Idan Gurevich

416-450-5036 | idan.gurevich@gmail.com | LinkedIn | GitHub | idangurevich.dev

Professional Experience

Advanced Micro Devices (AMD) Markham, ON
Incoming Embedded Firmware Engineer Intern May 2026 – Sept 2027

WDI Wise Device Inc. Vaughan, ON
Junior Software Developer Jan 2025 – Present

- Enhanced the **C++** sensor performance test application by implementing **SNMP-based discovery** protocols and resolving critical defects, ensuring strict adherence to quality specifications while significantly reducing failures.
- Engineered automated testing software for autofocus sensors using **C++** to replace manual QA processes, accelerating the validation of sensor calibration and performance metrics across diverse operating conditions and environments.
- Architected a next-generation camera interface and DLL in **C++** for confocal microscopy systems, enabling advanced imaging capabilities with robust API integration and comprehensive error handling for hardware communication.
- Administered **70+ Jenkins projects** and engineered automated integrations between **SVN, JIRA, and Jenkins** via post-commit webhooks, reducing integration errors by **30%** through the implementation of validation checks.

Software Co-op Student May 2024 – Jan 2025

- Supported the maintenance of large-scale Jenkins projects across multiple teams, troubleshooting complex CI/CD pipeline issues to ensure reliable software delivery processes and maintain high development across the organization.
- Developed **Groovy scripts** for Jenkins project migration and collaborated with QA engineers to triage and resolve **100+ defects**, strengthening release delivery timelines through systematic defect resolution in legacy components.
- Designed and deployed automated GUI testing scripts using **AutoIt** to validate legacy Windows applications, reducing manual regression testing time by **20%** and ensuring software stability across diverse operating system environments.

Technical Skills

Languages: C/C++ (C++17/20), Python, Java, C#, Groovy, SQL, Bash
Embedded & Robotics: STM32, OpenCV, PID Control, Sensor Fusion (Kalman), MuJoCo, UART/I2C, ImGui
Cloud & DevOps: Kubernetes, Docker, Jenkins, AWS, NATS, Redis, GitHub Actions, OpenTelemetry, CI/CD
Tools & Platforms: Git, SVN, CMake, JIRA, Visual Studio, Linux (Ubuntu/Embedded), AutoIt

Selected Projects

Hardware-in-the-Loop Robotics Simulator | *C++, Python, MuJoCo, STM32, UART* 2026

- Engineered a bidirectional hardware-in-the-loop system syncing **MuJoCo** physics simulations with **STM32** firmware via a custom binary serial protocol with CRC-8, bridging the sim-to-real gap for autonomous robotic arm validation.
- Implemented a **50Hz real-time control loop** in **C++20** featuring PID position algorithms and filter sensor fusion to continuously correct simulation states against physical telemetry, ensuring synchronization with sub-50ms latency.
- Developed a robust Python middleware layer using **pySerial** to manage asynchronous communication states, providing a real-time 3D telemetry dashboard that visualizes joint angles and error metrics to facilitate immediate debugging.

AeroForge: Vision-Based Drone Control Framework | *C++20, OpenCV, PhysX, YAML* 2025

- Developed a cross-platform **C++20** drone control framework (Windows, macOS) featuring modular architecture and safety mechanisms including configurable geofence boundaries, hold-to-enable controls, and stop functionality.
- Implemented high-performance template matching and Kalman filtering using **OpenCV** to achieve sub-frame latency for real-time object detection, 3D pose estimation, and PID-based control during autonomous flight maneuvers.
- Built a professional debug HUD using **ImGui** to visualize real-time telemetry and implemented a **YAML-based configuration system**, allowing for rapid tuning of PID coefficients and parameters without requiring recompilation.

Distributed Multiplayer Matchmaking Platform | *C++17, Python, Docker, NATS, K8s* 2025

- Designed a production-grade distributed matchmaking platform using **C++17** and **Python** microservices, supporting **10,000+ concurrent players** with sub-100ms latency through MMR-based dynamic queuing and region constraints.
- Developed backend services for session orchestration and lobby management using **NATS** for asynchronous communication, Redis for caching, and the OpenTelemetry stack (Prometheus, Grafana) for system monitoring.
- Orchestrated the containerized deployment of microservices using **Kubernetes** and **Docker**, implementing policies based on active player load to ensure availability and seamless performance during competitive traffic periods.

Education

Toronto Metropolitan University Toronto, ON
Bachelor of Computer Science *Expected May 2027*

Relevant Coursework: Autonomous Mobile Robotics · Operating Systems · Unix C/C++ · Computer Organization I/II
Discrete & Data Structures · Computer Security · Artificial Intelligence · Linear Algebra