# Interfacce

### IEnemy

Interfaccia che definisce i metodi in comune tra tutti i nemici:

* void Hurt(float damage); //Take damage and decrease enemy healt
* void Revive();
* void BeTriggered(); //Transit to the hostile state and start to chase the player
* void MoveTo(Vector3 point); //Move the enemy from the current position to point position in input
* void Patrol(Vector3[] path);

### ReactiveEnemy

Interfaccia che implementa l’interfaccia ReactiveObject. Definisce inoltre i metodi:

* void ReactToExplosion(float damage);

# Classi

## Manager

### SpawnPoint

Un oggetto SpawnPoint è un empty GameObject con un apposito script che permette di settare l’id dell’area a cui afferisce tale punto di spawn e l’id della tipologia di nemico da far spawnare.

### EnemiesAreaTrigger

Trigger invisibile che quando attraversato invoca l’apposito metodo dell’EnemiesManager che popola l’area. Potrebbe eventualmente dire al manager di recuperare i nemici ancora presenti nell’area precedente.

### EnemiesManager

Tale classe si occupa della gestione dei nemici nella scena.

Al lancio istanzia un pool di nemici e li rende disponibili per essere piazzati nelle varie aree della mappa. Il numero di nemici massimo da istanziare per ogni tipologia è limitato e predefinito. Ciascun nemico appena istanziato viene disattivato e viene salvato un riferimento su un apposito dizionario **enemiesPool**, il quale ha come chiavi un intero che identifica la tipologia di nemico.

Tipologie:

* 0 -> sceglie il manager a random
* 1 -> Drone
* 2 -> MiningSpider
* 3 -> SecurityGuy
* 4 -> Tank

Contestualmente effettua una ricerca di tutti gli SpawnPoint presenti nella scena e li mappa in un dizionario spawnPoints(int areaID, List<SpawnPoint>).

Un altro dizionario enemiesArea(int areaID, List<Enemy>), contiene per ogni area l’insieme dei nemici da far spawnare in quell’area (o già spawnati in quell’area).

Invocando un apposito metodo PopolateArea(int areaID), il manager piazzerà negli appositi SpawnPoint con ID dell’area uguale ad areaID la tipologia di nemico associata a ciascun SpawnPoint, prelevandoli dal pool. Accedendo al dizionario enemiesArea, con questo metodo è possibile ripopolare un area con gli stessi nemici oppure modificando il dizionario è anche possibile popolarla con nemici diversi.

Quando un nemico muore, quest’ultimo invoca il metodo KillEnemy, esposto dal manager stesso, il quale si occupa di disattivare il nemico (eventualmente sostituendolo con un corpo inerme se necessario), e riallocandolo nel pool.

Quando il player abbandona l’area e attraversa un apposito collider EnemiesAreaTrigger, il manager collezionerà tutti i nemici ancora vivi presenti nell’area precedente per renderli nuovamente disponibili, attraverso il metodo CollectEnemy. A questo punto popolerà la nuova area raggiunta dal player con l’apposito metodo PopolateArea.

Dopo un tot di tempo, i cadaveri presenti nella scena vengono collezionati per essere riutilizzati, tramite il metodo CollectCorps().

## Nemici

### Enemy

### Drone

### MiningSpider