

## Ethan Bodnar – SE2205 – Assignment 1

Alternative A: Have simple pickups (cubes, cylinders, etc.)

Alternative A presents the user with basic shapes as pickups, such as spheres, cubes, cylinders, or capsules, all of which are the default 3D GameObjects within Unity. This option offers the easiest possible implementation, as all that is required is that a prefab be designed as desired and created into the game through the main class. This alternative, however, offers the user a simple and basic pickup, which could detract from their experience with the game.

Implementing simple pickups may cause the game to feel too simple, and the user may feel as if the developer did not put enough time or effort into the design. On the other hand, it does make the job of the developer slightly easier, as they simply need one GameObject shape per different type of pickup, which is very simple to implement. Overall, this would allow the developer to use simpler GameObjects, which would cut down the development time, but would not change any of the classes, materials, or scripts whatsoever.

Alternative B: Have complex pickups made from multiple simple shapes (stars, etc.)

Alternative B presents the user with pickups that are far more complex than the basic 3D GameObjects provided in Unity. This option presents a slightly harder implementation, as it would require the developer to create compound shapes by rotating/adjusting GameObjects, setting them as children of another GameObject, and storing that compound shape as a new prefab (while also ensuring the collision system holds up by making sure each part of the compound shape has a Trigger Box Collider). This alternative offers the user much better and more impressive pickups than just simple 3D shapes, which will enrich their experience with the game and leave them disposed to thinking that the developer put more time and effort into the game design. It does, however, cause the game creation to be slightly more difficult, as the design difficulty increases as the pickup items get more and more complex, so the developer who designs a game according to these standards will take more time and have a tiny bit more trouble. Compared to alternative A, this method requires the exact same amount of code in the Main class (for the same number of pickup items in game), so the extra difficulties are relatively little. Overall, this alternative would allow the player to have a more enriched experience with more complex pickups, but would slightly increase the development time, although the developer would not have to add any additional classes, materials, or scripts as compared to alternative A.