Ryan Fleck

Email: employ@ryanfleck.ca – GitHub: RyanFleck
Phone: 613 501 4043 – Website: ryanfleck.ca

Education

University of Ottawa – BASc Computer Engineering

- Graduating Winter 2021, currently in fourth year, four courses remaining

Experience

Wise Assistant – Backend Engineer — June 2020 – Present

- Worked in a team of three to develop a Django backend for a new health application
- Used Django Rest Framework to write secure model serializers for frontend usage
- Adapted multiple third-party authentication libraries to a custom user model
- Wrote an image processing pipeline to resize profile pictures and store in Amazon S3
- Called the SendGrid API to send account verification and password reset emails

IBM – Extreme Blue Technical Intern — Summer 2019

- Prototyped a tool to replace deprecated libraries within compiled Java EE binaries
- Used Javassist to manipulate bytecode, wrote algorithms to update class functions
- Architected a transformation-rule organization system using the reflections library
- Created team development workflow with docker, shell & python scripts, Travis CI

MNP LLP – Co-Op Developer — January – December 2018

- Spoke directly with MNP clients to investigate and address reported bugs
- Wrote (in team of three) a replacement Drupal backend for an internal SiteCore CMS
- Developed features, fixed bugs in a client's Teamsite (Java EE/Spring) CMS
- Wrote technical and user documentation for a client's CMS created with SiteCore
- Wrote AEM HTL/JSP components, configured workflow & security for bid prototype

Volunteering

UO Supermileage – Motor Controller Developer — September 2019 – Present

- Responsible for re-writing the embedded brushless DC motor controller code using C
- In a team of two, updated code from polling to interrupt-based sensor reading
- Manage the website redesign team, migrate to the Hugo static site generator.

Xalgorithms Foundation – Developer/Contributor — April 2018 – Present

- Implemented a prototype React application to test a potential method of rule writing
- Wrote unit tests in Scala for a core system component, the open source rule interpreter
- Tested the rule parser against sample rules, proposed improvements, reported bugs

Projects

- Influx TMS – Team Management System written for SEG3102 course with Django
- Socket.IO Trivia – Developed a UX trivia webapp in Express with scoreboard & chat
- Fake News – Created a flask app to serve content generated from the given URL
- Telegram Chat-Bots – Using Python and NodeJS, wrote bots to respond to key words from friends, query Wolfram Alpha, and provide tools for running D&D games in-chat
- Games – Numerous 1-4 player experimental demos in Godot, Unity, Unreal, Phaser
- Slack BusBot – In a team of two, wrote a slack chat-bot using the OC-Transpo API

Technology

I am familiar with the following technology, and can utilize test-driven design with industry standard design patterns and tools to create and deploy secure business applications.

Python	Java	Java EE	C#	.NET	Razor	HTML	CSS	JavaScript
C	C++	Arduino	Ruby	XML	JSON	REST	SOAP	PostgreSQL
JUnit	Jest	Android	WS	PHP	Spring	Mongo	Scala	HC12 ASM
Docker	Hugo	Jekyll	Go	XSL	Heroku	SEO	SQL	LaTeX

Additional Information

Java Experience

I'd like to elaborate on my Java experience during my previous placement at IBM as an Extreme Blue Technical Intern, during other internships, and at the University of Ottawa.

The bulk of my experience with Java was tinkering with the Java Virtual Machine and writing a sort of re-compiler at IBM. Though I have also used Java extensively while working at MNP to build out content management systems, and in class during programming and Software Architecture courses, my deepest forays into the JVM took place at IBM, working with the WebSphere and Open Liberty platforms.

While I can't divulge the details, I was working in a team with four members; two other technical interns, one business intern, and myself. The goal of the internship was to develop a business pitch and working technical prototype before the end of the summer, in order to present to a panel of IBM executives in New York. Our application had to take compiled binaries for an older version of an enterprise Java system, surgically remove parts of the application that called deprecated functions or used outdated libraries, and replace what we could with new/compatible bytecode. The deprecated parts of the binary that could not be updated were to be flagged. We estimated that, based on the targeted libraries, we would be able to transform at least half of the client applications to use modern libraries with zero developer work for the client. As far as I know, the project that our team started in 2019 has been carried forward and is currently in development.

At MNP, I was tasked to fix bugs in an airline's website/cms. This involved working with a rather old copy of Opentext TeamSite, navigating a massive codebase, and learning how to manipulate XML with XSLT to present data to the frontend.

In my Software Architecture course, we had to develop multiple applications running on GlassFish and using Java Server Faces (JSF) as a presentation technology. Though I completed my final project in this course with React/Django, I learned a lot about building flexible, scalable enterprise applications.

With all of the above in mind, I've probably had about eight months of real enterprise Java experience. This is complemented by years of other programming experience, from the frontend and web applications, all the way down to embedded C. I still have a lot to learn, but I already have a rather solid foundation for new experiences to build upon.