Sophia Ruduke

Software Developer with 3 Years of Experience

Contact

s.ruduke@gmail.com

(780) 566 3282

Edmonton AB

https://sruduke.github.io

Sophia Ruduke

Education

Bachelor of Science:

Specialization in Computing Science - 2024

University of Alberta

Programming Languages

- Python
- Java
- C
- R
- SQL
- Bash
- JavaScript

Web Technologies

- HTML5
- CSS3
- React
- Django
- RESTful
- WebSockets

Models Built

Summary

Recent Bachelor of Science graduate with Specialization in Computing Science from the University of Alberta. My coursework has covered a wide range of topics including cybersecurity, image processing, web applications and architectures, software quality and testing, and app development. Concentrated extensively on Artificial Intelligence during my studies, gaining in-depth expertise in data acquisition, cleaning, and formatting, along with strong skills in model tuning, performance evaluation, and interpretability.

With two years of experience at Kymera Systems, I've honed my problem-solving, system design, full-stack development, automation, and collaboration abilities. Initially, I contributed as a software developer and tester, leveraging my OOP skills in Java module development. Transitioning into a systems integrator role, I directly engaged with clients and their SCADA systems, overseeing version control and database management. My responsibilities expanded to include automation – utilizing Ansible, Docker, and custom Systemd services to develop solutions – ultimately integrating DevOps practices into my position.

I am passionate about artificial intelligence, data management, and software development. I am always eager to expand my skills and knowledge.

Experience

Kymera Systems Leduc, AB May 2022 - May 2024

Software Developer & Tester, Systems Integrator, DevOps Specialist

- Developed, maintained, and tested Ignition modules written in Java. This typically included drivers for protocols that are not natively supported by Ignition.
- Removed security vulnerabilities in all modules by replacing passwords saved in the gateway database from plaintext to their hashed counterparts.
- Co-developed a baseball application in Ignition that provided advanced scorekeeping, roster management, and a real-time OBS scoreboard overlay for game streams.
- Proposed and implemented indexing, query optimizations, and data retention policies to improve the response times of large queries. This resulted in a 5-10x decrease in query execution time.
- Facilitated provisioning and conducted deployments of edge devices and Human-Machine Interface (HMI) devices using Ansible.

- **Regression:** linear, logistic, polynomial
- **Classification:** decision trees, random forest, naive Bayes, perceptron, support vector machine (SVM)
- **Reinforcement Learning (off and on policy):** temporal difference learning, Monte Carlo, SARSA
- **Object Detection:** convolutional neural network (CNN), deep neural network
- **Language Processing:** transformers

Software Process/Quality

- Agile Practices
- Scrum
- Test Driven Development

Tools

- Docker
- Ansible
- Ignition
- GitHub Actions
- Tailscale
- Vyos
- Proxmox
- Microsoft Office

Data Management

- Provisioned and tested Vyos routers. Contributed to vyos-modular to add support for the Sagitta release; this required creating a custom class for building the Sagitta image and altering the commands used in the build process.
- Rebuilt an Alarm Rationalization Tool using Ignition and Java that facilitated alarm creation and management. The application leveraged a powerful UI alongside Excel support which decreased alarm configuration time by ~60%.
- Developed an interactive and modular Python tool that enabled large database migrations from InfluxDB to MariaDB.
- Created and managed virtual development environments using technologies such as Proxmox, Quick Emulator, VMWare, VirtualBox, and Docker.
- Performed administration for Windows, Linux and Ignition servers including project version control and databases for clients.
- Supported clients through networking, Docker, and Ignition issues.
- Developed a mechanism to update network settings on HMIs (isolated kiosks) through the Linux USB subsystem, bypassing the lack of network communication and command line access.

Projects

ChimpChat

https://github.com/sruduke/chimp-chat

- **Overview:** A centralized social media server where users can like, comment, and share posts with friends. It has the ability to interface with other platforms through its API, allowing for interaction between users of different services.

- **Responsibilities:**
  - Full implementation of the REST API, using Swagger for documentation.
  - Primary frontend and backend designer.
  - Led a team of 5 using practices such as agile and code review. Coordinated frequently and delegated tasks to each member.
  - Implemented a multi-threaded caching system to speed up response times and reduce network congestion.
  - Configured and hosted a development and production Postgres database on a local Linux server.
  - Deployment of the production server.

- **Technologies Used:** Django, HTML/CSS/JavaScript, Postgres, AJAX, Swagger, Heroku, REST API development.

- MariaDB
- Postgres
- InfluxDB
- MongoDB
- Microsoft SQL Server
- MySQL Server
- SQLite

Skills

Programming

Problem Solving

Machine Learning

Test Automation

Attention to Detail

Collaboration

Databases

Networking

Troubleshooting

Automation

Documentation

Customer Relations

Test Automation & CI/CD

- **Overview:** An IOT smart home application, TartanHome, provides automatic control over heating/cooling, humidity, lights, locks and alarming. The application lacked a test suite, so using GitHub Actions and various code quality tools, a CI/CD pipeline was implemented.

- **CI Pipeline:**
  - Build the Java application using Gradle.
  - Static analysis reports using Error Prone, PMD, and SpotBugs.
  - Unit and integration tests using JUnit5 and Mockito.
  - Generate reports for mutation tests using Pitest, code coverage using Jacoco, and technical debt using SonarQube.
  - System tests using Pytest.

- **CD Pipeline:**
  - This pipeline is run only if there are no errors in the CI pipeline.
  - Docker image for the application is built and published to the local Docker registry.
  - The production server pulls the new docker image and hosts the latest changes.

- **Technologies Used:** GitHub Actions, Docker, Cybera Rapid Access Cloud, SonarQube, JUnit5, Mockito, Jacoco, Pitest, PMD, SpotBugs, Error Prone, PyTest.

LED Strip Controller Application

- **Overview:** Developed a remote-controlled LED strip system using an Arduino Uno WiFi, allowing control via a web application accessible on any phone or laptop connected to the local network.

- **Hardware Design:** Constructed the circuit with logic MOSFETs and resistors for each color channel, ensuring efficient and reliable operation.

- **Software Development:** Implemented a REST API on the Arduino to expose functions for turning lights on/off, changing colors, and adjusting modes. The API facilitates seamless interaction with the LED strip through the web application.

- **Key Features:**
  - **Remote Control:** Users can manage the LED strip remotely, enhancing convenience and usability.
  - **Alarm Integration:** Added an alarm feature that mimics a sunrise effect, gradually increasing light intensity minutes before the set alarm time to provide a gentle wake-up experience.

- **Technologies Used:** Arduino programming, REST API development, web application integration, logic MOSFETs, and resistor circuit design.