**Cross Platform Research**

Jason Captor

**I. How does cross-platform software building work? What are the processes involved and what is required when porting a piece of software from one platform to another?**

Cross-platform software development usually starts by choosing an engine or platform with support for multi-platform game development, such as Unity or Unreal Engine. These engines will mostly handle building and exporting a consistent experience across all platforms specified, though due to hardware differences, it is best to tweak things like graphics settings to optimise for each platform.

When taking an existing game or piece of software and porting it to a new console or platform it was not originally designed for, you usually need to replicate the software in a new engine first, which can be an enormous challenge. That is the reason this is done infrequently, and why games are so often made in engines that can support multiple platforms from the outset, so it is much easier to port to consoles or other platforms later on as required.

**What are the standards for cross-platform development and libraries?**

* Decide on the platforms you will be supporting at the outset, and don’t try to add new platforms in later
* Use non-platform specific code wherever possible
* Encapsulate platform-specific code such that code not for a specific platform is hidden from users on that platform

**II. What are some different cross-platform development libraries?**

1. Xamarin: An open source mobile development framework owned by Microsoft. Based on C#, it allows mobile app developers to create one ‘base’ version of an app, and have Xamarin automatically create a version which works on each target platform.
2. Unity: A game engine with native support for mobile, PC and consoles. Unity allows developers to set their target platforms when starting a new project, and provides tools to assist in ensuring games look good and run well on each of the selected target platforms. Console development requires Unity Pro, however.
3. Unreal Engine: A powerful game engine owned by Epic Games. Because Unreal is designed for AAA games, the size of the games created with Unreal tend to be much larger than those created on other engines, but will often feature a higher level of graphical fidelity. Unreal can be used to create mobile games, but is more suited for PC and consoles with more powerful hardware.

**III. What sources did you use for finding out this information? Provide references.**

Kevuru Games. (2021, July 7). Cross-Platform Game Development: When Versatility Matters. <https://kevurugames.com/blog/cross-platform-game-development-when-versatility-matters/>

GameReactor. (2023, February 7). Cross-platform game development. How to make a game for everyone? <https://www.gamereactor.eu/blogs/yuriy+denisyuk/391593/Crossplatform+game+development+How+to+make+a+game+for+everyone/>

Backblaze. (2008, December 15). Brian’s 10 Rules for How to Write Cross-Platform Code <https://www.backblaze.com/blog/10-rules-for-how-to-write-cross-platform-code/>

Reddit (2019, February 17). Cross platform C++ standard library or WinAPI? <https://www.reddit.com/r/cpp/comments/ari0ic/cross_platform_c_standard_library_or_winapi/>

Net Solutions. (2023, April 7). The Ultimate Guide to Cross Platform App Development Frameworks in 2023   
<https://www.netsolutions.com/insights/cross-platform-app-frameworks-in-2019/#best-cross-platform-app-development-frameworks>

Unity. (n.d.). Maximise Multiplatform Game Development <https://unity.com/solutions/multiplatform>