IS-340 FINAL PAPER

Project Ill-uminate has been a work in progress for some time now. My main goal for this project is to just make video games using open-source software. But there are sub goals that I would like to accomplish with this project as well. Some of those goals is to contribute to open-source software and hardware in some way, whether it be money or code contribution. Other goals include making philosophical stories through entertainment mediums such as movies and games. I want to tell these stories because I would like to carry on the torch and continue to propagate the legends of virtue in hopes of inspiring others. I want to inspire others because I want to live in a world with more educated minds and empathetic hearts. In addition to all of that I want to promote sustainable lifestyles. I for one enjoy breathing clean air, drinking clean water, and eating clean food. Meaning that there are a lot of poisons polluting our environment, and my goal is to reduce some of that poison. The word poison is somewhat vague in this context, but in general a poison is something that either makes you sick or kills you.

Luckily for a lot of us there are many organizations within different industries working on the same goals as I am. There isn't anything inherently unique or new about my project, but rather a combination of already existing things into one project. My project is intended to be operated by a minimum of one person which will be me. If more people share an interest in my project, then it can be scaled up or down according to interest. The projects' ability to scale with relative ease would be due to the resources being used to operate it. Resources such as the use of open-sourced software and renewable energies. By using these two types of resources it removes the need for overall maintenance and costs. Using open-sourced software does not often have fees or purchase requirements, and renewable energy has low maintenance and low long-term costs.

I want to use open-source software because of the low cost of entry if any. Most open-source software is free, and some are very powerful life Blender and Unreal Engine. By using open-source software I can take the project further than trying to pay it. By removing the obstacle of money to work on the project I can have a much bigger return of investment, which will allow me to have more finds to support the other goals of the project. The core open-source software I plan to use in this project are Blender, Unreal Engine, and Godot. Blender is a 3D modeling software with the ability to animate and edit videos. It was used to create a film named "Flow" which went on to win an Oscar for best animated film. It has a large community with users and contributors alike. There are many free resources to learn how to use it and on how to develop its code. The project will use Blender to create the original 3D models of the games or movies to be produced by my community or simply by myself. Once the 3D models are done, they would be exported to either game engines named Unreal Engine or Godot for further development.

The core open-source software that I would like to use for project Ill-unminate is Unreal Engine. It is a video game engine that has been used by many big-time studios and indie game developers. It has received tremendous support and has become a very powerful piece of software. The best part is that it's free to use, but it does have a royalty fee should anyone want to use it. Which is not a too bad and considering that the fee is used to further develop and improve the engine. Which is a huge undertaking and relieves the general user from that burden. But if a user should desire to alter the engine they are allowed to do so because the engine itself is open-sourced, and the source code can be accessed by anyone. However, Epic Games is the primary owner and developer for the engine and ultimately decides what features or modifications get added to the engine.

Epic games keep the majority control of the Unreal Engine, but they are open to contribution from the community. They hold an annual event called “Unreal Fest” where community members gather and showcase their ideas to other community members along with the Epic Games team. That is why I have such a strong interest in using the Unreal Engine, not just because it is a very good piece of software, but because of how engaged the community is with it as well. In addition, there are a ton of free and paid assets that are provided to users and 3D artist in general via the (now called) FAB. Thus, it greatly expedites a user’s ability to start creating games and reduces the cost of either creating or paying for assets. Which makes Unreal that much more appealing for creating games than any other engine now. However, Unreal Engine is not the only game engine available to be used so freely, but one other notable engine is Godot.

Godot has been around for a while now, but it is only recently that is gained popularity and momentum. Mostly due to Unity’s (another game engine) terms of service upset many of its users. What makes got different than most game engines are that is has no royalty fees or any type of purchase necessary. Unfortunately, its UI and overall interface design is not all that great now. But it is improving as it grows in popularity, it has gained great publicity and funding from other development studios and businesses. The reason I bring up Godot is because I wanted to highlight that is has no royalty fees for publishing something with its engine. This is very attractive for long term goals where things are more uncertain. I would like to make contributions to it eventually once my coding skills further develop. I would like to bring it up to the same level as Unreal Engine in some certain features such as ray-tracing and digitized geometry. In addition to these core software pieces there are a number of other software that not inherently free or open source but otherwise play a crucial role in project Ill-minate’s progress.

A key piece of software that would be very useful within project Ill-uninate is Discord because it is a “free” to use chat app that is heavily adopted by the tech communities. Other software includes Affinity’s suite of apps that can create and edit pictures. It is appealing to me because of its one-time purchase fee and no subscription models. Which is another save in cost for long term usage. While Blender has video editing capabilities it is probably better to use a more dedicated piece of software such as DaVinci Resolve because it is a high-quality editing software free to use. It has a paid version with more features than the free version, but the free version is pretty good as it is. And to get some of the videos for editing I plan to use OBS Studio which is a video recording and streaming software. I want to use OBS because it is free and open source. Which is an additional reduction in cost for long term use. There are several other pieces of software I want to use such as Github, Linux, and Gaea. They all have their unique uses and are free to use.

GitHub is a powerful platform for version control and collaboration, widely used by developers to host code, track changes, and work together on projects seamlessly. It supports open-source development and makes sharing and managing code repositories straightforward. Linux, on the other hand, is a versatile open-source operating system known for its stability, security, and customizability. It is widely used in server environments, software development, and even personal computing by those who prefer more control over their systems. Gaea is a specialized terrain design tool often used in game development and visual effects production. It allows users to create detailed and realistic landscapes using a node-based workflow, making it ideal for artists and developers who want to generate complex terrain with a high degree of artistic control. All of these tools are either completely free or offer robust free versions, making

them accessible and valuable resources for anyone interested in technology, development, or digital art.

There are going to be other pieces of software that are going to be used now and ones that are going to be developed in the future that are also going to be used. But they will be used when there is a need for them, but for now only the named software will be used for the initial stages of project Ill-uminate. In addition to knowing how to use the named software I also intend to learn the coding languages that make up that software. Specifically, the languages C++, Python, and Unix. I want to use them because C++ allows for efficient memory management and high-performance applications. Python is known for its simplicity and versatility, making it ideal for rapid development and data analysis. Unix provides a powerful environment for managing systems and automating tasks through shell scripting.

With all these software and languages, I would like to develop and produce mainly video games. But, if the community expands large enough, I would like to produce movies, plugins, and even delve into hardware development. These are all task that will take time and are not explicitly planned for all to be completed. The main goal to complete is just one video game. I hope that it is a good enough success to bring in funding and a community of people with shared interest. If not, then I will try to make another game and so on. At the bare minimum I just want to make enough money to be able to live comfortably. Anything more will be used to expand the project and make efforts in delivering the mentioned goals. However, even if all the resources are open-source and free there is still some type of cost in the form of technical debt. It will take some time to learn how to use those pieces of software along with the languages that build them.

In addition to property ownership, computer hardware, the use of sustainable utilities, social media platforms, and keeping up with updates done to the software by its maintainers.

Even if all the resources used in a project are open-source and free, there is still a cost in the form of technical debt. It takes time to learn how to effectively use these tools and the programming languages they rely on, especially when dealing with varying levels of documentation and community support. In addition, managing property ownership—such as servers, domains, or licenses—can accumulate technical debt if not handled with long-term planning, as neglect may lead to instability, legal complications, or costly migrations. Similarly, reliance on computer hardware introduces debt when older or inconsistent machines cause performance issues or incompatibilities, eventually requiring time-consuming upgrades and refactoring. The use of sustainable utilities also contributes to debt when systems are not built with energy efficiency or environmental standards in mind, as retrofitting solutions later becomes more complex and costly.

Furthermore, integrating social media platforms into applications or workflows introduces ongoing technical debt due to the constant evolution of APIs and platform policies, which demand continuous attention and adaptation. Failing to track and implement updates can result in broken features, reduced engagement, or security risks. Likewise, staying current with updates from software maintainers is essential but often overlooked; ignoring updates can lead to compatibility issues, exposed vulnerabilities, and larger codebase changes down the line. While these factors may not carry immediate financial costs, they represent long-term burdens that accumulate silently, ultimately requiring significant effort to resolve if not addressed early in the development lifecycle.

In addition to resources management, I must account for the human variable and in its governance within the project. For the project's governance I would like for it be unionized. In a perfect world we would not need unions, but we don't, so unions are needed for negotiating ideal working conditions. I would like to foster a cooperative environment within the community, because competition only considers the individual growth of a person or organization. As where cooperation considers the needs and well-being of multiple people or organizations collectively. So that means that resources and time can be spent equitably among the community to address any specified need. At the moment I am intended to be the only member of this project and so therefore the money generated by it will go towards me. Should the community grow, and I should have employees or contributors then I would like to compensate them equitably.

If I should have employees with some type of rank system, then I would like to equitably distribute the money made equitably. My current plan is to just increase the pay rate by 10% per rank among the employees or contributors. Any excess would be used to support other goals of the project or donate it to other open-source projects. While having extra money is good, it can also create unease among employees, contributors, or even just members of the community if the funds were used in secrecy. So, that is why I would also like to govern the project with transparency to foster trust among all members of the community. I would make it so that the stakeholders of the project are the employees themselves. This would make employees more engaged and attentive to the decisions making within the project. The idea is for the employees to feel more connected to the project than just mere workers or a means for a paycheck. For it is their time they are contributing, and it should be balanced with the life we all want to live. So that is why I also want to incorporate the 4-day work week, to allow either myself or potential employees to not feel burdened to work.

While I may be the only member right now, I have no real guidebook or code of conduct as I try to do my best to uphold the values I want to convey through this project. However, should more people be involved in the project then it would become necessary to enforce certain regulations and practices in order to sustain the organization. In general, I want for people to just respect one another, which unfortunately some people do not care to do so, so that is why I feel compelled to enforce it. It can correlate that if a person disrespects another human being, then they probably don't care too much for the environment either. So, I also want to enforce members of the community to respect nature, for it is where we live after all. It can also be correlated that a person's lack of respect for another will also not respect their privacy. So, I would also like to enforce the respect of the privacy of not just the community, but for anyone really. These correlations are not exclusive or mutual to one form of disrespect to another, but it someone who simply doesn't respect another person may not really care for their well-being either.

The idea of these specific code of conduct is to prevent any problems that may hinder the project or harm the community. But these types of things are learned or taught that is what this project aims to do. In addition to these I would also like for the code of conduct to include the importance of citing sources. Given by how much information is being transmitted to people these days, it is easy to spread misinformation if not properly informed. Misinformation can be very harmful depending on how it is used. So, my goal is to minimize misinformation and disinformation from hurting the community. But part of knowing how to stop the spread of misinformation one must also know how to improve the land in which they inhabit. Which is I would like to also add the practicing of sustainable habits within the organization. By being

conscious of the everyday decision, one makes and the impact the have on the environment one can make decisions that can positively impact the environment as opposed to harming it.

There is a lot that goes into any project, and especially to one as ambitious as this one. So, the need for documentation is crucial to keep things organized and persistent. I want to promote a “document as you go” type of documentation system where a developer documents their progress as they make changes. This way there is no need to try and remember all the changes done in one long sitting. It can be a written document, a voice recording, or even a video recording. All these formats have their advantages and disadvantages but can be used interchangeably. With modern software and hardware, a lot of it can be automated.

With automation a developer can do things faster and more efficiently. However, not all automation products are perfect, but the aim is for the automated process to be noninvasive, sustainable and ethically sourced. In the age of big data there have been some serious consequences for people who became victims of automating processes that where invasive. Personal information can be taken and used against a user. Which is something I want to prevent within the community. Most “AI”s today uses very large amounts of electricity and very harmful to the environment and us as a planet. My goal is to minimize the carbon footprint of the automation processes within the organization when using automated products. Additionally, there has been controversy about the data gathered to train the AIs. There have been reports of art being stolen or pictures or even written work that were taken to be used to train the AI models we see today. I don’t to support larceny or products that use stolen property within the organization, because it was mostly small artist who suffered the most. Big name companies did

not have that big of an impact or at all because of the amount of money they can put to protect their intellectual properties.

In conclusion I just want to make silly games with open source software and do my best to enjoy life.