



VirtuaaliTampere

Projektin suunnittelu, organisointi ja työohjeet

Oska Piirainen

OPINNÄYTETYÖ
Kesäkuu 2025

Rakennusarkkitehdin tutkinto-ohjelma

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Rakennusarkkitehdin tutkinto-ohjelma

PIIRAINEN, OSKA
VirtuaaliTampere
Projektin suunnittelu, organisointi ja työohjeet

Opinnäytetyö 59 sivua, joista liitteitä 35 sivua
Kesäkuu 2025

Opinnäytetyö käsittelee uutta projektia työnimeltään VirtuaaliTampere, jossa mallinnetaan pääpainotteisesti Tampereen kaupungin historiallisia rakennuksia. Opinnäytetyön tavoitteena oli suunnitella projektin kulku, organisointi ja tarkat tekniset työohjeet, joiden pohjalta projekti pystytään toteuttamaan ilman projektin suunnittelijan läsnäoloa Tampereen ammattikorkeakoululla järjestettävässä harjoittelussa. Tutkimuksen kohteena oli työtavat ja sopivimmat ohjelmistot.

Työ toteutettiin projektiluontoisesti hyödyntäen myös olemassa olevaa tietoa. Suunnittelu ja organisointi perustuu pääosin työn laatijan kokemukseen aikaisemmissa samankaltaisissa projekteissa. Kokemusta ja tietotaitoa hyödynnettiin myös työtapojen ja prosessin vaiheiden valinnassa. Kyseinen projekti vaati aikaisempien tutkimuksien lisäksi uutta tutkimusta ja eri metodien kokeilua sulavan prosessin takaamiseksi. Työn aikana projektin vaiheet käytiin läpi alusta loppuun osana työtä ja ensimmäisessä kahden kuukauden harjoittelussa. Työtapojen valintojen perusteluina toimi pääasiassa työtä varten tehdyt kokeilut.

Työn tuloksena toteutui kattavat ohjeet VirtuaaliTampereen toteuttamiseksi. Projektin prosessiin löydettiin kyseisellä hetkellä sulavat ja ajankohtaiset työtavat ja ohjelmistot. Työtä hyödynnetään tulevina vuosina VirtuaaliTampere-projektissa ja sitä voi myös soveltaa muihin samankaltaisiin projekteihin.

Asiasanat: 3D-mallinnus, kaupunkimalli, visualisointi, virtuaalitodellisuus, unreal engine

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Construction Architecture

PIIRAINEN, OSKA:

VirtualTampere

Planning, Organizing and Development of an Instruction Manual for the Project

Bachelor's thesis 59 pages, appendices 35 pages

June 2025

The purpose of this thesis was to plan and organize a project called VirtualTampere and develop a step-by-step instruction manual for the modeling process. The project aims to create a full 3D city model of the City of Tampere, centering mainly on the historical buildings of Tampere. The focus of the study was to find methods and softwares to make VirtualTampere as a practical training opportunity possible for the students at Tampere University of Applied Sciences.

The study was carried out as a project. Decisions concerning the organization, methods and softwares are based on experiences in similar projects, experimenting during the process and using the available data on the subject.

In conclusion, the appropriate methods and softwares were found and proved successful. The instruction manual was compiled, and it will be used as a starting point for the project VirtualTampere. The thesis, most importantly the instructions, can also be highly useful for students, teachers, employees and employers in other similar work in the same field.

Key words: 3D modeling, city model, architectural visualization, virtual reality, unreal engine

SISÄLLYS

1	JOHDANTO	6
2	Kaupunkimalli ja CityGML	7
3	Kolmiulotteinen mallinnus	8
	3.1 Kolmiulotteisen mallin rakenne.....	8
	3.2 Mallinnusprosessi.....	8
	3.2.1 Tiedonkeräys ja mallintaminen	9
	3.2.2 Teksturointi	9
	3.2.3 Optimointi	10
	3.2.4 Renderöinti	10
4	Ohjelmat	11
	4.1 Archicad	11
	4.2 Unreal Engine 5	11
	4.3 Cetopo	13
	4.4 Ohjelmien välinen tiedonsiirto ja Datasmith	13
5	Prosessi	14
	5.1 Maastomalli	14
	5.2 Rakennukset	15
	5.2.1 Rakennusten mallinnus	15
	5.2.2 Vienti	15
	5.2.3 Tuonti	15
	5.2.4 Rakennusten teksturointi	16
	5.3 Rakennusten ja kokonaisen mallin optimointi	16
	5.3.1 LOD	16
	5.3.2 Nanite	17
	5.4 Yhdistäminen	18
6	Ohjeet	19
7	Julkaisu	20
8	Pohdinta.....	21
	LÄHTEET	23
	LIITTEET	24
	LIITE 1. VirtuaaliTampere tekniset työohjeet	24

ERITYISSANASTO tai LYHENTEET JA TERMIT (valitse jompikumpi)

ArchiCad	Kolmiulotteinen tietomallinnusohjelmisto
Unreal Engine	Pelimoottori
Cetopo	Karttapalvelu, josta voi ladata maastomalleja
Datasmith	Plugin, joka muuntaa projektitiedostoja sopivaan tiedostomuotoon ohjelmien välillä.
Template	Esiasetukset/mallipohja
Plugin	Laajennus/lisäosa
Maastomalli	3D-malli, jossa on maanpinnan muodot ja muut maastoon kuuluvat elementit
LOD	Level Of Detail, geometrian tarkkuustaso/kaupunkimallin tarkkuustaso luokitus
Nanite	Unreal Enginen uusi optimointitekнологia
Objekti	Mikä tahansa mallintamiseen käytetty yksittäinen kappale
Renderöiminen	3D-mallin muuntaminen realistiseksi 2d-kuvaksi
Tekstuuri map	Kuvatiedosto, joka luo materiaalille erilaisia ominaisuuksia
Vertex	Kulmapiste
Low-poly	Pieni määrä monikulmioita, epätarkka
High-poly	Suuri määrä monikulmioita, tarkka

1 JOHDANTO

Työnimeltään VirtuaaliTampere on projekti, jossa mallinnetaan Tampereen rakennuksia keskittyen vanhoihin rakennuksiin, jotka ovat ajansaatossa purettu tai tuhoutuneet esimerkiksi sodan vuoksi. Projektin tavoitteena on säilyttää Tampereen arkkitehtonista kulttuuriperimää. Projekti on ollut Tampereen Ammattikorkeakoulun henkilöstön ja muiden tahojen, kuten museokeskus Vapriikin ja Tampereen kaupungin kiinnostuksenkohteena jo vuosia.

Kokemuksia VirtuaaliViipuri-projektista käytetään hyväksi tulevan projektin toiminnan suunnittelussa. VirtuaaliViipurissa mallinnettiin vanhaa Viipuria ajalta, jolloin se kuului Suomeen. Rakennukset mallinnettiin tarkasti lähdemateriaalien pohjalta ja niistä tuotettiin kuvia, jotka ladattiin VirtuaaliViipurin nettisivuille.

VirtuaaliTampereessa on pitkälti sama periaate, mutta käytetyt tekniikat ja ohjelmistot on päivitetty ajantasaisiksi. Suurin VirtuaaliTampereen edistysaskel on se, että kaupunkimallista ei tuoteta kuvia, vaan malli julkaistaan virtuaalitodellisuutena, jossa voi kävellä ja katsella ympäriinsä reaaliaikaisesti.

Projektissa käytetään lähteenä useilta eri tahoilta saatuja piirustuksia ja kuvia. Vapriikin näyttely Kadonneet Kaunottaret on yksi merkittävistä lähteistä ja on ollut inspiraationa projektin luomisessa.

VirtuaaliTampere toteutetaan kesäisin Tampereen Ammattikorkeakoululla, ja se tarjoaa harjoittelupaikkoja koulun rakennusarkkitehtiopiskelijoille. Projektissa harjoittelijalla kehittyä jo koulussa opitut taidot ja sen lisäksi harjoittelija oppii uusia taitoja erikoistuen 3D-mallintamiseen ja visualisointiin.

Tämän opinnäytetyön tavoitteena on mahdollistaa projektin toteutus ja laatia tarkat ohjeet projektin rakenteelle ja tekniselle toteutukselle. Tärkeää on myös kuvata työryhmän organisointi ja roolit sekä päättää käytettävät ohjelmistot. Ohjeiden tulee ohjeistaa projektiryhmän ohjaajan toimintaa ja suorittavan portaan työntekijän toimintaa. Ohjeet julkaistaan tekstitiedostona, jossa voidaan käyttää linkkejä muun muassa ulkopuolisten tahojen ohjevideoihin sekä kuvia. Käytettävien ohjelmistojen koulutus ja ohjeistus on rajattu opinnäytetyön ulkopuolelle.

2 Kaupunkimalli ja CityGML

Kaupunkimalli on digitaalinen malli, jonka tavoitteena on kuvata kaupunki tai muu rakennettu ympäristö kolmessa ulottuvuudessa. Malli voi koostua yksittäisistä rakennuksista tai kokonaisesta kaupungista mukaan lukien infrastruktuurin. Mallintamisessa hyödynnetään esimerkiksi tietokoneavusteista mallintamista (Computer-Aided Design CAD). (Atlas 2025).

Kaupunkimalleja hyödynnetään useissa eri tarkoituksissa, kuten kaupunkisuunnittelussa, katastrofien ehkäisyssä, arkkitehtisuunnittelussa, ympäristötieteissä, virtuaalitodellisuuksissa ja pelaamisessa. (Atlas 2025)

CityGML on avoin tietomalli, joka perustuu XML-formaattiin. CityGML on hyväksytty tietomallistandardi ja sen tavoitteena on kehittää kaupunkimalleihin liittyvää luokittelua sekä samalla kaupunkimallien ylläpidon tehokkuutta. CityGML kuvaa mallinnukset viidellä eri tarkkuustasolla. Luokittelulla voidaan määritellä kaupunkimallin tarkkuustaso alustasta riippumatta. LOD0 on CityGML tarkkuustasoista yksinkertaisin ja sisältää korkeustietoa vain maastosta. LOD1 kuvaa maaston ja rakennukset laatikkomaisina massoina. Rakennusten kattomuotoa ei ole esitetty mallissa. LOD2 tuo lisää yksityiskohtia ja muotokieltä rakennuksiin sekä esittää karttaa jakavien elementtien, kuten teiden ja tonttien pintamateriaaleja. LOD3 on tarkka esitys rakennusten arkkitehtuurista. Tarkimmassa LOD4-tarkkuustasossa rakennuksissa esitetään myös sisätilat kalusteineen. (Varala 2022, 10)

VirtuaaliTampere mallinnetaan tarkkuustasolla LOD3. Tämä tarkkuustaso ei kuitenkaan määrittele jokaisen yksityiskohdan vaadittua laatua, johon tämä projekti pyrkii. Työntekijöiden ja työnjohtajan on käytettävä omaa silmää ja luotava ei pelkästään tarkka, vaan myös visuaalisesti näyttävä ja laadukas tuotos, joka on verrattavissa samaa teknologiaa käyttäviin peleihin. Vaatimuksista on kerrottu lisää työohjeissa (LIITE 1, 16–32)

3 Kolmiulotteinen mallinnus

Kolmiulotteinen mallinnus tapahtuu CAD-ohjelmistoilla. Seuraavissa kappaleissa käydään läpi, mistä kolmiulotteinen malli koostuu sekä mallintamisen vaiheet.

3.1 Kolmiulotteisen mallin rakenne

Kolmiulotteinen malli koostuu kulmapisteistä (vertices), reunoista (edges), ja pintalohkoista (faces). Pintalohkoja kutsutaan yleensä monikulmioiksi (polygons). Huippupisteet ovat pisteitä mallissa, jotka merkkavat kolmioiden kärjet. Reunat yhdistävät pisteet ja muodostavat monikulmiot. (Danthree Studio n.d.). Monikulmiot jaetaan monesti kolmioiksi (triangles).

Kolmioiden määrä vaikuttaa suoraan mallin tarkkuuteen. Mitä enemmän kolmioita, sitä tarkempi malli. Samalla kolmioiden määrä vaikuttaa mallin kokoon ja siihen, kuinka raskasta mallia on käsitellä. Kolmioiden määrään liittyen puhutaan low-poly ja high-poly malleista. Low-poly mallissa on vähemmän polygoneja, eli kolmioita, ja on kevyempi pyörittää, mutta myös epätarkempi. High-poly mallissa on isompi määrä kolmioita, eli on tarkempi, mutta vaatii koneelta enemmän tehoa. Kolmioiden määrän tasapainottaminen on yksi mallintamisen tärkeimmistä asioista ottaa huomioon, kun pyritään saamaan reaaliaikaisesti pyörivä lopputulos.

Fyysisten muotojen lisäksi mallissa on usein tekstuuritietoja, jotka määrittävät pintojen materiaalien ulkonäön. Tekstuuritiedoston koko vaikuttaa myös koko mallin kokoon ja sen käsittelyyn. Tekstuurit koostuvat usein monesta useasta eri "mapista" eli kuvatiedostosta, joka määrittää materiaalille eri ominaisuuksia. Tekstuureista lisää ohjeissa (LIITE 1, 22–25).

3.2 Mallinnusprosessi

Mallintamisen kolme tärkeintä vaihetta on tiedon keräys, mallin rakentaminen ja mallin renderöinti. Tiedon keräämiseen hyödynnetään useita eri tekniikoita, kuten lasermittaus ja valokuvaaminen. Mallin rakentaminen on prosessi, jossa kerätty tieto muunnetaan 3D-malliksi. Mallin renderöinti on viimeinen prosessi, jossa

luodaan lopullinen visuaalinen esitys mallista. (Atlas 2025). Peleissä ja virtuaalitodellisuuksissa myös optimointi on tärkeä vaihe.

3.2.1 Tiedonkeräys ja mallintaminen

Mallintaminen alkaa yleensä lähdemateriaalien, eli piirustusten, kuvien tai pistepilvien tutkimisesta, paitsi uutta rakennusta suunnitellessa. Piirustukset tarjoavat mitat mallintamista varten, tai vaihtoehtoisesti monessa CAD-ohjelmassa piirustuksen voi asettaa suoraan mallinnuksen pohjalle ja mallintaa piirustuksen päälle. Valokuvia voi hyödyntää suurpiirteiseen malliin tai yksityiskohtiin, joista ei välttämättä aina ole piirustuksia. Pistepilvet voi muuntaa suoraan 3D-malliksi.

Mallintaminen tarkoittaa fyysisen 3D-mallin rakentamista. Siihen kuuluu esimerkiksi seinät, katto, ikkunat ja muut rakennuksen muodot. Arkkitehtisuunnitteluun käytettävissä ohjelmistoissa on työkalut jokaiselle rakennusosalle, mikä helpottaa rakennuksen perusosien tekemistä. Monissa peliteollisuuden ohjelmistoissa voi olla vain tavanomaisia geometrisia muotoja, kuten särmiöitä, kartioita, lieriöitä ja palloja.

3.2.2 Teksturointi

Hyvälaatuiset tekstuurit ovat yksi tärkeimmistä tekijöistä laadukkaan mallin saavuttamiseksi. Ilman tekstuureja malli koostuisi vain tasavärisistä pinnoista. Tekstuurikuvat ovat pieniä otteita kokonaisesta pinnasta. Kuvaa toistetaan vaaka- ja pystysuunnassa pinnan alueen laajuisesti. Toistaminen tuo esiin tekstuurien suurimman vaikeuden; pinta näyttää helposti ruudukkoiselta, jos tekstuurikuvan reunat eivät sulaudu toisiinsa saumattomasti. Jos kuvassa taas on liian vähän eloa, se näyttää monotoniselta. Sen takia pinnoille asetetaan monesti tekstuurien päälle yksittäisiä epätäydellisyyksiä, kuten kuluneisuutta tai likaa.

Tekstuuria voi parantaa myös lisäämällä värikuvan lisäksi esimerkiksi pinnanmuotoja, kiiltävyyttä tai läpinäkyvyyttä muokkaavia ”mappeja”. Kaikkia tekstuuri-mappeja hyödyntämällä voi saada hyvin todenmukaisen ja näyttävän materiaalin. (LIITE 1, 22–25.)

3.2.3 Optimointi

Optimointi on yleisesti käytetty termi peliteollisuudessa, joka tarkoittaa pelin pyörittämiseen tarvittavan prosessointitehon minimointia niin, että visuaalinen ilme pysyy silminnähden samana. Optimointi on kriittinen vaihe, jotta saadaan mahdollisimman hyvälaatuinen malli, joka ei ole liian raskas. Huonosti optimoitu malli voi hidastaa peliä merkittävästi, tai jopa tehdä siitä käyttökelvottoman liian huonolla tietokoneella. Toinen vaihtoehto on, että malli on todella yksinkertainen ja epätarkka, jolloin optimoinnille ei ole niin suurta tarvetta.

Optimointi on yksinkertaisuudessaan kolmioiden määrän pienentämistä mahdollisimman paljon kuitenkin siten, että mallin laatu ei heikkene. Optimointia on myös tekstuurien kuvatiedostojen koon pienentäminen ilman näkyvää eroa. Edellä mainittuihin tavoitteisiin yleisimmin käytetty metodi on pienentää tarkkuutta mitä kauempaa kohdetta katsotaan.

3.2.4 Renderöinti

Renderöinti tarkoittaa 3D-mallin muuntamista realistiseksi 2d-kuvaksi tai videoksi. 3D-malliin lisätään tekstuurit, valaistus, varjot ja muita visuaalisia efektejä mallin elävöittämiseksi. Mallintaminen keskittyy mallin muotoon, kun taas renderöinti saa mallin näyttämään realistiselta. (Dimension 5 2020-2025)

Renderöinnin yleisin tapa on offline-renderöinti, joka keskittyy laadukkaaseen lopputulokseen. Se hyödyntää monimutkaisia algoritmeja luodakseen fotorealistisia valokuvia ja videoita, joita nähdään elokuvissa, animaatioissa ja arkkitehtonisissa visualisoinneissa. Tämä prosessi vie kuitenkin aikaa. (Dimension 5 2020-2025)

Toinen vaihtoehto on reaaliaikainen renderöinti, joka on suunniteltu nopeaksi ja mahdollistaa malliin tehtyjen muutosten näkemisen välittömästi (Dimension 5 2020-2025). Tarkemmin sanottuna, mallissa nähdään kaikki tavanomaisen renderöinnin tuomat visuaaliset efektit reaaliaikaisesti. Reaaliaikainen renderöinti on erityisen hyödyllistä peleissä, virtuaalitodellisuuksissa ja interaktiivisissa esityksissä (Dimension 5 2020-2025).

4 Ohjelmat

Seuraavissa kappaleissa on kuvattu käytettävät ohjelmistot, niiden käyttötarkoitukset ja valintaperusteet ohjelmistoille. Opinnäytetyössä tutkittiin ja testattiin vaihtoehtoisia ohjelmistoja ja arvioitiin niiden sopivuutta, tehokkuutta ja käytettävyyttä VirtuaaliTampere-projektin kontekstissa, sekä niiden keskinäistä yhteensopivuutta. Ohjelmistojen, joita ei opeteta Tampereen ammattikorkeakoulun rakennusarkkitehtiopiskelijoille, käyttöönottoa harkittiin niiden opetteluun tarvittavan ajan mukaan.

4.1 Archicad

Archicad on tietokoneavusteinen mallinnusohjelma (CAD), joka on Tampereen ammattikorkeakoulun rakennusarkkitehtikoulutuksessa pääasiallisesti käytettävä mallinnusohjelma. Opiskelijoilla on paljon kokemusta ohjelman käyttämisestä, minkä vuoksi se on hyvä valinta projektin mallinnusohjelmaksi. Projektista saatu kokemus Archicadin käytössä edesauttaa vastavuoroisesti opiskelijan koulutusta ja on hyödyksi hänelle tulevaisuudessa.

Archicadia käytetään teksturoimattomaan 3D-mallin tuottamiseen. Archicadissa on erityisesti rakennusten mallintamista varten suunnitellut työkalut ja on siksi nopea ja helppo. Se myös takaa laadukkaan tuloksen projektin tarpeisiin.

Edellä mainittujen etujen vuoksi Archicad valittiin vaihtoehtoisten 3D-mallinnusohjelmien joukosta. Myöhemmässä vaiheessa käytettävän ohjelman Unreal Engine 5 hyödyntäminen mallinnusvaiheessa olisi edullista käytettävien ohjelmien minimoimiseksi. Sitä ei kuitenkaan opeteta opiskelijoille, se ei sinänsä edistä opiskelijan keskeistä koulutusta eikä siksi tullut valituksi kyseiseen vaiheeseen.

4.2 Unreal Engine 5

Pelimoottori (Game Engine) on ohjelmistokehitysympäristö, joka tarjoaa laajan valikoiman työkaluja ja ominaisuuksia videopelien ja muiden interaktiivisten sovellusten kehittämiseen. Pelimoottorit yksinkertaistavat ja nopeuttavat pelinkehitysprosessia tarjoamalla valmiita ratkaisuja moniin pelinkehityksen osa-alueisiin, kuten grafiikkaan,

ääniin, fysiikkaan, tekoälyyn, animaatioon ja verkkopeliominaisuuksiin. (P 2025)

Unreal Engine 5 on uusimman teknologian omaava pelimoottori, joka on opinäytetyön kirjoittamisen hetkellä yksi parhaimmista työkaluista pelien luomiseen, mallintamiseen ja visualisointiin. Unreal Engineä käytetään VirtuaaliTampere-projektissa teksturoimiseen ja optimointiin.

Merkittävin uudistus pelimoottoreihin Unreal Enginen versiossa 5 on Nanite-teknologia, joka mahdollistaa monimutkaisten ja raskaidenkin mallien optimoimisen huomattavasti kevyemmäksi. Toinen VirtuaaliTampere-projektin kannalta ratkaiseva ominaisuus Unreal Enginessä on sen mahdollisuudet luoda laadukkaita ja realistisia tekstuureja. Kolmas ja tärkein ominaisuus on mahdollisuus luoda mallista reaaliaikainen virtuaalitodellisuus.

Edellä mainitut ominaisuudet tekevät Unreal Enginestä parhaan vaihtoehdon teksturoimiseen, optimointiin ja lopputuotoksen luomiseen. Archicadissa pystyy luomaan tekstuurit, mutta Unreal Enginen työkalut ovat merkittävästi monipuolisemmat. Twinmotion on arkkitehtivisualisointiin tarkoitettu yksinkertaisempi versio Unreal Enginestä, joka on monelle rakennusarkkitehtipiskelijälle entuudestaan tuttu. Siitä huolimatta opinäytetyön tutkimuksen ja testien perusteella Unreal Enginen opettelu koettiin kannattavaksi. Nanite-teknologiaa ei myöskään ole missään muussa ohjelmistossa, ja virtuaalitodellisuuden luominen onnistuu parhaiten Unreal Enginessä.

Reaaliaikainen virtuaalitodellisuus luo myös rajoitteita. Perinteisesti arkkitehtivisualisoinnit tehdään offline renderöimällä, jolloin kolmioiden määrästä, mallin koosta ja optimoinnista ei tarvitse huolehtia. Tällöin yksityiskohtia voi olla kuinka paljon tahansa ja se vaikuttaa ainoastaan kuvien renderöimiseen kuluvaan aikaan. Reaaliaikaisessa renderöinnissä täytyy kuitenkin pitää huoli, ettei kolmioiden määrä nouse liian korkeaksi. Nanite-optimointitekniikka on huomattava apu kyseiseen haasteeseen.

4.3 Cetopo

Cetopo on lisäosa, jolla voi tehdä tarkkoja ympäristömalleja käyttäen oikeaa dataa tutuissa ohjelmistoissa (Cetopo n.d.). Käytännössä se tarkoittaa sitä, että VirtuaaliTampere projektiin voi luoda maaston hyödyntäen mitattuja korkeuskäyriä. Cetopo saa tiedot maanmittauslaitoksista. Kyseinen vaihe onnistuu Archicadin sisällä.

4.4 Ohjelmien välinen tiedonsiirto ja Datasmith

”Tiedostomuoto ilmaisee tietokoneiden tallennusmedialle talletetun tiedoston rakenteen. Tunnistamisen helpottamiseksi käytetään tiedostopäätteitä. Tunnistat ne tiedostonimen lopussa pisteellä eroteltuna (esim. testi.jpg)”. (Peda.net n.d.)

Jokaisella ohjelmistolla on yleensä omat tiedostomuotonsa, mutta ne käyttävät osittain myös samoja tiedostomuotoja. VirtuaaliTampere-projektissa käytettävät ohjelmistot luovat tiedostomuotoja, jotka eivät suoraan sovi ohjelmasta toiseen. Sitä varten käytetään Datasmith-lisäosaa.

Datasmith on plugin eli lisäosa, jolla tiedostot muunnetaan sopivaan muotoon ja siirretään Archicadista Unreal Engineen. Ohjelmasta on useita versioita, jotka soveltuvat eri ohjelmille. Datasmith on helppokäyttöinen ja luotettava työkalu tiedostojen muuntamiseen ja siirtämiseen.

5 Prosessi

Opinnäytetyötä tehdessä kokeiltiin prosessin jokainen vaihe yksitellen, tutkittiin eri vaihtoehtoja ja löydettiin parhaat toimintamallit projektin toteuttamiselle. Tässä luvussa käydään läpi prosessin vaiheet, niissä esiintyneet ongelmat ja perustelut hyväksi todetuille ratkaisulle. Tarkemmat tekniset ohjeet vaiheisiin löytyvät VirtuaaliTampereen työohjeista (LIITE 1, 16–32).

Prosessi mallintamisesta valmiiseen kaupunkimalliin on monivaiheinen. Ensimmäinen vaihe on rakennusten mallintaminen ArchiCadissa, jonka jälkeen ne teksturoidaan Unreal Engineessä. Samaan aikaan luodaan maastomalli hyödyntäen Cetopoa, Archicadia ja Unreal Enginea. Lopuksi korttelit yhdistetään yhdeksi isoksi malliksi Unreal Engineessä.

5.1 Maastomalli

Maasto on mahdollista mallintaa kolmella eri tavalla. Sen voi mallintaa manuaalisesti hyödyntämällä korkeuskäyriä ja ArchiCadin pintatyökalua. Kyseinen tapa on työläs ja vaatii aikaa. Toinen vaihtoehto on ladata Cetopo-sivustolta valmis maastomalli. Kolmas tapa on hyödyntää jo olemassa olevia Tampereen kaupunkimalleja, jos mahdollista. Opinnäytetyössä ohjeistetaan toinen vaihtoehto.

Cetoposta ladataan sopiva alue ja käytetään siihen sopivaa templatea, joka jaetaan yhteiselle palvelimelle. Sen jälkeen maastomalli tuodaan ArchiCadiin Cetopo-pluginilla, josta se vietään Datasmithilla Unreal Enginelle sopivaan tiedostomuotoon ".udatasmith". Tämä prosessi tehdään, koska Datasmith optimoi tiedoston tehokkaasti ja pienentää tiedoston kokoa huomattavasti. Cetoposta suoraan Unreal Enginelle sopivassa tiedostomuodossa ladattuna tiedoston koko on merkittävästi suurempi.

Tämän opinnäytetyön tutkimuksen puitteissa tiet päädyttiin mallintamaan itse Archicadin laattatyökalulla, koska Cetoposta ladatut tiet eivät ole riittävän hyvälaatuiset, eivätkä ole mallinnettavan ajankohdan mukaisia. Cetoposta ladataan tiet 2D-merkintöinä, joiden mukaan tiet mallinnetaan kolmiulotteiseksi Archicadissa.

5.2 Rakennukset

Jokainen harjoittelija luo malleja joko kortteleissa tai yksittäisinä rakennuksina riippuen rakennusten mallintamiseen kuluva ajasta. Rakennusten projektitiedostot pidetään mallintamisvaiheessa mallintajan omana tiedostona, jotta he voivat muokata tiedostoa erillään toisistaan, ja että tiedostot voidaan pitää mahdollisimman kevyinä.

5.2.1 Rakennusten mallinnus

Rakennukset mallinnetaan Archicadissa, sillä siihen ohjelmaan opiskelijoilla on jo riittävät taidot. Mallinnukseen käytetään apuna sopivia lähdemateriaaleja. Rakennuksen projektitiedostoon on hyvä ottaa viitteeksi maastoa korttelin alueelta, jotta rakennukset myötäilevät pinnanmuotoja. Maasto on tässä vaiheessa vain avuksi, ja se jätetään pois seuraavassa vaiheessa. Rakennukset mallinnetaan tarkasti ulkopuolelta, mutta sisäpuolet jätetään mallintamatta. Pinnat teksturoidaan myöhemmässä vaiheessa, koska Unreal Engineessä on siihen paremmat työkalut kuin ArchiCadissa.

5.2.2 Vienti

Ensin kortteli muunnetaan objektiksi Archicadissa ja sitten kortteli tallennetaan Datasmith-työkalulla tiedostoksi (LIITE 1, 20–21). Objekti tehdään siksi, koska testiprojektissa suoraan 3D-mallista viety datasmith-tiedosto (13,1 Mt) on miltei 500-kertaa isompi, kuin objektista viety datasmith-tiedosto (27,2 kt). Objektiksi muunnettaessa ei menetetä mitään seuraaville vaiheille tärkeitä tietoja tai ominaisuuksia. Tässä tapauksessa ei hyödynnetä Direct Link -ominaisuutta, koska objektia ei voi joka tapauksessa muokata sen tallentamisen jälkeen ja viety tiedosto on varsinkin usean ihmisen projekteissa vakaampi.

5.2.3 Tuonti

Unreal Engineessä luodaan uusi projekti Architecture-osiosta Blank-templatella. Datasmith tiedoston tuominen Unreal Engineen on yksinkertainen ja ongelmaton vaihe. Jotta projekti pysyy järjestelmällisenä, Unreal Engineessä on itse pidettävä

huolta tiedostojen hallinnasta (LIITE 1, 8). Tässä vaiheessa projektiin kannattaa luoda esimerkiksi ”geometriat” -kansio, jonne laitetaan kaikki projektin geometriset mallit.

5.2.4 Rakennusten teksturointi

Tekstuurit materiaaleille luodaan Unreal Engineissa. Materiaaleille, joille löytyy valmiita tekstuureja, voi käyttää niitä. Valmiita tekstuureita ei kuitenkaan aina ole juuri sopivan näköisiä. Jotta projektin tarkkuustaso pysyy korkealla, tekstuurit tehdään itse valmiin tekstuurin puuttuessa. Tekstuureja voi tehdä itse netistä löytyvillä kuvilla ja itse otetuilla kuvilla sekä hyödyntämällä kuvanmuokkausohjelmia. Kaikki löydetyt ja tehdyt tekstuurit tallennetaan yhteiselle palvelimelle, jotta voidaan hyödyntää mahdollisimman paljon samoja tekstuureja yhtenäisen ilmeen ja ajan säästämisen vuoksi.

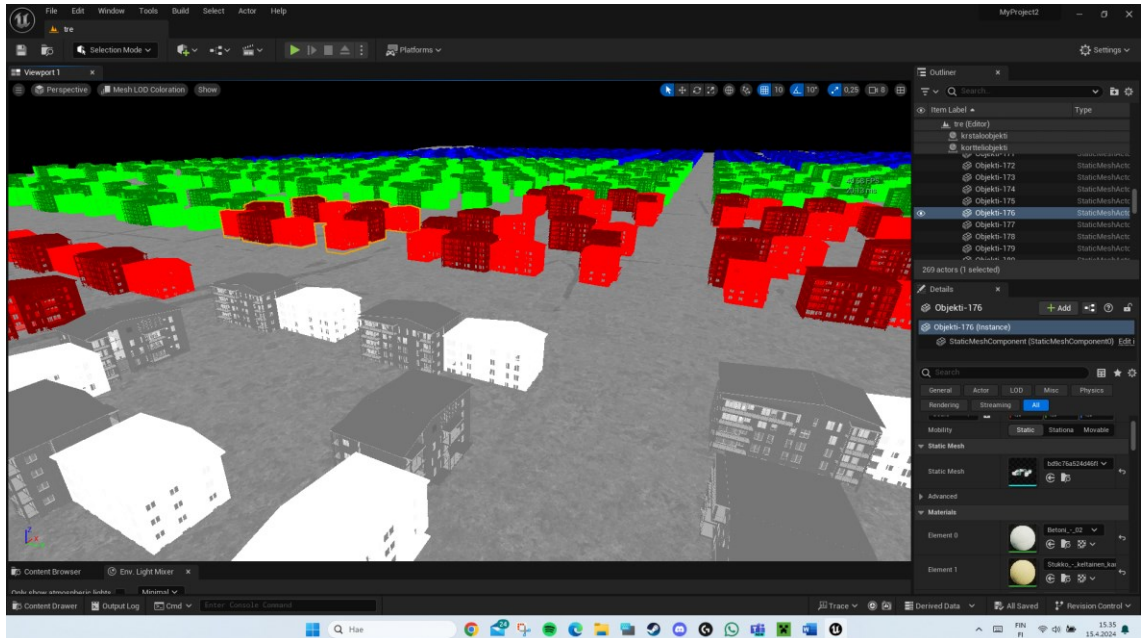
5.3 Rakennusten ja kokonaisen mallin optimointi

Optimointi on tärkeässä osassa VirtuaaliTampereen kaupunkimallia, sillä siinä pyritään aiempiin Tampereen kaupunkimalleihin verrattuna parempaan visuaaliseen ilmeeseen ja realismiin. VirtuaaliTampereen kaupunkimalli toteutetaan yksityiskohtaisesti ja visuaalisesti näyttävällä tavalla. Mallin täytyy pyöriä sulavasti reaaliaikaisena 3D-ympäristönä, jossa voi kävellä ja katsella mallia vapaasti. Edellä mainitut asiat vievät yhdessä paljon prosessointitehoja, minkä takia projektin tulee olla hyvin optimoitu.

5.3.1 LOD

LOD eli Level Of Detail tarkoittaa sitä, kuinka tarkasti geometriat näytetään. Tarkkuustasoja on yleensä muutamia tarkimmasta, eli alkuperäisestä tarkkuudesta, epätarkimpaan ja yksinkertaistetuimpaan. Katsojaa lähellä olevat geometriat näytetään tarkimmin kaikki yksityiskohdat mukaan lukien ja mitä kauempana geometriat näkyvät, sitä vähemmän yksityiskohtia niistä näytetään. Tämä säästää huomattavasti prosessointitehoa, koska jokaista kauimmaistakin yksityiskohtaa ei tarvitse luoda näytölle.

LOD-järjestelmä on Naniteen verrattuna vanhempi, mikä tuo mukanaan hyviä ja huonoja puolia. Naniten teknologia on edistyneempää ja kykenee suorituskyvyltään parempaan. Teknologia on kuitenkin vielä niin uutta, että se ei ole sataprosenttisen vakaa. LOD-järjestelmä toimi opinnäytetyön testiprojektissa luotettavasti, mutta tarjosi Naniteen verrattuna huonommat optimointitulokset.

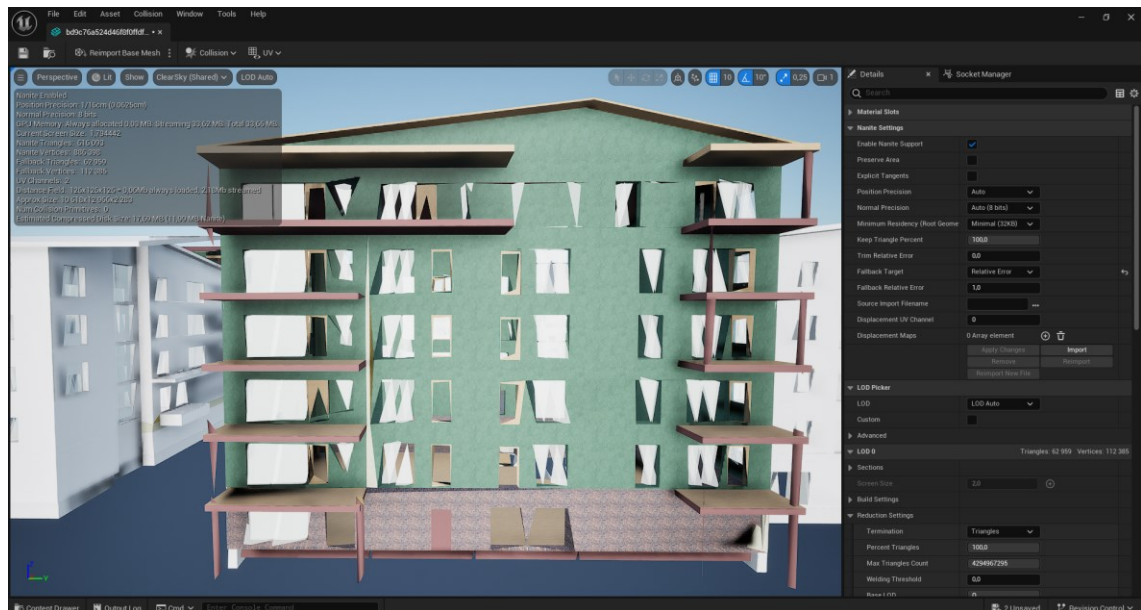


KUVA 1. LOD-tasot eri väreillä

5.3.2 Nanite

Nanite on Unreal Engine 5 päivityksen mukana tullut uusi dynaaminen optimointitekniikka, joka pienentää geometrioiden monimuotoisuutta automaattisesti. Tarkemmin Nanitea on kuvattu Unreal Enginen sivuilla (Unreal Engine 2025)

Nanite pystyy teoriassa parempaan lopputulokseen optimoinnin suhteen verrattuna LOD-järjestelmään. Opinnäytetyön testiprojektissa Naniten käytössä ilmeni kuitenkin ongelmia (KUVA 2). Nanite ei välttämättä pysty prosessoimaan ArchiCADista tuotuja malleja. Ongelma saattoi johtua myös testiprojektissa käytetystä tietokoneesta.



KUVA 2. Nanite tuhoaa mallin geometriat

Testiprojektissa Nanite saatiin lopulta toimimaan. Nanite vaatii lisää käytännön tutkimusta VirtuaaliTampere-projektin puitteissa projektissa käytettävillä tietokoneilla ja malleilla. Naniten käyttö täytyy lopulta päättää tapauskohtaisesti riippuen sen toimivuudesta kyseiseen malliin.

5.4 Yhdistäminen

Kun yksittäinen rakennus tai kortteli on valmis, se tuodaan maastomalliin. Yhdistämällä maastomalli ja yksittäiset rakennukset vasta loppuvaiheessa säädetään kuormaa aikaisemmilta vaiheilta. Kokonainen projekti, jossa on useita rakennuksia, kasvillisuutta ja yksityiskohtia on nopeasti hyvinkin raskas, mikä vaikeuttaa projektin kanssa työskentelyä.

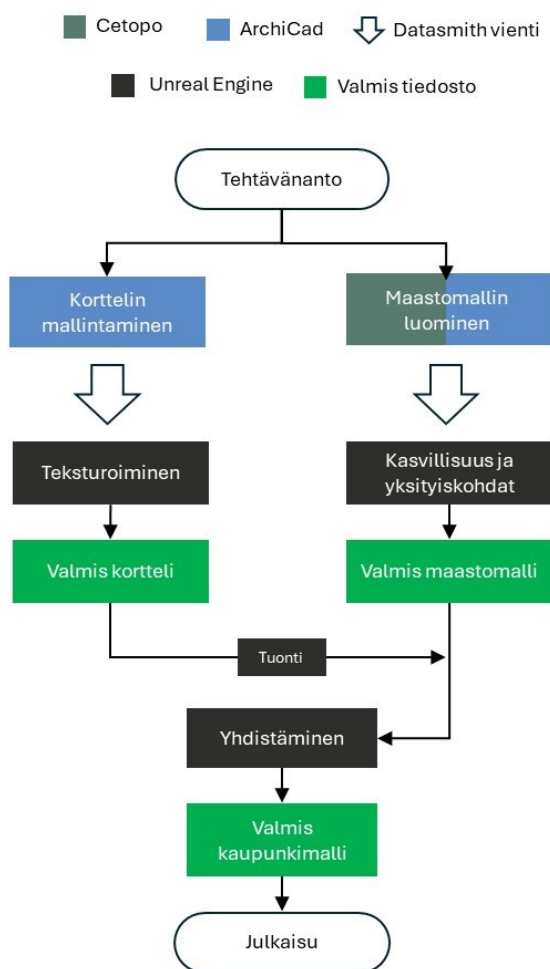
Yhdistämisvaihe on yksinkertainen ja pitäisi teoriassa onnistua helposti. VirtuaaliTampere-projektin edetessä tulee kuitenkin todennäköisesti pulmakohtia, joihin täytyy kehittää ratkaisut tapauskohtaisesti.

6 Ohjeet

Ohjeet on luotu selkeäksi ja helposti ymmärrettäväksi aloittavalle harjoittelijalle. Projektista annetaan ensin yleiskuva harjoittelijalle. Sen jälkeen on kuvattu tarvittavat ohjelmistot ja niiden vaatimukset tietokoneelle, eri opintovuosien harjoittelijoiden työroolit, lähdemateriaalien käyttö sekä jokainen prosessin vaihe yksityiskohtaisesti. Lopussa on lisäksi raportointiosio, jonne raportoidaan projektin etenemisestä.

Opinnäytetyössä luotiin selventävät vuokaaviot projektin etenemisestä. Vuokaaviot antavat hyvän käsityksen eri vaiheista isommassa mittakaavassa. Ne luotiin PowerPoint-ohjelmalla. Ohjeista löytyy neljä kaaviota (LIITE 1, 4, 17, 19, 31), joista yksi esimerkki on alla (KUVA 3).

Prosessin työvaiheet ja ohjelmistot



KUVA 3. Vuokaavio koko prosessista

7 Julkaisu

Päätavoite on julkaista tulevaisuudessa VirtuaaliTampereen kaupunkimalli reaaliaikaisena virtuaalitodellisuutena. Virtuaalitodellisuus julkaistaan pelinä, jonka pyörittämiseen tarvitaan todennäköisesti tehokas tietokone. Vaihtoehtoja, miten suuri yleisö pääsee käyttämään peliä, täytyy tutkia enemmän.

Vapriikin kanssa on käyty neuvotteluja heidän tiloihinsa luotavasta vr-ympäristöstä, jossa yleisö saisi tutkia kaupunkimallia vr-laseilla. Vapriikille olisi mahdollista järjestää riittävän tehokas tietokone, minkä seurauksena käyttäjän ei tarvitse huolehtia tietokoneesta ja kaupunkimalli olisi saavutettava kaikille.

Jotta VirtuaaliTampere saisi mahdollisimman paljon näkyvyyttä ja se olisi saavutettava mahdollisimman monelle, kaupunkimallin esittelyä varten luotaisiin nettisivut. Kaupunkimallia kokonaisuudessaan ei todennäköisesti ole mahdollista saada nettiselaimessa toimivaksi versioksi. Vaihtoehtoisia esitystapoja on renderöidyt valokuvat ja videot tai Googlen karttojen tapainen katunäkymä. Katunäkymä koostuisi useista 360 asteen panoraamakuvista, joiden välillä on mahdollista liikkua.

8 Pohdinta

Opinnäytetyön tuloksena tuotettiin päivitettävä word-dokumentti, jossa projektiorganisaation rakenne on kuvattu yksityiskohtaisesti mukaan lukien harjoittelijoiden tehtäväkuvaukset (LIITE 1, 10–12). Ohjaajan toimintaan kuuluu muun muassa projektin työntekijöiden johtaminen, laaduntarkastus, mallintaminen ja erityisesti vaativammat työtehtävät sekä ohjeiden päivittäminen. Ohjaaja valitaan kokeneimpien harjoittelijoiden joukosta. Muiden työntekijöiden työtehtäviin kuuluu pääasiassa mallintaminen. Dokumentissa on useita linkkejä projektin toiminnan kannalta tärkeiden vaiheiden kuvaaviin ohjevideoihin.

Ohjeiden teknisessä ohjeistuksessa kuvataan yksityiskohtaisesti jokainen prosessin vaihe. Vaiheisiin kuuluu mallintaminen, teksturoiminen, optimointi, yhdistäminen ja julkaisu. Tampereen ammattikorkeakoulun rakennusarkkitehtiprosessille uusiin aiheisiin on tarjottu riittävästi taustatietoa niiden ymmärtämiseksi. Projektin toteuttamiseen tarvittavat ohjelmistot on valittu helppokäyttöisyyden ja tehokkuuden mukaan. Riittävien perusvalmiuksien omaava harjoittelija kykenee suorittamaan VirtuaaliTampere-projektin edellyttämät työtehtävät opinnäytetyössä tehtyjen ohjeiden perusteella. Tätä päätelmää tukee vuoden 2024 ja 2025 kesinä toteutetut onnistuneet harjoittelut.

Muun muassa tekijänoikeuksiin ja harjoitteluun liittyvät seikat määritetään harjoittelusopimuksessa. Mahdollisia kuvamateriaaleja käytettäessä täytyy huomioida tietosuojalain asetukset. Harjoittelussa noudatetaan Tampereen ammattikorkeakoulun yleisiä eettisiä ohjeita (TUNI n.d.).

Ohjeissa on kaikki tarvittava tieto projektin aloittamiseen ja riittävästi myös sen jatkamiseen. Mahdollisia ohjeisiin jääneitä virheitä tai puutteita tulee korjata harjoitteluprojektin edetessä. Teknologia kehittyy vauhdilla, mikä vaikuttaa projektin työtapoihin. Siitä syystä ohjeita tulee päivittää säännöllisesti ajantasaiseksi. Ohjeisiin sisällytettiin projektin raportointiosio, jonne voi lisätä pienellä kynnyksellä uusia ideoita ja huomioita sekä raportoida työn edistymistä.

VirtuaaliTampere-projektissa on potentiaalia laajentamiseen tulevaisuudessa. Opinnäytetyössä tehty suunnitelma keskittyy lähitulevaisuuteen ja sen

toteuttaminen tuottaa yhden historian ajankohdan Tampereen keskustan alueesta. Myöhemmin tulevaisuudessa ajankohtia voi mallintaa useita ja hyödyntää esimerkiksi liukusäädintä, jolla voi siirtyä eri vuosien tilanteiden välillä. Myös alueen rajausta voi kasvattaa huomattavasti.

LÄHTEET

Atlas. 2025. 3D City Modeling. Verkkosivu. Viitattu 27.5.2025. <https://atlas.co/glossary/3-d-city-modeling/>

Danthree Studio. n.d. What Is A 3D Model? (Definition & Examples) Verkkosivu Viitattu 1.6.2025. <https://www.danthree.studio/en/blog-cgi/what-is-a-3D-mesh-model-definition-examples>

Dimension 5. 2020-2025. What does 'Rendering Meaning' refer to in modern design?. Verkkosivu. Viitattu 2.6.2025. <https://www.d5render.com/posts/what-does-rendering-mean>

Cetopo. n.d. How It Works?. Verkkosivu. Viitattu 2.6.2025. <https://cetopo.com/how-it-works-context-models>

P, Juha. 2025. Pelimoottori. Createch Finland 10.2.2025. Viitattu 3.6.2025. <https://www.createchfinland.fi/fi/glossary/pelimoottori/>

Peda.net. n.d. Yleisimpiä tiedostomuotoja. Verkkosivu. Viitattu 3.6.2026. <https://peda.net/opetussuunnitelma/ksops/aanekosken-kaupungin-perusopetuksen-tvt-ops/sisalto/aanekosken-varhaiskasvatuksen-tvt-suunni32/sisalto/materiaalit/yleisimpia-tiedostomuotoja>

Unreal Engine. 2025. Nanite Virtual Geometry. Verkkosivu. Viitattu 13.4. 2025. https://dev.epicgames.com/documentation/en-us/unreal-engine/nanite-virtualized-geometry-in-unreal-engine?application_version=5.0

TUNI. n.d. TAMKin opiskelijan käsikirja. Verkkosivu. Viitattu 4.6.2025. <https://www.tuni.fi/fi/opiskelijan-opas/kasikirja/tamk>.

Varala, V. K. 2022. Kaupunkimalli Unreal Engine – pelimoottoriin. https://www.theseus.fi/bitstream/handle/10024/751456/Varala_Veetu.pdf?sequence=2&isAllowed=y

LIITTEET

LIITE 1. VirtuaaliTampere tekniset työohjeet

Työohjeet kokonaisuudessaan liitteenä alla.



VirtuaaliTampere työohjeistus

Ohjelmat, työtehtävät ja tekniset ohjeet 2025

SISÄLLYS

1	JOHDANTO	3
2	Ohjelmat	5
	2.1 Tietokoneen vaatimukset	5
	2.2 Archicad	8
	2.3 Unreal Engine 5	8
	2.4 Datasmith	8
	2.5 Cetopo	9
3	Työnjako	10
	3.1 1. vuoden opiskelijat.....	10
	3.2 2. vuoden opiskelijat.....	10
	3.3 3. vuoden opiskelijat.....	11
	3.3.1 Johtaminen	11
4	Materiaalit	13
	4.1 Tiedostojen hallinta	13
	4.2 Lähdemateriaalit.....	14
	4.3 Tekstuurit / objektit	14
	4.4 Muut lähteet	15
5	Tekniset ohjeet	16
	5.1 Maastomalli	17
	5.2 Kortteli.....	19
	5.2.1 Korttelin mallinnus	19
	5.2.2 Vienti	20
	5.2.3 Tuonti	21
	5.2.4 Teksturoiminen	22
	5.2.5 Yksityiskohdat.....	25
	5.3 Optimointi.....	26
	5.3.1 LOD	27
	5.3.2 Nanite	29
	5.4 Yhdistäminen	31
6	Julkaisu.....	33
7	Raportointi	34
	7.1 Työnalla/tehty.....	34
	7.2 Suunnitelmat/Muistiinpanot	34
	7.3 Hyödylliset linkit.....	35
	7.4 Yhteyshenkilöt.....	35

1 JOHDANTO

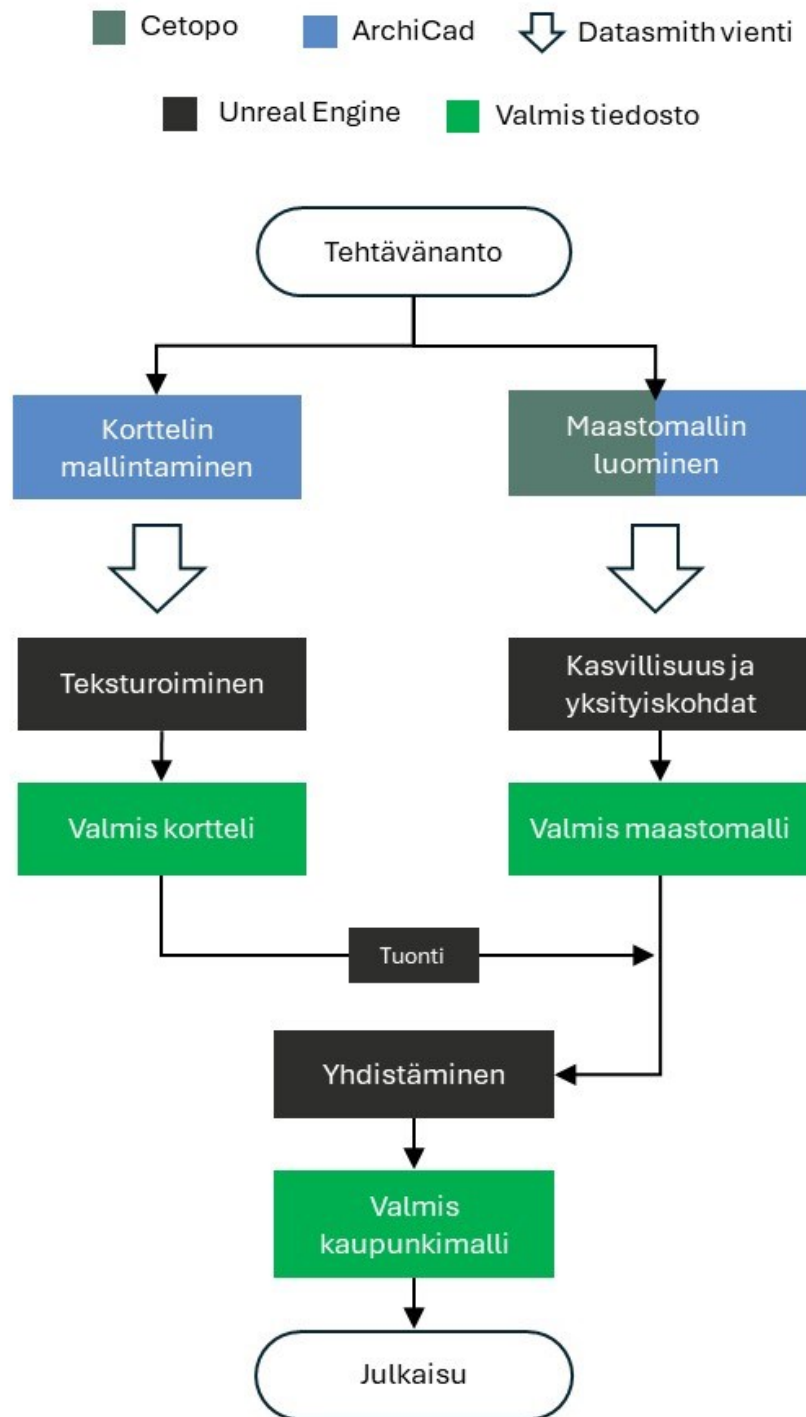
VirtuaaliTampere -projektissa mallinnetaan Tampereen rakennuksia menneiltä vuosilta mukaan lukien olemassa olevat rakennukset ja jo puretut rakennukset. Lopputuloksena on realistinen 3D-ympäristö, jossa voi liikkua ja katsella ympärilleen. VirtuaaliTampere -projekti on suunnattu TAMK:n rakennusarkkitehtiprosesseille.

Mallintaminen aloitetaan Tampereen keskustasta ja kohteiksi otetaan museokeskus Vapriikin Kadonneet kaunottaret -näyttelyn rakennuksia. Aloitetaan Hämeenkadusta. Projektin edetessä suunnitellaan tulevaisuutta tarkemmin. Pitkän tähtäimen tavoitteena on saada usean eri vuosiluvun tilanteet kaupunkimalliksi.

Tämä tiedosto mahdollistaa projektin aloittamisen ja työskentelyn tehokkaasti ja sujuvasti. Projektia tehdessä kuitenkin todennäköisesti opitaan lisää ja sitä mukaa ohjeitakin päivitetään. Jokaista projektin työntekijää kannustetaan tutkimaan uusia työtapoja ja tekniikoita sekä jakamaan tiedon vähintään projektin johtajan kanssa, joka päivittää ohjeet ja tiedottaa muitakin työntekijöitä.

Seuraavassa vuokaaviossa on yksinkertaistetusti esitetty projektin kulku kokonaisuudessaan. Ohjelmista ja jokaisesta vaiheesta kerrotaan tarkemmin seuraavissa luvuissa

Prosessin työvaiheet ja ohjelmistot



Kaavio 1

2 Ohjelmat

Tässä osiossa käydään läpi mitä ohjelmia projektissa käytetään ja mitä vaatimuksia niillä on koneille. Varmista, että voit käyttää mainittuja ohjelmia ja palveluita. Kaikki tarvittavat ohjelmistot tulisi löytyä TAMK:n koneilta. Tarvittaessa (esim. jos käytät omaa konetta) asenna puuttuvat ohjelmat ja luo tunnukset palveluihin, joihin sinulla ei niitä vielä ole.

2.1 Tietokoneen vaatimukset

Projektia tehdään pääasiassa TAMK:n koneilla, jotka pyörittävät tarvittavia ohjelmia hyvin. Jos käytät omaa konetta, tulee varmistaa, että se täyttää seuraavat ehdot:

Archicadın vaatimukset (projektiin mid-range tai high-end):

	Recommended configuration			Minimum Hardware Requirements
Project size	Entry-level Residential homes	Mid-range Apartment and office Buildings	High-end High rises, Hospitals	
CPU	Intel Core i5 AMD Ryzen 5 Apple M1	Intel Core i7 AMD Ryzen 7 Apple M2	Intel Core i9 AMD Ryzen 9 Apple M2 Max	Intel/AMD x64 or Apple ARM multi-core processor
Memory	16 GB RAM	32 GB RAM	64 GB RAM	8 GB RAM
Disk	SATA SSD	NVMe SSD	NVMe SSD	5+ GB of free space
Graphics card	4 GB VRAM DirectX 11 compatible graphics cards	6 GB VRAM DirectX 11 compatible graphics cards	8 GB VRAM DirectX 11 compatible graphics cards	DirectX 11 compatible graphics card
	Note: To utilize the Redshift rendering engine, an NVIDIA graphics card with CUDA 7.0 or higher, or AMD's Navi/Vega is required. It is recommended to have at least 8GB of VRAM for optimal Redshift rendering performance.			
Display resolution	FHD+ (1920x1080)	2K (2560x1600)	5K (5120 x 2880)	1440x900
Operating System	Windows 10 Windows 11 macOS 14 Sonoma macOS 13 Ventura			Windows 10 64-bit (min. build 1809) macOS 12.6 Monterey
Optional Hardware Peripherals	All major plotters, printers, and digitizers can be used with Archicad. Driver support is provided by the operating system.			
Internet	Fast internet access is essential for tasks such as connecting to a user's Graphisoft ID, using the License Manager Tool, accessing BIMcloud from a remote location, and downloading updates for Archicad			
Protection key	To utilize Archicad 27 for commercial purposes, either a cloud license, a CodeMeter software key, or a CodeMeter hardware key is required.			

Unreal Engine 5 vaatimukset Windows:lle:

Recommended Hardware	
Operating System	Windows 10 64-bit version 1909 revision .1350 or higher, or versions 2004 and 20H2 revision .789 or higher
Processor	Quad-core Intel or AMD, 2.5 GHz or faster
Memory	8 GB RAM
Graphics Card	DirectX 11 or 12 compatible graphics card with the latest drivers

Unreal Engine 5 vaatimukset Mac:lle:

Recommended Hardware	
Recommended Operating System	Latest MacOS 13 Ventura
Minimum Operating System	macOS 12.5 Monterey
Processor	Quad-core Intel, 2.5 GHz or faster
Memory	8 GB RAM
Video Card	Metal 1.2 Compatible Graphics Card

Unreal Engine 5 vaatimukset Linux:lle:

Recommended Hardware	
Operating System	Ubuntu 22.04
Processor	Quad-core Intel or AMD, 2.5 GHz or faster
Memory	32 GB RAM
Video Card	NVIDIA GeForce 960 GTX or Higher with latest NVIDIA binary drivers
Video RAM	8 GB or more
RHI Version	Vulkan: AMD (21.11.3+) and NVIDIA (515.48+)

2.2 Archicad

Archicad on kolmiulotteinen rakennussuunnitteluohjelmisto. Projektissa sitä käytetään rakennusten mallintamiseen. Lataa tarvittaessa uusin versio MyGraphisoft -sivulta.

Ohjeissa oletetaan, että työntekijä hallitsee Archicadin käytön riittävän hyvin opintojen pohjalta, eikä sen käyttöä ohjeisteta erikoisia työvaiheita lukuun ottamatta.

Lisää Archicadista luvusta [5.1](#) eteenpäin.

2.3 Unreal Engine 5

Unreal Engine on pelimoottori. Sitä käytetään mm. teksturointiin, kasvillisuuden ja yksityiskohtien lisäämiseen ja lopputuotteen tekemiseen. Tarvittaessa rekisteröidy tai kirjaudu Epic Games:iin, asenna Epic Games Launcher ja asenna sieltä Unreal Enginen uusin versio.

Alustava tutoriaalivideo Unreal Enginen käyttöön, joka kannattaa katsoa: [Unreal Engine 5 for Architecture - Full Beginner Course \(youtube.com\)](#)

Lisää Unreal Enginen käytöstä luvusta [5.2.3](#) eteenpäin.

2.4 Datasmith

Datasmith export plugin on työkalu, jonka avulla voi siirtää projekteja Archicadista Unreal Engineen. Pluginilla voi linkittää archicad projektin Direct Link -ominaisuudella Unreal Engine projektiin ja työskennellä ohjelmistojen välillä reaaliaikaisesti. Toinen vaihtoehto on exportata tiedosto Archicadista ja importata tiedosto Unreal Engineen. Tarvittaessa lataa Graphisoft Archicad Exporter Unreal Enginen Datasmith -sivulta.

Lisää Datasmithin käytöstä luvuissa [5.2.2](#) ja [5.2.3](#)

2.5 Cetopo

Cetopo on nettisivu, josta voi ladata maastoa. Mikäli tehtävänäsi on maastomallin tekeminen, rekisteröidy tai kirjaudu Cetopoon. Tehtävänjaosta lisää [seuraavassa luvussa](#)

Maaston lisäämiseksi Archicad-projektiin tarvitset Cetopo pluginin Archicadiin. Tarvittaessa asenna plugin Cetopon Resources sivuilta. Lisää Cetopon käytöstä luvussa [5.1](#)

3 Työnjako

Jokaiselle työntekijälle jaetaan omat tehtävät ja vastuut. Työnjohtaja voi jakaa tehtäviä yksilöllisten vahvuuksien ja kiinnostuksen kohteiden mukaan, mutta pääasiassa seurataan seuraavissa luvuissa ilmenevää työkokoja. Myös riippuen kokoonpanosta, muutoksia vastuualueisiin voi tehdä.

Työtehtäviä on karkeasti ottaen:

- [Maastomallin tekeminen](#)
- [Korttelin tekeminen](#)
- [Yhdistäminen](#)
- [Julkaisu](#)
- [Johtaminen](#)

Näistä tehtävistä korttelien tekeminen vie eniten työvoimaa.

Työtehtävät on jaettu TAMK:n rakennusarkkitehtien vuosiluokkien mukaan. Katso tehtäväsi vuosiluokkasi mukaan seuraavista luvuista.

3.1 Ensimmäisen vuoden opiskelijat

Ensimmäisen vuoden opiskelijat keskittyvät pääasiassa korttelin tekemiseen. Siihen kuuluu [lähdemateriaalien tutkiminen](#), [korttelin mallintaminen](#), [korttelin teksturoiminen](#) ja [optimointi](#). Kaikessa saa myös apua kokeneemmilta työntekijöiltä.

Työnjohtaja osoittaa sinulle mallinnettavan korttelin. Voit myös ehdottaa mikä kortteli sinua kiinnostaa. Jos sinulla on mielessäsi jokin muu sinua kiinnostava tehtävä, johon olisit mielestäsi sopiva, ehdota sitä työnjohtajalle.

3.2 Toisen vuoden opiskelijat

Toisen vuoden opiskelijat keskittyvät pääasiassa korttelin tekemiseen. Siihen kuuluu [lähdemateriaalien tutkiminen](#), [korttelin mallintaminen](#), [korttelin teksturoiminen](#) ja [optimointi](#). Kaikessa saa myös apua kokeneemmilta työntekijöiltä.

Työnjohtaja osoittaa sinulle mallinnettavan korttelin. Voit myös ehdottaa mikä kortteli sinua kiinnostaa. Jos sinulla on mielessäsi jokin muu sinua kiinnostava tehtävä, johon olisit mielestäsi sopiva, ehdota sitä työnjohtajalle.

Riittäväällä kokemuksella toisen vuosiluokan työntekijä voi myös ottaa [kolmannen vuoden opiskelijan tehtäviä](#).

3.3 Kolmannen vuoden opiskelijat

Kolmannen vuoden opiskelijat ovat pääsääntöisesti kokeneimpia. Tämä luku käsittelee kokeneimpia työntekijöitä erityisesti VirtuaaliTampere -projektissa. Kokenein työntekijä nimitetään työnjohtajaksi. Tapauksessa, jossa jonkin muun vuosiluokan työntekijällä on enemmän kokemusta projektissa, ottaa hän sen sijaan työnjohtajan roolin. Riippuen työmäärästä, työnjohtaja voi nimittää myös apujohtajan, ja jakaa tehtäviä hänelle tai muille osaaville työntekijöille.

Työnjohtajan vastuisiin kuuluu:

- [Johtaminen](#)
- [Maastomallin tekeminen](#) ja [lähdemateriaalien tutkiminen](#)
- [Yhdistäminen](#)
- [Julkaisu](#)

Kolmannen vuoden opiskelijat, jotka eivät ole työnjohtajan roolissa, tekevät samoja tehtäviä, kuin [alempien vuosiluokkien työntekijät](#). Työnjohtajakin voi ottaa kyseisiä tehtäviä tekemisen puutteessa.

3.3.1 Johtaminen

Työnjohtaja on vastuussa kokonaisuudesta ja muista työntekijöistä. Tämä tarkoittaa seuraavia asioita:

- Tulevan suunnittelu ja seuraavien mallinnettavien kohteiden päättäminen
- Tehtävien jakaminen
- Kortteleiden osoittaminen työntekijöille

- Työntekijöiden tuotosten laadun tarkastaminen
- Työntekijöiden avustaminen
- Työohjeiden päivittäminen, [raportointi](#)
- Aikataulusta huolehtiminen
- Ongelmakohtien ja kysymysten esittäminen eteenpäin (esim. vastuupettajalle)

4 Materiaalit

Lähdemateriaalit pitävät sisällään:

Korttelit

- Asemapiirustukset/-kaavat
- Pohjapiirustukset
- Julkisivukuvat
- Valokuvat
- Värit ja materiaalit
- Kaikki mikä auttaa

Maastomalli

- Asemapiirustukset/-kaavat
- Valokuvat
- Viheralueiden ja kasvillisuuden sijoittelu
- Kaikki mikä auttaa

KAIKKI VIRTUAALITAMPEREEN TUOTOKSET POHJAUTUVAT LÄHDEMATERIAALEIHIN REALISTISEN JA TODENMUKAISEN KAUPUNKIMALLIN TOEUTTAMISEKSI. EI IMPROVISOINTIA!

4.1 Tiedostojen hallinta

Projektia varten koululla on yhteinen palvelin, jonne tallennetaan kaikki tiedostot sopiviin kansioihin. Tiedostot nimetään selkeästi ja ymmärrettävästi, jotta ne ovat helposti löydettävissä myös muille, kuin tiedoston luojalle. Näin voidaan hyödyntää esimerkiksi samoja tekstuureja useaan projektiin.

Jokaiselle korttelille annetaan koodi, jonka mukaan kaikki kortteliin liittyvät tiedostot nimetään. Koodit annetaan asemakaavan tai kiinteistötunnuksen mukaan.

4.2 Lähdemateriaalit

Alkuun projektiin saadaan lähdemateriaalia Vapriikilta. Tämä aineisto löytyy koulun palvelimelta.

Näistä lähteistä voi etsiä lisää lähdemateriaaleja, kuten piirustuksia ja kuvia:

- Aamulehden arkisto
- Kansallisarkisto
- Finna.fi
- Kadonneet kaunottaret
- Tampereen kaupungin piirustusarkisto
- kartat.tampere.fi/oskari
- paikkatietoikkuna
- vanhatkartat.fi
- rakennusvalvonnan arkisto
- seurakunnan arkisto/digitaalinen arkisto
- [E-lomake - Varauslomake asiakirjojen katselua varten \(tampere.fi\)](#)

4.3 Tekstuurit / objektit

Näiltä sivuilta voi etsiä tekstuureja ja objekteja:

- Quixel (suositeltu)
- AmbientCG
- Poly Haven
- Poliigon
- Architextures
- UE Marketplace
- CGTrader
- Textures.com
- Google kuvahaku

4.4 Muut lähteet

Muita tapoja kerätä materiaaleja:

- Valokuvaus
 - o Mallinnettavien rakennuksien materiaaleiksi voi luoda tekstuureja valokuvaamalla kyseisiä rakennuksia, jos ne ovat vielä olemassa. Valokuvia voi myös käyttää referenssikuvina, tai [tehdä ornamentteja displacement mapilla](#).
- Dronekuvaus
 - o Dronella voi ottaa esimerkiksi ortokuvia ja luoda niistä kattotekstuureja.
- Fotogrammetria
 - o Fotogrammetrialla voi luoda 3D-malleja valokuvien perusteella.
- Laserkeilaus
 - o Laserkeilaamalla saa tarkat mitat rakennuksesta

5 Tekniset ohjeet

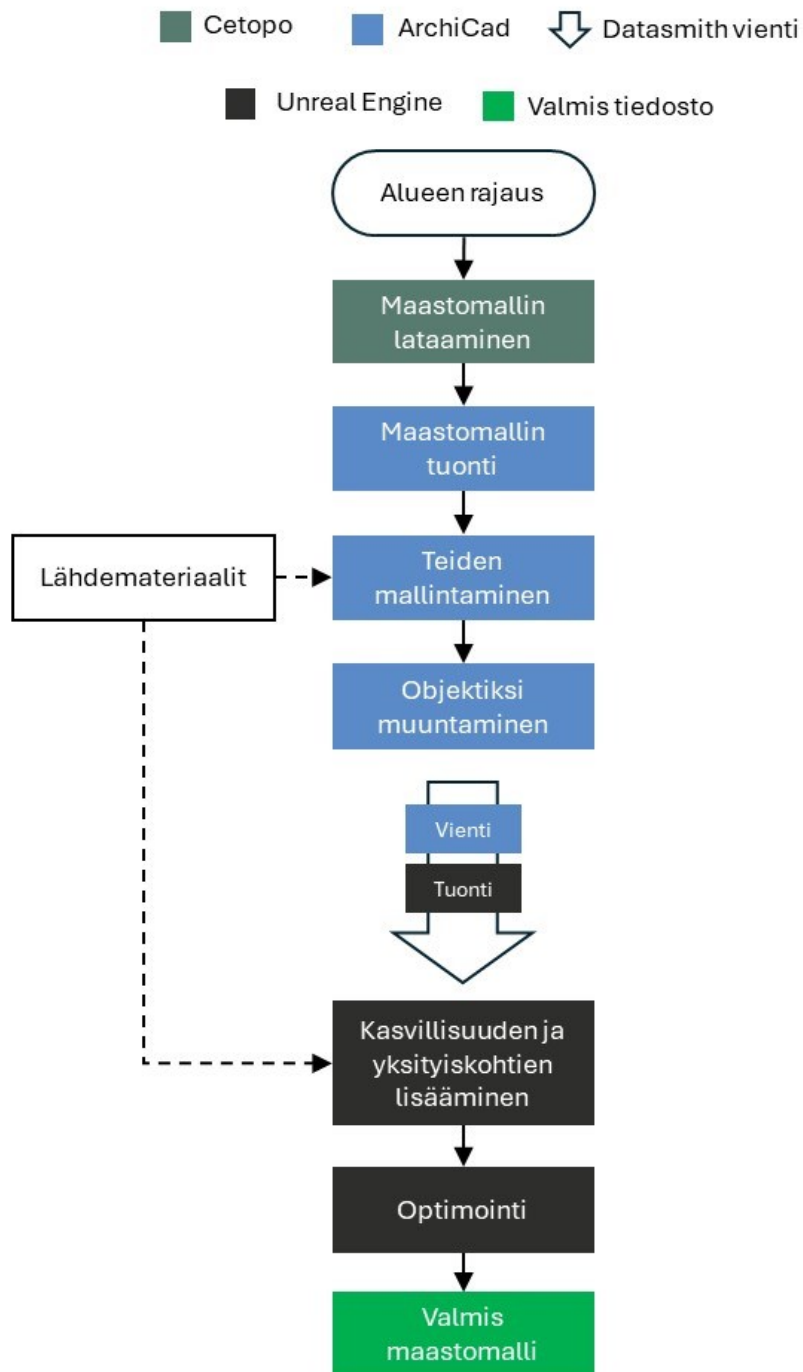
Tässä luvussa on ohjeet kaikista työvaiheista. Katso ohjeet tarvittaviin työvaiheisiin tehtävänkuvasi mukaan.

VirtuaaliTampereen tarkkuustaso on CityGML luokituksen mukaan LOD3 (täysin eri termi, kuin kappaleen 5.3.1 LOD, joka liittyy optimointiin). Tämä tarkoittaa tarkkaa esitystä rakennuksen arkkitehtuurista lukuun ottamatta rakennuksen sisätiloja ja kalustusta. Myös visuaalinen ilme on tärkeä ja materiaalit on oltava realistisia, totuudenmukaisia ja laadukkaita. Maastomalli, kasvillisuus, näkyvä infrastruktuuri ja kadulla olevat esineistö ja yksityiskohdat tulevat näyttää realistiselta ja laadukkaalta. Lopputuotteesta pyritään saamaan ajankohtaisen teknologian puitteissa mahdollisimman visuaalisesti näyttävä. On kuitenkin muistettava pitää huolta, ettei polygonien määrä kasva liian suureksi.

Tarkemmin vaatimusten toteuttamisesta seuraavissa luvuissa.

5.1 Maastomalli

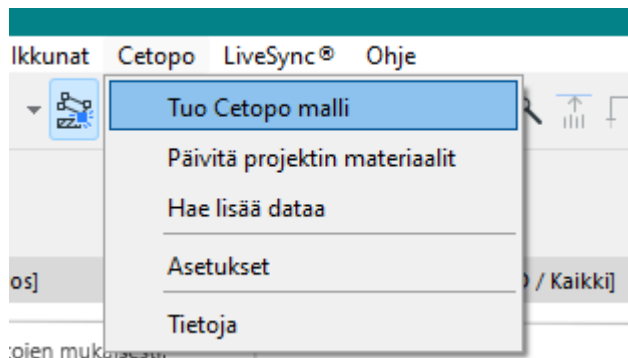
Maastomallin työvaiheet



Maastomalliin ladataan maaston muoto Cetopon nettisivuilta. Tähän on valmiiksi template VirtuaaliTampereen palvelimella. Templatea voi tarvittaessa muokata. Cetopon nykytilanteen maastomallia voi hyödyntää läpi Tampereen lähihistorian. Tampereen maasto on pysynyt 1900-luvulta keskustan osin kutakuinkin samana

lukuun ottamatta 1930-luvulla rautatieaseman ympäristön maan pinnan kaivamista. Maastomalliin tilanteesta ennen 1930-lukua on mallinnettava korkeampi maanpinta cetopon maastomallin päälle. 1930-luvun jälkeen voi käyttää pitkälti suoraan cetopon mallia. Yksityiskohtaisemmat maan muokkaukset täytyy kuitenkin varmistaa vuosikymmenkohtaisesti.

Maastomallin tuonti Archicadiin onnistuu cetopo-pluginin avulla:



Valitse lataamasi cetopo tiedosto.

Tiet täytyy mallintaa itse piirustusten, kuten asemakaavojen tai karttojen perusteella. Cetoposta on saatavilla vain nykyhetken tieverkosto, mikä ei ole käytökelpoinen kovinkaan pitkälle historiaan. Mallintaminen onnistuu laattatyökälulla. Tiet piirretään laatalla pohjanäkymässä, jonka jälkeen ne booletetaan maanpinnan mukaisesti käyttämällä boolean toimenpiteitä (leikkaus).

Tarvittaessa maastomallitiedoston kokoa voi pienentää muuntamalla maastomallin ensin objektiksi ennen datasmith tiedostoksi muuntamista. Tämä vaihe ei välttämättä ole pakollinen, jos datasmith tiedosto ei ole liian suuri.

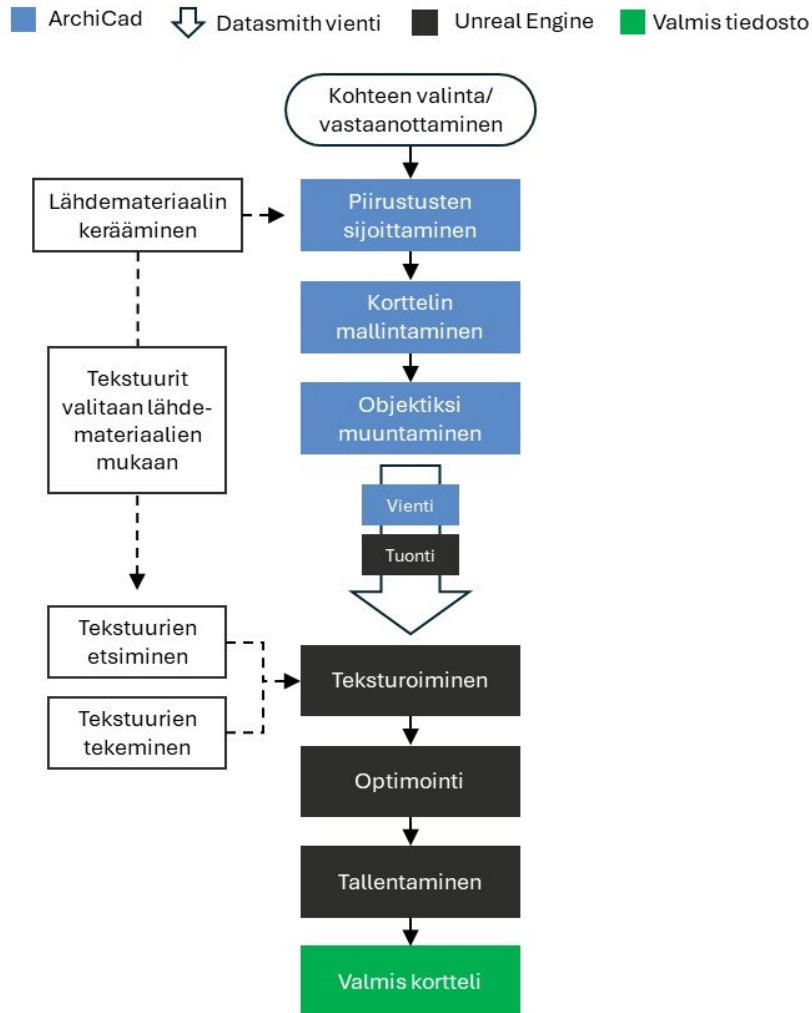
Objektiksi ja Datasmith-tiedostoksi muuntaminen on ohjeistettu luvussa [5.2.2](#). Unreal Engineen tuonti on ohjeistettu luvussa [5.2.3](#)

Unreal Engineessä luodaan kasvillisuus ja yksityiskohdat kuten katulamput yms. Myös teille ja maastolle annetaan materiaalit. Kasvillisuuden lisäämisestä ohjeita luvun [2.4](#) tutoriaalissa. Materiaalien luomisesta luvussa [5.2.4](#)

Optimoinnista lisää luvussa [5.3](#)

5.2 Kortteli

Korttelin työvaiheet



5.2.1 Korttelin mallinnus

Kortteli tai rakennus mallinnetaan Archicadissa pitkälti koulussa opituilla taidoilla. Ensimmäiseksi tarvitaan lähdemateriaalit, mielellään ainakin pohjapiirustus ja julkisivukuvat. Archicadissa luodaan vastaavat näkymät ja tuodaan piirustukset näkymille oikeassa mittakaavassa. Piirustuksista voi halutessaan muuttaa taustan läpinäkyväksi. Sen jälkeen rakennusta voi alkaa mallintamaan piirustusten päälle.

Mahdollisuuksia saada rakennukset oikeaan kohtaan maastomallissa on kaksi.

- Jo rakennuksien mallinnusvaiheessa ladataan kartta pohjaksi ja mallinnetaan rakennus oikeisiin koordinaatteihin, jolloin rakennus on maastomalliin tuodessa oikeassa kohdassa
- Mallinnetaan rakennus ilman koordinaatteja ja asetetaan rakennus oikeaan kohtaan manuaalisesti

Molemmissa tavoissa on puolensa ja paras tekniikka selviää kokeilemalla. VirtuaaliViipurissa käytettiin jälkimmäistä tapaa, mutta ainakin pienimmissä projekteissa koordinaattien hyödyntäminen on viisasta.

Jos mallintaa kokonaisen korttelin, kannattaa maastoa ottaa korttelin alueelta referenssiksi, jotta rakennukset asettuvat oikeille koroille.

Jokainen rakennuksen materiaali täytyy olla eri pintamateriaalilla/surfacella, jotta myöhemmässä vaiheessa Unreal Engineissä jokaiselle voi määrittää oman materiaalin.

Archicadissa voi tehdä joitain yksinkertaisia yksityiskohtia palkeilla, pilareilla ja muunteilla, mutta Unreal Engineissä on paremmat keinot monimutkaisempiin yksityiskohtiin. Ks. luku [5.2.5](#). Mallintaja voi itse päättää, mikä sopii mihinkin tilanteeseen parhaiten.

Mallintaessa ota huomioon, että jokainen yksityiskohta ja muoto lisää polygoneja, joka hidastaa mallin pyörittämistä. Esimerkiksi ikkunoihin kannattaa käyttää yksiläisiä ikkunaobjekteja polygonien määrän minimoimiseksi.

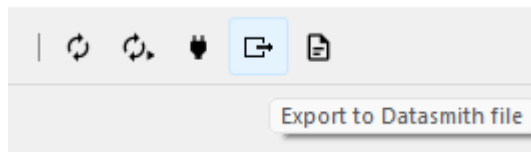
5.2.2 Vienti

Kun kortteli/rakennus on täysin valmis, se muunnetaan objektiksi. Objekti vie vähemmän tilaa ja useita kortteleita/rakennuksia on helpompi käsitellä samassa projektissa objektina. Kun objekti on tehty, sitä ei voi muokata enää. Jos muutoksia haluaa tehdä, ne pitää tehdä alkuperäiseen malliin ja tehdä uusi objekti.

Objektiksi muuntaminen:

- Avaa 3D näkymä
- Laita näkyville kaikki ja pelkäästään ne mitä haluat objektiksi
- Valitse kaikki
- Arkisto -> Kirjastot ja objektit -> Tallenna valinta nimellä -> Objekti
- (File -> Libraries and objects -> Save selection as -> Object)

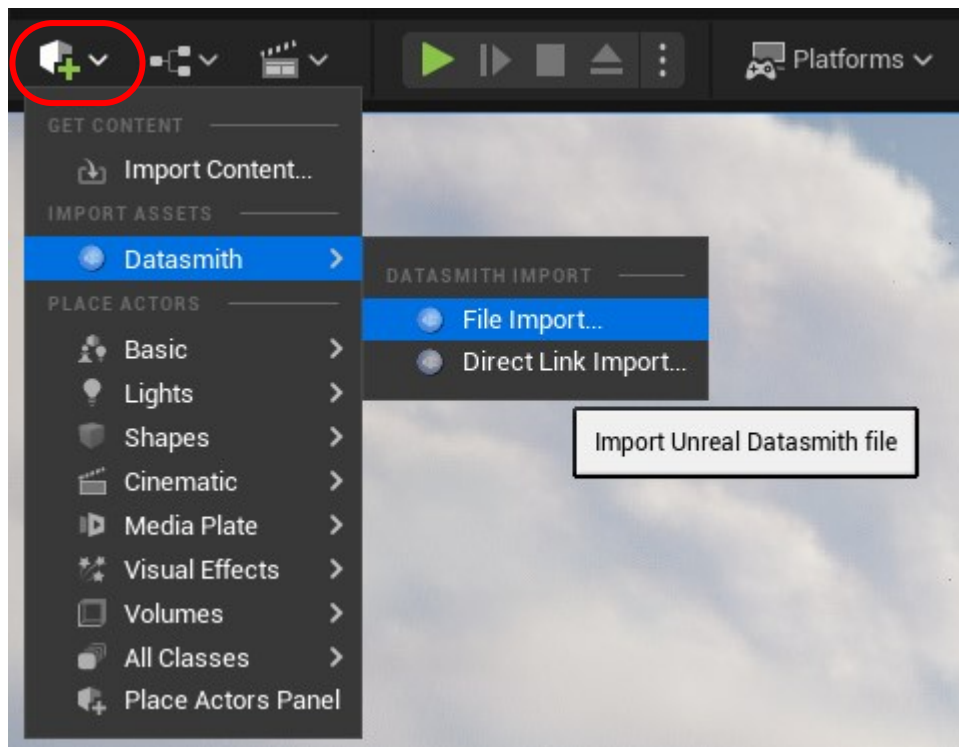
Sijoita objekti 3D näkymään ja laita vain objekti näkyville ja piilota kaikki muu. Exporttaa Datasmith tiedostoksi Datasmith exporter pluginilla omaan group drive kansioosi.



Voit myös halutessasi kokeilla Direct Link ominaisuutta, mutta tiedostoksi exporttaaminen on luotettavampi ja yksinkertaisempi tapa.

5.2.3 Tuonti

Seuraavaksi Datasmith tiedosto tuodaan Unreal Engineen. Luo uusi Unreal Engine projekti ja nimeä se selkeästi. Luo projektisi Content Draweriin kansio "Geometry". Paina:



Valitse Archicadista viemäsi datasmith tiedosto ja tuo tiedosto Unreal Engineen luomaasi Geometry kansioon.

5.2.4 Teksturoiminen

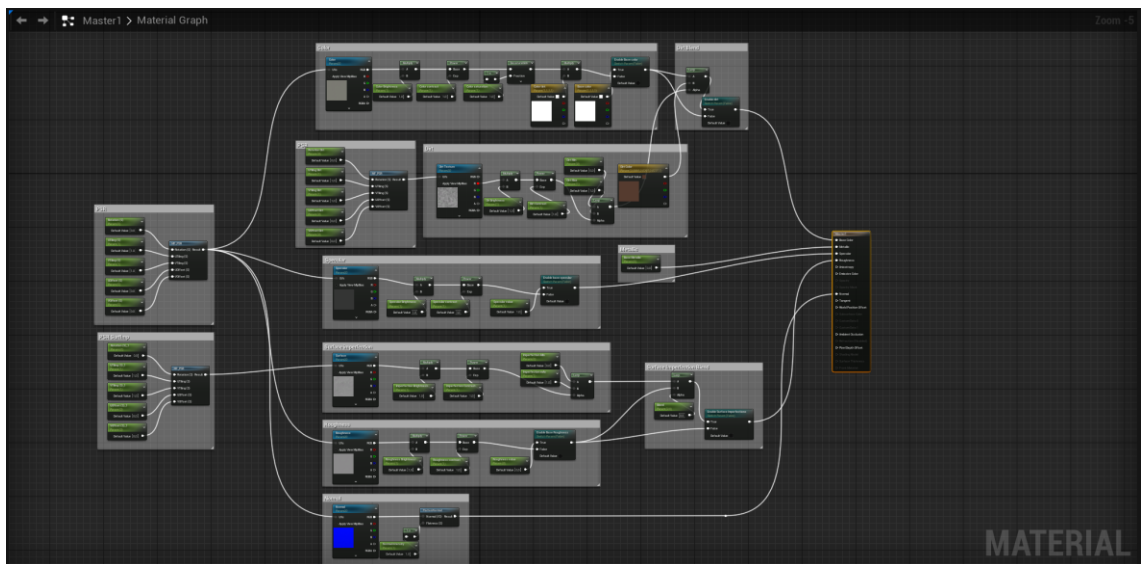
Materiaalit koostuvat useista mapeista eli valokuvista, jotka kertovat tarvittavan tiedon. Yleisimpiä ovat:

- Albedo/Diffuse/Color
- Normal
- Roughness
- Specular
- Opacity
- Displacement
- Metallic

Albedo on materiaalin väri, eli värillinen kuva materiaalista. Normal kertoo ohjelmalle, miten valot ja varjot muodostuvat materiaalin pinnalle, mikä saa materiaalin näyttämään kolmiulotteiselta ilman ylimääräistä geometriaa. Roughness on materiaalin kiiltävyys. Specular kertoo kuinka tarkka tai sumea materiaalin heijastus on. Opacity on materiaalin läpinäkyvyys. Displacement mapilla voi luoda oikeaa geometriaa. Metallic luo metallimaista kiiltoa.

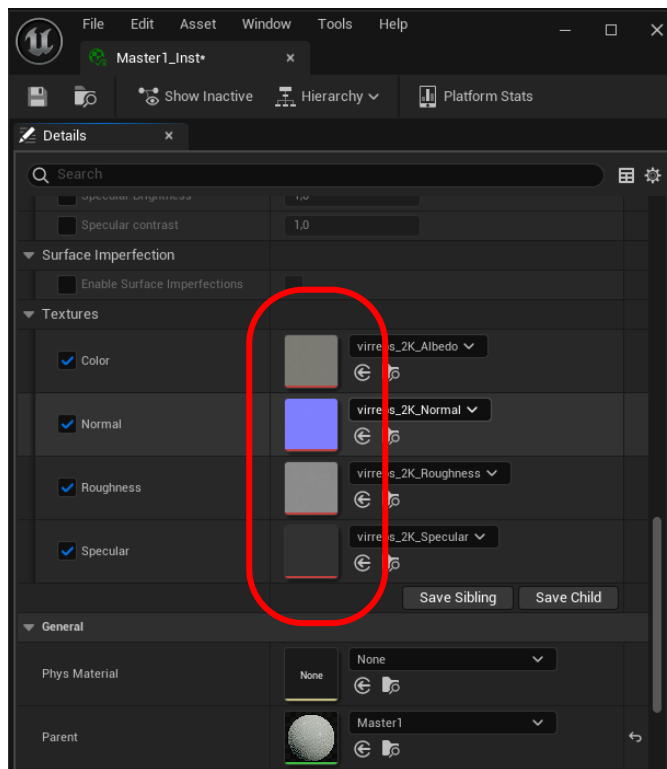
Luvussa 4.3 esitellyistä lähteistä voit etsiä materiaaleja ja ladata tarvittavat mapit.

Teksturoiminen tapahtuu Unreal Engineissä. Jokaista materiaalia varten on master materiaali, josta voi muokata haluamansa materiaalin. Master materiaali löytyy palvelimelta (S:\virtuaalitampere_sto-3358\Materiaalit\MasterMaterial1\Content\MasterMaterial). Kopioi (HUOM! ÄLÄ SIIRRÄ) koko MasterMaterial kansio oman projektisi vastaavaan Content kansioon ja käynnistä Unreal Engine tarvittaessa uudelleen. MasterMaterial on luotu [tämän](#) tutoriaalin pohjalta, jos haluat tietää siitä lisää.

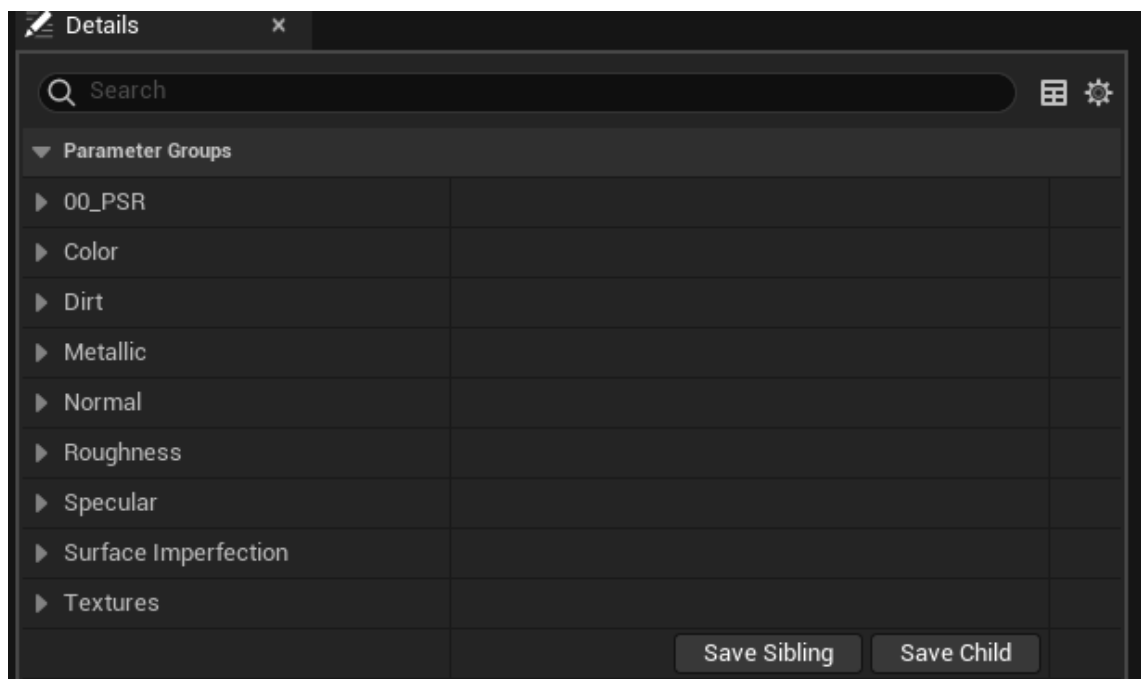


MasterMaterialin material editor. Tätä ei tarvitse muokata

MasterMaterialista luodaan uusia instanceja painamalla siitä hiiren oikealla näppäimellä ja valitsemalla "new instance". Näistä instanceista tehdään tarvittavat materiaalit. Tuo projektisi materiaalin tekstuurit ja mapit ja siirrä ne luomaasi instanceen raahaamalla tekstuurin kunkin kuvan päälle.



Laittamalla nämä neljä tekstuuria materiaaliin saat jo hyvännäköisen materiaalin. Sen lisäksi voit vapaasti muokata materiaalin muita parametrejä.



Valmiin materiaalin sävyn korjaaminen onnistuu asettamalla albedo saturation nolaksi, jolloin alkuperäinen albedotekstuuri on mustavalkoinen, ja sen jälkeen lisäämällä albedo tint:iin oikea väri.



Oikean värin voi saada valokuvasta korjaamalla valkotasapainon photoshopissa. [Best 3 Ways to Fix Bad Color Cast on Photos \(youtube.com\)](#). Sen jälkeen valitse pipetillä oikea väri ja kopioi RGB arvot photoshopista Unreal Engineen. Voit joutua säätämään värin kirkkautta, jotta se näyttää oikealta, mutta sävyn pitäisi olla oikea. Jos saatavilla ei ole kuvaa kohteesta tai muuta tietoa väristä, joudut arvioimaan värin silmämääräisesti.

5.2.5 Yksityiskohdat

Jotkut yksityiskohdat ovat liian monimutkaisia Archicadille, jolloin turvaudutaan Unreal Engineen. Vaihtoehtoja on ainakin netistä 3D-objektin lataaminen, alusta asti itse mallintaminen ja displacement mappien käyttäminen.

Luvussa 4.3 käydyistä lähteistä voi etsiä valmiita 3D-objekteja. Näistä voi saada sopivan yksityiskohdan suoraan tai yhdistelemällä useita. Hakusanaksi voi laittaa esim. "Ornament 3D models". Ensimmäisenä kannattaa tutkia, mitä on jo olemassa ennen kuin alkaa itse tekemään. Jos mallinnettavalle yksityiskohdalle ei löydy kutakuinkin samanlaista objekta, tai yksityiskohta on niin tunnistettava, että sen täytyy olla juuri samannäköinen, pitää yksityiskohta mallintaa itse. **ÄLÄ LAATAA LIIAN RASKAITA**

[Tämän](#) tutoriaalin opettamilla keinoilla voi koittaa mallintaa itse. Mallintaminen sopii parhaiten yksityiskohtiin ja koristeisiin, jotka ulkonevat joka suuntaan, ja joille ei löydy valmista 3D-objektia netistä

Yksityiskohdat ja koristeet, jotka ovat yhdellä pinnalla ja ulkonevat vain yhteen suuntaan, voi tehdä displacement mapilla. Siihen tarvitset tarkan valokuvan kohdesta neutraalissa hajavalotuksessa ja mielellään mahdollisimman suorassa perspektiivissä. Suorista ja rajaa valokuva perspective crop -työkalulla photoshopissa ja editoi mahdolliset heittovarjot pois. Valokuvasta tehdään displacement map photoshopissa kohdasta Filter -> 3D -> Height map. Tee tämä vaihe kopioidulle layerille, jotta menetät alkuperäistä kuvaa.

Voit myös tehdä displacement mapin itse, jos valokuvaa ei löydy tai et voi käydä valokuvaamassa. Displacement map on mustavalkoinen kuva, jossa tummemmat arvot ovat pinnasta sisääpäin ja vaaleammat arvot pinnasta ulospäin. 50 % harmaa on pinnan tasolla.

Seuraavaksi displacement map tuodaan Unreal Engineen ja lisätään objektiin [tämän](#) tutoriaalin mukaisesti. Halutessasi voit vielä asettaa valokuvan materiaalin albedo tekstuuriksi.

5.3 Optimointi

Korttelit/rakennukset ja maastomalli täytyy optimoida, jotta kaupunkimalli pyörii pelinä jouhevasti. Unreal Engine on tehokas pelimoottori ja pystyy pyörittämään todella yksityiskohtaisiakin malleja, mutta ihmeisiin sekään ei kykene. Tämä on otettava huomioon jo mallinnusvaiheessa, käyttämällä harkitusti polygoneja tarvittavissa määrin. Tavoitteena VirtuaaliTampereessa on mallintaa kaikki yksityiskohdat tarkasti ja mahdolliset kolmiulotteiset yksityiskohdat mielellään kolmiulotteisesti, kuitenkin pitämällä huoli, ettei polygonien määrä kasva liian suureksi.

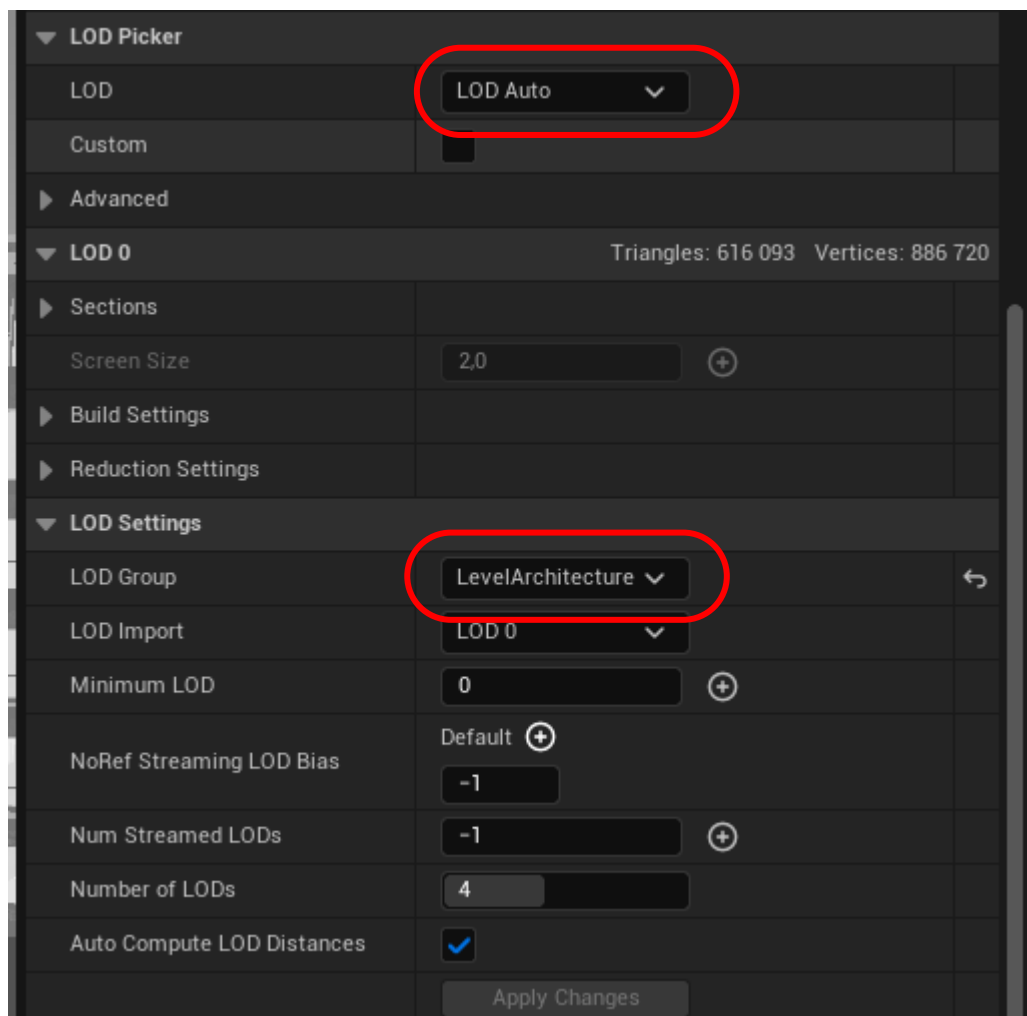
Mallintamisen jälkeen mallia voi optimoida edelleen käyttämällä LODeja tai Nanitea. Näistä Nanite on ensisijaisesti käytettävä optimointimenetelmä. Nanite kannattaa aktivoida kaikissa mahdollisissa geometrioissa, jos se ei aiheuta vikoja mallissa, valaistuksessa tai jollain muulla osa-alueella. Optimointimenetelmä on kuitenkin valittava parhaimman tuloksen ja suorituskyvyn mukaan, jota voi testata esimerkiksi pelaamalla tasoa ja seuraamalla FPS:ää.

Naniten käytössä on huomattu ongelmia archicadista tuotujen mallien kanssa. Ongelmat saattavat johtua joko tietokoneesta, Unreal Enginen versiosta tai archicad mallista. Harjoitteluprojektin edetessä asiaa pitää tutkia enemmän.

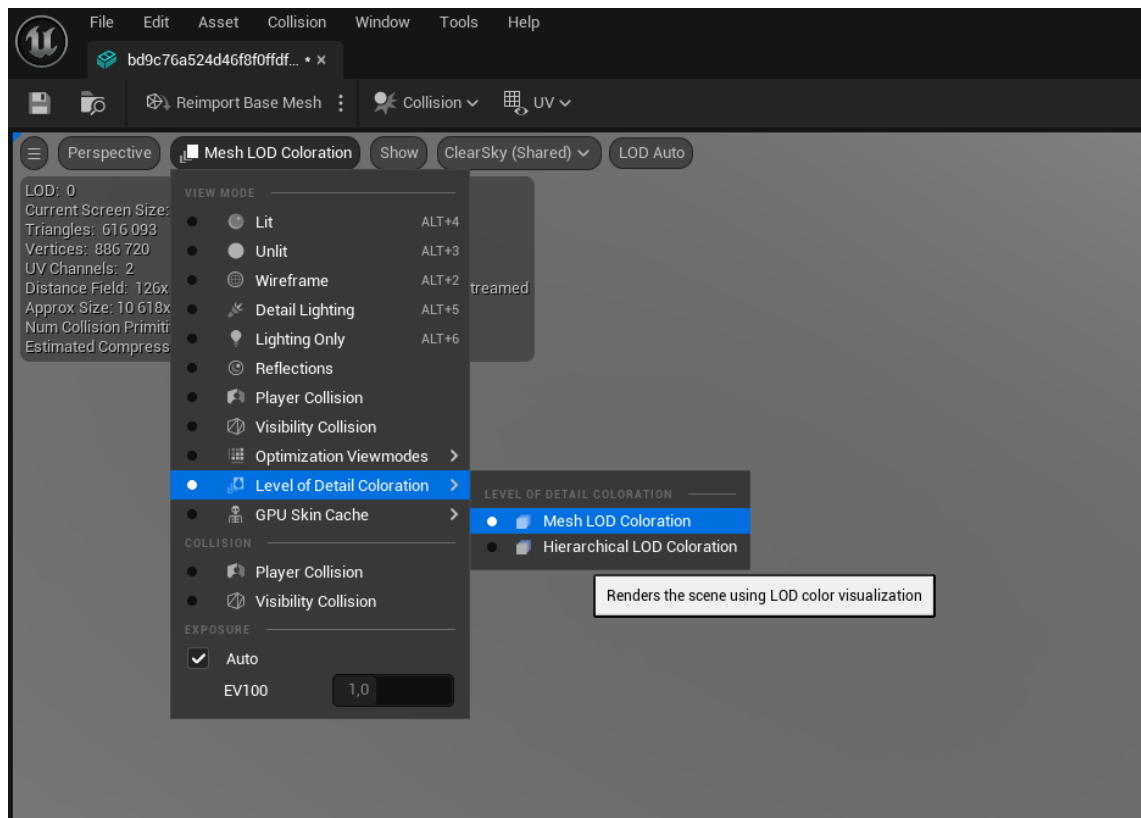
5.3.1 LOD

LOD eli Level Of Detail tarkoittaa sitä, kuinka tarkasti geometriat näytetään (eri termi, kuin CityGML LOD). Tarkkuustasoja on yleensä muutamia tarkimmasta, eli alkuperäisestä tarkkuudesta, epätarkimpaan ja yksinkertaistetuimpaan. Katsojaa lähellä olevat geometriat näytetään tarkimpana kaikki yksityiskohtat mukaan lukien ja mitä kauempana geometriat näkyvät, sitä vähemmän yksityiskohtia niistä näytetään. Tämä säästää huomattavasti prosessointitehoa, koska jokaista kauimmaistakin yksityiskohtaa ei tarvitse luoda näytölle.

LODin asetukset löytyy painamalla geometriasta hiiren oikealla ja valitsemalla edit. Näin aukeaa details-välilehti kyseiselle geometrialle. LOD Auto ja LevelArchitecture asetukset toimivat jo hyvin, mutta tarvittaessa asetuksia voi muokata tilanteen mukaan.



LODien toimivuuden voi tarkistaa laittamalla seuraavan view moden päälle:



Yllä olevassa näkymässä on esitetty eri LODit eri väreillä, joiden vaihtumisen huomaa liikkeessa kauemmas tai lähemmäs. Myös ”Triangles” ja ”Vertices” näkymän vasemmassa laidassa pitäisi muuttua.

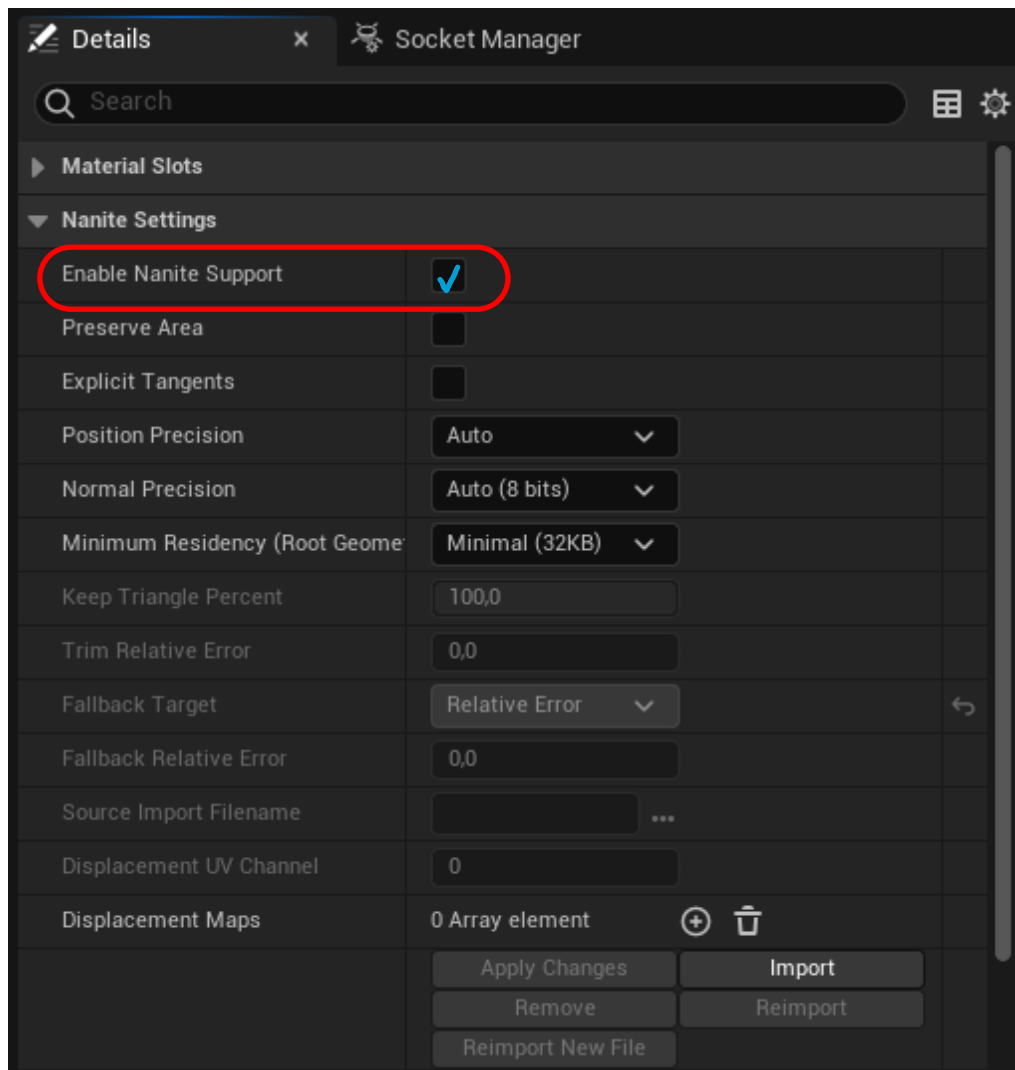
5.3.2 Nanite

Nanite on Unreal Engine 5 päivityksen mukana tullut uusi dynaaminen optimointitekniikka, joka pienentää geometrioiden monimuotoisuutta automaattisesti. Sen sijaan, että tarkkuustasot vaihtuisivat porrastetusti (LOD), Nanite pienentää polygonien määrää liukuvasti kauemmas mentäessä. Lopputulos on sulavampi ja tehokkaampi.

Naniten käyttöönotto on yksinkertainen.

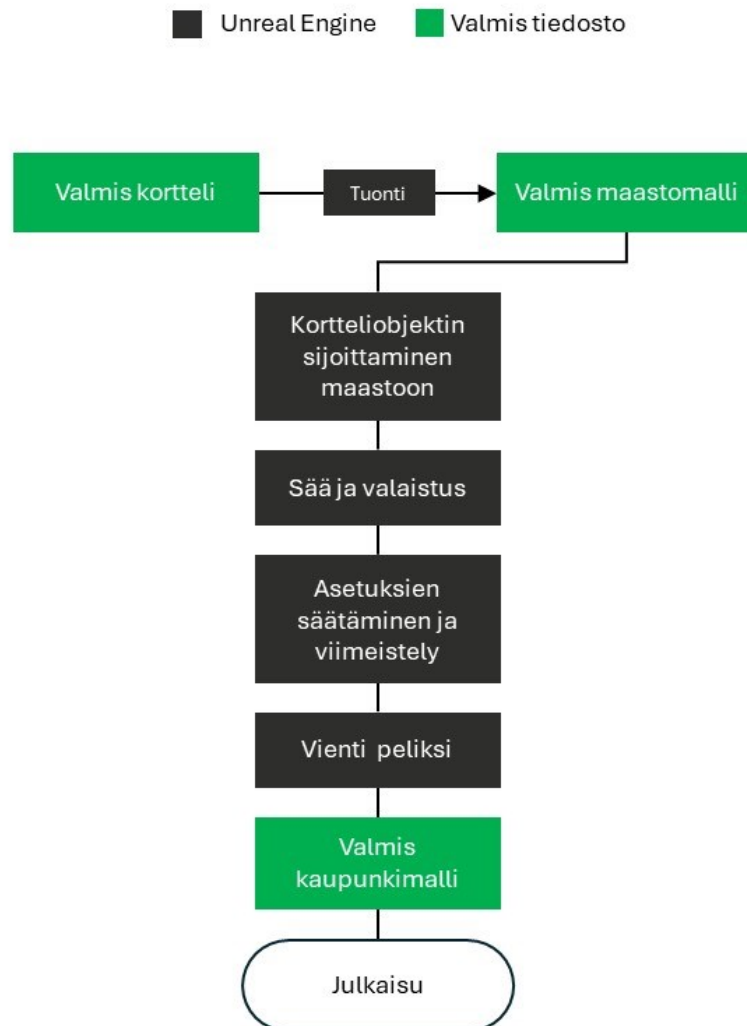
- Project settings
 - o Hae Nanite
 - Varmista, että Nanite ruudussa on rasti
- Joko:
 - o Valitse yksi tai useampi geometria Content drawerista, klikkaa hiiren oikealla, valikosta kohta ”Nanite” ja rasti ruutuun.

- Tai, paina geometriaa hiiren oikealla, valitse "Edit" ja laita rasti ruutuun "Enable Nanite Support"



5.4 Yhdistäminen

Yhdistämisen työvaiheet



Kun kortteli tai yksittäinen rakennus on valmis, tuodaan se maastomalliin [tämän](#) ohjeen mukaan. Tämän vaiheen toteutus riippuu siitä, onko rakennukset sijoitettu valmiiksi oikeisiin koordinaatteihin. Jos käytetään manuaalista tapaa, täytyy rakennukset sijoittaa silmämääräisesti oikeille paikoilleen.

Projektille päätetään jokin tietty kellon aika ja sää ja asetetaan asetukset sen mukaan. Yleisesti hyväksi todettu sää on puolipilvinen keskipäivän aikoihin. Näin saadaan tasainen valaistus ilman liian isoja ja kontrastisia varjoja. Rendereityihin kuviin verrattuna reaaliaikaisessa kaupunkimallissa on enemmän pelivaraa, ja sääskenaarioita voi tehdä jopa useampia tai liukuvan. Etuna tässä on se, että kaupunkimallia voi esitellä myös esimerkiksi tunnelmallisessa iltavalaisuksessa kaupungin omien valonlähteiden, kuten katulyhtyjen ja ikkunoista paistavien valojen kanssa. Kyseinen valaistus vaatii tietysti lisää työtä ja on todennäköisesti ajankohtainen vasta myöhemmässä vaiheessa projektia.

Koko projektin optimointiasetukset ja muut yleiset asetukset täytyy tarkistaa tässä vaiheessa. Jos ei ilmene ongelmia, on projekti valmis vientiä varten.

Projekti viedään ainakin peliksi muotoon .exe. Tässä muodossa saadaan irti kaikki hyödyt, mitä Unreal Enginellä on tarjota. Pelissä voi liikkua vapaasti ja mallien tarkkuus on niin hyvä kuin mahdollista. Vienti tehdään [tämän](#) ohjeen mukaan.

Sen lisäksi on tutkittava mahdollisuuksia kevyemmästä versiosta kaupunkimallin esittelyyn, joka toimisi myös selaimessa. Tästä lisää seuraavassa kappaleessa.

6 Julkaisu

Kaupunkimalli julkaistaan todennäköisesti ainakin Unreal Enginellä pyörivänä ”pelinä” (.exe) jossa voi liikkua vapaasti ja katsella ympärilleen. Peliä voisi pelata vr-laseilla, ja tätä varten järjestettäisiin vr-tila Vapriikkiin. Asiasta on keskusteltu Vapriikin kaupunkimallivastaavan Antti Liuttusen kanssa.

VirtuaaliTampereelle voisi tehdä omat nettisivut. Nettisivuilla tulisi olla esimerkiksi tietoa projektista ja mahdollisesti demovideo. Selaimessa toimivan kaikille saatavilla olevan kaupunkimallin mahdollisuutta voisi tutkia. Jos edellä mainittu ei ole mahdollista, nettisivuille voisi renderöidä valokuvia tai videoita kohteista. Myös panoraamakuvat, joiden välillä voi hyppiä, on yksi mahdollisuus.

7 Raportointi

Työnjohtaja raportoi tähän lukuun, mitä on jo tehty ja mitä suunnitelmia tulevaisuudelle on, jotta eri vuosien harjoittelijat pysyvät kartalla.

7.1 Työnalla/tehty

Kohteita:

- Raatihuone
- Tampereen teatteri

Maastomallia on aloitettu tekemään vuoden 1914 tilanteesta.

7.2 Suunnitelmat/Muistiinpanot

Projekti aloitetaan 1914-luvun Hämeenkadusta ja sen varrella olevista rakennuksista.

Esimerkki kohteita:

- Keskustorin ympäristö
- Nordean talo
- Sumeliuksen talo
- Vanha kirkko
- Aleksanterin kirkko
- Commercen talo
- Palanderin talo,
- Vanha rautatieasema

Rajaus:

puutarhakadusta poikki

aloitetaan rautatieaseman ja Aleksanterin kirkon välistä

Muistiinpanot:

- 30-luvulla rautatieaseman ympäristön maanpintaa muokattu
- yhteyshenkilö seurakuntaan: joku krista

- valokuvista/kartoista mallinnetaan tiet
- hämeensilta on muuttunut
- Rautatieaseman vieressä pinninsilta

Suunnitelmat pidemmälle tulevaisuuteen:

- Aluetta toivotaan laajennettavan. Myös muita vuosikymmeniä(?) voisi mallintaa ja tehdä joko yksi peli jossa slideri vuosille, tai eri levelit eri vuosille.
- Taustamusiikin/äänimaiseman luomisesta on puhuttu erään TAMKissa opiskelleen musiikintuottajan kanssa.

7.3 Hyödylliset linkit

- [Unreal Engine 5 for Architecture - Full Beginner Course \(youtube.com\)](https://www.youtube.com/watch?v=...)
- [E-lomake - Varauslomake asiakirjojen katselua varten \(tampere.fi\)](https://www.tampere.fi/...)
- [Unreal Engine Materials in 6 Levels of Complexity \(youtube.com\)](https://www.youtube.com/watch?v=...)
- [Best 3 Ways to Fix Bad Color Cast on Photos \(youtube.com\)](https://www.youtube.com/watch?v=...)
- [Unreal Engine 5 Beginner Modeling Tutorial - Learn to Model Inside Unreal! \(youtube.com\)](https://www.youtube.com/watch?v=...)
- [How to use displacement maps in unreal engine 5 - Micro tutorials for beginners \(youtube.com\)](https://www.youtube.com/watch?v=...)
-

7.4 Yhteyshenkilöt

- Oska Piirainen, projektin suunnittelija, ohjeiden laatija
- Jaakko Aumala, TAMKin lehtori, projektin ohjaava opettaja
- Minna Nyström-Järvinen, TAMKin lehtori, koulutusvastaava
- Antti Liuttunen, Vapriikin kaupunkimallivastaava