

# Wesley Sequeira

SOFTWARE DEVELOPER · BUSINESS AND COMPUTER SCIENCE MAJOR

☎ 778-990-5351 | ✉ wesley.sequeira@hotmail.com | 🏠 wesleysequeira.com | 📱 wes1498 | 🌐 wesleysequeira

## Skills

### Programming Languages

C#.NET · C++/Linux · Java · HTML/CSS

### Development Tools / Frameworks

.NET Framework · Vue.js · React.js · Git · PowerShell · Visual Studio

### Other

TeamCity CI · Octopus Deploy · MS SQL Server/T-SQL · Entity Framework · Jira

## Education

### University of British Columbia

Vancouver, CA

B.COMMERCE IN BUSINESS AND COMPUTER SCIENCE

Sep. 2016 - PRESENT

- Cumulative GPA: 3.7/4.33 (80%)
- Key Courses (grade/course average): Computer Hardware and Operating Systems (85%/67%), Internet Computing (85%/77%), Algorithms and Data Structures (86%/72%)

## Internship

### RBC Global Asset Management

Vancouver, CA

SOFTWARE DEVELOPER - OPERATIONAL & SIPP TEAM

May. 2019 - Dec. 2019

- Designed and implemented an Application Health Page which improved incident reporting and reduced health monitoring by 60% using ASP.NET MVC, Vue.js and SQL Server with Entity Framework 6
- Implemented a CI/CD pipeline for the status page to quicken release cycles and seamlessly run Rhino Mock unit tests using TeamCity CI and Octopus Deploy CD
- Automated the Statement of Investment Policies and Procedures (SIPP), a complex investment documentation process used by the Wealth Management division written in C# with .NET Framework, and React.js with Jest and Enzyme for testing
- Enhanced database design using defensive programming techniques to tackle error handling, loss of data integrity, and changes in requirements and new versions with T-SQL on Microsoft SQL Server
- Improved over 50 different ETL workflows across the organization through writing PowerShell scripts and using Secure Shell FTP/SFTP with the ActiveBatch IT Automation software

## Projects

### FTP Model

Jan. 2020 - April. 2020

- Built an FTP client-server network in C (server) and Java (client) using the RFC-959 Internet Standards for the FTP specifications
- Implemented the Factory Design Pattern to decouple the client from implementation while adhering to the SOLID design principles and promote reliable and efficient file sharing
- FTP model also handles a more secure, firewall friendly passive (PASV) data transfer processes command initiated by the client

### Transaction Load Handler

Dec. 2019

- An online simulator that sends a high volume of network requests through server-side Python scripts to emulate scaling burst load transactions
- Takes advantage of containerization technology and handling requests using AWS Elastic Container Service (ECS) to run the Docker application on a scalable cluster
- Implemented an automatic load balancing (ALB) service that spins up new pods at 80% CPU capacity to reduce execution time

### UBC Query Engine

Dec. 2018 - April. 2019

- A full stack development project that queries multiple data stores for relevant course, instructor, and building information in UBC to improve the course registration process
- Used Test Driven Development (TDD) as a black-box testing strategy to better understand user specifications and requirements, while decreasing the amount of debugging work needed during development
- Developed the system API in Node with Typescript which parses JSON format datasets, and uses quick and effective EBNF-based querying to search for results
- Adapted the project to provision REST endpoints to facilitate remote clients

### Stock Trading Platform

Sep. 2018 - Dec. 2018

- A stock trading platform which allows clients to create an account, add a portfolio, and hold various investments pulled from the Capital IQ dataset
- Utilized ER Diagrams and a Relational Model in designing our database to help eliminate any functional dependencies that violate BCNF standards
- Integrated the MAMP solution stack for local testing on MAC OS, MySQL for RDBMS, PHP for server rendering, and Bootstrap with ColorLib CSS for the frontend UI