Aarjav Patni

+1 437 484 2405 | aarjav.patni@uwaterloo.ca | aarjavpatni | aarjavpatni | Website

Education

University of Waterloo — Waterloo, ON
Bachelor of Software Engineering — *Sep 2023 - May 2028*

Skills

Languages: C, C++, Python, Rust, Go, TypeScript, JavaScript, Java, VHDL, Assembly, HTML, CSS
Technologies: React, React Native, Django, NextJS, MySQL, PostgreSQL, Redis, Selenium, Kafka, Pandas, Docker

Experience

WATonomous — Waterloo, ON
Software Engineer – Server Cluster Division — *Aug 2024 – Present*

- Developed WATcloud CLI – a **Rust**-based **Linux** shell featuring easy access to info such as compute cluster status, daemon process status, and per-user usage quota for the cluster machines
- Enhanced system stability by resolving API integration issues in error reporting systems (**Prometheus**, **Sentry**)

Readwise — Toronto, ON
Software Engineer — *May 2024 – Aug 2024*

- Developed an internal **Django** API to streamline feedback ticket triaging, handling **300+** daily requests, and integrated it with Slack for daily customer experience (CX) statistics
- Engineered a **TypeScript**-based tool to interact with the API, allowing the CX team to create task issues instantly and save **50+** hours per month
- Implemented a document search feature in **React** for the **Rust** desktop app to overcome Safari Webkit limitations
- Resolved auto-update issues in the **Rust** desktop app to prevent conflicts with the **TypeScript** implementation
- Increased PDF to HTML accuracy by **75%** using Diffbot parser APIs, greatly enhancing textbook and research paper readability

Waterloo Aerial Robotics Group — Waterloo, ON
Software Developer – Embedded Flight Software — *Jan 2024 – Jun 2024*

- Revamped the ZeroPilot (flight software) CI/CD pipeline, implementing a **GitHub Actions** workflow to automate testing and deployment
- Resolved issues related to the **Docker** setup for ZeroPilot, enabling local development of the flight software on Macbooks running Apple Silicon

Projects

ZkGraphEngine | *GitHub*

- Engineered a **Rust**-based computational graph library optimized for use in zero-knowledge proofs, supporting arithmetic operations, constraint checking, and a hint system for custom computations
- Implemented comprehensive error handling and logging, ensuring robust performance and easy debugging
- Designed a detailed testing suite to cover edge cases and validate complex graph computations
- Documented future improvements like graph visualization and topological traversal for enhanced usability

Hostel Management System | *GitHub*

- Developed a web application to digitize the process of managing student details, billing, payments, and invoices
- Built using **NextJS** for the frontend and **Supabase PostgreSQL** for the backend
- Implemented an admin interface to view, add, and update student and billing records using the Supabase API
- Automated student billing process by integrating logic to dynamically calculate rent and utilities costs based on past monthly billing data, and generate invoices accordingly

RediSpeed | *GitHub*

- Built a lightweight **Redis**-like in-memory key-value store in **C++**, implementing core operations (GET, SET, DEL) and advanced features such as data replication and persistence
- Engineered support for streams and transactions along with INCR, MULTI, EXEC, and TYPE commands
- Implemented a TTL (Time To Live) mechanism for automatic key expiration, and engineered a robust TCP server to handle concurrent client connections, ensuring efficient and scalable data management

Automation Programs | *GitHub*

- Implemented bulk editing for **Todoist** tasks based on due date, priority, and project, enhancing task management
- Developed a feature to add bulk editing capabilities to **Trello** boards, improving project organization efficiency
- Created an automation system for time blocking using the **Google Calendar API**, retrieving tasks from Todoist