



TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND
Tartu kolledž

RÕUGE ÖÖBIKUORU VAATETORNI VIRTUAALTUURI RAKENDUSE ARENDUS

RAKENDUSKÕRGHARIDUSE TEADUSTÖÖ

Üliõpilane: Mihkel Reiljan
/nimi/

Üliõpilaskood: 178347EDTR

Juhendaja: Vladimir Kuts, Teadur
/nimi, amet/

AUTORIDEKLARATSIOON

Kinnitan, et olen koostanud lõputöö iseseisvalt. Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneridiplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

Autor: Mihkel Reiljan
11.01.2020

Töö vastab bakalaureusetöö/magistritööle esitatud nõuetele

11.01 2021.

Juhendaja: Vladimir Kuts
/ allkirjastatud digitaalselt /

Kaitsmisele lubatud

".....".....2021.

Kaitsmiskomisjoni esimees

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina Mihkel Reiljan (*autori nimi*) (sünnikuupäev: 21.10.1997)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose Rõuge Ööbikuoru vaatetorni virtuaaltuuri rakenduse arendus

mille juhendaja on Vladimir Kuts,

1.1 reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2 üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

¹*Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil.*

Allkirjastatult digitaalselt (*allkiri*)

11.01.2021 (*kuupäev*)

TalTech Tartu Kolledž

LÕPUTÖÖ ÜLESANNE

Üliõpilane: Mihkel Reiljan, 178347EDTR (nimi, üliõpilaskood)

Õppekava, peeriala: EDTR 17/17 Küberfüüsikalised süsteemid (kood ja nimetus)

Juhendaja(d): Vladimir Kuts, Teadur, vladimir.kuts@taltech.ee

Lõputöö teema:

(eesti keeles) Rõuge Ööbikuoru vaatetorni virtuaaltuuri rakenduse arendus.

(inglise keeles) Virtual tour application development of the observation tower at Rõuge Ööbikuorg.

Lõputöö põhieesmärgid:

1. Luua keskkond mille sees kasutaja viibib kasutades punktipilvi ja 3D mudeleid.
2. Luua funktsionaalsus ilma virtuaalreaalsuse prillideta, mille abil saaks torni vaadelda.
3. Luua virtuaalreaalsuse funktsionaalsus, mille abil saaks keskkonnas ringi liikuda ning torni vaadelda.

Lõputöö etapid ja ajakava:

Nr	Ülesande kirjeldus	Tähtaeg
1.	Punktipilvede töötlemine punktipilve programmide abil Unity tarkvaras kasutamiseks	24. sept
2.	Andmete põhjal keskkonna arendamine ja kohandamine reaalse maastiku alusel	20. okt
3.	Torni vaatlemine hiire ja klaviatuuri abil	15. nov
4.	Virtuaalreaalsuse funktsionaalsuse loomine	1. dets

Töö keel: Eesti keel **Lõputöö esitamise tähtaeg:** 11.01.2021 a

Üliõpilane: Mihkel Reiljan Allkirjastatult digitaalselt 11.01.2021 a
/allkiri/

Juhendaja: Vladimir Kuts Allkirjastatult digitaalselt 11.01. 2021 a
/allkiri/

Programmijuht: Helle Hallik ".....".....2021 a
/allkiri/

Kinnise kaitsmise ja/või lõputöö avalikustamise piirangu tingimused formuleeritakse pöördel

EESSÕNA

Käesoleva lõputöö teema sai valitud seetõttu, et kaastudeng andis teada Rõuge valla arengukava koosolekutel arutluse all olnud plaanist Ööbikuoru vaatetornile ja sealsele ümbruskonnale luua virtuaaltuur. Vaatetornile lisaks on vallal idee näidata ka muinasmaja ja orgu ennast. Tudengid leidsid et tornist mudeli ja virtuaaltuuri tegemine oleks hea katsetus selleks, et teda saada kui teostatav see projekt on. Kui kunagi hakatakse vaatetornist virtuaaltuuri projekti teostama, oleks ehk meie tehtud tööd head sisendid tulevastele projektidele. Lõputööd viitaksid ka sellele, mida saaks tulevikus paremini teha, millistest vigadest hoiduda ja milline oleks sellise töö maht.

Põhilised algandmed on kogutud maa-ameti veebilehelt kõrgusandmete sektsioonist [1] kasutades x-gis kaardistiku [2], Rõuge Ööbikuoru vaatetorni platsi pealt droonifotodest ning kaastudengi käest saadud mudelist.

LÜHENDITE JA TÄHISTE LOETELU

CSF – (inglise k. Cloth Simulation Filter)

GPL – (inglise k. GNU General Public License)

HDRP – (inglise k. High-Definition Render Pipeline) ehk kõrglahutuse esitamise töötuselementide jada mis paneb rõhku graafilisele kvaliteedile.

LiDAR – (inglise k. Light Detection and Ranging) on meetod kauguste mõõtmiseks kasutades laserkiiri.

Mesh – 3d modelleerimises on mesh mingi objekt mille defineerivad punktid, servad ja pinnad.

SOR Filter – (inglise k. Statistical Outlier Removal)

Steam – Steam on digitaalne mängude platvorm, antud töös kasutatud virtuaalreaalsuse funktsionaalsuse tagamiseks.

V – (Universal Render Pipeline) ehk universaalne esitamise töötuselementide jada.

VR – Virtuaalreaalsus(inglise k. Virtual Reality).

VR-less - ilma Virtuaalreaalsuse komponendita.

XR – Laiendatud reaalsus(inglise k. Extended Reality), kõik kombineeritud päris ning virtuaal keskkonnad ja inimese ning masina vahelised koostoimed.

SISUKORD

EESSÕNA	5
LÜHENDITE JA TÄHISTE LOETELU.....	6
SISUKORD	7
SISSEJUHATUS	8
JOONISTE LOETELU	9
TABELITE LOETELU	10
1. VIRTUAALTUURID	11
1.1. Virtuaalreaalsuse populaarsus tänapäeval	11
1.2. Virtuaalreaalsuse eelised ja puudused.....	12
1.3. Virtuaalreaalsuse populaarsus.....	12
2. MÄNGUMOOTORID	14
3. VIRUTAALTUURI LOOMINE UNITY KESKKONNAS	15
3.1. Unity mängumootori valimine	15
3.2. Unity VR funktsionaalsus.....	15
3.3. Tähtsamad kasutatud elemendid/komponendid Unity tarkvaras	16
4. PUNKTIPILVED JA MESH MUDELID	20
4.1. Lastools ja las2las tööriist	20
4.2. CloudCompare	20
4.3. Agisoft Metashape	22
5. TÖÖKÄIK.....	24
5.1. Punktipilvede töötlemine	24
5.1.1. Tekkinud probleemid pilvede töötlemisel	27
5.2. Unity VR-less stseen	27
5.3. Tornitulede juhtimine skriptide abil.....	28
5.4. Kasutatud skriptid	29
5.5. Unity VR stseen	29
5.5.1. Probleemid VR implementeerimisega.....	30
5.5.1.1. Teleporteerumine.....	30
5.5.1.2. Menüü ümberehitus	31
6. VÄLJA KUJUNENUD RAKENDUS	33
6.1. VR-Less stseen	33
6.2. VR stseen.....	35
7. TÖÖ ANALÜÜS.....	36
8. MAKSUMUSE ANALÜÜS	37
KOKKUVÕTE.....	38
SUMMARY	39
KASUTATUD KIRJANDUSE LOETELU	40
LISAD	43

SISSEJUHATUS

Antud töö on tehtud Tallinna Tehnikaülikooli Tartu kolledži küberfüüsikaliste süsteemide eriala rakenduskõrghariduse valdkonnas mis on valminud aastal 2021.

Lõputöö eesmärgiks on luua virtuaaltuur Rõuge Ööbikuoru vaatetornist. Loomise käigus uurida millised võimalused on Unity mängumootoris võrreldes rohkem klassikalisema viisiga, kus kasutatakse kas videomaterjali või 360 kraadilisi pilte. Nagu ka eessõnas öeldud sai, siis tudengid said idee Rõuge vallalt. Teema kõlas huvitava ja vajalikuna, millest saaks luua midagi, mis on kultuuriselt märkimisväärne ja mida saaks digitaalselt kolmemõõtmeliselt jäädvustada. Kuna plaanis oli saada kaastudengi käest 3D mudel, mis oli tehtud punktipilvest, siis otsustas tudeng kasutada punktipilvi ka oma töös just keskkonna kujutlemiseks.

Käesolev töö jagunes nelja etappi. Esimeseks etapiks oli saada kätte punktipilved millega dekoreerida stseenid kus torn hakkab asuma. Lisaks sellele oli vajalik punktipilvedele korrekture teha kasutades töötlusprogramme. Programmideks olid CloudCompare ja Lastools. Teiseks etapiks kasutati töös Unity mängumootorit. Antud tarkvara abil luuakse 2D, 3D ja laiendatud reaalsuse (XR) mängu ning elamusi. Programm ei piirdu ainult sellega, Unity tarkvara kasutatakse palju ka teistes valdkondades nagu tööstus ja filmindus. Tarkvara abil oli võimalik kokku panna kogu projekt mängitavaks rakenduseks, mida on võimalik avada igas arvutis, mis on piisavalt võimekas. Unity tarkvarasse tõsteti kõik punktipilved, mida stseenide arendamiseks vaja läks ning ka maapind ja kaastudengi poolt saadud torni mudel. Kolmandaks etapiks tuli luua stseen, mis töötab hiirt ja klaviatuuri kasutades, ilma virtuaalreaalsuseta. Tänu sellele saab kasutaja, kellel pole virtuaalreaalsuse (VR) seadmeid, vaadelda torni. Kolmanda etapi tulemusel on rakendusse lisatud ka valgustuse värvide muutmise süsteem. Seda seetõttu, et Rõugel on omalaadne traditsioon- iga uue vallakodaniku sünni puhul värvub torni kuppel ehk „kuldmunna“ õhtuhämaruses kas siniseks või roosaks, vastavalt sündinud ilmakodaniku soole. Kui vallas on sündinud tüdruk, siis lähivad põlema roosat värvi projektorid ja kaugusest on näha, et kuppel on roosa. Kui aga vallas on sündinud poiss siis on kaugusest näha sinist värvi kuplit. Viimaseks etapiks oli virtuaalreaalsuse funktsionaalsuse loomine. Selleks on kasutatud SteamVR pistikprogrammi, mille abil läbi Steami kasutajaliidese tekib võimalus kasutada VR peaseadet stseenis ringi liikumiseks ning avastamiseks.

JOONISTE LOETELU

Joonis 1. Ühendatud VR peaseadete arv Steam platvormil viimase viie aasta jooksul [9].....	13
Joonis 4.Varade asukoht Unity programmis.....	16
Joonis 2.Näidis kaamera mänguobjektist vasakul ning selle sätetest ülevaate menüüs paremal.....	17
Joonis 3.Näide C# skriptist mis paigutatakse mänguobjekti	18
Joonis 5. Stseenivaade Unity tarkvaras	18
Joonis 6. Tööriistade kujutatud vooskeemina (vt. Lisa 6	24
Joonis 7. Lastools tarkvara kasutamine	25
Joonis 8. Pildid enne (üleval) ja pärast (all) SOR Filtri kasutamist	26
Joonis 9. Näide tornil olevatest tuledest ja nende muutusest.....	28
Joonis 10. Paremal pool on näide teisest objektist ja vasakul pool esimesest.....	31
Joonis 11. Näidis nooltest mille abil saab asukohta muuta.....	32
Joonis 12. Rakenduse valikumenüü.....	33
Joonis 13. Avavaade vajutades Desktop nupule	33
Joonis 14. Päevase ja öise menüü valikud.....	34
Joonis 15. Päevane vaade.....	34
Joonis 16. Öine vaade	35
Joonis 17. Nupud värvide ja aja muutmiseks VR stseenis.....	35

TABELITE LOETELU

Tabel 1. Mõned virtuaalreaalsuse plussid ja miinused [7],[8]12

Tabel 2. 10 populaarsemat mängumootorit [10].....14

1. VIRTUAALTUURID

Virtuaaltuur on simulatsioon pärismaailmas olemas olevast kohast, mis tavaliselt koosneb kas videoklippidest, 360 kraadistest piltidest või mängudena mis on virtuaalreaalsus prillidega vaadeldavad. Samuti võidakse kasutada multimeedia mõjusid nagu heli, muusika, jutustamine ja tekst. [3] Mõned tuurid on saadaval brauseris ning mõned on alla laaditavad kas läbi rakenduse või otse arvutisse. [3] Antud töös on tehtud alla laaditav versioon virtuaaltuurist, mida saab kas lihtsalt arvutis või VR peaseadme abil vaadelda.

Tuure luuakse ka 3D skaneerimise tehnoloogiate abil. Tehnoloogiat, mida kasutati enamasti kinnisvara sektoris, kasutatakse nüüd ka ajalooliselt tähtsate kohtade või objektide digitaalseks säilitamiseks. Näiteks hetkel on võimalik virtuaalselt külastada JFK varjendi muuseumit, mis sulges oma ukseid aastal 2017. [4]

Virtuaalselt on võimalik teha huvitavaid ja silmaringi avardavaid tuure, mis on muidu aega ning raha nõudvad. Tekib ka võimalus näidata kohti, mis on raskesti ligipääsetavad. Näiteks võib tuua kosmosejaama[5], aktiivsete vulkaanide ning polaarpiirkondade tuurid. Paljud firmad teevad virtuaaltuure selleks, et teha endale reklaami. Ostjatele näidatakse tooteid või asukohti sellisel moel, et ostja saaks parema ettekujutluse sellest, mida ta ostma hakkab. [3]

1.1. Virtuaalreaalsuse populaarsus tänapäeval

Aprillis on ülemaailmne maapäev (inglise k. Earth day) ning inimesed, kes seda tähistavad, kogunevad tavaliselt erinevates kohtades. Aastal 2020 muutus aga pandeemia ajal väljas käimine keeruliseks ning firmad hakkasid rohkem toetuma virtuaaltuuridele. Seda peetakse sobivaks võimaluseks kogeda tehnoloogiat, mida mõned eksperdid peavad ökosõbralikuks lahenduseks turismindusele. [6]

Kuna enamus maailmast on reisiipiirangute all siis paljud on pöördunud virtuaalreaalsuse poole otsimaks kogemusi, mis viiks neid maailma erinevatesse paikadesse. Näiteks võib tuua Machu Picchu ja Galápagose saared. Selle tehnoloogia potentsiaal tuli nähtavale siis kui inimesed jäid mitmeteks nädalateks koju. Sellega seoses paljudes kohtades langes CO2 tase drastiliselt. Paljud veekogud muutusid oluliselt puhtamaks ja isegi loomad hakkasid linnades julgelt ringi liikuma. [6]

Virtuaalruumide trend on viimasel ajal olnud tõusuteel. Valeriy Kondruk, VR reisimise firma Ascape tegevdirektor on öelnud, et nende rakenduse alla laadimiste arv kasvas eelmise aasta detsembrist kuni aprillini 60%. Kuid hetkel pole veel teada kas virtuaalreaalsuse populaarsus jääb ka samaks peale pandeemia lõppu. [6]

1.2. Virtuaalreaalsuse eelised ja puudused

Mõningate inimeste arvates ei ole virtuaalreaalsuse tehnoloogia veel valmis pakkuma tõsiselt elamusi kuna 360 kraadiseid virtuaalreaalsuse videoid vaadatakse tavaliselt läbi VR prillide nagu näiteks Oculus Rift, Oculus Quest või mõne muu VR peaseadmega. Lisaks sellele saab vaadata ka kasutades telefoni ning Google Cardboardi, mis on nendest valikutest autori arvates kõige halvemad. Öeldakse, et VR prillid on kallid, rasked, võivad tekitada iiveldust ja pole mugavad kanda rohkem kui 30 minutit. [6]

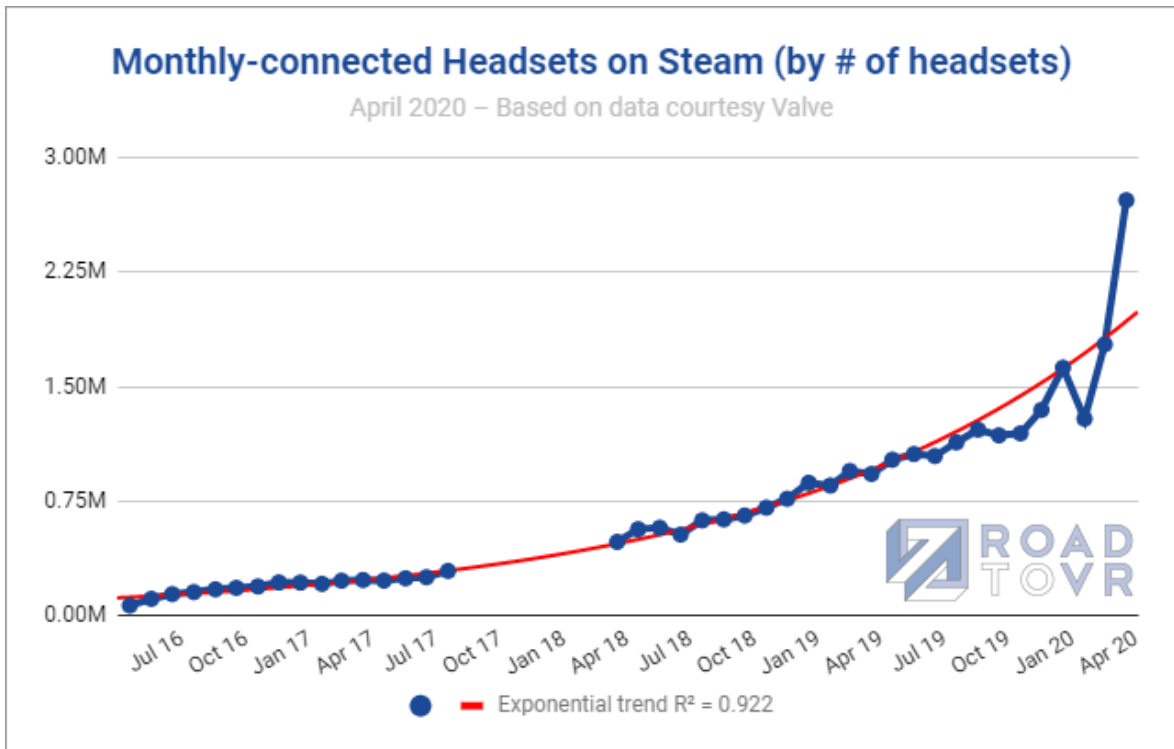
Kui inimene saab ise ringi liikuda ja otsustada kuhu ta soovib liikuda on tuur palju kaasahaaravam ja huvitavam ning vähem iiveldust tekitav.

Tabel 1. Mõned virtuaalreaalsuse plussid ja miinused [7],[8]

Eelised	Puudused
Võib aidata inimestel paljude foobiate vastu võidelda	Võib merehaigust ja iiveldust tekitada
Tänapäeva lahendused odavamad	Tööriist, mis ei lase veel kõigi meeltega ümbruskonda kogeda ning on palju ruumi vajav
Tähelepanu haaravam võrreldes ainult arvutit kasutades	Pole konkreetset reisi jagamise võimalust

1.3. Virtuaalreaalsuse populaarsus

Viimaste aastate jooksul on ka suurenenud VR kasutajate arv ülemaailmselt, mis avab uued võimalused virtuaalruumide arendamiseks. Hetkel on väga palju erinevaid firmasid, mis toodavad erinevaid peaseadmeid mis on mõeldud erinevateks kasutusteks. Edaspidi on oodata väga palju uut ja põnevat virtuaalreaalsuse maailmast.



Joonis 1. Ühendatud VR peaseadete arv Steam platvormil viimase viie aasta jooksul [9]

Ülaltoodud joonisel on näha, et VR peaseadete arv on pidevalt kasvamas. Viimane hüpe tulenes uue väga oodatud virtuaalreaalsuse mängu väljalaskest nimega "Half-life: Alyx". Antud VR mäng lisas juurde pea miljon peaseadet ning joonis näitab kui palju peaseadmeid inimesed tegelikult omavad. [9]

2. MÄNGUMOOTORID

Mängumootor on tarkvara arenduskeskkond mis on loodud videomängude, industriaal- ning ka paljude muude rakenduste ja lahenduste loomiseks arvutitele, mängukonsoolidele ning nutitelefonidele. Mängumootorid lubavad arendajatel lihtsal viisil lisada oma mängudele füüsilisi omadusi, sisestusmeetmeid, renderdamise meetodeid, skriptimise võimalusi ja palju muud. Mängumootorid on taaskasutatavad komponendid, mida mänguarendajad saavad kasutada oma mängu raamistiku ehitamiseks. See annab arendajatele rohkem aega mida saab kasutada mängu eripäradele nagu tegelased, tekstuurid, kuidas objektid omavahel suhtlevad ja palju muud. Olemas on palju erinevate eesmärkidega mängumootoreid mida saab iga inimene vabalt kasutada ning arendada just endale sobivat rakendust või mängu. [10]

Tabel 2. 10 populaarsemat mängumootorit [10]

1.	Unreal engine	Üks kõige populaarsemaid mängumootoreid, mille tugevuseks on ainulaadsete mängude loomine
2.	Unity	Võimalus kergelt luua mitme platvormilisi sisukaid 3D elamusi tasuta
3.	GameMaker	Mängumootor ei vaja kasutajapoolset programmeerimisalaseid teadmisi
4.	Godot	Suur valik tööriistu, pole vajadust „uuesti ratast leiutada“
5.	AppGameKit	Kergesti õpitav mängumootor, ideaalne algajatele, harrastajatele ja väikearendajatele
6.	CryEngine	Tasuta, silmatorkav graafiline visuaalsus
7.	Amazon Lumberyard	Tasuta, tugev ja täielikult kohandatav
8.	RPG Maker	Igäüks võib seda 2D mängude loomiseks kasutada
9.	LibGDX	Java arendus platvorm mängude loomiseks
10.	Urho3D	Tasuta, väiksemahuline, mitme platvormiline mängumootor

3. VIRUTAALTUURI LOOMINE UNITY KESKKONNAS

Unity on mitmeplavormiline mängumootor mis väljastati aastal 2005 ning aastast 2018 toetab rohkem kui 20 erinevat platvormi. [11] Tarkvara kasutatakse 2D ning 3D mängude, VR/AR/XR lahenduste, simulatsioonide kuid ka paljude muude rakenduste loomiseks. Unity tarkvara on kasutusele võetud ka filminduses, autotööstuses, arhitektuuri valdkonnas ja nii mõningates teisteski valdkondades. [12]

3.1. Unity mängumootori valimine

Virtuaalтуure on võimalik luua igas keskkonnas, olgu see Google Maps, Google enda Tour Creatoriga [13] või mõne mängumootoriga nagu Unreal Engine. Tööks sai valitud Unity mängumootor kuna tarkvara on üks populaarsemaid ning sellel on palju erinevaid ressursse, mida on võimalik vabalt kasutada parima tulemuse saamiseks. Mõned aastad tagasi võis arvata, et Unreal Engine on parem mängumootor aga hetkel on Unity näidanud, et selle programmiga on võimalik tõhusalt luua samasuguseid imeilusaid stseene ning hea funktsionaalsusega rakendusi. See on ka üks põhjus miks tänapäeval võib leida väga palju inimeste enda poolt tehtud mängu mis on leitavad internetist.

Unity pakub väga palju erinevaid õppemeetmeid ning ressursse nii algajatele kui ka kogenumatele. Unity enda rakenduse programmeerimisliidese dokumentatsioon on väga hästi kokku pandud ning arusaadav koos kasutajasõbralike näidetega mida saab alati ise läbi proovida.

Unity graafiline kasutajaliides on samuti väga kasutajasõbralik ning kompaktne. See annab hea ülevaate stseenis juhtuvast ning seda on väga kerge tundma õppida ja kasutada.

3.2. Unity VR funktsionaalsus

Unity mängumootorile on arendatud mitmeid erinevaid viise virtuaalreaalsuse implementeerimiseks. Sisse on ehitatud XR pistikprogrammide juhtimine (inglise k. XR Plugin Management) mille abil saab kergelt lisada projektile VR funktsionaalsust. Lisaks sellele on olemas ka kolmanda osapoole pistikprogramme nagu näiteks SteamVR mida kasutati ka antud töös. Unity toetab Oculuse, WMR (inglise k. Windows Mixed Reality) ja muid peaseadmeid. Ühel või teisel viisil on võimalus leida toetust enda soovitud peaseadmele. [14]

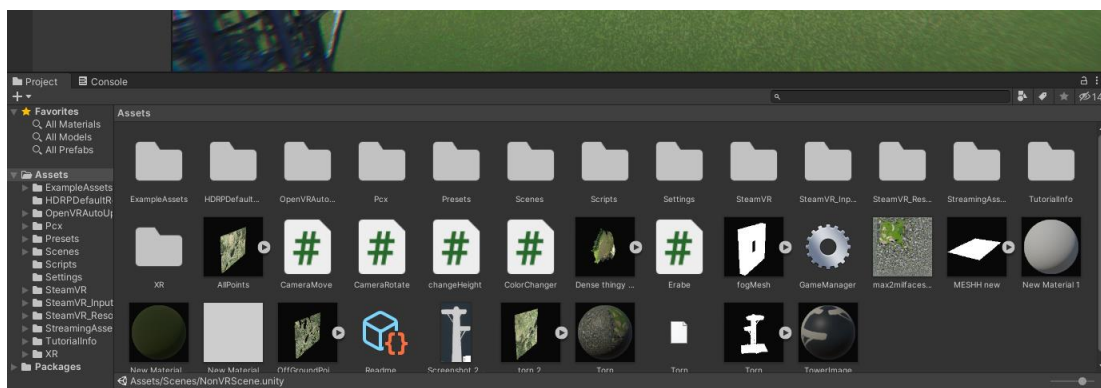
3.3. Tähtsamad kasutatud elemendid/komponendid Unity tarkvaras

Elemendid või teisisõnu komponendid mängivad põhilist rolli mängu või rakenduse töötamises, interaktiivsuses ning eesmärgis, mida see kindel rakendus saavutab. [15] Alapeatükis on välja toodud tähtsamad komponendid mis kuuluvad Unity arenduskeskkonda.

Projekt Unity keskkonnas on kaust või asukoht mis hoiab tervet mängu või rakendust koos kaasatud skriptimise kogudega mida läheb tarvis rakenduse käivitamiseks.[15]

Varade pakid/pakendid(inglise k. Asset Packages) on eelkompileeritud mänguvarade klaster. Unityga on kaasas erinevad pakendid. Pakendit võib märgendada kui zip faili mida saab lahti pakkida Unity keskkonnas.[15]

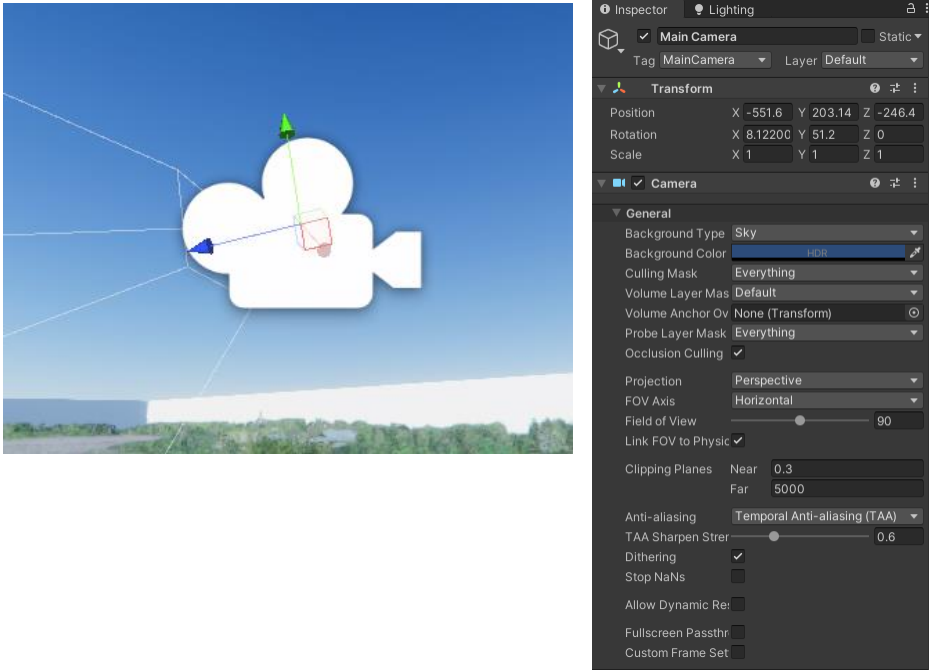
Varad (inglise k. Assets) on objektid mida saab projekti lisada, näiteks pildid, mudelid, skriptid, tekstuurid ja palju muud mis on Unity poolt toetatud (vt. Joonis 4). [15]



Joonis 2.Varade asukoht Unity programmis

Mänguobjektid (inglise k. GameObject) on põhilised komponendid mis kujutavad tegelasi, esemeid ning keskkonda. Iseenesest on nad tühjad objektid ning ei saavuta palju kuid nad on konteinerid, mille sisse saab paigutada teisi komponente mis tagavad tegeliku funktsionaalsuse.[15]

Ülevaate menüü (inglise k. Inspector) näitab mänguobjektile olevaid sätteid nagu näiteks positsioon, pöörlemine ning suurus. Peale selle on veel välja toodud kõik erinevad komponendid mis sellele mänguobjektile lisatud on. Näiteks skriptid, füüsika elemendid ja palju muud (vt. Joonis 2).



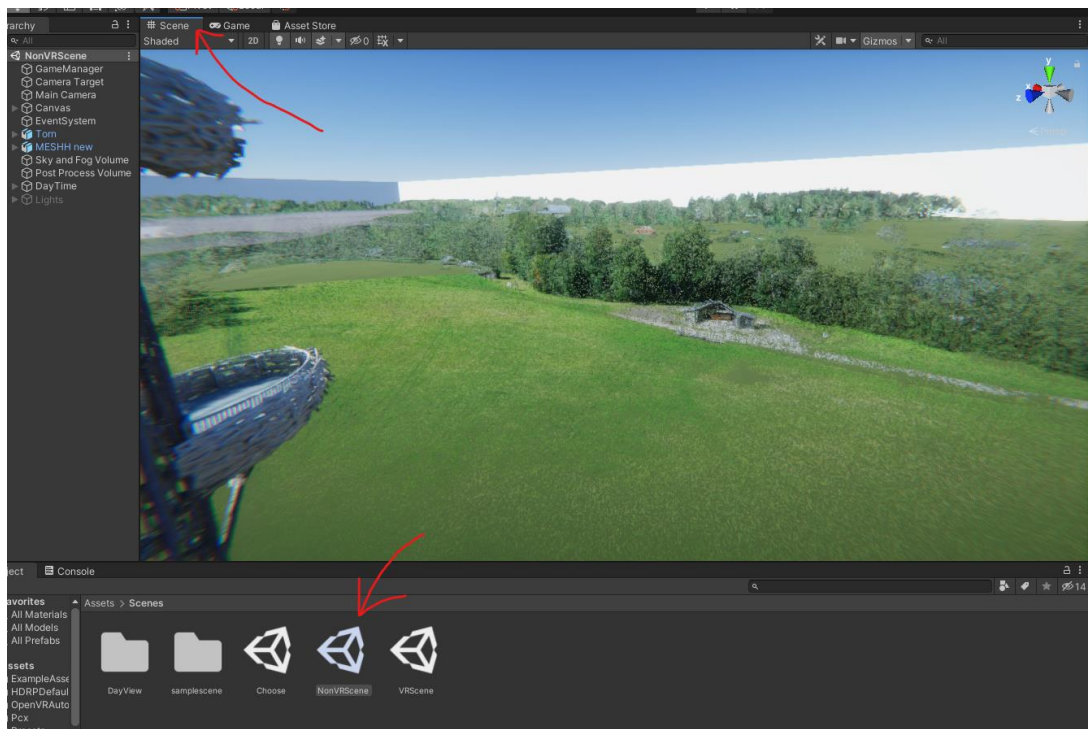
Joonis 3. Näidis kaamera mänguobjektist vasakul ning selle sätetest ülevaate menüüs paremal

Skriptid on peamised komponendid, mis loovad rakendusele elu sisse. Skriptid ütlevad mänguobjektidele kuidas käituda. Mänguobjektide ning skriptide kooslus on see, mis loob mänguelamuse. Unity tarkvara toetab mitmeid objekt orienteeritud skriptimise keeli. Põhiline keel mida Unity kasutab on C#. Skriptide arenduskeskkonnaks on vanemates, enne Unity 2017.3, versioonides MonoDevelop. Keskkond aitab automaatselt koodi lõpetada, näidata kus on tekkinud vead ja võtta otseteid ilma, et peaks ise tervet koodi välja kirjutama. Kui arendust teha Unity versioonides mis on peale 2018.1-te saab kasutada ka Microsofti enda poolt loodud teksti haldus programmi Visual Studio for Community. [16] Unity enda poolsetes C# skriptides on kaks põhifunktsiooni mis on kasulikud ning mida pidevalt kasutatakse. Start, mis läheb tööle iga kord kui stseeni mängitakse. Tavaliselt kasutatakse funktsiooni siis, kui on vaja paika panna mingid väärtused mängu alguses. Update funktsiooni kutsutakse esile iga kaadri (inglise k. Frame) vahetusel. Update funktsiooni käib kõik mida tahetakse programmi jooksmise ajal juhtida nagu näiteks aja möödumist (vt. Joonis 2). [16]

```
Example.cs # X
Assembly-CSharp Example
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Example : MonoBehaviour
6 {
7     // Start is called before the first frame update
8     void Start()
9     {
10    }
11 }
12
13 // Update is called once per frame
14 void Update()
15 {
16 }
17 }
18 }
```

Joonis 4. Näide C# skriptist mis paigutatakse mänguobjekti

Stseeni Unity keskkonnas võib märgendada kui alust või vanemobjekti mille sisse saab paigutada mänguobjekte. Kasutatakse vähemalt ühte stseeni ning kui neid on mitu siis ühendatakse need omavahel läbi skriptimise. Kõik need objektid töötavad omavahel, et luua sujuvat kogemust mängu või rakendust kasutades. Stseeni saab vaadelda stseenivaatest (inglise k. Scene View) (vt. Joonis 4) ning hoiustatakse Assets kaustas. [15]



Joonis 5. Stseenivaade Unity tarkvaras

Üleval toodud joonisel on ülemine nool stseenivaade ning alumine nool stseeni fail kuhu salvestatakse stseeni kohta stseeni andmed.

Valmis olevad komponendid (inglise k. Prefab) on taaskasutatavad mänguobjektide kolleksioonid, mida saab stseenist varade kausta tõmmata ning peale seda lõpmatuid kordi ükskõik millises stseenis uuesti kasutada. [15]

Unity sisaldab kahte skriptitud graafilist töötuselementide jada (inglise k. Scriptable Render Pipeline(SRP)): High Definition Render Pipeline (HDRP) ja Universal Render Pipeline (URP). Mõlemal on erinevad eesmärgid. HDRP jada on mõeldud PC, Xbox ja PlayStation platvormidele millel on palju graafilist jõudlust kuid URP on mõeldud Universaalseks kasutamiseks nagu ka nimi ütleb. Lisaks on ta mõeldud kasutajasõbraliku lahendusena läbi mille on võimalik kergelt tõsta ning optimeerida enda projekti. [17]

4. Punktipilved ja mesh mudelid

Mõningad punktipilved võivad olla mitte ühilduvad erinevates programmides ning neid tuleb teisendada ühest failiformaadist teise. Selleks on kasulikud punktipilvede töötlemise programmid nagu CloudCompare, Agisoft Metashape ja Lastools. Samuti on saadaval programmid ka autodeski poolt nagu näiteks ReCap, Revit ja AutoCAD. [18]

Antud peatükk räägib neljast programmist, mida kasutati punktipilvede parendamiseks. Sellele lisaks on ära toodud tööriistad mida kasutati puhastamiseks ning uute mesh mudelite loomiseks.

4.1. Lastools ja las2las tööriist

Lastools tarkvara koosneb paljudest erinevatest tööriistades. Seda kasutatakse punktipilvede töötlemiseks ning selle peamine eripära on see, et tarkvara on väga kiire, vajab vähe mälu ning sisaldab palju erinevaid tööriistu mis tulevad kasuks punktipilvede töötlemisel.

Õeldakse et las2las on Light Detection and Ranging (LiDAR) failide protsessimise Šveitsi armee nuga. See teisendab, filtreerib, muundab, parandab, muudab mõõtkava, tõlgendab, pakib kokku ja teeb veel palju muud punktipilvedega. Tööriist on 100% avatud lähtekoodiga. [19]

4.2. CloudCompare

CloudCompare on veel üks tarkvara mida saab kasutada 3D punktipilvede töötlemiseks. Algupäraselt oli ta loodud kahe punktipilve võrdlemiseks, näiteks laser skaneerimisega saadud punktipilved. Lõpuks arendati programm aga rohkem tavaliste punktipilvede töötlemise jaoks. CloudCompare on avatud lähtekoodiga projekt mida saab igaüks kasutada ning endale sobivaks programmeerida kui on piisavad oskused. Järgnevalt on ette toodud funktsioonid mida kasutati CloudCompare tarkvaras pilvede korrektuuride tegemiseks. [20]

Apply Transformation tööriist aitab kasutajal muundada punktipilve või mõnda muud objekti. Muundamise sisestusmeetodeid on mitmeid: muundamismaatriks, pöörlemistelg ja nurk + tõlgendus. [20]

Clone tööriist kloonib ette antud objekti, kasulik punktipilvede varundamiseks enne muutuste tegemist. [20]

Cloth simulation filter (CSF) on õhust LiDAR andmete filtreerimise algoritm mis põhineb riide simulatsioonil. Tööriista kasutatakse maapinna punktide eraldamiseks teistest punktidest. Filter sisaldab palju erinevaid sätteid, mille abil saab vajaliku punktipilve mis on sobilik etteantud tööle. Saab muuta riide resolutsiooni mis ütleb ette kui tiheda ruudustikuga algoritm töötab. Maksimum iteratsioon (inglise k. Max iteration) muudab simulatsiooni korduste arvu ja klassifitseerimise lävi (inglise k. Classification threshold) ütleb ära määra mille järgi pinda klassifitseeritakse maa-alaks või mitte. [20]

Merge tööriist liidab kokku kaks või enam olevat objekti. Saab kokku liita punktipilvi või meshe. [20]

Piksliteks muutmise (inglise k. Rasterize) tööriist muundab punktipilved 2.5D ruudustikuks ning peale seda saab seda kas punktipilvena või raster (geotiff) pildina eksportida. Kasulik pinnamudelite loomisel ning mängude loomisel suurte avatud maaalade jaoks. [20]

Subsample tööriist vähendab pilves olevaid punkte, kasutades erinevaid meetodeid nagu näiteks juhuslik (inglise k. random), mis võtab mingi arv punkte suvalises järjekorras. Ruumilises (inglise k. Spatial) meetodis peab kasutaja sisestama minimaalse kauguse kahe punkti vahel ning CloudCompare võtab punkte originaalpilvest nii, et ükski punkt pole täpsustatud pilves lähemal kui kasutaja poolt sisestatud kaugus. Mida suurem ühik seda vähem punkte jäetakse alles. Puu andmestruktuuri (inglise k. Octree) meetod laseb kasutajal valida alajaotuse taseme mille põhjal punktipilve lihtsustatakse. [20]

Statistical Outlier Removal (SOR) filter tööriista kasutatakse punktipilvede puhastamiseks. Filter arvutab iga punkti keskmise kauguse tema naaberpunktist ning eemaldab punktid, mis on kaugemal kui keskmine kaugus pluss standardhälve mis sisestatakse kasutaja poolt. [20]

4.3. Agisoft Metashape

Kolmandaks programmiks mida kasutati punktipilvede töötlemisel ümbruskonna loomiseks oli Agisoft Metashape. Programmi abil saab luua piltidest punktipilvi ning mesh mudeleid. Metashape on üks populaarsemaid professionaalseid punktipilvede töötlemise ning loomise programme. Alapeatükis on ära toodud milliseid funktsioone programmist töö tulemuseks kasutatud sai. [21]

Align photos tööriista abil leiab Agisoft iga kaamera asukoha ja orientatsiooni ning joondab ette antud pildid ja loob hõreda punktipilve mudeli. Seda kasutatakse järgnevates astmetes korrektse punktipilve loomiseks. [21]

Mõningatel juhtudel tuleb eemaldada punktid, mis on kõrge veaga. Gradual selection tööriista kasutatakse üleliigsete punktide eemaldamiseks. Punktid eemaldatakse kindlate kriteeriumide põhjal. [21]

Esimeseks kriteeriumiks on Reprojection error. Kui error on kõrge siis see tähendab tavaliselt, et pildid on halvasti joondatud. Teiseks kriteeriumiks on Reconstruction uncertainty. Kõrge arvu puhul võivad erinevad punktid olla eraldatud objektist mida tahetakse jäädvustada. See tekitab punktipilves müra. Nende punktide eemaldamine ei mõjuta optimeerimise täpsust aga oleks kasulik eemaldada enne geomeetrilise pinna loomist nagu mesh mudel. Kolmandaks kriteeriumiks mida kasutati on Projection accuracy. See kriteerium aitab filtreerida välja punkte, mis on halvema täpsusega kuna nad on suuremad. [21]

Peale punktide eemaldamist tehakse Optimize cameras tööriistaga kaamerate optimeerimist, mis tõstab piltide täpsust kindla punktipilve põhjal ning saadakse parem pilv. [21]

Build dense cloud tööriista abil saab Agisoft Metashape programmis luua tihedaid punktipilvi. Pilv luuakse kaamera positsioonidest ning arvutatud sügavuskaarti põhjal mis on kokku pandud igast stseenis olevast kaamerast. Tihedat punktipilve on võimalik kasutada mesh mudeli loomiseks. [21]

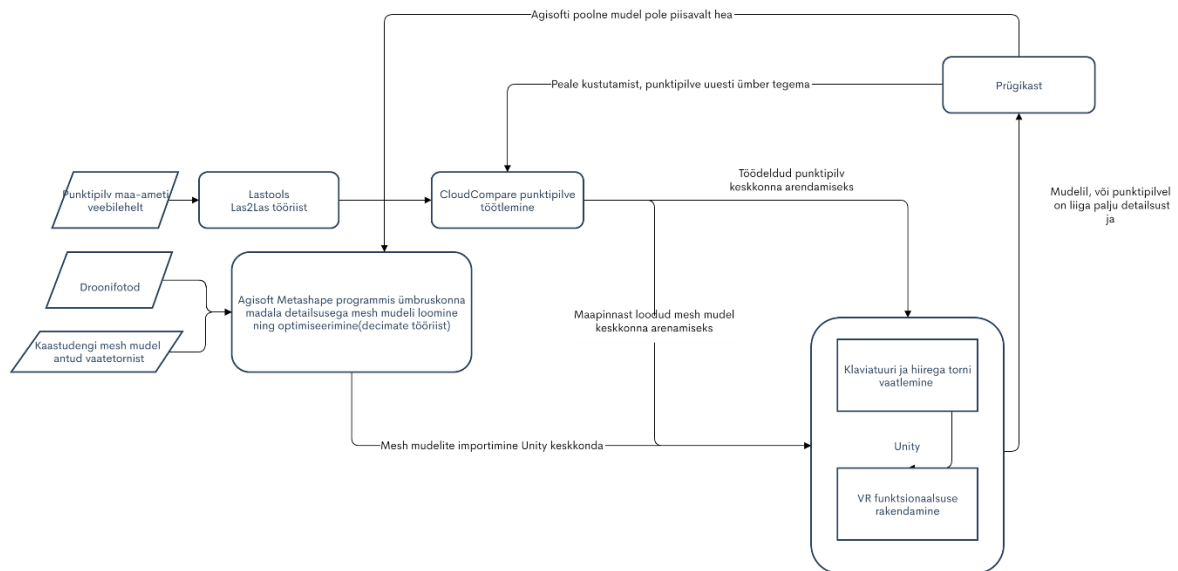
Build Mesh tööriista kasutatakse mesh mudeli loomiseks kas sügavuskaartidest, tihedast-, hõredast punktipilvest. Igal valikul on oma plussid ja miinused. [21]

Mesh mudelile tekstuuride saamiseks kasutati Build Textures tööriista mis kasutab piltidest saadud informatsiooni ning annab mesh mudelile pärismaailma värvid. [21]

Decimate Mesh tööriista abil saab Agisoft Metashape keskkonnas lihtsustada mesh mudelit. Mida vähem on meshis kolmnurki, seda rohkem jääb arvutile jõudlust teiste operatsioonide jaoks. [21]

5. TÖÖKÄIK

Järgnevas peatükis räägitakse töökäigust ning projekti valmimisel tekkinud probleemidest. Töö tegemine punktipilvede faasis oli väga segamini kuna oli vaja testida mis lahendused Unitys töötaksid ning mis ei tööta. Töökäik jagunes 4 etappi.

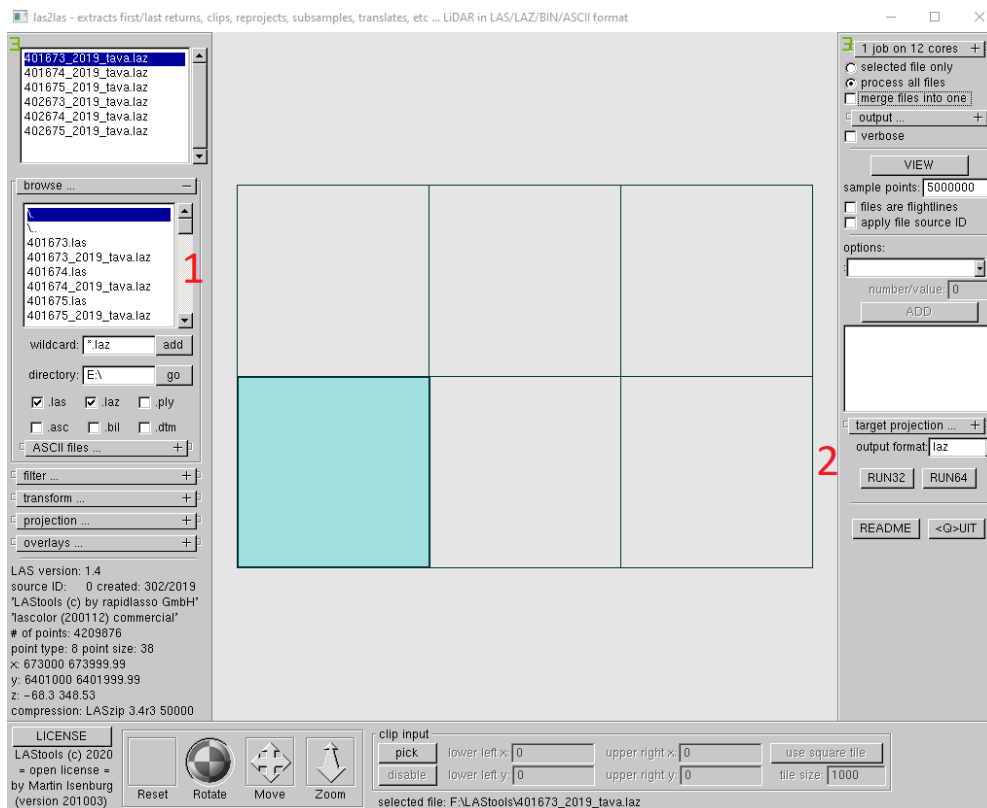


Joonis 6. Töökäik kujutatud vooskeemina (vt. Lisa 6)

5.1. Punktipilvede töötlemine

Enne kui sai Unity mängumootoris punktipilvi kasutada, tuli korrigeerida need välises tarkvaras. Tudeng kasutas selle jaoks Lastools ja CloudCompare tarkvara. Maa-ala andmed on saadud maa-ameti geoportaalist, kust saab alla laadida erinevaid ruumiandmeid. [1] Tudeng hankis endale Ööbikuoru vaatetornile 6 lähedal asuvat aerolaserskaneerimise kõrguspunktide andmestiku. Lisaks hõrendatud punktide kogu mis olid kasulikud maapinna kõrgusandmete paika panemiseks.

Kuna maa-ameti kõrguspunktide laz failiformaat pole ühilduv ühegi punktipilve töötlemise tarkvaraga v.a. lastools, siis tuli teha konversioon tarkvara las2las funktsionaalsusega. Selle abil sai lahti pakkida laz failid las failiformaati mis avanevad erinevates tarkvarades.

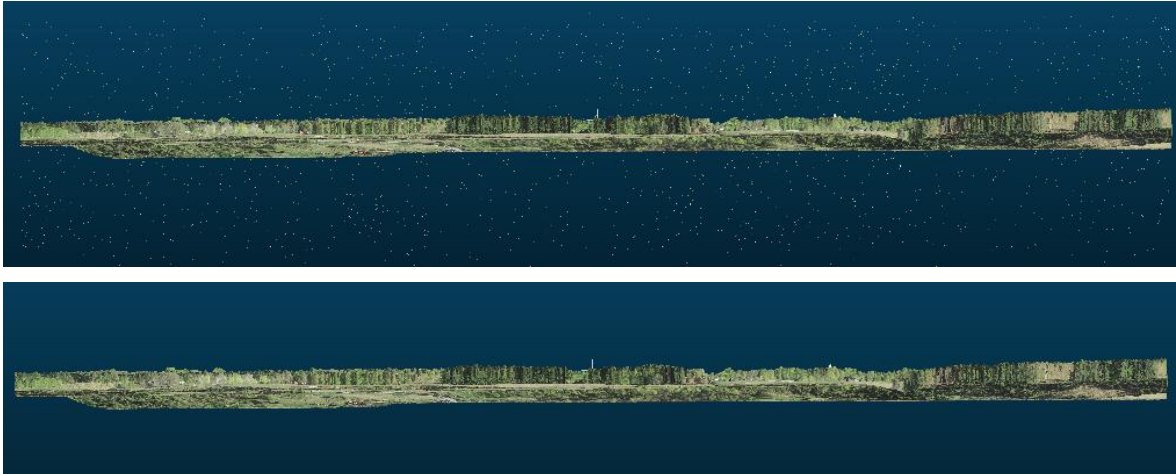


Joonis 7. Lastools tarkvara kasutamine

Kõrguspunktide kogu oli vaja lahti pakkida laz formaadist las formaati. Selleks sai kasutatud Lastoolsi tarkvara poolt saadaval olevat las2las tööriista (vt. Joonis 7). Tööriistas tuli lisada browse aknast failid tarkvarasse ning seejärel vajutada nupule RUN32, et alustada lahti pakkimist las formaati.

CloudCompare tarkvara abil luuakse maa-ala kõrgusandmetest maapind ning saab lahendatud punktipilvede edasine konversioon ply failiformaati mida kasutatakse Unity tarkvaras punktipilve kuvamiseks.

Punktipilved imporditi CloudCompare tarkvarasse ning seejärel liideti kõik kuus pilve kokku kasutades merge funktsiooni. Peale seda kasutas autor kõrguspunktide puhastamiseks SOR Filtrit kuna maa-ameti enda punktipilves on õhus palju punkte mis on vigased. Filtris on kasutatud vaikimisi sätteid.



Joonis 8. Pildid enne (üleval) ja pärast (all) SOR Filtri kasutamist

Ülemisel joonisel on näha, et enne SOR Filtri kasutamist on maa-ameti poolt saadud punktipilv suvalisi punkte täis, kuid kui kasutada SOR filtrit siis saab nendest mittevajalikest punktidest lahti.

Peale punktide filtreerimist sai hakata välja võtma maapinna punkte. Selleks tuli kasuks *CSF*. Kasutati algsätteid. Peale *CSF*-i kasutamist oli pilv valmis ning enne eksportimist oli vaja kasutada Apply transforms tööriista. Sellest räägitakse peatükis 5.1.1.

Hõrendatud punktide kogust sai tehtud CloudCompare tarkvaras maapinna mudel. Selleks kasutas autor Rasterize tööriista kasutades 0.5 astme suurust z teljel, et saada pealtvaade maapinnast. Vajutades mesh nuppu saadi kätte eksporditud mesh mudel mida polnud vaja muundada Apply transform tööriistaga. Mesh sai eksporditud obj faili formaadina otse Unity projekti.

Agisofti tarkvara jäi viimaseks mille abil loodi ümbruskonna 3D mudel lisadetailsuse saamiseks. Eesmärgiks oli lisaks punktipilvedele veel lisa mesh mudel luua, mis täiustaks stseeni parema kvaliteediga.

Kaastudengi käest saadud piltide põhjal oli võimalik luua torni ümbruskonnast hõre punktipilv kasutades align points tööriista. Peale seda kasutati gradual selection ja optimize cameras tööriista vaheldumisi käies läbi kõik kolm agisofti peatükis eelnimetatud kriteeriumit mida töös kasutati. Esimesena leiti punktid mis ei sobinud valides Reconstruction error ning kasutades taset (level) 10. Peale igat punktide eemaldamise korda kasutati Optimize cameras tööriista valikuga Adaptive camera model fitting. Teisena leiti Reprojection error ning kasutati taset 0.5. ja kolmandaks leiti Projection accuracy taset 10. Kui kõrgendatud punktipilv sai töödeldud, loodi tihe

punktipilv kasutades Build dense cloud tööriista (Lisa 3). Tänu tihedale punktipilvele saadi kätte ka sügavusmodelid mis sellega kaasnesid.

Sügavusmodelite põhjal loodi mesh mudel kasutades build mesh model tööriista ning tekstuurid mesh mudeli peale kasutades build textures tööriista. Kui tekstuurid olid mesh mudeli peale lõplikult ehitatud, siis sai mudeli eksportida obj formaadis Unity projekti edasiseks kasutamiseks.

5.1.1. Tekkinud probleemid pilvede töötlemisel

Punktipilvede importimise ajal tuleb ette Global menüü valik mis muudab punktipilve kordinaatsüsteemi. Kuna Unitys pole sisse ehitatud kordinaatsüsteemi, kaob punktipilv ära kui valida global shift menüüst „Preserve global shift on save“.

Veel üks probleem mis tekkis oli muundatud punktipilv. Millegipärast kui eksportida punktipilv originaalvormis Unity keskkonda siis on terve pilv ümber pööratud. Nagu näiteks kui päris maailmas on tee ja puud paremal pool siis punktipilvel on puud vasakul pool. Täpselt sama efekt mis tekib peeglisse vaadates kus kõik on vertikaalselt ümber pööratud. Lahendus sellele probleemile oli lihtne. Apply transformation tööriista kasutades 4x4 maatriksi viimane väärtus muuta -1-ks, mis pöörab terve pilve õigeks.

5.2. Unity VR-less stseen

Tudeng hakkas Unity mängumootoris uut projekti looma. Enne projekti loomist on kasutajale ette toodud mitu erinevat malli, mille abil saab tõhusalt oma ettekujutlust tulevasest rakendusest täide viia. Põhilised kaasas olevad mallid mida kasutaja saab valida on *2D* mida kasutatakse 2 dimensiooniliste mängude loomiseks; *3D* mida kasutatakse 3 dimensiooniliste mängude loomiseks; *3D with extra* mis sisaldab lisaks 3D funktsionaalsusele veel lisa varasid mis tulevad kasuks 3D mängude loomisel; *High Definition RP* ning *Universal Pipeline*. Tudeng proovis neid kahte renderdamise jada oma projekti kaasata. Lõpptulemus on saadud HDRP jada kasutades. Kuna antud projektis on suurem rõhk torni ning ümbruskonna esitlusel, siis oli otstarbekam kasutada HDRP jada mis aitas kaasa stseeni kõrgemale kvaliteedile.

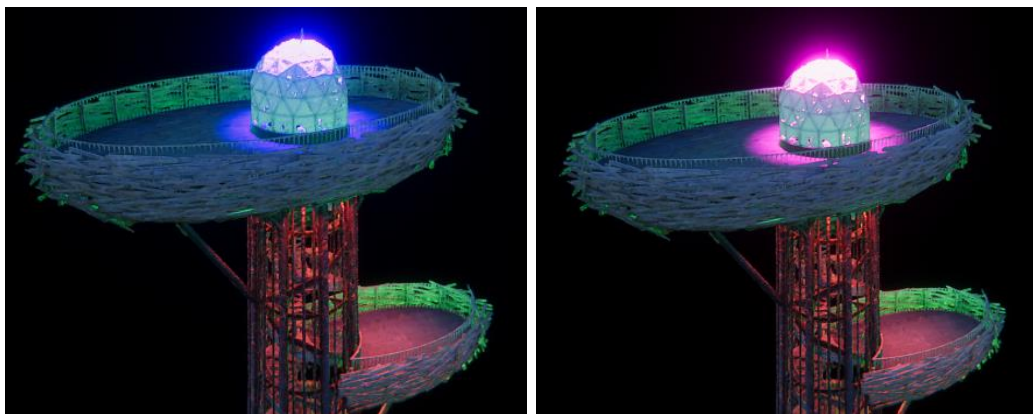
Järgmisena tuli autoril luua stseen ning tuua sisse kõik relevantssed objektid nagu maapinna, torni ning ümbruskonna mudelid. Maa-ameti poolt saadud punktipilved ümbruskonna dekoreerimiseks ja kõik muu vajalik. Kuna Unity tarkvaral pole sisse

ehitatud punktipilvede vaatlemise valikut, siis tuli leida välja arendatud pakk koodi, mis seda teeb. Githubist leiti *Pcx importer* mis renderdab Unity keskkonnas kõik punktipilved. Kui *Pcx importer* pakk Unity projekti sisse tuua töötleb ta kõik *ply* formaadis punktipilved läbi ning seab nad valmis kasutuseks. Peale selle paki hanget on kõik vajalik, mida ise looma ei pea, olemas.

Kõige tähtsamad komponendid projektis on menüüd mille abil on võimalik navigeerida stseenide vahelt ning mõjutada mängus olevaid mänguobjekte kasutades skriptide ja menüü nuppude kooslust. Uuemates Unity versioonides on väga lihtne püsti panna töötavad menüüd ning liita need kokku skriptidega.

5.3. Tornil tulede juhtimine skriptide abil

Tornil on projektorid ning LED ribad eri värvuse jaoks. Kindlasti oli väga huvitav välja mõelda moodust kuidas luua funktsionaalsus, et saaks muuta torni värve vastavalt vajadusele. Hetkel on olemas kolme värvuse süsteem, kus saab valida sinist või roosat vastavalt sellele kas on sündinud poiss või tüdruk. Lisaks on veel võimalus valida jõuluvärvi mis on seatud punaseks (vt. Joonis 16). Kogu skripti osa on sisestatud faili „*ColorChanger.cs*“.



Joonis 9. Näide tornil olevatest tuledest ja nende muutusest

Joonisel 9 on näha et vasakul pool on sinist värvi tuli ning kui vajutada tüdruku sünni nuppu siis muutub tuli ühtlaselt sinisest värvist roosaks värviks. Nupud on nähtaval joonises 16.

Tulede loomisel tekkis omaette probleem: kui ühes lähikonnas on liiga palju erinevaid tulede mänguobjekte, siis lakkavad osad mänguobjektid töötamast. Lahenduseks vähendati mänguobjekte ning kasutati tõhusamaid valguse kiirgamise meetmeid.

5.4. Kasutatud skriptid

Peatükis räägitakse kümnest skriptist mis on lisatud rakendusele ning millised on nende skriptide funktsionaalsused. Kõik skriptid on üles laetud githubi veebilehele (Lisa 1).

GameStart on kõige esimene skript mida kasutatakse rakenduses. Selle skripti abil saab valida kas kasutaja soovib kasutada VR funktsionaalsust või mitte.

GameManager juhib enamuste nupuvajutuste registreerimist, mänguobjektide eemaldamist ja juurde lisamist ning kaamera fikseerimist VR-Less stseenis.

CameraMove on skript mida kasutatakse kaamera sujuvaks liigutamiseks ümber torni.

ColorChanger skript haldab stseenis oleva torni tulede muutmist. Värvide muutmise skript määrab ära värvi intensiivsuse ning värvi.

ChangeHeight skript on loodud kaamera kõrguse vahetamiseks mida kasutatakse VR-Less stseenis.

FindFingerL, *FindFingerR* leiavad VR stseenis kasutaja nimetissõrme ning lisavad sellele sfääri mida kasutatakse kasutaja poolsete mõjutamiste tajumiseks.

SimpleCameraController on skript mis lubab kasutaja vabalt stseenis ringi liikuda kasutades klaviatuuril w, a, s, d klahve ning hiirt. Shift klahvi peal hoides liigub kaamera kiiremini edasi.

SwitchTime on skript mida kasutatakse VR stseenis päeva ja öö vahetamisel valgustuste näitamiseks.

TeleportArrows on skript mida kasutatakse VR stseenis kasutaja ringi liigutamiseks pikemate vahemaade puhul, näiteks torni 2 korruse vahetamine ja eemale liikumine tornist.

5.5. Unity VR stseen

Kõige viimaseks etapiks jäi VR funktsionaalsuse rakendamine. Selleks tuli Unity keskkonda lisada SteamVR pistikprogrammi pakk Unity enda *Asset store* menüüst (Lisa 2). Peale pistikprogrammi lisamist küsitakse sisestusmeetmete automaatset paika panemist millele tuleb *Accept* all nuppu vajutada. Kindlasti tuli autoril vajutada

play nuppu, mis asetseb üleval keskmisel real, et lõplikult valmis seada SteamVR pistikprogrammi sisestused. Kuna pistikprogramme arendatakse pidevalt edasi siis võib kindla peale öelda, et autoripoolsed käigud võivad tulevikus muutuda.

Pistikprogrammi alla laadimise ning importimise tulemusel oli nüüd võimalik stseeni sisse tuua *Player prefab* ning ära kustutada *Main camera* mänguobjekt. See mänguobjektide kollektsioon on kokku pandud pistikprogrammi loojate enda poolt ning lahendab paljud virtuaalreaalsuse rakendamise etapid. Autor sai lõpuks vaadelda mängumaailma läbi VR peaseadme.

Ringi liikumiseks on mitmeid erinevaid võimalusi millest põhiline on teleportatsioon. Teleportatsioon valiti selle tõttu kuna see on kõige kasutaja sõbralikum valik mis ei tekita iiveldust.[22]

Teleportatsiooni korral kasutaja näitab kontrolloriga, kuhu ta liikuda soovib ning mängu teleporteerub kasutaja antud asukohta. Sellise võimaluse rakendamiseks tuleb stseeni sisse tuua *Teleporting* komponent mille võib asetada juhuslikku kohta ning kasutaja poolne teleporteerumine hakkab automaatselt tööle. Selleks, et oleks pinda, mille peale teleporteeruda on vaja stseeni sisse tuua või luua mänguobjekt. Selleks võib olla kas mõni algne ese, näiteks kuubik või midagi millel on rohkem kompleksust mõnest 3D modelleerimise programmist.

Peale mänguobjekti sissetoomist tuleb objektile lisada uus komponent nimega *Teleport area*, mis paneb paika ala, mille sees saab teleporteeruda. Alapeatükis 5.5.1.1 on välja toodud ka probleem mis võib tekkida peale *Teleport area* komponendi lisamist objektile.

Mängumenüüd ja *GameManager* mänguobjekt VR stseenis puudub, põhjus on ka välja toodud alapeatükis 5.5.1.2.

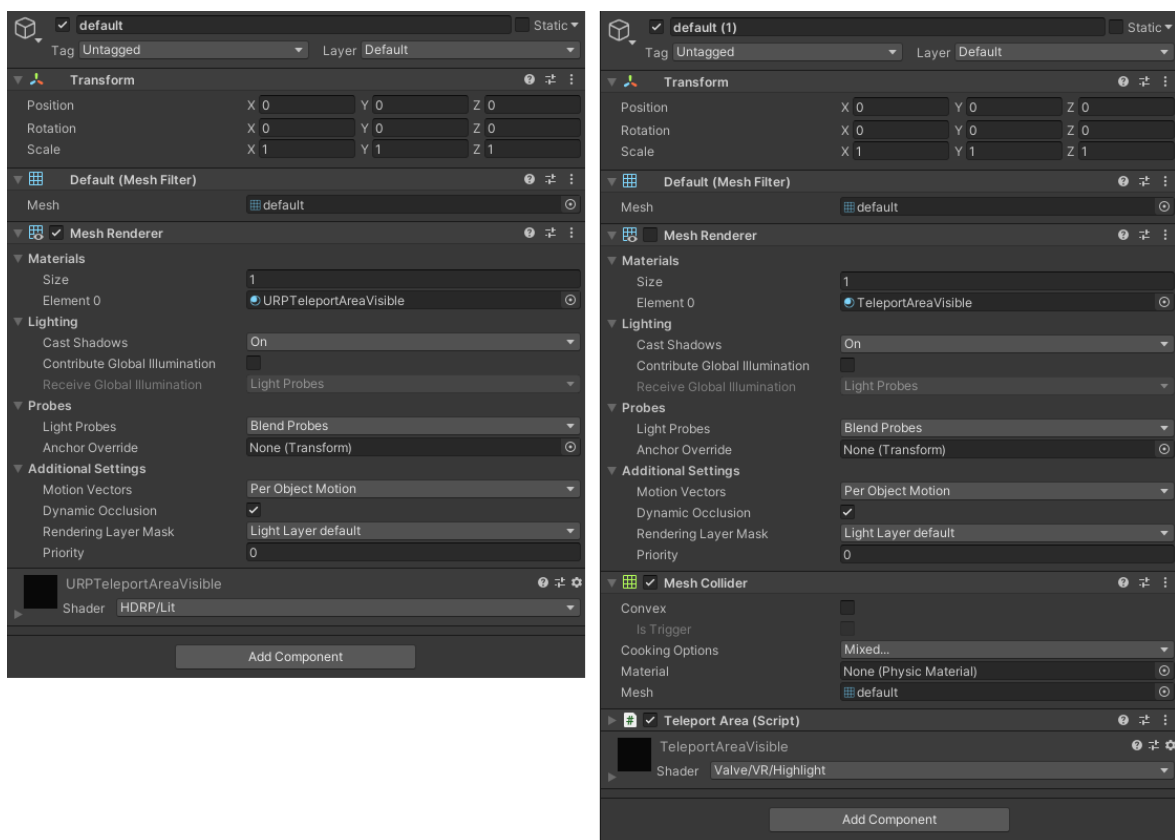
5.5.1. Probleemid VR implementeerimisega

Alapeatükis on välja toodud probleemid mis tekkisid programmis virtuaalreaalsuse rakendamisel.

5.5.1.1. Teleporteerumine

Peale *Teleport area* komponendi lisamist objektile ei töötanud teleporteerumine kuna ei eksisteerinud kokkupõrke pinda mille peale kasutaja osutama peab. Lahendus on kopeerida

ning kleepida samasse kohta objekt mille peale teleporteerutakse. Joonisel 10 on välja toodud originaalne mänguobjekt ning selle kleebitud mänguobjekt. Mängukomponentides on näha viite komponenti: *Transform*, *Default(mesh filter)*, *Mesh renderer*, *Teleport Area* skript ja *Mesh Collider*. Esimene mänguobjekt koosneb ainult *Transform* ja *Mesh renderer* komponentidest. Teisele mänguobjektile on juurde lisatud *Mesh collider* ja *Teleport area* skript. Komponente saab lisada vajutades komponendi nimele parema hiireklõpsuga ning vajutades valikule *Remove component*. Komponentide lisamiseks tuleb vajutada allolevale nupule *Add component* (vt. Joonis 10). Selline viis lahendas probleemi stseenis ringi teleporteerumisel.



Joonis 10. Parem pool on näide teisest objektist ja vasakul pool esimesest

5.5.1.2. Menüü ümberehitus

Kõige suuremaks probleemiks menüüdega oli, et peale VR süsteemi rakendamist tuli ümber teha terve süsteem kuidas kasutaja päeva ja öö valikut vahetab kuna VR stseenis toimib ekraanil olevate menüüde mõjutamine teisiti. Autor eemaldas menüü täielikult. Lõplikuks maailma mõjutamise ja kasutaja liikumise lahenduseks VR

keskkonna jaoks tulid interaktiivsed objektid. Tornil eri kõrgustel liikumiseks kasutati juhtivaid nooli mille peale nimetissõrmega vajutades saab kasutaja muuta asukohta. Värvide muutmiseks piisas aga nuppudest, mis said paigutatud tornist eemale ning töötasid samal põhimõttel (vt. Joonis 11).



Joonis 11. Näidis nooltest mille abil saab asukohta muuta

6. Välja kujunenud rakendus

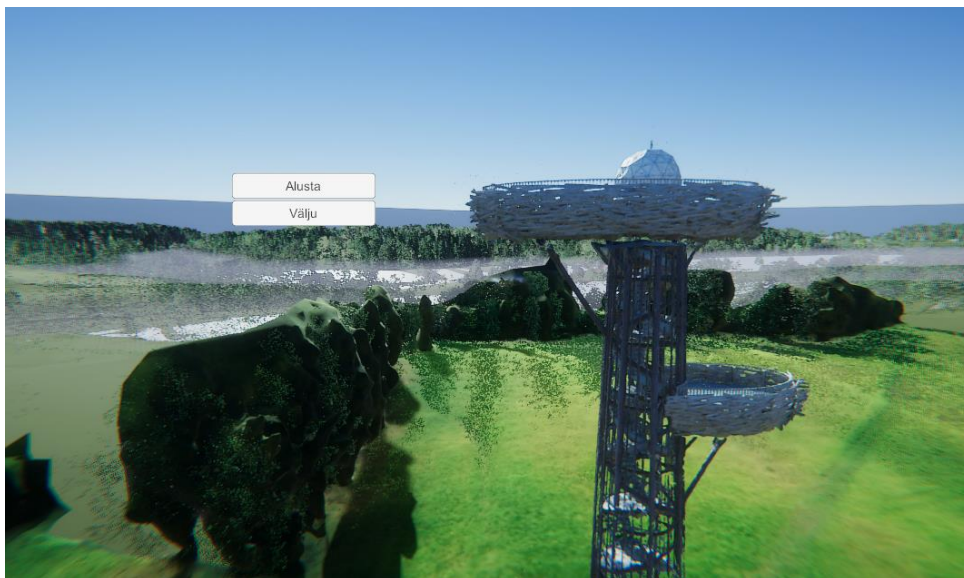
Lõpuks sai valmis rakendus, mille abil on võimalik vaadelda Rõuge Ööbikuoru vaatetorni ning ümbritsevat loodust. Rakenduse käivitades tuleb ette 2 valikut, „VR“ ja „Desktop“. Vajutades nupule „VR“ käivitatakse stseen milles saab kasutaja avastada torni virtuaalreaalsus prillide abil. Kui aga vajutada nupule „Desktop“ siis saab kasutaja vaatetorni vaadelda hiire ja klaviatuuri abil (vt. Joonis 12). Valmis rakendus on üles laetud veebi (vt. Lisa 4).



Joonis 12. Rakenduse valikumenüü

6.1. VR-Less stseen

Selles stseenis saab lihtsal moel vaadelda Rõuge Ööbikuoru vaatetorni ja uurida kõike mis on rakendusse lisatud. Stseen algab kahe valikuga. „Alusta“ nupule vajutusel viiakse kasutaja edasi järgmisesse menüüsse ja „Välju“ nupule vajutades saab rakendusest väljuda (vt. Joonis 13).



Joonis 13. Avavaade vajutades Desktop nupule

Vajutades nupule „Alusta“ viiakse kasutaja edasi menüüvalikute „Päevane vaade“ ja „Õine vaade“ juurde kus kasutaja saab nende kahe vahel valida (vt. Joonis 14).



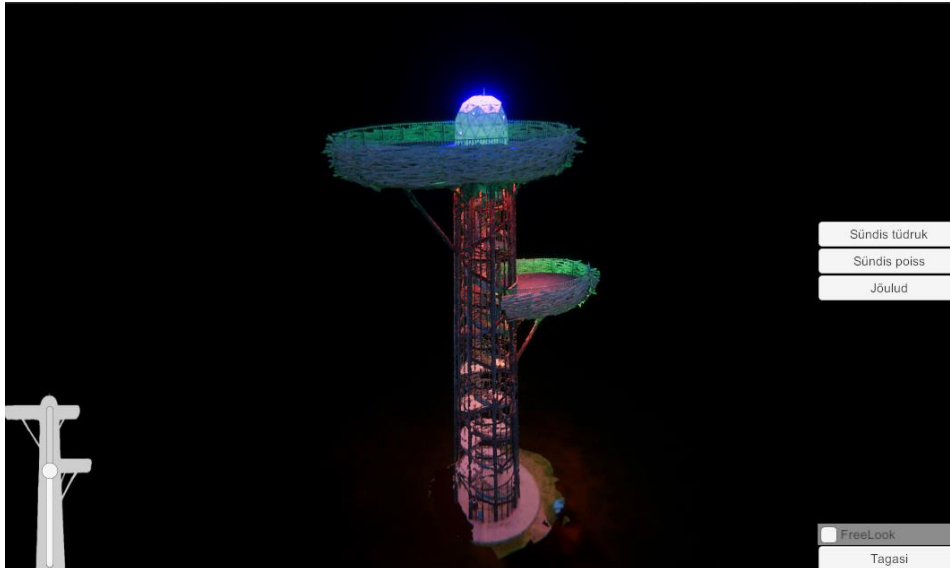
Joonis 14. Päevase ja öise menüü valikud

Valides „Päevane vaade“ saab kasutaja vaadelda torni iga külje pealt. All vasakul on võimalus muuta kui kõrgelt on soov torni vaadelda ja on võimalik ringi liikuda lohistades hiire vasaku nupuga. Rakendusele on ka lisatud „FreeLook“ režiim mille abil saab tervet stseeni avastada kasutades w, a, s, d klahve ning hoides all hiire paremat nuppu (vt. Joonis 15).



Joonis 15. Päevane vaade

Valides „Õine vaade“ saab kasutaja vaadelda torni pimedas režiimis ning tänu sellele on parem näha ka tulesid mis tornis on. Õine vaade toimib samal põhimõttel mis päevane vaade kuid saab muuta „kuldmuna“ värvi mis on omane Rõuge Ööbikuoru vaatetornile (vt. Joonis 16).



Joonis 16. Õine vaade

6.2. VR stseen

VR stseeni funktsionaalsus on samalaadne hiire ja klaviatuuri stseenile. VR avab uued võimalused interaktiivsuse poole. Tänu VR-ile saab kasutaja parema kogemuse ja ettekujutuse antud keskkonnast. Stseeniga seotud interaktiivseid elemente ning VR-Less stseeni saab näha videos mis on lisade all (vt. Lisa 5).



Joonis 17. Nupud värvide ja aja muutmiseks VR stseenis

7. Töö analüüs

Töö lõpptulemusena valmis virtuaaltuur Rõuge Ööbikuoru vaatetornist. Kasutades Unity mängumootorit oli võimalusi päris mitmeid. Mängu/tuuri loojale olid kättesaadavad väga paljud erinevad ressursid mille abil saab kasutaja ette tuua objekte kui ka stsenaariumeid, kasutades erinevaid meetodeid mis annavad samasuguse tulemuse kuid töötavad erinevalt.

Antud töös kirjeldati programme mida kasutati ning mida nende programmidega tehti.

Edasiarendust on projektile kindlasti võimalik juurde teha ning on mitmeid muudatusi mis saaks parendada hetkel valminud lõputööd. Nagu näiteks, objektide sisse laadimise järjekorda panemine ning maapinna tükeldamine nii, et jupid mida pole kasutaja poolt näha kaotatakse ära selleks ajaks kuni kasutaja nende poole uuesti pöördub. Peale eelmainitu näeb veel autor, et virtuaalreaalsuse funktsionaalsust saab kõvasti edasi arendada ning optimeerida. Märkimisväärsem optimeerimise võimalus on terve projekt ümber teha URP lahendusele mis võtab vähem graafilisi ressursse. Üks suur edasimineku oleks võimalus torni ja ümbruskonda vaadelda veebibrauserist mis tõstaks kättesaadavust paljudele inimestele.

Kui tulevikus tekiks võimalus, siis oleks töö autoril kindlasti soov teha rohkem koostööd Rõuge vallaga ning panna kokku elamus mida oleks võimalik kogeda kõigil Eesti inimestel.

8. Maksumuse analüüs

Peamine programm mida kasutati on Unity. Unity on tasuta tarkvara nii kaua kuni isik ei teeni oma töö pealt rohkem kui 100 tuhat dollarit aasta jooksul. [23]

Teise programmina kasutati CloudCompare tarkvara on vabavara mida võib kasutada endale sobival viisil. General public license (GPL).[24]

Kolmandaks programmiks kasutati Lastoolsi mis on tasuta tarkvara. [19]

Neljanda programmina kasutati Agisoft metashape mis on programm mida saab kõikide funktsioonidega kasutada 30 päeva kui hankida 30 päevane prooviperiood tasuta. Pikema ajaliseks kasutamiseks tuleb osta litsents mis maksab 179 dollarit. Tudeng kasutas prooviperioodi. [25]

Kõik alla laaditud pistikprogrammid nagu SteamVR ja Pcx Point cloud Importer olid tasuta.

Virtuaalreaalsuse testimiseks kasutati originaalset HTC Vive peaseadet mida hetkel müügis enam pole kuid ostmise ajal läks maksma 650 eurot. Juhtpultidena kasutati Valve Index pulte mis maksavad 300 dollarit. [26]

KOKKUVÕTE

Lõputöö eesmärgiks oli luua Rõuge Ööbikuoru vaatetorni kohta virtuaaltuur mida oleks võimalik kogeda nii virtuaalreaalsus seadmetega kui ka vabalt arvutis klaviatuuri ning hiirega.

Autor rääkis virtuaaltuuridest ning millised on tänapäeval tuuride eesmärgid ja mis viisidel inimestele asukohti ette tuuakse, põhiliselt 360 kaardiste videote või piltide alusel.

Autor tõi välja ka virtuaalreaalsuse populaarsuse ning kuidas see võib olla üks pääsetee ökosõbralikuma reisimise poole. Peale selle toodi välja virtuaalreaalsuse eelised ja puudused.

Töö tegemise käigus puutus autor kokku erinevate punkt pilve tarkvaradega nagu Lastools, Agisoft Metashape ja CloudCompare. Autor tõi välja ka mängumootorid mida on võimalik kasutada, ning mis on nende eelised. Mängumootoritest kokku Unity arenduskeskkonnaga ning kirjeldas enda valiku põhjust. Töös on ka kirjeldatud töökäiku ning kasutatud ressursse.

Unity võimaldas autoril luua mitmekesise rakenduse, millel on olemas kaks erinevat võimalust Rõuge Ööbikuoru vaatetorni vaatlemiseks. Esimese võimalusena saab kasutada lihtsalt hiirt ja klaviatuuri ning teise võimalusena saab ennast teise maailma transportida kasutades virtuaalreaalsuse seadet. Seejuures näha torni just sellisena nagu inimene näeks seda oma enda silmadega.

Kasutades mängumootorit tuuri loomiseks oli autoril rohkem võimalusi sisse tuua interaktiivsust, anda kasutajale paremat ruumitaju ning lubada iseseisvat liikumist ja uurimist mis pole võimalik traditsiooniliste 360 videote ja piltide abil.

Autori poolt püstitatud eesmärgid said lahenduse läbi uurimistöö ja praktilise töö mille käigus pandi kokku virtuaaltuur. Edasiarenduse võimaluseks oleks edasine koostöö Rõuge vallaga ning rakenduse enda parendamine.

Summary

The purpose of this thesis was to develop a virtual tour application of the observation tower at Rõuge Ööbikuorg which could be experienced in virtual reality and freely on the computer with just using the mouse and keyboard.

The author talked about virtual tours and what is their purpose in today's society and in which ways different locations are brought up to the user, usually as 360-degree videos or pictures.

The author also brought up the popularity of virtual reality and how it could be one of the escape routes to a more economically friendly way of traveling. Besides that, the favorable and unfavorable factors of virtual reality were mentioned.

While doing the thesis author got exposed to many different point cloud editing softwares like Lastools, Agisoft Metashape, and CloudCompare. Autor also mentioned many different game engines that could be used and what favorable factors each game engine has. From the game engines, the author got familiar with Unity and described their reason for picking that game engine. It is also explained how the workflow was done and how different resources were used.

Unity allowed the author to create a diverse application that has two different possibilities for exploring the observation tower. The first one is by just using a mouse and keyboard and the other possibility is to transport yourself into a different world by using a virtual reality device which allows the person to see the tower just like they would, by going out to the field by themselves. Some people prefer the latter.

When using a game engine to create a virtual tour the author had more opportunities to bring in more interactivity, give the user a better sense of scale and allow a more nonlinear movement where the user can explore as they please which wouldn't be possible with conventional 360-degree videos and pictures.

The goals set by the author were solved through research and practical work where a virtual tour was put together. Possible further cooperation with Rõuge parish and improvement of the application itself are intended for further development.

KASUTATUD KIRJANDUSE LOETELU

- [1] "Kõrgusandmete allalaadimine",
<https://geoportaal.maaamet.ee/est/Ruumiandmed/Korgusandmed/Laadi-korgusandmed-alla-p614.html>,
kasutatud 04.11.2020, [Võrgumaterjal]
- [2] "X-gis Maainfo kaardirakendus", <https://xgis.maaamet.ee/maps/XGis>,
kasutatud 04.11.2020, [Võrgumaterjal]
- [3] "What is a virtual tour",
<https://360virtualbusinessstours.com/360-virtual-business-tour-faqs/what-is-a-virtual-tour/>,
kasutatud 04.11.2020, [Võrgumaterjal]
- [4] "maritime museum uses 3d scanning technology to digitally preserve jfk bunker museum prior to closing",
kasutatud 04.11.2020, [Võrgumaterjal]
- [5] "ISS Virtual Tour", <https://esamultimedia.esa.int/multimedia/virtual-tour-iss/>,
kasutatud 04.11.2020, [Võrgumaterjal]
- [6] "Is virtual travel here to stay, even after the pandemic subsides?",
<https://www.nationalgeographic.com/travel/2020/04/can-virtual-reality-replace-real-tourism-during-pandemic-and-beyond/>,
kasutatud 06.11.2020, [Võrgumaterjal]
- [7] "Virutal reality in mental health",
https://www.researchgate.net/publication/6395995_Virtual_reality_in_mental_health_A_review_of_the_literature,
kasutatud 06.11.2020, [Teaduslik kirjandus]
- [8] "Virtual reality vacation has its pros and cons", <https://www.tourism-review.com/travel-tourism-magazine-virtual-reality-vacation-pros-and-cons-article2748>,
kasutatud 08.11.2020, [Võrgumaterjal]
- [9] "Analysis: 'Half-Life: Alyx' Adds Nearly 1 Million VR Users to Steam in Record Gain", <https://www.roadtovr.com/steam-survey-vr-headset-growth-april-2020-half-life-alyx/>,
kasutatud 06.11.2020, [Võrgumaterjal]
- [10] "10 best video game engines", <https://www.gamedesigning.org/career/video-game-engines/>,
kasutatud 10.11.2020, [Võrgumaterjal]

- [11] "Unity platforms", <https://unity.com/features/multiplatform>,
kasutatud 14.11.2020, [Võrgumaterjal]
- [12] "Automotive & transportation", Unity, <https://unity.com/solutions/automotive-transportation-industrial-bundle>,
kasutatud 11.11.2020, [Võrgumaterjal]
- [13] "Tour creator", <https://arvr.google.com/tourcreator/>,
kasutatud 10.11.2020, [Võrgumaterjal]
- [14] "Unity manual",
<https://docs.unity3d.com/2019.3/Documentation/Manual/VROverview.html>,
kasutatud 14.11.2020, [Võrgumaterjal]
- [15] „Game project elements of Unity 3D“,
<https://www.studytonight.com/3d-game-engineering-with-unity/elements-of-unity3d>,
kasutatud 14.11.2020, [Võrgumaterjal]
- [16] "Coding in C# in Unity for beginners", <https://unity3d.com/learning-c-sharp-in-unity-for-beginners>,
kasutatud 20.11.2020, [Võrgumaterjal]
- [17] "Unity HDRP", <https://unity.com/srp/High-Definition-Render-Pipeline>,
kasutatud 25.11.2020, [Võrgumaterjal]
- [18] "What are the most popular types of point cloud processing software?",
<https://info.vercator.com/blog/what-are-the-most-popular-types-of-point-cloud-processing-software>,
kasutatud 13.11.2020, [Võrgumaterjal]
- [19] „las2las“, <https://rapidlasso.com/las2las/>,
kasutatud 07.11.2020, [Võrgumaterjal]
- [20] "Cloudcompare manual", www.cloudcompare.org,
kasutatud 07.11.2020, [Võrgumaterjal]
- [21] "Agisoft manual", https://www.agisoft.com/pdf/metashape-pro_1_5_en.pdf,
kasutatud 07.11.2020, [Võrgumaterjal]
- [22] "Point & Teleport Locomotion Technique for Virtual Reality"
<https://dl.acm.org/doi/10.1145/2967934.2968105>
Kasutatud 06.11.2020, [Võrgumaterjal]
- [23] „Plans and pricing“, <https://store.unity.com/#plans-individual>,
kasutatud 29.12.2020, [Võrgumaterjal]
- [24] CloudCompare, <http://cloudcompare.org/>,
kasutatud 29.12.2020, [Võrgumaterjal]

- [25] "Buy Agisoft Metashape", <https://www.agisoft.com/buy/online-store/>,
kasutatud 29.12.2020, [Võrgumaterjal]
- [26] "Valve Index controllers",
https://store.steampowered.com/app/1059550/Valve_Index_Controllers/,
kasutatud 29.12.2020],[Võrgumaterjal]

LISAD

(Lisa 1) Projekti koodi osa:

Nerabwyn/-bikVaade: Ööbikuoru vaatetorni virtuaaltuuris kasutatud skriptid (github.com)

(Lisa 2) Kasutatud SteamVR Plugin:

SteamVR Plugin | Integration | Unity Asset Store

(Lisa 3) Tööriist mille järgi on tehtud agisofti punktipilv:

Agisoft PhotoScan workflow part 1 (with ground control points) - YouTube

(Lisa 4) Valmis kujunenud rakendus

<https://mega.nz/file/jx8GgK5D#SUjNkTd-CKbCLo04cngOmT209AMOW0E6k6RhBw8MgEA>

(Lisa 5) Video link VR ja VR-Less stseeni kohta:

<https://youtu.be/R6dNjIwEevc>

(Lisa 6) Vookeem

