# Salim Moshe

kassasalim4@gmail.com — (425) 504-4698 — [LinkedIn](#) — [Portfolio](#) — Bellevue, WA

## SUMMARY

As a senior Computer Science student at UW's Paul G. Allen School, I'm passionate about creating software that makes a real difference. My experience spans full-stack development, systems programming, and AI integration, with hands-on projects in everything from multithreaded web servers to privacy-focused browser extensions. I love solving complex technical challenges and building scalable solutions that improve user experiences.

## EDUCATION

**University of Washington, Paul G. Allen School of CSE, Seattle, WA**

Senior, Bachelor's degree in Computer Science (Data Science Option)              *Expected: June 2026*

**GPA:** 3.90 — Dean's List — Academic Excellence Scholarship

**Relevant Courses:** Algorithms, Data Structures & Parallelism, Software Design and Implementation, Software Engineering, Artificial Intelligence, Machine Learning, Systems Programming, Interaction Programming, Introduction to Data Management

**Bellevue College, Bellevue, WA**                                       *Jul 2022 – Dec 2023*

Associate in Arts and Science (**GPA:** 3.98)

**Relevant Courses:** Introduction to Java Programming, Introduction to C# Programming

## SKILLS

**Programming Languages:** Java, Python, C#, C, C++, Dart, JavaScript, TypeScript, SQL, HTML

**Databases:** SQLite, Firebase (Realtime Database / Firestore)

**Development Environments:** IntelliJ, Eclipse, VS Code, PyCharm

**Libraries & Frameworks:** Flutter, React, NumPy, PyTorch, JDBC

**Runtime Environment & Other Tools:** Linux, Node.js, Git, Firebase, LaTeX, Tableau, Vega-Lite, JSON

**Soft Skills:** Teamwork, Communication, Detail-Oriented, Organized, and Efficient

## PROJECTS

**CookiesShallNotPass - Privacy-Focused Chrome Extension**              *Sep 2025 – Dec 2025*

University of Washington – Software Engineering Project — JavaScript, Chrome Extension APIs, Git

- Built a Chrome extension that automatically blocks non-essential tracking cookies and removes intrusive cookie banners, giving users a cleaner browsing experience without manual intervention.
- Designed flexible user controls including whitelist/blacklist functionality and three aggressiveness levels to balance privacy protection with site functionality.
- Implemented real-time cookie blocking using Chrome's Declarative Net Request (DNR) and Cookies APIs, with inter-module communication via chrome.sendMessage for seamless coordination.
- Created visual analytics dashboard displaying blocked cookies, top blocked domains, and specific cookie statistics to give users transparency into their privacy protection.
- Utilized chrome.storage for local preference and statistical data persistence, with chrome.storage.sync for 30-second interval synchronization of whitelist/blacklist across devices.

**File System Search Engine with Web Server**                          *Jul 2025 – Aug 2025*

University of Washington – Systems Programming Project — C, C++, Git

- Developed a console-based search engine in C using POSIX I/O, linked lists, and hash tables to crawl directories, parse files, and build an in-memory inverted index.
- Created a query processor to handle multi-word searches, rank results by frequency, and display relevant documents.
- Extended with a multithreaded C++ web server (hosted on attu for testing) that handled client HTTP queries and returned results in a browser with links to matching files.
- Patched security vulnerabilities including cross-site scripting (XSS) and directory traversal; validated correctness with unit testing and Valgrind.

**DawgDealz Marketplace App**                                          *Nov 2024 – Dec 2024*

University of Washington Final Project — Flutter, Dart, Firebase, JSON

- Collaborated in a team to develop a marketplace app using Flutter and Dart, with Firebase for authentication, data storage, and image hosting.
- Implemented camera/gallery integration to capture/upload images to Firebase Storage and store item details in Firebase Database.
- Developed functionality to fetch and display uploaded items on the homepage, as well as enable users to view and delete their own listings.
- Integrated Google Gemini API to automatically generate item descriptions from item names if the seller opted not to write one.