

Harsh Verma (he/him)

✉ harshkardam246@gmail.com | 🌐 heyharsh.me

🐙 github.com/harshverma27 | 💼 linkedin.com/in/harshhvermaa

SUMMARY

Electrical Engineering student at NIT Hamirpur and GNOME contributor, working mostly on test infrastructure for GIMP and dogtail. Built GIMP's first end-to-end UI tests on headless Wayland, and a hermetic test session package that shipped as the headline feature of dogtail 2.1.0. Credited by Red Hat for two GIMP memory-safety CVEs. Outside GNOME, builds Nucleus, a Rust toolchain for STM32. Also contributed to gnome/gtk, gnome/gegl, and meson/meson.

EDUCATION

- **National Institute Of Technology** Hamirpur, India
Bachelor of Technology - Electrical Engineering *July '24 - May '28 (Expected)*

TECHNICAL SKILLS

- **Languages:** Rust, Python, C, C++, Kotlin, SQL, Bash
- **Embedded & Hardware:** Embedded C, OpenOCD, PCB Design, Power Management Circuits, SBCs
- **Testing & CI:** dogtail, AT-SPI, Unit Testing, GitHub Actions, GitLab CI/CD, CMake
- **Backend & Mobile:** Django, REST APIs, WebSocket, MySQL, Firebase, Android (Jetpack Compose)
- **Tools & Desktop:** Git, Linux, GTK, PyQt6, LSP, CLI Design, FFmpeg

OPEN SOURCE EXPERIENCE

- **GNOME** [GitLab \(@harshverma\)](#)
Active Contributor *February '26 - Present*
 - **Featured Contributor - GIMP:** Earned recognition in the official [GIMP GSoC 2026 Midpoint Progress](#) blog post for Wayland UI test automation, CI-Fairy commit-format integration, and About dialog improvements.
 - **End-to-End UI Testing - GIMP:** Built *12 end-to-end UI tests* that drive real GIMP windows through the accessibility tree on a headless Wayland session, integrated into GitLab CI, an approach never before attempted in the project.
 - **[CVE-2026-97185 \(CVSS 7.8\)](#) - GIMP:** Found an out-of-bounds write in the GIMPpressionist plug-in's preset loader, where unvalidated vector indices from untrusted preset files could corrupt fixed-size arrays, affecting RHEL 7 through 10. *Credited by Red Hat.*
 - **[CVE-2026-90947 \(CVSS 7.8\)](#) - GIMP:** Hardened the lighting plug-in's preset loader against a buffer overflow, clamping an unvalidated light count read from untrusted preset files before it could write past the fixed-size array. *Credited by Red Hat.*
 - **Unit Testing - GIMP:** Built a Unit Testing Framework from scratch for the core GIMP library, *covering 15+ files*; designed cases for core functionality and edge conditions, integrated the suite into GitLab CI, and surfaced a previously unnoticed environment-variable bug through test analysis.
 - **About Dialog - GIMP:** Added a version-information display to the About dialog and crash reporter with a one-click copy button, giving users and bug reporters an exact snapshot of build and environment details.
 - **Blur Strength - GIMP:** Replaced traditional weighted blur algorithms with Gaussian-distributed kernels and refined the kernel specifications to match user expectations.
 - **CI Optimization - BABL & GEGL:** Diagnosed a GitLab CI flaw that let a single commit spawn two parallel pipelines and authored the fix, *cutting CI resource usage by 50%*.
 - **Commit Log Validation - GIMP:** Authored a check-commit-log CI job that validates commit titles and descriptions on pull requests, enforcing project commit-message standards across every merge request.
- **dogtail - GUI Test Automation Framework (under Red Hat)** [MR \(dogtail\)](#)
Contributor *August '26 - Present*
 - **v2.1.0 Release:** Delivered the new hermetic package as the headline feature of the upstream *dogtail 2.1.0* release (August '26).
 - **Hermetic Test Sessions:** Engineered a hermetic package that runs the framework's UI tests with no desktop session, bringing up a private D-Bus, at-spi2-registryd, and headless Mutter on demand with input via Mutter's RemoteDesktop API, so tests run in CI where GNOME Shell is unavailable. Submitted upstream across *14 files*.
 - **Wayland Input Fixes:** Debugged and fixed two Wayland input defects surfaced by the new session: keys stuck after release, and pointer motion re-anchoring mid-gesture.

PROJECTS

- **Nucleus** - *A modern STM32 developer platform (v0.2.0 released, [YouTube](#))*: Engineered a CLI-first alternative to STM32CubeIDE, a *9-crate Rust workspace* that reduces a **3 GB** Eclipse IDE to a **10 MB** binary. Stored hardware configuration in a single version-controlled `stm32.toml`, validated by a constraint solver across pin/AF conflicts, clock-tree legality, DMA stream arbitration, and NVIC priority. Built a dual-backend HIL test runner that executes the same firmware against QEMU and real silicon over SWD, plus a lockstep mode that reports the first point of sim/silicon divergence. Also authored an LSP server for live editor diagnostics, a fuzz-tested ITM/CoreSight decoder streaming over WebSocket to a React/Canvas dashboard, and a VS Code display extension. Official board support: **STM32F446RE + STM32F411RE**.
Tech Stack: Rust, TypeScript, React, CMake, OpenOCD, QEMU
- **SparkIDE** - *A beginner-friendly block-based IDE for Arduino (v1.0.0 released)*: Launched a lightweight, visual IDE that brings Scratch-like block programming to Arduino development on Linux. Wrapped Google Blockly in a native PyQt6 desktop shell, loaded offline through QtWebEngine, so beginners can drag and drop blocks and watch the corresponding Arduino C++ generate live in a side panel. Authored **50+ custom blocks** covering the standard Arduino library (digital/analog I/O, timing, serial, variables, logic, and loops), and integrated `arduino-cli` for board discovery, one-click compile, and flashing, with build logs streamed into a docked output console.
Tech Stack: Python, PyQt6, JavaScript (Blockly), arduino-cli
- **Kissan-Connect** - *An Android-based marketplace app for farmers*: Developed an Android marketplace app with Jetpack Compose connecting farmers directly to consumers. Built the backend on Django for core application logic and the Android client's API, with Firebase handling authentication, storage, and product listings.
Tech Stack: Python, Kotlin, Jetpack Compose, Firebase, Django

VOLUNTEERING EXPERIENCE

- **Society for Promotion of Electronics Culture (SPEC)** *NIT Hamirpur*
Technical Lead *August '26 - Present*
 - **Breakout Brigade 5.0**: Led a hands-on hardware workshop for **300+ freshers** at NIT Hamirpur, guiding first-year students through practical electronics.
- *Executive* *August '25 - May '26*
 - **Electrothon 8.0**: Managed the 8th iteration of an MLH-backed hackathon for **500+ students** across multiple colleges, overseeing technical and administrative logistics end to end.
 - **Breakout Brigade 4.0**: Directed a hands-on hardware workshop for **120+ students** at NIT Hamirpur; attendees rated it the best of the four workshops held.
- *Volunteer* *September '24 - May '25*
 - **Electrothon 7.0**: Organized an MLH-backed hackathon with **2500+ registrations** and administered venue arrangements for 500+ students from nearby colleges.

PATENTS

- **Router Attachment and Method for Intelligent Network Traffic and Environmental Monitoring**: Co-invented a router-attached device that simultaneously monitors network traffic and the surrounding environment. It detects suspicious domains and abnormal traffic spikes while onboard sensors register smoke, flame, and heat, then throttles or blocks risky traffic and alerts the owner on any anomaly. Filed by NIT Hamirpur as Indian patent application IN202611081064 A1, published August 2026.

HACKATHONS

- **HackSecure** - *NITH*: Developed an Android cybersecurity application that let users quantify all data entering and leaving the device, engineering a custom network tunnel and an on-device VPN to capture and inspect the traffic.
- **Rekkathon** - *NITH*: Engineered a portable network-security device that protected against multiple cyber attacks (MITM, DDoS, phishing), blocked network-wide ads via private DNS routing, included fire and smoke sensing, and provided live video monitoring for the admin.
- **Hack 5.0** - *NITH*: Created a Chrome extension that converted live sign language into readable English using per-frame hand recognition.

CERTIFICATIONS

- **HackerRank Python Developer Certification**: [Certificate](#) (February '25)
- **HackerRank SQL Developer Certification**: [Certificate](#) (January '25)