

Chris Page

 ChrisPHP |  chrisphp.com |  chris@chrisphp.com |  +44 07743693504

SUMMARY

Full Stack Developer specializing in robotics infrastructure with 4 years of experience building services for autonomous robots. Skilled in JavaScript, ROS1/2 and Python alongside backend applications like Flask and FastAPI.

PROFESSIONAL EXPERIENCE

Saga Robotics - Full Stack Developer

Nov 2021 - Jan 2026

- Developed intuitive GUI for autonomous robots using JavaScript and Python
- Allowed remote users to view GUI and enable the robot to begin operation using websockets
- Used WebRTC to livestream LiDAR data to remotely view point cloud information
- Built a comprehensive web-based monitoring platform to oversee 100+ operational robots in real-time, providing remote support and fleet management for distributed robotics infrastructure using React, Python Flask Next.js and MQTT
- Developed backend services using Python, Flask and FastAPI to handle real time data streams from robotic systems using ROS2
- Utilized MongoDB, Cognite and PostgreSQL databases for efficient data storage of robot operational logs and telemetry
- Enabled remote fleet management across the US and UK improving robot uptime to 97.2%
- Containerized all software including robots using Docker for easy and consistent deployment

EDUCATION

2020 - 2021	Msc by Research Computer Science at University of Lincoln	Pass
2017 - 2020	BSc (Hons) Computer Science at University of Lincoln	(Grade 2:1)
2015 - 2017	BTEC software development lvl 3 at Derby College	(D*D*D*)

TECHNOLOGIES AND LANGUAGES

Languages: JavaScript, TypeScript, Python, Odin
Frontend: Next.js, React
Backend: Flask, FastAPI, SocketIO, MongoDB, CogniteSDK, ROS1/2, MQTT, WebRTC
Other: Procedural Generation, Agile/Scrum, Docker, Git, GCP, Agile

PROJECTS

3D Plant Modelling with L-systems — Academic Project

[Link to Project](#)

Technologies: Python, L-systems, Procedural Generation, 3D modeling

- Developed procedural generation algorithms using L-systems to create accurate 3D strawberry plant models with L-systems
- Implemented parametric controls for plant growth such as temperature, sunlight and gravity
- Findings showed that L-systems were promising at developing authentic strawberry plants for simulations

Pixel Texture Forge

[Link to Project](#)

Technologies: Python, Flask, Docker, JavaScript

- Developed algorithms to convert high resolution textures to pixel art using K-means clustering
- Uses Flask, Gunicorn and Docker to work cross platform
- Modify the colour palette and create Wang tiles with the created textures

Odin Libraries — Open Source

Technologies: Odin

- Developed various tutorials and open source libraries for the Odin programming language
- Ported algorithms that had not been made available in that language such as L-systems, Jump Point Search and Bezier curves
- Created an autotiler that displays the correct tile based on a grid array with wang tilesets

VoxelLibre (MineClone 2) — Open Source

[Link to Project](#)

Technologies: Lua, Open Source Collaboration

- Implemented the Anvil and Stonecutter crafting features to the game enhancing gameplay mechanics
- Collaborated with international development team using git workflow and code review process