Documentație - partea de testare

În Godot, mergem la AssetLib, vom descărca, iar apoi vom instala GUT - Godot Unit Testing( ne asigurăm ca avem toate căsuțele bifate) .

Mergem la ProjectsSettings, Plugins și vom activa **GUT**.

Cream o nouă scenă pe care o denumim TestRunner, iar apoi adugam nodul Gut pe care il redenumim corespunzător.

Cream un folder **test**, iar altul în interiorul lui numit **unit**, unde vom crea un scripturile pentru testare**.**

În *Inspector* ne asigurăm ca este bifata Run On Load. Pentru *Directory1*, vom atașa calea către folderul **unit,** creat anterior.

În scriptul **test\_player.gd**, preluam în variabila *player* scripul *Player.gd* folosit, apoi ne creăm o funcție care va testa dacă *PlayerData.deaths* se va mari cu o unitate atunci cand Player-ul moare. Apelăm metoda *die()* din *Player*, apoi cu ajutorul funcției *assert\_eq([],[],[])* din Gut testăm dacă valoare obtinuta este egala cu cea așteptată.

În scriptul **test\_coin.gd**, preluam în variabila *coin* scripul *Coin.gd* folosit, apoi ne creăm o funcție care va testa dacă *PlayerData.score* se va mari cu o unitate atunci cand Player-ul colectioneaza un coin. Apelăm metoda *picked()* din *Coin*, apoi cu ajutorul funcției *assert\_eq([],[],[]*) din Gut testăm.

În scriptul **test\_enemy.gd**, preluam în variabila *enemy* scripul *Enemy.gd* folosit, apoi ne creăm o funcție care va testa dacă *PlayerData.score* se va mari cu o unitate atunci cand Player-ul ucide un Inamic. Apelăm metoda *die()* din *Enemy*, apoi cu ajutorul funcției *assert\_eq([],[],[])* din Gut testăm dacă valoare obtinuta este egala cu cea așteptată.

În scriptul **test\_nodes\_free.gd**, preluam în variabila *autoFree* scripul *autofree.gd* din Gut, apoi ne creăm o funcție care va testa dacă se eliberează după ștergerea unui nod. Cream un nod nou, il adaugam(copil al celui de mai sus), apoi apelăm metoda *free\_all()*. Cu ajutorul funcției *assert\_freed([],[])* din Gut testăm dacă nodul adaugat a fost șters.

În scriptul **test\_emitted\_signal.gd**, preluam în variabila *signalWatcher* scripul *signal\_watcher.gd* din Gut, apoi ne creăm o funcție care va testa dacă semnalul a fost emis. Cream o clasa nouă *SignalObjec*t, pentru semnale fără parametrii, care va emite semnalul, apoi apelăm metoda *get\_emit\_count* pentru signalWatcher si verificam daca a fost emis.

Dăm play și vom vedea ca a trecut toate testele(print-screen mai jos).

