

Alexandre Guerard

Oxford, UK • alex@guerard.us • 07423 646697 • [GitHub](#) • [LinkedIn](#)

Software engineer with 13 years of professional experience that includes contributions to products delivered to millions of users across multiple platforms. Pragmatic and outcome-driven, with a proven ability to navigate large codebases and leverage expertise to lead and enforce best practices with the goal of maintainable, testable, and scalable code. Effective communicator and collaborator, skilled in managing stakeholder relationships and partnering with product owners and contributors across disciplines and at all career levels. Driven by an interest in self-improvement through personal projects, courses and self-study, with current interests spanning compilers, language design, and programming paradigms.

EXPERIENCE

Independent Work

July 2024 - present

Pursuing independent software development work while relocating to the UK.

Riot Games

Dec 2019 - June 2024

Riot Games operates the multiplayer online video-games League of Legends (LoL) and Teamfight Tactics (TFT).

Staff Software Engineer (Apr 2023 - June 2024)

Senior Software Engineer (Apr 2021 - Apr 2023)

Software Engineer (Dec 2019 - Apr 2021)

Hired with work focused on internal developer tools for LoL and TFT and promoted to senior and staff roles with impact at the studio level.

- Named Tech Lead for the Content Workflow team, responsible for the internal tools and processes used by artists and game designers to create the content (visual effects, UI, Maps...) for LoL and TFT.
- Selected to lead a new Player Experience team formed after a reorg, with the team focused entirely on UI workflows and UI-related code within the game engine.
- Played key role in the launch of TFT in Southeast Asia, a top corporate priority for 2023.
 - Drove efforts to update game engine capabilities (e.g. UI layout, rendering) to fully support tablets.
 - Led development of new tools for use by UI designers to create and validate UI components that leveraged the new rendering and layout capabilities.
- Led the modernization of automated processes that extract/export/import in-game text for language localization using C++, Python, and Jenkins.
 - Implemented monitoring capabilities with Slack integration to enable localization teams to identify and resolve issues with no engineering support.
 - Efforts significantly reduced QA team workloads for content validation and freed up engineering resources.
- Collaborated with UI designers and engineering teams to build a new modern toolset for designers using C++ and Qt, featuring a modern experience similar to commercial tools. The project resulted in a promotion.
 - Performed all initial technical tasks and later managed a small engineering team in execution.
 - Coordinated multiple teams from other disciplines to ensure alignment.
 - The new tooling and associated workflows improved designer satisfaction, enabled designers to create UI content with no support from engineering, and simplified creation and previewing of UI content across platforms (e.g. PC, tablet, phone) and languages.
- Deprecated old workflows and tools for multiple disciplines to streamline tooling and simplify codebases for game engine and tools.
- Contributed to development of new tooling in C++ and Qt used for artists to create visual effects and shaders for LoL, expediting development and iteration of new effects.
- Automated several data sanitization and version management processes using Python, Jenkins, and proprietary CI/CD pipeline tools to convert time-consuming and error-prone tasks into reliable and easily monitored jobs.
- Mentored junior and mid-level engineers.
- Served on the technical interviewers pool responsible for interviewing candidates for multiple Riot teams.
- Served in on-call rotation overseeing biweekly patch releases. Triaged technical issues and escalated as needed.

Pretend, LLC.

May 2011 - Nov 2019

Pretend builds macOS/iOS visual effects and editing software for the film industry.

Software Engineer

Hired as one of the first engineers, at the startup stage, on a product development effort.

- Responsible for development of an editing and visual effects software product targeted to small film productions using a mix of C, C++, Objective-C, and UI frameworks (e.g. AppKit, Qt).
- Made significant contributions to the non-linear editing software and a multi-threaded 3D and image processing engine using OpenGL and GLSL.
- Used OpenGL and the Apple CoreAudio library to build audio and video playback rendering.
- Successfully lobbied for and implemented continuous integration processes and tooling that included macOS Server, clang-format, Xcode static analyzer, unit and UI testing with Xcode testing tools, and a bash script to export builds for QA testing.
- Developed procedures to add a test case to reproduce all new bugs identified, preventing reintroduction of old bugs.

Prior experience includes two internships in France focused on video surveillance devices and computer vision for vehicles.

PERSONAL PROJECTS

- **Interpreter** - Wrote an interpreter in Python for a toy scripting language that supports functions, classes, inheritance, and dynamic typing.
- **Compiler** - Built a compiler in Python for a statically typed scripting language that supports functions, with ability to generate code for LLVM or WebAssembly.
- **Ray Tracer** - Wrote a Ray Tracer in C++ with multithreading support to explore different approaches to software development and learn physically-based rendering, color science, and 3D geometry.
- **Cryptopals** - Solved cryptography challenges using Clojure to gain exposure to Clojure, FP, and real-world cryptographic attacks and their implementation.

EDUCATION

"Diplome d'Ingenieur" (MS Electrical Engineering equivalent), Grenoble Institute of Technology - Grenoble, France (2010)

- Specialization in Digital Signal processing with emphasis on hardware and image processing.

BS, Engineering Sciences, Grenoble Institute of Technology - Grenoble, France (2008)

SKILLS

C++, C, Objective-C, Python, TypeScript, JavaScript, Clojure, Haskell, macOS, Linux, Unix, Windows, Emacs, XCode, Visual Studio, git, svn, Perforce, clang, Make, CMake, FastBuild, CI/CD, Jenkins, Qt. Bilingual French and English.