Enzuo Mou . Jarvis Consulting

As a Computer Science graduate from Queen's University, I bring strong technical skills developed through diverse project experiences. I've designed and built an interactive prompt sharing website using Next.js, MongoDB, NextAuth, and Tailwind CSS, showcasing my full-stack development skills. My previous experience includes developing a user management system at a computer equipment company and building a stock data analysis web application. My proficiency spans Java, Python, JavaScript/TypeScript, Springboot, React, Angular, NextJs, and Django, with competence in tools like Git/Github, Docker, and agile methodologies. Passionate about computer technology's impact on society, I'm equipped with strong problem-solving skills and a knack for bug resolution. As an enthusiastic junior software engineer, I look forward to contributing to innovative technology solutions in my future career.

Skills

Proficient: Java, Springboot, RDBMS/SQL, React, Angular, NextJs, Node.js, JavaScript/TypeScript, Python, Django

Competent: Git/Github, Docker, HTML/CSS, Networking, Linux/Bash/Shell, Agile/Scrum

Familiar: Machine Learning/Deep Learning, Google Cloud Platform, Echarts, C, Numpy, Pandas, Opencv, Pytorch

Jarvis Projects

Project source code: https://github.com/Jarvis-Consulting-Group/jarvis_data_eng-En4zo

Cluster Monitor [GitHub]: Developed a Linux monitoring agent that enables users to easily monitor and store machine specifications and usage information for multiple Linux systems. The agent utilizes Bash scripts to gather hardware information from the machines, which is then securely stored in a PostgreSQL database that is provisioned using Docker. To ensure accurate and up-to-date information, resource usage is fetched at regular intervals using Crontab. The collected data can then be easily analyzed and manipulated using SQL queries, providing users with valuable insights and performance metrics for their Linux systems.

Manipulate an Existing Database [GitHub]: To deepen our understanding of relational databases, relational database management systems, structured query language, optimizations, and data models, we familiarized ourselves with these concepts through hands-on work. Using DBeaver as an IDE, we practiced manipulating existing data and compared our query results with expected outcomes. The existing data was stored in a file, clubdata.sql, which was imported into a PostgreSQL database deployed in a Docker container. This hands-on approach allowed us to gain a practical understanding of the concepts and strengthen our skills in the field.

Core Java Apps [GitHub]:

- Twitter App: Implementing a Java application that allows users to post, show, and delete tweets on Twitter using the Twitter REST API V2. The TwitterApp follows the MVC architecture, where the models encapsulate and manage Tweet objects that are displayed in JSON format. The controller layer interacts with client input (args), parses it, and calls the service layer to return results. The service layer handles the business logic and interacts with the DAO layer to communicate with the Twitter REST API. The data access layer (DAO) interacts with the Twitter REST API to perform post, show, and delete actions on tweets.To implement this design, we use Twitter REST API v2 and combine it with HTTP requests to perform the actions. We also use the OAuth 1.0a protocol to create an additional HTTP Authorization header to authorize HTTP requests. To convert JSON strings from HTTP responses to Tweet objects and vice versa, we use the fasterXML Jackson package. Springboot framework is used to manage dependencies and set dependencies via the class constructor. For testing, we utilize integration and unit testing with the libraries Mockito and JUnit4. To manage all the above dependency packages, we use Maven.Finally, we deploy the Twitter application within a Docker container and upload it to Dockerhub for easy use.
- JDBC App: Interacting with a PostgreSQL database using the JDBC API, this project implements standard CRUD operations (Create, Read, Update, Delete) on the database, utilizing Java version 8. To simplify deployment, we create a container image using Docker to run PostgreSQL within it. The application uses java.sql packages to establish connections with the database and execute SQL statements to perform the CRUD operations. Our project code is managed using Git and GitHub, while dependencies such as the org.postgresql driver are managed using Maven.
- Grep App: Implementing the widely used Linux command grep in Java using Java 8. This project is designed for anyone who wants to quickly search for strings within files. With the grep app, users can filter files and find target strings more efficiently, thereby saving time. The project utilizes Java's Stream API and lambda functions for

efficient string searching within files. Maven manages dependencies, including slf4j and log4j for log management. The Java Scanner and BufferedWriter classes are used to read and write data from and to files respectively. The project is built and packaged into a jar file using Maven, and the jar file is further packaged into a Docker image for easy deployment. The Docker image is available on Docker Hub for easy access.

Springboot App [GitHub]: Integrate a trading platform with a REST API that front-end applications can consume, enabling users to manage client profiles and accounts, monitor portfolio performance, and execute securities trades efficiently. Utilize Java 8 and Spring Boot for the implementation of the REST API, encompassing the platform's core business logic, such as managing trader profiles, executing security orders, and monitoring portfolio performance. Ensure data persistence with a PostgreSQL database. Acquire market data from IEX Cloud, a service that provides free access to market data through its REST API, and incorporate it into the Spring Boot application.

Hadoop [GitHub]: Explore and evaluate Hadoop's capabilities, specifically Apache Hive, to solve complex business problems using distributed computing. Gain proficiency in core Hadoop components, such as MapReduce, HDFS, and YARN, by provisioning a Hadoop cluster on Google Cloud Platform using Dataproc. Utilize Apache Hive and the Zeppelin Notebook to practice various HiveQL queries, compare methodologies, and optimize data retrieval, storage, and access.

Highlighted Projects

Prompt Sharing Website [GitHub]: Designed and built an interactive prompt sharing website using Next.js for front-end and back-end functionality, integrating MongoDB as the database. Implemented NextAuth for secure user authentication and Tailwind CSS to create a responsive, modern interface. Users can post, share, and view prompts, while their personal profile allows access to their individual posts upon login. The platform fosters community engagement, creativity, and knowledge sharing with a seamless and secure user experience.

Stock Data Analysis System: Developed a web application using Django as the back-end framework and Apache ECharts for data visualization and stock trad ing history. My full-stack web development skills in HTML,CSS, and Javascript were utilized to design a RESTful back-end server for storing stock information persistently in MySQL.

Distributed System and Inter-Thread Communication Mechanism(Java): Realized inter-thread communication mechanism using Java Sockets and RMI to allow different processes/thread to communicate over TCP/IP network. Compared different page replacement algorithms in a centralized system that uses virtual memory. Used a distributed system and implemented each page replacement algorithm on particular computer server.

GNU Jami Architectures Analysis: Analyzed and studied architecture, data flow of GNU Jami, and interaction subsystems of GUN Jami. Provided GNU Jami with features, replacing the central server with blockchain to store user accounts.

YOLOv3 Object Detection(Python): Implemented object detection model with YOLOv3 and VOC2007+2012 dataset. Realized real-time detection of 20 types of objects in video (including people, cars, boats, cats, dogs, etc.) Evaluated using mAP value, obtaining mAP of 84.22%.

Professional Experiences

Junior Software Developer, Jarvis (Jan 2022-present): Joined Jarvis as a junior software developer to develop, consult, and assist customers with their software solutions. I work in an Agile/Scrum development environment, where I participate in daily scrum meetings with the scrum master to discuss project progress. And our sprint meeting held every two weeks. At Jarvis, I have gained exposure to various technologies commonly used in professional technical environments, such as Java, RDBMS, Linux, Docker, SpringBoot, React, Angular, Hadoop, etc.

Software Engineer, Start Computer Equipment Corporation (July2021-Sep2021): Tasked with comprehending the framework of the bank evaluation machine's management system and designing the user management page. The management system was implemented using Java and Spring Boot. I successfully completed the user management page, including adding the ability to query machines based on time, region, and relevant data stored in a MySQL database. Daily progress was reported to my instructor, and I also participated in weekly technology-sharing meetings with the team where the team leader would share interesting new projects and emerging technologies on Github with everyone.

Education

Queen's University (2018-2022), Honours Bachelor of Computing, Computer Science, Queen's School of Computing
- Dean's Honour List (2021-2022)

Miscellaneous

- Enjoy snowboarding on the slopes and gain experience in maintaining various snowboard gear. Aim to become a CASI Level 2 instructor.
- I enjoy road cycling and riding with a team at a pace of 27-30 km/h, covering distances of 50-110 km. I am also fond of riding sports gear.
- Worked as an IELTS teaching assistant in high school, assisting students with dictation and creating IELTS learning plans.