



TALLINNA TEHNIKAÜLIKOO  
INSENERITEADUSKOND  
Virumaa kolledž

**Virtuaaltuuri loomine Unity´ s Kohtla-Järve linna osa  
3D-mudeli näitel**

**Creating a virtual tour in Unity on the example of a 3D model of  
the part of Kohtla-Järve city**

TELEMAATIKA JA ARUKAD SÜSTEEMID ÕPPEKAVA LÕPUTÖÖ

Üliõpilane: Vladlena Utt

Üliõpilaskood: 182737EDTR

Juhendaja: Natalja Maksimova,  
lektor

# AUTORIDEKLARATSIOON

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.

"...." ..... 20.....

Autor: .....

/ allkiri /

Töö vastab rakenduskõrgharidusõppe lõputööle/magistritööle esitatud nõuetele

"...." ..... 20.....

Juhendaja: .....

/ allkiri /

Kaitsmisele lubatud

"...." ..... 20.....

Kaitsmiskomisjoni esimees .....

/ nimi ja allkiri /

## **LIHTLITSENTS LÕPUTÖÖ ÜLDSUSELE KÄTTESAADAVAKS TEGEMISEKS JA REPRODUTSEERIMISEKS**

Mina Vladlena Utt (sünnikuupäev: 17.09.1994)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose Virtuaaltuuri loomine Unity´s Kohtla-Järve linna osa 3D-mudeli näitel, mille juhendaja on Natalja Maksimova,

1.1. reprodutseerimiseks säilitamise ja elektroonilise avaldamise eesmärgil, sealhulgas 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 punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta kolmandate isikute intellektuaalomandi ega isikuandmete kaitse seadusest ja teistest õigusaktidest tulenevaid õigusi.

# TalTech Inseneriteaduskond Virumaa kolledž

## LÕPUTÖÖ ÜLESANNE

**Üliõpilane:** Vladlena Utt, 182737EDTR

Õppekava, peeriala: EDTR17, Telemaatika ja arukad süsteemid

Juhendaja(d): lektor, Natalja Maksimova, natalja.maksimova@taltech.ee

Konsultant:

### Lõputöö teema:

(eesti keeles) Virtuaaltuuri loomine Unity´s Kohtla-Järve linna osa 3D-mudeli näitel

(inglise keeles) Creating a virtual tour in Unity on the example of a 3D model of the part of Kohtla-Järve city

### Lõputöö põhieesmärgid:

1. Luua keskkond, kus kasutaja tutvub Kohtla-Järve linna 3D-mudeliga.
2. Luua virtuaaltuur Unity´s, kus kasutajal on võimalik ringi liikuda.
3. Luua interaktsioonipunktid, kus kasutaja saab linna põhiliste objektide kohta teavet kas audio- või videosalvestuste kaudu
4. Teoreetilises osas tutvustada lugejale Unity ning Blenderi rakenduste põhifunktsionaalsust.

### Lõputöö etapid ja ajakava:

Nr	Ülesande kirjeldus	Tähtaeg
1.	Kohtla-Järve linna 3D-mudeli loomine Blender rakenduses.	10.04.2021
2.	Unity rakenduse tundmaõppimine ning teoreetiliste materjalide otsimine.	08.04.2021
3.	3D-mudelite ning nende tekstuuride importimine Unity´sse.	15.04.2021
4.	Virtuaaltuuri loomine Unity´s.	29.04.2021
5.	Audio- ning videofailide lisamine interaktsioonipunktidesse.	05.05.2021
6.	Lõputöö kirjaliku osa vormistamine dokumentatsiooni järgi ning selle esitamine. Dokumentatsioon valmib kogu praktilise töö koostamise ajal.	10.05.2021

<b>Töö keel:</b> eesti keel	<b>Lõputöö esitamise tähtaeg:</b>	“27”mai 2021a
<b>Üliõpilane:</b> Vladlena Utt	/allkiri/	“.....”..... 20.....a
<b>Juhendaja:</b> Natalja Maksimova	/allkiri/	“.....”..... 20.....a
<b>Konsultant:</b>	/allkiri/	“.....”..... 20.....a
<b>Programmijuht:</b> Žanna Gratšjova	/allkiri/	“.....”..... 20.....a

# SISUKORD

EESSÖNA .....	7
LÜHENDITE JA TÄHISTE LOETELU .....	8
SISSEJUHATUS .....	9
1. VIRTUAALTUURI MÕISTE .....	10
1.1 Virtuaaltuuri loomise tööülesanded ja -etapid .....	12
2. LINNA KAARDI LOOMINE .....	13
3. BLENDER .....	16
3.1 Võrdlus teiste rakendustega .....	16
3.2 Funktsionaalsus .....	17
4. UNITY .....	24
4.1 Võrdlus teiste rakendustega .....	24
4.2 Funktsionaalsus .....	25
5. VIRTUAALTUURI ARENDAMINE .....	26
5.1 3D-mudeli Unity´sse importimine .....	26
5.2 Puuduolevate objektide lisamine Unity´s .....	27
5.3 Esimese isiku vaatenurga lisamine .....	28
5.4 Interaktsioonipunktide lisamine .....	29
5.5 Projekti lõpetamine .....	30
KOKKUVÕTE .....	32
SUMMARY .....	33
KASUTATUD KIRJANDUSE LOETELU .....	34

## **EESSÕNA**

Antud lõputöö idee kujunes praktikate sooritamiste käigus. Praktikate ülesanneteks olid 3D-modelleerimine Blender rakenduses. Lõputöös arendati antud ideed edasi Virumaa kolledzi tellimusel. Antud diplomitöö eesmärgiks on Kohtla-Järve linna osa 3D-mudeli loomine Blenderis ning sellest tutvustava virtuaaltuuri loomine Unity´s, lisades sinna interaktsioonipunktid olulisemate hoonete juurde. Aktiivsed punktid tutvustavad objekti lähemalt kas video- või audiosalvestuse abil. Töö tulemusena valmis virtuaaltuur, kus kasutaja saab jalutada ja tutvuda Kohtla-Järve linnaga seda füüsiliselt külastamata.

Soovin tänada oma juhendajat diplomitöö juhendamise, nõuannete eest ning abistamisel lõputöö teema sõnastamisega.

Võtmesõnad: virtuaaltuur, 3D-modelleerimine, Blender, Unity, diplomitöö

## **LÜHENDITE JA TÄHISTE LOETELU**

AR – liitreaalsus

VR – virtuaalreaalsus

3D – 3-dimensiooniline

2D – 2-dimensiooniline

OSM – OpenStreetMap

GIS – Geographic Information System



## SISSEJUHATUS

3D-modelleerimine on objekti virtuaalse 3D-mudeli loomine. Antud diplomitöös luuakse Kohtla-Järve linna 3D-mudel ning selle põhjal virtuaaltuur, kus kasutajal on võimalik linnaga tutvuda. Sarnast virtuaaltuuri ei ole Kohtla-Järve linna 3D-mudelist varem loodud. Antud töö eripäraks on ka interaktsioonipunktide olemasolu, mis annavad lisateavet olulisemate objektide kohta.

Kuna töö praktiline pool osutus oodatavast mahukamaks, otsustati piirduda Kohtla-Järve linna osa loomisega.

Võrreldes Google Maps virtuaaltuuriga, kus on linna panoraampildid, on antud rakenduses võimalik näha hooneid igast vaatest. Kuigi antud mudel oli koostatud enamasti eelnimetatud panoraampiltidele toetudes, tegi autor lisaks ise mõne hoone pilte, et luua tervikvaade. Eravalduse puhul, kuhu ligipääsu ei olnud, loodi improviseeritud hoone mudelit.

Antud lõputöös käsitletakse 3D-modelleerimist Blender rakenduses ning virtuaaltuuri loomist Unity´s. Kirjeldatakse tuuri loomise etappe ning iseärasusi.

Töö esimeses pooles tutvustatakse virtuaaltuuri mõistet, uuritakse eelnimetatud rakenduste funktsionaalsust, lisaks kasutatud rakendustele ka teisi sarnaseid ning võrreldakse neid omavahel.

Töö teises pooles kirjeldatakse virtuaaltuuri loomise protsessi ning erinevaid probleeme, mis sellega seoses tekkisid.

Antud lõputöö eesmärgiks on Kohtla-Järve linna osa 3D-mudeli loomine Blenderis ning sellest tutvustava virtuaaltuuri loomine Unity´s, lisades sinna interaktsioonipunktid olulisemate hoonete juurde.

Selle eesmärgi saavutamiseks on püstitatud järgmised ülesanded:

- Luua Kohtla-Järve linna hoonete 3D-mudelid Blenderis.
- Luua virtuaaltuur Unity´s, kus kasutajal on võimalik ringi liikuda;
- Luua interaktsioonipunktid, kus kasutaja saab linna põhiliste objektide kohta teavet kas audio- või videosalvestuste kaudu;
- Uurida ja võrrelda modelleerimise ja 3D-tuuri loomiseks kasutatavaid sarnaseid rakendusi ning tutvustada lugejale Unity ja Blenderi põhifunktsionaalsust.

# 1. VIRTUAALTUURI MÕISTE

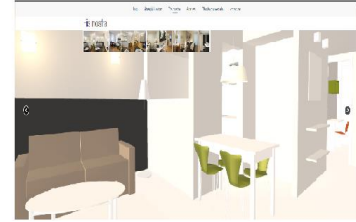
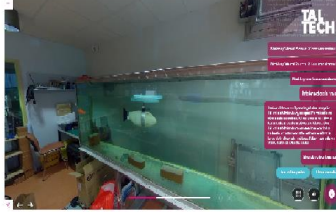
Virtuaaltuur on olemasoleva koha simulatsioon, mis luuakse piltidest, videoklippidest või 3D-mudelist. Kasutajal on võimalik kogeda tunnet olla kohas, kus ta füüsiliselt tegelikult ei ole. Tuure on võimalik vaadata läbi personaal-, süle- või tahvelarvuti, aga ka mobiiltelefonist. Tuuri sisse saab integreerida heliefekte ja muid interaktsioonipunkte, näiteks muusikat, teksti või häälsõnumit, mis kirjeldaksid antud kohta. [1] Järgnevalt kirjeldatakse erinevaid virtuaaltuuri liike.

Üheks alaliigiks on 360 tuur ehk panoraamtuur, mis on väga populaarne kinnisvaramaaklerite, reisijate, pildistajate, hotelliomanike seas [2]. Kinnisvara valdkonnas luuakse rakenduse abil kas interaktiivne korruse plaan või kogu hoone tuur [1]. Turismi valdkonnas kasutatakse tuure selleks, et tutvustada külastajale kohta enne, kui inimene füüsiliselt sinna saabub. Nii saab ettekujutuse muuseumist, hotellist või must ajaloolisest kohast [3]. Selline tuur luuakse staatiliste panoraampiltide abil, kus on võimalik liikuda erinevate lokatsioonide vahel. Vaatleja näeb teda ümbritsevat asukohta 360 kraadi ulatuses.

Lisaks eelnimetatule on olemas ka videotuur, mida on võimalik luua, filmides asukohta videokaamera abil. Tuuri looja kõnnib koos kaameraga ühest punktist teise või kasutab selleks drooni. Selle meetodi eelis võrreldes virtuaaltuuriga on vaatenurga pidev muutumine. Küll aga nõuab filmimine rohkem tehnilist oskust ning varustust kui tavaline pildistamine. [4]

Viimane tuuri tüüp, mida selles peatükis käsitletakse, on 3D-virtuaaltuur, kus luuakse asukoha 3D-mudel. Sellise tuuri eelis võrreldes eelmistega on see, et vaatleja võib liikuda sinna, kuhu ise soovib ning ei ole piiritletud kindla vaatepunktiga. Ka selle töö raames on loodud linna 3D-mudeli põhjal virtuaaltuur, milles on ka multimeedia komponente. Sellise tuuri valik on põhjendatud ka sellega, et seda on võimalik edasi arendada, näiteks hoonete sisse kõndimise võimaluse lisamine või uste dünaamiline avamine. Kuna antud projekti tellijaks on Virumaa kolledz, siis tulevikus saab linna mudelit kasutada mängude arendamiseks.

Joonisel 1.1 on toodud näiteid erinevatest virtuaaltuuride liikidest: panoraam-, video- ning 3D-virtuaaltuur.



Joonis 1.1 Tuuride pildid: TalTech panoraamtuur [5], TalTech videotuur [5] ning interjööri 3D-virtuaaltuur [6].

Virtuaaltuure on võimalik luua lisaks erinevatele seadmetele ka liitreaalsuse (AR) kui ka virtuaalreaalsuse (VR) jaoks. Need ühendavad digitaalset ja füüsilist maailma. AR on reaalsuse ühendamine mitterealsega. Sellised telefoni peale allalaetavad rakendused, kus on läbi kaamera võimalik näha mitterealseid objekte, on AR rakendused. AR laiendab dramaatiliselt viise, kuidas meie seadmed saavad aidata igapäevastes tegevustes, nagu teabe otsimine, ostlemine ja enda väljendamine. VR võimaldab kogeda tunnet olla seal, kus füüsiliselt antud hetkel ei asuta. Läbi VR seadmete saab näha reaalseid asukohti ning objekte, mis asuvad teisel mandril. [7]

On olemas mitmeid rakendusi ja veebilehekülgi, kus saab luua virtuaaltuuri. Erinevad rakendused võimaldavad erinevat tüüpi tuuride loomist. Kõik rakendused ei võimalda 3D-mudelist tuuri loomist, seega kasutati selleks Unity mängumootorit. Järgnevalt kirjeldatakse rakendusi, kus saab luua panoraam- või videotuure, integreerides sinna kas VR või AR.

Üheks selliseks veebirakenduseks on Klapy. Seda kasutavad kinnisvaraagendid, fotograafid, dronistid, hotelli omanikud, õppeasutused, arhitektid, ka eraisikud. See on tasuta ning lihtne rakendus ja võimaldab luua panoraamvaatega tuure. Selleks, et ise luua virtuaaltuur, tuleb registreerida, valida tuuri tüüp (videomängud, 3D-pildid, turism, kinnisvara jne). Järgnevalt tuleb teha 360 ümbruskonna panoraampilt, laadida see üles, täita kõik kohustuslikud väljad (tuuri nimetus, võtmesõnad, kirjeldus) ning tuur saab sellega valmis. Soovi korral pakub rakendus lisada interaktsioonipunkte ühest kohast teise liikumiseks. Need võivad sisaldada endas informatsiooni või pilti. Tuuri võib jagada teistega tasuta. Koduleheküljel on võimalik vaadata ka teiste poolt loodud tuure erinevates kategooriates. [2] Kuna see rakendus piirdub vaid piltide abil tuuri loomisega, ei ole sinna võimalik alla laadida videosid ega 3D-mudeleid.

Teiseks populaarseks virtuaaltuuri loomise rakenduseks on Google VR. Sellel on rohkem võimalusi võrreldes esimesega. See on samuti veebipõhine rakendus, kus saab luua lisaks panoraamvaatega tuurile ka lihtsamat 3D-keskkonda. On olemas ka tuuri virtuaalreaalsusega integreerimise võimalus. Rakendust on sama lihtne kasutada ning tegutsemise põhimõte on sarnane, lisaks pakub ta palju juba olemasolevaid malle, mida

võib kasutada oma tuuri loomiseks. Panoraamtuuri loomiseks võib kasutada kas oma pilte või Google Street View vaateid. Interaktsioonipunktide võimalus on samuti olemas. Tuuri saab jagada lingina. [8]

Tänapäeval on oluline, et virtuaaltuur oleks saadaval igast maailma otsast, seega on otstarbekas teha tuur kättesaadavaks veebist. Kuna selliseid rakendusi on loodud väga palju, siis selles peatükis piirduti vaid paari rakenduse ülevaate tegemisega. Sõltuvalt vajadustest saab leida tuuri loomise võimalusi igaühele.

## **1.1 Virtuaaltuuri loomise nõuded ja etapid**

Antud diplomitöö raames loodud virtuaaltuur vastab järgmistele nõuetele:

- töötab arvutil;
- on loodud Kohtla-Järve linna hoonete 3D-mudelite baasil. Mudelid ei ole hoonete täpsed koopiad, aga on visuaalselt õiged;
- tekstuurid ei ole realistlikud;
- linna infrastruktuur on täpne;
- linnas on olemas punktid, kust saab audio- või videosalvestuse abil lisainfot objekti kohta;
- linnas liikumine toimub esimese isiku vaatenurgast;
- linnas liikumise juhtimine toimub klahvide abil, rotatsioon aga hiire liigutusega;
- igat hoonet võib vaadelda igast küljest;
- hoonete sisse ei ole võimalik liikuda.

Linna 3D-mudeli üleminek staatilisest olekust dünaamilisse toimub Unity´s, kus lisatakse aktiivsed punktid, mis viitavad erinevatele lokatsioonidele. Nendes interaktsioonipunktides on vaatlejal võimalik saada teavet objekti kohta. Linnas liikumine toimub esimese isiku perspektiivis ning juhtimine toimub klaviatuuri ning hiire abil.

Enamus diplomitööst kirjeldab virtuaaltuuri loomise etappe valitud rakendustes. Igas peatükis on toodud näited ning valitud rakenduste võrdlus teiste sarnastega.

Tuuri loomise järjestus on järgmine:

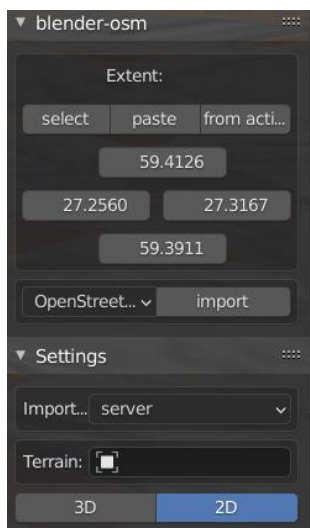
- linna kaardi loomine;
- hoonete 3D-mudelite loomine Blenderis;
- Virtuaaltuuri arendamine Unity´s.

## 2. LINNA KAARDI LOOMINE

Esimene etapp, mis virtuaaltuuri loomise raames teostati, oli kaardi loomine. Linna mudeli loomiseks on vajalik baas, millele toetudes saab luua 3D-mudelid võimalikult täpsete mõõtmetega ning suhtelise asukohaga. Linna modelleerimisel oli vajalik iga hoone suhteline asukoht linna piires. Antud lõputöö raames keskenduti kahele kaardi importimise võimalusele: *OpenStreetMap (OSM)* ning *Geographic Information System (GIS)* lisad.

### OSM

OSM on avatud lähtekoodiga laiendus, mis on välja töötatud Vladimir Elistratov poolt. Sellel on nii tasuta kui ka tasuline versioon. [9] Kaardi importimiseks kasutati tasuta versiooni. Selle kasutajaliides on intuitiivselt arusaadav (vt Joonis 2.1).



Joonis 2.1 OSM lisa

*Select* nupu vajutamisel suunatakse kasutaja veebileheküljele, kus on kujutatud maailma kaart. Kasutajal tuleb valida piirkond, mille ta soovib Blenderisse importida. Seejärel kopeerib ta soovitud asukoha koordinaadid ning sisestab neid OSM-i *Paste* nupu abil. Järgnevalt on võimalik valida, millist tüüpi kaarti soovetakse importida. Valikuid on neli: *OSM*, *terrain*, *image overlay*, *Global Positioning System Exchange Format (gpx)*. Esimene impordib antud juhul põhilisemad teed ning kõikide hoonete kas 2D- või 3D-mudelid. Hoonetel ei ole tekstuure, vaid ainult baasvärv. *Terrain* (maastiku objekt) loob valitud suurusega reljeefse pinna, mis on ka tekstuurita. Seega on antud töö raames selle kasutamine ebaefektiivne, kuna sellele tuginedes ei ole võimalik hooned linnas õigesse kohta asetada. Linna 3D-mudeli loomiseks kasutati *image overlay* meetodiga saadud pinna, mis impordib linna 2D-kaardi. Teede ning hoonete lisamiseks kasutati OpenStreetMap-i (vt Joonis 2.2) 2D-mudeleid. Antud 3D-mudeli loomisel aga ei ole kasutatud hoonete mudeleid, vaid need olid loodud iseseisvalt Google Maps-ist saadud

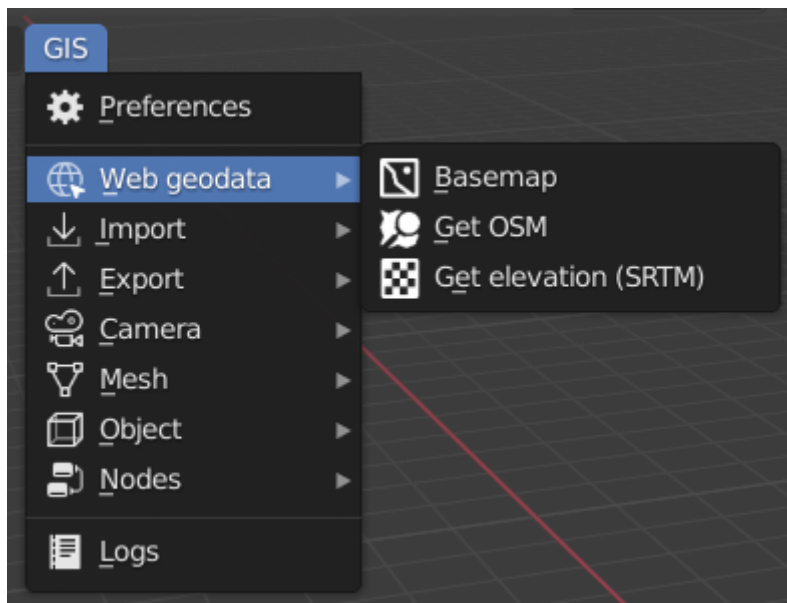
hoonete mõõtmete järgi. Kõrguseks oli vaikumisi seatud 3 meetrit ühe korruse kohta. Viimane valik, mida pakub *OSM*, on *gpx*, mis kujutab endast *Extensible Markup Language (XML)*-põhist tekstivormingut *Global Positioning System (GPS)*-andmete salvestamiseks, kuid selle lõputöö raames ei olnud vajadust seda tüüpi andmete importimise meetodit kasutada [10].



Joonis 2.2 Kohtla-Järve Järve linnaosa *OSM* abil imporditud kaart

## **GIS**

Teine võimalus kaardi importimiseks on *GIS* (vt Joonis 2.3). See laiendus impordib samuti valitud piirkonna kaarti Google satelliidilt. Erinevus on selles, et maailma kaart ilmub otse Blender rakendusse, mitte ei suunata kasutajat veebileheküljele. Valik *Basemap* lisab rakendusse kaardi, kust saab valida piirkonda. *G* klahviga on võimalik sisestada otsitav piirkond, *E* klahviga lõigata see välja. Lisaks on võimalik ka hooneid ning teid importida, nagu ka *OSM* laienduses, kuid 2D-formaadis hooneid ei teki, vaid ainult teed. Samas tekivad selle valikuga importides kaardile ebavajalikud piirkonnad, millel ei ole tekstuure ning ka teedel ei ole laiust (vt Joonis 2.4). Pinnale on võimalik ka reljeefust lisada, kuid antud töö raames otsustati mitte seda kasutada.



Joonis 2.3 GIS lisa



Joonis 2.4 Kohtla-Järve Järve linnaosa GIS abil imporditud kaart

Kahte laiendust võrreldes ning proovides mõlemaid, tuli autor järeldusele, et *OSM* sobib antud olukorras kaardi loomiseks paremini, kuna võimaldab rohkem valikuid, näiteks 3D- ning 2D-mudelite vahel valimist, ka teed on realistlikumad ning roheluse ja vee kohale tekivad vastavad baasvärvid, mida *GIS* ei võimalda.

## 3. BLENDER

Blender on tasuta 3D-objektide modelleerimise rakendus, mis oli loodud 2002. aastal. Selle abil saab luua animatsiooni, simulatsiooni, töödelda videot ning luua mängu. Blender sobib hästi individuaalseks või väikeettevõtetes kasutamiseks. Blender töötab nii Linuxi, Windowsi kui ka MacOS operatsioonisüsteemides. [11] See võimaldab luua nii madal- kui ka kõrgpolügonaalsete mudelid. Järgmises peatükis tutvustatakse põhjalikumalt Blenderis modelleerimise võimalusi.

### 3.1 Võrdlus teiste rakendustega

On olemas mitmeid rakendusi, kus saab modelleerida objekte. Kuna tellijaks on Virumaa Kolledz, siis programmide valik tuleneb tellija võimalustest. Aktiivselt on kasutusel kolm modelleerimise programmi: SketchUp Make, SolidWorks ning Blender. Autoril on kogemust lisaks eelnimetatud rakenduste kasutamisel ka 3D Studio Max-is. Järgnevalt võrreldakse rakendusi modelleerimise seisukohalt.

Autodesk 3D Studio Max on Autodesk toode, mis on loodud modelleerimise, arhitektuurse kujundamise ning projekteerimise eesmärgiga. Animeerimiseks kasutatakse seda programmi harvem. Selle esimene versioon oli välja lastud 1990. aastal. Programm on kasutatav Windows ja Mac operatsioonisüsteemides. 3Ds Max kasutajaliides on lihtsam kui Blenderi oma, kuid kommertseesmärgiga on selle tarkvara kasutamine tasuline. Üliõpilased ning õppejõud saavad seda õppeesmärgil kasutada tasuta kuni 3 aastat. See sobib suuremahuliste projektide loomiseks, Blenderit aga kasutatakse rohkem väiksemate ettevõtete või eraisikute poolt. Kõige populaarsem modelleerimise tehnika on polügonaalne modelleerimine, nagu ka Blenderis, ning hõlmab endas redigeeritavat pinda või hulknurka. Selle tehnika abil luuakse nii kõrg- kui ka madalpolügonaalsete mudelid. Rakenduses on olemas osakeste süsteem, mis võimaldab modelleerida vihma, lund, udu, tuld, vett jne Autodesk 3ds Max-is on 20 erinevat visualiseerimismoodulit. Enamus nendest on eraldi allalaetavad programmid. Blenderis on 2 enamkasutatavat renderdamismootorit: Eevee ja Cycles. [12]

SketchUp on 3D-mudelite loomise programm, mida kasutavad enamasti arhitektid ja sisekujundajad. See oli loodud 2000. aastal ning oli tasuta. Ka praegu saab seda kasutada tasuta (SketchUp Make), kuid on ka tasuline versioon (SketchUp Pro), mis annab rohkem võimalusi. Nii Blenderis kui ka SketchUp-is saab luua fotorealistlikke 3D-mudeleid, kuid viimane piirdub vähese tööriistade arvuga, Blenderis aga on neid tunduvalt rohkem. See võib olla ka SketchUp-i boonus, kuna kasutajaliides on selles rakenduses lihtne ning arusaadav. Mõlemad rakendused pakuvad visualiseerimise pilveteenust, mis säästavad renderi loomise jaoks nõutavat aega. Kuna aga SketchUp-i rakenduse eesmärk erineb Blenderi omast, siis on ka siin veidi teistsugused funktsioonid olemas. SketchUp võimaldab hoonete struktuuri- ja päevavalguse



analüüsi. Lisaks võimaldab see luua täiuslikke kahedimensioonilisi ehitusjooniseid. [13] SolidWorks sobib eelkõige tööstuses kasutatavate esemete modelleerimiseks. Nagu ka eelnimetatud rakendused, on SolidWorks tasuta õppeasutuste jaoks, muul juhul on see tasuline. Kuna SolidWorks kasutab täpseid mõõtmeid ning on loodud enamasti inseneride jaoks, siis antud diplomitöös käsitletud virtuaaltuuri loomise jaoks see ei sobi.

Kuna antud töö raames oli autor huvitatud vaid modelleerimise võimalustest ning mugavusest, vaatamata laiale valikule, mis kõik eelnimetatud rakendused pakuvad, osutus valituks Blender. Olulisim valiku põhjendus on varasem tihedam kokkupuude selle rakendusega. Kiirklahvid, objektidega manipuleerimine, põhilised modifikaatorid ning tekstuuridega töötamine on juba omandatud antud programmis, seetõttu nõuab diplomitöö praktilise osa koostamine tunduvalt vähem aega. 3D Studio Max rakenduses on autor modelleerinud vaid ühe töö, seega on kokkupuude pinnapealne. SketchUp pakub autori arvates vähem valikuid.

## **3.2 Funktsionaalsus**

Blender on 3D-mudelite loomise rakendus, mis pakub rohkelt erinevaid valikuid. Antud lõputöö raames piirdub rakenduse ülevaade vaid hoonete modelleerimise kirjeldamisega, kuna seda on kasutatud vaid selle eesmärgiga, kirjeldades detailsemalt ühe hoone loomise kõiki etappe.

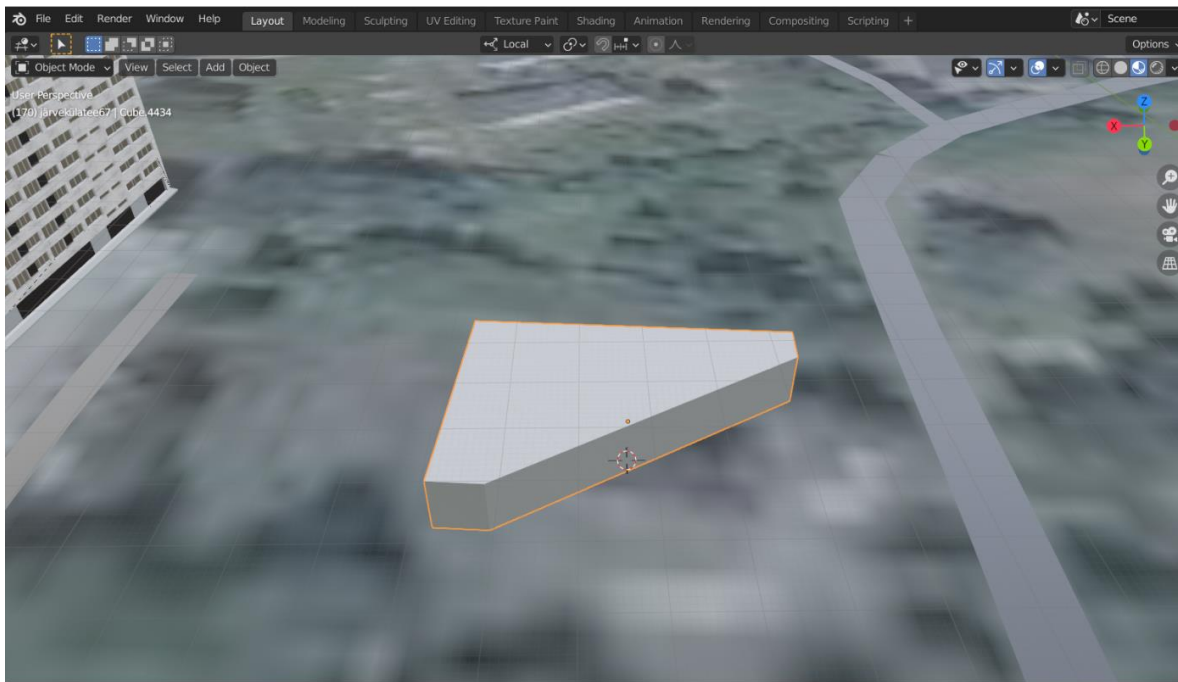
Objektiks on Kohtla-Järve Gümnaasium. Hoonete mõõtmete saamiseks kasutati Google Maps joonlauda, aga kuna eelnimetatud hoone on uus, ei ole selle mõõtmeid kaardil veel olemas. Blenderisse sisestati endise hoone mõõtmed ning vajadusel muudeti neid veidi. Aluseks oli võetud kooli olemasolev 3D-mudel, mis on vabalt kättesaadav (vt Joonis 2.1).



Joonis 2.1 Kohtla-Järve Riigigümnaasium [14]

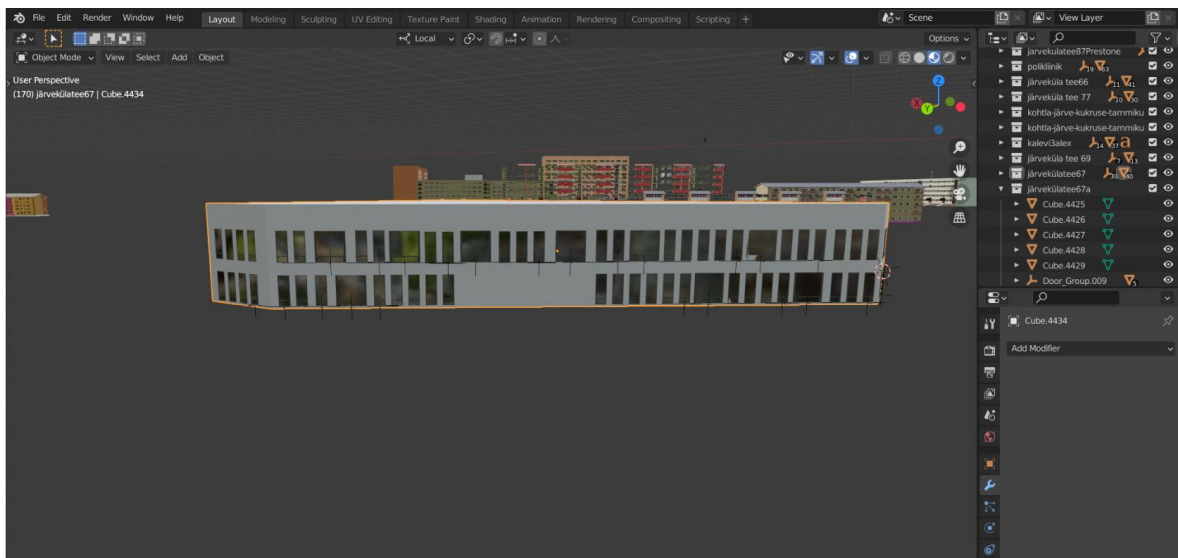
Kõikide hoonete loomise aluseks on kuubi lisamine. *Shift+A* klahvide vajutamine toob esile menüü, kus saab luua erinevaid kujundeid. Kuna antud hoone on viisnurk, siis oli vaja redigeerimisrežiimis (*Edit Mode* ning *Object Mode* vahel liikumiseks kasutatakse klahvi *TAB*) luua kujundile juurde külgi. *Ctrl+R* klahv lõikab kujundit, tekitades soovitud arvu külgi ja tippe.

Üleliigsete tippude kustutamise järel tekkis soovitud kuju ning külgede arvuga kujund (vt Joonis 2.2).



Joonis 2.2 Kooli baaskujund

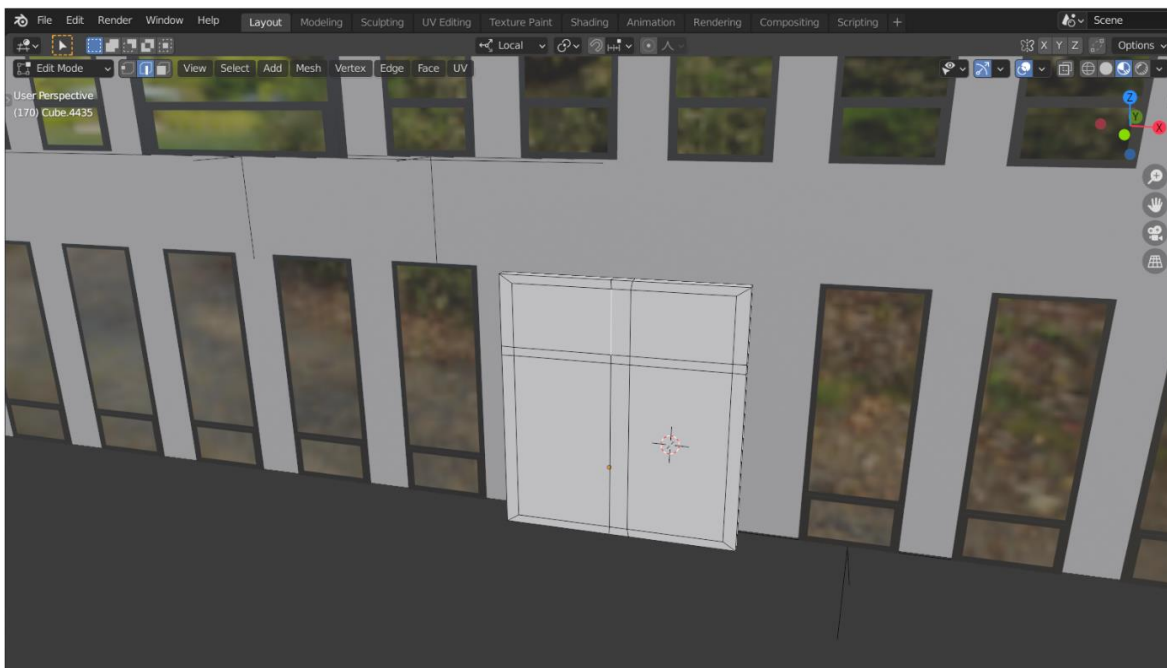
Akende loomiseks kasutati Blenderisse sisseehitatud paketti nimetusega *Archimesh*. Selles on valmis modelleeritud sellised üldkasutatavad objektid nagu aken, uks, katus, trepiastmed, riiulid (vt Joonis 2.3). Objektide mõõtmeid saab soovi korral muuta, lisaks on erinevaid võimalusi nende modifitseerimiseks, näiteks aknaruutude arvu võib suurendada, aknaraami tihedust reguleerida, soovi korral võib aknalauda eemaldada.



Joonis 2.3 Kooli akende loomine

Kuna soovitud kujuga ust ei ole *Archimesh* pakettis olemas, siis lõi autor selle ise (vt Joonis 2.4). Ukse kuju aluseks on kuup, mille mõõtmeid saab nii X-, Y- kui ka Z-telje suunas muuta. *Ctrl+R* klahvi abil lõigati uksest välja vajaliku arvu pindasid. *E* klahv

pressib pinna kas sisse või välja. Klaaside tegemiseks ukse sees kasutati sissepressimist. Üleliigsed jooned kujundi peal kustutatakse ära klahvi *X* abil redigeerimisrežiimis, valides *Dissolve Edges*.



Joonis 2.4 Kooli tagumise ukse loomine

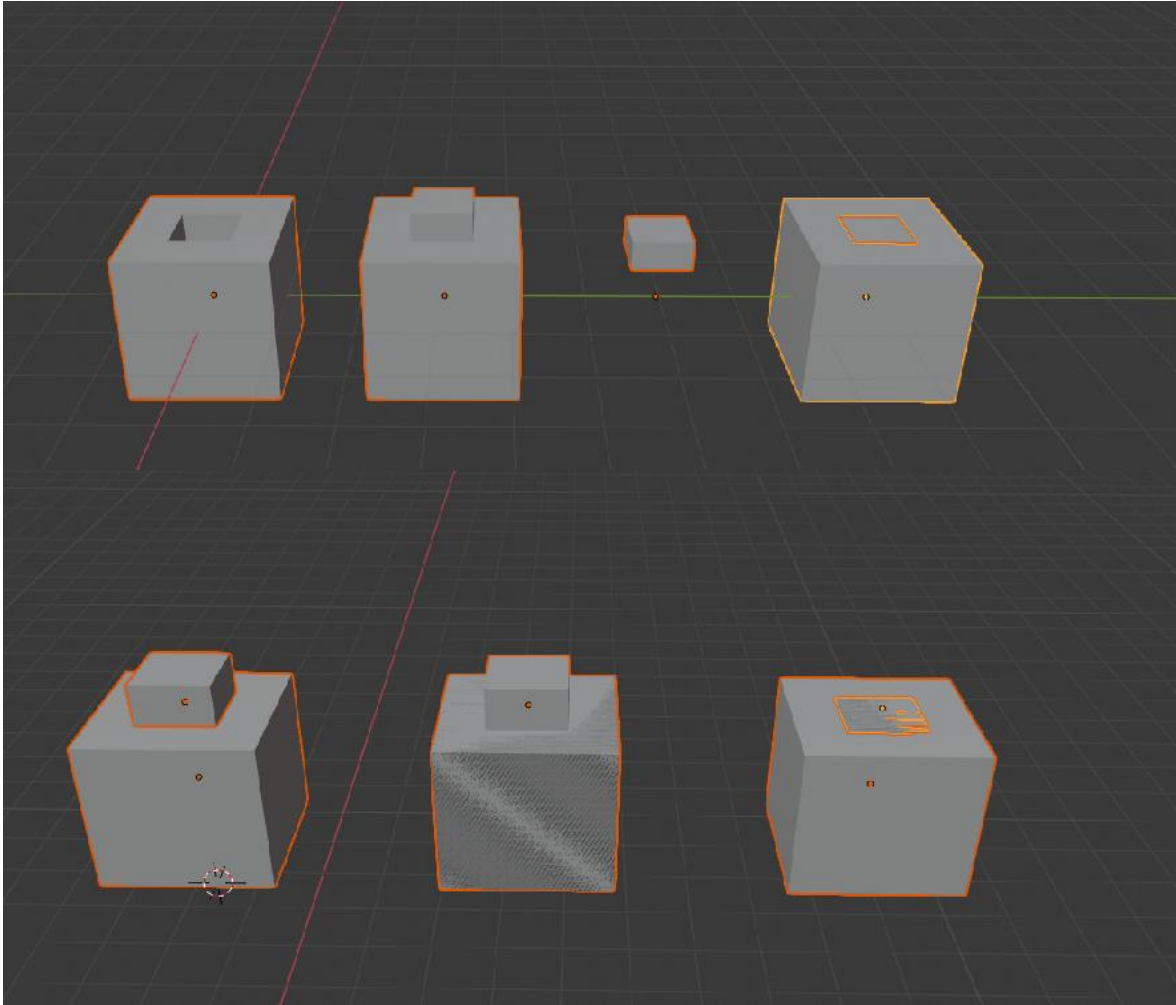
Ukse käepidemed olid loodud ringi kujundi abil, valides tippude arvuks 32. Poolringi saavutamiseks kustutati pooled tipud ära. Allesjäänud kujundile oli rakendatud käsku *I* klahvi abil, mis lõi poolringi sisse veel ühe väiksema poolringi. Sisemine pind kustutati ära, alles jäänud kujundi pressiti välja. Pinnale sileduse lisamiseks kasutati käsku *Shade Smooth* objekti kontekstmenüüst, kuhu pääseb hiire paremale klahvile vajutades (vt Joonis 2.5). Teise käepideme võib luua erinevate viisidega: kas massiivi, peegeldamise või dubleerimise abil. Antud juhul otsustati kasutada dubleerimist *Ctrl+D* klahvide abil. Seda tüüpi dubleerimine ei suurenda faili mahtu, kuna uus objekt pärandab kõiki muutusi, mida teostatakse algobjektiga ning vastupidi. Teine dubleerimisvõimalus on *Shift+D* klahvide abil, mille puhul tekib kaks teineteisest sõltumatut objekti.



Joonis 2.5 Ukse käepideme loomine

Sama põhimõttega, mida eelnevalt on kirjeldatud, luuakse hoonele katuse piir ning sissepääs. Aknad luuakse massiivi modifikaatori abil, sisestades soovitud akende arvu kas X-, Y- või Z-telje suunas ning nendevahelise vahemaa. Vajadusel lisatakse mitu massiivi modifikaatorit, kui on vaja ühte akent paljundada nii mitme telje suhtes. Kui enam ei ole vaja teha muudatusi modifikaatorites, tuleb need rakendada, vajutades nuppu *Apply*.

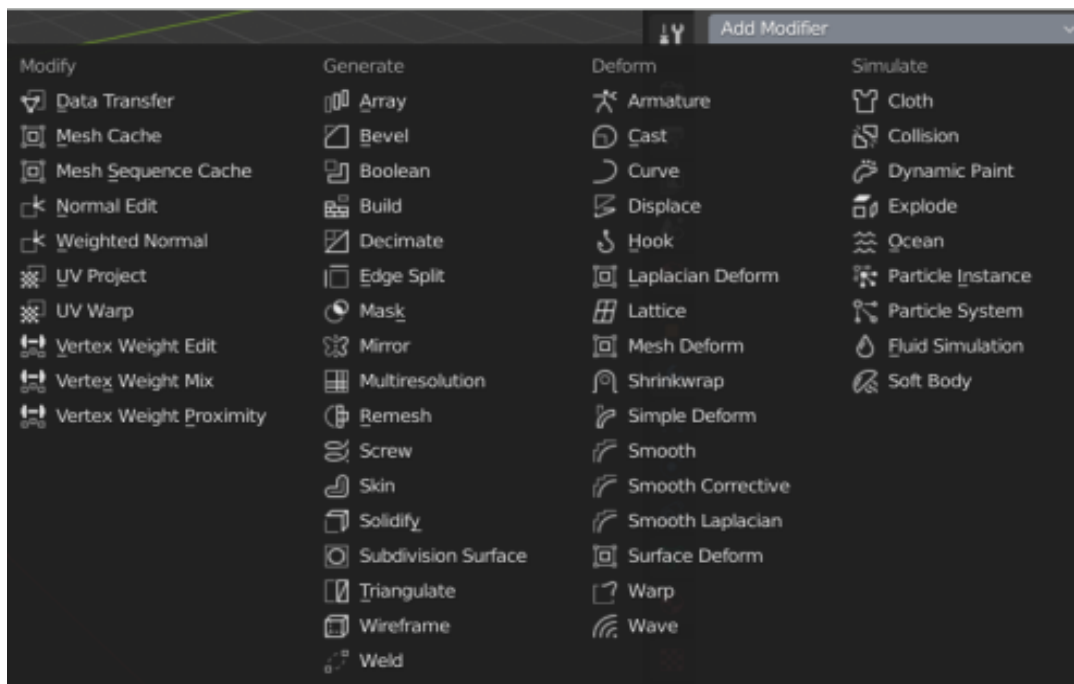
Hoone sisse süvendite loomine akende jaoks teostati *Boolean* funktsiooni abil. Blenderis on kahte tüüpi *Boolean*-i. Üks on modifikaator, teine on laiendus nimega *Bool Tool*. Antud töö raames kasutati enamasti laiendust, kuna modifikaatori tulemusena loodud kujund ei olnud alati rahuldav. Lisaks on *Bool Tool* tööriistal ühe võrra rohkem valikuid võrreldes modifikaatoriga (vt Joonis 2.6).



Joonis 2.6 *Boolean* funktsiooni kasutusvõimalused: ülemisel joonisel on loodud kokku 8 kuupi, millest 4 on suuremad ning 4 väiksemat nende peal. Kasutatud on *Bool Tool* tööriista. Vasakult paremale on kuupidega teostatud järgmisi funktsioone: *difference*, *union*, *intersect*, *slice*. Alumisel joonisel on 6 kuupi ning kasutatud on modifikaatorit, milles on 3 valikut: *difference*, *union*, *intersect*.

Blender pakub lisaks eelnimetatule suure hulga erinevaid modifikaatoreid (vt Joonis 2.7). Modifikaatorid on automaatsed toimingud, mis mõjutavad objekti geomeetriat. Nende abil saab kiirendada toiminguid, mis käsitsi sooritades võtaksid palju aega. Ühel objektil saab kasutada mitu modifikaatorit samaaegselt ning rakendada neid, kui muutusi enam ei tehta. Muutused jäävad sel juhul püsivaks.

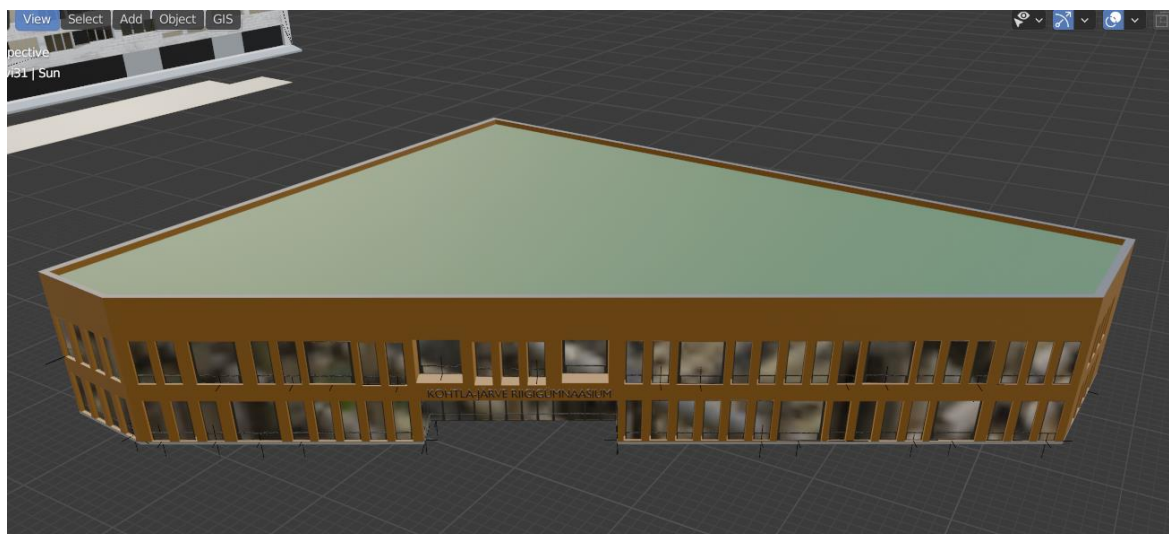
Lisaks enamkasutatavale massiivi ning boolean modifikaatorile, oli kolmandal kohal peegeldus, mis võimaldab peegeldada valitud objekti redigeerimisrežiimis piki tema enda telge ning objekti alguspunkti suhtes.



Joonis 2.7 Modifikaatorite loetelu Blenderis

Materjalide loomiseks kasutati Blenderi sisseehitatud tekstuure, sest neid on võimalik automaatselt Unity´sse importida. Kuna tegemist on madalpolügonaalset linna mudeliga ning eesmärgiks ei olnud realistlikkus, vaid linna üldvaate loomine, siis kasutati hoonete värvimisel enamasti baasvärve.

Hoone peale teksti kirjutamise põhimõte on sama, nagu ka teiste objektide loomisel. Blenderis on olemas teksti lisamise valik. Teksti kirjutamine toimub redigeerimisrežiimis, suurust ning materjale on võimalik muuta vastavalt vajadusele (vt Joonis 2.8)



Joonis 2.8 Kooli lõplik vaade

## 4. UNITY

Unity on mängumootor, mis on välja töötatud Unity Technologies poolt 2005.aastal. Seda mootorit võib kasutada kahe- ja kolmedimensiooniliste, virtuaalreaalsuse ning liitreaalsuse mängude loomiseks. Rakenduse põhiline programmeerimiskeel on C#. Unity pakub rohkem kui 25 platvormi tuge, mille hulgas on Windows, Mac, Android, iOS ning Linux. Mootorit kasutatakse tänapäeval lisaks videomängude loomisele ka sellistes harudes nagu filmi- ja autotööstus, arhitektuur, inseneritöö ja ehitus. Unity´l on nii tasuta kui ka tasulist versiooni. Tasuta versiooni võib kasutada kas väikeettevõtetel või isiklikel eesmärkidel, kui iga-aastane tulu loodud mängude müümisest ei ületa \$100,000. Tasulise versiooni maksumus arvutatakse personaalselt, sõltuvalt Unity baasil loodud mängude müümisest tulust. [15]

### 4.1 Võrdlus teiste rakendustega

Mängumootorid on korduvkasutatavad komponendid, mida arendajad kasutavad mängu raamistiku loomiseks [16].

Antud peatükis tuuakse lühiülevaade teistest populaarsetest mängumootoritest, mida võib kasutada nii virtuaalruumi kui ka erinevate mängude loomiseks.

Unreal Engine on mängumootor, mis on välja töötatud Epic Games poolt. Esimene seal loodud mäng ilmus 1998. aastal. Mootor on kirjutatud C++ keeles ning toetab paljusid platvorme. Seda kasutatakse lisaks mängude loomisele ka filmi- ja teletööstuses. Alates 2015. aastast saab seda alla laadida tasuta ning rakenduse looja nõudis 5% müügitulust, kuid hiljem teatas, et mängudele, mis on avaldatud läbi Epic Games Store, ei laeku lisatasusid. Seda tänu mängu *Fortnite* suurele edule, mis oli loodud antud mängumootoril. Alates 2020. aastast uuendati reegleid – ei ole vaja maksta tasusid seni, kuni arendaja ei teeni oma esimese miljoni dollari. Rakenduse viimane versioon, Unreal Engine 5, toetab kõiki olemasolevaid süsteeme, kaasa arvatud PlayStation 5 ja Xbox Series X. [17] Selle mängumootori tugevus on selles, et seda saab muuta nii palju, et iga mäng muutuks unikaalseks. Küll aga on algajatele raske seda kasutada, selleks on loodud lihtsamad mängumootorid nagu Unity ja teised. [16]

Cry Engine on 2002. aastal saksa eraettevõtte Crytek poolt loodud mängumootor. Mänge saab luua suure hulga platvormide jaoks: Microsoft Windows, Linux, Nintendo Switch, PlayStation 3 ja 4, Wii U, Xbox 360 ja One, 2020. aastal lisandus ka Androidi testprogramm. Enne 2018. aastat oli seda võimalik kasutada tasuta, peale seda aga lisati 5% müügitulu maksmise tingimus. Programmeerimiskeel on C++. [18]

Viimane mängumootor, mida antud töö raames tutvustatakse, on GameMaker, mis on laialdaselt kasutusel, kuna ei nõua programmeerimiskust. Mängud selle mootori peal on loodavad paljude platvormide jaoks. kasutatakse enamasti 2D mängude loomise jaoks, kuna sellel on piiratud võimalus 3D graafika kasutamiseks. [19]



Virtuaaluuri on võimalik luua ka eelnimetatud rakendustes, kuid Unity´ s on see tehtud lihtsamaks tänu valmislahenduste ning -objektide olemasolule. Sarnase mahuka projekti loomine nõuab seetõttu tunduvalt vähem aega.

## 4.2 Funktsionaalsus

Antud alapeatükis tutvustatakse Unity neid võimalusi, mida oli virtuaaluuri loomise jaoks kasutatud. Detailsem ülevaade koos näidetega on järgmises peatükis.

Unity's on lai valik nii tasuta kui tasulisi mudeleid ning tekstuure. Need on saadaval *Asset Store´ s*, mis kujutab endast tekstuuride, animatsioonide, 2D- ja 3D-mudelite, audiofailide, erinevate valmislahenduste ja õppematerjalide kogu. Selliseid lahendusi loob Unity Technologies ning selle kogukonna liikmed. Paljud objektid on tasuta. On võimalik ka iseseisvalt luua midagi ning müüa seal. [20]

Lisaks on võimalik luua baaskujundeid, näiteks kuup, sfäär, silinder. Ka saab lisada audio- või videofaile. Selle töö raames kasutati maastiku objekti, millele saab lisada tekstuure, rohtu, puid, põõsaid. Eelnimetatud objekte saab lisada ka käsitsi kaustast üle tuues, kuid see on aeganõudvam, kuna need tuuakse üle ühekaupa ning kopeeritakse klahvide *Ctrl+D* abil.

Objektile saab lisada erinevaid atribuute *Add Component* nupu abil: animatsioon, audio- ja videosalvestused, pildid, füüsikaline käitumine nii 2D- kui ka 3D-objektide jaoks (näiteks jäik keha).

Objekte saab stseenile üle via *Assets* kaustast. Nendega saab teostada transformatsioone: muuta suurust, pöörata erinevate telgede suhtes, liigutada. Mängurežiimi aktiveeritakse *Play* nupu abil kasutajaliideses. Projekti loomise sätetes saab valida, millise platvormi jaoks see luuakse: kas arvuti-, veebi- või telefonirakendusena, ka mängukonsooli jaoks. Unity pakub ka iseseisvalt koodi kirjutamise võimalust ning selle rakendamist mängudes ja muudes projektides. Kõiki eelnimetatud funktsioone on kasutatud virtuaaluuri loomise jaoks.

## 5. VIRTUAALTUURI ARENDAMINE

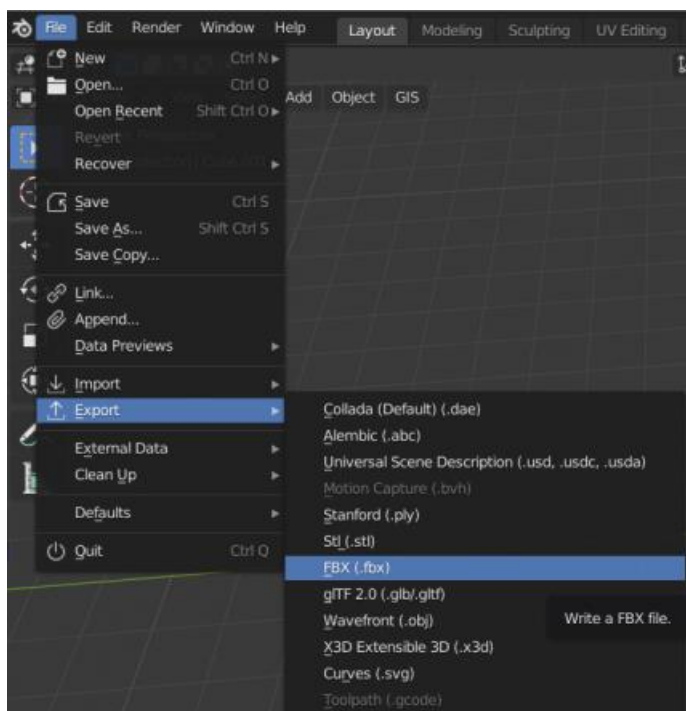
Virtuaalne 3D-keskkond võimaldab kogeda keskkonda, kuhu muidu oleks juurdepääs piiratud kas rahaliste kulude, keskkonnaohtude või muude piirangute tõttu [1]. Antud lõputöö raames loodi Kohtla-Järve linna osa madalpolügonaalne 3D-mudel, et vaatleja saaks virtuaalselt tutvuda olulisemate linna vaatamisväärsustega. Eelnevalt kirjeldati 3D-mudeli ning kaardi loomise etapid. Järgnevalt tutvustatakse virtuaaltuuri loomist Unity´s.

### 5.1 3D-mudeli Unity´sse importimine

Kogu linnamudelit on võimalik Blenderist eksportida erinevates formaatides: *.dae*, *.abc*, *.bvh*, *.svg*, *.ply*, *.obj*, *.fbx*, *.glb*, *.blend*, *x3d*. Unity poolt on vastuvõetavad *.obj*, *.blend*, *.fbx*, *.dae*.

Antud lõputöö raames imporditi mudel kahes enamkasutatavas formaadis: *.blend* ja *.fbx*. Kõige lihtsam on importida *.blend* formaati, kuna sel juhul tuleb fail lihtsalt üle viia Unity projekti kausta. Tehes muudatusi Blenderis, levivad need ka Unity´sse. [21]

Selleks, et importida faili formaadis *.fbx*, tuleb esialgu Blenderi fail eksportida formaati *.fbx*, mida saab teha menüüs (*File – Export- FBX(.fbx)*) (vt Joonis 5.1). Järgnevad etapid on samad, mis *.blend* formaadi puhul – tuleb fail üle viia Unity projekti kausta *Assets* ning Unity´s see stseenile üle kanda. [22]



Joonis 5.1 Blenderi faili eksportimine .fbx formaadiks.

Unity´ s on proovitud kasutada mõlemaid eelnimetatud formaate. Valituks aga osutus *.blend* formaat, kuna see on vähem mahukam ning muudatuste vajaduse tekkimise korral on need teostatavad automaatselt ka Unity´ s.

Baasvärvid, mida kasutati hoonetel, on automaatselt Unity´ sse integreeritavad. Need tekstuurid, mis on Blenderisse imporditud (näiteks *image texture*), tuleb Unity projekti *Assets* kausta üle viia.

Blenderis loodud mudeli importimine Unity´ sse võib tunduda lihtne, kuid võivad tekkida mõned raskused, kuna nende kahe rakenduse vahel on erinevusi. Esimene erinevus on koordinaatide süsteemis. Blender kasutab paremakäelist koordinaatide süsteemi, mis tähendab, et Z-telg on vertikaalne. Unity kasutab vasakukäelist, kus vertikaalne on Y-telg. [21] Korrekse importimise jaoks on vajalik Blenderi kõikidele objektidele määrata lähteväärtused, st rotatsioon, asukoht ruumis peavad omama väärtust 0, mõõtmete väärtused peavad olema 1. Kõik hooned asetati ühele tasapinnale. Selle eelduseks oli, et kõikide hoonete asukoht Z-teljel on 0. Järgnevalt tuleb rakendada objektidele *Object-Apply-All Transforms to Deltas*, mis määrab neile vaikeväärtused.

Blenderi failis on vaja ära kustutada kaamera ning valgustus, kuna see võib segada esimese isiku kaamerast vaatamist Unity´ s. Projekti käivitamisel võib üleliigne kaamera takistada õige vaatevälja nägemist.

## 5.2 Puuduolevate objektide lisamine Unity´ s

Antud lõputöö raames keskendutakse 3D-mudelitele ning 2D-tekstuuridele. Unity poes leidub transpordi, tegelaste, dekoratsioonide, taimestiku 3D-mudeleid, animatsioone, video- ja audiofaile, tekstuure. Üheks kõige populaarsemaks tasuta 3D-mudelite komplektiks on *Nature Starter Kit 2*, mis sisaldab puid ja põõsaid, millest on võimalik luua erinevaid variatsioone. Käesoleva töö jaoks kasutati kaasa arvatud ka seda komplekti (vt Joonis 5.2). [20]

Valitud objekt tuleb esialgu alla laadida kas rakendusest või veebileheküljelt, seejärel ilmub nupp, mis lubab mudeli importida otse projekti. Avaneb nimekiri, kust saab valida, milliseid objekte ning tekstuure importida.



Joonis 5.2 Objektide lisamine stseenile

Lisaks rohelusele kasutati ka teisi valmislahendusi. Nendeks on mänguväljakud, pingid, tänavalaternad, autod, liiklusmärgid.

Unity´s on olemas *terrain* ehk maastiku 3D-mudel, mida kasutati maastikuna ka antud tuuri loomisel. See võimaldab teostada erinevaid manipulatsioone. Esiteks, maastikut on võimalik tõsta või alandada teatud kohtades (lisada reljeefsust), siledaks teha, erinevaid tekstuure lisada, nupuvajutusega lisada rohkelt puid, põõsaid ning rohtu teatud kohtadesse. Kõiki neid tegevusi tehakse pintsliga, mille suurust ning objektide lisamise tihedust on võimalik muuta. Tekstuure on võimalik samuti laadida alla poest. Maastiku tekstuurid (roheline, asfalt) olid tasuta lisatud projekti.

### 5.3 Esimese isiku vaatenurga lisamine

Esimese isiku vaatenurga lisamiseks kasutati valmislahendust, mis on tasuta kättesaadav Unity poest. Seda laetakse alla *Assets* kausta ning esimese isiku mudel lisatakse stseenile. Kaamera alguspunkti on võimalik liigutada kõikide telgede suunas, mängija algkiirust saab muuta samuti.

Liikumine tuuri ajal toimub klaviatuuri noolte abil, rotatsioonid sooritatakse hiire abil ning hüpata saab tühikuga. Lisaks on ekraani keskel kursor, millega saab vajutada interaktsioonipunktidele.

Et vaatlejal ei oleks võimalik liikuda läbi hoone või muu objekti, rakendatakse nii kaamerale kui objektile *Collider*. See määrab objekti kuju füüsiliste kokkupõrgete ajal ning ei pea omama objekti kuju [23]. Esimesel isikul on see juba vaikimisi sisseehitatud, mõnedel Unity poest imporditud objektidel samuti. Kui seda ei ole, siis on võimalik see käsitsi objektile lisada *Inspector* alast, valides *Add Component* menüüst sobiva *Collideri*. Kõige lihtsamad ning ühtlasi ka vähem protsessori ressursi tarbivad 3D-*Colliderid* on

kuubi-, sfääri- või kapslikujulised, mida on ka antud projektis kõige rohkem kasutatud.

## 5.4 Interaktsioonipunktide lisamine

Interaktsioonipunktid annavad lisateavet linna kohta. Antud lõputöö raames kasutati aktiivseid punkte, mis asuvad olulisemate hoonete juures, ning millele vajutades saab vaateleja kas audio- või videosalvestuse abil lisateavet objekti kohta. Antud peatükis kirjeldatakse Virumaa kolledzit tutvustava video lisamist interaktsioonipunkti.

Punkti loomiseks kasutati *User Interface (UI)* nupu mänguobjekti. *UI* kujutab endast tööristakomplekti, mille abil saab luua mängu menüüd ning teisi kasutajaliideseid [24]. Nupu kuju on võimalik muuta kas sisseehitatud või Unity poest imporditud piltide abil. Antud diplomitöös kasutati tasuta poest imporditud nuppude komplekti, kust valituks osutus küsimärgi kujutis (vt Joonis 5.3).



Joonis 5.3 Virumaa kolledzi interaktsioonipunkt

Video lisamiseks loodi objekt *Video Player*, kuhu *Assets* kaustast lisati vajalik video. Selleks, et videot oleks võimalik korduva nupule vajutamisega peatada, tuleb nupule lisada uus skript. Unity's on olemas sisseehitatud funktsioonid, mida on võimalik kasutada koodi kirjutamisel. Antud juhul oli kood kirjutatud, kasutades funktsioone *Play()* ja *Pause()*. Videosalvestust on võimalik kujutada ka mõnel 3D-objektil.

Audiosalvestuse lisamine on analoogne video lisamisele. Audiofail lisatakse mängukomponendi *Audio Source* sisse. Audio loomiseks salvestati dikteeritud tekst tasuta helisalvestuse tarkvaras.

Lisaks eelnimetatud aktiivsetele punktidele, millele on võimalik vajutada, on stseenile loodud ka 3D-osutid Blenderis, mis näitavad olulisemate hoonete asukohti ning peatänavate suunda (vt Joonis 5.4).



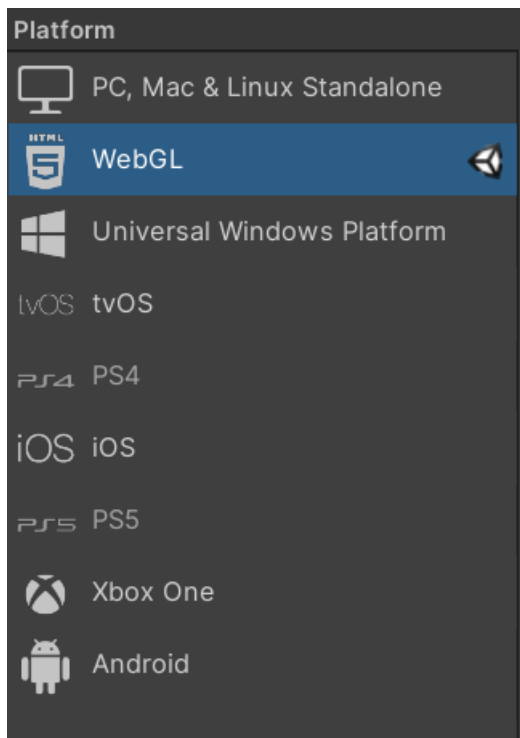
Joonis 5.4 Osuti, mis näitab peatänavate suunda

Interaktsioonipunktide olemasolu võimaldab lihtsamalt tutvuda linnaga, kuna nende abil saab infot objektide kohta. Tänu osutite olemasolule on linnas jalutades raskem eksida.

## 5.5 Projekti lõpetamine

Unity pakub erinevaid platvorme, kus saab loodud projekti kasutada (vt Joonis 5.3). Antud lõputöö raames otsustati luua veebi- ning konsoolirakendus.

Projekti ehitamisel tuleb muuta selle sätteid: vaikimisi on seatud, et projekt käivitatakse personaalarvutil. Veebiversiooni loomiseks tuleb valida *WebGL*. Peale seda käivitub tuur lokaalses võrgus.



Joonis 5.3 Platvormi valimine Unity´s

Et tuur oleks nähtav ja kättesaadav igaühele, tuleb kasutada veebilehekülgi, mis võimaldavad tasuta Unity´s loodud mängu üles laadida. Antud lõputöö raames valminud tuur on kättesaadav lingilt <https://simmer.io/@vladlenautt/linn>.

Konsoolirakenduse loomine on sarnane eelmisele: platvormiks valitakse *PC, Mac & Linux Standalone* ning Unity ehitab projekti iseseisvalt loodud kausta, kust seda saab käivitada.

Kuna fail osutus liiga mahukaks, kustutati töö valmides kõik objektid *Assets* kaustast, mida ei ole kasutatud.

## KOKKUVÕTE

Käesoleva lõputöö eesmärgiks oli luua Kohtla-Järve Järve linnaosa 3D-mudelist virtuaaltuur, kus saab ringi liikuda, hooneid vaadelda ning tähtsamate ajalooliste objektide kohta lisainformatsiooni lugeda või kuulata.

Töös kirjeldati rakendusi, milles antud töö oli loodud ning lisaks anti lühiülevaade ka teistest sarnastest rakendustest. 3D-mudeli loomise rakenduseks oli Blender ning virtuaaltuur oli loodud Unity´s. Mudel oli eksporditud Blenderist *.blend* formaadis ning Unity´sse imporditud. Autor põhjendab, miks osutusid valituks just eelnimetatud rakendused ning faili formaat. Lisaks anti lühiülevaade sellistest rakendustest nagu 3D Studio Max, SketchUp, SolidWorks, Unreal Engine, CryEngine, VR Google.

Töö autor kirjeldas virtuaaltuuri loomise kõiki etappe ning raskusi, mis tekkisid selle loomise käigus. Probleemid, mis tekkisid lõputöö praktilise osa koostamise ajal:

- Mahuka faili käsitlemine;
- Üleliigsete polügonide kustutamine;
- Tekstuuride importimine Blenderist Unity´sse.

Kõikidele probleemidele oli pakutud ka omapoolne lahendus. Kõige mahukam osa oli mudeli loomine Blenderis. Kõik hooned on unikaalsed, madalpolügonaalsed mudelid. Linna kaart oli imporditud *OSM* abil. Ülejäänud detailid, nagu puud, põõsad, valgustus, olid võetud Unity poest.

Töö täitis oma eesmärgi, kuna töö tulemusena valmis Kohtla-Järve linna osa virtuaaltuur, mida on võimalik alla laadida rakendusena kui ka veebis kasutada. Antud tööd on võimalik ka edasi arendada, lisades 3D-mudelile rohkem detailsust. Kuna antud mudel on madalpolügonaalne ning suure töömahu tõttu ei sisalda ta väiksemaid detaile ning ka kõiki teid ei ole kujutatud, siis oleks edasiarenduse võimaluseks antud mudeli täiustamine.

Joonised, millele ei ole viidatud, on autori poolt tehtud.



## **SUMMARY**

The goal of this thesis was to develop a virtual tour for a 3D model of Kohtla-Järve city and to add active points which would provide additional information about important city buildings. The virtual tour can be experienced both on the web and as a console application. It can be navigated by using mouse and keyboard.

Different types of virtual tours were described in the thesis, as well as various modelling applications and game engines. Applications for creating virtual tours are for example Klapy and Google VR. In addition, a brief overview of such 3D modelling applications as Blender, 3D Studio Max, SketchUp Make, SolidWorks was provided. Game engines that were described in this thesis are Unity, Unreal Engine, Cry Engine and GameMaker.

The basis for creating 3D models was importing the map of the city. OpenStreetMap add-on was used for this purpose. Blender was used to create a 3D model of the city. All the models are unique and low-polygonal. All the steps of creating a tour in Unity are described in the thesis. Some objects, for example benches, lamps, cars and playgrounds were imported from Unity Asset Store.

Virtual tour has active points where user can get more information about important buildings in the city. Active points have video and/or audio recordings, which start working by mouse clicking on them. There are also signposts on the crossroad where the directions of the main streets are indicated. The last ones were modelled in Blender.

The thesis fulfilled its goal, because as a result, a virtual tour was developed. The virtual tour can also be further developed by adding more details to the 3D model of the city.

## KASUTATUD KIRJANDUSE LOETELU

1. Technopedia Inc. Virtual Tour. [Online] <https://www.techopedia.com/definition/4920/virtual-tour> (13.02.2021).
2. Klapy. Virtual Tours. [Online] <https://www.klapy.com/> (13.02.2021).
3. Hernandez, A. The Best 10 Free and Open-Source Virtual Tour Software Solutions. [Online] <https://www.goodfirms.co/blog/best-free-open-source-virtual-tour-software-solutions> (15.02.2021).
4. Virtual Tours Easy. Virtual Tour vs Video Tour: Identifying What's Best for Real Estate Marketers. [Online] <https://blog.virtualtoureasy.com/7620-virtual-tour-vs-video-tour/> (23.03.2021).
5. TalTech. [Online] <https://www.tuur.ee/taltech/> (05.05.2021)
6. CL3VER Apartment Interior Virtual Tour. [Online] <http://3d.cl3ver.com/19PxLY> (05.05.2021)
7. Google. Augmented Reality, Virtual Reality. [Online] <https://arvr.google.com/> (11.04.2021).
8. Google. Create a virtual tour. [Online] <https://arvr.google.com/tourcreator/> (11.04.2021).
9. Blender Osm. [Online] <https://awesomeopensource.com/project/vvoovv/blender-osm> (20.04.2021).
10. Hazzard, C. What is a GPX File? [Online] <https://hikingguy.com/how-to-hike/what-is-a-gpx-file/> (25.04.2021).
11. Blender Foundation. About. [Online] <https://www.blender.org/about/> (20.03.2021).
12. Petty, J. What is 3ds Max & What is it Used For? [Online] <https://conceptartempire.com/what-is-3ds-max/> (25.04.2021).
13. Graham, C. Sketchup Vs Blender | What's the Difference? [Online] <https://www.buildercentral.com/sketchup-vs-blender/#:~:text=Sketchup%20and%20Blender%20are%20entirely,two%20very%20different%20software%20programs>. (11.04.2021).
14. Liiva, S. Kohtla-Järve Riigigümnaasiumi ehitab Nordecon. [Online] <https://majandus.postimees.ee/6404562/kohtla-jarve-riigigumnaasiumi-ehitab-nordecon> (25.04.2021).
15. Unity. [Online] <https://unity.com/> (26.04.2021).
16. Dustin, K. The Power of Video Game Engines: Every Game Developer's (Not-So-Secret) Weapon. [Online] <https://www.gamedesigning.org/career/video-game-engines/> (29.04.2021).
17. Unreal Engine. [Online] [https://en.wikipedia.org/wiki/Unreal\\_Engine](https://en.wikipedia.org/wiki/Unreal_Engine) (05.04.2021).
18. Dealessandri, M. What is the best game engine: is CryEngine right for you? [Online]

- <https://www.gamesindustry.biz/articles/2020-01-16-what-is-the-best-game-engine-is-cryengine-the-right-game-engine-for-you> (15.04.2021).
19. GameMaker Studio 2. [Online] <https://www.yoyogames.com/ru/gamemaker> (20.04.2021).
  20. Unity Tehcnologies. Quick guide to the Unity Asset Store. [Online] <https://unity3d.com/ru/quick-guide-to-unity-asset-store> (04.04.2021).
  21. Carolo, L. Blender to Unity: How to Import Blender Models in Unity. [Online] <https://all3dp.com/2/blender-to-unity-how-to-import-blender-models-in-unity/#:~:text=To%20import%20the%20model%20to,Assets%20panel%20to%20the%20scene>. (29.04.2021).
  22. Keil, J., Edler, D., Schmitt, T., Dickmann, F. Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines. [Online] <https://link.springer.com/article/10.1007/s42489-020-00069-6#ref-CR20> (15.04.2021).
  23. Unity Technologies. Colliders. [Online] <https://docs.unity3d.com/Manual/CollidersOverview.html> (10.05.2021).
  24. Unity Technologies. Unity UI: Unity User Interface. [Online] <https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/index.html> (10.05.2021).