Game 201T: Design Challenge Analysis Questions

1. Based on the technologies / software you listed in Question 3 of the Design Challenge's Research Exercise, which of these technologies / software do you think will be most useful for the game you are prototyping as part of your final project?

I feel that scripting will be most useful to my game, Last Post. Out of the three I have mentioned in my Research Exercise (Scripting, Maya and Unreal Editor), scripting will provide the most transferable skills to my game. I am creating in Godot so I will be using GDScript. Both Maya and Unreal are 3D creation software while my game is a 2D side scroller so scripting will provide the most useful skill transfer.

2. (50 words) What is the cost of using this technology / software? How much money does it cost; how time will you need to spend learning it? Are there any other resources you it requires?

The cost is minimal for using GDScript. It is already included in Godot and does not require licensing. The cost associated with the use of this technology is time. I do have a programming background so the only learning associated with programming will be learning the GDScript syntax and Godot's node/signal structure.

3. (50 words) Are there any alternatives or competitors to this technology / software? Why did you choose it over the alternatives?

Yes, in Godot there is also an option to use C# for scripting. In other games engines Python, Lua or visual scripting (such as Unreal's Blueprints) are used. I chose GDScript because it is included in the game engine and does not require any additional set up. Also because I have a background in coding I do not think it will be very difficult to learn a new syntax.

4. List three ways that this technology will benefit your project.

1. I can easily store and retrieve states for objects such as the shake meter for bridges and the grow/shrink state of the mail bag

2. Scripting is included directly in the Godot editor which allows real time testing with no additional external compile steps required

3. I already have a background in coding so learning the GDScript syntax should be faster for me than someone who has no coding background

5. List three ways that using this technology / software might hamper or complicate your project.

1. There are some differences between the GDScript language and languages I already know but there is still a learning curve

2. There is a lack of tutorials available for the GDScript language compared to some other scripting languages that are used in other game engines (C# in Unity for example) due to community size

3. More complicated systems in the game such as the storm system increasing across several obstacles may require some knowledge of the Godot structure that I have not acquired but the actual implementation should not be too difficult

6. (50 words) How do you plan on addressing or compensating for the limitations you listed in Step 5?

Because I already have a background in coding I will be able to use my knowledge of other languages in order to learn the GDScript language rather than treating it as a completely new learning experience. I will begin by creating small scripts to become more familiar with the node structure in Godot and then move to more complex systems once I become more comfortable with the node structure.