Game Mechanics

* You can use escape to pause the game while on gameplay screen.
* A golf cart that serves as a delivery point for NPC that carries balls to calculate score.
* NPC that is sweeping the area according to its stamina and choosing the ball that gives the highest point to deliver.
* A menu for quitting the game in playtime and a main menu to start the game or quitting it.

Decision Making Algorithm

NPC decides which ball to pick by looking at how much stamina it has, how far is the ball and how many points the ball will give if delivered. Each ball is masked with a layer that only collides with itself and nothing else. This allows us to sweep a large area without costing too much performance. The sweep area is determined by using half of the current stamina because the NPC will go about the same distance when delivering the ball. If stamina gets so low that the NPC can’t find anymore balls to pick the game automatically ends and a message is shown.

Assumptions Made

Assumed that it is okay to:

* Use Cinemachine for easy camera setup,
* Use an empty object’s transform to setup an unloading point,
* Visualize the ball getting carried.
* Use a layer mask to optimize sweeping algorithm and deny physics for the balls. Because if the layer was made to collide with the ground to give the balls physics, this would affect the game’s performance by having to check if the ball is in a different position than it was when it was first chosen and having to check for tags of every object that the sphere collider has found. Checking tags could be avoided by using a second empty object and giving it a layer mask with a collider that solely serves as a checking method, but this would still cost us performance due to checking every frame for movements.