KOLBY SAMSON | RESUME

Student at University of Washington-Bothell

Email: kosamson@uw.edu
Website: kosamson.github.io
GitHub: github.com/kosamson
LinkedIn: linkedin.com/in/kolbysamson

Education

University of Washington-Bothell - B.S. Computer Science & Software Engineering   Sep 2019 - Jun 2022

· 3.88 GPA

· Relevant Coursework: Data Structures & Algorithms, Software Engineering, Operating Systems, Information Assurance & Cybersecurity

Skills

Programming Languages -

· Java, C++, Python

Tools & Technologies -

· Git, GitHub, IntelliJ IDEA, Visual Studio Community, Visual Studio Code

Projects

Skill Bank - Skill Practice Tracker   Sep 2020 - Oct 2020

· Desktop **Java** Application allowing users to track accumulated skill practice hours

· Designed Java **Swing GUI** to allow users to intuitively view and manage their unique skill banks

· Integrated skill tracking with to-do list functionality to boost user productivity

· Implemented application unit testing using the **JUnit** testing framework

Raava Discord Bot - Server Logging Bot Application   August 2020

· **Python** Bot Application connected to servers on the Discord communications platform

· Leveraged the **Discord API** through the discord.py API Wrapper library

· Expanded upon Discord's "Audit Log" feature with additional logging events to support server administrators

· Implemented custom utility commands for server users and administrators to view logs and retrieve server information

Collectibles Store Simulator - Store Management Simulation   June 2020

· Command-Line **C++** Application simulating a collectibles store

· Designed customer database, inventory system, and transaction management system using object-oriented design principles

· Integrated the **Factory and Command design patterns** to enable greater program extensibility

· Implemented simulation logic using data structures such as hash tables, binary trees and arrays

Image Segmentation Application - Seed-Based Image Segmentation   March 2020

· Desktop **C++** Application allowing for segmentation of images into distinct color regions

· Leveraged custom image wrapper library to process input images

· Analyzed image pixels recursively to generate distinct color segments

· Implemented linked list data structures to keep track of connected groups of pixels