Sultan Luchezarniy

Lead/Senior C++ Developer

xottab-duty@yandex.ru| github.com/xottab-duty | t.me/xottab_duty

Summary

Experienced C++ developer and team lead with a strong background in game engine development, optimization, and cross-platform porting. Over **10 years** of hands-on **experience** in the games industry, including leading teams, modernizing legacy codebases, and delivering stable releases of well-known titles. Skilled in **C++17/C++20**, low-level systems programming, and performance optimization. Proven ability to manage teams, coordinate with multiple departments, and deliver technically complex projects on time and with high quality. Passionate about clean architecture, maintainable code, and empowering teams to achieve their best results.

On my sociable and creative side, I love building communications between people, participate in discussions with a cup of tea and jokes. I've built and nurtured a thriving community of players, modders, and developers.

Seeking to leverage my skills in developing high-quality AAA games.

Commercial experience

C++ Team Lead July 2024 – August 2025
European company (NDA)

Project:
Remaster of the game, updates, and fixes for correct operation on modern platforms.

Languages:
C++.

Responsibilities:
- Leading a team of 5 developers: planning tasks, distributing workload, controlling deadlines and quality.
- Interacting with various teams and departments: conducting code reviews, finding optimal solutions, implementing features requested by the Gameplay team, timely responding to issues discovered by the QA department, proposals and remarks.

Tasks:
- Updating, fixing, refactoring, and optimizing legacy code.
- Debugging, identifying reasons for incorrect operation of legacy code on modern platforms.
- Applying modern C++17/C++20 solutions, achieving correct operation on both outdated and modern platforms.

Achievements:
- Stable release on Steam, the game "went gold" practically without bugs, i.e., despite major internal changes, during development, it was possible to "preserve everything as it was", which was highly appreciated by players.
- Full integration with Steam: support for Workshop, achievements, multiplayer.
- Support for 4K resolution, stable operation on modern hardware and Windows 10/11.

System Engineer March 2024 – July 2025
NTC IT ROSA

Project:
Mobile version of ROSA Linux distribution.

Languages:
C, QML.

Responsibilities:
- Implementing features, debugging, bug finding and fixing, logging in JIRA.
- Working with system drivers and user interface.

Software Developer August 2022 – February 2024
European company (NDA)

Project:
Porting a well-known game in the First-Person Shooter genre to new platforms.

Languages:
C++, CMake, Lua.

Responsibilities:
- Managing a team of 2 people, mentoring, 1-on-1.
- Decomposing, distributing, and tracking tasks.
- Mentoring and developing team members: conducting code reviews, discussing solutions, teaching best practices.
- Managing tasks in Jira.

Tasks:
- Setting up CMake and MSBuild configurations.
- Porting the game engine to new platforms.
- Refactoring and optimization using C++17.
- Implementing multithreaded code using mutex, atomic.
- Integrating the ImGui UI library and Tracy profiler.
- Implementing gamepad controls.
- Profiling, identifying performance bottlenecks and their elimination, optimization.

Achievements:
- Reduced the amount of routine code maintenance work by 2 times and optimized the time spent on tasks, thanks to adopting an architectural decision.
- Refactoring the DirectX rendering architecture, after which it became possible:
- Rapid prototyping and implementation of new graphics techniques.
- Double performance increase.

Software Developer November 2020 - July 2022
Sperasoft

Projects:
- Saints Row (2022)
- Madden NFL 22
- Mortal Kombat Mobile and Injustice 2

Languages:
C++, Lua, ActionScript.

Responsibilities:
- Working within Agile, Scrum, and Kanban methodologies.

- Participating in and conducting code reviews.
- Debugging, ensuring stability, and fixing bugs.
- Refactoring and optimization.
- Interacting and coordinating with both my team members and external teams.
- Logging in Jira.
- Maintaining documentation in Confluence.

Tasks:
- Implementing gameplay logic using Blueprints.
- Developing the character customization system interface with a rich set of appearance settings for the character and the ability to share presets with other players.
- Implementing user action telemetry in the UI.
- Updating the Unreal Engine to a newer version.

Achievements:
- The character customization system was positively received by players, and thanks to it, many high-quality character presets were created.
- Updating Unreal Engine from version 4.19 to 4.27 yielded several positive results:
- Increased the performance of the 3D Art team, thanks to new features and tools in Unreal Engine that became available.
- Reduced human resource costs for code support, thanks to a reduction in engine modifications, including backports, which became unnecessary.
- Improved positive user experience, thanks to a reduction in bugs and crashes.
- Featuring in Google Play, thanks to the fact that the number of users without crashes reached 98%.

Side projects

Lead Developer
OpenXRay
https://github.com/OpenXRay/xray-16

August 2015 – December 2015,
January 2017 – now

Project:
An improved version of the X-Ray Engine, the game engine used in the world-famous S.T.A.L.K.E.R. game series by GSC Game World.

Languages:
C++, CMake, Lua.

Responsibilities:
- Managing and leading an open-source project: planning and prioritizing features, coordinating contributors, conducting code reviews, and maintaining code quality.
- Handling full development cycle: from architecture design and implementation to testing, debugging, and release management.
- Communicating with the community: responding to bug reports and crash logs, providing technical support, maintaining documentation, and posting development updates in social networks.
- Ensuring high code quality and performance through continuous profiling, optimization, and refactoring.
- Maintaining compatibility and stability across multiple hardware and operating systems.

Tasks:
- Refactoring and modularizing engine subsystems, improving its architecture, fixing design flaws (including recognized by original developers).
- Porting to various new platforms.
- Implementing features in all possible areas including gameplay, scripting, AI, UI, sound, physics, renderer.
- Profiling performance and implementing optimizations, parallelizing the code, making it multithreaded.
- Managing issue tracking, reviewing pull requests, merging contributions, and ensuring overall project stability.

- Performing QA: testing new features, reproducing and fixing bugs, analysing crash reports.
- Maintaining project visibility and user engagement through community interaction and support.

Achievements:

- Built and maintained a sustainable and well-known friendly community around OpenXRay, contributing to the larger S.T.A.L.K.E.R. community and helping players, modders, and developers.
- Gathered a team of motivated developers.
- Ported the engine to Linux, macOS, *BSD, Haiku, and multiple architectures (x64, ARM, ARM64, E2K, PPC64LE), ensuring stability and performance across platforms.
- Brough OpenGL rendering support in collaboration with peers, enabling the engine to run on non-Windows platforms.
- Implemented shader cache support in the OpenGL renderer (using GL_ARB_get_program_binary), reducing game startup times by 2-3x.
- Implemented DirectX 10 feature level support within the DX11 renderer, simplifying maintenance and reducing duplicated code.
- Improved renderer portability by abstracting graphics API calls and separating frontend from backend logic.
- Implemented a low overhead custom work-stealing task scheduler, improving engine multithreading efficiency and responsiveness.
- Added full gamepad support and implemented an automatic UI focus system, reducing manual asset editing for UI designers.
- Developed a real-time UI debugger and weather editor (based on ImGui), increasing productivity of UI and level designers.

Education

Software Engineer in Applied Informatics, Tyumen State University, 2015-2019

Language

Russian – Native

English – B2 Upper-intermediate

Polish – A2 Elementary

Skills

C++20, Lua
C# (beginner)
UE Blueprints (beginner)
ImGui, Tracy

Debugging
Profiling
Optimization
Code review
Tasks decomposition, distribution and tracking

STL, multithreading, atomic
DirectX 9, DirectX 11, OpenGL
OOP, Algorithms, Data structures