```
┌─────────────────────────────────┐                    ┌──────────────────────────┐
│                                 │                    │                          │
│             Class               │                    │        Interface         │
│          OBJECT POOL            │                    │         IPoolable        │
│            Spawn()              │────────────────────│      Action OnDisable    │
│     Queue of IPoolable objects  │                    │                          │
│    hashSet of  IPoolable objects│                    │                          │
│                                 │                    └────────────┬─────────────┘
│                                 │                                 │
│                                 │                                 │
└─────────────────────────────────┘                    ┌───────────┴──────────────┐
                                                        │                          │
                                                        │                          │
                                                        │    Object:IPoolable      │
                                                        │    ObjectBehaviour()     │
                                                        │                          │
                                                        └──────────────────────────┘
```

INTERFACE (GoonBuilder)
SetSpeed()
SetPointVal()
SetSprite()

BUILDER(GoonSpawner
Makes "goon" instance with methods
from interface, and values from
Config classes

SUBJECT 1
CONFIG
(HatGoon)
SetPointVal(30)
SetSpeed(2f)
SetSprite(0)

SUBJECT 2
CONFIG
(ParaGoon)
SetPointVal(50)
SetSpeed(3f)
SetSprite(2)

SUBJECT 3
CONFIG
(HeliGoon)
SetPointVal(75)
SetSpeed(4f)
SetSprite(1)

HatGoon

ParaGoon

HeliGoon

Observer Pattern:

class ScoreTracking

// value of our current score
// Text Mesh Pro field for the score

// function for Incrementing Score(parameter for points)
{
        // Update the score;
        // Write the updated score;
}

class TargetBehavior

// has a function for taking damage which updates the score, notifying the observers from ScoreTracking