Baizhou Chen

baizhou.chen@uwaterloo.ca | 1-604-727-7587 | Andre-Chen

Skills

- Proficient in **C++**, **Python**, **Java**
- Proficient in **SQL** database
- Experienced in **Git** version control
- Experienced in web development framework and tools like Node.js, **React**, React Hook, Redux, and Chrome DevTools
- Experienced in developing Android Mobile App with **Android Studio**
- Experienced in Data Science Tools like **Apache Spark** with RDDs and DataFrames
- Good teamwork and communication skills

Experience

Web Developer - MHS Victoria (Victoria, BC) May 2019 - September 2019

- Implemented a landing page for Pre-Launch stage
- Setup and customize online web store theme on Shopify
- Customize web components by using Liquid, CSS and Bootstrap4
- Remediate bugs and issues found during testing

Education

Bachelor of Mathematics June 2020

University Of Waterloo, Waterloo, ON

Minor in Computer Science

Projects

Covid-19 Big Data Visualization (Python, Spark, pandas, Matplotlib, SQL)

- Analyzing and visualizing large dataset of Covid-19 to present trends of cases and death, and compared total cases and death rates with different countries
- Using DataFrames and SQL to filter out the required data for future analyzation
- Using Pandas and Matplotlib to perform data visualization

Photo Gallery (Java, Android Studio)

- An interactive mobile application that allows users to load images and display these images in a dynamic layout that responds to device orientation.
- Experiences in developing and debugging through Android Studio

Paint （Java)

- Designed asynchronous graphical application with model-view-controller (MVC) architecture
- Implemented the distributed event handling system by using observer design pattern
- Integrated features to allow user to load and save paintings files from hard drives.

Breakout (C++, Xlib)

- Integrated X Window System (X11) to provide graphical user interfaces and input device capabilities
- Implemented realistic paddling physics to provide a visually smooth bouncing animation
- Adapted double buffering to overcome bottleneck issues

Catan (C++, Xlib)

- Implemented the game with observer design pattern
- Used UML for system modeling and data modeling
- Led and managed a group of 3 people to implement the program and design the GUI

WLP4 Compiler (C++, MIPS)

- Compiles WLP4 sources, a C-like programming language, into MIPS executables
- Utilized Abstract Syntax Trees to parse syntactic structures
- Implemented error checking mechanics to display appropriate error messages

OS-161 operating system (C, Bash, Git)

- Modify and implement functions like semaphores, locks, condition variables, system calls and simple virtual memory system in kernel