Luke Davis

luke.davis.computer
linkedin.com/in/PartlyFluked
luke@davis.computer
github.com/PartlyFluked

EDUCATION

March 2012 – November 2015

Bachelor of Science (Software Engineering, Computer Science)

University of Western Australia

WORK EXPERIENCE

January 2022 – present

Programmer

OzGrav research group

- Designing and implementing the CI/CD iterative development system.
- Containerise build system to use Docker/Apptainer and write deploy scripts / bash functions to utilise this in HPC environments .
- Refining pipeline critical path to lower latency.
- Implemented pipeline artifact analysis framework.

October 2018 – present

Software Engineer

Elemint

- Architected and managed the operations and continuous integration of the systems development life cycle.
- Managed cloud services for provisioning of computational and storage resources.
- Designed and administrated production databases.
- Developed full stack applications for AI model deployment and inference.
- Planned and constructed multiple on-premises CI/CD and GPU compute servers.
- Refactored legacy ML models and systems to modern frameworks (for ongoing R&D).

April – October 2018

Technical Consultant

Lateral Capital Ventures

- Performed market research and codebase analysis on software start-up companies.
- Developed Web3 connected asset swapping interface.
- Developed a high availability market maker and liquidity provider agent.

SKILLS

Fields Systems Administration, Cloud Computing, Database Design and Management, Parallel Computing, DevOps Automation, Computer Vision, Machine Learning.

Programming Lanugages Python, Javascript, C++, SQL, Bash, Ruby, Solidity, Kotlin, Java, HTML.

Frameworks and Libraries Pytorch (with Ray Tune, DeepSpeed, and ZeRO), Node.js, Redis, Flask, FastAPI, Vue.js, Slurm, Ruby on Rails, WinAPI.

Developer Tools Linux, git, Docker, Kubernetes, Jupyter, VSCode, geth, Jenkins, Systemd, WSL.

Productivity Tools Microsoft Office, Google Drive Applications, LaTeX.

Cloud Microsoft Azure, Amazon AWS.

PERSONAL PROJECTS

Public Timetable scheduling with Particle Swarm Optimisation, written in C++ and parallelised with OpenMP.

Android app for user-friendly usage of Shamir's Secret Sharing algorithm, written in Kotlin.

Hearthstone simulator and agent using the Monte Carlo Tree Search algorithm, written in C++ with a WinAPI GUI.

Private Cryptocurrency exchange arbitrageur (over 1000 BTC traded).