Jason Hu

📞 (416) 909-3633
✉ jason.h.7249@gmail.com
in /hujason1
○ /HuIsJason
🌐 Portfolio

🎓 Education

University of Toronto Sep 2018 – Present
Candidate, Honours Bachelor of Science *Toronto, ON, Canada*

› Specializing in the **Entrepreneurship Stream** of **Computer Science Co-op**

🥧 Experience

Software Developer Jan 2020 – Aug 2020
OpenText, DevX (Developer Experience) team *Richmond Hill, ON, Canada*

› Created a document management app built with **React**, **React Router**, **Material-UI**, and **TypeScript** to demonstrate OpenText's cloud content management APIs and services
› Created a **Python Tornado** dashboard app to display unit test results using **D3.js** and **Bootstrap**
› Configured automated testing **Python script** to accommodate Windows VMs using **PowerShell** and **Paramiko**

⚙ Skills

Languages JavaScript/TypeScript, Java, Python, HTML, CSS, SQL, C

Libraries/Frameworks React, Node.js, Express, Spring Boot, Material-UI, Bootstrap

Technologies Git, MongoDB, Neo4j, Unix, Postman, JIRA, GitHub

Methodologies REST APIs, microservices, full stack development, NoSQL, Agile Scrum, version control

⚗ Projects

Safer Strides – Crime heat mapper app

› A mobile app built with **React Native** that displays a crime heat map over Toronto with real-time police reports from Toronto Police Twitter
› Automated **web scraping** of police report tweets using UiPath's **Robotic Process Automation tool**
› Wrote a **Python script** to parse web scraped tweets into **JSON** objects

Planit – Trip advisor app

› A full stack mobile app built with **React Native**, **Java Spring Boot**, and **MongoDB** that generates vacation itineraries through **Google Maps API**
› Implemented itinerary manipulation such as adding and deleting specific events from the itinerary while shifting existing ones, as one of the main features
› Created **CRUD REST API** endpoints for itinerary information and filters, queried with **JSON**
› Followed the **Agile Scrum** development process while using **JIRA** for project management

Spotify-lite – Backend microservices

› A backend for a music player service composed of two **REST API microservices** built with **Java Spring Boot**
› Stored user info in a **MongoDB collection** and playlists in a **Neo4j graph** respectively
› Wrote **Cypher graph queries** for nodes up to three relationship depths away
› Used **Postman** for testing endpoint calls and microservice communication