

## **Tarkkuusammuntapelin kehitys virtuaalitodellisuudessa**

Timo Heikkilä

Haaga-Helia ammattikorkeakoulu

Amk-opinnäytetyö

2021

Tietojenkäsittelyn koulutusohjelma

## Tiivistelmä

**Tekijä(t)**

Timo Heikkilä

**Tutkinto**

Tietojenkäsittelyn koulutusohjelma

**Raportin/Opinnäytetyön nimi**

Tarkkuusammuntapelin kehitys virtuaalitodellisuudessa

**Sivu- ja liitesivumäärä**

40 + 1

Tämän opinnäytetyön tarkoituksena oli tehdä virtuaalisen todellisuuden videopeli, joka luodaan Unity-pelimootorilla. Pelin avulla selostetaan, kuinka tämän opinnäytetyön VR-peli tehtiin Unitylla ja minkälaisia työkaluja, paketteja ja VR:lle yksinomaisia tekniikoita käytettiin pelin rakennuksessa. Pelin aiheena oli tarkkuusammuntapeli, jonka avulla pelaaja pystyy turvallisesti harjoittelemaan aseiden käsittelyä ja aseiden mekaniikkaa. Itse pelistä ei tule täysin valmista produktia, mutta pienellä jatkokehittämisellä sitä voitaisiin käyttää opetustarkoitukseen tai viihdekäyttöön.

Peli on rakennettu Unity pelimootorilla ja C#-kielellä, sekä työkalujen ja liitännäisten kautta peliä voidaan pelata lähes millä tahansa VR-laitteella. Tämä onnistuu OpenXR-standardilla ja ilmaisilla VR-kehitykseen kuuluvilla työkalupaketeilla. Lopputuloksena saatiin peli, jonka avulla voitiin esitellä VR-kehitystä ja Unityn käyttöä VR-pelien rakentamiseen. Peliä optimoitiin, jotta se toimisi tehokkaasti eri laitteilla ja se tuottaisi mahdollisimman vähän liikepahoinvointia. Lopuksi peli julkaistiin Itch.io-nettisivulle.

Tietoperustassa kerrotaan ensin virtuaalisesta todellisuudesta ja VR-laitteista. Luvussa käydään läpi mitä on virtuaalinen todellisuus, sen historiaa, tämänhetkisiä käyttökohteita ja miten VR-teknologiaa voidaan kehittää eteenpäin. Tämän jälkeen käydään läpi Unityn historiaa ja minkälaista kehittämistä Unitylla on. Samalla kerrotaan mikä pelimootori on ja miksi niitä käytetään.

Peliprojektin läpikäynnin aikana kerrotaan yleisesti pelin kulusta ja tärkeistä VR-pelin rakennusvaiheista, niin että lukija ymmärtää, mitä eri asioita VR-pelin rakentamiseen tarvitaan. Pelissä käytetyt työkalut ja paketit käydään läpi, jotta voidaan esitellä, miten ne asennetaan ja kuinka ne auttavat VR-pelien eri rakennusvaiheissa. Toiminnallisuuden kuvauksessa ei suuremmin käydä läpi työkaluja ja kehitysvaiheita, jotka eivät liity VR-kehitykseen, kuten animoinnin, äänien ja ympäristön tekemistä.

**Asiasanat**

Unity, virtuaalinen todellisuus, pelimootori, videopeli

## Sisällys

1	Johdanto .....	1
2	Virtuaalinen todellisuus .....	5
2.1	Historia .....	5
2.2	Laitteisto ja ohjelmisto.....	9
2.3	Käyttökohteet.....	11
2.4	Kehityskohteet .....	11
3	Unity-pelimoottori .....	14
3.1	Historia .....	14
3.2	Pelink kehitys Unitylla .....	15
4	Peliprojektin tekeminen .....	17
4.1	VR-Projektin aloitus .....	17
4.1.1	OpenXR.....	18
4.1.2	XR Interaction Toolkit.....	20
4.1.3	Asset Store ja muut paketit .....	22
4.2	XR Rig setup.....	23
4.2.1	Käyttöliittymä.....	24
4.2.2	VR:lle erityisiä ratkaisuja .....	26
4.3	Pelin kulku .....	27
4.4	Optimointi.....	29
4.5	Julkaiseminen .....	33
5	Pohdinta.....	34
5.1	Opinnäytetyön ja pelin onnistuminen.....	34
5.2	VR-pelin rakennus Unitylla .....	36
	Lähteet .....	37
	Liitteet.....	41
	Liite 1. Pelin sivu Itch.io-alustalla.....	41

## 1 Johdanto

Virtuaalinen todellisuus on jo suhteellisen vanha idea, mutta viimeisen viiden vuoden aikana se on nähnyt aktiivista käyttöönottoa eri aloilla ja varsinkin videopelien pelaajien kanssa. Pelialalle tarpeeksi hyvät VR-lasit saapuivat vasta vuonna 2016, jonka jälkeen teknologia, käyttäjämäärä ja käyttökohteet ovat laajentuneet nopeaa vauhtia. Virtuaalisen todellisuuden maailmat luovat uusia tapoja ja mahdollisuuksia tehdä jokapäiväisiä asioita, olipa se kotona tai työpaikalla. Virtuaalitodellisuus on erinomainen tapa opettaa käytännön asioita, ja tämän takia sen käyttö kouluttamiseen on tullut jäädäkseen. Tämä on myös suurimmista syistä, miksi yritykset ja tahot ovat alkaneet hyödyntämään VR-teknologiaa eri tarkoituksiin, kuten työntekijöiden kouluttamiseen, tuotteiden suunnitteluun ja esittelyyn. Tämän lisäksi terveydenhuollonala on alkanut tuottamaan potilaiden ongelmiin erilaisia hoitokeinoja, jonka takia uskotaan, että terveydenhuollonala tulee nostavamaan VR-markkinoiden arvoa kaikista eniten seuraavan seitsemän vuoden aikana. Viihdealalla VR-pelit ovat nousseet myös suurta vauhtia tehokkaampien laitteiden myötä ja monien tunnettujen peliyhtiöiden astuessa VR-markkinoille. Samalla VR-kehitys on parantunut ilmaisten pelimoottorien myötä, sillä ne nopeuttavat sovellusten kehitystä ja helpottavat kehitysprosessin suurta työmäärää erilaisien työkalujen ja liitännäisten avulla.

Tämän opinnäytetyön tarkoituksena on tehdä yksinkertainen virtuaalitodellisuuden tarkkuusammuntapeli, jonka avulla pelaaja oppii käyttämään pistoolia ja rynnäkkökivääriä oikein ja turvallisesti. Aseen käyttö on turvallista realistisessa 3D-ympäristössä, jossa käyttäjä pystyy opettelemaan aseiden turvallista käsittelyä, ja tämän avulla liikkeet jäävät hänen lihaskuormitukseensa seuraavaa kertaa varten. Pelillä on kuitenkin tiettyjä rajoja ja kaikkea tarpeellista ei voida opettaa pelin kautta, sillä VR-teknologia ei ole tarpeeksi kehittynyt simuloimaan todellisuutta kunnolla. Pelaaja ei esimerkiksi voi kunnolla oppia rekyylin hallintaa ja kunnan ampuma-asennon ottamista. Itse peliä on kuitenkin mahdollista käyttää rajallisena harjoitteluna tietyillä aloilla tai pohjana VR-pelin jatkokehitykseen. Peli tullaan optimoimaan tarpeeksi hyvin, ja kaikki vuorovaikutukset pyritään tekemään mahdollisimman immersiiivisesti ja interaktiivisesti, jotta voidaan vähentää pahoinvointia uusilla käyttäjillä. Peli kuitenkin pyritään saamaan tuntumaan mahdollisimman todelliselta, jotta pelikokemus olisi miellyttävä. Opinnäytetyön kannalta, pelin avulla esitellään Unity-pelimoottorin VR-pelinrakennuksen tärkeimmät osat, sekä minkälaista VR-pelinkehitys on Unitylla. Luvussa 4 käydään tarkemmin läpi pelin kulku, VR-ominaisuudet ja toiminnallisuus.

Opinnäytetyö on suunnattu pelinkehityksestä tai virtuaalisesta todellisuudesta kiinnostuneille, joten lukijan oletetaan vähintään omaavan jonkinlaista kokemusta peleistä ja mahdollisesti jopa pelinkehityksestä. Lukijan tulisi saada ymmärrys, kuinka VR-pelinkehitys on

mahdollista Unityn työkaluilla, ja pelinkehittäjät voivat jopa verrata, kuinka VR-pelink kehitys eroaa normaalista 3D-pelink kehityksestä.

Opinnäytetyön tietoperusta on rajattu kahteen lukuun, jotka ovat virtuaalinen todellisuus ja Unity-pelimoottori. Virtuaalisen todellisuuden luvussa kerrotaan laajemmin kuluttajamarkkinoista ja laitteista, kun taas yrityskäyttöpuolta kuvaillaan vain, jos se tuo arvoa kyseessä olevaan aiheeseen. Toiminnallisessa osassa käydään ensin läpi, kuinka VR-projekti aloitetaan Unitylla, ja tämän jälkeen käydään aiheittain läpi pelin eri VR-ominaisuudet ja käytetyt työkalut, liitännäiset ja paketit, joiden avulla VR-kehitys on mahdollista. Jokaisesta työkalusta ja paketista kerrotaan, miksi ne ovat tärkeitä ja kuinka ne nopeuttavat projektia. Yksi tärkeimmistä apuvälineistä on OpenXR-liitännäinen, jonka avulla peliä voidaan käyttää kaikilla tunnetuimmilla VR-laitteilla, ja toinen näistä on XR Interaction Toolkit -paketti, josta saadaan yksinkertaiset liikkumis- ja interaktiokomponentit VR-rakennukseen. Unityn kaupasta tuodaan muutama paketti, joiden avulla nopeutetaan rakennusta. Tämän takia toiminnallisessa osassa pääosin keskitytään siihen, miten peli valmistettiin VR-kehityksen osalta, joten muut Unity-pelink kehitykseen kuuluvat osat ja työkalut, kuten animaatiot, äänet ja 3D-mallien tekeminen ovat jätetty pois lähes kokonaan pelin läpikäynnistä.

Valitsin tämän aiheen, koska olen jo vuosia ollut kiinnostunut virtuaalisesta todellisuudesta, sekä suosittujen Oculusin Quest 2 -lasien julkaisu auttoi minua aloittamaan VR-kehityksen. Quest 2 -lasit ovat olleet suuri menestys ympäri maailmaa, ja samalla ne ovat tuoneet suuren määrän uusia käyttäjiä virtuaalitodellisuuteen. Pelink kehitystä Unitylla olen viimeisen vuoden aikana opetellut ensin 2D-pelin kanssa koulun ohjelmistoprojekti 2 -kurssilla, ja VR-kehityksestä aloin kiinnostua viime talvena. Valitsin Unity-pelimoottorin, sillä se on yksi suurimmista ja tunnetuimmista moottoreista, sekä minulla on jo kokemusta siitä. Unity myös sisältää erinomaiset työkalut VR-kehitykseen, sekä Unityn yhteisö on erittäin laaja, joten ongelmien ratkaiseminen on helpompaa, sekä yhteisön tekemien 3D-mallien ja muiden tuotteiden käyttäminen nopeuttaa projektin tekoa. Koska peli rakennetaan Unitylla, niin pelin ohjelmointiin käytetään C#-kieltä, jonka helppous oli myös yksi suurimmista syistä, miksi Unity valittiin pelimoottoriksi. Ennen opinnäytetyötä minulla ei ollut kunnan osaamista kokonaisen pelin julkaisemisesta, jonka takia tahdoin saattaa pelin loppuun saakka. Tämän ansiosta pystyn opettelemaan optimointia ja pelin valmistelua julkaistavaksi.

## **Keskeiset käsitteet**

### **XR**

Laajennetulla todellisuudella tarkoitetaan kaikkia ympäristöjä, jotka ovat luotu tietokoneilla, ja joissa yhdistetään todellisuus ja virtuaalisuus. XR sisältää siis virtuaalisentodellisuuden (VR),

lisätyn todellisuuden (AR) ja yhdistetyn todellisuuden (MR) teknologiat.

<b>VR</b>	Virtuaalisella todellisuudella tarkoitetaan tietokoneella simuloituja 3D-ympäristöjä. Käyttäjä voi olla vuorovaikutuksessa tämän ympäristön kanssa erilaisilla päähän laitettavilla lasilla ja ohjaimilla.
<b>Immersio</b>	Immersio, eli käyttäjän ympäristöön uppoutumisen tunne. Mitä enemmän käyttäjä tuntee todellisesti olevan vuorovaikutuksessa ympäristön kanssa, niin sitä parempi on immersio.
<b>HMD</b>	Head-mounted Display on päähän laitettavat lasit tai laite, jonka avulla käyttäjä pystyy näkemään virtuaalisen maailman.
<b>Haptinen palaute</b>	Haptista palautetta voidaan antaa käyttäjälle laitteiden avulla, jonka tarkoituksena on parantaa immersiota. Tämä on usein ohjaimien tuottamaa värinää, jonka avulla pyritään simuloimaan tuntoaistia.
<b>Liikepahoinvointi</b>	Liikepahoinvointi on pelaajan kokema pahoinvoinnin tunnetta eli motion sicknessiä, joka tapahtuu, kun hänen aistinsa lähettävät tasapainoaistille ristiriitaista tietoa.
<b>PCVR</b>	PCVR-laitteilla tarkoitetaan niitä lasia, jotka pitää yhdistää kaapelilla tietokoneeseen, jotta kyseinen tietokone voi tehdä pelin suorituksen ja renderöinnin lasien näytölle.
<b>Standalone</b>	Standalone headset eli itsenäiset lasit tekevät pelin suorituksen ja renderöinnin lasien oman prosessorin ja osien avulla, jotta ne olisivat langattomat, ja niitä voitaisiin käyttää ilman kallista tietokonetta. Useat itsenäiset lasit voidaan kuitenkin yhdistää tietokoneeseen langattomasti, jonka avulla voidaan käyttää tietokoneen suurempaa tehoa tiettyjen kokemusten suorittamiseksi.
<b>Asset</b>	Assetilla tarkoitetaan kaikkia Unityn sisäisiä tiedostoja, joilla peli rakennetaan, kuten skriptit, äänet, 3D-mallit ja tekstuurit.

<b>Peliobjekti</b>	<p>Peliobjektit ovat objekteja, joilla peli rakennetaan skenessä. Peliobjektit ainoastaan sisältävät objektin koordinaatit, rotaation ja koon pelinäköymässä, mutta niihin voidaan lisätä, esimerkiksi tekstuureita, skriptejä, ääniä ja muita komponentteja.</p>
<b>Skene</b>	<p>Skenellä tarkoitetaan yhtä pelin maailmaa tai tasoa, joiden välillä voidaan liikkua.</p>
<b>IDE</b>	<p>Integroitu ohjelmointiympäristö sisältää vähintään tekstieditorin ja koodin kääntäjän, sekä usein se sisältää debuggerin, versiönhallintatuen ja ohjelmiston profiloinnin.</p>
<b>Renderointi</b>	<p>Renderoinnilla tarkoitetaan kuvan hahmottamista tietokoneen näytölle.</p>
<b>URP</b>	<p>Universal Rendering Pipeline on Unityn valmistama renderointi-putki, joka on helppo käyttää ja kevyempi vaihtoehto muille putkille.</p>

## 2 Virtuaalinen todellisuus

Virtual Reality (VR) tai virtuaalinen todellisuus on simuloitu 3D-ympäristö, joka luodaan tietotekniikan avulla. Käyttäjä sijoitetaan kokemuksen tai simulaation sisään, jonka kanssa hän voi olla vuorovaikutuksessa erilaisten laitteiden avulla. Tämän aikana tietokone simuloi mahdollisen monia käyttäjän aisteja, kuten näköä, kuuloa, kosketusta ja jopa hajua, jotta käyttäjä tuntee olevansa todellisesti simulaation sisällä. Lähes todellisia VR-kokemuksia estää sisällön saatavuus ja tehokkaiden tietokoneiden kallis hinta. (Bardi 2020.)

Lowoodin (2020) mukaan virtuaalinen todellisuus on tietokonemallinnuksen ja simulaation avulla saavutettava keinotekoinen kolmiulotteinen ympäristö, jonka kanssa käyttäjä on vuorovaikutuksessa. VR-sovellukset upottavat käyttäjän luotuun ympäristöön, joka simuloi todellisuutta interaktiivisesti eri laitteiden, kuten lasien, kuulokkeiden, käsineiden ja jopa kokovartaloasujen avulla. Bardi ja Lowood kuvailevat virtuaalista todellisuutta samalla tavoin, ja Lowood vielä jatkaa kuvailua sanomalla, että ympäristön sisällä olemisen tunne saavutetaan liiketunnistimilla, jotka ottavat käyttäjän liikkeen ja siirtää 3D-ympäristön sisällä olevaa näkymää. Tämän avulla käyttäjän aivot tuntevat olevan läsnä virtuaalitodellisuuden ympäristössä, joka auttaa käyttäjää uppoutumaan kyseiseen maailmaan.

Yrityksille virtuaalisen todellisuuden käyttäminen koulutukseen ja simuloimiseen antaa heille suuren määrän sijoitetun pääoman tuottoa. VR:n käyttö säästää suuren määrän rahaa ja aikaa, sekä se parantaa työn turvallisuutta. VR-lasien käyttö taas vähentää simulaatioiden hintaa, sillä enää ei tarvitse ostaa kokonaista koulutustilaa, jonka sisälle usein rakennettaisiin koulutuksessa käytettävät laitteet ja järjestelmät. Laitteiden ylläpito on myös helpompaa, sekä VR-lasit ovat helppo kantaa mukana missä ikinä niitä tarvitaan. (Varjo 2021, 19-20.)

Maailmanlaajuisen virtuaalisen todellisuuden markkina-arvoksi arvioitiin 15,81 miljardia dollaria vuonna 2020, ja sen odotetaan kasvavan 18 %:n vuotuisella kasvuvauhdilla vuodesta 2021 vuoteen 2028. VR-teknologia on jo muuttanut peli- ja viihdeteollisuutta upottamalla käyttäjät virtuaalisiin maailmoihin. Tämän lisäksi VR-teknologian lisääntyminen muilla aloilla, kuten opetuksessa, terveydenhoidossa, työntekijöiden koulutuksessa ja autoteollisuudessa on johtanut VR-markkinoiden kasvuun. (Grand View Research 2021.)

### 2.1 Historia

Ensimmäinen virtuaalisen todellisuuden laite syntyi vuonna 1956. Tämä laite oli Morton Heiligin rakentama iso koppimainen kone nimeltä Sensorama, jonka sisälle mahtui neljä henkilöä kerrallaan. Väriävällä tuolilla, tuoksuntuottajalla, kaiuttimilla ja



stereoskooppisella 3D-näytöllä varustettu koppi pystyi upottamaan käyttäjän Sensoramalle yksinomaisesti kehitettyyn elokuvaan (Brockwell 2016.)

1980-luvulla tapahtui ensimmäinen VR-tekniikan kasvupyrähdys, sillä monet yhtiöt alkoivat kiinnostua omien VR-lasien kehittämisestä. Nasa alkoi valmistamaan omia laitteitaan, ja yhtiö nimeltä VPL alkoi kehittämään ensimmäisiä myyntitarkoitukseen tarkoitettuja VR-laseja ja hanskoja, jotka näkyvät kuvassa 1 (Thrillseeker 24.2.2020, 4–6 min.)



Kuva 1. VPL:n kehittämä HMD ja hanskat vuodelta 1985 (Barnard 2019)

Nasa kehitti virtuaalisen todellisuuden laitteen 1990-luvulla, jonka avulla Mars-robottia pystyttäisiin ohjaamaan maasta, kun taas The Virtuality Group julkaisi laitteen nimeltä Virtuality samoihin aikoihin. Virtuality oli ensimmäinen massatuotettu VR-laite, jota käytettiin pelihalleissa suosittujen pelien pelaamiseksi virtuaaliympäristöissä. 1990-luvun puolella välin, Sega ja Nintendo julkaisivat omat VR-laitteensa. Segan laite oli nimeltä Sega VR-1 ja Nintendon konsoli oli nimeltä Virtual Boy, mutta ne eivät onnistuneet houkuttelemaan monia asiakkaita (Barnard 2019.)

Tähän mennessä VR-laitteet eivät nähneet suurta käyttöä, sillä ne olivat kalliita, kömpelöitä ja niiden sisältö oli erittäin rajallinen. Monet epäonnistuneet laitteet jättivät pahan maun kuluttajien suuhun, kunnes vasta 2010-luvun alussa VR-tekniikka alkoi näkemään toivoa ja kiinnostusta. Tietotekniikan kehitys oli tuottanut tehokkaampia, halvempia ja pienempiä tietokoneita, sekä niitä löytyi lähes joka taloudesta, jonka ansiosta tulevaisuudessa olisi helpompi myydä tehokkaita VR-laitteita suuremmalle yleisölle.

Yksi ensimmäisistä henkilöistä, kuka tajusi tämän oli Palmer Luckey. Luckey tunnetaan tällä hetkellä virtuaalitodellisuutta johtavan Oculus-yrityksen perustajana. Oculus sai alkunsa vuonna 2010, kun 18-vuotias Luckey kehitti ensimmäisen prototyypin hänen Rift-laseistansa. Ennen näitä laseja kaikki muut laitteet käyttivät kalliita ja vaikeasti tuotettavia

linssejä, jotta ne pystyisivät korjaamaan kuvan vääristymän, mutta Rift-lasit pystyivät käyttämään halpoja linssejä, sillä kuvan vääristymän korjaus tapahtui ohjelmiston puolella. Lasien näyttöinä myös käytettiin älypuhelimien näyttöjä, sillä niiden hinta oli pudonnut suuresti älypuhelimien suosion myötä. Loppujen lopuksi lasit olivat paremman laatuaisia ja halvempia valmistaa kuin aiemmat laitteet. Vuonna 2012 Luckey rakensi ensimmäisen prototyypin ja dokumentoi työtään keskustelupalstalle, kunnes tunnettu pelinkehittäjä John Carmack huomasi Luckeyn päivitykset. Luckey lähetti prototyypin Carmackille, jonka jälkeen Carmack esitteli niitä eri journalisteille E3-messuilla. Puolentoista vuoden päästä Oculus nettosi 2,4 miljoonaa dollaria Kickstarter-kampanjassa. Oculus myytiin Facebookille kahdella miljardilla dollarilla vuonna 2014 (Clark 2014.)

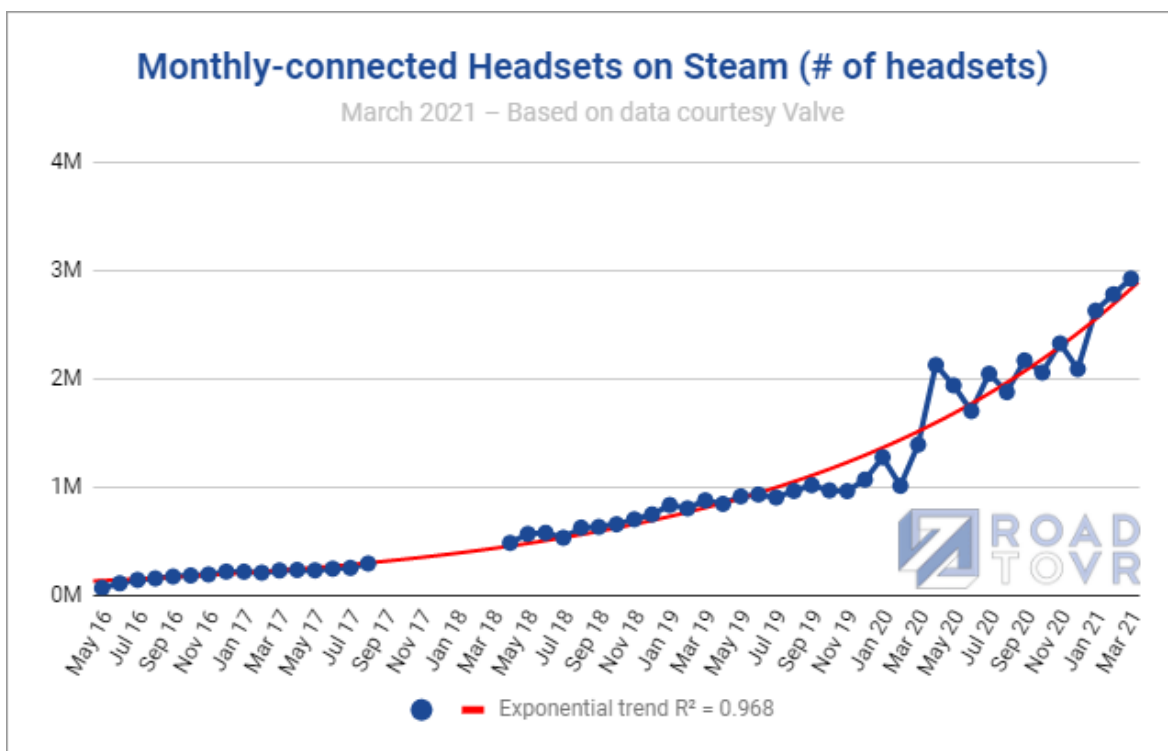
Samsung ja Google julkaisivat omat härpäkkeensä, joihin käyttäjä pystyi laittamaan puhelimensa ja muuttamaan sen VR-lasien näytöksi. Nämä laitteet olivat niin sanottuja 3DoF eli three degrees of freedom -laitteita, joka tarkoitti, että niillä ei voitu liikkua ympäristön sisällä, vaan lasella pystyttiin vain kääntämään päätä. Molempien tuotteet olivat erityisen suosittuja kuluttajien kesken, ja ne antoivat monille rajallisen ensikokemuksen virtuaalisesta todellisuudesta. Googlen lasit (kuva 2) olivat pahvista taitettu laatikko, joka voitiin ostaa muutamalla eurolla.



Kuva 2. Google Cardboard -laite (Riley 2017)

Vuonna 2016 alkoi Oculusin ja HTC:n kisa markkinoiden kärjestä. HTC julkaisi heidän Vive-lasinsa, jotka saivat etumatkan Oculusin Rift-laseihin, sillä Viven mukana tuli erinomaiset ohjaimet, kun taas Riftiä jouduttiin pelaamaan Xbox-ohjaimella. Oculus sai HTC:n etumatkan kiinni vasta vuonna 2018, sillä heidän uusien Oculus Touch -ohjaimien ja Rift-

lasien pakettitarjouksen hintaa pudotettiin. Tämän lisäksi Oculus julkaisi suosittuja pelejä ja sovelluksia yksinomaisesti vain heidän pelikauppaansa. PlayStation VR julkaistiin vuoden 2016 loppupuolella, ja ne olivat erittäin suosittuja PS4-konsolin pelaajien kesken. PlayStation VR myi viisi miljoonaa kappaletta 2019 vuoteen mennessä ja täysin päihitti PlayStationin odotukset. Oculus julkaisi Quest nimiset lasit, jotka olivat uudentapaiset standalone eli itsenäiset VR-lasit, joiden sisällä oli älypuhelimien tapaiset osat. Laseihin ei tarvittu tietokonetta, ja ne olivat täysin langattomat ja halvemmat muihin saman laatuisiin laseihin verrattuna. Vuonna 2019 Valve lopetti yhteistyönsä HTC:n kanssa ja julkaisi omat heidän Index-lasinsa, joilla oli huippuluokkaiset tekniset tiedot ja uudenlaiset ohjaimet, jotka pystyvät seuraamaan käyttäjän jokaista sormeaa (ThrillSeeker 24.2.2020, 15-22 min.)



Kuva 3. Steamiin yhdistettyjen VR-lasien määrä kuukausittain (Lang 2021a)

Kuten kuvasta 3 nähdään, virtuaalitodellisuus kasvoi suuresti vuonna 2020, sillä Valven odotettu Half-Life: Alyx -peli julkaistiin maaliskuussa Steam-alustalle, joka nosti kuukausittaisen VR-lasien käyttäjien määrän Steam-alustalla ensimmäisen kerran yli kahden miljoonan. Pelin suuri vaikutus käyttäjien määrään nähdään kuvassa 3. Oculus taas julkaisi Quest 2-lasit (kuva 4) vuoden 2020 lokakuussa, jotka ovat itsenäiset lasit samoin kuin edeltäjänsä, joten niiden käyttöön ei ole pakko omistaa erillistä tietokonetta, mutta siitä huolimatta, Quest 2 on noussut PCVR-laitteiden ja Steamin kärkeen. Tällä hetkellä ne vievät lähes kolmasosan kaikista Steamiin yhdistetyistä VR-laseista.



Kuva 4. Oculus Quest 2 VR-lasit ja lisätarvikkeet (Oculus 2021)

## 2.2 Laitteisto ja ohjelmisto

HTC johti Steamiin yhdistettyjen VR-lasien kuukausittaista määrää vuonna 2016 heidän Vive-laseillansa. Laseja oli huipullaan Steamiin yhdistettynä kaksi kolmasosaa kaikista lasista, mutta viime vuosina HTC:n lasien osuus on pudonnut alle 15 prosenttiin. Tämän vuoden marraskuussa Facebookin omistaman Oculuksen kaikki VR-lasit veivät 60 prosenttia kaikista Steamiin kuukausittain yhdistetyistä lasista. Steamiin yhdistettyjen VR-lasien määrä on myös kasvanut suuresti Quest 2 -lasien julkaisun jälkeen. Kaikista Steamien käyttäjistä 2,13 % oli yhdistänyt VR-lasit Steamiin tämän vuoden marraskuun aikana, kun taas tasan vuosi sitten määrä oli vain 1,29 %, vaikka samaan aikaan julkaistiin erittäin suosittu Half-Life: Alyx -peli (Heaney 2021.)

Kuvassa 5 nähdään heinäkuussa julkaistu Steamien kuukausittainen laitteistokyselyn VR-osio. Tämän mukaan Oculuksen Quest 2 -lasit vievät 32,6 % kaikista lasista, ja heidän Rift-lasinsa ovat toisella sijalla 17,8 prosentilla. Tämä tarkoittaa, että yli puolet yhdistetyistä laitteista ovat Oculuksen valmistamia. Valven laadukkaammat ja kalliimmat Index-lasit ovat kolmantena 16,2 prosentilla. Oculuksen tuotteet ovat noin kolme kertaa halvemmät kuin Valven Index-lasit, mutta Valve on pysynyt Oculuksen perässä lasien laadun, laajan näkökentän ja erinomaisien ohjaimien avulla.

Oculus Quest 2	32.56%	+1.49%
Oculus Rift S	17.79%	-0.68%
Valve Index HMD	16.23%	-0.45%
HTC Vive	10.37%	-0.86%
Oculus Rift	5.59%	+0.19%
Windows Mixed Reality	5.57%	+0.09%
Oculus Quest	5.33%	+0.10%
HTC Vive Pro	1.99%	-0.02%
HTC Vive Cosmos	1.78%	+0.03%
Riftcat Vridge	0.49%	-0.07%
iVRy	0.46%	+0.11%
HTC Vive Pro 2	0.34%	+0.26%
Pico Neo 3	0.24%	-0.03%
Sony PlayStation VR	0.16%	-0.03%
Pimax 5K Plus	0.14%	+0.01%
Pimax 8K	0.12%	-0.04%
HTC Vive Elite	0.08%	-0.06%
Pico Neo 2	0.08%	+0.03%
Iriun VR	0.03%	-0.01%
Pimax Artisan	0.02%	-0.03%
ALVR	0.02%	-0.00%
Oculus Rift DK2	0.02%	-0.02%
Other	0.58%	+0.02%
Steam users with VR Headsets	2.07%	+0.21%

Kuva 5. Steamin laitteistokyselyn VR-lasien osuudet heinäkuussa 2021 (Steam 2021)

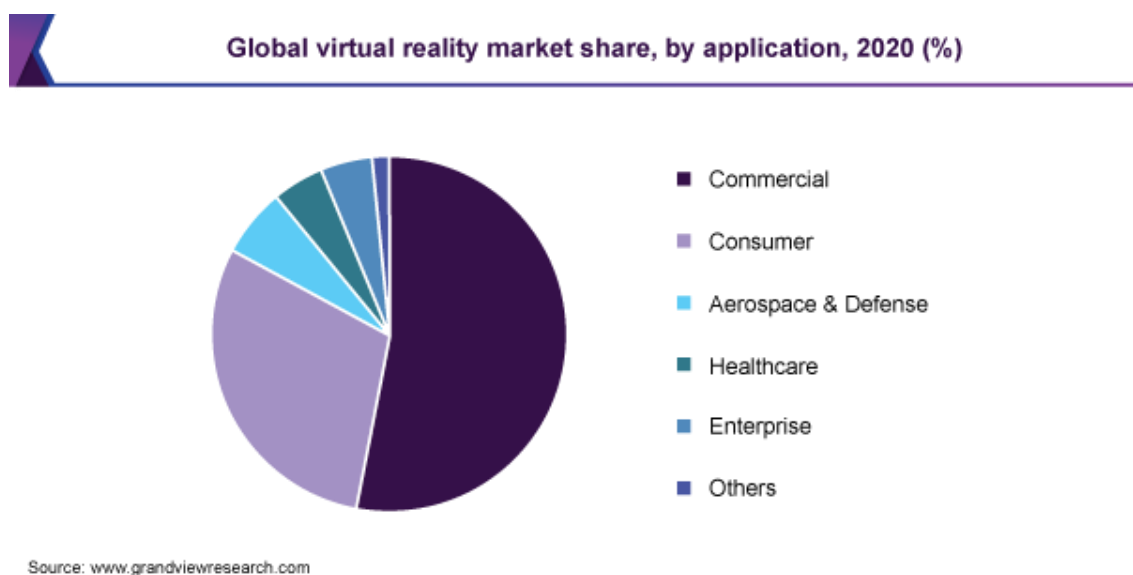
Oculus Quest 2 julkaistiin vuoden 2020 lokakuun puolessa välissä, ja Rec Room -videopelin latausten mukaan arvioitiin, että laseja myytiin viime vuoden neljännellä kvartaalilla noin 2–3 miljoonaa kappaletta. Tämän arvion mukaan lasit myivät kolme kertaa enemmän samalla aikamäärällä, kuin vuonna 2016 samoihin aikoihin julkaistu, sekä aikaisemmin titteliä pitäneet Playstation VR -lasit (Lang, 2021b.)

Quest-sarjan lasit ovat olleet erittäin suosittuja, sillä ne ovat halpoja, täysin langattomia ja erittäin hyvänlaatuisia hintaan verrattuna. Quest-lasit ovat itsenäiset lasit eli niillä voidaan pelata ilman tietokonetta lasien oman kaupan pelejä, mutta ne voidaan myös yhdistää USB:llä tai WiFin kautta langattomasti tietokoneeseen, jonka avulla voidaan pelata, esimerkiksi Steamin tai Oculuksen PCVR-pelejä. Suuren menestyksen takia monet yritykset ovat alkaneet valmistamaan omia itsenäisiä laseja, kuten HTC:n julkaisemat VIVE Pro 2 -lasit ja Picon Neo-sarjan lasit, mutta mitkään niistä eivät ole pystyneet päihittämään Quest 2 -lasien 350 euron hintaa ja Oculukselle yksinomaisesti valmistettuja pelejä.

Yritysmarkkinoille XR-laseja valmistava Varjo on suomalainen yritys, joka johtaa yrityksille tarkoitettavien lasien teknologiaa. Lasit toimivat ”ihmisen silmän tarkkuudella” ja niiden näköala kattaa 115 astetta. Laseissa on myös silmien ja käsien seuranta, LiDAR-skanneri ja värillinen pass-through, jonka avulla käyttäjä voi asettaa 3D-mallinnuksia todelliseen maailmaan. Varjon uusimmat lasit ovat nimeltään XR-3, ja ne maksavat 5495 euroa, jonka lisäksi pitää maksaa 1495 euron vuosittainen tilausmaksu (Graham, 2020.)

### 2.3 Käyttökohteet

Kuten kuvasta 6 nähdään, kaupallinen osuus maailmanlaajuisesta virtuaalitodellisuuden markkinoilla oli yli 53 % vuonna 2020, ja sen odotetaan pysyvän johdossa seuraavat seitsemän vuotta. Kaupalliset alat, kuten vähittäismyyjät, autosalit ja kiinteistöala tulevat kasvattamaan kaupallista puolta seuraavien vuosien aikana. Kuluttajien osuuden arvioidaan kasvavan suurella vauhdilla ja nousevan johtoon, mutta terveydenhuollon sektorin uskotaan nousevan eniten kaikkiin muihin verrattuna vuodesta 2021 vuoteen 2028. Yrityssektoria kasvattaa yrityskoulutuksen sekä viestintä- ja yhteistyövälineiden tarve organisaatioissa (Grand View Research 2021.)



Kuva 6. Maailmanlaajuisen virtuaalitodellisuuden markkinaosuudet vuonna 2020 (Grand View Research 2021)

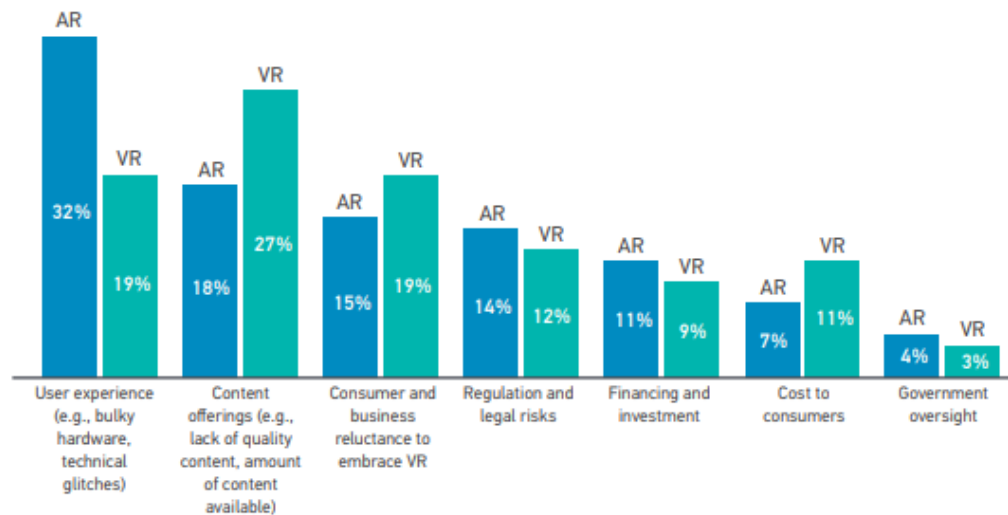
Virtuaalista todellisuutta käytetään hyväksi eri tavoilla eri aloilla, kuten autoteollisuudessa VR-tekniikan käyttö auttaa insinöörejä suunnittelemaan ja rakentamaan autoa konseptivaiheessa ennen kallian prototyyppivaiheen aloittamista, joka säästää rahaa ja aikaa. Terveydenhuollossa potilaille on alettu tarjoamaan VR-altistushoitoa, joka voi auttaa mielen-terveysongelmista kärsiviä henkilöitä. Matkailuala antaa potentiaalisille asiakkaille virtuaalierroksia, esimerkiksi maamerkkien, ravintoloiden ja hotellien luona (Grand View Research 2021.)

### 2.4 Kehityskohteet

Perkins Coien julkaiseman kyselyn mukaan (kuva 7), ammattilaiset uskoivat, että VR-tekniikan massakäyttöön otton esteenä ovat sisällön heikkous, käyttäjäkokemus ja vastahakoisuus VR:ää kohtaan. Ammattilaiset myös uskoivat (kuva 8), että ohjelmistojen

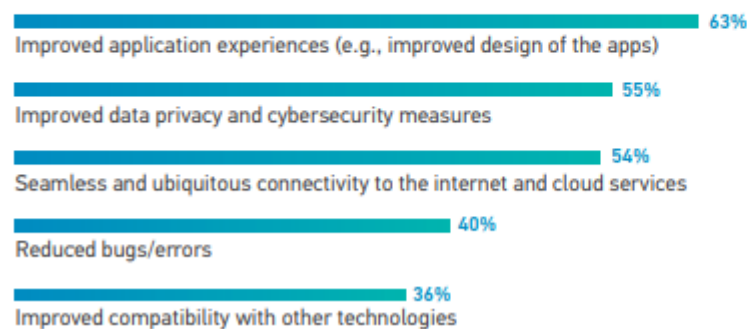
käyttäjäkokemuksen, tietosuojan ja verkkoon yhdistettävyyden parantaminen tulevat tekemään suurimman vaikutuksen kuluttajien kesken.

What is the biggest obstacle to mass adoption of AR and VR technologies?  
Select one for each technology:



Kuva 7. AR- ja VR-tekniologioiden esteet massakäyttöön otossa asiantuntijoiden mielestä (Perkins Coie 2020, 22)

What are the top improvements/solutions across immersive technology software that will make the greatest impact with consumers in the next two years? (Select all that apply)



Kuva 8. Suurimman vaikutuksen tuovat ratkaisut kuluttajien immersiiivisten teknologioiden ohjelmistoissa ammattilaisten mielestä (Perkins Coie 2020, 28)

Suuri ongelma VR-käyttäjien kesken on ollut matkapahoinvoinnilta tuntuva olo eli liikepahoinvointi, joka usein johtuu huonosti toimivasta ohjelmasta tai käyttäjän kokemattomuudesta. Käyttäjät voivat tuntea, esimerkiksi päänsärkyä, pahoinvointia ja pyörrytystä. Pelintekijät pyrkivät luomaan mahdollisen tehokkaasti toimivan sovelluksen vähentämällä pätkimistä ja parantamalla ruudunpäivitysnopeutta, sekä vähentämällä äkkinäisiä kiihdytyksiä ja liikkeitä, joita käyttäjä ei itse ole tehnyt (Trescak 2019.)

Kaikki mahdolliset tekniset parannukset VR-teknologiassa usein perustuvat immersion laajentamiseen ja parantamiseen. Immersiolla tarkoitetaan käyttäjän uppoutumista virtuaaliympäristöön niin, että hän tuntee olevan mahdollisimman uskottavasti vuorovaikutuksessa tämän ympäristön kanssa. Jotta saataisiin mahdollisimman hyvä immersio, niin käyttäjän aisteja pitää simuloida mahdollisimman todellisesti. Käyttäjä saa suurimman parannuksen paremmilla VR-laseilla, joissa on korkealaatuinen näyttö ja leveä näkökenttä. Tämän lisäksi laitteiden keveys ja mukavuus ovat erittäin tärkeitä, jotta käyttäjä ei tuntisi olevan lasien sisällä. Yritykset ovat myös yrittäneet luoda muita aisteja simuloivia tuotteita, kuten haptisia asusteita (kuva 9), jotka antavat käyttäjälle palautetta värisemällä, sekä tuoksuja tuottavia koneita. Nämä laitteet ovat usein kalliita ja pikkunäppäriä, joten ne eivät ole nähneet erityisen suurta suosiota markkinoilla.



Kuva 9. BHaptics TactSuit X40 -haptinen liivi (BHaptics 2021)



### 3 Unity-pelimoottori

Toisin kuin virtuaalinen todellisuus, pelimoottorit ovat olleet käytössä jo monta vuotta, tämän takia voitiin käyttää vanhempia lähteitä Unitysta ja pelimoottoreista, joka helpotti tiedonhakua. Tiedonhakua myös helpotti se, että pelimoottorien käyttö ja ominaisuudet eivät ole muuttunut paljoa viime vuosien aikana.

Peliala ja pelien kehitys kasvaa nopeaa vauhtia uusien laitteiden ja teknologioiden tullessa markkinoille. Tämän takia pelien rakentamiseen tarvitaan suuri määrä eri taitoja, kuten pelin grafiikoiden, animoinnin ja ohjelmoinnin tekemistä. Nykyään pelinkehittäjillä on valmiiksi tehtyjä ohjelmia ja työkaluja, joiden avulla he voivat säästää aikaa ja keskittyä muihin tehtäviin (Anumadu 25.9.2020.)

Alun perin peliyritykset kehittivät vain yhdelle pelille tarkoitetun renderointimoottorin, mutta lopulta yritykset alkoivat kehittämään yhtiön sisäisen pelimoottorin, jota he pystyivät käyttämään yhtiön omiin peleihin. Markkinoille alkoi kuitenkin saapua pelimoottoreita, joita kuka tahansa pystyi käyttämään. Pelimoottorit ovat erityisen hyödyllisiä pienille peliyrityksille, sillä niiden avulla voidaan säästää rahaa ja aikaa, sekä julkaisu eri alustoille on helppoa (Gamescrye 3.10.2016.)

Nykyään monet peliyritykset isosta pieneen käyttävät markkinoilla olevia ilmaisia pelimoottoreita säästääkseen resurssejaan. Monet pelimoottorit maksavat vasta, kun yritys on alkanut tienamaan tietyn määrän rahaa peleillään. Suurimpia moottoreita ovat, esimerkiksi Unity, Unreal Engine ja Godot. Näistä tunnetuin on Unity, sillä se on helppokäyttöinen aloittelijoille ja käyttäjien tekemät resurssit ja videot, sekä Unityn laaja yhteisö auttaa jokaisessa pelinrakennusvaiheessa.

#### 3.1 Historia

Unity perustettiin pienessä asunnossa Kööpenhaminassa vuonna 2004. Perustajat Nicholas Francis ja Joachim Ante tapasivat OpenGL-keskustelupalstalla, kun Francis oli etsimässä apua oman MAC-pohjaisen pelimoottorin tekemiseen. Ante auttoi tekemään Francisin projektia ja lopulta jäi tekemään sitä kokoaikaisesti, sillä hänen peliprojektinsa, jota hän teki toisen ryhmän kanssa ei kulkenut eteenpäin. Ante ja Francis muuttivat Francisin kaverin asuntoon. Tämän henkilön nimi oli David Helgason. Ante ja Francis päättivät aloittaa pelistudion, jonka laajaa teknistä pohjaa voitaisiin lisensoida. Ante ja Francis nimesivät yhtiönsä Over the Edge Entertainmentiksi, ja Helgasonista tehtiin yrityksen toimitusjohtaja. Kun Unity-pelimoottori oli saatu valmiiksi, he alkoivat kehittämään ensimmäistä peliä moottorin markkinointia varten (Peckham 2019.)

Vuonna 2005 OTEE julkaisi heidän ensimmäisen pelinsä Unitylla, jonka nimi oli Gooball. Tämän pelin voittojen avulla he hankkivat lisää ohjelmoijia, jotta he saisivat Unity 1.0 -version julkaisukuntoon. Pelimoottori toimi vain MAC-tietokoneilla, mutta Unity 2.5 lisäsi Windows kannatuksen, jonka avulla Unityn rekisteröityjen käyttäjien määrä nousi yli kahteen sataantuhanteen käyttäjään ennen Unity 3.0 version julkaisua (Haas 2014, 8-10.)

Wikipedian (2021) mukaan vuonna 2012 Unitylla oli miljoona rekisteröityä käyttäjää, ja vuoteen 2016 mennessä määrä nousi yli 5.5 miljoonaan. Vuonna 2020 ja usean rahoituskierroksen jälkeen, Unity listattiin New Yorkin pörssiin. Arvioksi Unitylle annettiin alussa 13.7 miljardia dollaria. Koska Wikipedia on välillä epäluotettava lähde, niin yhteenvedosta valitut kohdat varmistettiin olevan oikein muiden lähteiden kautta.

Unitylla on suuri vaikutus pelialalla, sillä vuonna 2020 puolet kaikista PC-, mobiili- ja konsolipeleistä tehtiin Unitylla, sekä tuhannesta suurimmasta mobiilipelistä yli kaksi kolmasosaa tehtiin Unitylla. Unitylla tehtyjä sovelluksia ladattiin myös keskimäärin 5 miljardia kertaa kuukaudessa (Unity 2021a.)

### 3.2 Pelinkehitys Unitylla

Unityn ladattua, käyttäjä saa suuren määrän työkaluja, joiden avulla sisällön kehitys on erittäin helppoa. Tehokkailla varjostimilla, fysiikkaan perustuvilla materiaaleilla, jälkikäsitteilyllä ja korkean resoluution valaistusjärjestelmillä Unity pystyy valmistamaan erinomaisia grafiikoita. Unity kannattaa kaikkia suurimpia konsoleita ja käyttöjärjestelmiä, joten pelit voidaan julkaista mille tahansa alustalle. Yhteistyö Unitylla on mahdollista sisäänrakennetulla versionhallinnalla ja pilvi-integraatiolla, jonka avulla projektin hallinta ryhmän kanssa on helppoa. Isot yhtiöt, kuten Ubisoft ja NASA käyttävät Unitya, sillä sen avulla pystytään toteuttamaan erinomaisia VR-ympäristöjä (Petty 2019.)

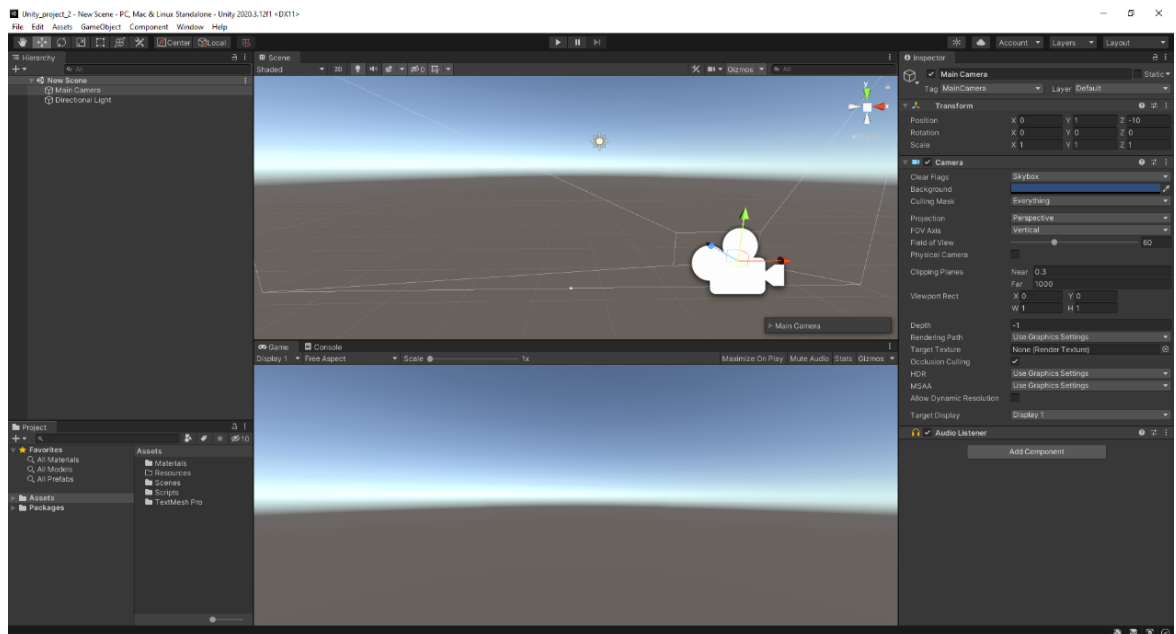
Unitylla voi valita eri lisenssejä, kuten personal, plus ja pro. Unity on ilmainen kaikille, kunnes yhtiö tai henkilö tienaa yli 100,000 dollaria peleillään. Plus-lisenssillä saadaan enemmän ominaisuuksia ja opetusmateriaalia, kun taas pro-lisenssi on tarkoitettu pelistudioille ja ammattilaisille, jotka tarvitsevat sisäistä tukea tai tienaa yli 200,000 dollaria peleillään (Petty 2019.)

Unity on pelimoottorin lisäksi IDE eli integroitu ohjelmointiympäristö, jonka avulla voidaan käyttää tarvittavia työkaluja yhdessä paikassa. Unityn visuaalisella editorilla on helppo raahata hiirellä eri elementtejä näkymään ja muunnella niitä. Koodia kirjoittaessa, Unity avaa valitun tekstieditorin, joista käytetyin editori on Microsoftin Visual Studio, joka toimii

saumattomasti Unityn kanssa. Unityn ohjelmointikielenä pääosin käytetään C#:a (Sinicki 2021.)

Unityssa voidaan käyttää useaa eri ohjelmointikieltä, mutta kaikista helpoin ja tunnetuin on C#. Microsoft kehitti C#-kielen vuonna 2000, sillä he huomasivat Javan suosion. Microsoft alkoi kehityksen ottamalla mallia Javasta, mutta syntaksi muutettiin tutummaksi C++ osajille (Gibson 2014, 238.)

Koska Unityn yhteisö on erittäin laaja, niin pelinkehitys on nopeaa valmiiden asettien avulla, sekä ongelmien ja bugien ratkominen on helppoa eri keskustelupalstojen ja hyvän dokumentoinnin kautta. Unity sisältää monia työkaluja, joiden ansiosta on mahdollista tehdä hankalia asioita ilman suurempaa tietoa kyseisestä aiheesta. Hyvät esimerkit tästä ovat OpenXR, joka helpottaa eri VR-lasien tavoittelua pelin sisällä ja Shader Graph, joka visuaalisella käyttöliittymällään helpottaa varjostimien koodaamista.



Kuva 10. Unity-näkymä pienen asettelun jälkeen

Kuvasta 10 nähdään Unityn käytetyimmät välilehdet. Keskellä kuvaa on kaksi ikkunaa, joiden avulla voidaan nähdä 3D-ympäristö. Ylempi ikkuna eli Scene-näkymä on tarkoitettu pelinrakennusta varten, sillä sen avulla voidaan siirrellä ja muokata peliobjekteja, kun taas alempi näyttää kaiken, mitä skenen kameran näköalassa on. Kamera on se peliobjekti, jota peli käyttää pelin renderoimiseksi näytölle. Vasemmalla ylhäällä löytyy Hierarchy-välilehti, josta nähdään kaikki skenessä olevat peliobjektit. Tämän alapuolella on kansiot, jotka sisältävät kaikki projektin resurssit ja assetit. Oikealla näkyy Inspector-välilehti, johon on avattu peliobjekti nimeltä Main Camera. Tämä välilehti siis näyttää peliobjektin tiedot, sijainnin skenessä ja komponentit, joista se on rakennettu.

## 4 Peliprojektin tekeminen

Projektin tarkoituksena oli tehdä VR-peli, jonka avulla voidaan demonstroida Unityn VR-pelinkkehitysprosessia ja VR-kykyjä. Vaikka Unity tarjoaa monia työkaluja pelinkehitykseen, niin pelinkkehitysprosessi silti vaatii valtavan määrän työtä ja eri taitoja. Tämän takia opinnäytetyössä tehtävän pelin aiheen tulisi olla melko yksinkertainen, mutta sen laajuuden kuitenkin kuuluisi sisältää paketteja ja komponentteja, jotka ovat erityisesti tarkoitettu VR-kehitykseen. Aiheeksi lopulta valittiin opetussimulaattorin tapainen tarkkuusammuntapeli, jonka avulla voidaan opetella aseiden käsittelyä 3D-ympäristössä, jotta aloittelijat voivat tutustua aseiden turvalliseen käyttöön ja sen mekaniikkaan.

Pelissä pelaaja voi opetella, esimerkiksi kuinka ase toimii, ja kuinka se ladataan aseeseen eri osiin viittaavien vihjeiden kautta. Pelin alussa voidaan harjoitella aseiden käyttöä lyhyellä ja pitkällä radalla. Lyhyellä radalla pelaaja opettelee käyttämään pistoolia, kun taas pitkällä radalla opetellaan rynnäkkökiväärin käyttöä. Kun pelaaja on opetellut mielestään tarpeeksi, hän voi ladata seuraavan skenen, joka on pidempi rata, jossa pelaaja itse navigoi talon ja sen takapihan läpi. Matkalla maaliin, pelaaja yrittää kaataa mahdollisimman monta eteen tulevaa maalitaulua. Maalitauluja löytyy myös huoneista, joihin päästääkseen pelaajan täytyy avata ovi. Maalitaulut pitää kaataa mahdollisimman tarkasti ja nopeasti, sillä lopuksi pelaajan suoritus pisteytetään, jotta hän voi seurata edistymistään.

Pelin rakennuksessa käytettiin Oculus Quest 2 -laseja ja ne yhdistettiin langattomasti tietokoneeseen Oculusin AirLinkin kautta. Rakennukseen olisi voitu kuitenkin käyttää mitä tahansa OpenXR:n kautta toimivaa laitetta. Quest 2 -lasien mukana tulee hyvät ohjaimet molemmille käsille, joita lasien kamerat seuraavat. Jotta lasit toimisivat Unityn kanssa, lasissa pitää olla hyväksyttyä Developer Mode -asetus, joka löytyy Oculusin puhelinsovelluksen asetuksista. Unity lisenssinä käytettiin opiskelijalisenssiä, jonka opiskelijat saavat ilmaiseksi, mutta se ei kuitenkaan eroa suuremmin ilmaisversiosta. Rakennuksessa käytetyn tietokoneen näytönohjain on GeForce RTX 3070, prosessorina on Ryzen 7 3700X ja RAM-muistia löytyy 16 gigatavua.

### 4.1 VR-Projektin aloitus

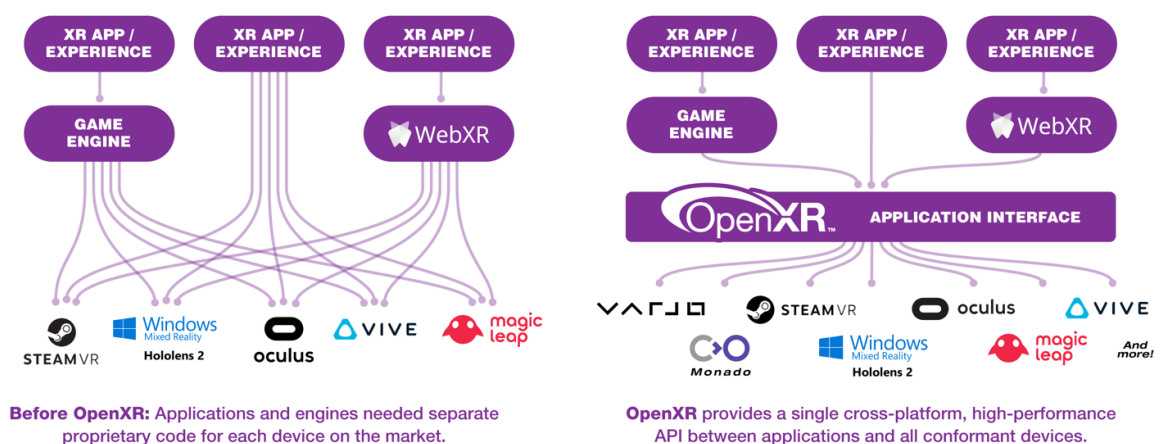
Ensimmäisenä pitää ladata Unity Hub, jonka avulla voidaan hallita kaikkia Unity-projekteja ja Unity-sovelluksen versioita. Lataus tehdään Unityn nettisivuilta, josta pitää ladata Unity Hub omalle käyttöjärjestelmälle. Kun lataus on valmiina, niin Hubista voidaan asentaa Unityn eri versioita ja käynnistää aiempia projekteja. Unityn version valinnassa on hyvä pitää mielessä, että uusimmissa versioissa saattaa olla bugeja ja muita ongelmia, joten usein suositellaan uusinta Long-Term Support -versiota, joka on tällä hetkellä vuodelta

2020 (Barnett 2021, 0-3 min). Seuraavaksi tehdään uusi projekti valitulla versiolla, sekä valitaan Universal Rendering Pipeline -templaatti ja annetaan projektille nimi. URP:n avulla voidaan nopeasti rakentaa optimoituja ja kevyitä sovelluksia, joka on tärkeää VR-peleissä (Unity 2021b.)

Kun projektinäkömää on saatu auki, on hyvä säätää projektinäkömää mieltymyksen mukaan työnteon tehokkuuden parantamiseksi. Projektin kansioiden, assettien ja peliobjektien järjestyksessä pitäminen on tärkeää varsinkin isoissa projekteissa, sillä se nopeuttaa resurssien etsintää ja vähentää virheitä. Projektin Hierarchy-välillehteen voidaan lisätä tyhjiä peliobjekteja, jotka periaatteessa toimivat skenen peliobjektien kansioina. Pelistä riippuen näitä voisi olla, esimerkiksi valaistus, ympäristö ja tavarat, ja niiden lapsiksi annetaan niihin kuuluvat skenessä olevat peliobjektit (Juegoadmin 21.1.2021.) Seuraavaksi voidaan asentaa liitännäiset, ulkoiset assetit ja muut paketit, jotka saadaan usein Package Managerista. Jotkin paketit tarvitsevat lisäasetuksia, joita voidaan säätää joko projektin asetuksista, työkalupalkista tai paketin omasta välillehdestä.

#### 4.1.1 OpenXR

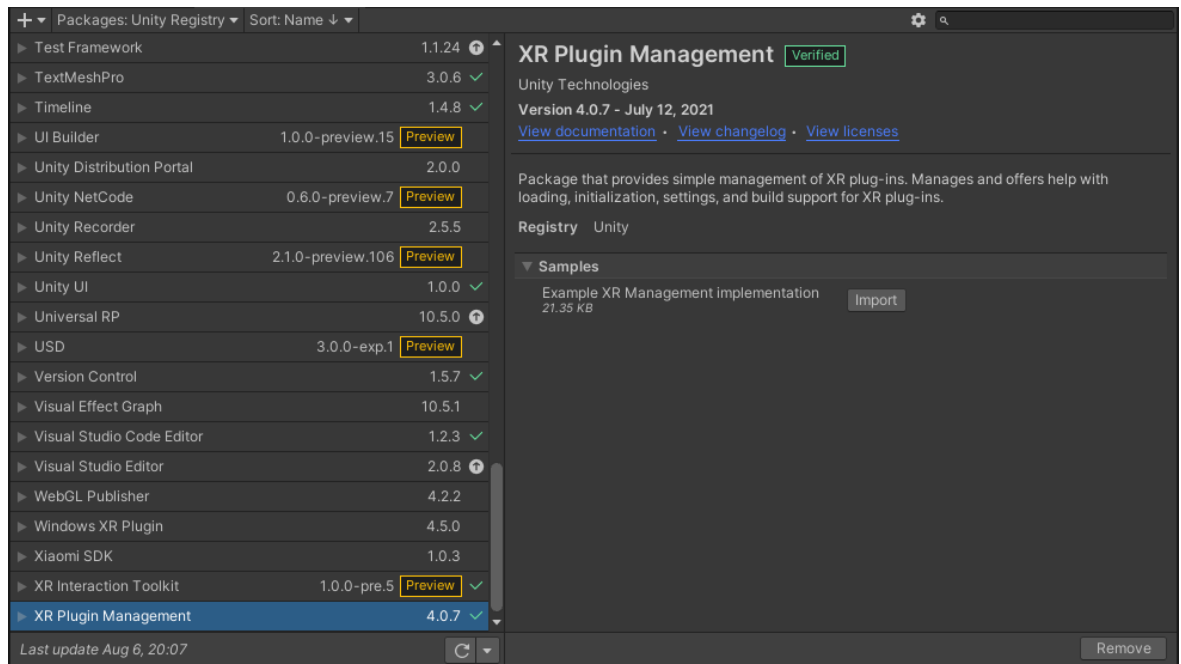
OpenXR on ilmainen ja avoin standardi, jonka avulla voidaan helposti tavoittaa kaikkia sitä kannattavia laajennetun todellisuuden laitteita. OpenXR toimii kaikilla suosituilla laitteilla, kuten Oculuksen, Valven ja HTC:n VR-laseilla (Khronos 2021.) OpenXR toimii siis laitteen ja sovelluksen välillä, kuten kuvasta 11 nähdään. Tämä liitännäinen on mielestäni yksi tärkeimmistä paketeista VR-kehityksessä, sillä se takaa, että kaikki laitteet toimivat kunnolla sovelluksen kanssa.



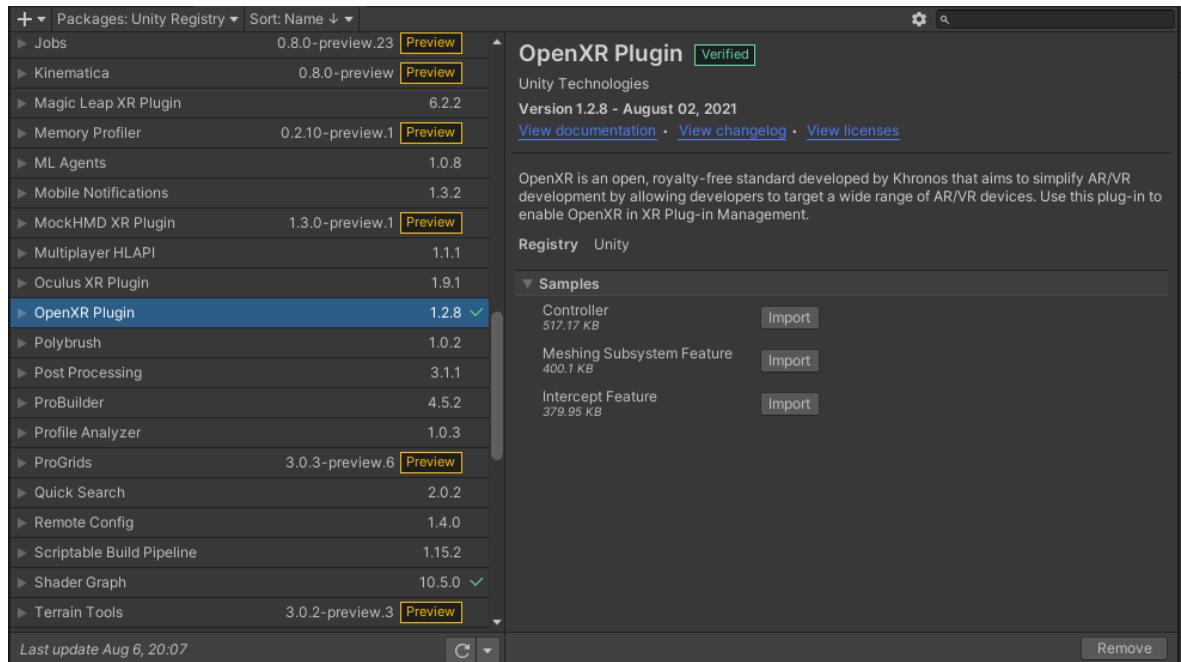
Kuva 11. Vasemmalla laitteiden tavoittelu sovelluksista ilman OpenXR:ää ja oikealla OpenXR:n kanssa (Khronos 2021)

Jotta OpenXR saadaan toimimaan, niin aloitetaan asentamalla XR Plugin Management (kuva 12) ja OpenXR Plugin (kuva 13). Molemmat niistä löytyvät Package Managerista,

mutta nähdäkseen oikeat paketit, pitää varmistaa, että managerin ylhäällä oleva Packages-valikko on laitettu Unity Registryksi.

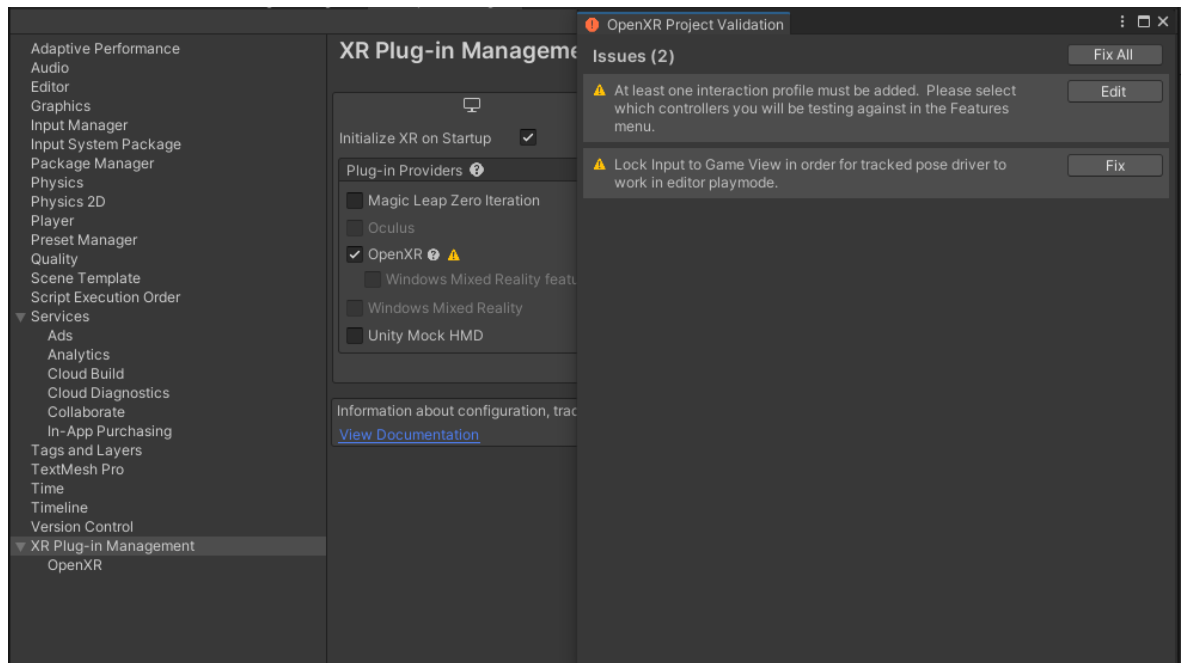


Kuva 12. XR Plugin Managementin asennussivu Package Managerissa



Kuva 13. OpenXR Pluginin asennussivu Package Managerissa

Tämän jälkeen mennään projektin asetuksiin ja valitaan listalla viimeisenä oleva XR Plugin Management (kuva 14), josta klikataan OpenXR-vaihtoehto. Kun OpenXR on päällä, niin sen viereen tulee varoitusmerkki, jota klikkaamalla avautuu ikkuna, josta voidaan korjata projektissa olevat ongelmat. Ikkunasta painetaan Fix All -nappia, jonka jälkeen yksi kohta jää varoittamaan vielä ohjainprofiilien puuttumisesta.



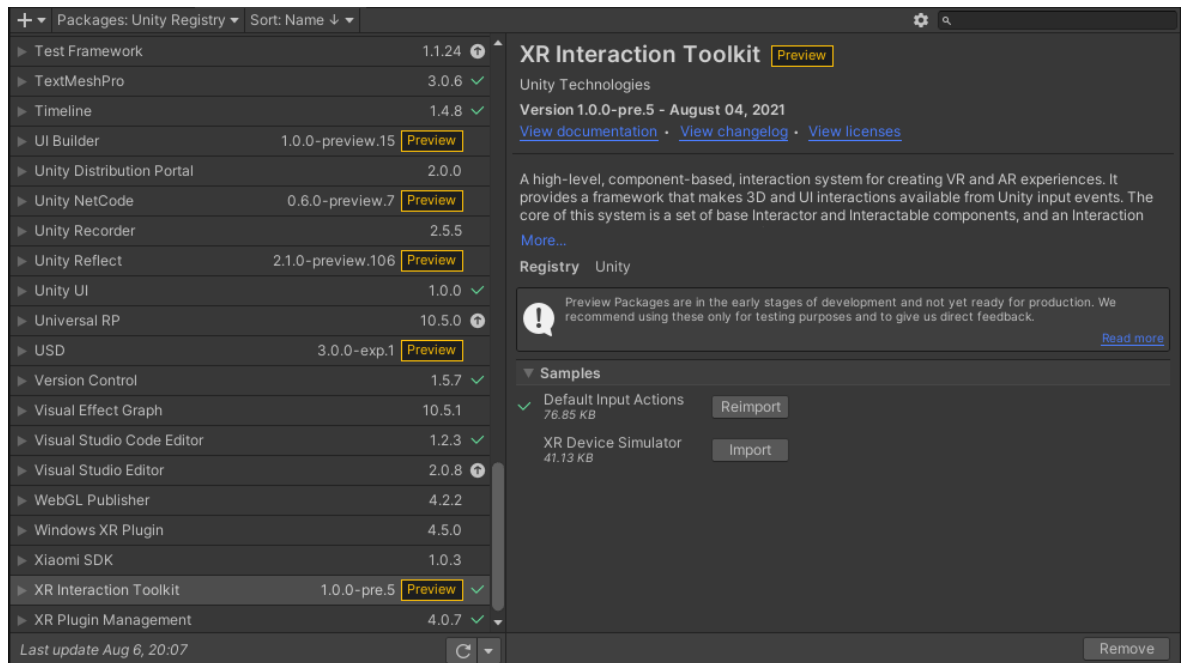
Kuva 14. XR Plugin Managementin käyttöönotto ja varoituksen korjaus projektin asetuksista

Ohjainprofiilit voidaan lisätä XR Plug-in Managementin alhaalle tulleesta OpenXR-sivulta. Interaction Profiles -kohdasta valitaan tarvittavat ohjainprofiilit, ja viimeisenä voidaan omavalintaisesti etsiä Player-sivun Other Settings -kohdasta löytyvä Active Input Handling, jonka asetukseksi laitetaan Both, jos projektissa käytetään vanhempia asetteja.

#### 4.1.2 XR Interaction Toolkit

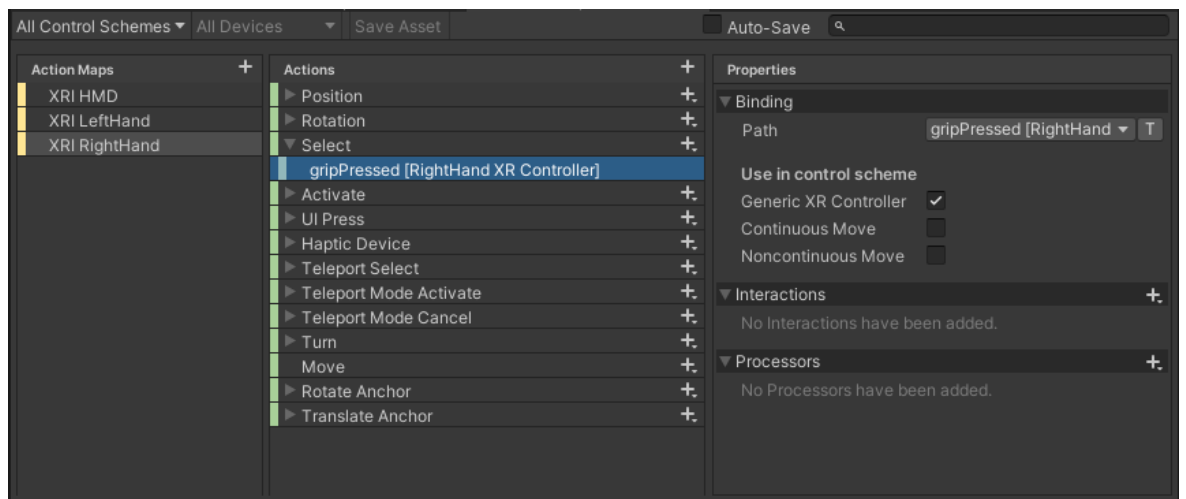
XR Interaction Toolkit on Unityn paketti, joka on komponenttipohjainen vuorovaikutusjärjestelmä VR- ja AR-kehitykseen. Paketista saadaan, esimerkiksi kaikkia laitteita kannattavien ohjaimien näppäin- ja liikeseuranta, perus vuorovaikutustoiminnot sekä haptinen palaute ohjaimille. Paketin mukana voidaan myös ladata yksinkertaiset liikkumis- ja kääntymiskomponentit (Unity 2021c.) Tämä työkalupakki on todella helppo käyttää ja sen avulla päästään nopeasti alkuun XR Rig -peliohjaimien kanssa. Kun aloitin VR-rakennuksen, niin muistan käyttäneeni tätä työkalupakkaa, sillä tietyt toiminnallisuudet, jotka tämä pakki ratkaisee, olivat liian hankalia aloittelijalle.

Ennen XR Interaction Toolkitin (kuva 15) asennusta pitää projektin asetuksista sallia Enable Preview Packages -kohta, joka löytyy Package Manager -kohdasta. Sen jälkeen paketti voidaan asentaa Package Managerista samoin kuin OpenXR Plugin, ja samalla ladataan Samples-kohdan alta löytyvä Default Input Actions.



Kuva 15. XR Interaction Toolkitin lataussivu Package Managerissa

Asennuksen jälkeen projektin kansioihin syntyy Samples-kansio, jonka lopussa löydetään viisi esiasetusta. Niistä voidaan valita mitä projektissa tarvitsee. Klikkaamalla tiettyä esiasetusta, Inspector-välilehteen aukeaa asetuksen tiedot ja ylhäällä olevaa nappia painamalla voidaan lisätä se projektiin. Samasta kansioista löytyy XRI Default Input Actions (kuva 16), jonka avaamalla löydetään toimenpiteet ohjaimien ja laitteiden eri syötteille.



Kuva 16. Toolkitista saatavat syötetoimenpiteet avattuna omaan ikkunaan

Viimeisenä mennään projektin asetusten Preset Manager -sivulle, josta ActionBasedControllerin Filter-kohdan tekstikenttiin kirjoitetaan Right ja Left. Tämän jälkeen on tärkeää varmistaa, että lasit ja ohjaimet toimivat Unityn kanssa. Testaamisen voi tehdä lisäämällä XR Rig -peliohjainten skeneen klikkaamalla oikealla näppäimellä Hierarchy-välilehteä ja etsimällä XR Rig (Action Based) -peliohjainten, jonka jälkeen lisätään pieni kuutio molempien



käsien lapseksi siksi aikaa, että voidaan lisätä kunnon kädet. Lopuksi ajetaan peli Unityssa ja tarkistetaan, että XR Rig toimii oikein, sekä Unity löytää lasit ja ohjaimet.

#### **4.1.3 Asset Store ja muut paketit**

Assetit ovat erilaisia tiedostoja, kuten ääniä, kuvia, 3D-malleja tai muita tiedostoja, joita Unity kannattaa. Unityn Asset Storessa on suuri määrä ilmaisia ja maksullisia asetteja, joita Unityn yhteisö on jakanut sinne (Unity s.a.) Monet suuret ja korkealaatuiset asettipaketit ovat maksullisia ja ilmaiset ovat usein yksinkertaisia materiaaleja ja tekstuureja. Ennen asettien lataamista pitää kirjautua sisään Unityn käyttäjätunnuksilla. Tämän jälkeen Unityn asetteja ja muita tuotteita voidaan ladata Asset Storen nettisivuilta, jotka oston jälkeen ilmestyvät asennettaviksi projektin Package Manageriin.

Tässä projektissa käytettiin Unityn omaa Terrain Tools -pakettia, jonka mukana saadaan ilmaisia realistisia materiaaleja ulkoympäristön maalaamiseen sekä työkalut maaston luomiseen. Toinen tärkeä paketti, jota projektissa hyödynnettiin, oli Probuilder, joka antaa laajemmat työkalut 3D-mallien luontiin ja käsittelyyn. Probuilderin avulla mallinnettiin, esimerkiksi toisen scenen talo ja huoneet. Muut ympäristön materiaalit ja tekstuurit olivat ilmaisia Unityn kaupasta, kun taas monet esineet ja välineet otettiin kaupasta saatavasta VR Interaction Framework -paketista, joka sisältää VR-malleja, kuten aseita, ovia, materiaaleja ja suuren määrän muita VR-pelinkehitykseen tarvittavia asetteja. VRIF maksaa tällä hetkellä 53,59 euroa. Näiden kolmen paketin avulla tämän VR-pelin kehitys nopeutuu, sillä 3D-mallinnuksen voi tehdä suoraan Unityn sisällä, sekä niiden avulla saadaan tarpeeksi hyvännäköisiä asetteja. Tosin jos projektiin halutaan parempia ja korkeampi laatuista 3D-malleja ja ympäristöjä, niin on parempi tehdä ne, esimerkiksi Blenderissä tai Mayassa, sillä ne sisältävät suuremman määrän työkaluja ja vaihtoehtoja 3D-mallinnukseen.

## 4.2 XR Rig setup

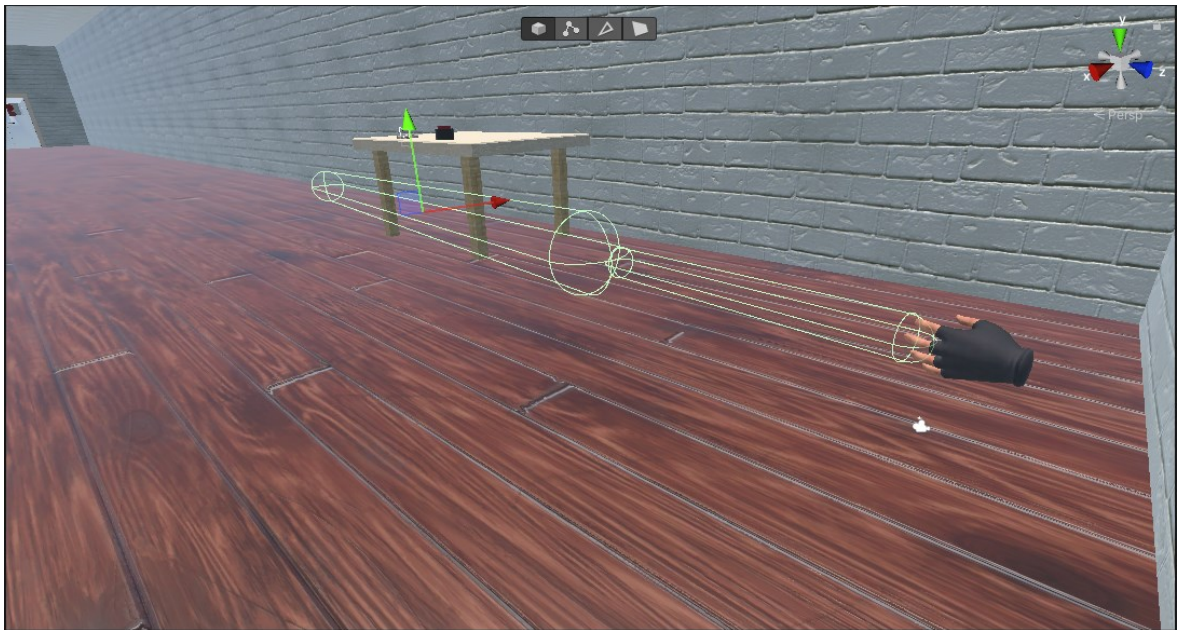


Kuva 17. Pelin XR Rig-peliobjektin rakenne hierarkiassa

Kuvasta 17 nähdään XR Rigin rakenne, ja seuraavaksi käydään läpi sen tärkeimmät peliobjektit ja niiden toiminnot. XR Rig toimii pelaajana ja siihen lisätään, esimerkiksi kamera, liikkumiskyky, animoidut kädet ja laitteiden seurantascriptit. XR Interaction Toolkitistä saadaan syötetoimenpiteet ja yksinkertaiset liiketoiminnot, joita käytettiin XR Rigin rakennuksessa. Rigin lapsena on PlayerController-peliobjekti, johon liitettiin Continuous Move ja Continuous Turn Provider, joiden avulla pelaaja voi liikkua ja kääntyä pelissä ohjaimien avulla. Samalla lisättiin Snap Turn Provider, mutta se laitetaan pois päältä, jotta pelaaja voi halutessaan laittaa sen jatkuvan kääntymisen tilalle menussa olevalla napilla. Snap turn eroaa jatkuvasta kääntymisestä niin, että se kääntää pelaajan näkymää tietyllä astemäärällä, joka parantaa liikepahoinvoinnista kärsivien käyttäjien kokemusta. Samaan peliobjektiin lisättiin kapselinmuotoinen collider-komponentti, jonka avulla pelaaja ei putoa lattian läpi tai kävele seinien läpi. Samalla lisättiin Rigidbody-komponentti, jotta painovoima vetää pelaajaa maahan päin. Rigiin olisi voitu lisätä juokseminen ja hyppääminen, mutta nämä lisäävät pahoinvoinnin mahdollisuutta, sekä ne eivät ole tarpeellisia tai sopivia tämän pelin laajuuteen.

Pelaajan päätä seuraa Center Eye Anchor -peliobjekti, joka sisältää kameran, audio kuuntelijan ja laitteen seurantakomponentin. Audio kuuntelija toimii periaatteessa pelaajan korvina, eli se löytää ympärillä olevat ääntä tuottavat lähteet, ja toistaa ne pelaajalle. Samalla lisättiin Tracked Pose Driver, joka kääntää VR-lasien liikkeen pelin sisälle muuttamalla peliobjektin koordinaatteja ympäristössä. Samanlaiset seurantakomponentit löytyvät molempien ohjaimien Anchor-peliobjekteista.

Models-peliobjekti löytyy molempien ohjaimien lapsista, ja ne sisältävät käsien 3D-mallit, sekä niiden sormien ja eri esineiden kädessä pidon animoinnin. Grabber ja RemoteGrabber ovat peliobjekteja, joiden sisälle lisättiin collider-komponentteja, jotka seuraavat niiden sisällä olevia peliobjekteja. Esineiden nostaminen tapahtuu siten, kun pelaajan käden collider osuu toisen esineen collideriin ja pelaaja painaa otenäppäintä, niin esine siirtyy pelaajan käden lapseksi. RemoteGrabberin (kuva 18) avulla voidaan nostaa kauempana olevia esineitä, sekä se auttaa pelaajaa nostamaan esineitä maasta, jotta pelaaja ei turhaan ala lyömään lattiaansa, yrittäessään nostaa pudonnutta esinettä.



Kuva 18. Vasemman käden RemoteGrabber-peliobjekti skenessä

Kun XR Rig oli suurin osin valmiina, niin siitä tehtiin prefab, jotta sitä voidaan käyttää muissa sceneissä. Prefab usein koostuu monesta eri assetista ja peliobjektista, joiden avulla on rakennettu kyseinen prefab. Prefab on helposti uudelleenkäytettävä peliobjekti, johon tehdyt muutokset päivittyvät yhdellä kertaa, muualla käytettyihin samoihin prefabeihin (llett 16.5.2020.)

#### 4.2.1 Käyttöliittymä

VR antaa uusia erikoisia tapoja tehdä käyttöliittymä ja menut. Monet pelit vielä valmistavat staattisia koko näytön käyttäviä menuja, jotka pysäyttävät pelin ja joskus jopa pilaavat pelaajan immersion, jotta kaikki asetukset ja tiedot mahtuvat niihin. Koska tämä projekti oli pieni, niin menun ratkaisun kanssa kokeileminen oli mahdollista. Menuksi peliin lisättiin pieni kanvas (kuva 19), joka laitetaan käden lapseksi, jotta menu liikkuu sen mukana. Oikeaan käteen lisättiin laserin tapainen säde, jolla voidaan käyttää menua. Menu aktivoituu, kun lasersäde osuu menun kanvas-komponenttiin. Menuun myös lisättiin kolme asetusta, lopetusnappi ja nappeja, joiden kautta pelaaja pystyy liikkumaan kaikkien ratojen välillä.



Kuva 19. Pelaajan menu ja laserkursori

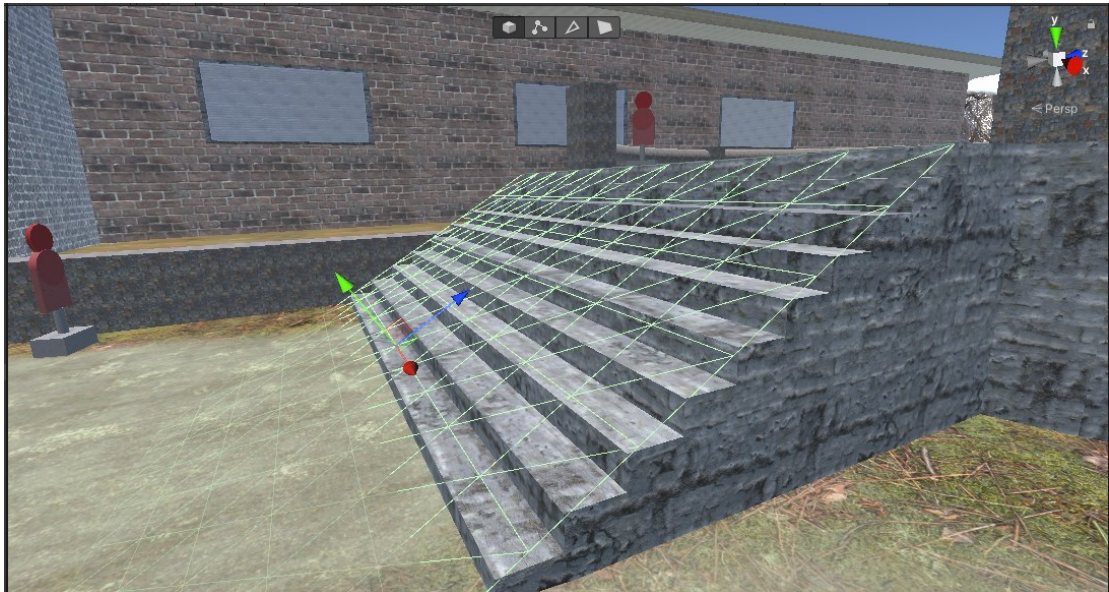
Seuraavaksi pohdittiin, kuinka aseiden käsittelyä opastavat ohjeet toteutetaan immersiiivisesti. VR Interaction Framework -paketti sisälsi Tooltip-peliobjektin, joka suurilta osin toimi hyvin tämän projektin kanssa. Tämä peliobjekti oli yksinkertainen pelaajan silmiä seuraava kanvas, jonka kulma yhdistetään Line Renderer -komponentilla kiinnitettävään peliobjektiin, kuten lippaaseen tai piippuun. Tämä komponentti piirtää kanvasin ja peliobjektin väliin viivan jokaisessa ruudunpäivityksessä, jonka avulla nähdään mihin ohje osoittaa. Ohje ei voi olla koko ajan päällä, vaan se pitää avata ja sulkea tarvittaessa. Tietty ohje siis laitetaan päälle ja pois päältä tiettyjen ehtojen tai tapahtumien perusteella. Yksi näistä ohjeista varoitti, jos aseiden piippu osoitti muualle kuin radalle. Tämä toteutettiin katsomalla aseiden ylimmän peliobjektin rotaation arvoa, ja jos se meni tietyn astemäärän yli, niin ohje laitettiin päälle. Yksi toinen näistä ohjeista näyttää, kuinka pelaaja voi irrottaa tyhjän lippaan aseesta. Tyhjän lippaan poiston ohje (kuva 20) aktivoituu, kun aseiden panokset loppuvat.



Kuva 20. Lippaan tyhjentyyessä, siihen liitetty tapahtuma laittaa kuvan ohjeen päälle

#### 4.2.2 VR:lle erityisiä ratkaisuja

Liikepahoinvointia on tärkeää estää VR-peliä rakentaessa. Pelissä mietittiin, mitkä ominaisuudet tuottavat suurimman määrän pahoinvointia ja sitten korjattiin tai poistettiin ne. Kuvasta 21 nähdään, kuinka rappuset tehtiin, jotta käyttäjän näkymä ei pompi jokaisen askelman kohdalla. Rappusten päälle laitettiin siis läpinäkyvä collider, joten pelaaja liikkuu tasaisesti rappusten yli ilman näkymän pomppimista. Peliin on hyvä lisätä eri liikkumiskeinoja, kunhan ne sopivat pelin aiheeseen. Teleporttaaminen on suosittu tyyli liikkua, sillä se vähentää liikepahoinvointia. Tämä liikkumiskeino toimii siten, että pelaajan kädestä lähtee kaareva lasersäde, ja pelaajan painaessa tiettyä nappia, hän siirtyy kohtaan, jossa laser ja lattia osuvat toisiinsa. Tämä tyyli ei sovi jokaiseen peliin, sillä liikkuminen teleporttaamalla ei ole kovin todellisen tuntuista. Tähän peliin olisi voitu myös lisätä juokseminen ja hyppääminen XR Interaction Toolkitin avulla, mutta ne eivät olleet tarpeellisia pelin toiminnallisuuden kannalta. Molemmat näistä myös saattavat aiheuttaa liikepahoinvointia, joten jos peli pakottaa pelaajaa esimerkiksi hyppimään jatkuvasti eri kohdissa, niin pahoinvoinnista kärsivän pelaajan on melkein pakko lopettaa kyseisen pelin pelaaminen.



Kuva 21. Rappusten päälle lisätty läpinäkyvä collider

Pelaaja voi nostaa esineen tai aseensa liikuttamalla käden esineen collideriin ja painamalla otentappia. Ase liikkuu pelaajan käden lappseksi, jotta se liikkuisi käden mukana. Samalla käsi animoi oteliikkeen ja sormet siirtyvät aseensa ennalta määrättyihin tartuntapisteisiin, jotta pelaajan ote näyttää mahdollisimman todelliselta. Aseensa osiin, kuten luistiin ja etukädensijaan lisättiin myös collider-komponentteja, jotta pelaaja voi pitää niistä kiinni. Aseensa lataaminen toimii vetämällä luistia taaksepäin, jolloin yksi panos siirtyy lippaasta aseeseen. Muuten aseet toimivat samoin kuin normaaleissa PC-peleissä, eli kun ase laukaistaan, ja siitä lähtevän Raycastin osuttua vahingoitettavaan esineeseen, aseensa vahinkomäärä vähentyy esineen terveispisteistä.

Pelissä käytettävät napit voidaan tehdä jousiliitosten avulla. Kun pelaajan käden collider osuu napin collideriin, niin jousiliitos antaa periksi ja nappi työntyy alaspäin. Tietty tapahtuma, kuten pelin ja ajastimen alkaminen tapahtuu, kun nappi on kulkenut alas tietyn määrän. Ovet toimivat lähes samoin tavoin kuin napit. Oven voi avata, kun ovenkahvan jousiliitos on kääntynyt tarpeeksi, kun taas oven aukeaminen tapahtuu saranaliitoksen avulla, joka auttaa ovea aukeamaan oikein.

### 4.3 Pelin kulku

Tässä alaluvussa kerrotaan lyhyesti, kuinka harjoittelu- ja rataskene kulkee eteenpäin. Projektissa siis käytettiin vain kahta sceneä, sillä harjoitteluskenen lyhyen ja pitkän harjoitteluradan välillä voidaan liikkua muuttamalla PlayerController-peliobjektin koordinaatteja. Kuvassa 22 nähdään käynnissä oleva pitkä harjoittelurata ja maalitaulut. Rataa ja skeneä voidaan vaihtaa pelin sisällä käyttöliittymän nappien kautta.



Kuva 22. Harjoituskenen pitkä tarkkuusammuntarata

Kun pelaaja aloittaa harjoitteluskenen pelin painamalla aloitusnappia, niin ensin entisen pelin pisteet nollataan, jonka jälkeen käytetään `InvokeRepeating`-metodia, joka alkaa lisäämään maalitauluja tietyin aikaväleihin satunnaisiin paikkoihin. `Coroutine`-metodi aloittaa heti laskemaan aikaa ja tietyn ajan päästä `InvokeRepeating`-metodi pysäytetään, joten peli loppuu siihen. Pelin pisteytysjärjestelmä tapahtuu pelaajan tuhottaessa taulun, joka laukaisee pisteenlaskentaskriptin metodin ja lähettää sille taulun aiemmin määrätyn pistemäärän. Laskentaskriptissä tuhotun taulun pisteet lisätään kokonaispistemäärään, jonka jälkeen skripti etsii kaikki skenen tulostaulut tagin avulla ja päivittää pisteet niiden tekstikenttiin.

Rataskene alkaa pelaajan syntyessä talon sisälle, jonka läpi pelaaja navigoi ja ampuu maalitauluja mahdollisimman nopeasti ja tarkasti. Maalitaulun päähän osuttaessa saadaan 13 pistettä, kun taas ylävartalosta saadaan 10 ja alavartalosta vain 5 pistettä. Peli alkaa pelaajan painaessa pöydällä olevaa aloitusnappia, joka aloittaa ajan laskemisen ja sijoittaa täyden lippaan pöydällä olevan pistoolin viereen. Toiseksi viimeisessä huoneessa pelaaja ottaa rynnäkkökiväärin, jonka jälkeen pelaaja siirtyy ulkopihalle, jossa maalitaulut ovat sijoitettu pidemmälle matkalle, sisällä oleviin maalitauluihin verrattuna. Kun pelaaja on päässyt radan loppuun, hän painaa nappia (kuva 23), jolloin ajanotto loppuu ja uusi laskenta alkaa, jonka lopussa pelaaja siirretään takaisin harjoituskeneseen. Rataskenen pisteytysjärjestelmä toimii samoin kuin harjoitteluskenessä, paitsi pelin manageriskripti hakee kokonaispistemäärän laskentaskriptistä, jonka jälkeen se sovittaa kokonaispisteet suoritus aikaan, josta saadaan lopullinen pistetulos.



Kuva 23. Rataskenen lopetusnappi, sekä kokonaispisteet ja maalitauluista saadut pisteet

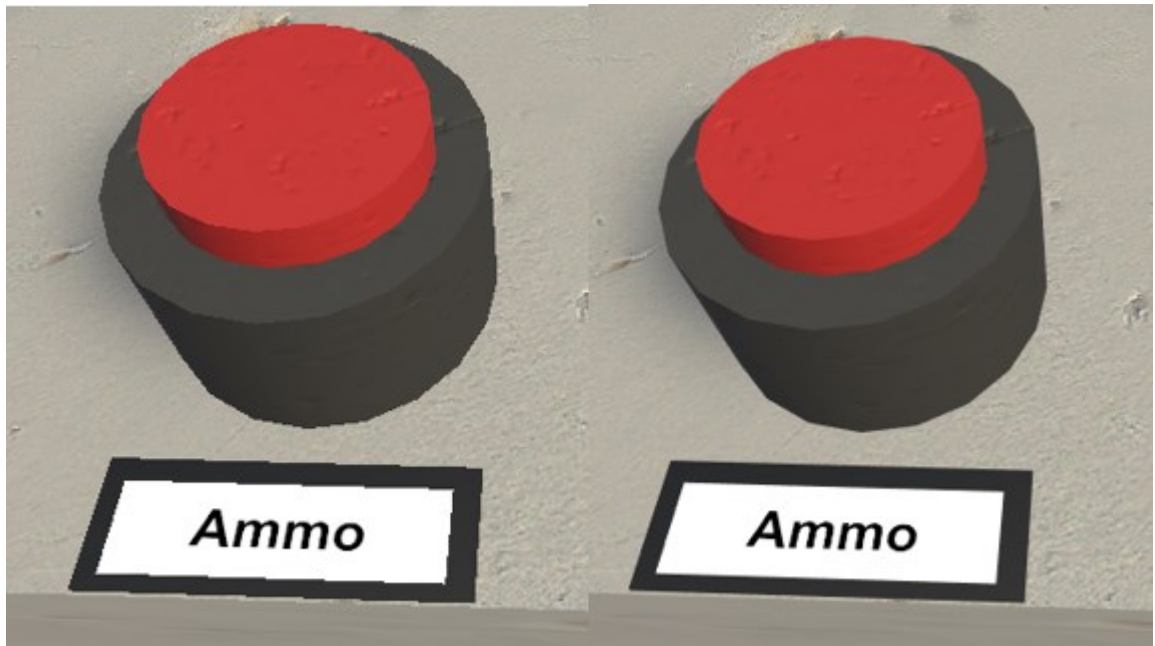
#### 4.4 Optimointi

Optimointi on erittäin tärkeää VR-peleissä, sillä huonosti optimoitu sovellus usein johtaa heikkoon ruudunpäivitysnopeuteen ja pätkivään peliin. Heikosti toimiva sovellus on yksi suurimmista syistä, miksi käyttäjät tuntevat pahoinvointia pelatessa, kun taas hyvin optimoitu sovellus vähentää pahoinvointia suuresti. Optimointi antaa monia muitakin etuja, kuten miellyttävämmän pelikokemuksen ja paremman kuvanlaadun, sekä sovellus käyttää vähemmän tietokoneen resursseja, joka johtaa siihen, että sovellusta pystytään suorittamaan heikommilla tietokoneilla (Skarredghost 28.11.2016). Peli yritetään saada toimimaan luotettavalla ruudunpäivitysnopeudella, jotta voidaan parantaa laatua ja pelaajan käyttökokemusta. Tämän lisäksi, vaikka projektin peli on pieni ja resursseilta vähän vaativa, sekä sitä suoritetaan tietokoneella, niin oli silti tärkeää optimoida peli hyvin ja käyttää mahdollisimman vähän korkealaatuisia ja kolmiomäärältään suuria asetteja, sillä jos peliä yritetään portata heikompitehollisille mobiililaitteille, niin peli ei menetä alkuperäistä laatuaan, ja pelin muuttamisesta johtuva työ vähenee.



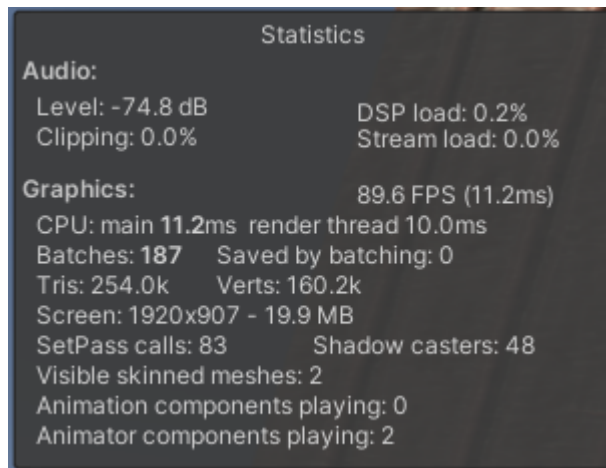


materiaalit ja esineet staattisiksi Inspector-välilehdestä, ja Universal Render Pipeline -asetuksista laitettiin Dynamic Batching -asetus päälle, joka auttaa tietokonetta piirtämään peliobjektit tehokkaammin ruudulle. Pienet staattiset peliobjektit voidaan piirtää samassa Batchissä ja tietokone tietää, että staattiset peliobjektit eivät tule liikkumaan ruudulla, joka nopeuttaa niiden piirtämistä. URP-asetuksissa HDR-asetus oli jäänyt päälle, joten se laitettiin pois päältä, koska siitä ei ole hyötyä tässä projektissa. Pelaajan kamerakomponentista poistettiin kaikki jälkikäsitteilyasetukset, joka parantaa suorituskykyä huomattavasti.

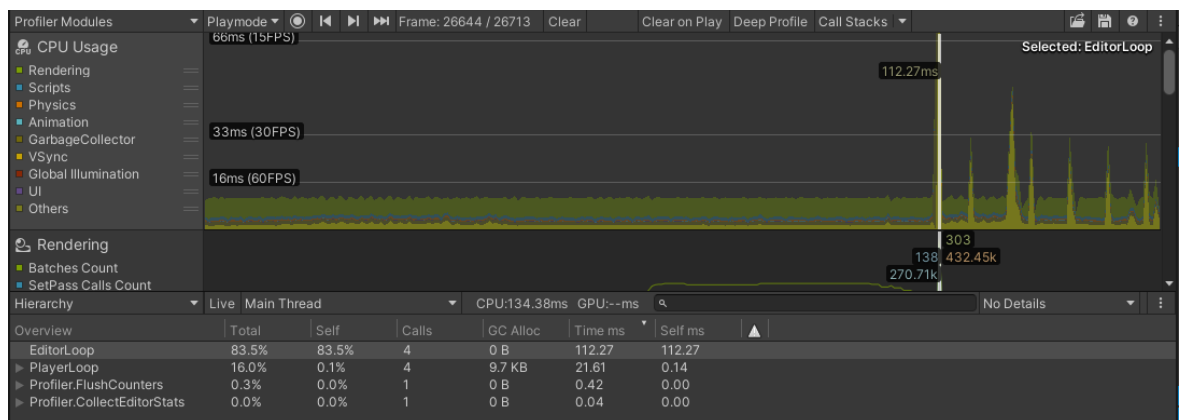


Kuva 26. Vasemalla anti-aliasing on pois päältä ja oikealla on lisätty kahdeksankertainen anti-aliasing

Pelin laatua parannettiin lisäämällä nelinkertainen anti-aliasing URP:n kautta, suurentamalla varjojen laatua, sekä tekstuuri-anisotrooppisen suodatuksen arvo nostettiin vähintään kahteen monessa assetissa. Kuten kuvassa 26 nähdään, anti-aliasing pehmentää reunojen pikseleitä, jotta esineiden kulmat näyttävät suuremmilta ja parempi laatuilta. Anisotrooppisen suodatuksen arvo oli tekstuureissa vain yksi, joka aiheutti isojen 3D-mallien laadun heikkenemistä liian lähellä kameraa. Arvo nostettiin suurissa malleissa kolmeen ja keskisuurissa kahteen, jotta suodatus ei näy liian lähellä kameraa. Pelaajan menuun lisättiin laatuasetus, jonka avulla voidaan vaihtaa pienemmän ja korkeamman asetuksen välillä. Pienempi laatuinen asetus on muuten samanlainen kuin korkeampi, paitsi anti-aliasing pudotettiin vain kaksinkertaiseksi, valojen laatua pienennettiin, varjojen laatu ja matka, jona ne renderöidään lyhennettiin, sekä renderöinnin resoluutiota pudotettiin hie-



Kuva 27. Statistics-ikkuna optimoinnin jälkeen



Kuva 28. Profiler-välilehden tiedot optimoinnin jälkeen

Optimoinnin jälkeen voidaan nähdä pelin suorituskyvyn parantuneen erittäin paljon. Kuva 27 on otettu hieman vilkkaammassa näkymässä kuin lähtökohdan kuva, ja suorituskyky on silti selvästi parantunut. Pelin ruudunpäivitysnopeus pysyi 89–90 päivityksen välillä ja se olisi ylittänyt tämänkin, jos pelin ruudunpäivityksen maksimiarvoa olisi nostettu. Tietokone siis prosessoi yhden ruudun 11,1 millisekunnissa, ja yhden ruudun päivitys näytölle kesti noin 10 millisekuntia. Koska VR-lasit voivat näyttää pelissä vain 90 ruudunpäivitystä sekunnissa, niin ruudun laskenta aina pysyy 11,1 millisekunnissa, mutta peliä suorittaessa Unitylla ilman laseja, sekä poistettaessa ruudunpäivityksen katto, ruudunpäivityksiä saadaan keskimäärin 300 sekunnissa. Samalla yhden ruudun laskenta kestää keskimäärin vain 3,3 millisekuntia. Statistics-ikkunan Batches-kentän arvo pieneni hieman, koska tietokone pystyy piirtämään peliobjektit dynaamisesti näytölle. Kuvasta 28 nähdään, että EditorLoop ja PlayerLoop ovat molemmat pienentyneet. Unity kuitenkin välillä aiheuttaa suuria piikkejä jostain syystä, mutta rakennetussa pelissä näitä ei tule enää. GC Allocation eli pelin suorituksen aikana syntyvän roskan määrä näyttää olevan riittävän pieni, joka tarkoittaa, että peli ei tuota turhaa roskaa ja siten hidasta peliä.

Jos pelistä haluttaisiin vielä tehokkaampi, niin voitaisiin käyttää object pooling -tyyliä, jossa peliobjektit ladataan heti pelin alkaessa, mutta ne jätettäisiin pois päältä, kunnes niitä tarvittaisiin. Tällä hetkellä esimerkiksi joka ikinen maalitaulu synnytetään täysin uutena satunnaiseen paikkaan pelin aikana, ja ne rikotaan, niihin osuttaessa tai niiden olemassaolon ajan loppuessa, joka nostaa roskan määrää. Valot myös lasketaan jokaisessa ruudunpäivityksessä, joka vie paljon tehoa, joten valot voitaisiin bakettaa (englanniksi baking), jotta valaistustiedot siirtyisivät välimuistiin ja valaistus olisi staattinen koko pelin ajan.

#### 4.5 Julkaiseminen

Projekti olisi voitu julkaista lähes tulkoon mille tahansa alustalle, sillä se on rakennettu OpenXR:llä, mutta koska peli ei ole kovinkaan erikoinen, niin päätin julkaista sen tunnetulle ja ilmaiselle Itch.io-nettisivulle. Julkaiseminen aloitetaan ensin rakentamalla peli Build Settings -ikkunasta. Seuraavaksi mennään kansioon, johon peli rakennettiin ja valitaan kaikki Unityn luomat tiedostot ja tehdään niistä Zip-kansio. Tämän jälkeen kirjaudutaan Itchiin ja tehdään uusi projekti. Projektin julkaisusivulla nimeksi annettiin Target Practice Game, sekä lisättiin Zip-kansio rakennetun pelin tiedostoista Uploads-kohtaan. Seuraavaksi kuvailtiin peliä, näppäinasetuksia ja käytettyjä asetteja kuvauskenttään. Lopuksi otettiin vielä muutama näyttökuvaa pelistä ja lisättiin ne sivulle. Pelin Itch.io-sivun voi nähdä liitteestä 1 tai tästä [linkistä](#).

## 5 Pohdinta

Opinnäytetyön tarkoituksena oli esitellä virtuaalisen todellisuuden pelinkehitystä Unitylla. Esittelyn avuksi rakensin pelin, jonka kautta voidaan käydä läpi kaikki projektin erityispiirteet ja erilaiset lähestymistavat, jota VR-kehityksen kanssa tulee. Tietoperustassa käytiin virtuaalista todellisuutta ja Unitya läpi, jotta lukija saa tarpeeksi hyvän käsityksen niistä. Pelimoottoreiden kuvaileminen oli tärkeää, jotta lukija ymmärtää, mitä niillä voi tehdä ja kuinka peliä ei tarvitse itse rakentaa alusta asti. Toiminnallisuuden kuvailemisessa kerrottiin, kuinka VR-projekti voidaan aloittaa työkalujen, kuten OpenXR:n ja XR Interaction Toolkitin avulla, sekä miten tämän projektin VR:lle erityiset ratkaisut tehtiin. Projektin kannalta voidaan myös huomata, kuinka vähän koodin kirjoittamista pitää tehdä, sillä Unity ja työkalut hoitavat suuren osan rakennuksesta. Suurin osa ohjelmoinnista kului pisteytysjärjestelmän ja käyttöliittymän rakentamisessa, sillä lähes kaiken muun pystyy tekemään Unity-näkymässä komponenttien ja valmiiden asettien avulla. Helpoin tapa havainnollistaa kaiken edellisen, oli mielestäni rakentaa oma VR-peli. Samalla pystyttiin hieman esittelemään, kuinka Unityn pelimoottori toimii.

### 5.1 Opinnäytetyön ja pelin onnistuminen

Mielestäni onnistuin rakentamaan hyvän pelin, jonka avulla voitiin esitellä VR-pelinkehityksen eri osia, kuten interaktioita ja optimointia. Opinnäytetyön kirjoittaminen oli myös helppoa pelin kannalta, sillä pystyttiin vain kertoa mitä pelin rakennuksen aikana tehtiin. Tietoperustan kirjoittamisessa auttoi entinen kokemus ja taito virtuaalisesta todellisuudesta ja Unitysta, joka myös helpotti jakamaan aiheet ja pohtimaan mitä niistä on tärkeää kertoa. Lähteiden kannalta oli hyvä katsoa ja varmistaa, että ne olivat ajan tasalla, sillä VR-tekniologia kasvaa ja muuttuu koko ajan. Tämän takia monet lähteet ja tietoperusta olivat suurimmin osin netistä. Virtuaalisen todellisuuden lähteistä katsottiin myös, että ne olivat mahdollisimman uusia, ja ne tulevat mahdollisimman luotettavista paikoista, kun taas Unityn lähteet pystyivät olemaan hieman vanhempia, sillä Unity on ollut markkinoilla jo vuodesta 2005.

Ennen opinnäytetyön aloitusta en ollut kokenut kunnollista pelin julkaisuprosessia, joten tahdoin saattaa kokonaisen pelin loppuun asti, jotta pystyisin harjoittelemaan optimointia ja kokonaisen pelin kokoamista yhteen pakettiin. Pääsin oppimaan enemmän Profiler-ikkunan käyttöä, jonka avulla pystyttiin tarkasti näkemään, mitkä pelin osuudet hidastivat pelin tehokkuutta. Pääsin myös pohtimaan, kuinka pelistä tehdään suoraviivainen ja helpokäyttöinen, jotta käyttökokemus olisi miellyttävä. Yritin vähentää liikepahoinvointia jokaisessa kohdassa rakentaessani peliä, vaikka en itse siitä kärsi. Liikepahoinvoinnin mielessä pitäminen on erityisen tärkeää, varsinkin simulaattoreissa ja muissa peleissä, jotka

ovat tarkoitettu käytettäväksi kokemattomien ihmisten kanssa, sillä he kärsivät siitä kaikista eniten. Huono kokemus voi täysin estää heitä enää koskemasta VR-laitteisiin tämän jälkeen.

Opinnäytetyön kirjoittamista olisin voinut aloittaa aikaisemmin, jotta alussa olisi ollut enemmän näytettävää, mutta yritin saada pelin hyvälle mallille ennen sitä. Kommunikointi ohjaajan kanssa hieman kärsi kesän takia, mutta lopulta opinnäytetyö kuitenkin saatiin tehtyä aika hyvin aikataulun mukaan. Itse pelin kannalta ei ollut suurempia ongelmia. Konseptin ja pelin suunnittelu onnistui hyvin kirjoittamalla alas ideoita ja eri rakennusvaiheiden tarpeita. Pelin rakentamista nopeutti heti alussa projektissa käytetyt työkalut, assetit ja paketit, jotka asettivat vakaan pohjan koko pelin kehitykselle. Kun törmäsin esteeseen pelin teon aikana, niin Unityn keskustelupalstoilta ja muista lähteistä oli helppo löytää ratkaisu ongelmaan. Toiminnallisuuden kannalta peli onnistuu simuloimaan aseiden käytön perusteet, sekä lisäksi voidaan seurata harjoittelun edistymistä ratojen pisteytysjärjestelmien avulla. Peliä voitaisiin hyödyntää, esimerkiksi ennen ammuntaradalle menoa, jotta osataan aseiden perustaidot, tai peliä voidaan pelkästään käyttää viihdekäyttöön peliaspektin ansiosta. En kuitenkaan usko, että peli on täysin valmiina kunnon käyttöön, sillä pelistä puuttuu tärkeitä toiminnallisuuksia. Tämän lisäksi virtuaalisessa todellisuudessa on muutenkin hankalaa simuloida oikean aseiden rekyyliä ja sen hallintaa todellisesti, sillä pelin aseet muistuttavat vain yhtä asemallia, mikä tarkoittaa, että peliin pitää lisätä enemmän aseita, sillä eri asemallit toimivat eri tavoin. Näiden syiden takia pelin avulla ei missään nimessä voida oppia kaikkea aseiden käsittelyyn kuuluvaa, joten pelin lisäksi on hyvä olla ohjaaja, joka valvoo ja opettaa loput aseiden käsittelyn perusteet tarvittaessa.

Tällä hetkellä ei ole tarkoituksena jatkokehittää peliä, mutta jatkokehittämällä peliä voitaisiin saada aseeseen enemmän tarpeellisia ominaisuuksia, kuten vaihtaminen varmistimen, sarjatulen ja yhden panoksen ampumisen välillä. Pelaajalle pitäisi myös lisätä taskut aseille ja panoksille, joka voitaisiin tehdä XR Interaction Toolkitin socket-komponenttien avulla. Peliin kuuluisi myös lisätä eri aseita käyttäjien tarpeiden mukaan, esimerkiksi jos Suomen puolustusvoimat jostain ihmeen syystä ottaisivat pelin käyttöön, niin pelissä olisi hyvä olla RK 95, PKM jne. Moninpelin ja eri pelimuotojen lisääminen peliin olisi erittäin kätevää myös tällöin, jotta pelin avulla voitaisiin harjoitella taktiikoita ja taistelua rakennetulla alueella. Samalla peliin saataisiin toistoarvoa, sillä peli on tällä hetkellä aika yksitoikkoinen. Näistä ideoista lähtisin ensiksi tekemään jatkokehitystä, ja tämän lisäksi netistä voidaan ostaa 3D-mallinnettu aseiden runko, johon voidaan yhdistää ohjaimet, mikä voisi auttaa nostamaan pelin immersiota. Jos peliä haluttaisiin jatkokehittää kaupalliseen käyttöön, niin pitäisi tarkistaa tarkemmin valmiiden assettien lisensointi, sillä useat ilmaiset assetit usein estävät niiden käytön kaupalliseen tarkoitukseen.

## 5.2 VR-pelin rakennus Unitylla

Unity-kehityksen aloittaminen on mielestäni helppoa varsinkin, jos henkilö omaa aikaisempia pelinkehitys taitoja. Unity käyttää C#-ohjelmointikieltä, joka on erittäin helppo oppia varsinkin olio-ohjelmoinnin osajille, ja sen koodia on muutenkin helppo ymmärtää. Unity ja C# ovat molemmat erittäin helppokäyttöisiä, joten vauhtiin pääsee nopeasti eri kurssien avulla. Uudet henkilöt voivat aloittaa ensin opettelemalla Unityn perusteet, joita löytyy joka puolelta nettiä. Unityn nettisivuilta voidaan suorittaa ilmaisia peruskursseja, joiden jälkeen henkilö saa hyvän käsityksen useasta pelinkehitysvaiheesta. Unityn näkymän käyttäminen on muutenkin intuitiivista, joten sen opetteleminen on todella helppoa ja nopeaa. Youtube sisältää suuren määrän perusteellisesti läpi käytyä videoita Unitylla kehittämistä, jonka kautta voidaan oppia tarkemmin, kuinka tietyt toiminnallisuudet tai järjestelmät tehdään. Aloittelijoiden on mielestäni hyvä käyttää ilmaisia assetteja projekteissaan, jotta he voivat keskittyä tärkeämpiin asioihin. Itse huomasin aloittaessani, että jos osaa olio-ohjelmointia, niin on järkevämpää aloittaa Unityn opettelu, tekemällä yksinkertaisia projekteja mahdollisimman aikaisin, sillä Unity-näkymän käyttö on kuitenkin suurin osa kehityksessä.

Ennen VR-kehitystä on hyvä osata Unityn perusteet, jotta ymmärtää, miten työkalut ja niiden käyttö onnistuu. Unity tarjoaa myös ilmaisen VR-kurssin, jonka avulla opitaan rakentamaan VR-projekti Unitylla, sekä kuinka tehdään yksinkertaisia VR-ratkaisuja, esimerkiksi liikkumiselle ja esineiden nostamiselle. VR-kehitystä helpottaa valmiiden työkalujen ja liitäntäisten suuri määrä Unityssa ja Asset Storessa. Kaikista tärkein pelin jakoa varten on tietenkin OpenXR, sillä se on erittäin kätevä, jotta käyttäjän ei itse tarvitse yrittää tavoitella kaikkia VR-laitteita. XR Interaction Toolkit antaa hyvän alun pelaaja-peliobjektin rakentamiseksi, joten pelin muita osia voi alkaa tekemään melkein heti, sillä käytettävän pelaaja-peliobjektin, sekä sen interaktiokomponentit saa valmiiksi todella nopeasti. VR-kehitys saattaa olla hankalampaa kuin normaali pelinkehitys, sillä pelaaja pystyy rikkomaan pelin helpommin, koska hänellä on mahdollisuus liikkua eri paikkoihin ja koskea eri esineisiin. Tämän takia pitää miettiä tarkemmin ominaisuuksia, joita peliin halutaan, sekä kehittäjän täytyy olla valmiina korjaamaan niistä johtuvat ongelmat. Samalla pitää miettiä, kuinka pelin fysiikat tehdään, esimerkiksi voiko pelaajan käsi mennä pöydän läpi ja pystyykö pelaaja kiipeämään huonekalujen päälle. Pelissä pitää myös ottaa huomioon sen suorituskyky, sekä kuinka paljon liikepahoinvointia se saattaa aiheuttaa. Näiden syiden takia monet pelit ovat tähän mennessä rajoittaneet pelaajan liikettä suuresti, jotta kehittäjille tulisi vähemmän työtä. Itse käytin XR Interaction Toolkitiä, jonka avulla pystyttiin välttämään tiettyjä ongelmia, sillä niiden ratkaisut tulevat paketin mukana.

## Lähteet

Anumadu, M. 25.9.2020. Why reinvent the wheel? These 5 mobile game engines can give you a head start. LogRocket. Luettavissa: <https://blog.logrocket.com/why-reinvent-the-wheel-these-5-mobile-game-engines-can-give-you-a-head-start/>. Luettu: 15.7.2021.

Bardi, J. 2021. What is Virtual Reality? [Definition and Examples]. Marxent. Luettavissa: <https://www.marxentlabs.com/what-is-virtual-reality/>. Luettu: 10.7.2021.

Barnard, D. 2019. History of VR - Timeline of Events and Tech Development. VirtualSpeech. Luettavissa: <https://virtualspeech.com/blog/history-of-vr>. Luettu: 10.7.2021.

Barnett, J. 28.4.2021. How to Make Oculus Quest 2 Games // Introduction to Unity VR Development. Youtube-video. Katsottavissa: <https://www.youtube.com/watch?v=wnn-dzHz-tA>. Katsottu: 10.9.2021.

BHaptics. 2021. BHaptics TactSuit X40. Luettavissa: <https://www.bhaptics.com/shop/tactsuit-x40>. Luettu: 15.7.2021.

Brockwell, H. 2016. Forgotten genius: the man who made a working VR machine in 1957. TechRadar. Luettavissa: <https://www.techradar.com/news/wearables/forgotten-genius-the-man-who-made-a-working-vr-machine-in-1957-1318253>. Luettu: 13.7.2021.

Carter, R. 2021. Varjo VR-3 Review: The Highest Resolution VR Device? XRToday. Luettavissa: <https://www.xrtoday.com/virtual-reality/varjo-vr-3-review-the-highest-resolution-vr-device/>. Luettu: 14.7.2021.

Clark, T. 2014. How Palmer Luckey Created Oculus Rift. Smithsonian. Luettavissa: <https://www.smithsonianmag.com/innovation/how-palmer-luckey-created-oculus-rift-180953049/?page=2>. Luettu: 13.7.2021.

Gamescrye. 3.10.2016. What is a Game Engine? Luettavissa: <https://gamescrye.com/blog/what-is-a-game-engine/>. Luettu: 15.7.2021.

Gibson, J. 2014. Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#. Addison Wesley.



Graham, P. 2020. Varjo Drops Cost And Ups Features For New XR-3 And VR-3 Headsets. VRFocus. Luettavissa: <https://www.vrfocus.com/2020/12/varjo-drops-cost-and-ups-features-for-new-xr-3-and-vr-3-headsets/>. Luettu: 19.8.2021.

Haas, J. 2014. A History of the Unity Game Engine. Luettavissa: [https://web.wpi.edu/Pubs/E-project/Available/E-project-030614-143124/unrestricted/Haas\\_IQP\\_Final.pdf](https://web.wpi.edu/Pubs/E-project/Available/E-project-030614-143124/unrestricted/Haas_IQP_Final.pdf). Luettu: 15.7.2021.

Heaney, D. 2021. Facebook Headsets Now Make Up 60% Of SteamVR Use. UploadVR. Luettavissa: <https://uploadvr.com/oculus-headsets-60-percent-steamvr/>. Luettu: 14.7.2021.

Ilett, D. 16.5.2020. Unity Tips | Part 6 – Prefabs. Daniel Ilett. Luettavissa: <https://danielilett.com/2020-05-16-unity-tips-6-prefabs/>. Luettu: 10.9.2021.

Juegoadmin. 21.1.2021. 7 Ways to Keep Unity Project Organized: Unity3d Best Practices. Juego Studios. Luettavissa: <https://www.juegostudio.com/blog/7-ways-to-keep-unity-project-organized-unity3d-best-practices>. Luettu: 15.9.2021.

Khronos. 2021. OpenXR. Luettavissa: <https://docs.unity3d.com/Packages/com.unity.xr.openxr@0.1/manual/index.html>. Luettu: 15.7.2021.

Lang, B. 2021a. [Updated] VR Headsets on Steam Hit Record High Three Months Running. Road To VR. Luettavissa: <https://www.roadtovr.com/monthly-connected-headsets-steam-3-million-march-2021/>. Luettu: 13.7.2021.

Lang, B. 2021b. Quest 2 May Be Selling Thrice as Fast as the Best Selling VR Headset, PSVR. Road To VR. Luettavissa: <https://www.roadtovr.com/quest-2-unit-sales-estimate-psvr-unit-sales/>. Luettu: 15.7.2021.

Lowood, E. 2020. Virtual Reality. Encyclopedia Britannica. Luettavissa: <https://www.britannica.com/technology/virtual-reality>. Luettu: 13.7.2021.

Grand View Research. 2021. Virtual Reality Market Size, Share & Trends Analysis Report By Technology (Semi & Fully Immersive, Non-immersive), By Device (HMD, GTD), By Component (Hardware, Software), By Application, And Segment Forecasts, 2021 – 2028. Luettavissa: <https://www.grandviewresearch.com/industry-analysis/virtual-reality-vr-market>. Luettu: 15.7.2021.

Oculus. 2021. Oculus Quest 2. Luettavissa: [https://www.oculus.com/quest-2/?locale=fi\\_FI](https://www.oculus.com/quest-2/?locale=fi_FI).  
Luettu: 15.7.2021.

Peckham, E. 2019. How Unity built the world's most popular game engine. Techcrunch.  
Luettavissa: <https://techcrunch.com/2019/10/17/how-unity-built-the-worlds-most-popular-game-engine/>. Luettu: 15.7.2021.

Perkins Coie. 2020. 2020 Augmented and Virtual Reality Survey Report. Luettavissa:  
<https://www.perkinscoie.com/images/content/2/3/v4/231654/2020-AR-VR-Survey-v3.pdf>.  
Luettu: 14.7.2021.

Petty, J. 2019. What is Unity 3D & What is it Used For? Concept Art Empire. Luettavissa:  
<https://conceptartempire.com/what-is-unity/>. Luettu: 15.7.2021.

Riley, S. 2017. Google Cardboard Review: Better Than Nothing. Tom's Guide. Luettavissa:  
<https://www.tomsguide.com/uk/us/google-cardboard,review-4207.html>. Luettu:  
14.7.2021.

Sinicki, A. 2021. What is Unity? Everything you need to know. Android Authority. Luettavissa:  
<https://www.androidauthority.com/what-is-unity-1131558/>. Luettu: 15.7.2021.

Skarredghost. 28.11.2016. How a developer can reduce motion sickness in VR games. The Ghost Howls. Luettavissa:  
<https://skarredghost.com/2016/11/28/how-a-developer-can-reduce-motion-sickness/>. Luettu: 11.9.2021.

Steam. 2021. Steam Hardware & Software Survey: July 2021. Steam. Luettavissa:  
<https://store.steampowered.com/hwsurvey>. Luettu: 19.8.2021.

ThrillSeeker. 24.2.2020. How Virtual Reality Became a Reality. Youtube-video. Katsottavissa:  
<https://www.youtube.com/watch?v=UW8NapTA9oc>. Katsottu: 13.7.2021.

Trescak, A. 2019. The main problem with virtual reality? It's almost as humdrum as real life. The Conversation. Luettavissa:  
<https://theconversation.com/the-main-problem-with-virtual-reality-its-almost-as-humdrum-as-real-life-126761>. Luettu: 14.7.2021.

Unity. 2021a. Welcome to Unity. Luettavissa: <https://unity.com/our-company>. Luettu: 15.7.2021.

Unity. 2021b. Universal Render Pipeline overview. Luettavissa: <https://docs.unity3d.com/Packages/com.unity.render-pipelines.universal@11.0/manual/index.html>. Luettu: 10.9.2021.

Unity. 2021c. XR Interaction Toolkit. Luettavissa: <https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@1.0/manual/index.html>. Luettu: 15.7.2021.

Unity s.a. Quick guide to the Unity Asset Store. Luettavissa: <https://unity3d.com/quick-guide-to-unity-asset-store>. Luettu: 10.9.2021.

Varjo. 2021. A Guide to the Transformative Power of Virtual Reality and Mixed Reality. Luettavissa: <https://varjo.com/e-book/guide-to-the-transformative-power-of-virtual-reality-and-mixed-reality/>. Luettu: 10.9.2021.

Wikipedia 2021. Unity Technologies. Luettavissa: [https://en.wikipedia.org/wiki/Unity\\_Technologies](https://en.wikipedia.org/wiki/Unity_Technologies). Luettu: 15.7.2021.

## Liitteet

### Liite 1. Pelin sivu Itch.io-alustalla

#### Target Practice Game

This game is made by Timo Heikkilä as a thesis project for Haaga-Helia university of Applied Sciences. The purpose of the project is to demonstrate VR-specific game development process with Unity and its VR capabilities.

**What is this game about?**

In this simple virtual reality game you are able to practice shooting targets on a shorter range with a pistol and on a longer range with an assault rifle. After practicing the mechanics, you'll be able to head over to a course, which you'll try to complete as fast as possible while gathering as many points as you can.

**Controls**

- Movement - Thumbsticks
- Shoot - Trigger
- Grab - Grip
- Magazine release - B
- UI - X

**Assets**

Most objects are from [VR Interaction Framework](#) by Bearded Ninja Games.

[Wooden Floor Materials](#) by Casual2D.

[18 High Resolution Wall Textures](#) by A dog's life software.

[High Quality Bricks & Walls](#) by Alessio Regalbuto.

[More information](#) ~

**Install instructions**

Download the game files, extract the zip file and run TargetPracticeGame.exe

**Download**

[Download](#) TargetPracticeGame.zip 132 MB

itch.io · Virtual by TimoHeikkila · Report · Embed · Updated 7 days ago

Games · Shooter · Free

[Follow TimoHeikkila](#)

[Add to Collection](#)

