Ibrahim Fazili

☎ +1 (647) 904-5015
✉ ibrahim.fazili@mail.utoronto.ca
in ibrahimfazili
IbrahimFazili

Education

University of Toronto (GPA: 3.42/4.0) 2018 – *Expected* 2023
Honours Bachelors of Science in Computer Science *Toronto, ON*

Coursework
Software Design, Algorithm Design, Machine Learning, Data Structure and Analysis, Software Tools and System Analysis, Software Engineering, Operating Systems, Information Security, Advanced Calculus, Lin Algebra

Skills

Programming Python, Java, JavaScript
Technologies React Native, NodeJS, Linux/Unix

Experience

Oracle May 2021 – Present
Software Developer Intern *Toronto, ON*

- Developed scalable, robust and easily maintainable backend code for Oracle marketing product integration
- Leveraged Spring and Java to make calls between Oracle's internal APIs and Confluent Kafka service to manipulate and update high-volume data, upwards of 4000 per call
- Verified entity creation followed certain protocols, such as custom javax validation constraints, via XML parsing and manipulation
- Improved project infrastructure by developing generic methods and entities in common libraries to be used by multiple projects. Reduced code footprint of processes such as XML parsing, WebClient creation as well as entity creation by 5%.
- Collaborated effectively with QA team to develop a wide set of reliable unit tests and end to end testing to ensure that the code is robust

Findr May 2020 – Jan 2021
Software Developer *Mississauga, ON*

- Developed front-end code for a startup initiative that aims to create an open source application that connect students to each other to boost productivity and allow them to reach their potential
- Built several core UI components and front-end features in React Native
- Implemented login, sign-up, and profiles/matches feature in mobile application using Javascript
- Leveraged React and Bootstrap to develop a responsive website for the company.

Projects

PyFile - Directory Memory Visualization Program Tech Stack: Python, Pygame

- Designed an interactive Pygame UI displaying how memory is allocated in a users file systems through shapes and colours
- Users perform actions on directories (cd, ls, rm, etc) through UI. Implemented using OS/Tree structures

C(t)alk - Real-Time Terminal Chat Client Tech Stack: C

- Users are able to connect to a server and able to send messages to one another in real time
- Supports multiple clients, with admin users having privileges to kick others off the server
- Implemented a custom protocol to transfer messages, emojis and commands all through a single socket connection