



CHALMERS



GÖTEBORGS UNIVERSITET



Ziega

Ett verktyg för rollspel i virtual reality

Kandidatarbete inom Tillämpad informationsteknologi

RASMUS DAVIDSSON
MICHAEL DRAGOMIR
NIKLAS KLING
JOHAN LJUNGBERG
HENRIK ROSTEDT
ULRIKA UDDEBORG

Ziega

Ett verktyg för rollspel i virtual reality

RASMUS DAVIDSSON
MICHAEL DRAGOMIR
NIKLAS KLING
JOHAN LJUNGBERG
HENRIK ROSTEDT
ULRIKA UDDEBORG

Institutionen för Tillämpad informationsteknologi
CHALMERS TEKNISKA HÖGSKOLA
GÖTEBORGS UNIVERSITET
Göteborg, Sverige 2017

Ziega

Ett verktyg för rollspel i virtual reality

RASMUS DAVIDSSON
MICHAEL DRAGOMIR
NIKLAS KLING
JOHAN LJUNGBERG
HENRIK ROSTEDT
ULRIKA UDDEBORG

© RASMUS DAVIDSSON, 2017.
© MICHAEL DRAGOMIR, 2017.
© NIKLAS KLING, 2017.
© JOHAN LJUNGBERG, 2017.
© HENRIK ROSTEDT, 2017.
© ULRIKA UDDEBORG, 2017.

Handledare: Staffan Björk
Examinator: Morten Fjeld

Institutionen för Tillämpad informationsteknologi
Chalmers Tekniska Högskola
412 96 Göteborg
Telefon: 031-772 1000

[Omslag: Bild från prototypen, översikt över spelplanen]

Institutionen för Tillämpad informationsteknologi
Göteborg, Sverige 2017

Abstract

Virtual reality has been around for a long time, but has recently become more popular. More accessible VR equipment has increased the usage of VR, especially within the video game industry, but there hasn't yet been extensive research regarding how game design is affected by VR.

This project explores the possibilities of using virtual reality to enhance tabletop roleplaying games. To do this, a prototype is developed and current ideas regarding interaction and design are studied. By implementing different functionality in the prototype they can be examined within the context of VR and whether the use of VR enhances the gaming experience can be explored. These functionalities are described and discussed in the report, but not all were implemented in the prototype due to constraints in the project.

During the project a few aspects where VR was most likely to improve or add new elements to roleplaying games were identified. But as there were no proper user testing conducted no definitive conclusions can be made. These aspects are only some of the possibilities of VR and there are certainly other areas that can be studied and explored in the future.

Keywords: Virtual Reality, Oculus Rift, Roleplaying Game, Unreal Engine, Chalmers, Bachelor Thesis.

Sammanfattning

Virtual reality har funnits under en längre tid, men det har nyligen slagit igenom stort. Mer lättillgängliga VR-system har ökat användningen av VR, inte minst inom dataspelsindustrin, men det har ännu inte gjorts någon större mängd forskning inom hur speldesign påverkas av VR.

I detta projekt utforskas möjligheter att förbättra bordsrollspel genom VR. För att göra detta studeras nuvarande idéer om interaktion och design inom ämnet och en prototyp utvecklas. Genom att implementera olika funktionalitet i prototypen kan dessa undersökas inom ramarna för VR och huruvida VR kan berika spelupplevelsen kan utforskas. Dessa funktionaliteter beskrivs och diskuteras i rapporten, men alla implementerades inte i prototypen på grund av begränsningar i projektet.

Under arbetet identifierades några aspekter där det är troligt att VR kan förbättra eller tillföra nya element till rollspel. Men då det inte utfördes några utförliga användartester kan inga definitiva slutsatser dras. Dessa aspekter är endast några av de möjligheter inom VR och det finns definitivt områden som kan utforskas ytterligare i framtiden.

Keywords: Virtual Reality, Oculus Rift, Roleplaying Game, Unreal Engine, Chalmers, Bachelor Thesis.

Förord

Vi skulle vilja tacka Staffan Björk för gott handledarskap, idéer och stort intresse för projektet. Vi skulle även vilja rikta ett tack till Daniel Sjölie som gav oss bra information och hjälp angående projektets tekniska aspekter.

Vi vill också tacka Thomas Hansson och Åsa Andblad på Visual Arena, som lät oss arbeta i deras lokaler och använda deras utrustning.

Innehåll

Figurer	xiii
Ordlista	1
1 Inledning	3
1.1 Syfte	3
1.2 Avgränsningar	4
2 Teori	5
2.1 Virtual reality	5
2.1.1 Åksjuka i VR	5
2.1.2 Interaktion	6
2.2 Rollspel	7
2.3 Fog of war	7
2.4 Nätverk	7
2.4.1 Replikering	8
3 Bakgrund	9
3.1 VR-utrustning	9
3.2 Digitala versioner av brädspel och rollspel	9
3.3 Tabletop Simulator	10
3.4 Stenkross Studios	10
4 Metodik	11
4.1 Spelmotor	11
4.2 Nätverksarkitektur	11
4.3 Synkronisering i Unreal Engine	12
4.4 Versionshantering av kod	12
4.5 Testning av prototyp	12
5 Genomförande	13
5.1 Planering	13
5.2 Förstudie	14
5.2.1 Besök hos Stenkross Studios	15
5.3 Utveckling	15
5.3.1 Grundprototyp	15
5.3.2 Fog of war	16

5.3.3	Flerspelarläge	18
5.3.4	Grab and drop	19
5.3.5	Tärningar	19
5.3.6	Turbasering	20
5.3.7	Karaktärsblad	20
5.3.8	Spelledarmeny	21
5.3.9	Interaktionsmeny	21
5.3.10	Startmeny	22
5.3.11	Feedback till användaren	22
6	Resultat	23
6.1	Prototyp	23
6.1.1	Förflyttning i VR-världen	24
6.1.2	Interaktion i VR	24
6.1.3	Spelfigurer	24
6.1.4	Startmeny	25
6.1.5	Karaktärsblad	27
6.1.6	Rutorna på spelplanen	27
6.1.7	Dörrar	28
6.1.8	Interaktionsmeny	28
6.1.9	Fog of war	28
6.1.10	Möjlighet att attackera andra spelare och monster	30
6.1.11	Turbasering	30
6.1.12	Möjlighet att slå tärning	30
6.1.13	Flerspelarläge	31
6.1.14	Verktyg för spelledaren	31
6.2	Koncept	32
6.3	Hur VR kan berika rollspelsupplevelsen	32
6.3.1	Känsla av närvaro trots avstånd	33
6.3.2	Mer visuellt stöd för inlevelse och koherens	33
6.3.3	Individuell representation av spelvärlden	33
6.3.4	Byta mellan olika perspektiv	33
7	Diskussion	35
7.1	Metodik	35
7.2	Genomförande	35
7.3	Resultat	36
7.3.1	Prototyp	36
7.3.2	Koncept	37
7.3.3	Hur VR kan berika rollspelsupplevelsen	38
7.4	Pålitlighet	38
7.5	Etiska reflektioner	39
7.6	Framtida arbete	40
8	Slutsats	41

Figurer

2.1	Ett nätverksdiagram över hur klienter kommunicerar med en server. Från [1]	8
2.2	Visualisering på hur ett peer-to-peer nätverk kan se ut. Från [2]	8
5.1	Två lösningar som provades för att visualisera en karaktärs liv för spelarna.	21
6.1	Ett headset och två handkontroller som representerar en spelare. . . .	23
6.2	Bild på startmenyn.	25
6.3	Figuren visar hur plattformen som pjäsen står på betar sig. Till vänster visas det för en pjäs som inte är vald, till höger för en vald pjäs. . .	26
6.4	Spelledarknappens tre olika lägen i startmenyn. Till vänster visas hur knappen ser ut om någon annan är spelledare, i mitten när ingen är spelledare, till höger när du är spelledare.	26
6.5	Startknappens två olika lägen i startmenyn, till vänster när spelet inte går och starta och till höger när spelet går att starta.	26
6.6	Ett karaktärsblad i prototypen.	27
6.7	Rutnät som visar vart pjäsen kan flyttas och var den inte kan flyttas.	28
6.8	En av tre dörrar i prototypen.	29
6.9	Interaktionsmenyn för en dörr i prototypen.	29
6.10	Exempel på hur det kan se ut med fog of war i prototypen. I denna figuren så är användaren den gula pjäsen.	30
6.11	Tärning i prototypen som visar siffran 6	31

Ordlista

Termer inom virtual reality

Virtual reality (VR) *sv: virtuell verklighet*, användandet av mjukvara och hårdvara för att ge en upplevelse av att befinna sig i en annan miljö.

Head-mounted display (HMD) *sv: huvudmonterad skärm*, en huvudmonterad VR-utrustning som innehåller en eller två bildskärmar som täcker största delen av synfältet och kan ge en 3D-effekt. Kan ha sensorer som följer användarens huvudrörelser.

Motion Controller *sv: rörelsekontroll*, en handhållen VR-utrustning som innehåller sensorer för att följa handrörelser. Har oftast knappar eller andra reglage för interaktion.

Termer inom rollspel

Rollspel en social aktivitet där flera spelare sätter in sig i olika roller och tillsammans berättar en historia kring dessa roller. Rollerna har ofta numeriska attribut som tillsammans med regler och tärningslag styr berättandet. Även föremål som spelpjäser och spelbräden kan användas i spelet.

Spelledare en spelare i ett rollspel som inte har en egen roll, utan istället ansvarar för att styra berättandet i allmänhet. Spelledaren tar hand om alla mindre roller och har oftast sista ordet i tolkning av reglerna.

Levande rollspel (Lajv) en typ av rollspel där spelarna själva representerar sina roller. Denna typen av rollspel fokuserar ofta mer på improvisation och skådespel där spelarna ofta är utklädda. Spelandet utförs ofta i ett större område med mer fysisk aktivitet istället för sittandes vid ett bord.

Termer inom datorspel

Fog of War *sv: krigsdimma*, en spelmekanik där områden i spelvärlden som inte utforskats döljs i dimma eller mörker och utforskade områden utom uppsikt för spelaren delvis döljs.

1

Inledning

Just nu gör *virtual reality* (VR) genomslag i datorspelsindustrin och VR-utrustning har börjat säljas till vanliga konsumenter. Eftersom tekniken som den ser ut idag inte funnits tillgänglig för konsumenter tidigare så har området VR inom spel förblivit relativt outforskat. Det finns därför stora möjligheter att utforska vad VR kan tillföra inom spel.

Rollspel har i sin traditionella form, bordsrollspel, haft fokus på gemensamt berättande, medan regler och visuella hjälpmedel har varit sekundära. När rollspel tog steget över till datorspel lades fokus på reglerna och det visuella på grund av styrkorna och begränsningarna i mediet. Berättandet i dessa är istället en förutbestämd historia som spelaren tar del av istället för att bestämma.

I den här rapporten kommer möjligheten att utveckla upplevelsen av traditionella rollspel med hjälp av VR att undersökas. Förhoppningen är att kunna utnyttja styrkorna hos digitala spel och nya möjligheter med VR utan att begränsa de berättande och sociala aspekterna av bordsrollspel.

1.1 Syfte

Syftet med detta kandidatarbete är att undersöka hur VR skulle kunna förbättra den traditionella spelupplevelsen för bräd- och rollspel. För att göra detta kommer en prototyp att utvecklas. Prototypen kommer att vara ett datorspel med stöd för flera spelare samt en spelledare. Spelledaren kommer att ha mer frihet och verktyg för att påverka spelet.

Under utvecklingsprocessen kommer bland annat att undersökas hur individuell visuell information i den virtuella miljön kan öppna upp nya spelmöjligheter, men även andra möjligheter och svårigheter med VR kommer att undersökas.

Under projektet kommer inte alla idéer att kunna utforskas i prototypen. Dessa idéer kommer tillsammans med resultat från prototypen att bli en del av diskussionen och rapportens förslag på möjliga svar på frågeställningen “Hur kan VR berika rollspelsupplevelsen?”.

1.2 Avgränsningar

Projektet kommer endast att fokusera på en prototyp för en av de mest kraftfulla typerna av VR-utrustning för konsumenter, Oculus Rift. Det finns många olika typer och varianter av VR-utrustning, men för att prototypen och undersökningarna inte ska begränsas av hårdvara har svagare utrustning valts bort. Stöd för inmatning har begränsats till handhållna rörelsekontroller, specifikt Oculus Touch.

Det existerar även många olika bräd- och rollspel, alla med olika regler och krav. Det har därför beslutats att fokusera arbetet på rutnätsbaserade rollspel, då detta i framtiden kan ge möjlighet till att även implementera andra typer av brädspel som använder sig av rutnät. Av liknande anledning kommer projektet inte heller att fokusera på att automatisera regler, för att inte begränsa spelarens möjligheter att spela olika typer av rollspel. Detta innebär att interaktioner såsom strider, fällor och pussel måste behandlas manuellt av spelaren och en eventuell spelledare.

2

Teori

I teorikapitlet redovisas flera vetenskapliga teorier som är relevanta för att förstå ämnet. Kunskapen hjälper även att förstå hur prototypen är uppbyggd och ger grund till beslut som har tagits under projektets gång.

2.1 Virtual reality

VR handlar om att genom tekniska lösningar ge en användare upplevelsen av att ha en fysisk närvaro i en digital miljö. Det görs genom att begränsa intryck från den lokala omgivningen och ersätta med intryck från den digitala, samt låta användaren påverka den digitala miljön genom naturliga fysiska handlingar. Idén om VR har funnits länge och kan spåras till “The Ultimate Display” [3], en artikel som tar upp nyckelbegreppen om inlevelse i en simulerad värld. Detta är grunden för dagens forskning inom virtuell verklighet.

VR-tekniken har många tillämpningsområden, exempelvis inom området människa-datorinteraktion och TV-spel. På senare tid har tekniken använts i större utsträckning för att ge en rikare upplevelse i konsol- och datorspel [4].

2.1.1 Åksjuka i VR

Åksjuka vid VR-användande, även kallat *cybersickness* eller *simulation sickness*, är något som inte är helt ovanligt att användare av VR drabbas av.

Simuleringsjuka är en form av åksjuka som uppstår vid simuleringar såsom vid användning av VR-utrustning. Detta uppstår när spelaren märker en skillnad mellan hur spelarens huvud rör sig och vad den ser, vilket framkallar yrsel, illamående, huvudvärk och kräkningar [5], [6], [7]. Detta obehag är ett vanligt hinder inom VR-utveckling och något som alla som utvecklar i VR behöver ha i åtanke.

Detta uppstår när det märks en skillnad mellan det vestibulära systemet som känner av huvudets rörelser och orientering, och det visuella systemet, vilket framkallar yrsel och illamående när skillnaden blir för stor.[5]

Det finns ett flertal sätt att undvika åksjuka. En av de viktigaste sakerna att tänka

på är att hålla en tillräckligt hög bildfrekvens. Därför bör applikationer i VR alltid optimeras så mycket som möjligt. Utöver detta bör utvecklare även ha synfältsinställningar i åtanke. Utvecklare bör alltid hålla sig till de värden som den aktuella VR-utrustningen har, och bör inte ändra detta värde eller tillåta användaren att ändra det själv. Detta är på grund av att värdet bör matcha headsetets och dess linsers egna fysiska form [8]. Tester har även utförts av äldre typer av VR-utrustning. Det framgick då att ökat synfält gav högre antal fall av simuleringssjuka [9], [10].

Dessutom är det viktigt att tänka på hur den virtuella miljön interagerar med användaren. Manipulation av användarens kamera, till exempel genom skakningar vid en explosion, bör undvikas då detta kan orsaka simuleringssjuka. Förflyttning av användaren bör alltid göras med en konstant hastighet i motsats till en stegvis accelererande rörelse, för att minimera mängden obehag [8].

Slutligen är det också bra att tänka på att använda svagare ljus och färger än vad man vanligtvis använder sig av. Starkt ljus kan nämligen påskynda simuleringssjuka hos användaren. Vid utveckling bör inte heller manipulera användarens synförmåga, med till exempel rörelseoskärpa, på engelska *motion blur*, [8].

Olika personer är olika toleranta mot simuleringssjuka och användare kan vänja sig vid att använda VR-utrustning och därigenom bygga upp en bättre tolerans mot simuleringssjuka [11]. Det är därför bra att testa sin produkt med så många olika sorters personer som möjligt för att försäkra sig om att man som utvecklare inte orsakar simuleringssjuka [8].

2.1.2 Interaktion

Det finns många problem med spel i VR som skiljer sig från traditionell spelutveckling. Två av grundarna av Oculus har hävdade att användargränssnitt är ett av 3 stora problem som spelutvecklare måste hantera i VR [12].

Vid VR-utveckling så behövs det tänkas på hur menyerna placeras. På en vanlig display placeras menyer oftast ut tvådimensionellt. Det är något som bör undvikas för VR, då detta kan leda till obehag hos användaren. En bättre lösning är att placera ut menyerna i den tredimensionella världen och på så sett tillåta användaren att titta bort från menyerna istället för att menyerna alltid är på samma plats på skärmen [13].

Även övrig interaktion i VR måste tänkas igenom. Om interaktionen ska kännas naturlig behövs andra kontroller än de som vanligtvis används vid spelande. Här måste också bekvämligheten för användaren komma i första hand. En naturlig interaktion som medför att användaren hela tiden måste hålla upp handen i luften medför till exempel ofta att användaren blir alltför trött och ger istället en sämre upplevelse [6].

2.2 Rollspel

Rollspel finns idag i många former, både analogt och digitalt. Det finns rollspel inom många olika genrer, till exempel fantasy, science fiction och skräck. [14].

Den traditionella varianten är det som på engelska kallas *tabletop roleplaying game* eller *pen-and-paper roleplaying game*, på svenska ungefär bordsrollspel. Det spelas vanligtvis i grupp runt ett bord, med en spelledare som styr spelet och ett antal spelare med varsin rollfigur. Spelet rör sig framåt genom att spelledaren berättar vad som händer, och spelarna reagerar på det med sina rollfigurer. Exempel på denna typ av spel är *Dungeons and Dragons* [15] och *Call of Cthulhu* [16].

I digitala rollspel är inte alltid den sociala aspekten så stor del och det finns digitala rollspel med endast en spelare, exempelvis *Dragon Age: Origins*[17]. Men det finns även spel med flera spelare som oftast spelas över internet med oftast ett stort antal andra spelare, det klassiska exemplet är *World of Warcraft*[18].

Gemensamt för både analoga och digitala rollspel är att spelaren tar på sig en roll som någon annan, och spelar och agerar som den rollfiguren. Genom detta skapar spelarna sin egen verklighet och utforskar med de kunskaper som de har. I flera fall utvecklar spelarna också sina sociala färdigheter med andra människor [19].

2.3 Fog of war

Fog of war, stridsdimma på svenska, är en spelmekanik som används för att visa vad spelaren och dess spelfigur kan se. Detta görs genom att dölja de områden av spelarens vy som dennes figur inte kan se, till exempel på grund av för långt avstånd eller att ett hinder blockerar. Det vanligaste scenariot där fog of war används är när spelaren har ett annat perspektiv än sin figur, såsom när spelaren ser spelvärlden ovanifrån. Själva spelmekaniken av fog of war har genom många spel skilt sig ifrån varandra, då detta kan implementeras på många olika sätt beroende på olika krav såsom prestanda och visuell design [20].

2.4 Nätverk

Nätverksarkitekturen hänvisar till hur maskinerna i ett nätverk är uppkopplade till varann. Inom nätverk för datorspel finns det två huvudsakliga typer av arkitekturer: *klient/server* och *peer-to-peer*.

Klient/server-modellen [21] innebär att alla enheter (klienter) är uppkopplade mot en server som hanterar all kommunikation och delandet av information mellan klienter.



Figur 2.1: Ett nätverksdiagram över hur klienter kommunicerar med en server. Från [1]

Peer to peer [22] innebär att alla enheter i nätverket är sammankopplade med varandra och att information skickas direkt mellan enheter utan att hanteras av en central server.



Figur 2.2: Visualisering på hur ett peer-to-peer nätverk kan se ut. Från [2]

2.4.1 Replikering

Nätverksarkitektur i spel består generellt av spegling av information och utskick av kommandon. Detta kallas även för replikering av variabler och funktioner.

Detta innebär att servern alltid har sista ordet vad det gäller spelets tillstånd. För att en klient ska kunna påverka spelet måste klienten skicka ett meddelande till servern, där servern hanterar uppdateringen av spelets tillstånd och att förmedla alla ändringar till alla klienter. Saker som inte är kritiska för spellogiken såsom grafiska effekter synkroniseras inte med servern för att minska belastningen.

Funktionaliteten för replikerade funktioner och variabler finns inbyggd i Unreal Engine. Detta skapas med hjälp utav ett *custom event* [23].

3

Bakgrund

Bakgrundskapitlet handlar om relaterade produkter eller tjänster till ämnet. Olika produkter som är behövliga och gynnsamma för att använda sig av den skapade prototypen nämns.

3.1 VR-utrustning

För att kunna uppleva VR måste användaren ha ett VR-headset. Det krävs i många fall även en kompatibel dator eller spelkonsol, men det finns även mindre kraftfulla varianter som använder sig av mobiltelefoner. Det finns ett flertal olika uppsättningar av VR-utrustning för privatpersoner. Några populära alternativ av det mer kraftfulla slaget är HTC Vive [24], Oculus Rift [25] och PlayStation VR [26].

Både HTC Vive och Oculus Rift kopplas till en dator, medan PlayStation VR behöver kopplas till ett Playstation 4 [27]. Alla tre varianter använder sig av medföljande tillbehör för att spåra användarens rörelser. Till skillnad från Rift och Playstation VR som inte tillåter någon större förflyttning, så tillåter HTC Vives system en större spelyta som spelaren kan gå runt i.

För att kunna interagera med den virtuella världen krävs även någon typ av kontroll. Vissa headset har stöd för traditionella handkontroller för konsoler, bland dessa Oculus Rift och PlayStation VR. För alla tre varianter som nämnts ovan finns dock handkontroller som har utformats speciellt för att användas i VR (ofta benämnda vid samlingsnamnet *motion controllers*).

Det finns även headset som inte har någon egen skärm, utan som istället använder sig av en mobiltelefon som skärm. Exempel på sådana headset är Google Cardboard [28] och Samsung Gear VR [29]. Dessa är billigare men mindre kraftfulla alternativ för VR.

3.2 Digitala versioner av brädspel och rollspel

Många brädspel och rollspel idag har både digitala och analoga versioner. Några exempel på brädspel med digital versioner är Ticket to Ride [30] och Smallworld

[31]. För rollspel finns digitala verktyg med stöd för olika rollspelssystem. De ger spelledaren möjlighet att dela information med spelarna och underlättar spelandet. Exempel på sådana verktyg är Fantasy Grounds [32], Roll20 [33] och Undercurrents [34].

Traditionella brädspel och rollspel är klassat som sällskapsspel eftersom den sociala aspekten utnyttjas för att spela [19], [35]. Men i många fall överförs dessa relativt oförändrat in i den digitala världen, med en digital spelplan, men ofta utan den sociala aspekten. I många digitala spel saknas ögonkontakt och social interaktion [36], vilket skiljer sig från den upplevelse som roll- och brädspel traditionellt ger.

Men det finns även mer traditionella brädspel som använder sig av digitala hjälpmedel, exempelvis XCOM: The Board Game [37] och Mansions of Madness Second Edition [38]. Dessa har speciella appar, tillgängliga för dator eller mobil, som syftar till att berika spelupplevelsen. I båda fall används apparna för att ge mer omväxling i spelet genom att slumpmässigt ändra vissa spelmoment mellan spelsessioner, och även för andra syften såsom att hålla ordning på regler. Ett annat sätt att införa digitala element i traditionella brädspel är genom elektroniska komponenter [39].

3.3 Tabletop Simulator

Tabletop Simulator är ett existerande datorspel av sandlådetyp som simulerar en miljö med ett bord, där olika *tabletop games*, ungefär bordsspel, kan spelas [40]. Spelet har även ett VR-läge med stöd för HTC Vive-enheter. Spelet har även realistisk fysik samt lokalt och online-flerspelarläge. I grundutbudet finns ett antal vanliga spel implementerade, såsom schack och poker, men det finns även ett rollspelsläge och ett fritt läge. Det finns även ytterligare tillägg (*DLC, downloadable content*) som kan köpas till som lägger till digitala versioner av existerande, traditionella brädspel.

3.4 Stenkross Studios

Stenkross Studios är en spelstudio som startades av en grupp tidigare Chalmers-studenter. Under sitt kandidatarbete som de gjorde våren 2016 utvecklade de ett strategispel i VR, Tablemen. De valde sedan att fortsätta arbeta med utvecklingen av spelet efter sitt kandidatarbete. Numera heter spelet istället Airborne Empires [41].

4

Metodik

Det finns många olika metoder och verktyg som kan användas, både för att programmera, planera och styra upp projekt. Under detta projekt kommer ett antal av dessa att användas för att underlätta arbetet.

4.1 Spelmotor

För att korta ner mängden kringarbete innan projektets frågeställning kunde tacklas bestämdes det att en spelmotor skulle användas.

Det finns en mängd spelmotorer på marknaden. De två mest kända spelmotorerna som erbjuder gratisversioner är Unity [42] och Unreal Engine 4 [43]. Båda tillåter att släpp av ickekommersiella produkter och tar inte ut några avgifter för det. Båda motorerna har även stöd för VR. De har även båda två stora användarbaser så finns det gott om information att hitta om eventuella problem.

Unreal Engine har dock andra fördelar över Unity. Unity programmeras direkt i C#, medan Unreal Engine erbjuder både ren C++-programmering och programmering i deras egna språk Blueprints. Blueprints är ett grafiskt skriptspråk, vilket troligtvis kommer göra det enklare att utveckla, då ett flertal gruppmedlemmar inte tidigare arbetat med varken C# eller C++. Om det skulle behövas göra stora förändringar har dock Unreal Engine bättre stöd för detta, då man som användare har tillgång till spelmotorns källkod och kan, vid behov, anpassa spelmotorn. Strukturen på Unreal Engine och dess utvecklingsmiljö gör även att utvecklingen av separata prototyper som senare integreras till en större prototyp är enkelt att göra.

På grund av dessa fördelar valdes Unreal Engine som spelmotor för projektet.

4.2 Nätverksarkitektur

Eftersom prototypen ämnas ha flerspelarläge för spelare över nätverk, behövdes efterforskning om nätverk. Då Unreal Engine stöder klient/server-modellen [44] så kommer prototypen använda sig av denna.

I prototypen kommer en dedikerad server användas. En dedikerad server innebär kortfattat att den simulerar själva spelvärlden och logiken, men tar inte hand om de grafiska delarna. Den behöver inte heller något användargränssnitt.

Användning av en dedikerad server ger flera fördelar. Klienterna behöver inte bearbeta så mycket information samt ingen stor bandbredd-uppstömning krävs av klientsidan. En annan fördel är att servern tar hand om all viktig data, vilket gör det svårare för spelarna att manipulera spelet och dess data med syfte att fuska. Därför är det lämpligast med en klient/server-modell för multiplayer-spel och är det som vi kommer att använda oss av i vår prototyp.

Denna modell kräver även en server-värd. Prototypen kommer att använda sig av en server från Amazon Web Services, vilket är ett stort och pålitligt företag som bör garantera att servern kan vara uppe i stort sett hela tiden.

4.3 Synkronisering i Unreal Engine

För att flerspelarfunktionaliteten skulle vara tillgänglig så kommer prototypen följa Unreal Engine designmönster för implementation av nätverk. Unreal Engine använder sig som tidigare nämnts av klient/server-modellen. Detta fungerar tack vare replikeringen som finns inbyggt i spelmotorn [45].

4.4 Versionshantering av kod

Versionshanteringssystemet Git[46] och sidan GitHub[47] kommer användas för att dela och arkivera prototyper och förändringar, detta för att systemet är smidigt och har använts tidigare av gruppmedlemmarna.

4.5 Testning av prototyp

För att testa olika egenskaper och funktioner i prototypen kommer systematiska användartest att köras. Innan testningen skrivs relevanta uppgifter och frågor upp som användaren ska utföra och svara på under testet. Under testningen beskrivs dessa uppgifter för användaren stegvis. Personen som sköter testningsprocessen kommer vara helt tyst tills användaren har utfört uppgifterna, men kommer att ta anteckningar på användarens tankar kring uppgifterna. Anteckningarna kring frågorna och uppgifterna tas sedan upp med resten av gruppen och prototypen kompletteras utifrån detta. Testningssystemet är en variant som liknar *thinking aloud* [48], där användaren gör olika uppgifter och säger allt som kommer på tanke under processen.

5

Genomförande

Genomförandet skedde i fyra faser. Den första fasen bestod av att lägga upp en plan och en viss struktur på prototypen om hur den skulle se ut och vad som skulle finnas med. Den andra fasen, förstudien, skedde delvis parallellt med planeringen. I den tredje fasen, som var den längsta, implementerades allt det funktionella och flera av de beslut som togs under planeringsfasen ändrades och reviderades under denna. Den fjärde fasen bestod av upprepning och hantering av olika visuella aspekter samt buggar.

5.1 Planering

Arbetet startade med att ta fram en idé om hur prototypen skulle vara samt gjorde en plan på hur vi skulle nå dit. Ingen av gruppmedlemmarna hade tidigare erfarenhet av VR-utveckling. Det bestämdes därför att projektet initialt behövde en upplärningsfas.

Gruppen diskuterade vilken funktionalitet som önskades i prototypen. Efter att ha diskuterat och prioriterat kom vi fram till ett antal saker som vi ville ha med.

Vi planerade en grundprototyp med följande funktioner:

- En rutnätsbaserad spelplan
- Möjligheten att förflytta sig och interagera med spelvärlden i VR
- En representation av spelaren med utrustning i spelvärlden
- Individuella spelfjäser på spelplanen
- Dörrar som kan öppnas och stängas
- Möjligheten att spela flera spelare samtidigt över internet

Följande funktioner planerades att få in efter att den grundläggande funktionaliteten var implementerad:

- Startmeny för att välja roll med mera
- Möjligheten att slå tärningar i spelvärlden

- Karaktärsblad med information om spelarens roll
- Individuell information i form av en fog of war (se 2.3)

Samt följande funktioner i mån av tid:

- Spelledarroll med större frihet och egna verktyg
- Verktyg för att redigera spelplanen inuti spelet
- Strider med monster som styrs av spelledaren
- Turbasering med begränsat antal handlingar
- Förstapersonsperspektiv
- Ljudeffekter
- Annan individuell information såsom mörkersyn
- Dolda dörrar och fällor
- Fler våningsplan

Följande bedömdes intressant men inte något vi gör i projektet:

- Stöd för röstkommunikation i spelet
- Föremål och inventory-system
- Anpassningsbar spelarrepresentation och spelpjäser

De första veckorna ägnades sedan åt att lära sig grunderna i Unreal Engine 4 innan utvecklingen för prototypen kunde påbörjas. Det gjordes även undersökningar angående hur önskad funktionalitet skulle fungera. Under denna period var tanken att respektive gruppmedlem skulle fördjupa sig inom ett område och förhoppningsvis implementera en lösning i ett eget projekt. Tanken var att sedan slå ihop våra lösningarna för att få fram en bra grund att jobba vidare med.

5.2 Förstudie

Efter den initiala planeringen började gruppen med att testa och introducera sig till VR och Unreal Engine. Mycket tid i början lades på att undersöka Unreal Engine.

I detta stadiet lades också en del tid på litteraturstudier av design och problem för VR (se 2.1), men även av rollspel (se 2.2). Dessutom undersöktes vilka liknande produkter som redan fanns på marknaden.

5.2.1 Besök hos Stenkross Studios

Ett studiebesök gjordes hos spelstudion Stenkross Studios. De delade med sig av ett flertal erfarenheter av utveckling av VR-spel.

De ansåg att en av de viktigaste aspekterna när man utvecklar för VR är interaktion. Om en spelare till exempel upplever en onaturlig rörelse eller om spelet beter sig annorlunda än vad som förväntas så finns risk för illamående eller disorientering hos spelaren. Detta är alltså samma problem som tidigare i rapporten benämnts som åksjuka (se 2.1.1).

I sin ursprungliga version av spelet använde de sig av Leap Motion[49]. Leap Motion kan användas tillsammans med en HMD och innehåller sensorer och programvara som identifierar formen av händer och deras rörelser. Tanken var att spelarna skulle kunna använda vanliga gester utan handkontroller, men det gav inte den effekt som de hoppades på. Istället upplevde de att bristen på fysisk feedback från spelvärlden gav en onaturlig känsla. Stenkross Studios nämnde även att spel med Leap Motion i dagsläget inte är särskilt populärt och att handkontroller är att föredra.

Det tredje stora problemet som uppkom för dem var prestanda. Eftersom de använder sig av många objekt som förflyttar sig i spelet så måste det vara mycket väloptimerat. För dem blev det en klar förbättring av att göra om koden i Blueprints till C++-kod istället. Grafiken måste även vara optimerad. Några tips som gavs var att hålla sig till en textur till varje modell, att inte använda sig av dynamiska skuggor och att undvika transparanta modeller.

5.3 Utveckling

Projektets utvecklingsprocess delades upp i två olika delmoment, där det första delmomentet var att utveckla en grundversion av prototypen som sedan kunde användas som en plattform för vidare arbete. Det andra delmomentet var att parallellt utforska och implementera olika typer av VR-fokuserad funktionalitet som kunde tros förhöja spelupplevelsen.

5.3.1 Grundprototyp

Det som identifierades som det mest grundläggande var rutnätet och hur spelarna skulle förflytta sig på detta. Varje ruta kände till alla närliggande rutor. Tanken med detta var att det skulle användas enare för att räkna ut vilka rutor en spelare kan gå till. Till en början var denna information statisk och manuellt angiven för varje enskild ruta. Vi insåg snabbt att prototypen behövde en bättre lösning för att undvika en stor mängd arbete vid nya eller större kartor. Problemet löstes genom att rutorna istället känner av om det finns några andra rutor i närheten och sparade den informationen när prototypen startades. I och med detta så behövdes inget extra

arbete göras utöver att placera ut rutorna när kartan skapades. Vi letade inspiration på hur kartan kunde se ut och placerade ut rutorna i ett sådant mönster.

För att kunna utveckla och köra prototypen behövdes även någon form av kameranavigering. Vi kom fram till att vi också behövde stöd för navigation med mus och tangentbord förutom navigation med handkontroller. Eftersom vi valde en vy där kameran tittar snett nedåt mot spelplanen blev förflyttningen inte trivial att implementera. Vi upptäckte att det kändes onaturligt att röra sig närmare marken vid förflyttning framåt. Därför valde vi att förflytta spelaren i ett plan parallellt mot spelplanen istället, något som krävde lite mer komplicerad matematik. Högst prioritet fick förflyttning med mus och tangentbord eftersom vi oftast skulle utveckla utan tillgång till VR. De flesta funktionerna designades även med möjlighet att använda mus och tangentbord för att lätt kunna testas.

Första implementationen av mus och tangentbordsstyrning gjordes enbart i ett plan utan rotation av kameran och fungerade bra, den inkluderade även zoom och ändring av observationsvinkel. Zoomen implementerades enbart som en förflyttning i höjddled, vilket vi bedömde som tillräckligt. När rotation skulle implementeras blev denna implementation av planförflyttningen tvungen att skrivas om med mer komplicerade matematiska formler vilka tog längre tid att reda ut och få korrekta. En första variant av förflyttning och rotation var centrerad på punkten där muspekaren träffade spelplanen, och kameran följde musens rörelser. Den varianten valdes bort då den bedömdes som opraktisk, så en variant där spelarens position styrdes med tangentbord och rotationen endast gjordes runt kamerans position implementerades istället.

När förflyttning i VR skulle implementeras behövdes problematiken med åksjuka (se 2.1.1) tas i åtanke. Vi tog inspiration från vårt besök på Stenkross Studios (se 5.2.1) där de utvecklar ett spel med liknande perspektiv som vår prototyp. Deras förflyttning baseras på att med en handkontroll ta tag i spelplanen och dra för att förflytta sig och att ta tag med båda handkontrollerna och dra dem mot och från varandra för att zooma in och ut samt rotera. Eftersom de redan haft åksjuka i åtanke och arbetat fram förflyttningen valde vi en liknande förflyttning. Vi valde dock att inte ta tag i spelplanen utan istället använda positionen i rummet, för att vår spelplan är mer ojämn. Med Unreal Engines stöd för handkontroller var det lätt att få ut positioner relativt till spelaren, men matematiken kring förflyttningen krävde mycket arbete att få rätt.

5.3.2 Fog of war

För projektet undersöktes ett flertal tillvägagångssätt (se 2.3). Dessa analyserades på ett teoretiskt plan och några undersöktes sedan ytterligare genom att implementeras.

När det kom till den visuella representationen för fog of war så undersöktes fyra olika metoder.

- **“Fysiskt” objekt:** Tekniken bygger på att ett objekt används i den “fysiska” världen, vilket menas att objektet är ett objekt i spelvärlden. Objektet i form av ett ensidigt plan sträcks ut och placeras så att det täcker hela spelnivån, detta för att blockera sikten för spelaren. En modifierbar textur används sedan under spelets gång för att skapa öppningar i planet på den yta där spelaren har förmågan att se.
- **Post-process material:** Liknande teknik som den föregående men i detta fall används inte ett “fysiskt” objekt, utan ett post-process-material som modifierar spelarens vy. En modifierbar textur används även i detta fall, men istället av materialet, som använder texturen som utgångspunkt för att modifiera den vy som spelaren får av spelvärlden.
- **Ljusmanipulering:** Ljusmanipulering bygger på att utnyttja spelmotorns redan implementerade ljusmekanik för att representera fog of war. Detta genom att lysa upp området som spelarens spelpjäsa har syn av och lämna resten utan upplysning.
- **Osynliga objekt:** Denna metod bygger på att dölja spelvärlden genom att endast låta de objekt i spelvärlden som spelpjäsen kan se vara synliga och hålla resten av dem osynliga.

Förutom den visuella representationen undersöktes även designfrågan rörande begränsningar av spelpjäsens synområde. Valet av detta begränsas inte lika mycket av kraven på fog of war från projektet, men påverkar implementationen. Tre olika versioner diskuterades inom gruppen, tagna från egna erfarenheter av olika spel.

- **Rumbaserat:** Den design där spelpjäsen har syn över allt inuti vad som är definierat som ett och samma rum. Att avtäckas kartan i större områden är ett relativt vanligt tillvägagångssätt i fysiska rollspel då detta förenklar avtäckningsprocessen.
- **Line of sight:** Bygger på en mera realistisk design där spelpjäsen endast kan se det som är tillräckligt nära och som inte blockeras av något så som till exempel en vägg.
- **Radiesyn:** I denna design låter man spelpjäsen se allt inom en viss radie, även om detta realistiskt skulle vara blockerat av något annat.

Under fog of wars utvecklingsprocess upptäcktes det att vissa krav behövde sättas på dess förmågor, vilket innebar att flera av de undersökta varianterna inte längre var några bra alternativ på grund av dess begränsningar. Dessa krav blev främst uppenbara vid implementeringen av tidigare versioner, vilka då itererades vidare på fram till den nuvarande versionen. Till exempel uppkom vissa av dessa för att det annars dolde element i spelvärlden som behövde vara synliga.

- Den visuella representationen måste tillåta att spelaren kan se spelvärlden från alla möjliga vinklar och perspektiv.
- Områden som slutar vara under sikt av spelpjäsen bör övergå till en halvdold

nivå där världen fortfarande går att se.

- Ska bygga på principen att andra pjäser endast är synliga om spelarens pjäs har syn över dess område.
- Spelarna ska aldrig själva döljas av fog of war oberoende av var de befinner sig.
- Menyerna bör aldrig döljas av fog of war.
- Behöver kunna hantera att spelvärlden innehåller stora objekt.

Designen som används för att uppnå kraven för projektet använder ett post-process-material för att visa spelpjäsernas "line of sight"-syn. Genom att använda ett post-process-material som visuell representation så tillät detta spelaren att kunna se spelvärlden från alla möjliga vinklar och perspektiv, samtidigt som det också tillät spelare och menyer att bli exkluderade från fog of war. Spelpjäsernas syn baserades på "line of sight" då detta gav ett mera realistisk form av syn och dess maximala radie kan då också manipuleras beroende på spelpjäsernas status, exempelvis en pjäs som ser dåligt i ett mörkt rum.

5.3.3 Flerspelarläge

I början när implementationen för prototypen påbörjades så hade vi inte i åtanke hur servern och klienter skulle kommunicera med varann. Detta märktes senare och bidrog med en del problem när vi skulle testa prototypen med två spelare i samma värld. Replikationen skedde aldrig på klientens sida utan endast på serverns sida, servern fick alltså aldrig information om att klienten hade förflyttat sig. Omstrukturering utav kod behövdes för att ändra logiken så att klient/server-modellen skulle fungera. Det tog lång tid innan vi fick detta att fungera eftersom mycket kod var tvungen att skrivas om och bli flyttad så att bara servern hanterade den koden och inte klienterna. Efter att vi löst de största problemen och det gick att se spelarna replikeras från både klient- och serversidan så var vi tvungna att för varje ny funktionalitet alltid ha i åtanke ifall servern skulle ta hand om denna eller inte vid implementering.

För att den dedikerade servern skall fungera behövde servern först kompileras. För att göra detta användes Visual Studio [50]. Kompileringen kunde ta upp till 6 timmar att genomföra. Resultatet efter en färdigbyggd kompilering resulterade i två olika filer.

Server.exe är själva filen som ligger på Amazons virtuella maskiner, filen gör så att uppkoppling mot servern är möjlig. När filen körts lyssnar servern på trafik mot port 7777.

Klient.exe är själva filen som alla användaren ska ha. När användaren trycker på filen så sker det en automatisk uppkoppling till servern.

Ett problem som uppstod när vi hade kompilerat klart den dedikerade servern var

att det inte gick att få någon koppling mellan den virtuella datorn och datorn som ville koppla sig till servern. Efter flera felsökningar så visade det sig att detta berodde på att en specifik port som Unreal Engine använder sig av för dedikerade servrar behövde öppnas samt lägga till en regel i brandväggen som låter alla att koppla sig till den här porten. När klienten nu kunde koppla upp sig mot servern så uppkom ett annat dilemma som var att klienten inte startade spelet i VR, detta fixades däremot enkelt med ett funktionsanrop i Unreal Engine.

5.3.4 Grab and drop

Att kunna plocka upp någonting i VR-miljön och sedan släppa det genom att använda en naturlig rörelse låg högt upp på prioritetsslistan. Detta system var tänkt att användas till flera olika system, såsom tärningar samt för att plocka upp pjäser och släppa dessa på rutorna. Tanken var att få en äkta brädspelskänsla, som när till exempel schackpjäser flyttas.

Systemet tog ganska lång tid att utveckla fullt. Det fanns många små buggar och allt fler upptäcktes under våra egna tester. Buggarna fixades allt eftersom arbetet fortgick. Systemet gjordes om en gång och refaktorerades helt för att förbättra koden.

Ett problem uppstod när spelaren försökte greppa tag i något i spelvärlden, eftersom det kräver att kontrollen eller handen måste i princip vara på det objektet som spelaren vill ta tag i. Användaren behövde röra sig långt för att ens nå objektet, eller släppa det på rätt plats. Skalan på spelvärlden ändrades för att undvika detta.

Under användartester av systemet skapade en annan aspekt förvirring. Om spelaren försöker interagera med till exempel en dörr som lasern pekar på, men har kontrollen för nära en pjäs kommer pjäsen att plockas upp istället för att interagera med dörren. Problemet löstes genom att inaktivera lasern och interaktionsmöjligheten när kontrollen är nära något som kan plockas upp.

5.3.5 Tärningar

Tärningar låg högt upp på prioritetsslistan eftersom det används väldigt ofta i rollspelsyften. Det fanns olika tankar och sätt hur vi kunde representera dessa tärningar. Några idéer var baserade på en slags slumpgenerator som bara visade nummer där värdena sparades. Genereringen av nummer skulle då ske genom ett klick på en meny eller liknande och skulle bara vara visuellt. Andra idéer var istället baserade på fysiska tärningar vilket representeras som riktiga tärningar, alltså till exempel fyrasidiga, åttasidiga, tiosidiga eller tjugosidiga. För att slå nummer måste dessa tärningar tas upp med handen och kastas så får gravitationen och fysiken hantera resten. Vi bestämde oss för att implementera det andra alternativet då det är mer likt det naturliga rörelsen. Inga praktiska jämförelser gjordes dock mellan de två olika alternativen.

Tärningssystemet var relativt enkelt att implementera eftersom delarna som krävdes för att kunna plocka upp och kasta tärningen var redan implementerade (se 5.3.4).

Efter intern testning upptäcktes att spelare lätt kunde fuska genom att släppa tärningen prakt ned för att få det värde på tärningen som önskades. En lösning som provades var att lägga till en liten rotation när tärningen släpps. När denna lösningen provades gav det en underlig känsla, eftersom den roterade extra mycket när den kastades den normalt.

5.3.6 Turbasering

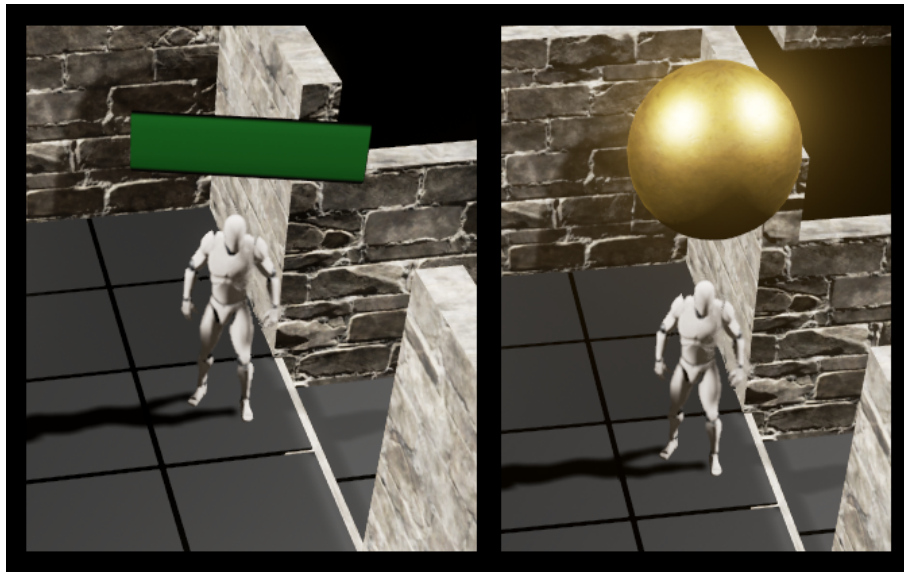
När vi skulle implementera turbaseringen så diskuterade vi två olika typer av idéer. Den ena var att spelledaren skulle sköta turbaseringen manuellt, utan hjälp från prototypen. Den andra idén var att skapa ett system i prototypen som hjälper spelledaren att hantera turbaseringen, där spelarna begränsas då det inte är deras tur. Vi valde det senare alternativet eftersom det kan vara jobbigt ifall en spelare, av misstag eller med flit, attackerar ett monster eller flyttar sin pjäs när det inte är dennes tur.

I systemet kunde spelledaren styra och välja vems tur det är och endast en person i taget utöver spelledaren kunde interagera med olika objekt. I den första versionen av systemet så kostade varje interaktion som gjordes ett antal handlingar, *actions*, som angav hur många interaktioner en spelare kunde göra innan det blir nästa persons tur. Vi valde dock senare att ta bort handlingar från turbaseringssystemet, då vi kände att det begränsade möjligheterna i prototypen.

5.3.7 Karaktärsblad

I prototypen var tanken att varje karaktär skulle ha olika egenskaper som exempelvis snabbhet, styrka och liv. Här tittade vi på flera olika lösningar för att visa dessa egenskaper för spelarna, där den första var *heads-up display* (HUD) [51] som ofta används för detta ändamål vid spel på en tvådimensionell skärm. Detta var dock något som vi insåg att vi ville undvika eftersom HUD inte är optimalt för VR-applikationer (se 2.1.2). Vi hade även två andra förslag som bara visade karaktärens liv, en tvådimensionell liggande stapel (HP-bar) samt en tredimensionell kula. Båda dessa placerades strax ovanför karaktären (se Figur 5.1).

Det slutgiltiga valet föll till sist på ett karaktärsblad som spelarna själva kan plocka fram. Det var främst två anledningar till att det blev just karaktärsbladet. Det första var att vi här kunde visa fler egenskaper än bara livet som HP-baren och kulan gjorde. Det andra var att HP-baren och kulan var lite klumpiga i VR-världen, då de i vissa vinklar skymde karaktärerna. Därför kändes det till slut som ett ganska självklart val med karaktärsblad som spelarna inte behöver se hela tiden och kan plocka fram vid behov, som dessutom samlade alla egenskaper på ett ställe.



Figur 5.1: Två lösningar som provades för att visualisera en karaktärs liv för spelarna.

5.3.8 Spelledarmeny

Utrymmet för att improvisera är betydligt mindre i ett digitalt spel jämfört med i bordsrollspel. Vi ville få in åtminstone en del av detta i vår prototyp.

Vår ursprungliga idé var någon typ av kartredigerare som kunde användas av spelledaren under spelets gång. Vi valde att börja med en enkel version av detta, en meny där spelledaren kunde välja olika objekt att placera ut i spelvärlden. Tanken var att ha detta som ett föremål som spelledaren kunde plocka fram i spelvärlden, då det är att föredra i VR framför tvådimensionella menyer (se 2.1.2).

När detta skulle implementeras var tanken att det skulle använda grab and drop-systemet som redan fanns implementerat (se 5.3.4). Detta gav dock problem när vi ville placera ut monster, då systemet inte var anpassat för att släppa spelpjäser som inte redan var utplacerade på kartan. Detta visade sig vara mycket svårt att lösa.

5.3.9 Interaktionsmeny

Interaktionsmenyn var inte med i vårt ursprungliga koncept. Behovet av denna funktion uppstod när vi insåg att spelare kan få mer ett interaktionsalternativ per objekt, men att spelaren inte kunde välja vilken av dessa som skulle utföras. En dörr i spelvärlden kan till exempel både öppnas och låsas.

För att hantera detta problemet implementerades en slags meny som placeras över objektet som spelaren interagerar med och visar vilka alternativ som finns. Initialt var varje menyval beskrivet med text, men efter efterforskning och kontakt med

andra grupper beslutades det att ikoner är att föredra vid användning i VR. Texten hade behövt vara mycket stor för att bli tydlig i spelvärlden, vilket inte passar så bra in i en liten meny som inte ska ta upp så mycket plats.

Användartester på detta bekräftade att detta var ett smidigt och bra sätt att interagera med, men användarna visste inte om att den fanns.

5.3.10 Startmeny

En funktion som vi ganska tidigt var överens om att prototypen skulle innehålla var att varje spelare innan spelet börjar ska kunna välja en spelarpjäs att använda i spelet. Spelarna skulle kunna välja mellan minst två olika spelarpjäser som skulle ha olika egenskaper som exempelvis olika steglängder. För att detta skulle bli möjligt så bestämdes det att en enkel startmeny skulle implementeras. Alternativet som fanns här var att göra en två- eller tredimensionell meny. Valet blev det senare alternativet, ett tredimensionellt rum där objekten som spelaren ska interagera med är placerade i rummet. Detta är vanligtvis ett bättre alternativ när man utvecklar för VR (se 2.1.2).

5.3.11 Feedback till användaren

I början var det ett stort fokus att implementera nya fungerade funktioner i prototypen. Under denna perioden så tänktes det inte på hur användaren skulle uppleva prototypen och hur prototypen skulle visa för denna huruvida exempelvis någonting går att interagera med eller inte. Under vecka 13 innehöll prototypen i princip bara två saker som hjälpte användaren att förstå prototypen. Den första var att de rutor som spelpjäsen kunde gå till bytte material för att tydligt urskilja dessa från övriga rutor. Även den rutan som spelaren för tillfället befann sig på bytte till ett material som stack ut. Den andra saken som underlättade för användaren var att dörrarna öppnades och stängdes när användarna interagerade med dem.

Men senare började användarupplevelsen att prioriteras högre. Först implementerades fler materialbyten på de objekt som redan fanns för att öka visualiseringen och ge användaren feedback på de interaktioner som gjordes. I huvudsak så var det materialbyten som skedde när användaren höll markören/lasern på objekt. Detta resulterade i att till exempel dörren lyste upp om den kunde interageras med den och att rutorna lyste upp i en ljusare färg eller blev röda när spelaren höll över dessa objekt.

Det fanns dock fortfarande vissa händelser i prototypen som inte gav användarna någon feedback då det var svårt att implementera visuella effekter på dessa. Ett av dessa fall var när spelledaren låste eller låste upp en dörr. Detta gjorde att ljud implementerades i prototypen för att komplettera den visuella feedbacken. För exemplet med dörren så innebar detta att det nu hörs ett dörrlås som vrids om när den blir låst eller låses upp.

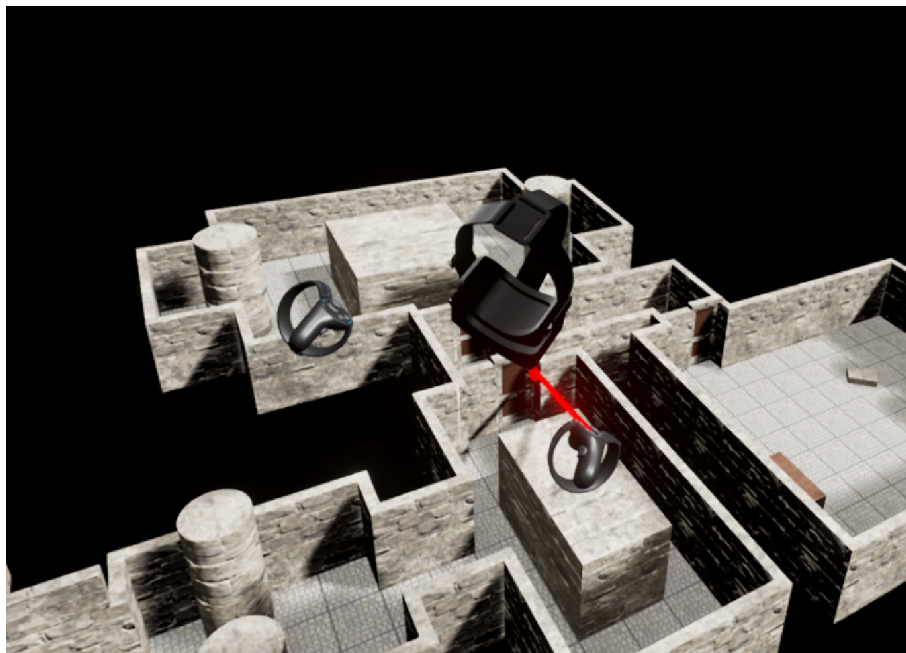
6

Resultat

Det slutgiltiga resultatet består av flera olika resultat vilket är uppdelat i kapitlet. Resultatet från prototypen, designmässigt resultat, hur lik prototypen är konceptet och hur VR kan användas för att göra ett intressant rollspelande.

6.1 Prototyp

Den slutgiltiga prototypen består av ett spel utvecklat i Unreal Engine 4 med stöd för Oculus Rift med Oculus Touch. Prototypen har stöd för flera spelare. I prototypen representeras spelarna av en modell som föreställer ett headset och handkontroller motsvarande spelarens utrustning som följer spelarens rörelser. Dessa är synliga för alla spelare. Spelarna är positionerade ovanför en rutnätsbaserad spelplan där varje spelare får en spelpjäs utplacerad, förutom spelaren som tar på sig rollen som spelledare.



Figur 6.1: Ett headset och två handkontroller som representerar en spelare.

6.1.1 Förflyttning i VR-världen

För att användaren ska kunna ta sig runt och se hela spelplanen har kontroller för förflyttning implementerats. Spelaren förflyttar sig genom att hålla inne en specifik knapp på endast en av handkontrollerna och sedan dra sig runt i omgivningen. Genom att hålla samma knapp nedtryckt på båda handkontrollerna samtidigt så kan spelaren istället rotera världen samt förflytta sig närmare eller längre ifrån objekt.

6.1.2 Interaktion i VR

Det finns två olika sätt att interagera med världen i VR. Den ena sättet är att använda sig av en laserpekare som pekar från högerhanden i prototypen. Det andra är att greppa tag i och flytta runt objekt med realistiska rörelser.

För att interagera med hjälp av laserpekaren riktar spelaren lasern mot ett objekt och trycker sedan på knappen för interaktion. Håller spelaren istället in knappen ett litet tag så kommer interaktionsmenyn fram (se 6.1.8). Det går även att interagera med objekt utan att använda sig av menyn genom att släppa interaktionsknappen innan menyn dyker upp. Detta kan vara smidigt när spelaren ska interagera med objekt som bara har en möjlig interaktion. Om denna variant används på ett objekt med fler än en möjlig interaktion så görs första möjliga interaktionen utan att spelaren själv kan välja vilken.

Med laserpekaren kan spelaren utföra alla interaktioner i prototypen, förutom att slå tärning. Det kan endast göras genom att greppa tag i tärningen.

När spelaren har kontrollen tillräckligt nära ett objekt som går att plocka upp och trycker på interaktionsknappen plockas detta upp. Objektet fastnar då på kontrollen. Släpps knappen så släpps även objektet och faller nedåt. När kontrollen kommer så nära ett objekt att det kan plockas upp försvinner lasern. Flyttar spelaren kontrollen ifrån objektet blir lasern synlig igen. Det går dock inte att använda sig av denna funktionen för att öppna dörrar eller andra liknande interaktioner, utan bara för att plocka upp och släppa objekt som är upplockbara.

6.1.3 Spelfigurer

I prototypen finns fyra olika pjäser som spelarna kan välja. För att visuellt skilja dessa från varandra har alla olika färger och former. I spelet har även pjäser olika egenskaper som att de exempelvis kan gå olika långt, vara olika bra på attacker eller ha olika mycket liv.

I spelet finns det även pjäser som representerar monster. De har en egen färg för att inte ska blandas ihop med spelarnas pjäser. Endast är spelledaren kan skapa och interagera med dem, i övrigt har de samma egenskaper som de andra pjäserna.

Alla spelpjäserna kan plockas upp av alla spelare, även om det inte är spelarens pjäs. Om någon annans pjäs plockas upp lyser pjäsen rött. När spelaren sedan släpper pjäsen förflyttas den tillbaka där den stod innan den plockades upp. När kontrollen är så nära att pjäsen kan plockas upp lyser pjäsen upp. När spelaren plockar upp sin egen pjäs och släpper den över en ruta, förflyttas pjäsen till rutan om den kan gå dit. Annars förflyttas pjäsen tillbaka där den stod.

6.1.4 Startmeny

Det första spelaren kommer till när prototypen startas är startmenyn. Menyn är ett kvadratisk rum där de fyra olika pjäserna som spelarna kan välja mellan står. I menyn väljer spelaren vilken pjäs som denne vill använda i spelet. Upp till en person kan även välja att vara spelledare istället för att välja en pjäs. På väggen till höger finns en knapp för att välja att vara spelledare. Till vänster finns en knapp som startar spelet, om alla spelare har gjort sitt val.

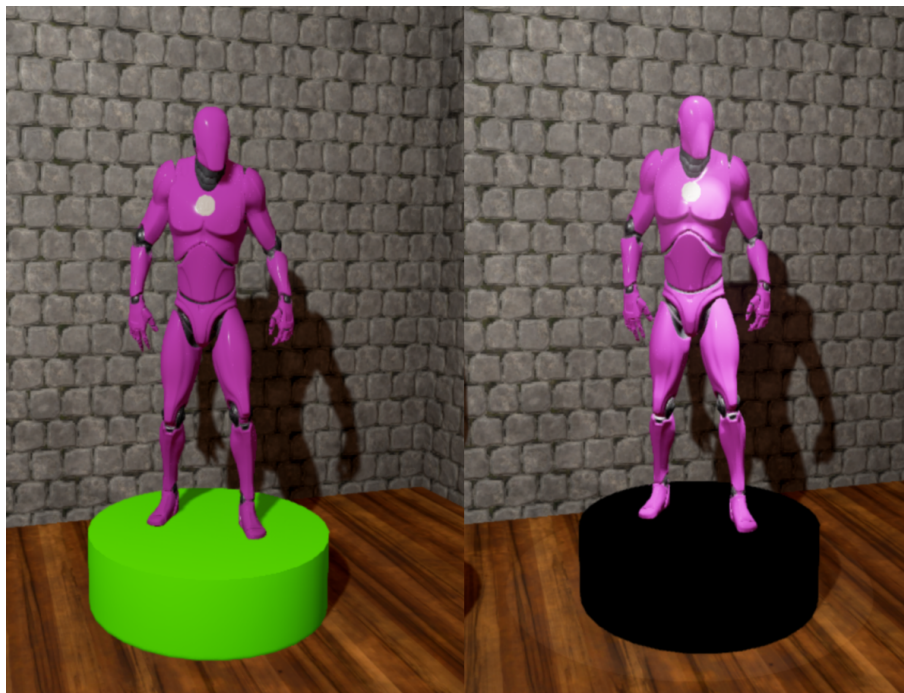
I menyn står pjäserna på olika plattformar. Dessa är från början gröna. När användaren väljer en pjäs blir plattformen under pjäsen svart och det lyser på pjäsen för att förtydliga vilken som är vald.

Spelledarknappen har tre olika lägen beroende på vad spelaren kan göra med den. En grå ruta kan väljas för att bli spelledare. En bock i rutan innebär att spelaren har valt att vara spelledare och ett kryss i rutan innebär att någon annan har valt att vara spelledare.

Startknappen ser nästan likadan ut som spelledarknappen och har två lägen istället för tre, en grön bock och ett rött kryss. Bocken visas om alla spelare har gjort sina val och spelet kan startas, annars visas krysset.



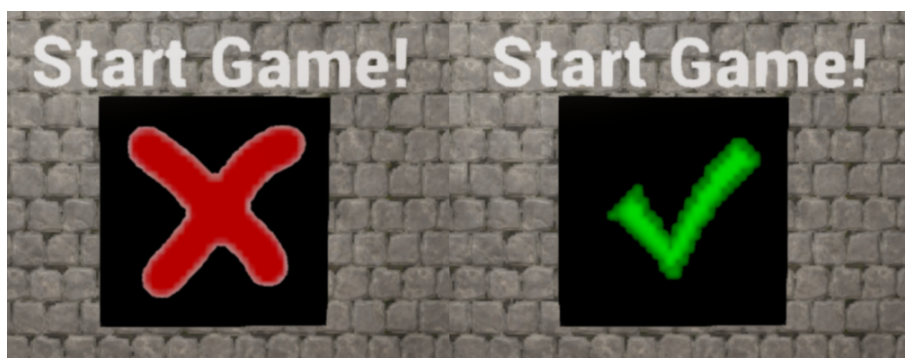
Figur 6.2: Bild på startmenyn.



Figur 6.3: Figuren visar hur plattformen som pjäsen står på betar sig. Till vänster visas det för en pjäs som inte är vald, till höger för en vald pjäs.



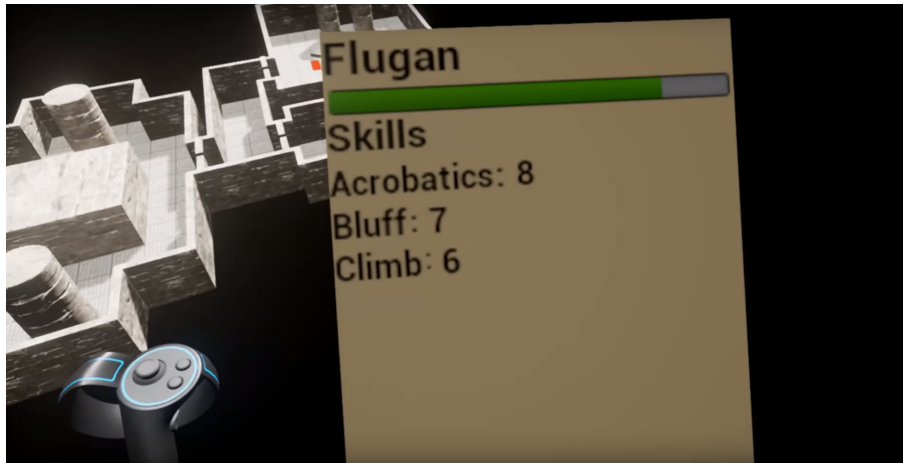
Figur 6.4: Spelledarknappens tre olika lägen i startmenyn. Till vänster visas hur knappen ser ut om någon annan är spelledare, i mitten när ingen är spelledare, till höger när du är spelledare.



Figur 6.5: Startknappens två olika lägen i startmenyn, till vänster när spelet inte går och starta och till höger när spelet går att starta.

6.1.5 Karaktärsblad

Spelaren kan få fram sitt karaktärsblad för sin pjäs genom att klicka på en knapp på VR-kontrollen. Då visas istället för kontrollen ett papper, där bland annat namn och hälsa står. Spelaren kan flytta på detta på samma sätt som VR-kontrollen flyttas på. När spelaren är färdig kan denne trycka på samma knapp för att ta bort karaktärsbladet.

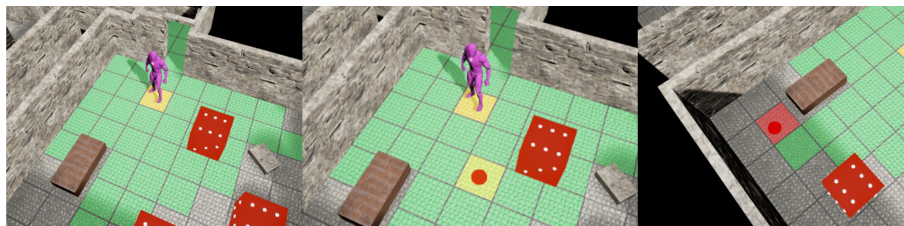


Figur 6.6: Ett karaktärsblad i prototypen.

6.1.6 Rutorna på spelplanen

Spelplanens rutor har en del viktiga funktioner i spelet. De byter bland annat färg beroende på vad spelaren kan göra med dem för att ge feedback till användaren. Den vanliga färgen på en ruta är grå. Runt sin pjäs kan spelaren ibland se gröna rutor, vilket indikerar att pjäsen kan förflyttas till en grön ruta genom att klicka på den. Rutan som spelarens pjäs står på visas som gul för att skilja på dem. Om spelaren pekar på en ruta med lasern byter den färg till ett ljusare material, exempelvis en grön ruta blir ljusgrön, för att ge användaren information om vilken ruta som pekas på. Om spelaren istället pekar på en grå ruta, en ruta som denne inte kan gå till för tillfället, blir rutan röd för att visa att det inte går att interagera med den.

För att räkna ut vilka rutor en pjäs kan flyttas till används en algoritm som börjar med rutan som spelaren står på och kollar vilka grannar den har. Sedan gör den samma sak för dessa rutor tills pjäsens fartantal är uppnått. Rutorna ingår in ett grafsystem där varje ruta har en referens till dess närmaste grannar för att göra detta möjligt. Rutorna känner av vilka rutor som finns i närheten när spelet börjar och sparar informationen under spelet.



Figur 6.7: Rutnät som visar vart pjäsen kan flyttas och var den inte kan flyttas.

6.1.7 Dörrar

Den spelplan som finns i prototypen innehåller tre stycken dörrar. Dessa dörrar blockerar rutan som den står på. Pjäser kan inte gå från ena sidan av en dörr till andra sidan om dörren är stängd.

Alla spelare kan öppna och stänga en dörr om spelarens pjäs står precis bredvid den, men spelledaren kan även välja att låsa en dörr. En låst dörr kan inte öppnas och kan bara låsas upp av spelledaren själv.

Om en spelare pekar på en dörr lyser dörren upp om den kan interageras med. Dörren ger även feedback när en spelare öppnar eller stänger den, genom rörelse och ett gnisslande dörrljud.

6.1.8 Interaktionsmeny

Eftersom det går att göra mer än en sak med vissa objekt på spelplanen så finns en meny för att välja interaktion. För att få fram den pekar spelaren på ett objekt med lasern samt håller inne knappen för interaktion. Det dyker då upp små vita bubblor med olika ikoner som motsvarar de interaktioner som kan utföras med det aktuella objektet. För att välja en interaktion förflyttar spelaren lasern till önskad interaktion och släpper sedan interaktionsknappen. Interaktionsknappen måste alltså hållas inne från det att spelaren tar fram menyn tills en interaktion valts.

6.1.9 Fog of war

När spelaren startar spelet är spelplanen täckt av mörker. Detta är fog of war, vilken döljer spelvärlden från spelaren. Spelaren kan bara se den del av spelvärlden som spelarens spelpjäs har uppsikt över och som därav har blivit avslöjat. När spelaren förflyttar sin pjäs avslöjas de tidigare dolda delar av spelvärlden som pjäsen får uppsikt över och de delar där pjäsen förlorar sin uppsikt över döljs. De områden som döljs täcks av ett skugglikt material som fortfarande avslöjar utseendet av spelplanen som den täcker, men samtidigt indikerar för spelaren att dess spelpjäs har förlorat sin förmåga att se detta område. Dessa skugglika områden representerar förmågan hos spelpjäsen att minnas de delar av spelplanen den sett. På grund av detta kan spelaren inte se andra spelpjäser när dessa rör sig i ett sådant område.



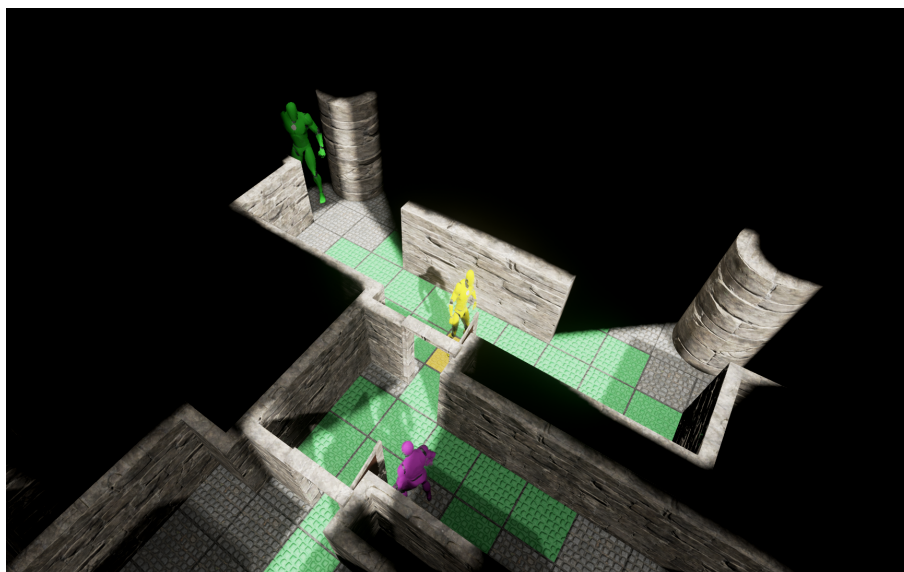
Figur 6.8: En av tre dörrar i prototypen.



Figur 6.9: Interaktionsmenyn för en dörr i prototypen.

Trots att spelvärlden döljs av fog of war så exkluderas spelarnas representation från detta, på så sätt separeras spelarna från spelplanen genom att de själva aldrig döljs,

vilket även gäller vissa andra spelföremål såsom tärningarna.



Figur 6.10: Exempel på hur det kan se ut med fog of war i prototypen. I denna figuren så är användaren den gula pjäsen.

6.1.10 Möjlighet att attackera andra spelare och monster

Spelarna kan attackera andra pjäser under spelet. För att göra detta så måste den egna pjäsen stå på en ruta som är direkt granne med rutan som den attackerade pjäsen står på. Spelaren klickar sedan på pjäsen som skall attackeras. För att förtydliga att en attack är gjord ges feedback i form av ljud. Denna funktionalitet blev inte färdigimplementerad, så vid en attack sker endast en textutskrift. Detsamma sker om den som blir attackerad dör till följd av attacken så skrivs detta också bara ut och ingenting händer med spelpjäsen i spelet.

6.1.11 Turbasering

Det finns ett turbaseringsystem som fungerar, men inte används. När det används försvinner möjligheten utföra olika interaktioner för alla utom två spelare. En av dessa spelare är alltid spelledaren, om sådan finns. Spelledaren kan även ändra vems tur det är. Den spelaren som spelledaren väljer kan interagera med världen igen, medan den förra spelaren inte kan det längre. Spelledaren kan inte inaktivera sin egen tur.

6.1.12 Möjlighet att slå tärning

På prototypens spelplan finns flera tärningar utplacerade i ett av rummen. Tärningarna är röda med vita prickar som motsvarar siffran (se Figur 6.11). Dessa

kan plockas upp och kastas precis som i verkligheten. Värdena som genereras när tärningen slås sparas i spelet, men används inte till något. För att använda sig av tärningens värde får det hanteras manuellt av spelarna. Tärningens sidor har en slags avkännare som känner av vilken sida den står på. Sidorna känner endast av om den slår emot rutorna. Ljudeffekter spelas även varje gång tärningen slår i ett annat objekt, för att ge en mer realistisk upplevelse till användaren.



Figur 6.11: Tärning i prototypen som visar siffran 6

6.1.13 Flerspelarläge

Det system som har implementerats för flerspelarläge är en fungerande kommunikation från klientapplikationen via en server. Servern lyssnar på port 7777 och tillåter alla uppkopplingar, samt tilldelar ett unikt ID för varje användare. Varje spelare kommunicerar endast med servern som i sin tur svarar tillbaka med olika svar beroende på vad den efterfrågas. När klienten är ansluten till servern så ser den alla uppkopplade klienter och all data replikeras till alla klienter i realtid.

Just nu klarar servern av åtminstone två spelare samtidigt, men eftersom inga ordentliga tester gjordes så är det osäkert hur många spelare servern klarar av och vilken hårdvara som krävs.

6.1.14 Verktyg för spelledaren

I spelet finns möjlighet för en av spelarna att vara spelledare. Den spelare som väljer att vara spelledare har extra funktioner tillgängliga under spelets gång. Spelledaren har ingen egen pjäs utan kan istället styra alla andra pjäser på spelplanen. Utöver det så är spelledaren den enda som kan skapa och styra monstren i spelet samt låsa och låsa upp dörrarna i spelet.

Spelledaren har även tillgång till en meny där tanken var att placera ut nya föremål i spelvärlden. I dagsläget kan dock endast ytterligare monster skapas, men inte placeras ut.

6.2 Koncept

Prototypen är en implementation av det koncept av programvaran som utvecklades under projektets planeringsfas. Många av avgränsningarna i prototypen var även avgränsningar i konceptet. Följande är olika avgränsningar som gjordes i prototypen på grund av tidsbrist men är del av konceptet.

Stöd för röstkommunikation är en del av konceptet som valdes bort i projektet på grund av begränsad tid och för att det finns alternativa lösningar för röstkommunikation. Stödet skulle innebära att varje spelares röst kommer från deras position i spelvärlden och blandas in med övriga ljud effekter och eventuell musik. Funktionen kan ytterligare utvecklas så att spelledaren kan dela upp spelare i kanaler som enbart kan kommunicera med varandra eller att ett par spelare på egen hand kan välja att prata enskilt på en kanal. Det är även tänkbart att spelarna skulle kunna ha verktyg för att förvränga sin röst.

I prototypen finns möjligheten att plocka upp och kasta 6-sidiga tärningar utplacerade på spelplanen. I konceptet fanns även andra tärningar, såsom 8-, 10- och 20-sidiga tärningar. Tärningarna är då inte utplacerade på spelplanen utan tas fram av spelarna vid behov. För att kunna kasta tärningarna på ett jämt underlag oavsett var spelaren befinner sig i världen tänktes även möjligheten att ta fram en tärningsbricka för att slå på.

Ytterligare en del av konceptet som inte kom med i projektet är ett läge med förstapersonsperspektiv. I första person följer pjäsen spelarens rörelser med hjälp av VR-utrustningen och spelarens kamera är baserad på pjäsens position och orientering. Detta skulle kunna användas för till exempel ett gestbaserat stridssystem eller för att interagera direkt med virtuella föremål. Förstapersonsläge kan även användas för att lösa pussel eller samla information som inte syns i överblicksperspektivet. Det kan även bli mer element av lajv i spelet, med mer fysiska delar än traditionella bordsrollspel.

6.3 Hur VR kan berika rollspelsupplevelsen

I detta avsnitt tas det upp ett antal olika sätt som VR skulle kunna förbättra rollspelandet. Eftersom projektet inte har haft tid att lägga fokus på användartester och därmed inte kan demonstrera att dessa sätt faktiskt förbättrar rollspelandet bör dessa ses som förslag på områden att studera vidare.

6.3.1 Känsla av närvaro trots avstånd

Eftersom syftet med VR är att ge en känsla av att befinna sig på en annan plats än användarens fysiska plats borde VR även kunna användas för att ge en känsla av att vara på samma plats som en annan person utan att befinna sig på samma fysiska plats, genom till exempel de visuella representationer av andra spelare som finns i prototypen.

VR-utrustning är dock av nödvändighet avskärmande från omvärlden, vilket istället kan leda till en minskad känsla av närvaro om den används tillsammans på samma fysiska plats.

6.3.2 Mer visuellt stöd för inlevelse och koherens

Eftersom VR bygger mycket på det visuella och därför behöver återskapa en virtuell miljö för användaren finns större möjligheter att ge en gemensam och detaljerad representation av spelvärlden.

Det är framförallt intressant ur synpunkten att det kan förhindra missförstånd om spelvärlden genom att ge spelarna möjligheten att själva utforska världen utan att behöva be spelledaren om mer detaljerad beskrivning.

6.3.3 Individuell representation av spelvärlden

Eftersom spelvärlden måste renderas lokalt för varje spelare och deras utrustning finns möjligheten att visa olika saker för olika spelare eller samma saker på olika sätt. Fog of war i vår prototyp är ett exempel på detta.

Det skulle kunna öka kommunikationen mellan spelare eftersom ett begränsat antal spelare får reda på informationen, istället för att alla hör samma information från spelledaren. Dessutom ger det möjligheten att få information utan att andra spelare vet om det, vilket är svårt när alla sitter i ett rum tillsammans.

6.3.4 Byta mellan olika perspektiv

Eftersom allting i VR är virtuellt tar det ingen tid att byta ut element eller ändra hur världen ser ut. Det gäller även hur spelaren upplever världen. Till exempel kan perspektivet bytas från ett överblicksperspektiv till förstapersonsperspektiv och samtidigt kan hur spelaren interagerar med världen ändras.

I fysiska rollspel är spelarna som regel begränsade till det perspektiv de valt att spela. I klassiskt bordsrollspel med miniatyrer har spelarna ett överblicksperspektiv medan spelarna i levande rollspel (lajv) spelar utifrån ett förstapersonsperspektiv. De olika typerna av rollspel har sina fördelar och nackdelar och VR skulle kunna utnyttja

6. Resultat

fördelarna och undvika nackdelarna genom att tillåta att byta mellan perspektiven under spelet.

7

Diskussion

I detta kapitel diskuteras utförandet av projektet. Här belyses fördelar och nackdelar med arbetssättet och slutresultatet. Mot slutet diskuteras även vad som kan vara bra att tänka på i framtida projekt.

7.1 Metodik

Valet att använda en spelmotor för att utveckla prototypen var vitalt för att kunna utföra projektet. Mycket komplicerade problem såsom rendering av grafik och integration med hårdvara behövdes inte tas i åtanke. Vi har helt kunnat bortse från eventuella problem med att använda VR-utrustning, då Unreal Engine löst detta åt oss. Det är oklart vilken påverkan valet att använda Unreal Engine istället för andra spelmotorer har varit.

Valet att använda versionshantering var bra, och Git var ett bra alternativ. Dock uppstod problem med Blueprints när samma fil har haft olika ändringar. Det finns inbyggt stöd för Git i Unreal Engine, vilket eventuellt hade kunnat hantera det bättre och underlätta under projektet.

I projektet gjordes valet att inte använda ett specifikt arbetssätt såsom *agile*, utan istället tilldela uppgifter på möten samt att använda Trello för att hålla ordning på uppgifter och planera efter hand. Detta var inte ett optimalt arbetssätt och att använda en etablerad metod hade antagligen varit bättre.

7.2 Genomförande

Valet att börja med att utforska Unreal Engine individuellt var troligtvis ett bra val. Det tillät varje gruppmedlem att utforska sina intresseområden samt få en känsla av utvecklingsmiljön i stort. Det finns dock olika saker som vi tror skulle kunna förbättra det här arbetssättet, exempelvis regelbundna möten där gruppmedlemmar tar upp intressanta saker de upptäckt eller problem de stötte på. På det sättet kan gruppen förhoppningsvis få en bättre gemensam förståelse för utvecklingsmiljön redan i detta stadiet.

I början av projektet så satsade vi på att snabbt få ihop en grundprototyp, för att sedan kunna lägga till funktioner allt eftersom. Tanken var att grundprototypen skulle innehålla de mest grundläggande funktionerna som exempelvis rutor, en pjäs att gå med och möjlighet att förflytta sig i VR-världen. Vi delade upp dessa grundläggande uppgifter mellan oss i projektgruppen så att varje person fick varsin del. Detta var lyckat, då det möjliggjorde för varje person att fördjupa sig inom sitt område. När grundprototypen blev klar kunde alla i gruppen börja lägga till mindre funktioner och jobba vidare på egen hand utan att behöva vänta på andras arbete, vilket var vad vi på förhand önskade.

Användartestning av prototypen är något vi hade velat göra i större utsträckning, men inte hann med. Det hade varit fördelaktigt att genomföra fler tester under projektets gång, för att få feedback från personer utanför gruppen om vad som de tyckte var prototypens styrkor och svagheter. Detta hade kunnat göra att vi hade behövt prioritera om vilka funktioner som skulle implementeras. Detta är visserligen en prototyp med syfte att utforska VR, men för att lättare kunna komma fram till vilka funktioner som har mer eller mindre potential hade tester underlättat.

Vi var inte helt säkra på hur vi ville att turbaseringen skulle fungera i början. Som vi tidigare nämnt i kapitel 5.3.6 hade vi flera olika idéer på hur det skulle implementeras. Idén om handlingar lät initialt bra, men senare kände vi att vi ville begränsa spelarnas möjligheter i prototypen så lite som möjligt. För att systemet med handlingar ska utöka möjligheterna istället för att minska dem hade vi fått lägga till mycket mer funktionalitet än vad vi trodde oss ha tid för. Till exempel hade en meny för att välja om turbaseringen ska vara på eller inte kunnat ta bort den begränsande aspekten. Vid denna tidpunkten hade vi ingen meny alls och vi kände att vi hade annat som vi ville prioritera högre. Vi tyckte också att prototypen hade behövt tydligt visa för användaren vems tur det var vilket hade krävt ännu mer tid. Därför valde vi att inte ha med någon turbasering alls.

7.3 Resultat

Det är svårt att bli helt nöjd med en prototyp som aldrig riktigt blir klar, inte minst vid viss tidspress. Men vi anser ändå att det finns delar i prototypen som vi har lyckats med ändå. Prototypen har gett oss många intressanta resultat och har även hjälpt oss att försöka svara på vår frågeställning. Detta diskuteras mer utförligt i delkapitlen.

7.3.1 Prototyp

Många av de funktioner som vi ville få med i prototypen lyckades vi implementera delvis eller helt och hållet. Vi är nöjda med att ha inkluderat såpass många saker i prototypen, men det är synd att allt inte blev fullständigt implementerat. Till exempel så finns det en turbaseringsfunktion klar i applikationen, men vi lyckades

inte lägga in den i prototypen på ett sätt som gjorde oss nöjda. I efterhand kan vi konstatera att vi borde fokuserat mer på att hitta sätt att få med alla utvecklade funktioner i slutprototypen.

Prototypen är långt ifrån perfekt och innehåller en hel del buggar. För en applikation med syfte att släppas till allmänheten hade mer tid behövt läggas på att bygga ett stabilt system, men eftersom syftet med vår prototyp är att utforska olika möjligheter i VR så har vi fokuserat på att implementera många olika idéer istället för att skapa ett felfritt program.

Vi är mycket nöjda med fog of war i prototypen. Många av de funktioner som är implementerade i vår prototyp är tillagda för att efterlikna vanliga funktioner i redan existerande analoga roll- och brädspel. Fog of war däremot är något som inte går att ha på samma sätt i analoga spel och som vi tyckte gav extra värde till upplevelsen när det implementerades i prototypen.

7.3.2 Koncept

Projektets koncept uppkom i planeringsfasen, men har byggts ut i samband med utvecklingen av prototypen och visar på vad vi hade velat implementera om vi hade haft tillräckligt med tid. Mycket av projektet har lagts ner på att få prototypen att fungera. Tyvärr hann det inte mynna ut i användartester fokuserade på de olika idéerna, så konceptet kan istället fungera som en intellektuell utforskning av hur rollspel kan på ett givande sätt implementeras i VR.

Eftersom vårt verktyg inte ska på egen hand berätta en historia utan vara ett verktyg för spelarna att berätta sin egen historia ansåg vi att röstkommunikation är central för upplevelsen och inte får begränsas av verktyget. Vi valde snabbt bort att implementera denna funktionaliteten, eftersom det finns många alternativa lösningar för att ordna röstkommunikation. Det är ändå en funktion som borde undersökas, då en ordentlig implementation med ljudkällor i VR-miljön och anpassad funktionalitet har potential att både öka känslan av närvaro men också skapa nya möjligheter i rollspelandet eller förenkla vissa interaktioner såsom att dela ut hemlig information.

Valet att ha med möjligheten att slå tärningar i konceptet gjordes för att vi kände att det fysiska i att slå en tärning i sig självt var del av rollspelandet. Många av de upplevelser som nu finns för VR innehåller moment där spelaren plockar upp föremål på liknande sätt som i verkligheten, något som förstärker känslan av verklighet i upplevelsen.

Förstapersonsperspektiv är inte unikt för VR, utan är också vanligt i många digitala spel. Det hänger på typen av interaktion som det möjliggör för att göra det unikt. Därför valde vi att prioritera ned detta, men vi tror ändå att det har stora möjligheter att tillföra något till upplevelsen. Eftersom spelaren ser spelvärlden från sin pjäs perspektiv skulle det kunna ge en starkare inlevelse och koppling till sin rollfigur. Detta kan liknas med lajv och skulle då även kunna återskapa element från lajv för

spelarna.

7.3.3 Hur VR kan berika rollspelsupplevelsen

Projektets syfte var att undersöka om VR kan förbättra rollspelandet. Men i projektet har vi inte haft tid att utveckla prototypen till den grad att den kan användas för att testa många av våra olika idéer om olika sätt som VR skulle kunna förbättra upplevelsen och därför inte heller kunnat utföra användartester fokuserade på detta. Till följd av detta baseras största delen av denna diskussion inte på våra upptäckter, vi diskuterar istället vad som troligtvis kan förbättra upplevelsen och som är intressant att utforska vidare.

Eftersom känslan att befinna sig på en plats, och även att befinna sig på samma plats som andra, redan har legat i fokus för mycket utveckling med VR så anser vi att det är både ett viktigt område att utforska och något som har nytta utanför rollspelande.

Några av de största sakerna vi tror att digitalisering av rollspel och brädspel ger är möjligheten till mer detaljerade och varierade visuella stöd och möjligheten att låta en dator ta hand om beräkningar, komma ihåg detaljer samt att fördela information till olika spelare. Att kunna utforska världen utan att direkt kommunicera med spelledaren kan möjliggöra helt andra spelmöjligheter där spelarna på ett helt eget sätt kan utforska världen och där mindre tid måste läggas på att beskriva omvärlden. En annan möjlig fördel är att spelarna inte kan missförstå en beskrivning av omvärlden och på så sätt göra misstag som kan förstöra spelupplevelsen för spelaren.

Många av dagens digitala rollspel har förstapersonsperspektiv och påminner mer om levande rollspel (lajv) än klassiska bordsrollspel. På grund av dessa spels popularitet kan de förväntas driva utveckling av industrin åt bättre teknik och kunskap runt den upplevelsen. Genom att utforska möjligheten att integrera förstapersonsperspektiv för att förbättra rollspelande med VR kan möjligheten finnas att integrera kunskap inom datorspelens övergång till VR med rollspelande.

7.4 Pålitlighet

Då projektet aldrig lämnade det stadie där nya funktioner fortfarande implementerades innebär det att vi aldrig kom till det stadiet där vi kunde fokusera på att reducera antalet buggar och fel i systemet. Eftersom varje ny funktion oftast introducerade nya buggar och vi inte prioriterade att fixa existerande buggar har det lett till att projektet i nuläget innehåller en hel del buggar.

Många av de funktioner som implementerades i projektet anpassades för att fungera tillsammans och skulle kräva viss modifiering för att fungera på egen hand. Något som däremot skulle kunna återanvändas för andra projekt utan någon större modifiering är spelarens representation i VR, ifall samma typ av förflyttning och

beteenden är intressant. Även systemet för rutnät är någorlunda lätt att separera. De flesta andra funktioner skulle kräva större modifieringar för att återanvändas i ett annat projekt individuellt. Däremot i ett projekt som använder sig av något eller båda av de nämnda systemen skulle individuella funktioner från prototypen lätt kunna läggas till i projektet, exempelvis tärningar. Projektet i sin helhet skulle relativt enkelt kunna omformas för fokus på andra brädspel, framförallt de spel som använder sig av någon form av rutnät.

Prototypen är utvecklad för VR-utrustning, specifikt en HMD med handhållna handkontroller. Men under utvecklingen behövdes möjligheten att testa utan VR-utrustning så därför finns även styrning med mus och tangentbord. Detta stöd är dock begränsat och saknar funktionalitet, men det är fullt tänkbart att utveckla stödet för flera funktioner, dock förloras de aspekter som VR tillför upplevelsen såsom möjligheten att slå en tärning på ett naturligt sätt. På liknande sätt kan kontrollerna bytas ut mot andra typer av inmatningar där mängden anpassningar som krävs kan variera, detta eftersom mycket av prototypen inte är beroende av hur den integreras med och kan anpassas efter inmatningstyp.

Det är däremot svårare att överföra prototypen och framför allt konceptet till exempelvis *mixed reality* där element ur den fysiska världen används i VR. Det huvudsakliga problemet med detta är att spelarna sannolikt inte befinner sig på samma plats, och därför inte har tillgång till samma fysiska miljö och föremål. Desutom är tanken i konceptet att spelarna ska kunna byta virtuell miljö under spelets gång, vilket kan kompliceras i mixed reality. Det är därför troligt att mixed reality introducerar fler problem än möjligheter.

Under projektet har många små och större tekniska beslut tagits. Till stor del har dessa tagits baserat på tidigare personliga erfarenheter och utan att vara rigoröst underbyggda. Därför kan det vara bra att undersöka alternativ till de val och metoder som valts i projektet. Eftersom fokus legat på att utforska frågeställningen har tekniska valen inte klassats som vitala.

På liknande sätt ska våra förslag till svar på frågeställningen inte ses som ordentliga slutsatser. Förutom den låga erfarenhetsnivån på deltagarna i projektet inom detta ämne så har inte någon ordentlig studie utförts. Det huvudsakliga och viktigaste för att kunna fastställa ett fullständigt eller delvis svar på frågeställningen är att utföra användartester. Den användartestning som utförts i projektet saknar den omfattning och fokusområde som behövs för en sådan studie. Istället bör resultatet ses som ett förslag till möjliga svar som motiveras utifrån erfarenheter i detta projekt.

7.5 Etiska reflektioner

Det här projektet handlar huvudsakligen om hur upplevelsen av att spela ett traditionellt analogt rollspel skulle kunna ersättas med användandet av VR-utrustning. Det skulle innebära att spelarna inte kan se varandra men också att samma upplevelse kan fås utan att befinna sig på samma plats. Eftersom den aktuella VR-

utrustningen inte är lätt att transportera och att kostnaden för utrustningen är signifikant kommer troligen de som spelar att befinna sig på olika platser.

Risken finns därför att en sådan här produkt skulle få personer att istället för att mötas på samma plats möts via produkten från olika platser. Det skulle potentiellt minska personernas interaktion med andra människor vilket är etiskt problematiskt. Å andra sidan kan produkten ha potential att tillåta personer som av olika skäl inte kan träffas på samma plats att få upplevelsen av rollspel tillsammans de annars inte kunde få. Så denna möjlighet kan ha potential att öka interaktionen med andra människor för vissa individer.

Kostnaden av utrustningen ger också upphov till viss etisk problematik. Om en sådan produkt kan ge en bättre upplevelse än den traditionella rollspelsupplevelsen finns en risk att personer blir uteslutna från rollspelandet på grund av att de inte har tillgång till VR-utrustning. Och en individ kan känna sig pressad till att köpa VR-utrustning för att spela trots att det inte passar in i budgeten.

7.6 Framtida arbete

På grund av projektets relativt korta genomförande och olika svårigheter i projektet har många av gruppens idéer inte implementeras och många områden inte utforskats. En idé handlade om att utöka spelledarens roll med möjlighet att i större utsträckning ändra saker i spelet, exempelvis möjlighet att bygga fällor, placera ut föremål eller ändra spelplanens utformning. Detta kan vidare utvecklas till ett redskap för att bygga upp spelplanen från grunden inne i spelet.

Ett annat spår är att ge spelarna möjligheten styra sin spelpjäs i förstapersonsvy. Det skulle kunna användas för att lösa pussel eller utföra mer fysiska moment. Eftersom VR-upplevelser i förstapersonsvy redan nu har utforskats i stor utsträckning är förmodligen blandningen av spelare med förstapersonsvy och överblicksvy ett mer intressant område att utforska.

Eftersom vi i projektet inte hade tid att göra utförliga tester så är det något som är intressant att göra i en framtida studie. Att utföra tester om på vilka sätt VR faktiskt förbättrar rollspelsupplevelsen skulle bidra med kunskap som kan förbättra utvecklingen och styra forskningen i framtiden.

8

Slutsats

I projektet utforskades frågeställningen “Hur kan VR berika rollspelsupplevelsen?” genom diskussion och utvecklandet av en prototyp.

Det tycks finnas sätt att använda VR för att förbättra upplevelsen och tillföra nya element till traditionella rollspel. Följande sätt har identifierats i projektet:

- **Känsla av närvaro trots avstånd**
Om spelarna är långt ifrån varandra kan de ändå få en upplevelse där de kan se varandra. Att sitta tillsammans är dock att föredra i detta fallet, då ansiktsuttryck är svårt att få med i VR i dagsläget.
- **Mer visuellt stöd för inlevelse och koherens**
Teknikens styrka i att förmedla visuell information kan hjälpa att ge en gemensam och fullständig bild av spelvärlden.
- **Individuell representation av spelvärlden**
Att enkelt kunna ge spelare olika representation av världen ger nya spelmöjligheter där spelarna kan upptäcka saker på egen hand.
- **Byta mellan olika perspektiv**
Att kunna växla mellan ett överblicksperspektiv och ett förstapersonsperspektiv ger en mer varierad upplevelse.

Det kan självklart finnas fler sätt som VR kan förbättra rollspel. Det här projektet har inte visat på att just sätten ovan faktiskt förbättrar upplevelsen men de är intressanta att utforska vidare.

Arbetet med prototypen i projektet har varit lärorikt som bas för att vidare utforska området VR och rollspel men utvecklingen nådde inte stadiet då den kunde användas för att visa på ovan nämnda möjligheter med VR.

Framtida arbeten kan fördelaktigen fokusera på att ta fram en prototyp som försöker visa på en av punkterna ovan. I ett sådant projekt bör mycket fokus läggas på användartester för att kunna göra en bedömning om det faktiskt förbättrar upplevelsen.

Litteraturförteckning

- [1] M. Bieg, “Client server model.” <https://en.wikipedia.org/wiki/Peer-to-peer#/media/File:Server-based-network.svg>, 2007. hämtat 2017-05-26.
- [2] M. Bieg, “Peer to peer network.” <https://en.wikipedia.org/wiki/Peer-to-peer#/media/File:P2P-network.svg>, 2007. hämtat 2017-05-26.
- [3] I. E. Sutherland, “The Ultimate Display,” *In proceedings of IFIPS Congress (New York City, NY, May 1965)*, vol. vol.2, pp. 506–508, 1965.
- [4] P. Bouvier, P. Chaudeyrac, and V. Biri, “Cross benefits between virtual reality and games,” in *International Conference on Computer Games, Multimedia and Allied Technology (CGAT’08)*, (France), p. 9, 2008.
- [5] D. Clarke, G. McGregor, B. Rubin, J. Stanford, and T. N. Graham, “Arcaid: Addressing situation awareness and simulator sickness in a virtual reality pac-man game,” in *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, CHI PLAY Companion ’16*, (New York, NY, USA), pp. 39–45, ACM, 2016.
- [6] J. Jerald, *The VR Book: Human-Centered Design for Virtual Reality*. New York, NY, USA: Association for Computing Machinery and Morgan & Claypool, 2016.
- [7] L. Rebenitsch and C. Owen, “Individual variation in susceptibility to cybersickness,” in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology, UIST ’14*, (New York, NY, USA), pp. 309–317, ACM, 2014.
- [8] Epic Games, Inc, “Virtual Reality Best Practices.” <https://docs.unrealengine.com/latest/INT/Platforms/VR/ContentSetup/index.html#vrandsimulationsickness>. hämtat 2017-03-02.
- [9] A. F. Seay, D. M. Krum, L. Hodges, and W. Ribarsky, “Simulator sickness and presence in a high field-of-view virtual environment,” in *CHI ’02 Extended Abstracts on Human Factors in Computing Systems, CHI EA ’02*, (New York, NY, USA), pp. 784–785, ACM, 2002.
- [10] A. S. Fernandes and S. K. Feiner, “Combating vr sickness through subtle dynamic field-of-view modification.” 10.1109/3DUI.2016.7460053, mar 2016.

- [11] S. Davis, K. Nesbitt, and E. Nalivaiko, “A systematic review of cybersickness,” in *Proceedings of the 2014 Conference on Interactive Entertainment, IE2014*, (New York, NY, USA), pp. 8:1–8:9, ACM, 2014.
- [12] C. Nutt, “The biggest challenges for vr game developers, straight from oculus.” http://www.gamasutra.com/view/news/198603/The_biggest_challenges_for_VR_game_developers_straight_from_Oculus.php, 2017. hämtat 2017-05-31.
- [13] D. A. Bowman, E. Kruijff, J. J. LaViola, and I. Poupyrev, *3D User Interfaces: Theory and Practice*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 2004.
- [14] J. E. Innes and D. E. Booher, “Consensus building as role playing and bricolage: Toward a theory of collabo.” <http://www.tandfonline.com/doi/abs/10.1080/01944369908976031>, mar 1999.
- [15] Wizards of the Coast, “D&D Official Homepage.” <http://dnd.wizards.com/>, 2017. hämtat 2017-05-08.
- [16] Chaosium Inc., “Call of Cthulhu.” <http://www.chaosium.com/call-of-cthulhu-rpg/>, 2017. hämtat 2017-05-08.
- [17] “BioWare | Dragon Age: Origins.” <http://dragonage.bioware.com/dao/>. hämtat 2017-05-29.
- [18] Blizzard Entertainment, Inc, “World of Warcraft.” <https://worldofwarcraft.com/>. hämtat 2017-05-29.
- [19] Ladousse Porter Gillian and Alan, Maley, *Role Play*. Oxford, NewYork: Oxford University Press, 1987.
- [20] E. Adams, *Fundamentals of Game Design*. New Riders Publications, 2 ed., 2010.
- [21] H. S. Oluwatosin, “Client-server model,” *IOSR Journal of Computer Engineering (IOSR-JCE)*, vol. 2278-8727, pp. 67–71, 2014.
- [22] G. Zhao, “Research on mobile p2p network architecture,” *Advanced Materials Research*, vol. 433-440, pp. 5250–5255, 2012.
- [23] Epic Games Inc, “Custom Events | Unreal Engine 4.” <https://docs.unrealengine.com/latest/INT/Engine/Blueprints/UserGuide/Events/Custom/>. hämtat 2017-05-25.
- [24] HTC, “Vive.” <https://www.vive.com/eu/>, 2017. hämtat 2017-05-08.
- [25] Oculus, “Oculus Rift.” <https://www.oculus.com/rift/>, 2017. hämtad 2017-05-08.
- [26] Sony Interactive Entertainment, “PlayStation VR.” <https://www.playstation.com/sv-se/explore/playstation-vr/>, 2017. hämtad 2017-05-

- 08.
- [27] S. Shuman, “PlayStation VR: the ultimate FAQ.” <https://blog.eu.playstation.com/2016/10/03/playstation-vr-the-ultimate-faq/>, 2016. hämtat 2017-02-28.
- [28] Google, “Google Cardboard.” <https://vr.google.com/cardboard/>. hämtat 2017-05-12.
- [29] Samsung Electronics CO, “Samsung Gear VR.” <http://www.samsung.com/global/galaxy/gear-vr/>. hämtat 2017-05-12.
- [30] Days of Wonder, “Ticket to Ride.” <https://www.daysofwonder.com/online/en/t2r/>, 2017. hämtat 2017-05-08.
- [31] Days of Wonder, “Small World 2.” <https://www.daysofwonder.com/online/en/smallworld/>, 2017. hämtat 2017-05-08.
- [32] SmiteWorks, “Fantasy Grounds.” <https://www.fantasygrounds.com/home/home.php>, 2017. hämtat 2017-05-08.
- [33] The Orr Group, “Roll20: Online virtual tabletop for pen and paper RPGs and board games.” <https://roll20.net/>, 2017. hämtat 2017-05-12.
- [34] B. Karl, J. Staffan, and B. Staffan, “Undercurrents: A computer-based game-play tool to support tabletop roleplaying,” in *DiGRA Nordic '10: Proceedings of the 2010 International DiGRA Nordic Conference: Experiencing Games: Games, Play, and Players*, 2010.
- [35] R. L. Mandryk and D. S. Maranan, “False prophets: Exploring hybrid board/video games,” in *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '02, (New York, NY, USA), pp. 640–641, ACM, 2002.
- [36] J. P. Zagal, M. Nussbaum, and R. Rosas, “A Model to Support the Design of Multiplayer Games,” *Presence: Teleoperators and Virtual Environments*, vol. 9, pp. 448–462, oct 2000.
- [37] Fantasy Flight Games, “XCOM: The Board Game.” <https://www.fantasyflightgames.com/en/products/xcom-the-board-game/>, 2017. hämtat 2017-05-08.
- [38] Fantasy Flight Games, “Mansions of Madness Second Edition.” <https://www.fantasyflightgames.com/en/products/mansions-of-madness-second-edition/>, 2017. hämtat 2017-05-08.
- [39] P. Johan, E. Daniel, and B. Staffan, “Augmented board games - enhancing board games with electronics,” in *DiGRA 2005 - Proceedings of the 2005 DiGRA International Conference: Changing Views: Worlds in Play*, 2005.
- [40] Berserk Games, “Tabletop Simulator.” <http://berserk-games.com/tabletop-simulator/>, 2017. hämtat 2017-05-08.

- [41] Eriksson Anders, Roos Christian, Olsson Daniel, Malmer Jimmy, Björklund Kevin, and Wecke Richard, “StenkrossStudios.” <http://stenkrossstudios.com/about>, 2016. hämtat 2017-02-23.
- [42] Unity, “Unity - Game Engine.” <https://unity3d.com/>, 2017. hämtat 2017-05-08.
- [43] Epic Games, “Unreal Engine 4.” <https://www.unrealengine.com/what-is-unreal-engine-4>, 2017. hämtat 2017-05-08.
- [44] Epic Games Inc, “Client-Server Model | Unreal Engine 4.” <https://docs.unrealengine.com/latest/INT/Gameplay/Networking/Server/>. hämtat 2017-05-12.
- [45] Epic Games, “Unreal Engine 4 | Replication .” <https://wiki.unrealengine.com/Replication>, 2017. hämtat 2017-05-12.
- [46] “Git.” <https://git-scm.com/>. hämtat 2017-05-12.
- [47] “GitHub.” <https://github.com/>. hämtat 2017-05-12.
- [48] Kraemer, “Thinking About Thinking Aloud: A Comparison of Two Verbal Protocols for Usability Testing,” *IEEE Transactions on Professional Communication*, vol. 47, pp. 105–117, jun 2004.
- [49] A. Nandy, *Introduction to Leap Motion*. Berkeley, CA: Apress, 2016.
- [50] L. Webster, “Visual studio ide, code editor, team services, & mobile center.” <https://www.visualstudio.com>, 2017. hämtat 2017-05-31.
- [51] D. Hutchison, “Video Games and the Pedagogy of Place,” *The Social Studies*, vol. 98, pp. 35–40, jan 2007.