Jonathan Chan

Computer Science Graduate @ UBC | Former SDE Intern @ Alida | Google SPS 2022 Participant

chan_jon@outlook.com (778) 953-6781 linkedin.com/in/chan-jon chanjonathan.github.io

SKILLS

Languages:	C, C++, C#, Python, TypeScript, JavaScript, Java, Go, Bash, SQL, HTML, CSS
Tools and Frameworks:	React, Knockout, jQuery, VanillaJS, Node, .NET, Jakarta Servlet, Express, FastAPI, Postman, scikit-learn
Testing Frameworks:	Jest, Sinon, Mocha, Chai, NUnit, Moq, JUnit

EXPERIENCE

Software Developer Intern | *C#, TypeScript, JavaScript, .NET, React, Knockout, jQuery, Jest, Mocha, NUnit* Sep 2022 – Aug 2023

Alida *Vancouver, BC*

- Took charge of **end-to-end development** of an epic, playing a key role in **design, implementation, and delivery** of a server-side C# feature, using .NET lifetime management, for seamless media transfer across regional pods, eliminating an entire manual workflow for all customers.
- Created a new .NET backend endpoint in collaboration with **cross-functional teams**, enabling customers to integrate their own localization service, as well as extended the accompanying Knockout and jQuery frontend, fully automating data entry, saving significant time and effort for our largest client.
- Engineered new frontend features in TypeScript and React while resolving legacy-modern stack cross-compatibility issues on webapp.
- Overhauled UI for mobile viewports, improving readability for over **50%** of users, fulfilling commitment mobile-first experience. Recognized for having identified the most defects in pre-launch **quality assurance** efforts.
- Devised product acceptance criteria in edge cases based on **requirements** in active collaboration with UX and product teams.
- Proactively contributed to **code reviews**, Scrum sprint planning, and ticket analysis, maintaining a critical focus on acceptance criteria, and preventing scope creep across multiple sprints.

PROJECTS

Barrelfish Microkernel Operating System Implementation | *C* Sep 2024 – Dec 2024

- Architected and implemented core kernel components for a microkernel OS, employing a capability-based access control model.
- Built a physical memory manager as a higher-level component using a rudimentary slab allocator to track, allocate, and reclaim RAM regions, enabling variable-sized allocations and reduced external fragmentation.
- Designed and developed a virtual memory manager to map pages between virtual and physical addresses, manage allocation and deallocation, and dynamically allocate the heap, avoiding resource conflicts and reducing external fragmentation.
- Created process management functionality, including spawning a child process and providing it memory, suspension, resumption, and termination, and devised mechanisms for state management and parent-child communication and resource sharing using LMP.
- Optimized IPC procedure that enables inter-process RPCs, modularizing message structures for improved extendibility.
- Bootstrapped a secondary core, coordinating RAM allocation and integration with the primary core for multi-core operation.
- Enhanced modular design by extending RPC functions to support LMP and UMP, enabling communication and resource sharing between the appropriate intra- and inter-core processes, while avoiding infinite RPC loops.

Raft-based Sharded Fault-Tolerant Key/Value Service | *Go* Feb 2024 – Apr 2024

- Engineered a reliable, scalable, Raft-based distributed system that establishes consensus and replicates logs in a fault tolerant manner.
- Implemented log compaction to curb memory usage by creating, distribution, and writing to storage snapshots, then trimming logs.
- Built on top of raft system to implement a key-value service, that guarantees consistent get, put, and append results.
- Further implemented a sharded system to distribute shards of key-value responsibilities among different groups of rafts.

Space Agency Database | *SQL, FastAPI, Python, React, TypeScript* Feb 2024 – Apr 2024

- Constructed a full stack web application to demonstrate a variety of complex SQL queries for a relational database.
- Designed a logical schema for a relational database and wrote a corresponding database creation and population script.
- Devised a backend REST API that formats queries dynamically based given target tables and parameters
- Crafted a frontend for operating on data tables, responsively reloading data in response to success codes.

Vehicle Collision Map, Google SPS 2022 | *JavaScript, Java, VanillaJS, Jakarta Servlet, GCP, Google Maps API, SQL* Jun 2022 – Jul 2022

- Developed a full-stack web application that displays location-based posts, using the Google Maps API
- Implemented backend HTTP endpoints in Jakarta Servlet, to query and manipulate database for all CRUD functionalities,
- Configured database tables on Google Cloud SQL database and implemented image upload and retrieval using Google Cloud Storage.

EDUCATION

Bachelor of Computer Science Sep 2021 – May 2025

University of British Columbia | GPA: 4.30 *Vancouver, BC*

BSc. Combined Biochemistry and Chemistry Sep 2015 – May 2020

University of British Columbia *Vancouver, BC*