

Oleksandr Kovalov

<https://www.linkedin.com/in/xairaven>

Location: Kropyvnytskyi or Kyiv

Email: oleksandr.kovalov.work@gmail.com

Mobile: +38 095 736 51 60

GitHub: github.com/xairaven

Age: 22 (born 2003)

SKILLS

- **English:** B1 (Spoken & Written), B2-C1 (Reading & Listening)
- Rust (*Primary Language*), C# (*WPF*), Java, C, Bash, Docker, Linux (*Ubuntu Server, CentOS, Oracle Server*), OOP, SOLID, SQL (*SQLite, MS SQL, MySQL, MariaDB*), Python, Apache, Networking, Git, WASM, Algorithms & Data Structures, UML, Github Actions.

EDUCATION

- **Igor Sikorsky Kyiv Polytechnic Institute** Kyiv
Bachelor, Computer Science – Digital Technologies in Energy *Sep. 2021 – Jun. 2025*
- **Igor Sikorsky Kyiv Polytechnic Institute** Kyiv
Master (Currently Studying), SWE – Computer Systems Software Engineering *Sep. 2025 – Jun. 2027*

EXPERIENCE

- **KPI-Craft** Kyiv
System Admin, Owner *Oct 2024 - Present*
 - **System Administering:** Deployed a Linux server, handled monitoring, backups, set up plugin system.
 - **CEO:** I founded the project, managed user support & advertising of it.
 - **Security:** Successfully repelled several attacks on the infrastructure.

PROJECTS

- **xailyser – Rust, libpcap, nom, WebSocket, egui:**
A modular deep packet inspection platform built in Rust, comprising a core DPI library for packet capture and protocol parsing (libpcap + nom), a multithreaded backend server that aggregates traffic statistics and streams JSON frames over WebSockets, and a cross-platform egui desktop client displaying real-time charts, device aliases, OUI vendor lookups, and packet details, demonstrating expertise in systems-level Rust, network protocols, concurrency, and UI development.
Link: github.com/xairaven/xailyser
- **xArpChat – Rust, Networks:**
A lightweight peer-to-peer chat application written in Rust, leveraging ARP (Address Resolution Protocol) for local network communication. It features message broadcasting without requiring a centralized server, demonstrating proficiency in low-level networking and Rust's concurrency model. Used Cursive as TUI library. Also used PNet, Crossbeam, smaz, thiserror libraries. Made Github Actions workflow for building & creating releases.
Link: github.com/xairaven/arpchat-rs
- **FractalRenderer – Rust, WebAssembly:**
A Rust-based visualization tool for generating IFS (Iterated Function System) and L-System fractals, originally two separate university lab projects merged into a single codebase. It offers both native GUI via egui and browser-based rendering through WebAssembly, highlighting your experience with Rust's frontend ecosystems and cross-platform deployment.
Link: github.com/xairaven/FractalRenderer
- **Blood Bank Management – Linux, Apache, Docker, Ansible (Not my project, just deployment):**
Blood Bank Management System using PHP, MySQL, HTML, CSS3 + Bootstrap. Deployed it on two local virtual machines in several variants: Manually started Apache on the first and created MySQL DB on the second; Docker + Docker-Compose variant; Configuration set up with Ansible.
Link: github.com/xairaven/BBMS-Deploy

CERTIFICATES

- **Linux & Network Administration:** PortaOne, 2024
- **DevOps Crash Course:** SoftServe, 2023
- **English Certificate 71/100:** EF Set, June 2023