Discussion of arcade/vanilla build

Input
From the initial stages of planning the prototype, we wanted to keep the project structured in a way that would easily be portable to the arcade cabinet. The main emphasis was on the control scheme and being able to map the controls to the correct layout. We found that the arcade cabinet controls were mapped to keyboard inputs so there was no need to implement full controller support, though we did consider it. Both in the alpha and in the final implementation the input system was designed around the concept of remappable keybinds.

[Not yet implemented] In the final implementation these maps are saved to and loaded from a text file that is accessible to be changed by the user. For the arcade build this text file is created with the map for the correct arcade controls and the option to rebind keys in-game is disabled.

Aspect Ratio
The difference in screen resolution and aspect ratio between the vanilla and the arcade versions of the game presented an issue as the base implementation of the engine did not support window scaling. We implemented the majority of the related systems and features without taking this into account.

The original thought was that we could make the game at a fixed resolution and that would solve any issues surrounding this, however this was not the case. With the current implementation the system level window generated has an inconsistent size based on the system's current resolution. For example on a 1080p monitor with a 16:9 aspect ratio the game displays correctly, however on a 4k monitor with the same ratio, all of the displayed content is offset to the left and scaled down. This issue becomes even more evident when you run the game in full screen resolution.

As we know that the aspect ratio of the target arcade cabinet is 4:3 we could have a version of the game that simply accounts for the screen size in pixels and adds an offset to all renderable objects on creation. The issue with this though is that we do not have a confirmed resolution. The monitor most likely has a resolution of 960x720 but again with no way to confirm this it just goes to show how that solution is not appropriate as it is in no way scaleable.

The best solution to this is to request the system details about aspect ratio and resolution and use these in comparison to the default that we have been using, 1920 x 1080. In doing this we can create a scalar value and then use this figure in the renderer to both adjust the position and size of the objects on screen.

Direct3D
As the engine is dependent on direct3D we will need to ensure that it is installed on the machine. This should be preinstalled in windows.

It is also worth noting that the engine is Windows only and would require a wrapper for the game.exe to be playable on any other systems. We felt it was necessary to highlight this as many hobbyist arcade machines are built as emulation cabinets either using a stripped down version of windows or similarly small builds of linux.

We could address this directly within the engine by having a layer of abstraction between the engines renderer api and the targeted library. This is done by other engines like Unreal and Unity and allows you to select which renderer you want to use.