**Clearwater Analytics**

Clearwater Analytics is a financial technology startup with headquarter in the heart of Boise, Idaho. Clearwater provides web-based portfolio management for thousands of Clients. Customers who utilize their platform include Facebook, Cisco, and Oracle.

Internally, Clearwater puts great emphasis on staying at the forefront of web technologies. An agile development process is used with weekly training seminars during lunch to ensure developer’s skills stay sharp. While the exact technologies used vary by team, popular frameworks include: Java 8, Jersey 1 & 2, Microsoft SQL Server, Angular JS 1 & 2, JQuery, MongoDB, and a plethora of other technologies.

I am starting work at Clearwater full time with the title of **Software Engineer** on July 10th, 2017. Wish me luck!

**Clearwater Analytics (Internship)**

I interned at Clearwater Analytics in the summer of 2016. The intern team consisted of seven interns with two full time employees as team leads. The team initially worked on a project that utilized RESTful endpoints, a T-SQL database, RabitMQ Queue, and a frontend composed of Angular1 framework.

After approximately a month, this project was moved into the integration testing phases. While a portion of our team was needed to complete this, four of our group moved onto a series of sprints focused on improving a legacy product. This product was used by client services

**Compressor Controls Corp (Internship)**

Compressor Controls Corporation is a global provider for control devices for turbomachinary. A wholly owned subsidiary of Roper Industries, CCC has offices around the globe. This includes offices in Russia, Saudi Arabia, and East Asia.

I interned at Compressor Controls Corporation the summer of 2015. While there, I mainly focused on C# (.net) development of an application to automate the build process of Series6 compressor projects. This was both my first time working in a corporate environment and working in C#. As a result, I learned a massive amount, even given my short tenure in CCC.

**Applicable Skills**

**Java**

My experience in it can be divided into two categories: academic, and professional. Academically, Java has a special place in my heart. Not only was Java the 1st language I learned on, but I also programmed the majority of projects during my university days in Java.

Professionally, I am less tied to sentimental feelings. However, my company does use Java 8 EE for the vast majority its backend code. This has given me experience in Java well beyond what I could have learned in university classrooms. Some libraries in use are guice dependency injection, the jesery2 framework, and various custom internal libraries. Learning these, these along with design patterns and business logic, is continually improving my skills in Java.

**Javascript**

I learned the vast majority of my javascript knowledge upon beginning my employment at Clearwater Analytics. My work tends to use a menagerie of javascript frameworks to create the UI of both client-facing applications and internal tools. At my time at Clearwater, I have worked in Angular1,Jquery, along with the tiniest bit of Node.js.

Javascript is an...interesting language. I don’t feel outspoken nor afraid when saying there are many things wrong with the platform. In many ways, I find it oddly similar to C++ in that it has completely overgrown its original purpose and can cause serious issues if used incorrectly. That being said, there can be no denying Javascript’s reach and impact on the software development world.

**Python**

Unlike much of my knowledge, my primary vehicle of experience and knowledge of python is through personal projects. While I will occasionally code in Java in my spare time, out of either familiarity or a perceived shortcoming in a specific area not covered by my job, python is my language of choice outside of work. This is mainly because python allows development of an application quickly, without much of the boilerplate that defines other languages.

While most of my experience of python comes from a hobbyist's perspective, I would recommend it in many scenarios. Python not be as extensible, performant, or easy to develop on an enterprise level, as Java. However, there are many projects and companies that are looking for quick deployment without much of the baggage other languages carry. Python is an amazing language, so long as it’s not shoehorned into situations where it doesn’t belong.

**HTML/CSS**

While I am neither a web designer nor web admin, I still understand the importance of basic HTML and CSS knowledge. My work makes it easy; we mostly make use of bootstrap on non-client facing applications so that teams can spend less time fiddling with aesthetics and more time programming. HTML is mostly done in the perspective of integration with javascript.

However, my desire to learn more about HTML and CSS drove me to create this website. I felt that learning these items through action would facilitate my understanding and improve my skills as a developer. While graphic design is not my forte, I was pleased how different CSS libraries allowed me to write a fairly attractive website without too many difficulties.

**RESTful Endpoints**

My job makes extensive use of restFUL endpoints to accomplish a large majority of our goals. The usage can be divided into two different categories: Interactions with UI, and interactions with other services. The vast majority of our user interfaces are web clients that interact with exposed endpoints utilizing RESTful architecture. The second use case is interaction between projects maintained by different teams. Both cases make use of Jersey1 or Jersey2 frameworks to create and expose their endpoints.

While RESTful endpoints is a programming paradigm rather than a language, I feel that it is extremely important in modern software development. Simply put, most user/computer interactions happen from the browser. RESTful endpoints are an excellent way of using traditional languages without having to worry about language-specific UI quirks. It can also serve up information fairly fast in an easily consumable format. While they are not an end-all, RESTful endpoints are an extremely important part of my toolset.

**Testing Frameworks**

A portion of my role at Clearwater is integrating testing frameworks into our stack, as well as ensuring existing testing frameworks and methodologies are being executed in an optimal manner. Proper testing not only allows The QA team to do their job more efficiently, but also promotes better code, especially in the longer term.

As clearwater uses Java and Javascript for the majority of their frontend and backend, a majority of our tests are some sort of unit test. These are written in the jasmine or javascript framework. It is the responsibility of every developer to write efficient tests of their code. In addition, there are a number of both integration tests and muti-platform frameworks that clearwater takes advantage of. Notable ones are the Pact and Rest Assured framework. Proper testing, while a bit dull, allows us to develop properly and, ultimately, more efficiently.

**Build Systems and Professional Deployment**

I love tinkering away in my spare time. It is not only fun, but improves my skill as a software developer. However, there are many experiences and skills that I cannot achieve from a hobbyist's perspective alone. Working at clearwater gives me experience in a professional environment.

This experience manifests itself in many ways. Some are benign things to working with Maven,Artifactory,NPM, and other tools in a manner more advanced than I would reach on a github project. Others, such as teamwork, inter-team communication, and the art of writing code that can be maintained and built upon for years instead of thrown out and refactored once every couple years. Not only are these skills integral to professional software development, but they are valuable in any profesional environment.

**Agile Development**

The Agile development process, while not complicated, is an important part of the vast majority of modern developers' process. My team participates in daily stand-ups, weeklong sprints, and uses epics extensively to plan longer term goals. The agile process enables both me and my team members to develop more effectively.

**SQL**

Full disclosure: I am not a database administrator.

While more and more stacks are switching to a NoSQL database option. SQL remains an important part of the modern development ecosystem. SQL is still an extremely viable option for many companies, especially ones working with more relational systems. SQL, while not a programming language, is an essential part of a developer's toolset.