Godot and Pygame Comparison

Synthesis document

Delgado Darren

February 9, 2025

1 Introduction

For the last semester of the computer science bachelor of the University of Strasbourg, we are tasked with the ambitious project of creating a game of an existing table top or card game.

Many tools are available at our disposal to code our chosen card game into a digital game, these include programming languages such as Pygame, Godot, or Unity with its C#.

This small synthesis will go through the advantages and disadvantages of Godot and Pygame to close in on my opinion on which language to chose for our project.

2 Godot

2.1 Advantages

A big advantage for Godot is the fact that it is easily used for multi-platform projects. Given that the second part of our project will be to have either a mobile or a web version, we will need to allocate time to that conversion accordingly. Using Godot this might simplify our life, if we choose to go for a mobile version.

2.2 Disadvantages

In comparison to Pygame, which is based on Python, Godot will be a new programming language to get into for every member. We will not have certain aspects that we have already used before, other than the general coding principles

3 Pygame

3.1 Advantages

One of the immediate advantages that come to mind is the fact that each member has already used python over the course of our bachelor's degree. Some more than others, but it would still act as something we already partially know.

Moreover, the game we have chosen is a simple card game, which we plan on having a simple 2D visual. This means that we do not necessarily need the best performing language, because we will not stress its capabilities to the maximum.

3.2 Disadvantages

Pygame is not the best language to use performance wise, which means that learning Pygame might not be as productive for our future endeavors. If we were to take the time to already learn Godot now, those of us that might see a future in the Gaming industry will already have a clear head start.

Another possible disadvantage could be that Pygame is primarily focused on 2D graphics, offering a straightforward API for rendering images, shapes, and text. It's sufficient for simple games but lacks advanced graphical features, which will not be a problem for our project. However, as mentioned before, it could serve as an unproductive choice for those of us that want to venture into the domain of Gaming.

Lastly, while extensible through Python's vast ecosystem of libraries, Pygame itself is somewhat limited with its built-in features. Developers often need

to integrate additional libraries for extended functionality.

4 Conclusion

In conclusion, I'd say the choice between these tow engines should be Godot. It gives us a new tool in the vast library of languages we have already learned and seems like a better fit for our future endeavors.