**Architecture Description:**

The Double Dash Dodge & Deceive project will use the Model View Controller (MVC) software architecture.

The Model is where our data is stored. In unity, this looks like our vectors or position variables for our game objects. Also, the game objects themselves are in the model.

The View is the UI or HUD. Whatever our user is interacting with. It accepts inputs, like the unity character input. It also outputs displays on the screen to the user.

The Controller is the middle ground. In our case, it's our scripts for player movement and progress bar (so far just this). This does all the dirty work to turn the data in the model into something to be displayed to the user through the view

**Justification:**

This architecture was chosen because it is particularly good at simulating virtual worlds, due to the separation between model and view components. This makes it the architecture of choice for interactive video game projects such as Double Dash Dodge & Deceive. It is also the architecture that most closely matches Unity’s built in architecture, with rendering happening separately from the scripting layer that the programmer is exposed to.