Interfaces, inheritance, and scriptable objects are all utilized in this deliverable.

Why Interfaces

Interfaces for easy implementation of “required methods” or methods that the script that utilizes them is forced to implement in one way or another. For example, the shield and health items take the iRechargableItem interface to force them to implement the method: RechargeItem(float rechargeTime). This allows the coder to not forget any particular method and to make it simple to make something rechargeable simply by inheriting the interface for it.

Why Inheritance

Inheritance is incredibly variable and even interfaces utilize inheritance to allow for their uses. Inheritance allows any particular script to take a lot of generic variables or methods without having to explicitly put those variables and or methods into the script itself. For example, the HitScanGun inherits from WeaponBase which gives it a ton of basic variables and methods that any weapon will likely use. This allows the coder to implement many methods or variables that are repeated between similar items or weapons.

Why Scriptable Objects

Scriptable Objects are utilized to ease implementation of multiple different variables for any particular type of object and to swiftly allow a script to access these variables with a single reference. Scriptable Objects are used for all 6 subtypes of the 2 categories of collectables. For the packs it names the object, sets how much it's worth, what type of pack it is, and how long it takes to recharge. For the weapons, they are given a name, type, max ammo, and current ammo. These are all utilized in their respective game objects to allow for easy access of variables in their respective scripts.