

SCRIPT CHANGES:

spawner.cs:

Line 18: Added a static bool variable to flag when player is spawning an object and is not panning

Lines 37 & 51: Set panning to false when spawning and true once object has spawned

Lines 43 & 49: add a y axis offset to the spawning object to keep object in sight (above finger) when placing object

G_ProteinCmdCtrl.cs :

Changed G-protein 'socket' colors to white in Start() and renamed Explode() to DestroyObj().

NEW SCRIPTS:

PanAndZoom.cs

A pan and zoom script has been added and is a component of Main Camera in the Models scene (game play scene). The script currently works for touch screen devices ONLY and does not function in the editor. Pan and zoom can only be tested in a game executable.

Maximum zoom is currently (and reasonably) set to x3. Pan is also restricted to set boundaries.

There is an issue when a player attempts to pan WELL outside of the boundary. It appears to be limited to the lower left quadrant. After so many attempts to pan outside the boundary the code will recognize one touch as two and begin to zoom in and out with unpredictable behavior. Further investigation into this anomaly is required. It is important to note this anomaly occurs when a player attempts to pan EXCESSIVELY out of bounds but appears functional within a reasonable panning range.

HomeScreen.cs

Per Monday's discussion, everybody's name is added to the list of developers. In adding those names, I took some creative liberties and redesigned the home screen to include animated cell proliferation and an information button that will display a brief overview of the game development and a list of developers.

Lastly, the drop down menu has been rearranged so the objects are not in order of process interaction. Dr. Cline requested future functionality randomize the position of these objects. The text labels for these objects are un-parented from the objects themselves to prevent the text from dragging with the object at the time of placement. You may wish to re-parent the text and update spawner.cs to only drag the parent object. This may simplify the randomizing code such that you would not have to relocate the object AND its text.

That is all. I am typing up code development document outlining the progression of the project over this term. I will upload it to the repository for your review as soon as it is complete.

Future Script Change Recommendation:

Remove the FindClosest method as it is unnecessary (GTP_CmdCtrl, G_ProteinCmdCtrl)

The algorithm was coded such that:

- First find the object's closest target object
- *Next check if any other targeting objects are closer*

It's redundant in that if you want to find a closest targeting object, have the target make a single call to the FindClosest method. There's no need for a targeting object to look for its closest target, only to have that target look for a closer targeting object.

Even calling the FindClosest from the target object does not guarantee it is the closest object at the next update (another targeting object, while roaming, may be pushed closer to the target than the object targeting it).

With the removal of FindClosest, the targeting code can look for any open target and lock on to it, while still ensures only one targeting object tracks one open target (no two targeting objects will target the same target object).