CV 2025

Sergei Ivanov

Summary

Independent and versatile developer specializing in robust and scalable systems. Passionate about leveraging modern programming paradigms and tools to solve complex problems. Proficient in full-stack development, distributed systems, and CI/CD automation, with a strong focus on high-performance languages like **Rust**.

Education

Education	*Start*	*End*	*Location*	*School*	*Status*
Secondary	09.2007	06.2016	Narva-Jõesuu	School nr. 5	Completed
Upper-Secondary	09.2016	06.2019	Narva	Upper-Secondary School nr. 1	Completed
Tertiary	09.2019	05.2021	Tallinn	TalTech	Uncompleted
Undergraduate	09.2023	06.2026	Narva	Narva College of University of Tartu	In-progress

Certifications

- **Foundational C# with Microsoft** *Issued by FreeCodeCamp, November 14, 2024* Verification Link

---

Skills

Programming Languages & Frameworks

- **Rust, Python, Kotlin, Deno, Next.js, Lua, Zig, JavaScript/TypeScript**
- **Web**: Next.js (Server Actions), Axum, FastAPI, Vercel
- **Database**: PostgreSQL, SQLite, MongoDB (requirements)
- **APIs**: REST, gRPC, WASM

Tools & DevOps

- **Version Control**: Git, GitHub
- **Containerization**: Docker
- **CI/CD**: Custom Lua script for package versioning
- **Project Management**: Jira, GitHub Projects

System Design & Architecture

- **API Communication**: Implemented REST and experimented with gRPC and TLS 1.3 for secure communication.
- **Microservices**: Practiced service-oriented architecture with Deno and Rust.
- **Packaging**: Created a Python package (`TestPyPIP`) and Rust library crate for code reusability.

Testing & Quality Assurance

- **API Testing**: Manual testing, Postman for REST APIs.
- **Debugging & Analysis**: Wireshark for network analysis, internal logging, and terminal error analysis.
- **Frontend Testing**: Knowledge of Cypress for automated frontend testing.
- **Testing Methodologies**: Experience with unit and integration testing.

Projects

1. High-Performance API & Microservices Exploration (2025)

- **Situation**: Identified the need to build a high-performance, secure system to handle microservices communication.
- **Task**: To develop a proof-of-concept API using Rust and Deno, exploring different communication protocols and secure data transfer.
- **Action**:
  - Designed and built a REST API using **Deno** and a high-performance backend using **Axum (Rust)**, exchanging data via Server Actions.
  - Explored and implemented **gRPC with TLS 1.3** to secure communication between a Rust client and a Rust API.
  - Successfully debugged and resolved incompatibility issues between Deno and the gRPC/TLS setup by isolating the problem and proving functionality with a Rust client.
  - Managed sensitive files (`.key`, `.pem`) with `.gitignore` to maintain security.

- **Result**: Gained practical experience with advanced protocols (**gRPC, TLS**) and high-performance languages (**Rust**), proving the ability to solve complex system-level problems and maintain secure configurations.

2. CI/CD Pipeline for Python Package (2025)

- **Situation**: Needed an automated system to manage the versioning and deployment of a personal Python package.
- **Task**: To build a simple, reliable CI/CD pipeline using familiar tools without relying on heavy frameworks like Jenkins.
- **Action**:
  - Created a custom **Lua script** to extract and automatically increment the package version in `__init__.py`.
  - Configured a Git pipeline to trigger on the `release` branch, executing the Lua script and pushing the new version.
  - Practiced package management and dependency isolation using `requirements-dev`, `requirements-db`, and `requirements-api` files.
- **Result**: Successfully automated the versioning process, demonstrating a foundational understanding of CI/CD, DevOps, and scripting for automation.

3. Full-Stack Web Application Deployment (2024)

- **Situation**: Wanted to build and deploy a full-stack application to production, integrating different technologies.
- **Task**: To develop a functional web app with a backend and deploy it to a cloud service.
- **Action**:
  - Developed a web app using **Next.js**, a **Kotlin** backend, and a **Docker** container for easy deployment.
  - Deployed the container to AWS, demonstrating the ability to manage and provision cloud resources.
  - Used **GitHub Projects** and a Kanban board for effective task management and project tracking.
- **Result**: Successfully deployed a working application, gaining critical experience in full-stack integration, Docker, and cloud deployment, which provided valuable feedback on user experience and product maturity.

Contributions and Projects

- **Game Performance Analysis**: Used **OpenHardwareMonitor** and **Intel VTune** to analyze a game's performance, identifying and reporting on caching and memory usage issues to the developers.

- **Open Source Contribution**: Identified and reported incompatibility issues with **NumPy, Pandas, and SciPy**, collaborating on GitHub to monitor the resolution.

Speaking Languages

- **Russian**: Native
- **English**: B2 (speaking, writing)
- **Estonian**: B2 (speaking, writing)

Certified through formal examination