**Version Control Software Evaluation Report**

**Math For Games**

**Jack Murray**

**10/20/21**

**Version Control Software:**

The version control software used was: GitKraken and Github Desktop

The client program was: Mjollnir (Unity Project)

**Performance:**

Gitkraken was extremely frustrating to use. Setting it up was a nightmare, especially when trying to access it from different computers. Oftentimes generating a new access token wouldn’t work on the new computer in order to access your old repo. Setting up a new remote every time was also annoying. In addition, Gitkraken doesn’t let you commit your changes while the applications are still open, meaning I had to close all of my work whenever I wanted to commit, which was aggravating.

Github desktop was much easier to use. It allowed me to directly link my account, and therefore my repos, without any hassle. Additionally, it lets you commit and push while the applications were still open. There was a small hiccup when initially setting it up and cloning the repo I already made with Gitkraken, but beyond that Github Desktop was extremely simple to use.

**Evaluation:**

Due to Gitkraken’s requirement of using access tokens and remotes, I can see it being very useful for large projects manned by multiple teams. It would be far easier to manage who is working on what items via Gitkraken over Github Desktop. However, for a small-scale project with 1 or a handful of people, Github Desktop is far easier to use.

**Environmental Considerations:**

If a company has limited data, and a lot of team members are trying to push and pull at the same time, using these resources could be quite time and resource consuming, especially when working with large file types such as mp4.

A self-hosted version control means you have to maintain the server yourself. Keeping it cool, supplying it power, and hiring some IT to make sure it keeps running properly are all costs that are necessary in keeping it running, otherwise you could lose access to your entire repo. The benefits of a self-hosted version control means you can control who has access to it, making security much more reliable.