

Ruslan Golovinskii

 Auckland, New Zealand ·  +64 210 210 0435 ·  golxzn@gmail.com ·  golxzn ·  golxzn

About me

Initiative senior C++ software engineer with **5+ years of experience** in game development, specializing in **performance optimization, engine development, and build systems**. At MYTONA, I worked across multiple games and internal projects, **optimizing for iOS, Android, and Emscripten**, while developing tools, libraries, and engine features.

I have extensive experience modernizing large C++ codebases and build infrastructure. For example, I migrated 2 legacy projects to modern CMake, reducing **Windows build times from 45 to 2 minutes** and **Android builds from 80 to 4 minutes**. I enjoy solving complex technical problems, simplifying systems, and building tools that make other engineers' lives easier.

Work Experience

MYTONA, Remote

Jun 2021 - Oct 2026

full-time C++ Software Engineer

"Cooking Diary" ~2 months

- Based on previous success, migrated Spine library from 3.6 to 4.2 version and convert **around 2700 spine assets**;
- Under direct CEO supervision, *rapidly* developed game features to increase ARPU and DAU in a lead role;
- Ported **legacy 10 y.o.** mobile game engine to **Linux** with Wayland/X11 backends;

full-time C++ Software Engineer

"Ravenhill" ~13 months

- Under direct CEO supervision, rapidly replaced the foundation of game and changed the core logic to support top-view city features.
- Alongside with Head of Animation and animation team, migrated Spine library from 3.6 to 4.2 version, made a assets conversion tool-set to upgrade **around 1000 spine assets**, added in-game viewer to accelerate QA check; all to support physics in spine animations;
- Initiated and organized small team to replace loading screen to fast transition to increase DAU;
- Integrated **Firestore Performance Monitoring** for iOS and Android devices in **one day**;
- Discovered, analyzed, and optimized the game loading time from **5 to 1 second**;
- Participate in migration from the **legacy project** to a **modern internal game engine**;
- **Highlighted the performance issues** by integrating Tracy library and guided QA and Dev teams within 3 days;
- **Accelerate JSON to BSON conversion** from **~5 minutes** up to **30 seconds**.

"Chief & Friends: Cooking Game" ~1 month

- Determined complex game systems relations and parallelize their initialization, which **amplified system initialization** 3 times from **1 second** to **~300 milliseconds**;
- **Profiled project building process** and enhanced "CMake" which reduced whole build time from **14 to 7 minutes**.

"New Internal Project" ~10 months

- **Setup new cross-platform project within few weeks** using modern CMake and new game engine;
- **Made a Lua script-based multi-threaded dialog system** independent module by oneself **within 2 weeks**, which allows non-programmers to set up a complex dialog trees and loops within few minutes;
- Reinvent the in-game widget system, which makes really complex windows logic as **easy as magic** through a **node-based Figma-like editor**, so designers could use it with no pain;
- Tightly work with engine team on new internal game engine to bulletproof and fulfil their solutions;
- Delivered a huge amount of robust and maintainable utilities, tools, and modules for our team. I really love to make my teammates' life easier and happier!

"Cooking Diary" ~3 years 5 months

- **Optimize floating point parse** using fast_float in the internal game engine, which escalated runtime resource-intensive JSON loading **up to 8 times among all the internal projects**;
- **Made a seamless FlatBuffers integration** and adopted 4 biggest and around 100 smaller intensively used config files, which **boosted the game loading to 2 seconds**. Automated source **code generation** based on the FlatBuffers' schemes in CMake. Provided robust tool to convert JSON to binary format;
- Integrated the **node system** to **optimize content integration**, additionally enforced the Godot-like editor;
- Implemented the Unreal Engine like **blueprint system**, focused on the multithreading and performance;
- **Migrated build system to CMake**, slashing Windows builds from **45 to 2 minutes** and Android from 80 to 4 minutes using unity build; retired legacy IDE projects; authored documentation; tripled throughput and cut **Dev→QA→Dev cycles from 120 to ~20 minutes**;

- Refactored and multithreaded **more than 20 internal tools** to make them run **from minutes to seconds**;
- Ship the Google-SignIn to the iOS platform, which increased the DAU for 3%.

Sky-Technologies, Saint-Petersburg, Russia

Jun 2020 - Jun 2021

full-time C++ Embedded Linux Engineer

- Designed and streamlined low-level Linux-based Point-Of-Sell (Castles) terminal OS modules in C++98;
- Accomplished the QR Code Fast Payment System method on terminals and transaction log manager;
- Authored shared libraries and communication protocols for IC integration, which were widely used by our clients.

Skills

- **Programming Languages:** C++20/17 (STL, Boost), Python, Lua, scripting languages, pwsh, bash.
- **Build & CI:** CMake (Advanced), Git (Advanced), LLVM, Gradle, SVN, Jenkins, CI/CD pipelines;
- **Graphics & Engines:** OpenGL ES 2.0/3.3, Vulkan, Metal; **Engines:** Godot, Unreal Engine 5, Unreal-style scripting;
- **Tools & Libs:** Tracy, Valgrind, RenderDoc, clangd, CppCheck, FlatBuffers, fast_float, ENTT, RTTR, simdjson;
- **Concepts:** Data Oriented Design (DOD), Object Oriented Design (OOP), Domain Oriented Design (DOD), Design Patterns, Multi-threading, Benchmarking, Architectures.
- **Environment:** Environment: Any OS, Neovim, Visual Studio Code, Android Studio, XCode, Slack, Wrike.

Education

YOOBEE College, Auckland, New Zealand (Upcoming)

Nov 2026 – Nov 2027

Master of Software Engineering

Polytechnic College of Municipal Economy, Saint-Petersburg, Russia

Jan 2018 – Jul 2022

Programming in computers' systems, Associate's degree

University of Colorado Boulder, Online

Aug 16, 2025

Embedded Software and Hardware Architecture

Miscellaneous

- Language:
 - English** - TOEFL iBT Score 5.0 (C1);
 - Russian** - native;
- Actively participated in internal company's conferences with educational presentations based on actual work situations;
- Founded and officially launched the college's first club, dedicated to programming with modern C++. I led sessions twice a week for a year, delivering structured lessons, practical coding tasks, and visual presentations. This experience ignited my long-term passion for teaching, which I continue today by mentoring a few students privately;

Latest Personal Projects

[lukewarm-show]

<https://github.com/golxzn/golxzn.github.io>

Modern 3D rendering using Physic-Based Rendering in pure JavaScript. You could see it by yourself in life demo!

- Unreal-Engine 4 Physic-Based Rendering Pipeline;
- Self-implemented GLTF model files parser

lox-lang

<https://github.com/golxzn/lox-lang>

Simple and fast scripting language made in C++20 and Data Oriented Design.

- Data-Oriented and Fast parsing and run-time execution;
- Efficient Abstract Syntax Tree implementation.

esp32-s3-game

<https://github.com/golxzn/esp32-s3-game>

Using Espressif SDK and FreeRTOS, I tried to make a game in micro-controller using only documentation and my own optimizing skills.

- Double-Buffered Rendering using TFT 3.5 ili9486 through Dedicated GPIO (I'm planning to migrate to Octal-SPI to reach 60 FPS);
- SPIFF file system;
- Multiple simultaneous WAV file support using LEDC and PWM.
- DualSense® controller support though HID over USB;

You can find much more projects on my github!