Jennings Leavitt
726 N Valley View Dr
St. George, UT, 84770
(435) 414-1164
Jennings.leavitt@gmail.com

SKILLS

Languages: Go, C++, C, Python, Java
Tools/Frameworks: gRPC/Protocol Buffers, JSON, Docker, Kubernetes, GCP, Github, GitLab, SQLite, CI/CD, CI pipelines, Linux/Unix OS, Shell: Bash & Zsh including scripting
Coding Standards: REST api's, Unit & Integration Testing, Code Reviews, Design Patterns, microservices
Databases: PostgreSQL, MongoDB, SQLite
Additional Experience: Hadoop, CassandraDB, DynamoDB, Apache Spark, CouchDB, and Redis

EDUCATION

Brigham Young University, Provo, UT
B.S. in Computer Engineering 2018
(Minors in Mathematics and Computer Science)

Brigham Young University, Provo, UT
Graduate studies in Electrical and Computer Engineering
Jan 2018 - Apr 2021

EXPERIENCE

TCN, Inc.   **May 2021 - Aug 2022**
St. George, UT
Software Engineer II - Backend
- Utilizing Agile methodologies, worked as a backend software engineer while collaborating with front end teams
- Developed Go & Python REST APIs as part of a new complex, web-based, workforce management product for call centers
- Designed numerous effective PostgreSQL queries to create, select, update, and delete data from a large number of tables in our database
- Product used historical call load data as one of a number of inputs to several algorithms, including machine learning, to predict future call loads and then create schedules to handle those call loads—including scheduling agents with corresponding skill sets to successfully field the predicted call load
- Stack and Procedures included: Go, Python, gRPC, PostgreSQL, Docker, Kubernetes, GCP, REST api's, Unit & Integration testing, GitLab - VCS, CI/CD, CI - pipelines, Hexagonal Architecture, Code Reviews

BYU – Configurable Computing Lab   **May 2017 – Apr 2021**
Provo, UT
Back End Software Engineer

- Refactored a repository, improving tools most widely used for data collection in radiation testing, making it more flexible, modular, extensible, and maintainable
- Utilized OOP principles such as encapsulation, inheritance, and abstraction to untangle code and model it in a simple but accurate way to improve performance, user experience, and maintainability
- Interfaced with Linux kernel drivers and applied overall expert Linux knowledge, as well as regression testing
- Developed code in C++ for use on a device running embedded Linux (Archlinux)
- Pioneered a new method of device monitoring for radiation testing, creating new tools that use the onboard CPU of a Xilinx Zynq Ultrascale+ MPSoC (Multi-Processor System on a Chip) to monitor itself as the device under test (DUT)
- Wrote code in C, C++, and minor ARM V8 assembly in the development of the MPSoC self-monitoring project
- Integrated custom firmware, Linux kernel drivers, and Linux applications to create tools
- Automated the initialization of the application on boot, logging radiation effects, and configuring a watchdog timer (WDT) based system reset of the device(s) for when radiation effects had produced an unrecoverable state on the board
- Relevant Skills: Extensive experience with linux, embedded Linux such as Petalinux, building embedded linux kernels, embedded kernel drivers, and embedded applications. Expertise developed in learning from complex technical documentation and experience with real-time operating systems (RTOS) in the formation of the MPSoC project

PUBLICATIONS

Anderson, Jordan & Leavitt, Jennings & Wirthlin, Michael. (2018). Neutron Radiation Beam Results for the Xilinx UltraScale+ MPSoC. 1-7. 10.1109/NSREC.2018.8584297.