Jonathan Chan

4th Year Computer Science @ 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:	TypeScript, JavaScript, Python, Java, C#, C, C++, 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*

- Engineered new frontend features in TypeScript and React and resolved legacy-modern stack cross-compatibility issues on webapp.
- Diagnosed and resolved UI defects, and overhauled frontend for mobile viewports, **increasing readability for over 50%** of users, fulfilling commitment to a mobile-first experience. **Recognized for having identified the most defects** in pre-launch blitz testing.
- 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.**
- Devised product behavior acceptance criteria in edge cases in active ongoing **collaboration** with UX and product teams.
- Contributed proactively in Scrum sprint planning and ticket analysis, maintaining a critical eye on acceptance criteria, and preventing scope creep across multiple sprints.

PROJECTS

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.
- Implemented a backend REST API that formats queries dynamically based given target tables and parameters
- Crafted a frontend for operating on tables, responsively reloading data in response to success codes.

Campus Explorer | *Express, React, TypeScript, Chai, Google Maps API, Google Routes API*   Jan 2024 – Apr 2024

- Constructed a full stack classroom exploring web application that loads and displays data parsed from online resources.
- Implemented backend data façade that parses file inputs into workable data and stores datasets in persistent disk storage.
- Supports a query feature that parses and returns results for SQL-like queries including projection, selection, and aggregation.
- Created a REST API to send data to and retrieve data from the backend.
- Crafted a frontend that displays buildings on a map by coordinates, where their classroom details can be selected and viewed.
- Leverages the Google Routes API to display estimated travel times between multiple selected classrooms.

TabTriage, BCS Hacks 2023, 3rd Place | *JavaScript, OpenAI API, chrome.tabs API, HTML, CSS*   Mar 2023

- Innovated a Chrome extension employing AI to categorize and label tabs for improved organization.
- Utilized OpenAI API to produce categorizations and category labels.
- Called chrome.tabs API to extract browser tab data, create labelled groups, and move tabs into groups.

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: 4.33   *Vancouver, BC*

BSc. Combined Biochemistry and Chemistry   Sep 2015 – May 2020
University of British Columbia   *Vancouver, BC*