

Prototype / MVP

Thomas van der Molen

S2-DB05

Introductie

In sprint 1 ben ik begonnen met het programmeren van mijn engine waar uiteindelijk mijn spel op gebouwd wordt. Hierbij heb ik geprobeerd de belangrijke onderdelen zoals: Het renderen van frames, het opvangen van gebruiker inputs en de “physics” te implementeren.

Deze basis functionaliteiten zijn bepaald in het EersteVersieRequirements bestand.

Monogame

Zoals in de one slide is uitgelegd bouw ik mijn game engine en de game zelf in c# met de [MonoGame framework](#). Hierbij maak ik veel gebruik van online resources en het boek “[monogame-mastery](#)”

Uitvoering

Bij het programmeren ben ik begonnen met het opzetten van een systeem waarmee ik gemakkelijk sprites naar mijn game kan renderen, ook heb ik meteen een gamestate systeem toegevoegd omdat het mij het beste leek om dit meteen in het structuur te verwerken.

Hierna heb ik het systeem gemaakt waarmee ik naar inputs van de speler luister en deze op een hele vrije en makkelijke manier laten verwerken. Bij dit onderdeel heb ik veel hulp gevonden bij het monogame-mastery boek om een efficiënte implementatie te vinden.

Tijdens het maken van het input systeem ben ik ook begonnen met het maken van mijn object sprites, dit bestaat voornamelijk uit een speler en grond object, deze objecten moeten natuurlijk met elkaar kunnen reageren. Daarom was mijn volgende stap om een collision detection systeem op te zetten, dit zal ervoor zorgen dat ik kan controleren als bijvoorbeeld een speler op de grond land en vervolgens blijft staan in plaats van door de grond te vallen. Dit systeem heeft mij veel problemen veroorzaakt en is nog niet perfect.

Hierna was het tijd om een volledig werkende physics engine te maken. Momenteel zou ik het niet een “engine” noemen aangezien het nog steeds redelijk klein is en werkt vanuit de speler sprite. Dit onderdeel komt neer op een xVelocity en yVelocity die bepalen waar de speler de volgende frame heen beweegt. Door alles te bepalen met een x en y variabele hoop ik het systeem onderhoudbaar te houden. Ook al is het systeem niet al te moeilijk, het “grappling hook” systeem heeft mij nog steeds veel moeite gekost. Tijdens het maken van dit onderdeel kwam ik er steeds achter hoe ingewikkeld mijn idee zou worden en dat het mogelijk niet eens al te leuk zou zijn om te gebruiken. Om deze reden heb ik het hele systeem herschreven waardoor het grapplinghook systeem net iets anders werkt en niet echt meer een grapplinghook is. Gelukkig heb ik door deze herschrijving alles nog een keer goed door kunnen kijken en werkt de nieuwe functionaliteit een stuk beter.

Als laatste functionaliteit heb ik ook al een connectie met een MySQL database opgezet, hierbij gebruik ik de MySQL database te vinden op het [fhict selfservice portaal](#). Momenteel haalt mijn spel de speler's naam en score op uit de database en schrijft deze naar tekst op het scherm. Zelf heb ik eigenlijk nog maar weinig ervaring met het opzetten van zoiets maar dit is gelukkig zonder problemen gelukt.

Afsluiting

Momenteel werken deze onderdelen volledig en is de functionaliteit die ik in mijn MVP af wilde hebben (en benodigd was na een feedback moment met docent) geïmplementeerd.



Figuur 1 Het spel in actie

Momenteel kan het spel dus gespeeld worden met A en D om links en rechts te gaan, verder kan de linkermuisknop ingehouden worden om een soort sprong op te laden hoe langer deze sprong wordt opgeladen en verder de muis van de speler is hoe verder de speler zal springen.