Solution description

Most of the solution is done C++, blueprints are used for visuals and composition of command center. Command center functionality is split between 2 components, drone spawner and drone control. These components function without each other and you can use them to extend functionality of another actor (demonstrated on mothership drone).

Drone

Excluding basic drone functionality, this class also handles all logic for registration of drone to command center. Main reason for doing it here was easy handing in situations where multiple command centers overlap. Drones are spawned only by drone spawner component. This is purely because I don't know how to spawn actors in UE4 with parameters and I need to have the drone initialized before it starts to handle overlap events.

- Drone speed is set in corresponding blueprint class
- Movement commands are done by vectors
- Drone generates commands for itself when lost
- Drone starts doing new command immediately after reception, regardless of its previous state
- If drone is registered, and it has finished its command, it stops and waits for new command

Registration flow

If the owner of spawner component has drone control component, drone is registered to it and stays registered until it leaves its range (requested by designer). Drone holds array of drone controls in range. When drone enters range of new drone control, it registers to it and adds it to the array. When drone leaves, it deregisters, and checks the array for drone controls in range. If there are any it registers to the first one. If the array is empty, flag is set to indicate the drone is lost.

Heavy drone

Inherits most of the functionality of drone, overrides OnHit() and GenerateCommand() to work properly when hit. Has separate blueprint, which allows customization of visuals.

Drone spawner

Spawn various types of drones. Drone type, quantity and spawn frequency are set in blueprint. Only BP variants of drone classes should be spawned.

Drone control

Owner actor needs to have a collision component so drones can register at the beginning of overlap. Frequency of broadcast is set in blueprint. When new command is broadcasted, event is fired on all registered drones and they receive new command.

Possible improvements

- Paramatrized construction of drone, would allow to spawn without spawner component
- Acceleration and Decelerations for drones
- Finish mothership implementation, as is it tends to crash