

Amaze

Inledning

I projektets planeringsstadie kom gruppen överens om att ett eget VR-spel skulle vara en spännande utmaning då ingen hade tidigare erfarenhet inom det.

Resultatet blev Amaze, ett VR-spel för två spelare. Personen med VR-glasögon blir placerad i en labyrint med en lykta i handen och för att ta sig ut behöver spelaren lösa ett antal pussel. Den andra personen har en PDF framför sig där instruktioner för varje pussel finns. Spelarna behöver därför samarbeta för att ta sig ut ur labyrinten innan lyktan slocknar.

Valet av VR som plattform var en fråga som diskuterades en del innan allt drog igång men eftersom några av medlemmarna i gruppen hade tillgång till *Samsung Gear VR* så föll det naturligt på den typen av plattform.

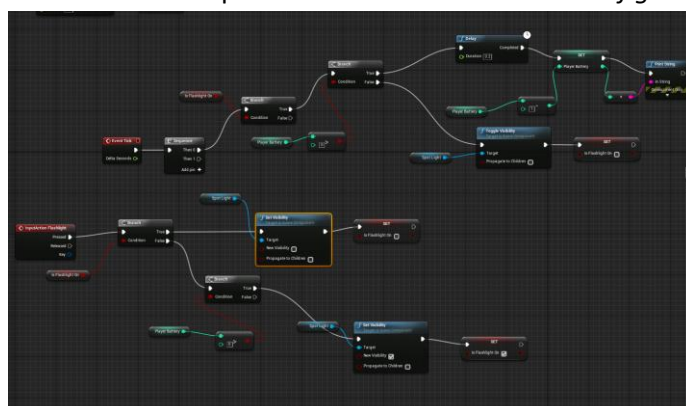
Arbetsprocess

Gruppen valde att använda sig utav spelmotorn *Unreal Engine 4* då det innehöll alla nödvändiga verktyg för att utveckla ett spel för VR.

För versionshantering användes *Github* för att på ett enkelt och smidigt sätt sammanfoga gruppmedlemmarnas arbete till samma plats. Majoriteten av arbetet utfördes på datorerna i sal K4507 och medlemmarna arbetade på olika datorer vilket medförde komplikationer när allas arbete skulle sammanfogas. Github löste detta åt oss genom att ha en gemensam datakatalog som alla kunde pusha sitt arbete till, dock tillkom även här vissa problem. Exempel på detta var att en del sparade filer försvann och vissa uppladdningar inte gick att ändra.

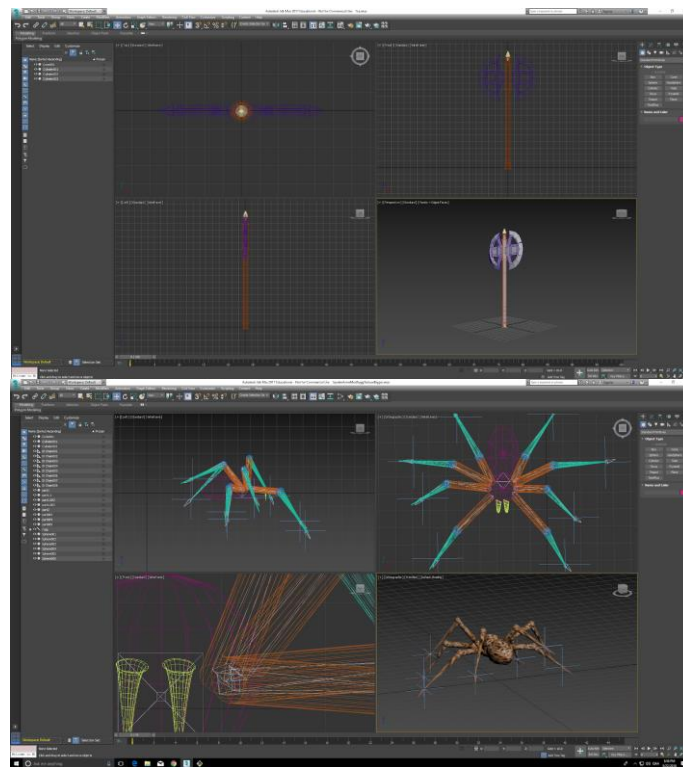
Arbetet fördelades så att vissa satt och modellerade objekt som skulle in i spelet, medan andra satt med den tekniska delen av spelet och spel-logiken.

Spel-logiken skapades främst med hjälp av Unreal Engines egna programmeringsspråk *Blueprint* som är en typ av blockprogrammering. I projektets tidiga stadier prioriterades karaktärens rörelser med kameran och sedan fortsatte arbetet med till exempel ljuslyktans egenskaper och hur den ska fungera. En typisk överblick hur en blueprint kunde se ut kan ses nedan i *figur 1*.



Figur 1. En överblick på en Blueprint i Unreal Engine 4

Parallellt med att dessa spelfunktioner utvecklades satt några andra gruppmedlemmar och 3d-modellerade olika objekt som sedan skulle användas i spelet, två av dessa kan ses i *Figur 2*. Objektet skapades i *3DS Max* samt *Blender* och exporterades senare som .fbx-filer för enkel implementation. Materialen som skulle användas var bilder som kommer från internet och då användes en mjukvara som heter *CrazyBump* för att skapa maps för normaler, bump, AO och spekulära. Detta var däremot lite för krävande för den mobila plattformen så material från Unreals egna mobilbibliotek användes istället. Även displacement maps användes först men detta är avaktiverat för mobiler på grund av hårdvarubegränsningar. Samtliga material på väggar och mark är "sömlösa" vilket får effekten att två objekt som ligger bredvid varandra smälter ihop utan skarv. Ljudeffekter såsom fotsteg, dörrknarr, och diverse bakgrundsljud har implementerats som triggas vid vissa händelser, när spelaren markerar objektet, eller plockar upp något.



Figur 2. 3D-modulerade objekt.

Objekten som modellerats är bland annat yxorna, spindeln samt dess animation/material, facklor, bokhyllor, böcker, hörnspelare, tändstickor och ljuslyktan. Spelnivån, dvs väggarna och marken är uppbyggda direkt i Unreal Engine 4 och ljuset från himlen ska simulera nattljus som sprids över labyrinten, där en dimma ligger och skapar mystik. Inuti labyrinten kommer en liten mängd ljus in ovanifrån men det är spelarens egna ljuslykta som lyser upp scenen närmast spelaren mest. Tanken är dock att spelaren ska befinna sig i en mörk miljö som sedan när ljuslyktan släcks blir kolsvart och spelaren då har förlorat omgången. Spindelväv har importerats från internet som png-filer med transparent bakgrund och sedan animerats. Små stenar har lagts ut på golvet och diverse buskage har lagts in för att göra upplevelsen verkligare. Spelaren klarar spelet när den tar sig upp för trappan

och därmed ut ur labyrinten. Nedan i *Figur 3* visas en PDF som används av spelare två för att assistera personen med VR.



Figur 3. Hjälpmedel för spelare 2.

Resultat

Vi har skapat en labyrint med en otäck miljö där en användare måste ta sig ut ur labyrinten genom att lösa pussel. Användaren har även till förfogande en "kartläsare" som hjälper till att lösa dessa pussel och navigera i labyrinten.

Gruppen är nöjd med resultatet, även om det finns klara förbättringsmöjligheter. Spelet fungerar bra och vi ser det som en grund för att kunna utveckla det i framtiden. Nedan (*Figur 4*) syns en skärmdump på bokhyllan som öppnas efter att ett pussel har lösts.



Figur 4. Bokhylla som öppnas i spelet för att komma vidare.

Diskussion

På grund av att vi valde att anpassa spelet för Samsung Gear VR så medföljde även gränser på hur mycket vi skulle kunna implementera. Spelvärlden fick inte vara för stor och objekten som visades

samtidigt fick inte ha för hög mängd polygoner. Enligt rekommendation från Samsungs egna dokumentation fick sammanlagda objekten som skulle visas på skärmen för varje bildruta samtidigt inte överskrida 100 000 polygoner. Om antalet polygoner gick över denna gräns kunde bildhastigheten (framerate) försämrats.

Texturer fick inte vara större än 2048x2048 pixlar och på grund av att vi använde mobil som plattform kunde inte displacement maps användas.

Vi insåg snabbt hur mycket man fick begränsa sig när man arbetar med realtidsrendering med denna typ av plattform.

Men det fanns även fördelar med Gear VR framför exempelvis HTC VIVE. Med Gear VR kunde vi kontinuerligt exportera en version av spelet och testa den på olika platser, något som inte skulle varit möjligt med HTC VIVE. Vi hade tillgång till egen HTC VIVE men det tar lång tid att sätta upp allting så det fungerar varje gång man vill testa spelet. Det fanns även möjlighet att få hyra VR-Labbet på Visualiseringscenter C men detta skulle också komplicerat testprocessen som var nödvändig under utvecklingsstadiet.

Från början var det meningen att spelaren skulle ha en ficklampa för att navigera runt med och batterier skulle finnas utspridda i labyrinten. Spelaren kunde inte tända ficklampan om batteriet tog slut och det fanns ett begränsat antal batterier i spelvärlden. Detta visade sig inte fungera på mobilen då den inte stödjer rörliga spot lights. Därför implementerades istället en point light som därmed lyser upp väggar åt alla håll. Designmässigt löste vi detta genom att modellera en ljuslykta istället och att man plockar upp tändsticksaskar som "fyller" på längden lyktan kan vara aktiv.