

# SHOOT AND HIDE

Sergi van Ravenswaay

3017869

**Introductie**

Het spel 'Shoot & Hide' is een simpele FPS waar 2 spelers omstebeurt een wapen krijgen. De andere moet zich verstoppen en proberen niet gepakt te worden. Zodra jij diegene doodschiet krijg jij een punt. Het spel bestaat uit 2 rondes van 1 minuut en daarna zal de score naar de server worden gepusht.

## **UML**

De UML die ik heb gemaakt is compacter en ik heb niet alle Player.. scripts in de UML gezet. Dit om overzicht te houden in de UML. Aangezien de player met andere scripts interacte en de rest aangestuurd werd door de player zijn script. De UML kan je vinden op de Github van mij!

## **Design Patterns**

Singleton – Voor belangrijke managers heb ik gebruik gemaakt van een singleton. Dit om snel data aan te passen en te wijzigen. Zeker bij networking is het belangrijk om deze op elke client te synchroniseren.

Fine State Machine – Deze heeft verschillende states die worden gesynced over het netwerk. Zo runt elke client in dezelfde state en kan de gameplay aan de hand daarvan worden aangepast.

## Scripts

Hieronder staan de scripts uit dit project en wat hun functie daarvan is:

EventManager	Hier worden events in aangeroepen d.m.v. statische functies die je kan aanroepen.
GameManager	Deze doet de belangrijke netwerkfuncties. Het opslaan van online spelers en het (de)registreren ervan. Hier kon je de spelers aanroepen en daar functies in het spelersscript mee aanroepen.
RoleManager	De RoleManager verdeelt aan het begin in wie de hunter is en wie moet wegrennen voor de andere speler.
WeaponManager	De WeaponManager staat op de speler en beheert de wapens. Om het wapen te enablen en te disablen en deze ook te kunnen switchen.
AppManager	Dit is een belangrijke manager voor de connectie met de Database. Hier word alles wat in de <i>AuthenticationManager</i> word aangeroepen opgeslagen. Hier kan je ook data uithalen die in de database werd opgeslagen.
Authentication-Manager	Deze manager is er om in te loggen en te registreren bij de game. Dit kan via het inlogscherf en werkt door middel van een enumerator en de <i>AppManager</i> waar die alle data in opslaat.
WebManager	Voor Highscores en het updaten als de game is gestart was de WebManager. Die stond in de main scene en daar kon je de data aanpassen.
Player	De Player is het mainscript waar alles van de speler werd opgeslagen. Denk aan zijn rol en de health.
PlayerSetup	Hier werd alles gedaan om de player klaar te maken voor zijn game. Zo word de UI hier gespawnd en word hij geregistreerd bij de <i>GameManager</i>
PlayerController	Hier werd alle input vertaald naar de <i>PlayerMotor</i> om de speler te bewegen.
PlayerMotor	In dit script werden alle waardes van de <i>PlayerController</i> omgezet in beweging.
PlayerShoot	Hiermee kon je schieten en andere classes activeren waar dit over het netwerk werd gesynchroniseerd.
PlayerInterface	De Player zijn interface werd groot en deels aangestuurd door dit script
PlayerWeapon	Wapens werden met dit script aangemaakt. Zo kon je meerdere modulaire wapens aan maken die vervolgens makkelijk in het systeem konden worden gezet.
StateManager	Hier werd alles van de states geregeld en over het netwerk gesynchroniseerd
...State	States met hun eigen unieke functies
StateMachine	De StateMachine die ik hiervoor heb gebruikt

MatchSettings	Een script met de matchsettings die makkelijk kon worden aangepast en uitgebreid.
PauseMenu	Om het spel gemakkelijk op pauze te zetten.
Settings	Om de settings scherm te gebruiken in de lobby en je data aan te passen in de database.

## **Het proces**

Mijn proces begon best stroef. Ik had een kaartspel wat wou bouwen (een soort heartstone met allemaal aspecten). Dit werd uiteindelijk een te groot concept en ik besloot met nieuwe ideeën te komen. Een concept met een tower waar je omstebeurt blokken op moest gooien was het nieuwe concept. Na lang proberen heb ik toch besloten een van mijn eerdere games om te bouwen naar een multiplayer game.

Het concept is dat je omstebeurt op elkaar moet schieten. Toen ik aan dit concept begon waren er nog twee weken over. Ik heb toen gezorgd dat de bestaande game zonder bugs werkt. Je kan omstebeurt schieten en de score word na 2 rondes van 1 minuut doorgestuurd naar de server.

Dit ging niet zonder slag of stoot. Zo heb ik veel problemen gehad met het synchroniseren van de state machine over het netwerk. Zo was het moeilijk om alle states en het gedrag goed over het netwerk werkend te krijgen. Ook was het op het juiste moment doorsturen van de score een klus.

Uiteindelijk zijn beide problemen zonder problemen werkend gekomen op de client en server. Wel wil ik de volgende keer beter onderzoek doen naar hoe ik de synchronisatie het beste kan aanpakken!