

Iita Merilehto

Tyylitellyn 3D-skenen luominen portfolioa varten



Tradenomi
Tietojenkäsittely, peliala
Syksy 2024



KAMK • University
of Applied Sciences

Tiivistelmä

Tekijä: Iita Merilehto

Työn nimi: Tyyllitellyn 3D-skenen luominen portfolioa varten

Tutkintonimike: Tradenomi (AMK), tietojenkäsittely.

Asiasanat: 3D-mallinnus, 3D, portfolio, peligrafiikka, peliympäristö, pelitaide, pelimoottori.

Opinnäytetyössä luotiin tyyllitelty 3D-ympäristö tekijän pelialan graafista portfolioa varten. Tässä kirjoitusosuudessa käytiin läpi työn eteneminen ja avattiin työskentelytapoja mahdollisimman selkeästi, jotta 3D-mallintamisesta mahdollisesti vasta vähän tietävä lukija myös ymmärtäisi. Työn käytännön osuus sisälsi teoksen suunnittelun, mallien mallintamisen ja valmiiden mallien siirron pelimoottoriin, jossa skene renderöitiin video ja kuva muotoon.

Työ aloitettiin teoksen suunnittelulla, sisältäen tyyllivalinnan ja työn luonnostelun. Vaikeuksia tuotti oikean tyylin löytäminen, mutta siinä päädyttiin vähä-keskivertopolygoniseen, kun huomattiin ensimmäisen tyyllivalinnan hyvin tyyllitellystä ja selvästi vähäpolygonisesta olleen liian haastava toteuttaa. Käytettäviksi ohjelmistoiksi valikoitui koulussa ja vapaa-ajalla eniten käytetyt Blender, Krita, Illustrator ja Unreal. Mallintaminen suoritettiin laatikkomallintamistekniikalla ja teksturointi tekstuuriatlasta hyödyntäen. Mallit mallinnettiin Blenderissä, joillekin tehtiin animaatiot ja kaikki teksturoitiin. Sitten nämä vietiin pelimoottori Unreal 5:een fbx-muodossa. Skene itsessään, mallien valmistuttua, kasattiin ja renderöitiin siellä.

Opinnäytetyö koskee pelialaa, joten projektiin oli aiheellista ottaa pelimoottori mukaan, vaikka koko teoksen olisi voinut toteuttaa vain Blenderissä. Kaikki mallit saatiin mallinnettua yhtenevällä tyyllillä ja koko skene näyttää sopivan yhteen. Tekstuuriin sävyt onnistuttiin myös valitsemaan toistensa kanssa sopiviksi. Niiden lopullista kuvissa (ja videossa) näkyvää värikylläisyyttä muokattiin vielä Unrealin Post Process-efektin avulla.

Tekijän tarkoitus oli keskittyä työssään mallintamiseen, mutta lopulta kuitenkin monet muut projektille olennaiset asiat veivät jopa enemmän huomiota. Usein nämä koskivat osa-alueita, joista tekijällä oli vähemmän kokemusta ja jotka piti oppia työn edetessä esim. materiaalien ja efektien teko vei yllättävän paljon aikaa. Tätä olisi voinut välttää ostamalla ja hyödyntämällä näihin osa-alueisiin erikoistuneiden artistien valmiiksi tekemiä materiaaleja ja efektejä. Teos kuitenkin saatiin tarpeeksi korkeaksi laatuiseksi ja julkaistua portfolioon ([linkki](#)), ArtStationiin.

Abstract

Author: lita Merilehto

Title of the Publication: Creating a Stylized 3D Scene for Portfolio

Degree Title: Bachelor of Business Administration, Business Information Technology

Keywords: 3D-modeling, game art, environment art, portfolio, scene creation, game engines, vfx

This Bachelor's thesis is about modeling a stylized 3D scene for game art portfolio. The theoretical part of the thesis goes through the process of creating the digital piece and opens some relevant concepts in 3D modeling world as simply as possible, for even the less inexperienced in 3D modeling to read. The practical part of the thesis included planning of the concept, modeling 3D models and moving them to the game engine for final compositing and rendering to picture and video format.

The art piece itself started with planning, so choosing a style and doing some sketching. Finding the right style proved to be slightly difficult. A stylized and very lowpoly style was too hard to learn fast enough for this project so the final style was a little more towards mid poly. Programs chosen to be used, based on the author's familiarity with them, were Blender, Krita, Illustrator, and Unreal. Box modelling was used as a modelling technique and texturing was done using a texture atlas. Modeling was done in Blender as along with the animations for some of the props, all of which were also unwrapped (and textured). The models were imported to Unreal game engine as fbx files.

The thesis targets game industry so it was fitting to implement the use of a game engine for the project pipeline, even though all that was done there could have been done in Blender. The project was overall a success. All models fit together style-wise, and the whole scene looks cohesive. Colors used in textures are also pleasing for the eye and fit together. Their final look in rendered pictures (and video) was tweaked by using Unreal's Post process effect.

The author's aim was mostly to focus on the modeling, but they were easily distracted by other relevant items the project needed. Usually, these were concepts that the author had only a limited amount of experience on beforehand and thus needed some extra attention e.g. materials and effects inside Unreal. This could have been avoided by buying another artist's readymade materials and effects. The work was finished and appeared pretty enough for the portfolio linked in ArtStation.

Sisällys

1	Johdanto	1
2	Millainen on hyvä portfolio?	2
3	Työn suunnittelu.....	4
4	Ohjelmistoista.....	8
5	Mallintamisesta	9
6	Teksturoidi.....	12
7	Pelimoottoriin siirtyminen.....	15
	7.1 Skenen kasaaminen.....	15
	7.2 Materiaalit	17
	7.3 Efektit	20
	7.4 Animaatiot.....	23
	7.5 Valaistus ja kamera.....	26
8	Lopetus	28
	Lähteet	29

Termiluettelo

ArtStation: Sivusto on suunnattu visuaalisten alojen taiteilijoille. Sisältää portfolioita, työpaikkailmoituksia, digitaalista kauppamateriaalia ja opetusvideoita.

Blender: 3D-mallinnusohjelma.

Illustrator: Adoben vektorigrafiikan piirto-ohjelma.

Krita: 2D-rasterigrafiikan piirto-ohjelma.

Polygoni: Vähintään kolmen verteksin ja tahkon muodostama monikulmio. (engl. Polygon)

Renderöinti: Prosessi, jossa skenestä tai mallista muodostetaan kuva taikka video.

Särmä: Kahden verteksin väliin muodostuva reuna. (engl. Edge)

Skene: Rakennettu 3D-ympäristö/näkymä. (engl. Scene)

Tahko: Kolmen tai useamman särmän väliin muodostuva alue. Täyttää tyhjää tilaa särmien välissä ja tekee mallista näkyvän. (engl. Face)

Topologia: Viittaa 3D-mallin rakenteeseen.

Teksturointi: Tekstuurin lisääminen 3D-mallin pintaan.

UV-kartoitus: 3D-mallin leikkaaminen 2D-muotoon teksturointia varten. (engl. UV unwrapping)

Unreal Engine (UE): Pelimoottori.

Verteksi: Pienin osa, josta 3D-objekti koostuu. Esitetään pisteenä. (engl. Vertex)

1 Johdanto

Tämän opinnäytteen päämääräksi laitoin tarpeeksi korkeatasoisen työn toteutuksen, jotta sen voisinkin laittaa portfolioon pelialan työnhakua varten. Tavoitteeseen pääsin hitaasta alusta ja kömpelöstä keskivaiheesta huolimatta. Käytännön osuus opinnäytteestä käsittää itse digitaalisen teoksen toteutuksen ja teoriaosuus tämän kirjoituksen, jonka tarkoitus on auttaa minua tekijänä refleктоimaan mennyttä työprosessia kuin myös avaamaan 3D-mallintamista siitä kiinnostuneille. Pyrin kertomaan työprosessin kulusta mahdollisimman selkeästi ja lineaarisessa aikajärjestyksessä.

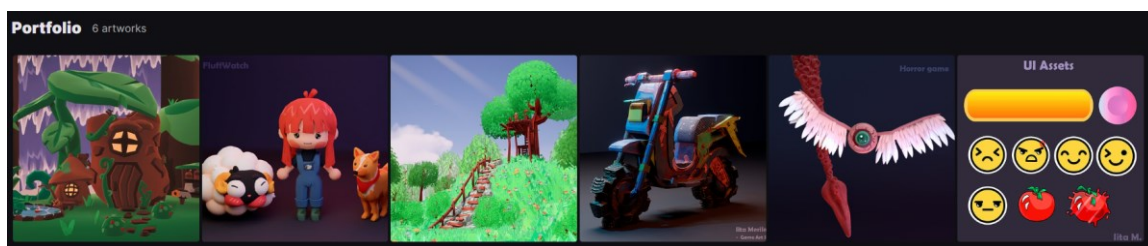
Ikäväkseni huomasin nopeastikin kirjoitustyön alkaessa, kuinka hankala 3D-mallintamisen sanasto on kääntää suomen kielelle. Koulutukseni on nojannut vahvasti englanninkieliseen termistöön ja käytettävät ohjelmistotkin ovat englanniksi. Hyödynsin termejä kääntäessä Janne Joensuun opinnäytetyötä ”3D-alan sanasto” [1]. Kiitos hänelle sen tekemisestä, kuin myös muille lähteissä mainituille, joiden työt auttoivat omani toteutuksessa.

Tässä kirjoituksessa käyn läpi suurimmilta osin vain tälle projektille olennaisia työskentelymenetelmiä. Esitettyjä käytänteitä ei tule pitää ainoana oikeana tapana toimia, vaan koko työssä käytiin läpi useitakin iteraatioita ja opin runsaasti uutta työn edetessä ja sovelsin tarpeen niin vaatiessa.

2 Millainen on hyvä portfolio?

Pyrin hakemaan pelialalta töitä nimikkeellä 3D-generalisti. Portfolion tehtävä on viestiä potentiaalisille työnantajille työnhakijan teknisestä osaamisesta, siitä mitä hän haluaa tehdä kyseisellä alalla ja jossain määrin myös luovuudesta. Sanontakin kuuluu: ”Yksi kuva kertoo enemmän, kuin tuhat sanaa”. Portfolio itsessään koostuu useista töistä, joista jokaisen olisi hyvä tukea toistaan. Hahmo artisti J. Hill käsittelee (Youtube) videossaan ”How to build a Portfolio as a Game Artist” pilareita ja tiiliä, joista portfolio koostuu. Pilarit ovat isoja ja näyttäviä töitä, jotka myös vievät paljon aikaa tehdä. Tiilet taas ovat pienempiä ja nopeampia projekteja, jotka näyttävät osaamista esimerkiksi vain yhdestä tietystä osa-alueesta. [2.]

Omassa portfolioissani (kuvassa 1) voi nähdä tällä hetkellä olevan kahdesta kolmeen pilaria ja viimeisin näistä syntyi tässä opinnäytetyössä. Jatkossa haluan sisällyttää enemmän pienempiä töitä eli näitä ”tiiliä”, joita myös voisi kuvailla ”sankarimalleina” (eng. Hero prop). Nämä työt osoittavat suurinta mahdollista osaamista artistilta ja usein keskittyvät vain yhteen monimutkaisempaan malliin. Portfolioissani näitä on tällä hetkellä vain yksi (skootteri).



Kuva 1. Kuvakaappaus oman portfolioni etusivusta.

Yhdessä portfolion työssä ei voi tuoda kaikkea osaamista esille, tämä veisi aikaa tehdä ja teoksesta voi tulla helposti liian sekava. Eikä tiettyä projektia aikoinaan tehdessä kaikkea välttämättä osannutkaan. Uusi projekti kuukausien päästä voi demonstroida taitoja, jotka edellisessä olivat vielä vajavaisia.

Portfolioon ei kannata laittaa töitä, joita ei ole vienyt loppuun asti, taikka niiden taso ei omasta mielestä yllä siihen, mihin pystyy. Hill myös painottaa videossaan, ettei portfolio ole paikka kaikille artistin töille [2]. Lyhyt ja ytimekäs on tässä tapauksessa hyvä asia, varsinkin jos määrän kautta töiden laatu ja oleellisuus alkaa kärsiä. Oleellisuudella tarkoitan esimerkiksi, että jos hakee alalle hahmoartistiksi, ei kannata sisällyttää enemmän kuvia ympäristötaiteesta kuin hahmoista. Hill mainitsee videollaan, miten on törmännyt uransa aikana ihmisiin, jotka yrittävät ahtaa itseään

muottiin, mihin eivät sovi, koska olettavat heidän täytyvän pystyä täysin venymään firmojen sen hetkisiin vaatimuksiin. Hillin itsensä mielestä peliartistin pitäisi pyrkiä tekemään sitä, mistä pitää, koska muuten ei näe järkeä artistiksi ryhtymisessä.

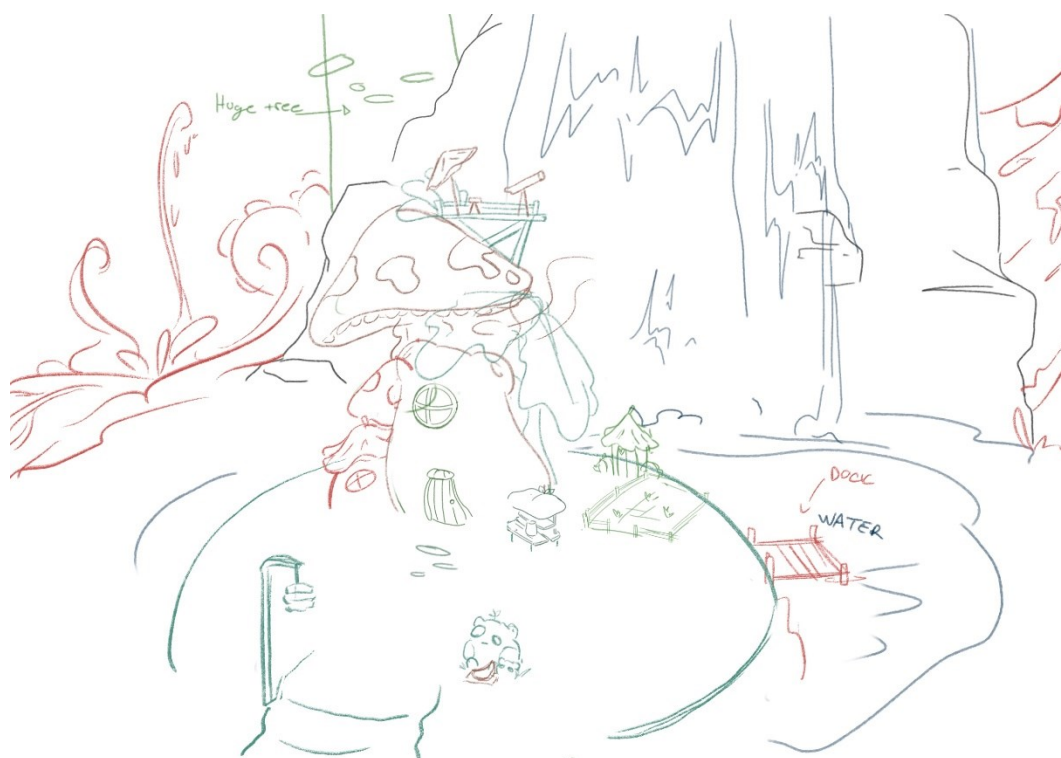
Itse olen aloittanut 3D-mallinnuksen vasta koulun alkaessa, eli noin reilu kolme vuotta sitten. En ole vielä varma, mitä haluan alalla tarkalleen tehdä ja tietoinen, ettei taitotasoni (vähäisen kokemuksen myötä) ole vielä sitä huippuluokkaa. Tämän vuoksi markkinoin itseäni 3D-generalistina, joka viittaa siihen, että olen valmis tekemään laajalti mitä tahansa 3D-mallintamiseen liittyvää. Osaan kylläkin myös piirtää digitaalisesti 2D:tä, joten ehkä termin pitäisi olla pelkästään ”peliartisti/graafikko”. Hillin mukaan artistin olisi kuitenkin hyvä täsmentää mahdollisimman tarkasti, mitä haluaa tehdä [2]. Itse en ole tästä aivan varma, varsinkin mitä tulee vasta-alkajiin. Pienissä studioissa pääsee (taikka joutuu) usein kuitenkin tekemään laajalti kaikkia erilaisia artistin työtehtäviä. Toki isommat pelistudiot, joilla on enemmän resursseja, yleensä hakevat erikoistuneimpia osaajia. Esimerkiksi konseptitaidetta voi tehdä tietty henkilö ja sitten sen varsinaisen 3D-mallintamisen hoitaa kokonaan toinen artisti.

Mihin alustalle portfolio olisi sitten hyvä tehdä? Ennen internetiä portfolion kuvat olivat paperille printattuna siistissä kansiossa. Nykyään kuitenkin portfolioita säilötään digitaalisessa muodossa monissa eri paikoissa internetissä. ArtStation on ehkä eniten käytetyin sivusto, jossa myös omani sijaitsee. Pidän siitä, sillä se on valmis helppokäyttöinen nettisivusto, jolla on jo suuri käyttäjäkunta. Uudet ihmiset voivat löytää artistin ja hänen työnsä esimerkiksi etusivun ehdotuksien joukosta, jos jokin työ alkaa yllättäen nostamaan suosiota. Vaihtoehtoina on myös luoda täysin oma nettisivu, taikka kaikessa yksinkertaisuudessa voi lisätä töistään kuvia tiedostoon (esim. Word, PowerPoint), jonka lähettää s-postiin hakukirjeen mukana työnantajille. Mielestäni jos portfolioa säilöö jollain nettisivulla, kannattaa siitä silti pitää omat kopiot esim. koneen ulkoisella kovalevyllä, sillä nettisivut voivat mennä alas taikka tili voi joutua väärin käsiin.

3 Työn suunnittelu

Jos työn haluaa saattaa portfolioon asti, se ei voi jäädä keskeneräiseksi, taikka sitä ei voi tehdä puhtaasti harjoitusmielessä. Oma tyylini 3D-artistina nojaa simppeleihin mobiilipelityyliin. Mobiililla peleissä on enemmän tehorojoituksia kuin tietokoneella, joten mallit ovat usein mahdollisimman vähäpolygonisia (yksinkertainen geometria). Tämä on itselle helpoin ja nopein tyyli toteuttaa, joten valitsin sen.

Keräsin nettisivu Pinterestistä ideoita ja lisäilin niitä mood boardiin. Mood board sanan voisi kääntää ”tunnelmatauluna”. Siihen lisätään kuvia, jotka inspiroivat väreiltään, ideoiltaan ja muotokieltään teosta, jota ollaan sen pohjalta toteuttamassa. Tein myös nopean luonnoksen, miltä skene (tässä työssä skene-sanalla viitataan mallintamalla rakennettuun ympäristöön) tulisi näyttämään (kuvassa 2).



Kuva 2. Skene-luonnos.

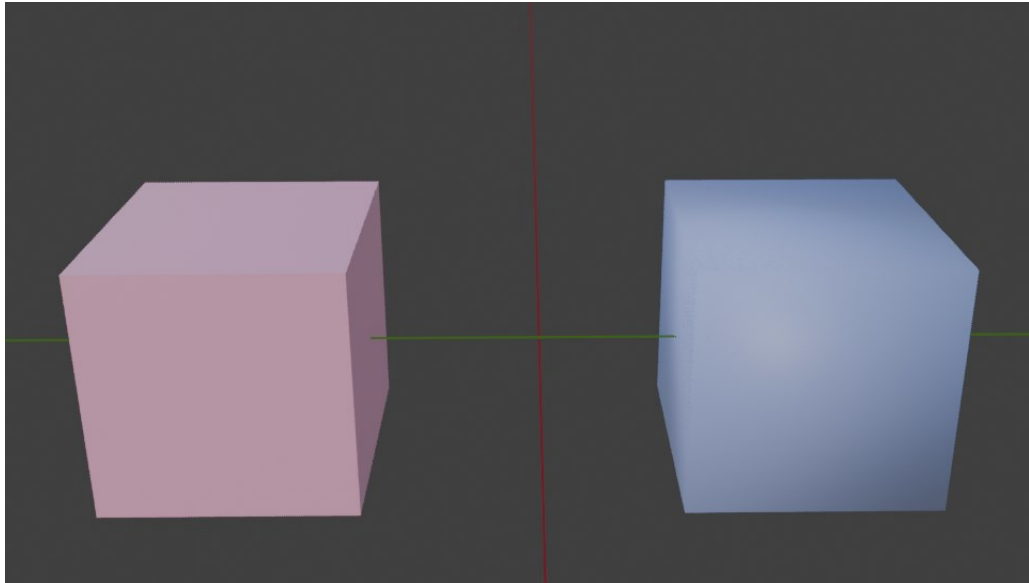
Työn alustamiseen kannattaa panostaa, jotta sen toteutus olisi mahdollisimman vaivatonta. Jos esimerkiksi tuntuu jäävänsä jumiin, olisi mahdollista saada vastauksia kysymyksiinsä alkuun hyvin koostetusta tunnelmataulusta taikka luonnoksesta. Tässä projektissa itse kiirehdin alkuvaiheen

liian hätäiseen ja jouduin mm. vaihtamaan hieman erilaiseen tyyliin projektin puolivälissä huomauttessani, etten saanut alkuperäistä ideaa vahvemmin tyylitelystä ja vähempipolygonisesta tyylistä toimimaan (esimerkki tästä kuvassa 3). Totesin sen olevan liian uutta itselle, enkä voinut uhrata opetteluun tarvittavaa aikaa.

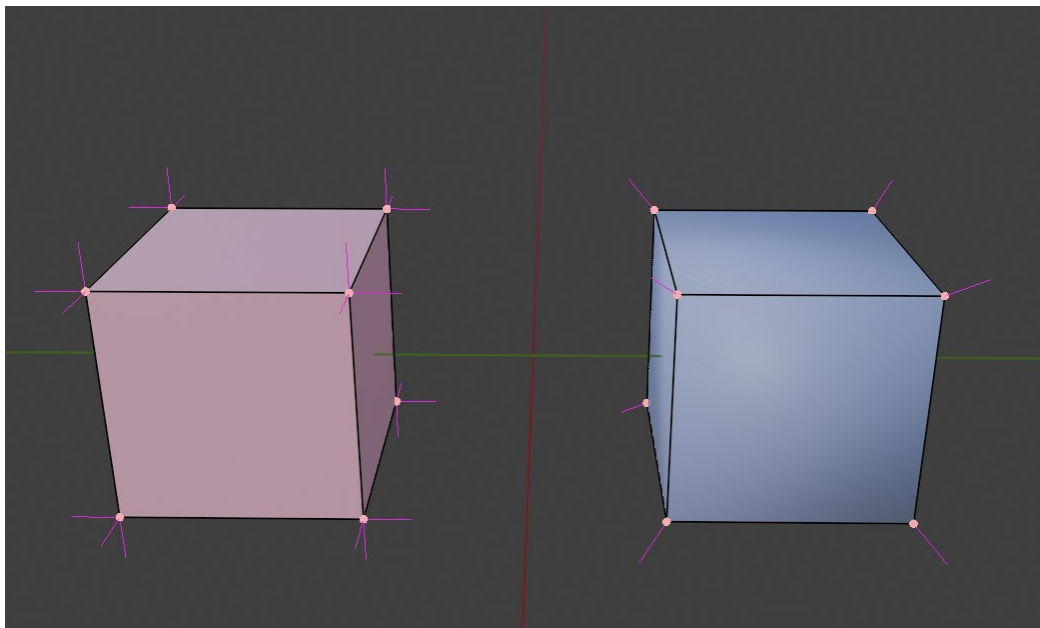


Kuva 3. Todella vähäpolygonisen tyylin esimerkki, jolla yritin aluksi skeneä toteuttaa. [3.]

Jotta tämän hyvin vähäpolygonisen tyylin olisi saavuttanut, olisi kannattanut käyttää niin sanottuja ”hard edgejä” eli vahvistettuja särmiä mallin reuna-alueilla. Myös eri tahkoja olisi saanut korostettua esimerkiksi lisäämällä pieniä värieroja. Mallinnusohjelma Blenderissä on myös mahdollisuus käyttää ”shade smooth”, eli varjosta sileäksi -asetusta. Se vaikuttaa pinnan varjostuksien laskentatapaan muuttaen normaalien suuntaa ja antaa näin illuusion sileästä pinnasta [4]. Tämä on nähtävillä kuvan 4 sinisessä kuutiossa. Normaali on vektori, joka on kohtisuorassa mallin pintaan nähden, ja jokaisella tahkolla on omansa (kuvassa 5 olevat lilat lyhyet viivat) [5].



Kuva 4. Varjostus sileäksi- ja matakksi-asetusten vaikutus mallissa.



Kuva 5. Varjostusasetuksien vaikutus normaalien suuntaan nähdään tässä. Kuva esittää vertek-
sien normaalien suunnat aina yhtä tahkoa kohti.

Tyyli, johon vaihdoin ja millä lopulta vein projektin loppuun, oli hieman korkeapolygonisempi ja sileämpi silmälle (kuva 6) kuin alkuperäinen idea edellä mainitusta tyylistä. Käytin varjosta sileäksi-asetusta ja lisäksi tiettyihin kohtiin varjostuksen siistimiseksi vahvistettuja reunoja (nämä ei näkyvillä kuvassa 6). Muutokseen vaikutti myös epäilemättä tämän opinnäytetyön kanssa päällekkäin menevä työharjoitteluni firmassa, jossa tein töitä juuri tällä pehmeämmällä tyylillä.



Kuva 6. Lopullisten mallien topologia esitettynä Blenderin rautalankatilassa (engl. Wire frame mode).

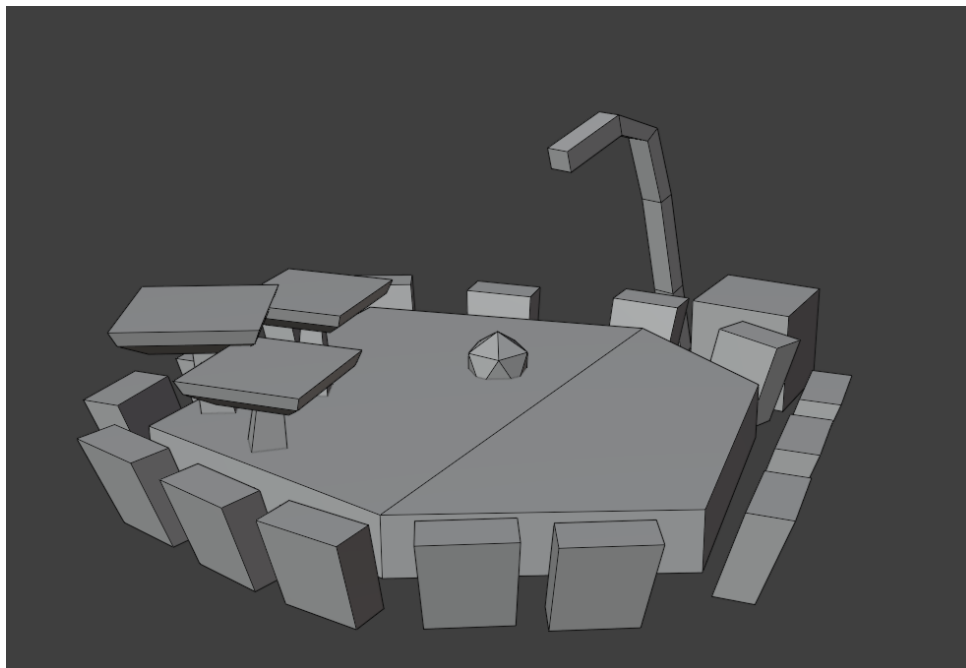
4 Ohjelmistoista

Opinnoissani olen käyttänyt pääosin Blenderiä mallintamiseen, joten päätin tähänkin projektiin ottaa sen. Kyseinen ohjelmisto on varsinkin monien indie-pelistudioiden suosima. Se sisältää runsaasti erilaisia ominaisuuksia, siihen löytyy jo paljon eri käyttäjien tekemiä lisäosia ja se on ennen kaikkea ilmainen. Kellä tahansa on oikeus käyttää, kopioida, muuttaa, jakaa lähdekoodia ja itse ohjelmaa. Isommat studiot suosivat kuitenkin herkästi muita ohjelmistoja, jotka ovat yleensä olleet käytössä pidempään, ovat erikoistuneempia ja joilla on hyvä virallinen asiakaspalvelukanava mahdollisten ongelmien ilmetessä. [6.] Blender ei myöskään kaikkien mielestä vielääkään sisällä niitä parhaimpia mahdollisia ominaisuuksia, joita joistain muista tiettyyn osa-alueeseen erikoistuneista ohjelmistoista saattaa löytyä.

Tässä projektissa kuitenkin käytin Blenderiä pääosin vain laatikkomallintamiseen ja animointiin. Koko skenen puolestaan (valaistuksineen ja efekteineen) kasasin pelimoottorin Unreal 5:n puolella.

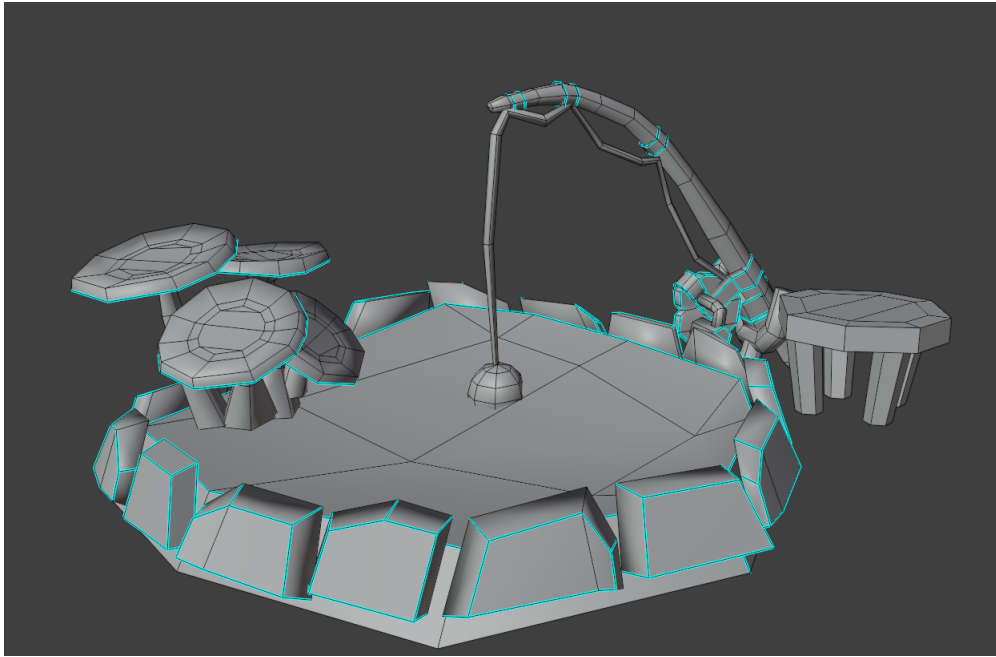
5 Mallintamisesta

Mallintamisen toteutin ”box modeling” eli laatikkomallintamistekniikalla. Se alkaa koostamalla muotoja primitiivikappaleista (usein kuutioita), esimerkki tästä kuvassa 7. Mallista saadaan aina tarkempi näitä kappaleita lisäämällä ja muotoilemalla mm. Blenderin perusominaisuuksilla: extrude, inture ja bevel.



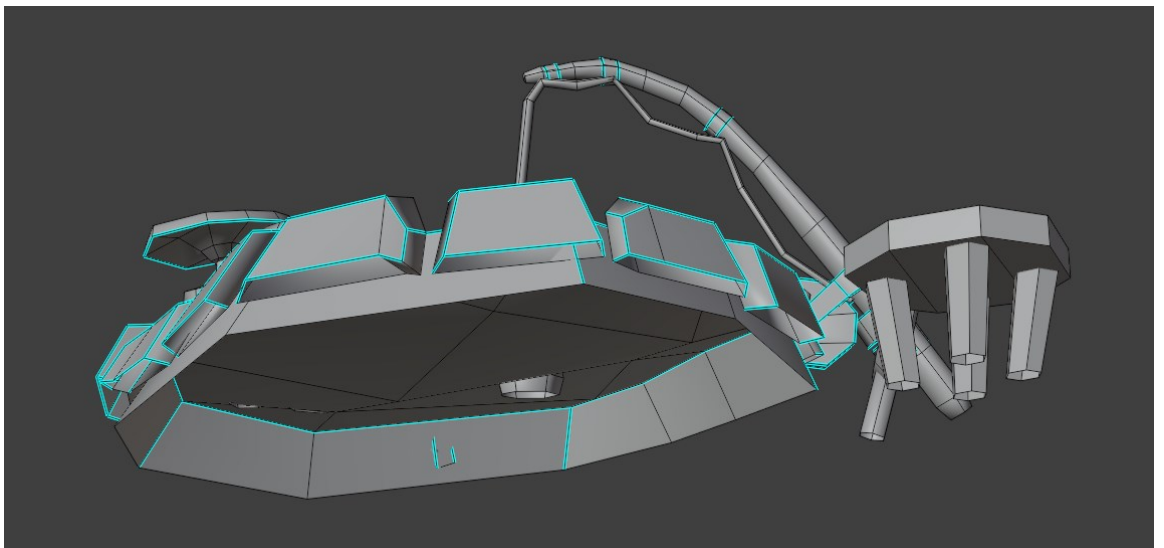
Kuva 7. Laatikkomallinnustekniikan ensimmäinen vaihe.

Siniset reunat kuvassa 8 ovat luvussa kolme mainittuja vahvistettuja särmiä (/reunoja). Nämä särmit auttavat mallin varjostuksen hallitsemisessa. Kuvasta voidaan nähdä niiden olleen tarpeellisia etenkin lammen reunakivissä. Näissä voi myös huomata varjojen olevan erityisen vahvoja kohdissa, missä vahvistettuja särmiä ei ole (taikka geometria on huolimattomasti toteutettu).



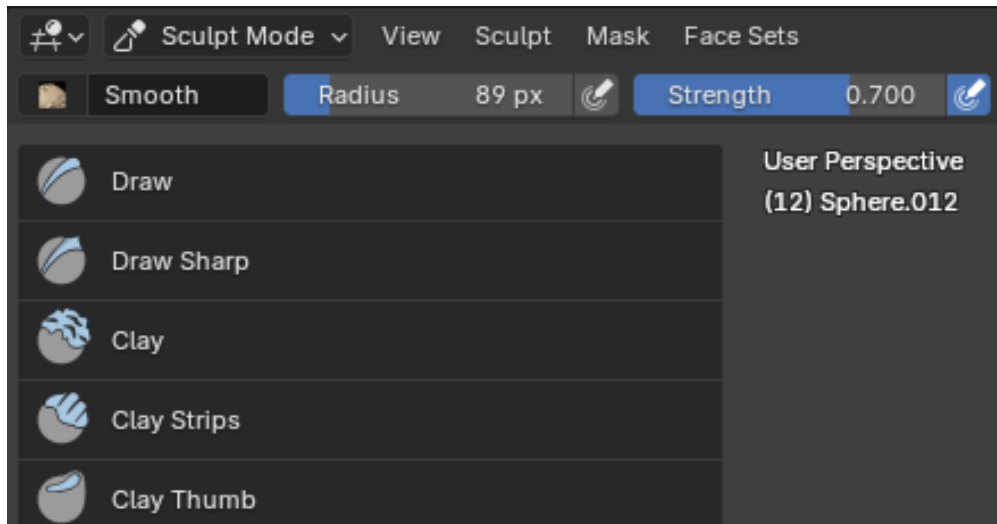
Kuva 8. Laatikkomallinnustekniikan lopputulos.

Mallit voidaan myös jättää avoimeksi pohjasta (kuva 9), jos pohja ei tule näkymään lopullisessa tuotteessa. Geometriaa säästyy ja tämä saattaa myös auttaa varjostusongelmissa. Olisi myös hyvä, ainakin loppuvaiheessa, viedä mallia pelimoottorin puolelle tarkistettavaksi, jossa lopullinen valaistus on. Siellä valaistus ja näin ollen mallin varjot saattavat muuttua paljonkin verrattuna Blenderin sisällä olevaan ympäristöön.



Kuva 9. Malli alhaaltapäin katsottuna, avoimet pohjat.

Joihinkin peleihin, varsinkin hahmoja tehtäessä, mallinnetaan sculpting-tekniikalla eli digitaalisella veistämisellä/muovailulla. Tapa on hyvä orgaanisten mallien tekoon ja antaa artistille vahvan kontrollin mallin pinnasta hyödyntäen siihen käytetyn ohjelmiston erilaisia muovailutyökaluja (kuvassa 10) ja näiden asetuksia. [7.]



Kuva 10. Joitain Blenderin muovailutyökaluja.

6 Teksturointi

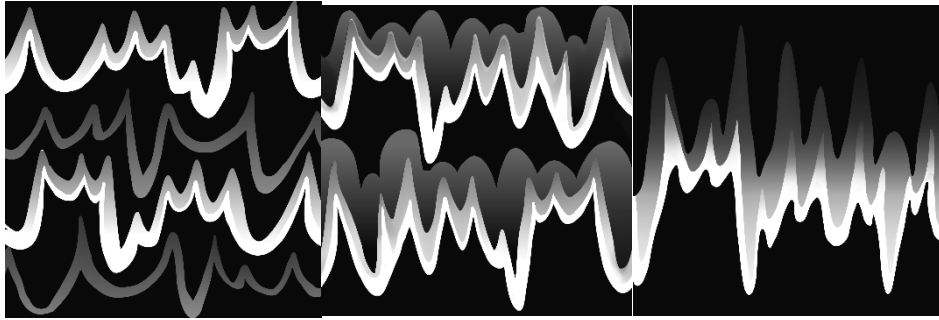
Alkusi pohdin käyttäväni Adoben Substance 3D Painteria teksturoinnissa, mutta kuitenkin tajusin työn tyylin olevan niin yksinkertainen, ettei niinkään monimutkaiselle prosessille olisi tässä tarvetta. Substance on 3D-mallien teksturointiohjelmisto. Se sisältää mm. monenlaisia valmiita älykäitä materiaaleja erilaisten tekstuurien toteutusta varten esim. nahka ja metalli, jotka vaativat perusvärin lisäksi kiiltoa ja syvyyseroja. Omat mallini käyttivät vain diffusea, eli värikarttaa (kuvassa 11). Niiden pintaan ei ollut tarvetta tehdä syvyys-taikka kiiltoeroja normaali- ja metallisuus-karttojen avulla. Näitä yksityiskohtia pystyi säätämään tarvittaessa malleille vielä Unrealin puolella materiaalien kautta. Materiaali sisältää tekstuurin (tai useampia) ja tämän asetuksia ja se liitetään malliin.

Mitä vähemmän tyhjää tilaa tekstuuriatlakseen jää, sitä parempi, tiivis kartta/atlas on hyvin optimoitu. Tekstuurit, joiden pitää olla tarkkoja ja jotka näkyvät selkeästi lopullisessa mallissa on hyvä laittaa karttaan isoimpina. Tekstuuriatlas (kuvassa 11) siis on yksi iso tiedosto/kuva, joka sisältää monia pienempiä tekstuureita [8].



Kuva 11. Itse tekemäni ja projektissa käytetty tekstuuriatlas.

Tässä projektissa mallien teksturointi hoidettiin valitsemalla väripaletti, kokoamalla liukuvärit ja piirtämällä kuviot Kritalla. Sanoisin skenen efektien tekstuurien olleen hankalimpia tehdä. Esimerkiksi vesiputouksen taustatekstuurin piirtäminen aiheutti päänvaivaa (kuvassa 12). Aluksi piirsin tekstuurit Kritalla, mutta lopulta vein ne Adoben Illustrator-vektoriohjelman muunnettavaksi. Varsinkin skenen isoissa vuorissa pikselit näkyivät selkeästi, sillä itse kartan koko oli liian pieni. Vektoreilla tehdessä kuvioita voi isontaa loputtomiin menettämättä kuvan laatua.



Kuva 12. Vesiputouksen erilaisia tekstuurikokeiluja, tehty toistuviksi.

UV-lyhenne viittaa 2D-tekstuurin akseliin. UV-kartta tallettaa verteksin sijainnit 2D-tekstuurissa. [9.] UV-saari on yhdistetty ryhmä polygoneja UV-kartassa, se vastaa mallissa aina tiettyä aluetta, johon tekstuuri heijastuu.

Jotta minkään tekstuurin pystyy saamaan mallin pintaan nätisti, pitää sille suorittaa avaaminen (engl. Unwrapping). Eli malliin merkitään saumakohtat ja se ikään kuin avataan 2D-pinnalle. Tässä projektissa suurin osa malleista käytti vain yhtä väriä per mallin osa. Saumoja ei tarvinnut käyttää, sillä kaikki verteksit yhdistettiin kasaan siinä UV-saarella, joka haluttiin tietyllä värillä (kuvassa 13 lilat pisteet). Kalan suomissa kuitenkin piti hyödyntää näitä aukaistuja UV-saaria (kuvan 13 tekstuuriatlaksen alaosassa), koska niissä piti näkyä koko kuvio eikä vain yksi väri riittänyt.



Kuva 13. Kala-malli ja sen UV-karttanäkymä.

7 Pelimoottoriin siirtyminen

Renderöinti on prosessi, jossa luodaan valokuvan realistisuuteen verrattavissa oleva kuva taikka video 3D-malleista. Tämä on viimeinen vaihe 3D-tuotannossa. [10.] Skenen olisi voinut hyvin kasata ja renderöidä pelkästään Blenderin puolella ja moni artisti tekeekin sillä hienoja ja monimutkaisia töitä aina alusta loppuun asti. Itse ajattelin kuitenkin pelialan opiskelijana olevan arvokasta liittää tähänkin prosessiin mukaan pelimoottoriin mallien siirto. Sinne artistin yleensä pitää siirtää oikeassakin peliprojektissa työnsä. Pelimoottoreista olen koulussa käyttänyt vain Unityä ja Unrealia, joista jälkimmäisessä koen olevan parempi visuaalinen käyttöliittymä artistille. Käyttämäni Unreal-versio oli projektin tekohetkellä uusin mahdollinen, UE5.

Mallit toin Blenderistä Unrealiin fbx-tiedostomuodossa, jos tarpeellista oli niissä mukana animaatio- ja luurakodata. Suurin osa malleista ei näitä kuitenkaan tarvinnut. Kyseiset ohjelmat luonnollisesti toimivat toisistaan poiketen, joten fbx:n asetuksia piti muokata. Ohjelmat käyttävät eri mitasuhteysiköitä. Blender toimii metreissä ja Unreal senttimetreissä. Geometria-asetusten kohdalla oli hyvä valita "sileys"-ala-asetuksen kohdalta tahko (/face), jollei halunnut Unrealin antavan varoitusilmoitusta "No smoothing group information was found", joka kerta kun fbx:n tuo sinne. Myös koordinaatistoakselien suunnat voivat vaihdella ohjelmistojen välillä, mutta niitäkin pystyy mallin fbx:n asetuksista muokkaamaan.

7.1 Skenen kasaaminen

Mallien saamisen jälkeen pelimoottoriin oikeassa muodossa, pystyi skenen kasaaminen alkamaan. Toin eriteltynä melkein jokaisen mallin, jonka sijaintia halusin muokata tarkemmin pelimoottorin puolella. Esimerkiksi farmin kohdalla itse aitaus on eriteltynä kottikärrystä, kun taas lammen kohdalla kaikki mallit jakkarasta onkeen ovat samassa paketissa (kuvissa 14–15).



Kuva 14. Farmi ja kottikärrymalli eriteltynä eri objekteiksi.



Kuva 15. Lampi ja kaikki sen osat yhdistettynä samaksi objektiksi.

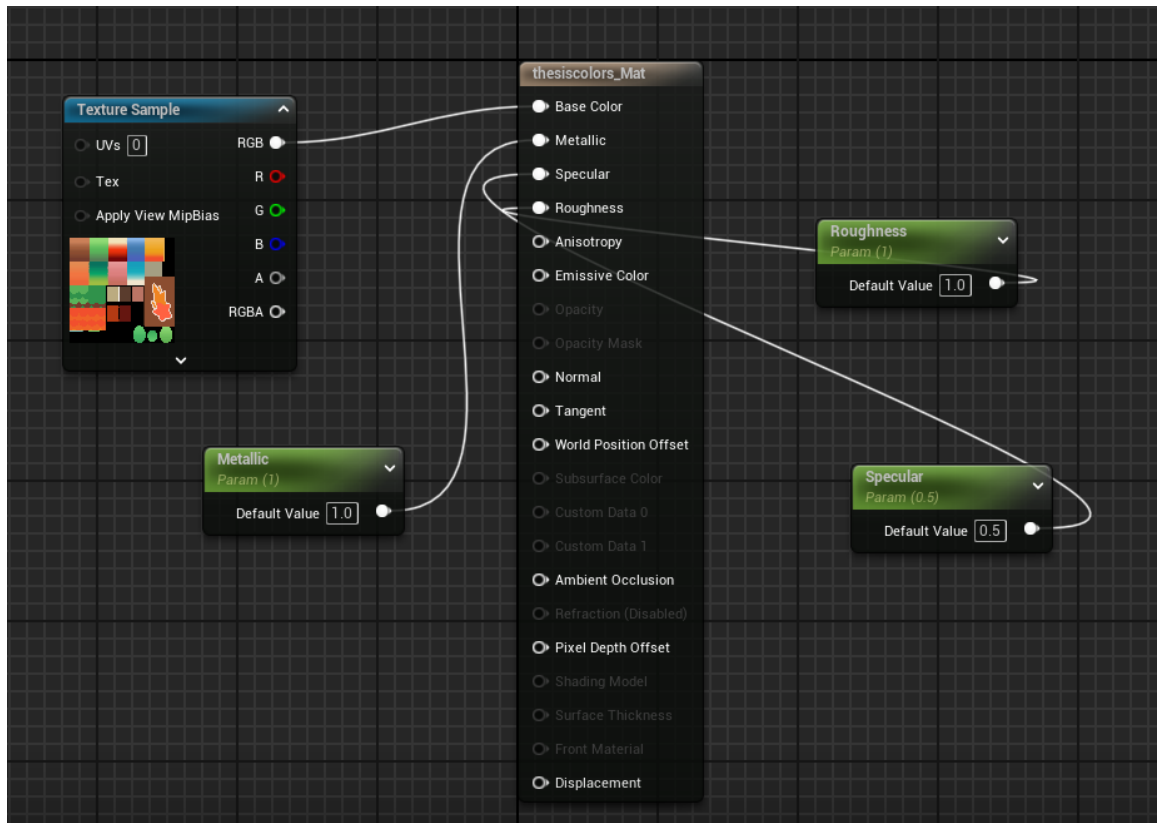
Skeneä (kuvassa 16) kasatessa pidin mielessä komposition säännöt. Tärkein alue kuvassa on keskellä ja vähemmän tärkeät reunoilla. Polku kuljettaa pelaajan silmää kohti kuvan tärkeintä aluetta eli kahta taloa. Tähtitorni ja laituri tukevat kuvaa sen vastakkaisilla puolilla, joten skene ei vaikuta missään kohtaa liian tyhjältä (taikka toisin sanoen tylsältä). Tärkeiden kohtien välillä on myös hieman tilaa silmälle hengähtää. Näitä yksinkertaisimpia alueita täyttävät pienemmät tukikappaleet mm. työpöytä ja farmi, joihin katse ei heti ensi silmäyksellä kiinnity.



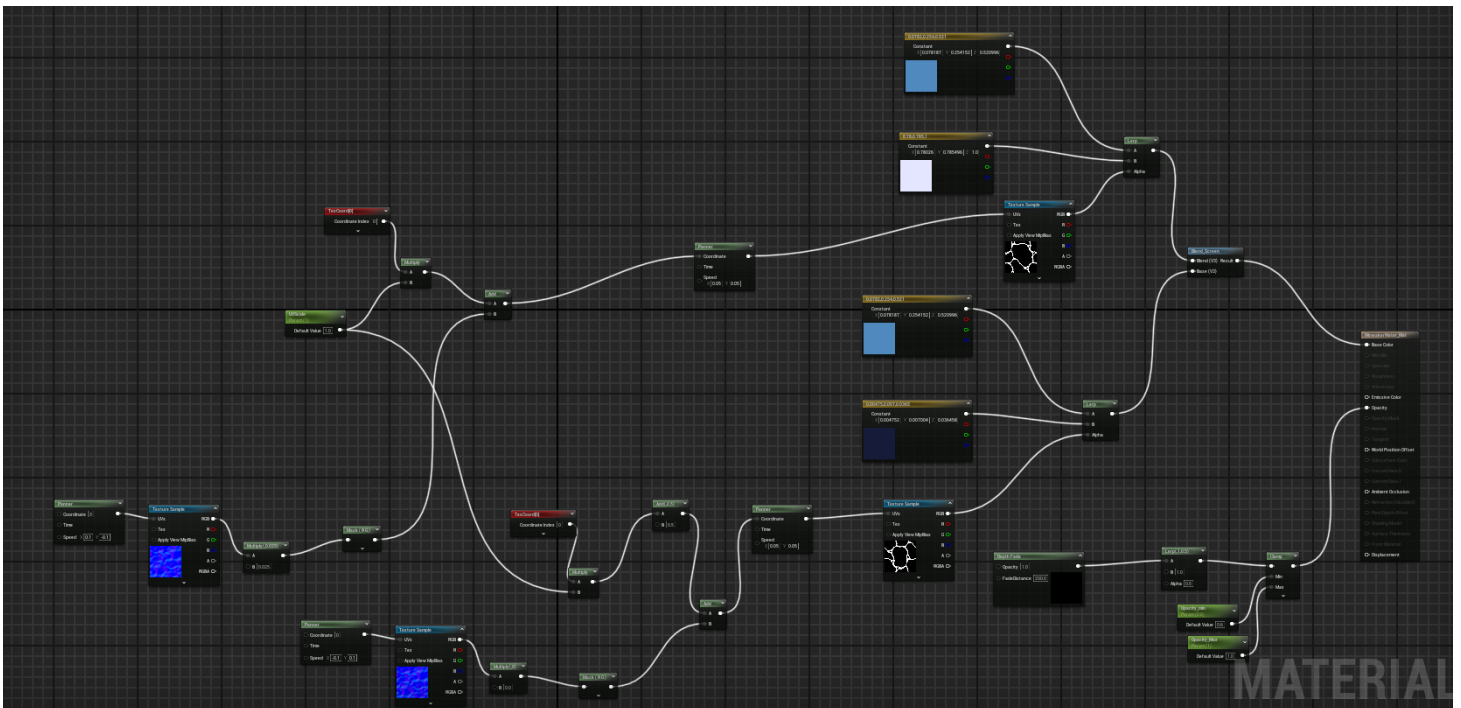
Kuva 16. Koko skene.

7.2 Materiaalit

Unrealin materiaaligraafi on mielestäni erinomainen (hyvin visuaalinen). Kuvassa 17 näkyvä perusmateriaali oli vaivaton tehdä, se vaati vain tekstuurin ja muutamien asetusten muuttelun. Dynaamiset materiaalit vesiputoukseen ja etenkin mereen olivat hankalimpia tehdä. Kuvan 18 merimateriaali vaati kaksi ”kerrosta” ja tekstuuriin piti lisätä liikkuvuutta. Materiaalien ja efektien teko kuuluu usein teknisen artistin työtehtäviin. Löysin netistä haastavimpien materiaalien ja efektien tekoon tutoriaaleja, joita seurasin mahdollisimman tarkasti.



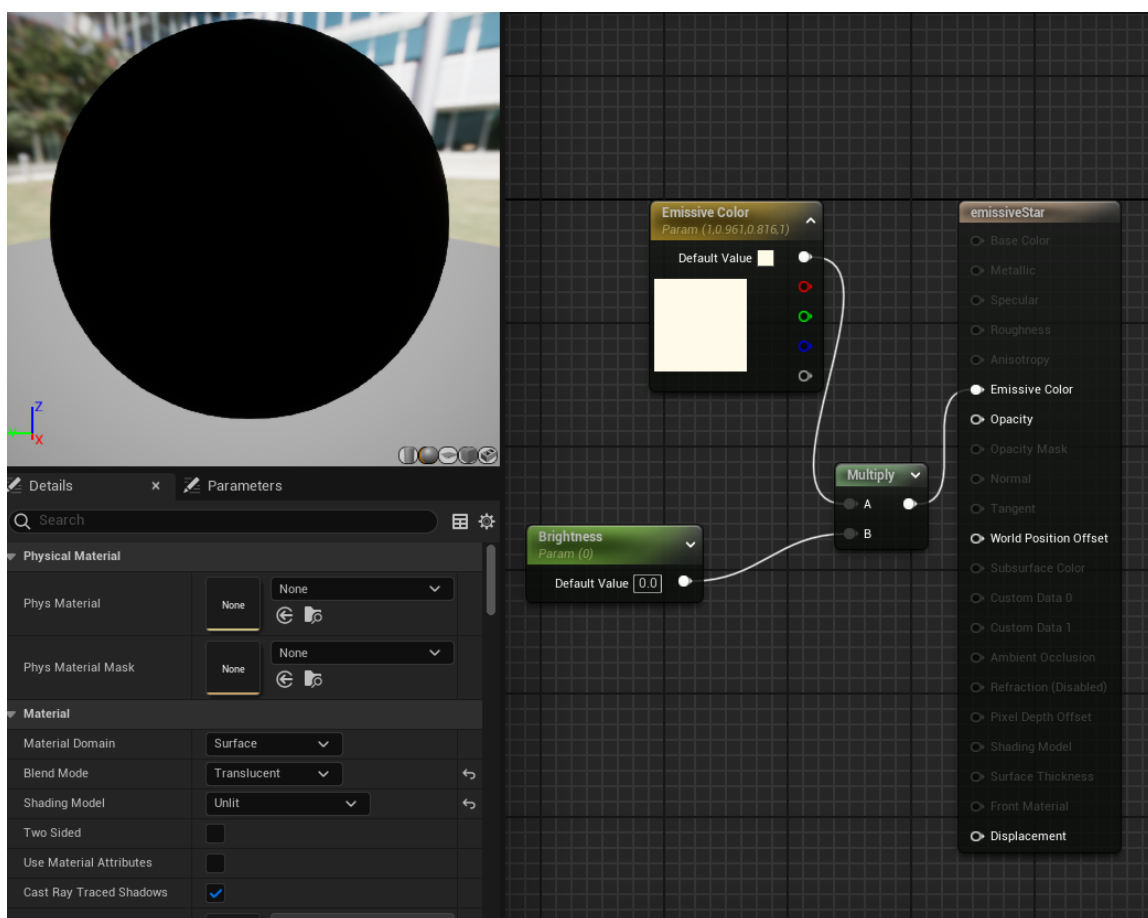
Kuva 17. Perusmateriaalin materiaaligraafi.



Kuva 18. Meren materiaaligraafi.

Videopalvelu Youtube on ollut hyvä väline koulutuksessaani itseopiskeluun ja olen kiitollinen de-veille, jotka sinne tekevät videoita pelialalta. Moni ei kuitenkaan ole opettaja, taikka aivan varmasti tiedä edes, mistä puhuu, joten kriittisyys näitä videoita katsoessa on tärkeää ja välillä saa vahvasti soveltaa, jos esimerkiksi käytössä olevaan ohjelmaan on tullut uusi päivitys taikka se mitä kaipaa ei ihan vastaa tutoriaalia. Oman skeneni pienemmän talon savupiipusta nouseva pilvi oli toteutettu tutoriaalin mukaan, joka käsitteli auton pakoputkesta tulevan savupilven tekoa.

Joidenkin mallien tiettyjen kohtien halusin hohtavan valoa, esim. talojen ikkunoiden. Näihin mal-leihin lisäsin Blenderin puolella paikan toiselle materiaalille perusmateriaalin lisäksi. Materiaali oli hohtava (emissive) ja toimi valaisematon tilassa, sen graafin näkee kuvassa 19. Skenen tähdet toimivat myös emissive-materiaalilla, eivätkä olleet pistevaloja.



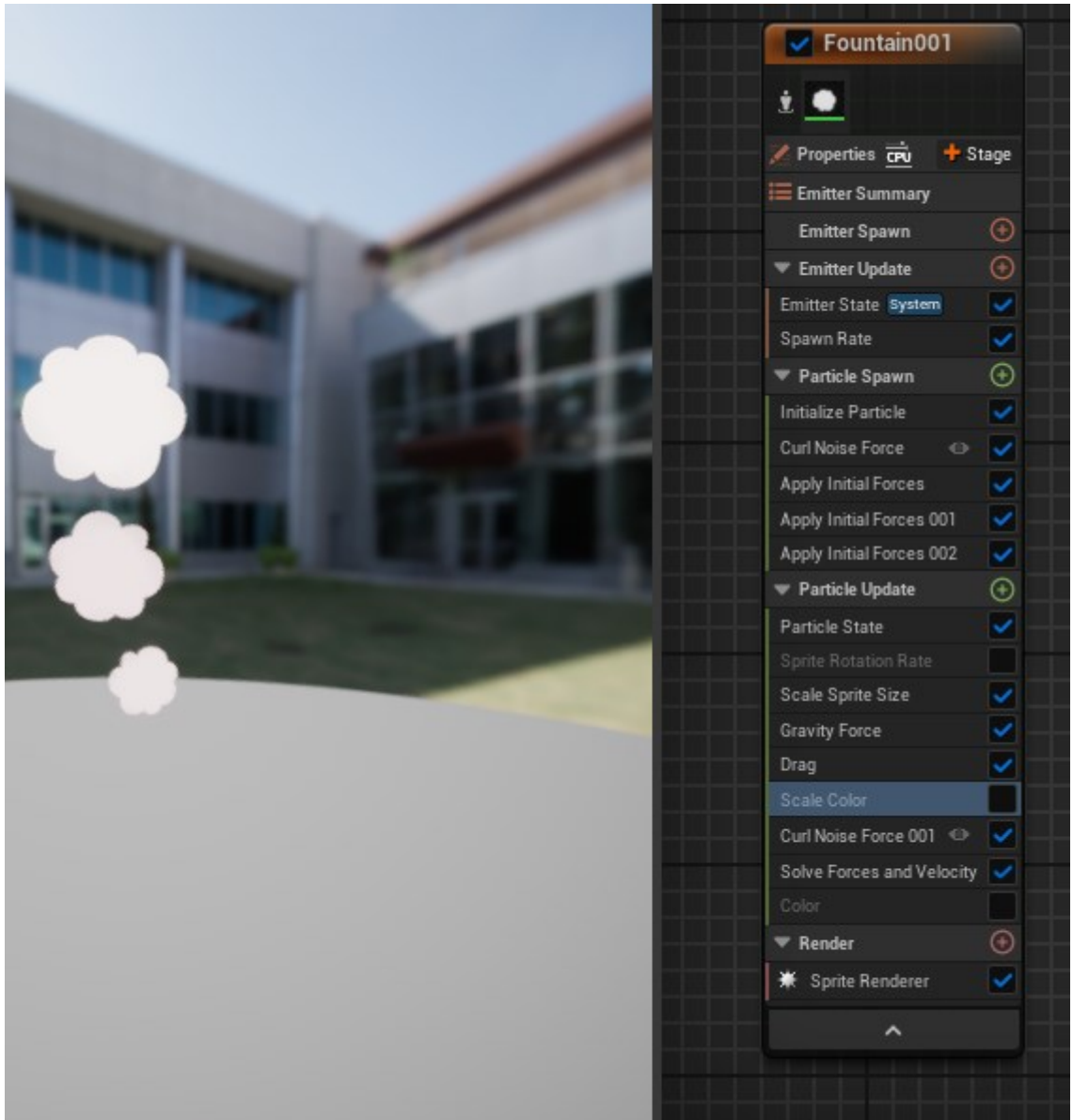
Kuva 19. Hohtava materiaali.

7.3 Efektit

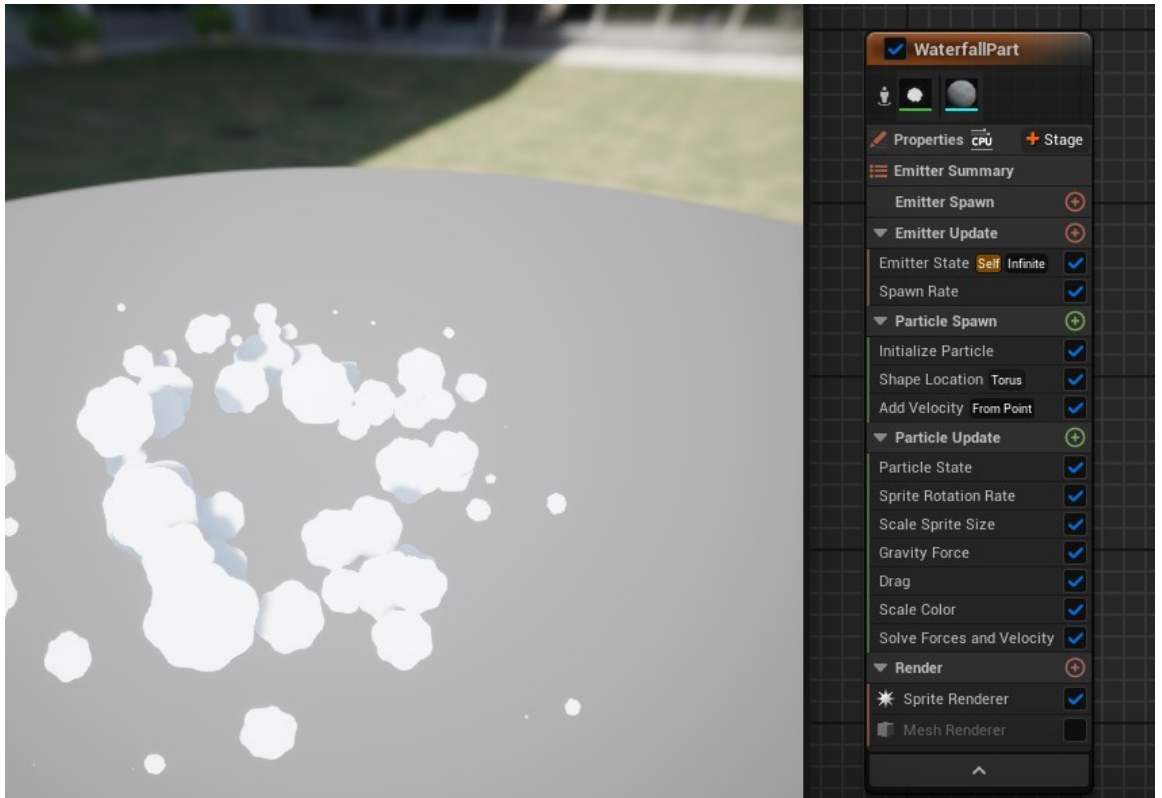
Skenessä on kolme partikkeliefektiä: vesiputouksen vaahto, savupiipusta nouseva savu ja hieman huonommin huomattavissa olevat silloin tällöin ilmestyvät tuuliviivat. Nämä tehtiin Unrealin Niagaralla. Partikkeliefektit ovat visuaalisia tehosteita, joita käytetään videopeleissä (ja elokuvissa). Niiden avulla voidaan luoda dynaamista grafiikkaa [11, s. 2].

Niagara on Unrealin VFX-työkalu (VFX on lyhenne visuaalisille efekteille), jonka avulla artistit pystyvät luomaan efektejä ilman koodarien apua. Niagara-systeemi sisältää emitterin (yhden tai useampia), jossa varsinainen efekti kootaan. Nämä emitterit ovat kuvissa 20–22 nähtävissä olevia oransseja laatikoita, jotka sisältävät erilaisia asetuksia. Niagara systeemiä luodessa voi valita, haluaako käyttää valmista pohjaa vai lähteä työskentelemään kokonaan tyhjästä pohjasta, johon itse lisää kaikki tarvittavat komponentit. Itse lähdin tässä projektissa työstämään efektejä valmiista pohjista, joita muokkasin sitten sopiviksi video tutoriaalien avulla.

Niin kuin kuvissa 20–22 näkyy, emitter sisältää paljon erilaisia asetuksia. Riippumatta efektistä kaikki tarvitsevat kuitenkin ainakin nämä peruskomponentit toimiakseen: 1) pitää määritellä miten, kuinka usein partikkelit syntyvät ja kuolevat 2) millaisen näköisiä nämä partikkelit ovat elämänsä aikana 3) miten partikkelit liikkuvat.

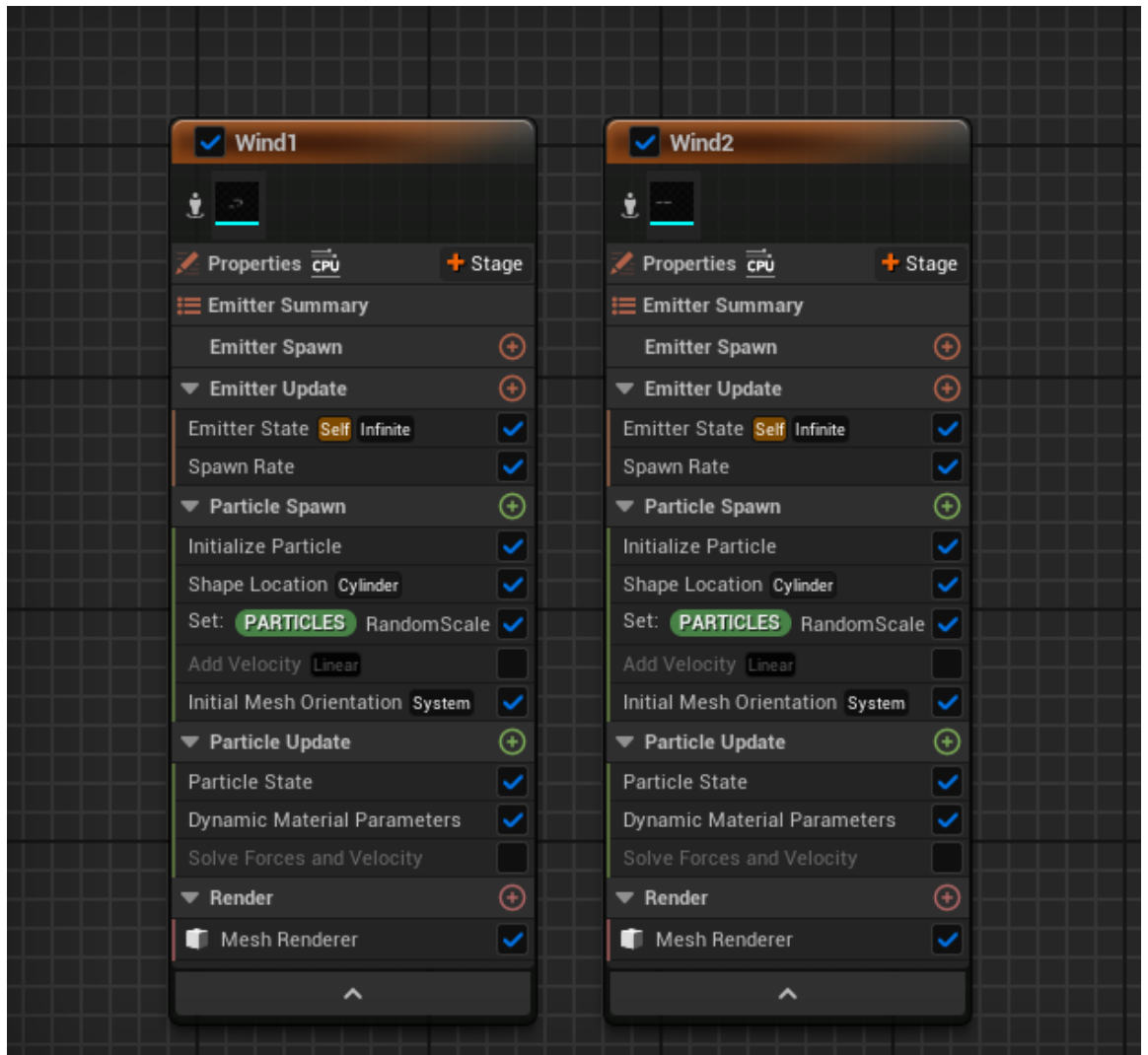


Kuva 20. Savupiipun savun efekti.



Kuva 21. Vesiputouksen efekti.

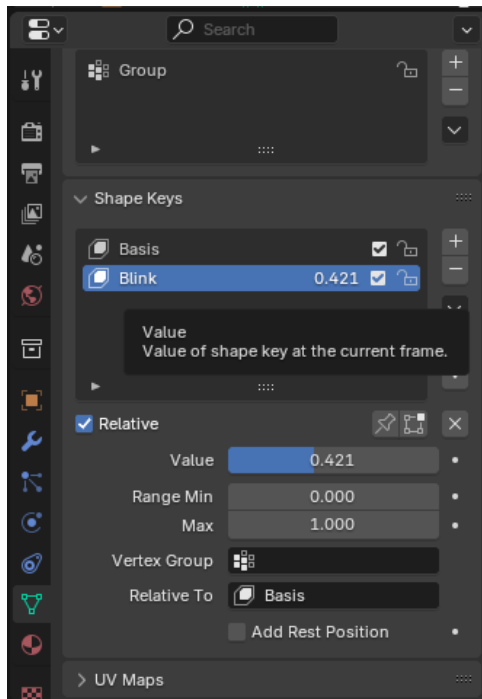
Efekteissä 20–21 käytettiin samaa 2D-pilvikuvaa, joka asetettiin emitterin render-kohtaan sprite-renderiin. Tuuliviivat taas käyttivät mesh-renderiä (kuvassa 22), sillä ne olivat 3D-malliin pohjautuvia. Mallin materiaalia liikutettiin ja häivytettiin sen edessä mallia pitkin, näin luotiin efekti katoavasta tuuliviivasta.



Kuva 22. Tuuliviivojen efekti. Sisältää kaksi emitteriä, sillä viivoja oli kahta hiukan toisistaan poikkeavaa tyyppiä.

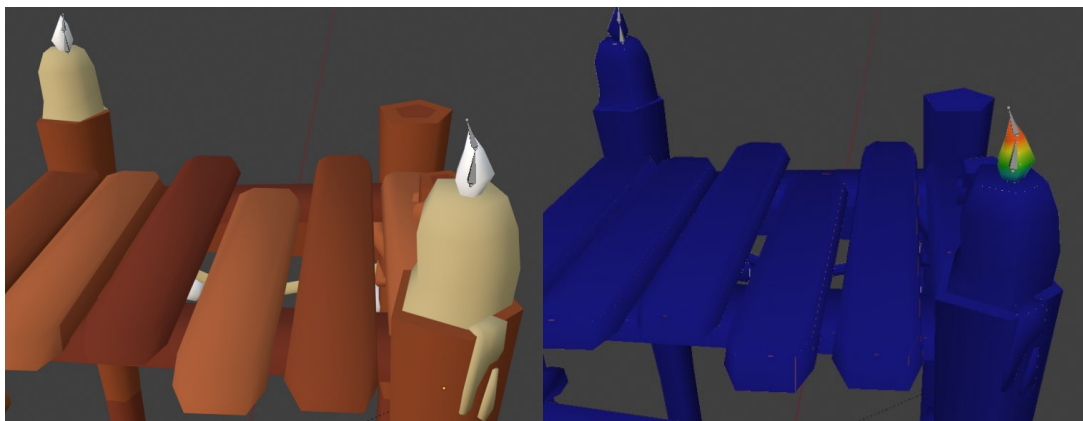
7.4 Animaatiot

Malleista kala, laituri ja puut tarvitsivat animaatiot. Käytin kahta erilaista tapaa tehdä ne. Kalan silmän räpäytys ja puiden heilunta tehtiin "shape key", eli muotoavaimilla (kuvassa 23). Tässä tekniikassa mallin geometriaa muovataan ja sen erivaiheille luodaan näitä "avaimia", mihin voidaan sitten animaatiota kootessa viitata. Muotoavaimelle muita nimityksiä on mm. "morph targets" ja "blend shapes" [12]. Tekniikka on erityisen hyödyllinen kasvojen liikkeitä animoidessa.



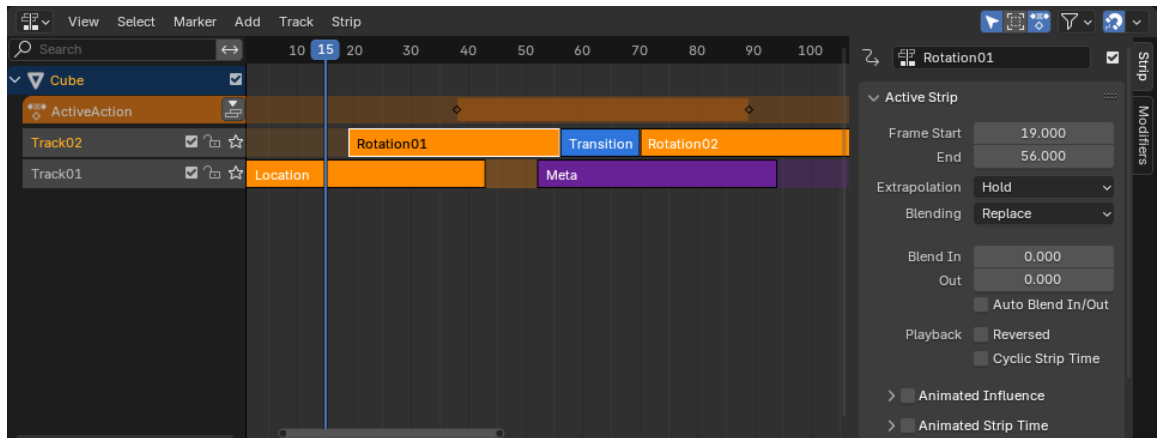
Kuva 23. Kalan muotoavaimet ja niiden asetukset Blenderissä.

Laiturin kynttilöiden liekkiin käytin yleisempää tapaa tehdä. Loin luut ja maalsin painot, jotka yhdistävät nämä luut malliin ja näin animoitujen luiden liike kontrolloi mallia (kuvassa 24). Tätä prosessia, missä objektille luodaan kontrollit animointia varten, kutsutaan ”riggaukseksi” (engl. Rigging). [13.]



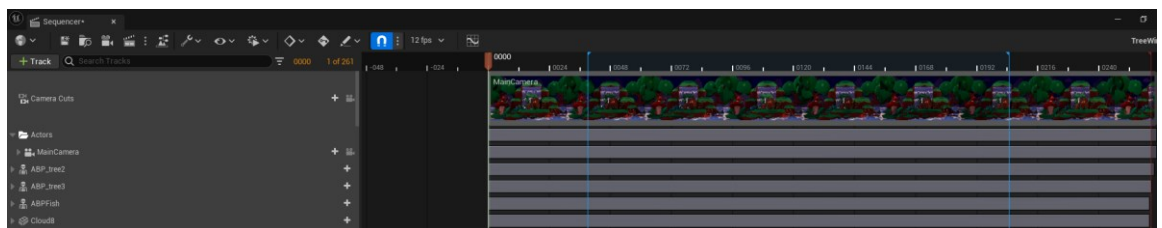
Kuva 24. Laiturin kynttilän liekkien luut ja painot.

Kun sain objektin animoitua Blenderissä, toin sen Unrealiin NLA-editorin kautta (non linear animation). NLA-editorissa pystyin tarvittaessa valitsemaan vain yhden tietyn mallilla olevan animaation kuvassa 25 trackin vieressä näkyvissä olevan tähti-ikonin avulla ja tuomaan sitten vain sen objektin mukana Unrealiin.



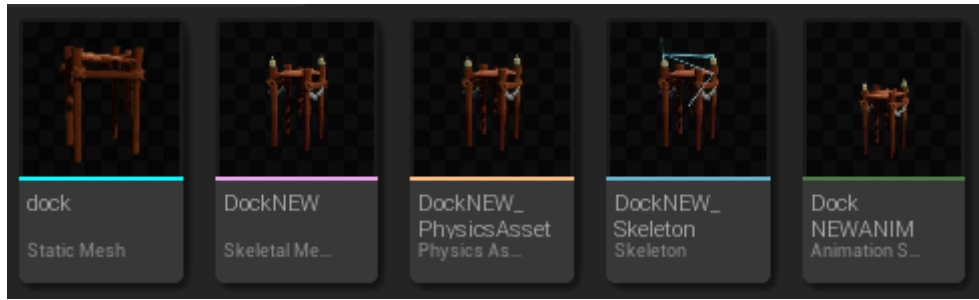
Kuva 25. NLA-editori [14].

Lisäsin vielä animaatiot Unrealin sequenceriin (kuvassa 26), jossa eri mallien animaatioiden ajoitusta suhteessa toisiinsa pystyi paremmin kontrolloimaan, kun näki koko skenen kasassa. Ikävä kyllä, jotta sai muotoavainanimaatiot toimimaan sequencerissa, piti ne ensin osata laittaa Unrealin Animaatio-blueprinttiin, joka sitten puolestaan laitettiin sequenceriin. Sequencer Unrealin puolella on kuin Blenderin NLA-editori. Sequencerissa animoin pilvien liikkeen ja joidenkin vedessä olevien objektien yksinkertaisen ylös ja alas päin liikkeen simuloidakseni aaltoja. Usein sequencerin kautta animoidaan myös kameralle liike, mutta tässä projektissa se ei ollut tarpeen.



Kuva 26. Unrealin sequencer

Animaatioita tuodessa Unrealiin, luo se yhdestä mallista useampia eri tiedostoja. Laitoin skeletal mesh-muodon (kuvassa 27 pinkki) skeneen ja valitsin sitten sille animaation pyörittäväksi (kuvassa 27 vihreä).



Kuva 27. Unrealin luomat tiedostot objektista, jossa mukana animaatio ja luuranko.

7.5 Valaistus ja kamera

Mikään skene ei näytä valmiilta ilman kunnollista valaistusta. Peleissä on monta tapaa hoitaa valaistus. Valoja on muutamaa erilaista tyyppiä Unrealissa, ja osa niistä löytyy myös Blenderin puolelta. Valotyyppejä, joita itse käytin projektissa olivat: suuntavalo (engl. Directional light), taivasvalo (engl. Sky light) ja pistevalo (engl. Point light). Valot itsessään voivat olla liikutettavuutensa perusteella joko staattisia, paikallisia, taikka liikutettavia. Nämä vaikuttavat valojen toiminta periaatteeseen ja liikutettavat käyttävät näistä kolmesta peleissä eniten resursseja [15].

Suuntavalo on käytössä ulkotiloissa, joissa halutaan luoda vaikutelma valosta, joka valaisee kaiken mahdollisen ympäristössä. Tämän valon suuntaa vaihtamalla saa efektin, että aurinko valaisee ympäristöä tietyssä kulmassa. Taivasvalo tallettaa skenen etäiset osat ja heijastaa ne ympäristöön valona [15]. Omassa skenessäni tämä toimi päävalon lähteenä. Pistevalo tuo vain hieman lisävalaistusta pienelle alueelle. Halusin tähtitornin kyltin erottuvan selvemmin, joten lisäsin valon siihen (kuvassa 28).



Kuva 28. Pistevalo skenessä valaisemassa kylttiä.

Joskus skenen valaistus (ja ylipäätään ulkonäkö) ei mm. valojen asetuksia muokkaamalla siltikään vastaa ihan sitä, mitä haluaisi. Esimerkiksi varjot voivat näyttää liian tummilta. Siksi skeneen voi lisätä ”Post process volumen”, jossa voi säätää mm. varjo- ja valokohtien kontrastia ja muokata renderöintiä varten skenen värisävyjä.

Pelimoottorin kamera pyrkii, niin kuin moottorin monet muutkin osat, imitoimaan tosielämänsä vastinetta. Kameralle ei tässä projektissa tarvinnut paljoa tehdä mitään, sillä se pysyi paikallaan videon aikana. Sitten kun otin joitain lähikuvia siirsin sen eripaikkaan ”pilot actor”-toiminnolla. Säädin kameran asetuksista kuitenkin sille aina oikean tarkkuuden ja zoomin. Näin kaikki tarvittava saatiin kasaan.

8 Lopetus

Renderöinnin suoritin Unrealin ”Movie render que”-ominaisuudella video- ja kuvamuotoon. Sitteen nämä kuvat ja videon laitoin ArtStationiin ja kirjoitin postauksen kuvaukseen lisätietoja teoksesta. Lisätiedoissa kannattaa mainita, mitä ohjelmia käytti, oliko projekti; koulu, työ vai henkilökohtainen ja siihen mahdollisesti muut osallistuneet. Skenen luominen oli monivaiheinen prosessi, mutta tiivistettynä ja pelkistettynä: 1) idean keksiminen, 2) mallien luonti, 3) skenen kasaaaminen, 4) renderöinti. Lopuksi vielä unohtamatta työn jakamista muiden nähtäväksi ja kommentoitavaksi. Kriitikki on oiva väline kasvulle!

Pidän siitä miltä skene näyttää. Kuitenkin huomaan sen heikkoina kohtina olevan puut ja pilvet. Pilvet olisi voinut tehdä 2D:nä ja laittaa materiaalin avulla liikkumaan. Itse materiaali olisi liitetty esimerkiksi puolipalloon, joka ympäröi koko skeneä (tätä kutsutaan skydomeksi). Skenessä puut ovat isoja ja selkeästi nähtävissä oleva elementti. Tekisin niistä vielä kiinnostavampia lisäämällä rungon tekstuuriin käsin piirrettyä, tyylieltyä kaarnakuviota ja sammalta (näkyvillä tähtitornin ja pienemmän talon pinnassa). Puiden lehtien reunoissa on myös havaittavissa visuaalista virhettä. Reunat näyttävät Unrealin puolella jostain syystä olevan vaaleampia kuin muu lehti. Syynä voi olla varjostus ongelmat tai jokin muu, kuitenkin tarkistin lehtien UV-kartan ja siinä en ainakaan nähnyt selvää virhettä.

Niin kuin mainitsin kappaleessa kaksi, jossa esittelin portfolion erilaisten töiden tyyppejä, tämä työ ei keskity ja esitä nimenomaan mallintamistaitojani niin hyvin kuin haluaisin. Se ei ole tiili vaan pilari. Tein projektin myötä laajalti erilaisia työtehtäviä, mitä joskus firmoissa useammat eri artistit tekisivät. Skenen luontiin ei olisi riittänyt vain mallien teko, vaan efektit, materiaalit ja valaistus toivat kaiken yhteen. Toki nämä efektit ja materiaalit olisi voinut ostaa valmiiksi tehtyinä. Silloin olisi säästynyt aikaa, jota olisi voinut delegoida mallintamiseen ja teksturointiin. Kaiken kaikkiaan näen kyllä tämän työn demonstroivan mukavan laajasti taitojani nimenomaan peliartistina ja 3D-generalistina siitä huolimatta, että mallintaminen jäi vähemmälle huomiolle.

Lähteet

1. Joensuu Janne, 3D-alan sanasto. [AMK-opinnäytetyö]. Kajaanin ammattikorkeakoulu; 2016. Saatavilla: <https://urn.fi/URN:NBN:fi:amk-2016060612045>
2. J Hill. How to Build a Portfolio as a Game Artist. [Viitattu 29.10.2024] Saatavilla: <https://youtu.be/44aPVgi6S6E?si=idSHmWySO33A7Ohr>
3. Tuotettu AI:lla. Mid Journey. [Viitattu 11.2024] Kuva saatavilla: <https://midlibrary.io/styles/low-poly>
4. Blender manual, Shading. [Viitattu 11.2024] Saatavilla: https://docs.blender.org/manual/en/2.91/scene_layout/object/editing/shading.html
5. What is a normal in 3D modeling: Understanding the Concept of Normals in 3D Space. Evelyn Thorne, Coohom 2024. [Viitattu 11.2024] Saatavilla: <https://www.coohom.com/article/what-is-a-normal-in-3d-modeling>
6. Quora, David Jeske. Muokattu viimeksi 2020. [Viitattu 11.2024] Saatavilla: <https://www.quora.com/Why-don-t-more-studios-use-the-free-Blender-software>
7. Map systems. 12 Best 3D Modeling Techniques Every 3D Designer should know. Saatavilla: <https://mapsystemsindia.com/resources/Beginners-3d-modeling-techniques.html>
8. Texture Atlasing: An Inside Look At optimizing 3D Worlds! Jeo Bernardi. Medium 2018. [Viitattu 11.2024] Saatavilla: <https://medium.com/immersedteam/texture-atlasing-an-inside-look-at-optimizing-3d-worlds-8a07145856d7>
9. Foundry, Modo Essentials docs: Modeling > UV Mapping. [Viitattu 11.2024] Saatavilla: <https://learn.foundry.com/modo/essentials/content/essentials/modeling/uving.html>
10. Adobe, Guide to 3D rendering. [Viitattu 11.2024] Saatavilla: <https://www.adobe.com/products/substance3d/discover/3d-rendering.html>
11. Johanna Puirava, Patikkelieliefektin luonti Unity pelimoottorilla. [AMK- opinnäytetyö]. Kajaanin ammattikorkeakoulu; 2021. Saatavilla: <https://urn.fi/URN:NBN:fi:amk-2021060714784>
12. Blender manual, Shape key introduction. [Viitattu 11.2024] Saatavilla: https://docs.blender.org/manual/en/latest/animation/shape_keys/introduction.html
13. Blender manual, Animation introduction. [Viitattu 11.2024] Saatavilla: <https://docs.blender.org/manual/en/latest/animation/introduction.html#animation>
14. Blender manual, Nonlinear animation introduction. [Viitattu 11.2024] Saatavilla: <https://docs.blender.org/manual/en/latest/editors/nla/introduction.html>

15. Unreal Engine documentation, Light mobility. [Viitattu 11.2024] Saatavilla:
https://dev.epicgames.com/documentation/en-us/unreal-engine/light-mobility?application_version=4.27

