Drone Controller Documentation

Feel free to contact me at any time for questions at: mario.haberle@gmail.com

- Drone set-up
- Explaining drone movement script
- Explaining Elisa script
- Camera script

Drone set-up

Create a Empty GameObject in Unity scene and we can name it “Drone”. Change its tag to “Player”(important). Now place your drone model into your new Empy GameObject “Drone” and reset its position to (0,0,0). If you do not have a drone model, you can make one out of cubes to represent your drone. If you have a drone model remember to put the rotors or wings how do you call it outside of the drone model into the Empty GameObject “Drone”. So now you should have an “Drone” GameObject with rotors/wings inside and a drone model.

Add a RigidBody Component and BoxCollider or if you have a MeshCollider to the “Drone” Empty GameObject and set it's mass to 10. You can change the mass of it but this script is configured to run on mass of 10.

Next add a DroneMovementScript onto the “Drone” Empy GameObject and ElisaScript.

That its for the Drone Configuration, the movement keys are WSAD to move forwards/backwards and left/right, keys to rotate and rise/lower the drone are IKJL.

Explaining the Drone Movement Script

We used fixed update loop since we are dealing with phsyics only. Inside fixed update loop you'll find:

ClampingSpeedValues(); - this method contrains the velocity of our drone to go above certain speed. For instance if the W key is pressed it will allow our drone to move forward with top speed of 10 unity metric system per second. If the key A or D is pressed only, it will allow side movement of 5 unity metric system per second. If we arent pressing anything, to ensoure our drone will not move, we will smooth damp its velocity to Vector3.Zero which is value of zero on every axis.

MovementUpDown(); - it listens for controls of moving up and down.

For instance, if user presses key I, the force that makes the drone move up will increase to 450 and if user presses the I key and wants to move sideways the drones force to move up will change to 500, since it's not going that fast forward it can go upwards much faster. I wont explain every moethod inside, they are actually just changing the force to move up to coresponding movement so the drone doesnt wall down if the force to move up is to low. You can also in the inspect of the drone see which is the current “UpForce” for the keys pressed so you can play with those values.

MovementLeftRight(); - just waits for the action of buttons A and D and it tillts the Drone rotation to one side if any of the corresponding keys are pressed. Also it applies force to the side for our drone to move.

Rotation(); - waits for inputs of J and L keys, it will rotate the drone for the amount of “rotationAmountKeys” variable value. And there we are calculating t he currentY rotation that is later applied, its just to smooth the things out to look nice a neat.

MovementForward(); - just waits for one of the W or S keys to be pressed and will move in a corresponding way applying movementForwardSpeed variable value to our drone force RigidBody. Laso it will tilt a drone in a correspodn way.

DroneSound(); - changes the drone pitch of a sound if put on the game object “Drone” to make it appeal like moving sound and noise.

And at the and of the FixedUpdateLoop() we apply our drone force up constantly to keep it on the right track also here we set the rotation to make our drone change rotation happens.

Explaining the ElisaScript

If you have any rotors/wings on your drone and you want them to rotate, in the ElisaScript of the gameObject “Drone” in the inspector increase the Elisa array size to the amount of rotors/wings you have. Then just simply drag and drop from the scene Hirachy gameobjects that you want to rotate.

Explaining the camera script

Its just a plain camera with a script that follows the drone, nothing special here. Its only important to set a tag on your drone to “Player”.