

RAY KIM

Kirkland, WA · welcomeraykim@gmail.com · (808)-462-0121 · <https://goldenkimchee.github.io/>

EDUCATION

University Of Washington

BS Computer Science & Software Engineering *GPA: 3.81*

Bothell, WA

June 2021 - July 2023

EXPERIENCE

Haggett Hall

Residential Community Standards Representative of Haggett People's Council

Seattle, WA

March 2019 - July 2020

- Increased student participation at dorm events by 120%
- Designated speaker at community living workshops with dozens of attendees
- Managed up to 10 eviction pleas and notices for tenants

Remote Tutor

Computer Science Tutor

Kirkland, WA

March 2021 - Present

- Analyzed and debugged JavaScript and Python code and provided feedback
- Organized tutoring environment to promote learning and productivity
- Taught various algorithms and data structures to those of 13-17 age range

PROJECTS

Content-based Image Retrieval App *Python, Tkinter, Docker, Pillow, Numpy*

<https://github.com/GoldenKimchee/CBIR-app>

Python app that analyzes an image's RGB values then calculates the color intensity and scheme with the Manhattan Distance formula. • Utilizes relevance feedback for machine learning. • Reduced image analysis time by 80% by using numpy and caching methods.

Video Game API *Java, SpringBoot, Azure, Hibernate, Postman, PostgreSQL*

<https://github.com/GoldenKimchee/Video-Game-API>

REST API that fetches video game review data. • Worked with a scrum team to design a schema and improve query efficiency. • Developed using Java SpringBoot and Hibernate ORM. • Hosted PostgreSQL database on Microsoft Azure.

ChatMe *HTML/CSS/JavaScript, Node.js, Express.js, Socket.io*

<https://github.com/GoldenKimchee/OpenChat>

Fullstack responsive chat app using JavaScript frameworks. • Developed fully functional backend with Node.js, Express.js, and Socket.io. • Created robust front-end with CSS Grid/Flexbox to be responsive on all devices.

Video Shot Boundary Detection System

<https://github.com/GoldenKimchee/Video-Shot-Boundary-Detection-System>

App detects the start and end of each shot, identifying both cuts and gradual transition. • Displays shots in Tkinter UI, where each shot can be played. • Implements twin-comparison based mathematical approach to distinguish shots as cut or gradual transition. • Improved video processing by 40% with numpy, caching, and bitwise operations.

TECHNICAL SKILLS

Languages:	Java, HTML/CSS/JavaScript, TypeScript, Python, C++, SQL
Frameworks/Libraries:	CSS Grid/Flexbox, Spring, Hibernate, React.js, Node.js, Express.js, Tkinter
Tools:	Git, Postman, Docker, Markdown, Heroku