Project: <https://github.com/Terrasean123/Maths-for-games-Project>

merge : <https://github.com/Terrasean123/Maths-for-games-Project/commit/a1b0871656fd437f7389e5ada28b5930a933945f>

A screenshot of a computer

Description automatically generated with medium confidenceA picture containing screenshot, text, software, multimedia software

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidencepull Request: <https://github.com/Terrasean123/Maths-for-games-Project/pull/2>

A screenshot of a computer

Description automatically generated

Problem Statement:

Small and Big Teams alike may find it Tedious to invest time and resource to development of the their video Game whilst also having a separate department for storage, management and deployment of the multiple versions of their project; using all manner of storage facilities such as drives and USB sticks to store the various versions of their Game can waste Time and risk efficiency. And teams may need to hire employees for version management resulting extra costs.

Version control systems such as GitHub, SourceTree and perforce are imperative to increase version control efficiency. Small and large teams to have their software versions stored on server where employees and managers can manage and manipulate the software within as they please.

Companies may choose to use perforce out of preference for file Level commits and alterations or maybe SourceTree for it beautiful User interface design and ease of access repository browsing; however these version Control systems contain their own issues. For example

GitHub is managed and monitored by the dedicated mangers under git resulting in no externalized cost to the team. The software and data is stored in a way that allows employees to access the same data from multiple device’s at no extra costs and allows for the dynamic merging of data from multiple branches of the same File.

Sources:

<https://www.codebasehq.com/blog/an-introduction-to-version-control>

<https://homes.cs.washington.edu/~mernst/advice/version-control.html>

<https://www.what-could-possibly-go-wrong.com/version-control/#cons-1>

<https://about.gitlab.com/blog/2020/11/19/move-to-distributed-vcs/#:~:text=The%20most%20popular%20of%20these,for%20any%20size%20software%20project>.

<https://stackshare.io/stackups/git-vs-sourcetree#:~:text=%22Distributed%20version%20control%20system%22%2C,reasons%20why%20SourceTree%20is%20favored>.

<https://www.anchorpoint.app/blog/git-vs-perforce-for-game-development#:~:text=to%20prevent%20conflicts-,Speed%20and%20stability,upload)%20it%20in%20the%20background>.