

# Gabriel Lacayo

[gabriel@lacayo.dev](mailto:gabriel@lacayo.dev) [Portfolio](#)  
(702) 530 7674 [LinkedIn](#)  
Los Angeles, CA [GitHub](#)

## Experience

[Ripple Effect Studios](#) | Los Angeles, CA

Software Engineer II December 2023 – Present  
Game Title: [Unreleased Battlefield Game](#) 3 Months  
Engine: Frostbite (C++ Runtime and Pipeline, C# Editor)

Software Engineer I June 2022 – November 2023  
Same project as above 1.5 Years

Software Engineer Intern May 2021 – August 2021  
Game Title: [Battlefield 2042 Portal](#) 4 Months  
Engine: Frostbite (C++ Runtime and Pipeline, C# Editor)

- Improved the Tweakables module for Frostbite Editor in WPF by adding search filters based on name, type, or tag, as well as copy/cut/paste functionality to significantly enhance workflow. Took examples from existing modules as well as feedback from designers and producers to make creating new Tweakables or modifying existing ones much easier in the editor.
- Added live editing of Tweakables so that changes can be made in-editor while the game is running in a debug build using the existing messaging system for live edits. Incentivizes designers to incorporate more Tweakables into their gameplay systems by providing a faster iteration loop, which normally requires having to wait for a cook then running the game again.
- Added memory footprint table of Tweakables in existing IMGUI debug tool to see at a glance how many Tweakables are present in the build and what the most expensive ones are, giving a concrete measurement of their usage.
- Fixed crashes due to single threaded memory with Tweakables. Investigation showed that the memory allocation of Tweakables need only be realm specific since dedicated servers and clients require only one instance. In-proc has both server and client in one process, but this change did not affect the stability of in-proc builds which are developer facing anyways.

## Student Projects

Gameplay Engineer August 2021 – May 2022  
Game Title: [Charon](#) - 3rd Person Boat and Combat Game 10 Months  
Engine: Unity 2020.3 (C#)

- Restructured team workflow to be scene additive due to bottleneck issues from version control. Added functionality that loaded all scenes relevant to main level. Checkouts and CLs from team became more atomic because of better focus on prefabs and individual scene changes.
- Created a [build version package](#) for QA purposes. Automatically populated the latest CL from perforce alongside semantic versioning. Showed in both editor and standalone executables using Unity build pipeline. Helped identify stale QA tickets which previously had no indication.

## Skills

C++, C#, Python  
Visual Studio, Perforce, Jira

## Education

University of Southern California May 2022  
B.S. in Computer Science Games