Huiyi Guo . Jarvis Consulting

I am a recent graduate of McGill University with a Bachelor's degree in Statistics and Computer Science. Through my previous internship experiences and the education I gained from university, I gained solid coding skills with Python, Java, JavaScript, SQL, and C as well as problem solving and collaboration skills that I can carry to future roles. During my studies, backend and full-stack development captured my heart as I can turn my creative ideas into applications that solve real-world problems. I enjoy learning new techniques and continuously improving them. I'm now sharpening my abilities with tools and technologies that are commonly used in the business and implementing my skills at Jarvis Consulting Group.

Skills

Proficient: Java, Angular, Node/Express, JavaScript, RDBMS/SQL, Python, Agile/Scrum, Git, Linux/Bash

Competent: Docker, R, MATLAB, C, Applied Machine Learning, Natural Language Processing, Microsoft 365

Familiar: Public Cloud(GCP), JUnit, MIPS assembly, Ocaml, Spring framework/Spring boot

Jarvis Projects

Project source code: https://github.com/jarviscanada/jarvis_data_eng_HuiyiGuo

Cluster Monitor [GitHub]: This project is about developing a Linux Cluster Monitoring Agent that provides metrics about system activities. Bash scripts are used to collect data from the host system every minute using cron and saved to PostgresSQL server that runs inside a docker container.

Core Java Apps [GitHub]:

- Twitter App: This is a Twitter CRUD application that uses Java DAO to communicate with Twitter's REST API to post, display, and remove tweets. The business logic was handled in the service layer. JUnit 4 and Mockito were used to test integration tests and unit tests of this project. The application was dockerized as a Docker image via Dockerhub for easy distribution and execution.
- JDBC App: This project implemented a Java Database Connectivity (JDBC) application using the JDBC API to connect to a database and run queries to obtain the data. The user can use the application to perform predefined actions on the database, such as CRUD operations. Java, JDBC, Maven, Postgres, and Docker were all used when building this application.
- Grep App: This application is a Java-based application that mimics the Linux command line grep function. This application searches files recursively in a given directory and outputs the matched text pattern using regex to a file. The project was completed using Maven Standard Directory Layout following industry standards. Additionally, Lambda API and Stream API are used for managing larger inputs. The grep application was then dockerized and uploaded to DockerHub as a Docker Image.

Angular-Express-SQL-Twitter [GitHub]: This is a simple full-stack Twitter application with CRUD functionality that enables posting, editing and deleting tweets as well as viewing tweet details. The back-end server uses Node.js + Express for REST APIs, the front-end side is an Angular App with HTTPClient and PostgresSQL is used for the Database.

Hadoop[GitHub]: Applied Hadoop, Hive, and a Zeppelin Notebook to solve business issues on a data set. Created a Hadoop cluster with a single master node and two worker nodes using Dataproc. The data was loaded into HDFS and used Apache Hive to analyse it. The HQL queries used to solve the business challenges were stored in the Zeppelin Notebook. In the end, the notebook provided solutions for all business issues. Each notebook cell was put through a series of trial-and-error tests during the testing process. The master and worker nodes were provisioned on Google Cloud Platform for the project's deployment.

Spark[GitHub]: Used Spark clusters to solve business issues on two datasets of data. A master node and two worker nodes were utilised in clusters. One dataset was kept in Google Cloud Storage, and the other was kept in Azure Databricks Storage. Google Cloud Platform and Azure Databricks were utilised in the project's deployment to provision the clusters.

Highlighted Projects

AI agent in game: Developed an artificial intelligence agent using Python that can play against a random agent in Colosseum Survival Game. The agent should maximize its own final score. Users will be able to see agents moves on an NxN game board. The Monte Carlo Tree Search method is used to decide the best next move.

Professional Experiences

Software Developer, Jarvis (2022-present): As a software developer, I was responsible for building multiple projects (see Jarvis projects above). I gained more expertise in scripting, Python, Java, JavaScript, SQL, and Object Oriented Programming. I worked in an agile environment, both participating in and leading daily scrum meetings. Also, I used Github to organize and manage project structures and to ensure the delivery of stable and clean code.

Software Development Engineer Intern, Tower Research Capital (2021 Summer): I designed, coded, and debugged a daily analysis program that analyzed the previous trading day transactions data using Python. I produced supporting reports and documentation to help senior team members complete their projects. I collaborated with the trading team to ensure the analysis results were clean and functional.

Trading Analyst Intern, United Overseas Bank (2019 Summer): I was responsible for analyzing data from various sources and compiling it into daily routine reports. I produced relevant presentations for other teams. I often received positive feedback from my supervisors and team members regarding my presentation quality and practical information.

Education

McGill University (2018-2022), Bachelor of Sciences, Statistics and Computer Science

Miscellaneous

- Reading
- Casual Gamer
- Badminton Player