end-to-end testing using Jest and React Testing Library to guarantee optimal performance and functionality of all components.

Py-Type

- Developed a Python-based touch typing practice console application, optimizing user experience across diverse system environments and operating systems through Docker containerization.
- Employed the curses library for enhanced text display, utilizing distinct colors and custom characters for clear user guidance and minimizing confusion during practice sessions regarding correct/incorrect input and whitespace.
- Implemented a robust user statistic tracking feature, enabling users to monitor their progress in terms of accuracy, maximum words per minute, and average words per minute across multiple sessions.