

Low poly -mallintamisen perusteet Blenderissä

Case: Tekoälypaja

Tiivistelmä

| | | |
|--|-------------------------------------|-------------------------------|
| Tekijä(t) Peltonen, Teija | Julkaisun laji Opinnäytetyö, AMK | Valmistumisaika Syksy 2020 |
| | Sivumäärä 32 | |
| Työn nimi Low poly -mallintamisen perusteet Blenderissä Case: Tekoälypaja | | |
| Tutkinto Insinööri (AMK), Tieto- ja viestintätekniikka | | |
| Ohjaavan opettajan nimi, titteli ja organisaatio Henri Koukka, lehtori, LAB-ammattikorkeakoulu | | |
| Toimeksiantajan nimi, titteli ja organisaatio Ville Teräväinen, projektipäällikkö, LAB-ammattikorkeakoulu | | |
| Tiivistelmä <p>Tässä opinnäytetyössä tarkastellaan eri mallintamisen, teksturoinnin ja animoinnin vaiheita Blenderillä. Blender on ilmainen ja avoimen lähdekoodin 3D-mallinnusohjelma eli kuka tahansa voi käyttää sitä harrastelijasta ammattilaiseen (Blender). Sillä pystyy tekemään 3D-mallintamisen lisäksi muun muassa sculptaamista, teksturointia ja animointia.</p> <p>Opinnäytetyön tarkoituksena oli tehdä low poly -malleja Tekoälypajan VR-peliprojektiin. Tekoälypaja on tekoälyn tutkimiseen ja kehittämiseen liittyvä työpaja, jonka tavoitteena on edistää terveyden ja hyvinvoinnin sekä kiertotalousalan pk-yrityksille sensori- ja tekoälyteknologian avulla (LAB University of Applied Sciences).</p> <p>Käytännön osuudessa tullaan käymään läpi muun muassa työn tavoitteesta ja sen toteutuksesta. Työn tavoitteessa kerrotaan taustaa VR-peliprojektista ja miltä low poly -mallien tulisi näyttää. Tämän lisäksi käydään läpi työn toteutusta muun muassa mitä ohjelmia mallien toteuttamisessa on käytetty ja miltä niiden lopputulos tulee näyttämään.</p> | | |
| Asiasanat 3D-malli, animaatio, Blender, low poly, malli, mallintaminen, teksturointi | | |

Abstract

| | | |
|---|------------------------------------|--------------------------|
| Author(s) Peltonen, Teija | Type of Publication Thesis, UAS | Published Autumn 2020 |
| | Number of Pages 32 | |
| Title of Publication Low poly modeling basics in Blender Case: Tekoälypaja | | |
| Name of Degree Engineer (UAS), Information and Communication Technologies | | |
| Name, title and organization of the supervising teacher Henri Koukka, Lecturer, LAB University of Applied Sciences | | |
| Name, title and organization of the client Ville Teräväinen, Project Manager, LAB University of Applied Sciences | | |
| Abstract <p>This thesis examines the different stages of modeling, texturing and animation with Blender. Blender is a free and open source 3D modeling program, meaning anyone from amateur to professional can use it (Blender). In addition to 3D modeling, it is capable of sculpting, texturing and animation, among other things.</p> <p>The purpose of the thesis was to make low poly models for Tekoälypaja's VR game project. Tekoälypaja is a workshop related to the research and development of artificial intelligence, the aim of which is to promote health and well-being and the circular economy for SMEs through sensor and artificial intelligence technology (LAB University of Applied Sciences).</p> <p>The practical part will cover, among other things, the goal of the work and its implementation. The goal of the work tells a background about the VR game project and what low poly models should look like. In addition to this, the implementation of the work will be reviewed, among other things, what programs have been used in the implementation of the models and what their result will look like.</p> | | |
| Keywords 3D model, animation, Blender, low poly, model, modeling, texturing | | |

Sisällys

| | | |
|-------|---------------------------------|----|
| 1 | Johdanto..... | 1 |
| 2 | Mallintaminen | 2 |
| 2.1 | Blender | 2 |
| 2.2 | Mallinnus | 3 |
| 2.2.1 | Box modeling..... | 5 |
| 2.2.2 | Polygon modeling | 6 |
| 2.2.3 | Sculpting..... | 7 |
| 2.2.4 | NURBS..... | 8 |
| 2.3 | Low poly | 10 |
| 2.4 | Polygon counter..... | 11 |
| 2.5 | Unity | 12 |
| 3 | Teksturointi | 13 |
| 3.1 | Teksturointipohjan luonti | 13 |
| 3.2 | UV Unwrap & UV Map | 13 |
| 3.3 | Mallin teksturointi..... | 15 |
| 3.4 | Teksturoinnin tarkastelua..... | 16 |
| 4 | Animointi..... | 18 |
| 4.1 | Bone & Skeleton..... | 18 |
| 4.2 | Rigging | 19 |
| 4.3 | Animaation toteutus | 20 |
| 5 | Käytännön osuus | 23 |
| 5.1 | Toimeksiantaja | 23 |
| 5.2 | Työn tavoitteet | 23 |
| 5.3 | Toteutus | 24 |
| 5.4 | Ongelmat ja ratkaisut..... | 27 |
| 6 | Yhteenveto | 30 |
| | Lähteet | 31 |

1 Johdanto

3D-malli on kolmiulotteinen tietokoneella tuotettu graafinen malli, joka voi olla yksinkertainen tai yksityiskohtainen (Rector 2019). Niitä käytetään pääsääntöisesti mainoksissa, elokuvissa ja videopeleissä. Jälkimmäisessä mallien käyttö on yleistynyt muutaman vuosikymmenen ajan, jolloin niiden käyttö yleistyi pikkuhiljaa.

Tässä opinnäytetyössä tarkastellaan eri mallintamisen, teksturoinnin ja animoinnin vaiheita Blenderillä. Näiden eri vaiheiden läpikäymisen tarkoituksena on pohjustaa käytännön osuutta.

Toimeksiantajana on Tekoälypaja, jonka tavoitteena on edistää terveyden ja hyvinvoinnin sekä kiertotalousalan pk-yrityksille sensori- ja tekoälyteknologian avulla. Tekoälypaja on osa LAB-ammattikorkeakoulun hanketta. (LAB University of Applied Sciences).

Tämän opinnäytetyön tavoitteena oli tehdä low poly -malleja toimeksiantajan VR-peliprojektiin, jonka mallinnukset liittyvät jollain tapaan luontoon ja retkeilyyn. VR-pelissä pelaaja pääsee low poly -maailman luontoon tekemään eri aktiviteettejä kuten jousiammuntaa ja tammenterhojen poimintaa. Pelissä pääsee myös ihaillemaan maailman luontoa.

VR-peliprojektin mallien mallinnus ja animoinnit toteutetaan Blenderillä. Mallintamisessa tullaan käyttämään box modeling -mallinnustekniikkaa, jolla on kätevä tehdä low poly -malleja. Mallien teksturoinnissa käytetään joko Blenderiä tai PaintTool SAI -piirustusohjelmaa riippuen mallin teksturoinnin yksityiskohtien tarpeesta.

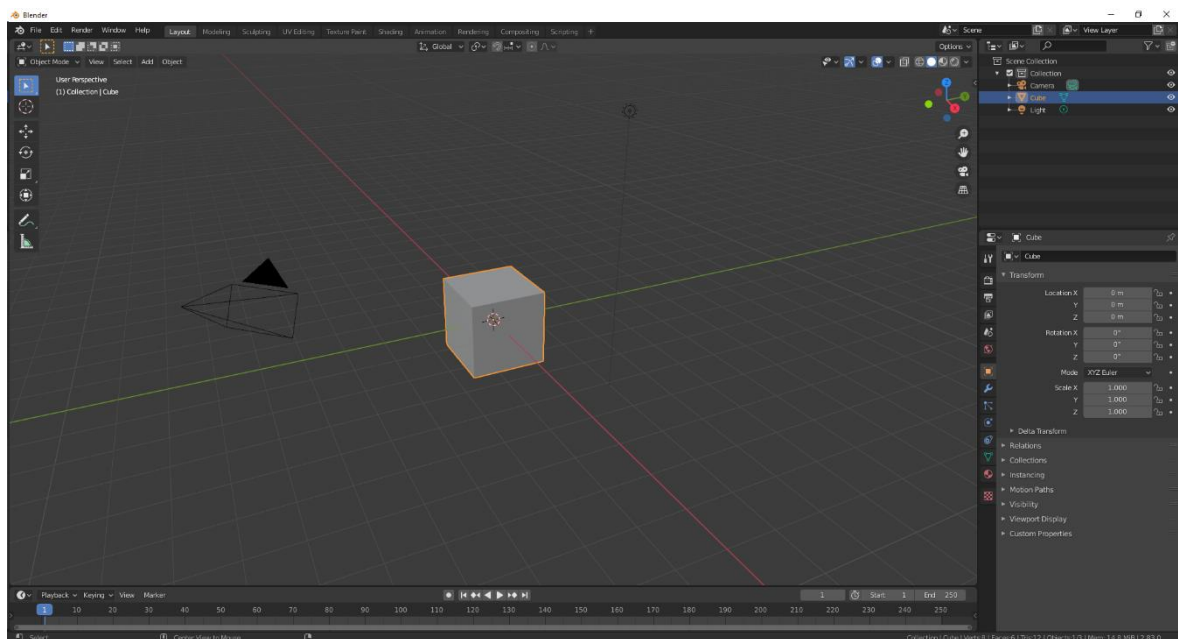
2 Mallintaminen

2.1 Blender

3D-mallintaminen alkaa itselle parhaan mallinnusohjelman valinnalla. Niitä löytyy paljon erilaisia eri mallinnuskäyttötarkoituksiin ja suurin osa ohjelmista on maksullisia ilmaisen kokeilujakson jälkeen. Myös suurin osa niistä saatavilla vain Windowsille ja/tai Macille. Parhaiten tästä joukosta erottuu mallinnusohjelma nimeltä Blender.

Blender on ilmainen ja avoimen lähdekoodin 3D-mallinnusohjelma eli kuka tahansa voi käyttää sitä harrastelijasta ammattilaiseen (Blender). Mallintamisen lisäksi sillä voidaan tehdä sculptaamista, teksturointia, animointia ja jopa renderöintiä. Käytännössä kaikki 3D-mallin tekovaiheet voidaan tehdä yhden ohjelman avulla ja tehden Blenderistä monipuolisen ohjelman.

Kuvassa 1 nähdään, kuinka Blenderin käyttöliittymästä on tehty mahdollisimman helposti ymmärrettävä. Keskellä käyttöliittymää on 3D-ympäristö, missä mallinnus tapahtuu ja siinä olevien objektien nimet näkyvät oikealla yläreunassa. Tämän alapuolella on eri asetuksia valittuun malliin liittyen. Alareunassa oleva palkki näyttää mallin mahdollisen animaation, jos semmoista on tehty. Yläpuolella vastaavasti löytyy eri välilehtiä jokaiselle osa-alueelle, mitä muuta mallille mallintamisen lisäksi voidaan tehdä.



Kuva 1. Kuvakaappaus Blender 2.83.0 käyttöliittymästä

Ohjelma on saatavilla Windowsin ja Macin lisäksi myös Linuxille. Tämä on hyvä Linuxin käyttäjille, koska mallinnusohjelmien valikoima kyseiselle käyttöjärjestelmälle on todella suppea. Blenderin pääsee lataamaan heidän omilta sivuiltaan tai Steam-pelipalvelimesta.

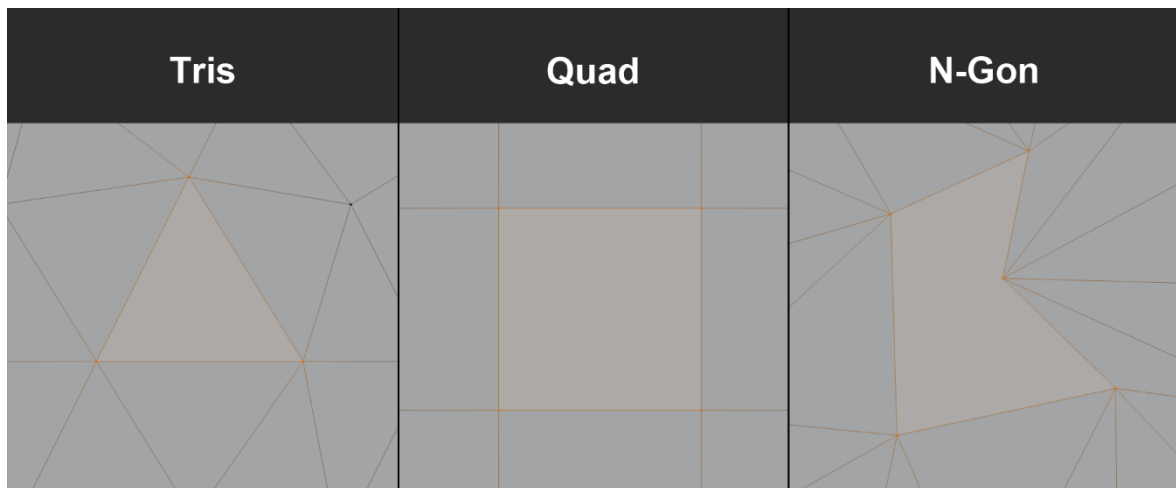
2.2 Mallinnus

3D-malli on kolmiulotteinen tietokoneella tuotettu graafinen malli, joka voi olla yksinkertainen tai yksityiskohtainen (Rector 2019). Sitä voidaan kutsua myös 3D-mallin lisäksi nimillä malli, polygonimalli ja objekti. Jälkimmäistä nimitystä voidaan käyttää silloin, jos yksittäinen malli on tehty useamman mallin yhdistelmästä.

3D-mallinnusohjelma tarvitaan mallin luomiseen ja tekemiseen, joita on olemassa paljon erilaisia. Eri 3D-mallinnusohjelmilla luodut mallit ovat tarkoitettu eri käyttötarkoituksiin esimerkiksi peliteollisuuden mallinnuksia tuotetaan yleensä Autodesk 3ds Maxilla tai Blenderillä. Vastaavasti SolidWorksiä käytetään mm. tuote- ja mekaniikkasuunnittelussa, joissa malleille tarvitaan tarkat mittasuhteet.

Mallin ulkonäkö muodostuu aina kolmesta osiosta: vertexeistä, edgeistä ja faceista. Vertexit ovat malleissa pisteitä, jotka yhdistyvät edgen avulla luoden 3D-mallille yksittäisen facen. Facet ovat polygoneja, joista muodostuu polygonimalli.

Polygonimallin yksittäinen polygoni muodostuu yleensä neljästä vertexistä ja edgestä. Nelikulmaista polygonia kutsutaan nimellä quad. Harvemmin malleissa käytetyt polygonit ovat nimiltään tris ja N-Gon. Tris muodostuu kolmesta vertexistä ja edgestä, joka tekee polygonista kolmion muotoisen. Vastaavasti N-Gon muodostuu vähintään viidestä vertexistä ja edgestä. (Turbosquid 3D Resources). Kuvassa 2 havainnollistaa, miltä kyseiset polygonit näyttävät.



Kuva 2. Polygonit tris, quad & N-Gon

Aloitettaessa mallintamista, kannattaa miettiä miten mallin haluaa rakentaa. Sen voi luoda joko yhdestä objektista alusta loppuun tai yhdistellä useampaa objektia tehden siitä yhden kokonaisuuden. Jälkimmäinen onnistuu valitsemalla kaikki yhdistettävät objektit niiden ollessa oikeilla paikoillaan ja käyttämällä join-toimintoa, joka yhdistää valitut objektit yhdeksi kokonaiseksi objektiksi. Objektien ei tarvitse välttämättä koskettaa toisiaan yhdistämisvaiheessa.

Yksittäinen malli suositellaan kuitenkin tekemään vain yhdestä objektista, jos se sisältää paljon animaatiota. Yhdistetyistä objekteista tehdyt mallit tulevat hajoamaan jonkin verran animaatiovaiheessa, vaikka objektien pinnat koskettaisivatkin toisiaan. Tämä johtuu siitä, että join-toiminto ei kiinnitä objekteja toisiinsa vertexien ja edgeiden avulla vaan ne ns. sulautuvat kiinni toisiinsa ja niitä voidaan jälkikäteen siirrellä pois toisistaan.

Yksittäistä mallia voidaan pyöritellä ja vaihtaa sen sijaintia X-, Y- tai Z-akselin suuntaisesti. Akselien värit näkyvät punaisena, vihreänä ja sinisenä. Punainen edustaa X-akselia, vihreä Y-akselia ja sininen Z-akselia. Varsinkin mallin pyörimiseen vaikuttaa paljon sen keskipisteen sijainti.

Keskipiste näkyy valitussa mallissa oranssina pisteenä, joka kertoo missä kohtaa mallia se sijaitsee. Esimerkiksi jos mallin keskipiste sijaitsee sen yläosassa, se ei tule pyörimään tasan tasaisesti. Tämä tulee vaikeuttamaan oikean kulman löytymistä ja tulee hankaloittamaan mallin sijoittamista tietylle paikalle. Tästä syystä keskipisteen määrittäminen malliin kannattaa luoda vasta mallintamisen jälkeen, koska mallintamisvaiheessa sen keskipiste tulee pysymään samana koko mallintamisen ajan eli se ei päivity mallinnuksen aikana.

Yksi tärkeä osa mallinnusohjelman hallintaa on opetella toimintojen pikanäppäimet, jotka helpottavat ja nopeuttavat mallintamista huomattavasti verraten toimintojen klikkailua kuvakkeista. Vähintään perustoimintojen pikanäppäimet kannattaa opetella ulkoa. Näitä ovat objektiin tarttuminen G-kirjaimesta (grab), skaalaus S-kirjaimesta (scale) ja rotaatio R-kirjaimesta (rotate). Myös X-, Y- tai Z-akselien käyttö perustoimintojen kanssa helpottavat mallin sijoittamista ja sen pyörittämistä. (Blain 2015, 44). Näiden avulla pääsee alkuun mallintamisessa.

Mallintamisesta on paljon eri tekniikoita, joista jokainen poikkeaa toisistaan jollain lailla. Jokaisella tavalla toteutettu malli on oikea tapa ja onkin sen tekijästä kiinni millä tavalla haluaa toteuttaa niitä.

2.2.1 Box modeling

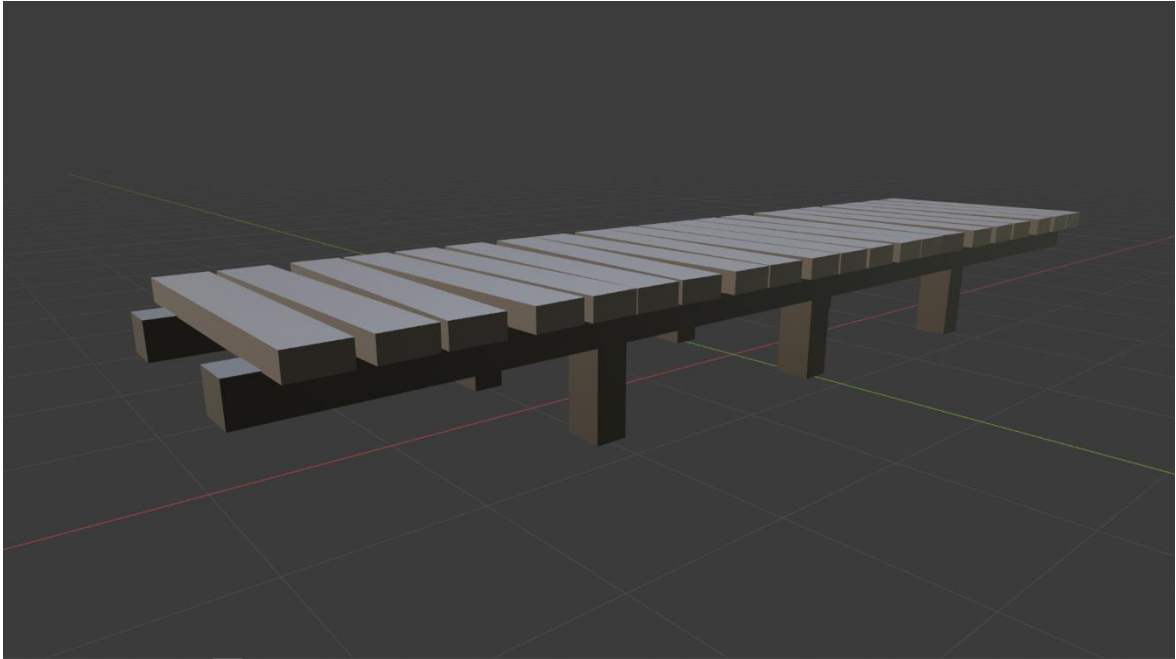
Yleisin mallinnustekniikka on nimeltään box modeling, jossa mallinnuksen luominen aloitetaan kuution tai pallon muotoisesta mallista (Selin). Kyseinen malli sisältää yleensä vain vähän polygoneja, mikä tekee siitä helposti muokattavan. Sitä muotoillaan niin kauan, että haluttu ulkonäkö saavutetaan. (Slick 2019).

Box modeling -tekniikkaa käyttävät 3D-mallintajat lisäävät malliin polygonien määrää jakamalla niitä. (Slick 2019). Nämä onnistuvat loop cutin, extruden ja subdivisionin avulla. Loop cutin avulla jaetaan malli osiin riippuen siitä, kuinka moneen osaan se halutaan jakaa. Toiminnolla voidaan jakaa malli X-, Y- tai Z-akselin suuntaisesti kiertäen mallin ympäri.

Extruden avulla saadaan malliin lisättyä uutta geometriaa, joka näkyy siinä joko kuoppaisena muotona tai uutena ulokkeena. Tämä toiminto helpottaa materiaalin lisäämistä niistä kohdista, joihin kohtiin halutaan lisätä geometriaa.

Tarvittaessa voi käyttää subdivision-toimintoa, joka lisää mallin geometriaa vertexien, edgejen ja faceiden väliin (Selin). Tämä lisää mallin polygonien määrää nelinkertaisesti, mikä tekee sen pinnasta sileämmän näköisen. Tätä toimintoa ei kuitenkaan ole pakko käyttää, jos malliin ei haluta lisää polygoneja.

Näiden toimintoja käyttäen box modeling -tekniikassa voidaan luoda yksinkertaisista aina monimutkaisiin malleihin eli mielikuvitus on vain rajana. Esimerkiksi kuvassa 3 oleva malli on toteutettu box modeling -mallinnustekniikalla. Sen mallintamisessa on käytetty apuna extrudea ja loop cutia.



Kuva 3. Box modeling -mallinnustekniikalla luotu laitur

Yhteenvetona box modeling on mallintamistekniikka, joka on aloittelijaystävällinen. Tästä syystä monet 3D-mallintajat suosittelevat kyseistä tekniikkaa ensikertalaisille, jotka eivät ole aikaisemmin mallintaneet.

2.2.2 Polygon modeling

Polygon modeling on saman tyyppinen mallinnustekniikka kuin box modeling, mutta tässä mallintaminen alkaa yhdestä vertexistä ja siitä aletaan kasvattamaan mallin kokoa käyttäen vertexejä ja edgejä. Niitä lisätään extruden avulla tehden faceja eli polygoneja, jotka luovat mallin. (Selin).

Mallinnuksen tekeminen polygon modeling -tekniikalla on yleensä hidasta. Siksi polygon modelingin kanssa käytetään mirror-toimintoa, jonka avulla saadaan mallin tekemiseen symmetristä jälkeä. Symmetrisyyttä voidaan toteuttaa X-, Y- tai Z-akselissa. Sitä voidaankin käyttää kaikkien symmetristen muotojen toteuttamiseen, joka nopeuttaa mallintamista huomattavasti. Myös subdivisionin käyttö on yleistä polygon modeling -tekniikassa mallin geometrian pehmentämiseksi (Selin).

Polygon modelingissa ja box modelingissa käytetään samoja toimintoja mallin tekemisessä, mikä tekee niistä melkein identtiset mallinnustekniikat. Kuitenkin mallintaminen aloitetaan eri tavoilla, joka tekee mallinnustekniikoista erilaiset. (Selin).

2.2.3 Sculpting

Sculpting eli sculptaaminen on ns. digitaalista veistämistä malliin ja sitä voi verrata saven veistämiseen. Digitaalinen veistäminen on tuonut helpotusta hahmojen ja olentojen mallinnuksen, jonka avulla saadaan tehtyä todella paljon yksityiskohtia malliin suuren polygonimäärän ansiota (Slick 2019).

Box modelingin ja polygon modelingin tavoin sculptaauksessakin käytetään vertexejä, edgejä ja faceja (Selin). Myös mallintaminen aloitetaan kuutiosta tai pallosta box modelingin tavoin. Kuitenkin näille molemmille muodoille lisätään polygonien määrää subdivision-toiminnon avulla ennen sculptausta, jotta mallin veistäminen tulisi olemaan helppoa.

Mallin veistämiseen käytetään toimintojen sijaan erilaisia työkaluja, joiden avulla saadaan tehtyä eri muotoja suoraan malliin. Näistä tärkeimmät ovat mallia kohottavat ja syvyyttä lisäävät työkalut. Työkaluista on myös erilaisia variaatioita, jotka helpottavat sculptaamista. Näitä ovat muun muassa mallin tasoittamis- ja venyttämistyökalut.

Sculptaamista käytetään yleensä silloin, kun mallin polygonien määrällä ei ole väliä tai halutaan tehdä low poly -mallille yksityiskohtia normal mapin avulla. Tämä tekee low poly -mallista high polyn näköisen. Low poly on malli, joka sisältää vähän polygoneja ja vastavasti high poly -malli sisältää paljon polygoneja (Munn; Derakhshani 2008, 240).

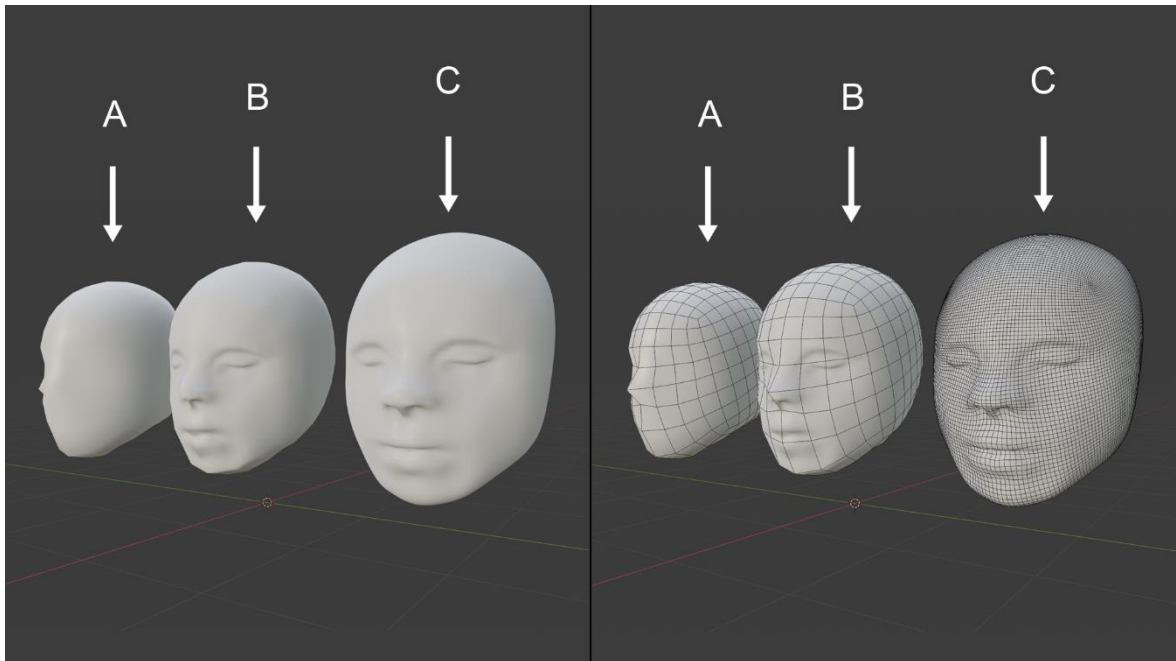
Normal mapin käyttäminen mallissa sen pinta saadaan näyttämään siltä kuin siinä olisi yksityiskohtia ilman geometrian lisäämistä (Treehouse 2015). Näitä ovat esimerkiksi kohoumat ja painanteet.

Tällä tavalla luodussa mallissa kannattaa luoda halutusta mallista ensin high poly -versio ja muotoilla se sculptaauksen avulla. Tämän jälkeen on helppo tehdä siitä low poly -versio decimate-toiminnon avulla, joka vähentää polygonien määrää mallista. Tämä tekee mallista identtisen näköisen high poly -version kanssa, mutta vähemmillä polygoneilla.

Tämän jälkeen high poly -mallista luodaan normal map low poly -mallille bake-toiminnon avulla. Ennen sen tekoa mallien sijainti pitää olla sama, jotta bake-toiminto saadaan onnistumaan. Syynä on se, että high poly -mallin ulkonäkö tulee kopioitumaan normal mapille

vaan niistä kohdista low poly -malliin, mihin high poly -malli koskettaa. Valmis malli tulee näyttämään high polyn ja low polyn sekoitukselta.

Alla olevasta kuvasta 4 näkyy, miltä normal mapilla varustettu low poly -malli näyttää (B). Sen teossa on käytetty sen oikealla puolella (A) olevaa low poly -mallia ja vasemmalla puolella (C) olevaa high poly -mallia hyödyksi. Mallien polygonit näkyvät kuvan oikealla puolella.



Kuva 4. Ihmisen kasvot toteutettu sculptaamisella

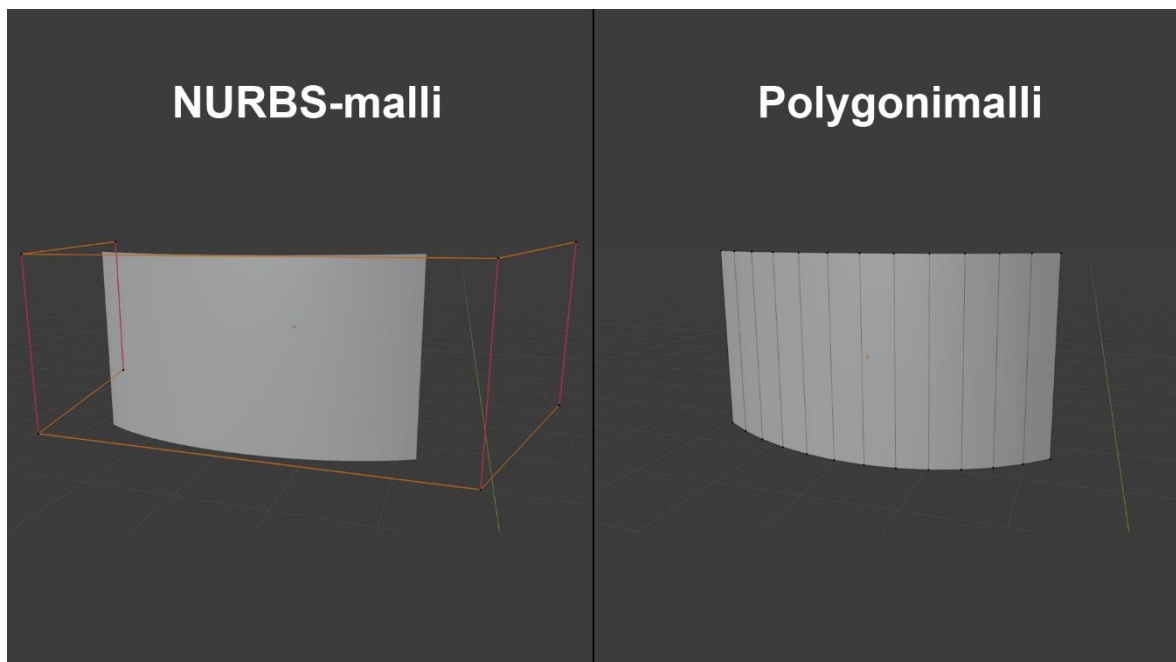
Yhteenvetona sculpting-mallintamistekniikka sopii niille, jotka haluavat tehdä näyttäviä ja yksityiskohtia omaavia hahmoja ja olentoja veistämällä. Sculptauksessa ei tarvitse murehtia polygonien määrästä, jos malli luodaan esimerkiksi portfoliota varten.

2.2.4 NURBS

NURBS eli non-uniform rational b-spline poikkeaa muista mallintamistavoista huomattavasti. Sillä luotu malli koostuu 3D-pinnasta, joka on upotettu verkkomaisiin käyriin eli se ei sisällä ollenkaan vertexejä, edgejä tai faceja. Tämä tarkoittaa sitä, että tällä mallinnustekniikalla luotu malli ei ole polygonimalli. (Slick 2019).

Yksittäistä käyrän muotoa voidaan muokata ohjauspisteiden (control vertices) avulla, joilla voidaan lisätä malliin kaarevuutta. Ohjauspisteet muistuttavat paljon ulkonäöltään ja liikkuvuudeltaan vertexejä, vaikka ovatkin kaksi eri asiaa.

Mallin teossa voidaan käyttää samoja toimintoja kuin muissakin mallinnustekniikoissa. Subdivision-toiminnon avulla saadaan lisättyä käyrään lisää ohjauspisteitä, joiden avulla voidaan tehdä malliin monimutkaisempia muotoja. Myös extrude-toimintoa voidaan käyttää lisätäkseen samalla lailla mallin geometriaa kuten muissakin mallinnustekniikassa. Kuva 5 havainnollistaa, miltä samannäköinen malli näyttää NURBS-mallina ja perinteisellä polygonimallinnuksella tehtynä.



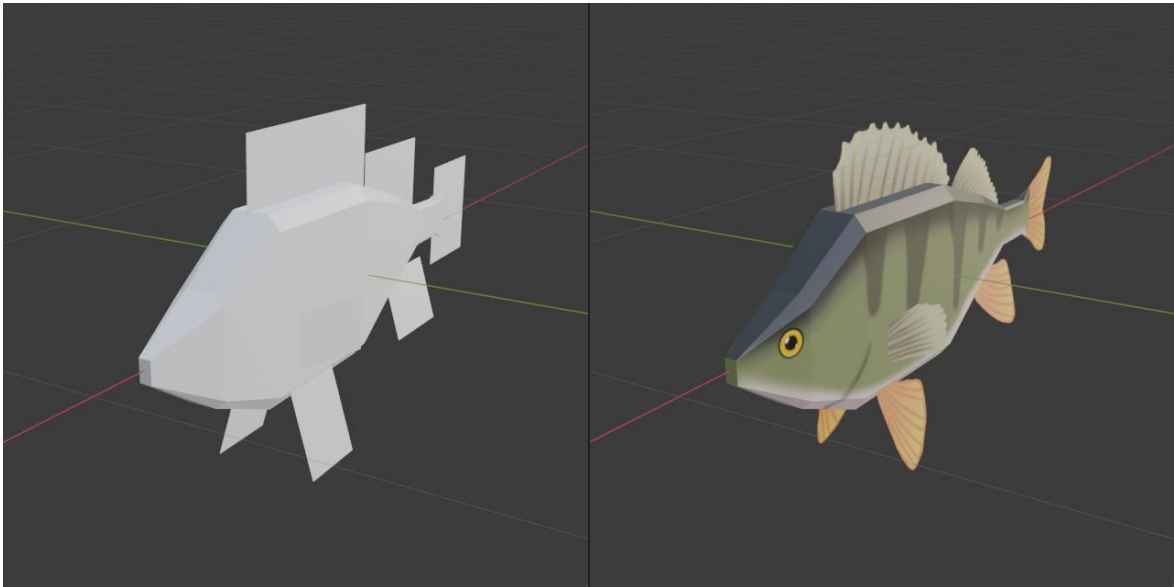
Kuva 5. Vertailussa NURBS-malli ja polygonimalli

NURBS-mallinnustekniikkaa käytetään enemmän muun muassa 3D-tulostettavissa kappaleissa, jolloin mallissa tarvitsee olla tarkat mittasuhteet ja sen pinta sileä. Polygoneista muodostuva mallin pinta ei ole samanlailla tasainen kuin NURBSilla tehty, koska NURBS-tekniikalla toteutettu mallin pintaan ei jää kohoumia samalla lailla kuin polygonimalliin. Tämän takia mallit toteutetaan yleensä CAD-ohjelmien kuten esimerkiksi Solidworksin avulla, koska niissä on enemmän työkaluja NURBSille ja ohjelmat ovat suunniteltu tarkkojen mitasuhteilla tarvitsevien mallien mallinnukseen. (Selin).

2.3 Low poly

Polygonimallit luokitellaan kahteen eri kategoriaan: low poly- ja high poly -malleihin. High poly -mallit sisältävät paljon polygoneja luoden mahdollisuuksia tehdä enemmän yksityiskohtia esimerkiksi sculptaamisen avulla. Vastaavasti low poly -malli sisältää vain vähän polygoneja tehden siitä helposti mallinnettavan. (Munn; Derakhshani 2008, 240).

Low poly -mallin ei tarvitse välttämättä koostua yksivärisistä pinnoista, jos siihen halutaan lisätä syvyyttä tai realismia. Teksturoinnin avulla low poly -malleihin saadaan tehtyä yksityiskohtia, jotka voivat olla jopa realistisia. Alapuoella oleva kuva 6 havainnollistaa, kuinka paljon pelkkä teksturointi vaikuttaa mallin ulkonäköön.



Kuva 6. Low poly -tyylillä toteutettu ahven ilman teksturointia ja sen kanssa

Vähäisen polygonien määrän ansiosta low poly -mallit vaativat vähemmän tehoa tietokoneelta verrattuna high poly -malleihin. Niitä käytetäänkin pääasiassa videopeleissä. Pelattaessa videopeliä polygonimäärän ollessa liian suuri, se ilmenee usein pelin hitautta tai jopa koneen ylikuumentumista. Pahimmassa tapauksessa peli voi rikkoa koneesta komponentin ylikuumentumisen seurauksena.

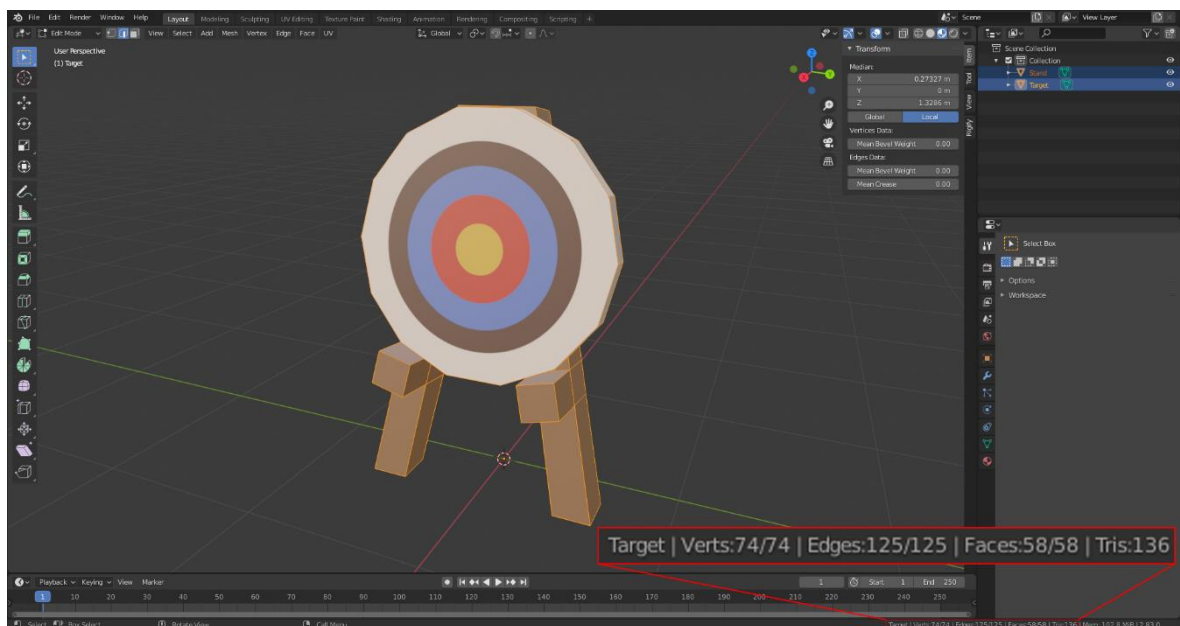
Low poly -mallin pinnan ei tarvitse olla kulmikkaan näköinen, vaikka se sisältäisi vähän polygoneja. Polygonien kulmat pystytään pehmentämään shade smooth -toiminnon avulla. Toiminto tekee siitä pehmeämmän näköisemmän ilman polygonien lisäämistä. Tästä syystä shade smooth -toimintoa käytetään yleisesti mallintamisessa. Tätä toimintoa ei kuitenkaan

tarvitse käyttää, jos halutaan säilyttää polygonimallissa sen kulmat terävinä. Kyseiset mallit ovat yleisesti tehty low poly -tyylillä.

2.4 Polygon counter

Aloitettaessa low poly -mallinnuksen tekoa on hyvä seurata polygonien määrää, jotta pysytään matalissa polygonien määrissä. Polygonimäärän seuraaminen onnistuu polygon counterilla, joka on yleensä 3D-mallinnusohjelmassa mukana. Polygon counter saattaa näyttää erilaiselta eri 3D-mallinnusohjelmassa, vaikka niiden käyttötapa on sama.

Polygon counterilla voidaan polygonien määrän lisäksi laskea mallista vertejejä, edgejä ja trisejä. Polygonien määrä näkyy faces-kohdassa. Blenderin 2.83.0-versiossa polygon counter löytyy ohjelman oikealta alanurkasta, joka näkyy kuva 7 tapaisesti.



Kuva 7. Maalitaulun sisältämät polygonit

Polygon counteria yleensä käytetään silloin, kun yksittäisen mallin polygonien määrä halutaan pitää tietyssä määrässä. Tämä on tärkeää etenkin peliteollisuudessa, jossa liiallinen polygonien määrä mallissa hidastaisi pelin pelattavuutta.

2.5 Unity

Unity on pelimoottori, jonka avulla voidaan kehittää 2D- ja 3D-pelejä eri alustoille. Myös VR-pelien ja jopa animaatioiden tekeminen on mahdollista. Tunnettuja pelejä, joita on luotu Unityn avulla ovat mm. Cuphead ja Hollow Knight.

Kyseisestä pelimoottorista on ilmaisia ja maksullisia versioita, riippuen siitä halutaanko tietää sillä tehdystä pelistä vai ei. Ilmaiset versiot Unitystä ovat Personal ja Student ja vastaavasti maksulliset versiot ovat Plus, Pro ja Enterprise. Kaikissa muista paitsi Student-versiossa voidaan tuottaa rahaa Unityllä tehdystä pelistä tai animaatiosta. Ainoastaan Personal-versiossa tuloraja on satatuhatta dollaria viimeisen 12 kuukauden ajalta, kun puolestaan muissa versioissa ei ole mitään tulorajaa. (Unity Store 2020).

Pelimoottori Unity on monipuolinen pelien teossa. Graafisessa toteutuksessa voidaan käyttää omia importattuja malleja, jotka ovat tehty esimerkiksi Blenderillä tai Autodesk 3ds Maxilla. Pelien toteuttamisessa voidaan käyttää myös Unityn omasta Asset Storesta ladattuja malleja eli asetteja. Assetit (eng. assets) ovat Asset Storessa olevia tiedostoja, joita ovat muun muassa 3D-mallit ja äänitiedostot. Asetteja on sekä ilmaisia että maksullisia. (Unity 2020a).

Graafisen toteutuksen lisäksi Unityssä voi koodata koodieditorin avulla. Näitä ovat muun muassa Visual Studio Community, Visual Studio ja Notepad++ (Unity 2020b). Projektin kooditiedostoihin tehdyt muutokset päivittyvät reaaliaikaisesti projektiin. Siten niitä ei tarvitse ladata uudestaan joka kerta projektiin, jos niitä muutetaan. Oletuksena Unity käyttää C#-koodauskieltä, joka on yksi yleisimmistä koodauskielistä (Unity Documentation 2018).

3 Teksturointi

3.1 Teksturointipohjan luonti

Teksturointi on tärkeä osa mallintamista. Sillä luodaan mallille pintamateriaali, jolloin malli ei näytä enää yksiväriseltä yksinkertaiselta geometrialta. Teksturoinnin toteuttamisessa voi käyttää mallinnusohjelman omaa tai kolmannen osapuolen ohjelmaa riippuen siitä, millä ohjelmalla teksturoinnin tekoa on tottunut käyttämään.

Mallin teksturoinnin luominen alkaa teksturointipohjan teosta, joka on aluksi tyhjä. Blenderissä teksturointipohja luodaan UV Editing -välilehdessä. Teksturointipohja voi olla minkä kokoinen tahansa riippuen siitä, kunhan se on sekä pysty- että vaakasuorassa saman pituiset. Teksturointipohja voi olla esimerkiksi kokoa 1024x1024px tai 2048x2048px. (Blender 2.90 Manual).

Tyhjän teksturointipohjan luomisen jälkeen se pitää muistaa liittää malliin, johon kyseinen teksturointi halutaan luoda. Ilman tämän vaiheen toteuttamista teksturointi ei tule näkyviin malliin, vaikka teksturointi olisi kokonaan valmis.

Kaikki mallit eivät kuitenkaan tarvitse teksturointia, jos malli tarvitsee vain yksittäisiä värejä ilman yksityiskohtia. Näitä värejä kutsutaan nimellä värimateriaalit. Ne voidaan lisätä suoraan malliin sen materials-välilehdeltä. Värimateriaalin lisäys on tehty helpoksi Blenderissä.

3.2 UV Unwrap & UV Map

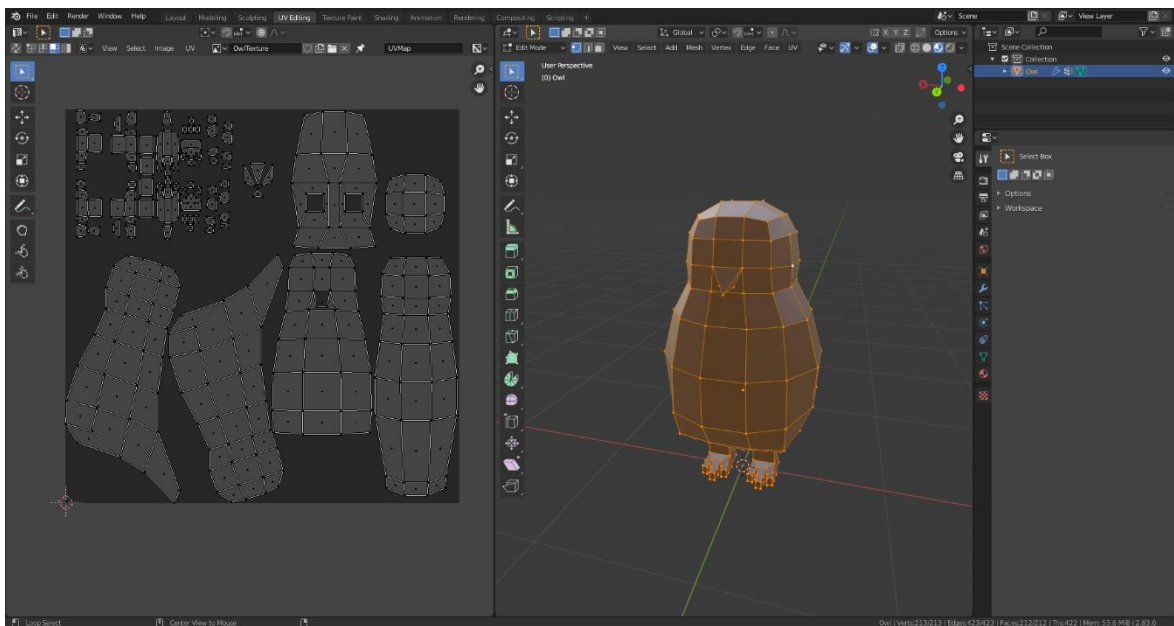
Teksturointipohjan luomisen jälkeen luodaan mallin olevasta pinnasta 2D-karttapohja, jota kutsutaan nimellä UV Map. UV-sanassa olevat U- ja V-kirjaimet vastaavat mallin X- ja Y-akseleita. (Denham 2020). Mallinnuksen kaikki facet eli polygonit kannattaa muistaa valita ennen UV Mapin tekoa, koska UV Unwrap tekee vain valituista mallin polygoneista UV Mapin. Lyhyesti UV Unwrap on UV Mapin tekovaihe, jonka pohjalle luodaan mallille teksturointia.

Blenderin UV Editing -välilehdessä UV Unwrapin voi luoda kolmella eri tapaa: Unwrap, Smart UV Project ja Lightmap Pack (Blender 2.79 Manual). Kaikki edellä mainitut tavat ovat käyttökelpoisia, vaikka ne poikkeavatkin toisistaan jonkin verran.

Unwrap on kaikkein yleisin tapa tehdä UV Map mallinnukselle. Ennen sen tekoa mallinnukseen tehdään saumoja (eng. seams), joka määrittää mallinnuksen UV Mapin ulkonäön.

(Blender 2.79 Manual). Saumojen avulla eritellään mallin erityyppisistä teksturointia tarvitavat osat toisistaan.

Smart UV Project on melkein samanlainen kuin Unwrap, ainoana erona Smart UV Project luo automaattisesti saumat UV Mapiin (Blender 2.79 Manual). Tämä säästää aikaa UV Unwrapauksessa ja se on aloittelijaystävällisempi vaihtoehto UV Mapin tekemiseen, jos sen teosta on joko vähäistä kokemusta tai siitä ei ole ollenkaan. Kuva 8 UV Mapin teossa on käytetty Smart UV Projectia.



Kuva 8. Pöllön UV Map UV Unwrapin jälkeen

Viimeisenä on Lightmap Pack, joka poikkeaa kahdesta muusta tavasta huomattavasti. Se pakkaa kaikki mallissa olevat polygonit tiiviisti vierekkäin, jota ei esiinny kahdessa muussa tavassa tehdä UV Map. (Blender 2.79 Manual). Tätä tapaa on hyvä käyttää, jos mallinnuksessa on paljon polygoneja ja niitä on vaikea saada mahdutettua kaikkia yhdelle teksturointipohjalle.

UV Mapin luomisen jälkeen se tulee näkyviin teksturointipohjan päälle 2D-karttapohjana. Tässä vaiheessa voidaan tarkistaa tarvittaessa, että UV Unwrap on onnistunut mieleisellä tavalla. Kuitenkin välillä saattaa tulla päällekkäisiä polygoneja, jotka sotkevat teksturoinnin tekoa. Päällekkäisyydet ovat kuitenkin harvinaisia ja ne voidaan erotella toisistaan vertejen, edgejen tai facejen avulla. Päällekkäisyyttä esiintyy harvemmin low poly -mallinnuksissa vähäisen polygonimääränsä ansiota.

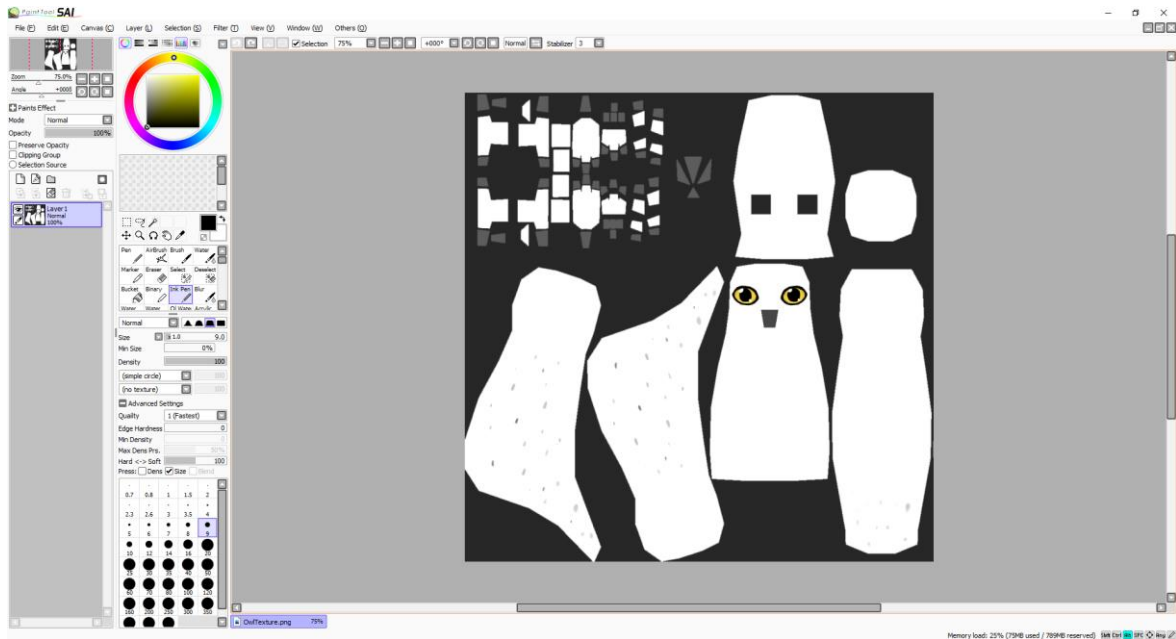
3.3 Mallin teksturointi

UV Mapin luomisen jälkeen aloitetaan teksturointi. Sen tekemiseen on muutamia vaihtoehtoja ja niistä voi valita parhaimman vaihtoehdon itselleen riippuen siitä, millä teksturoinnin haluaa toteuttaa.

Aluksi tehdään mallille pohjaväri, joka erottaa teksturointipohjan mallin UV Mapista. Pohjavärin laittaminen malliin tapahtuu maalaamalla suoraan mallin päälle bucket-työkalun avulla maalaten koko mallin yksiväriseksi. Malleihin voidaan tarvittaessa lisätä yksittäisiä värejä tiettyihin kohtiin suoraan teksturointipohjaan, jonka mallin alueet ovat eri väreisiä kuten esimerkiksi kuution yksittäiset facet.

Ensimmäisenä vaihtoehtona teksturoinnin toteuttamisessa voidaan käyttää Blenderin omaa teksturoinnin toteuttamiseen tarkoitettua ominaisuutta nimeltä Texture Paint. Sillä voidaan tehdä tarvittavia yksityiskohtia brush-työkalulla ja symmetristen kuviointien toteuttamisessa voidaan käyttää mirror-toimintoa. Yksityiskohdat voidaan tehdä joko mallin tekstuuriin tai suoraan sen pintaan.

Toinen vaihtoehto teksturoinnin toteuttamisessa on yleisempi ja enemmän käytetty: tallennetaan teksturointipohja, joka sisältää mallin UV Mapin pohjaväreineen. Tällaiset tekstuurit toteutetaan piirustus- tai kuvankäsittelyohjelmalla. Näitä ovat esimerkiksi PaintTool SAI ja Photoshop. Ohjelmat sisältävät enemmän työkaluja piirtämiseen kuin Texture Paint, joiden ansioita monimutkaisemmat yksityiskohdat onnistuvat näillä ohjelmilla paremmin. Esimerkki kuvan 9 teksturointi on toteutettu PaintTool SAIssa.



Kuva 9. Pöllön teksturointia PaintTool SAIssa

Kolmas vaihtoehto poikkeaa kahdesta aikaisemmasta, koska mallin teksturointipohja tarvitsee vain UV Mapin ilman erillistä värierotteluprosessia. Tähän syynä on se, että teksturointi toteutetaan maalaamalla yksittäistä tai useampaa valmista tekstuuria suoraan malliin teksturointiohjelman kuten Substance Painter tai Mudbox avulla. Ne sisältävät erilaisia valmiita tekstuureja, jotka ovat yleensä realistisia ja niiden väriä voidaan vaihtaa mieleisensä mukaan.

3.4 Teksturoinnin tarkastelua

Teksturoinnin luomisen jälkeen valmiin lopputuloksen näkee mallista, mihin se on luotu. Se näkyy suoraan teksturoinnin tallentamisen jälkeen, jos teksturointipohja oli liitetty malliin aikaisemmin. Valmis teksturointi näyttää alla olevan kuvan 10 mukaiselta.



Kuva 10. Pöllö valmiin teksturoinnin kera

Hyvin tehdyn teksturoinnin tunnistaa siitä, että se sopii täydellisesti malliin ja tehden mallista hyvännäköisen. Esimerkiksi teksturointiin ei tule vääristymiä mallin pinnalle näkymällä sen venymisellä. Tämä voi pilata vaikutelman hyvästä teksturoinnista, vaikka itse tekstuurissa ei olisi mitään vikana.

Tällaiset ongelmat yleensä johtuvat epäonnistuneesta UV Mapin teosta, jonka korjaaminen voi olla hankalaa jälkikäteen teksturoinnin ollessaan valmis. Pahimmassa tapauksessa koko teksturointiprosessi voidaan joutua tekemään alusta, jos teksturointi on ehditty tekemään loppuun asti ennen ongelman huomaamista.

UV Unwrapin teon jälkeen kannattaakin testata UV Mapin toimivuus esimerkiksi maalamalla viivoja suoraan malliin Texture Paintilla, jotta nähdään miten viivat asettuvat malliin. Viivojen väriksi kannattaa laittaa mikä tahansa väri, joka erottuu teksturoinnista selkeästi. Tällä menetelmällä nähdään mahdolliset virheet UV Mapissa ja ne saadaan siten korjattua.

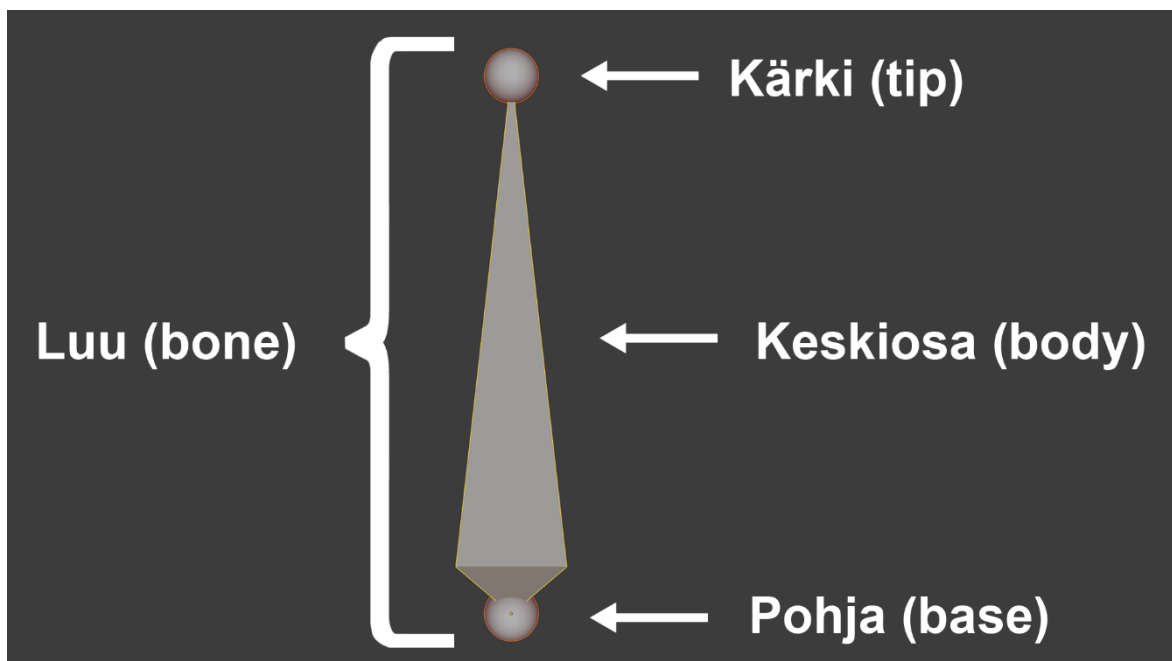
4 Animointi

4.1 Bone & Skeleton

Animoinnin avulla saadaan luotua mallille liikettä. Sen animaation tarve vaihtelee sitä mukaan, mihin käyttötarkoitukseen se on luotu. Animaatio voidaan toteuttaa joko erillisellä 3D-animointiohjelmalla kuten Mixamo tai 3D-mallinnusohjelmalla kuten Blender.

Ennen kuin aletaan suunnitella mallille animaatiota, luodaan sille skeleton. Se on nimensä mukaan ns. mallinnuksen luuranko, joka koostuu boneista eli luista. Niiden avulla saadaan luotua mallille animaatio. (Petty 2020).

Luu muodostuu kolmesta osasta: kärki (tip), keskiosa (body) ja pohja (base), jotka näkyvät kuvassa 11. Sen pituutta voidaan lisätä ja paikkaa liikuttamalla sen kärkeä, mikä helpottaa luun paikoilleen asettamista. (Blain 2015, 365).



Kuva 11. Yksittäisen bonen eli luun osat (Blain 2015, muunneltu)

Näiden kyseisten luiden avulla voidaan luoda animaatiota niistä kohdista, missä sitä tarvitaan. Yksinkertaisien animaatioiden kuten pelkän koko mallin sijainnin (location) rotaation (rotation) ja skaalan (scale) muuttamiseen ei tarvita erillistä skeletonia.

Skeleton voi muodostua vain muutamasta luusta, jos mallin animaatio koostuu parin luun liikuttelusta. Esimerkiksi tuulenvireessä heiluva kukka tarvitsee animaatioonsa vain muutamman luun. Mallin skeleton voi myös koostua kymmenistä luusta, jos mallin rakenne on monimutkainen ja paljon animaatiota tarvitseva.

Itse tehty skeleton ei ole ainoa keino animaation luontiin mallille. Blenderissä on mahdollista ladata add-on nimeltä Rigify, joka sisältää valmiita skelitoneja muun muassa ihmisestä, kissasta ja linnusta (Blender 2.81 Manual). Käyttäessä niitä säästää aikaa animaation teossa.

Ihmisen kasvojen kohdalla skeletonissa on kymmeniä eri luita, joiden avulla voidaan luoda tarvittavia eri ilmeitä. Syy luiden suureen määrään on se, että ilmeiden tekemiseen käytetään eri kasvojen osa-alueita kuten kulmakarvoja, silmiä ja suuta. Ilman suuren luiden määrää kasvoissa mallin kaikkia ilmeitä olisi mahdotonta toteuttaa.

Valmiin skeletonin sovittaminen malliin ei kuitenkaan ole mutkatonta. Yleensä laitettaessa semmoista malliin joudutaan siirtelemään luita omille paikoilleen. Tästä syystä niiden yhteensopivuus suoraan liitettynä ei ole koskaan sataprosenttista, vaikka valmis skeleton ja malli olisivat samassa mittasuhteessa. Ilman luiden siirtämistä oikeille paikoilleen mallin animaatio tulee todennäköisesti epäonnistumaan.

4.2 Rigging

Skeletonin luomisprosessia kutsutaan nimellä rigging, jonka teko alkaa yksittäisestä luusta. Siinä kasvatetaan luiden lukumäärää tehden siitä skeletonin. Riggingiä käytetään animaatioissa, joita näkee videopeleissä, TV-ohjelmissa ja elokuvissa. (Petty 2020).

Ensimmäisen luun sijainti kannattaa miettiä tarkkaan, koska se määrittää mallin animointia ja siten sen liikkumista paljon. Esimerkiksi puun skeletonin ensimmäinen luu kannattaa sijoittaa lähelle sen pohjaa, josta sitten yksitellen kasvatetaan luiden määrää puun muodon mukaisesti. Tällöin mallin luita liikuttaessa, sen pohja pysyy paikoillaan ja eikä liiku muun animaation mukana. Esimerkki puun skeletonista kuvassa 12.



Kuva 12. Puumalli ja siihen luotu skeleton

Luiden lukumäärää voidaan kasvattaa lisäämällä uusi luu sen kärkeen tai pohjaan. Molempiin päihin voidaan lisätä useampi luu tehden skeletonista rihmastomaisen. Luiden lisääminen tapahtuu extrude-toiminnon avulla ja niiden sijaintia voidaan liikuttaa X-, Y- tai Z-akselin suuntaisesti. Luiden pituudet kannattaa asettaa niihin kohtiin malliin, mihin on suunniteltu animaatiota.

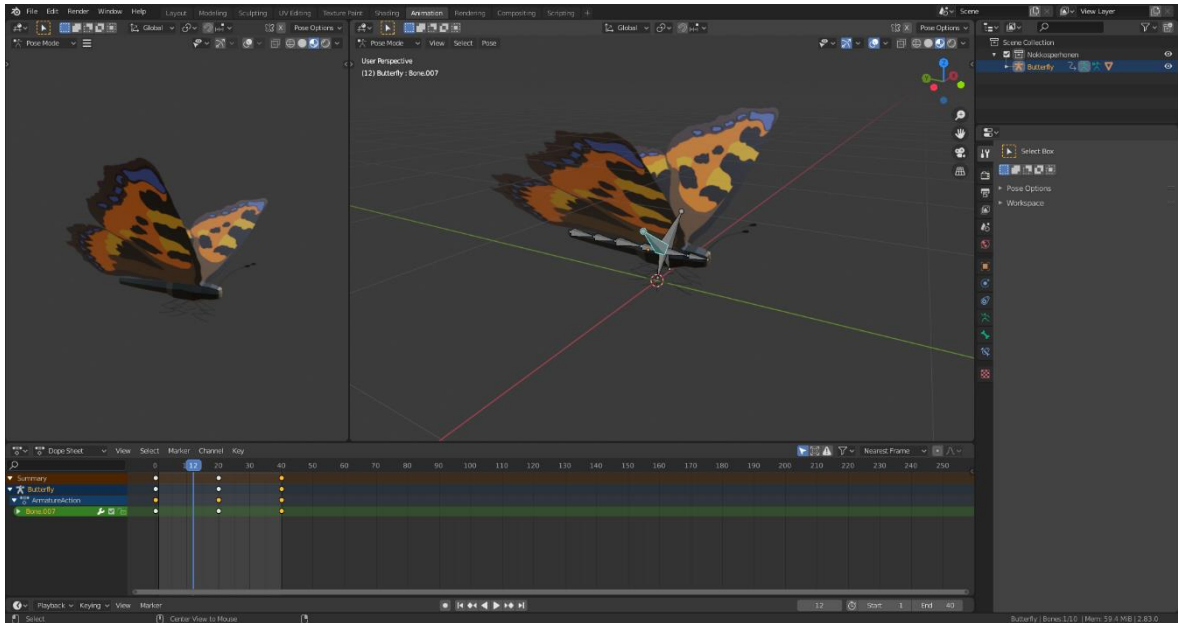
4.3 Animaation toteutus

Luiden määrä skeletonissa ja mallin geometria vaikuttavat paljon mallin animaatioon. Aloittaessa sen tekoa, kannattaa miettiä minkälaista animaatiota haluaa luoda mallille. Esimerkiksi kalan uimisanimaatio tarvitsee vaan muutaman liikkeen sen pyrstöön. Sen sijaan ihmisen juoksuanimaatio vaatii koko kehon liikkeiden animoimista, joka tekee mallin animaatiosta monipuolisemman.

Animointi kannattaa jakaa osiin, jos malli tulee sisältämään paljon liikettä. Tämän avulla on helpompi toteuttaa toimiva kokonaisuus, kun tiedetään millaisen animaation tulisi näyttää. Ilman kunnollista suunnitelmaa animaatiosta voi tulla sekava.

Blenderissä mallin animointi tapahtuu animointivälilehdessä nimeltä Animation. Siinä mallin näkyminen on jaettu kahteen osaan: vasemmalla puolella näkyy animaation lopputulos ja oikealla puolella tapahtuu animaation teko. Animaatiopuolella mallin mahdollinen skeleton näkyy, jos semmoinen löytyy mallista. Näiden alapuolella on aikajana, jossa näkyy sillä

hetkellä valitun luun animaatiot. Esimerkiksi kuvassa 13 näkyy, kuinka perhosen animointi on toteutettu Blenderin Animation-välilehdellä.



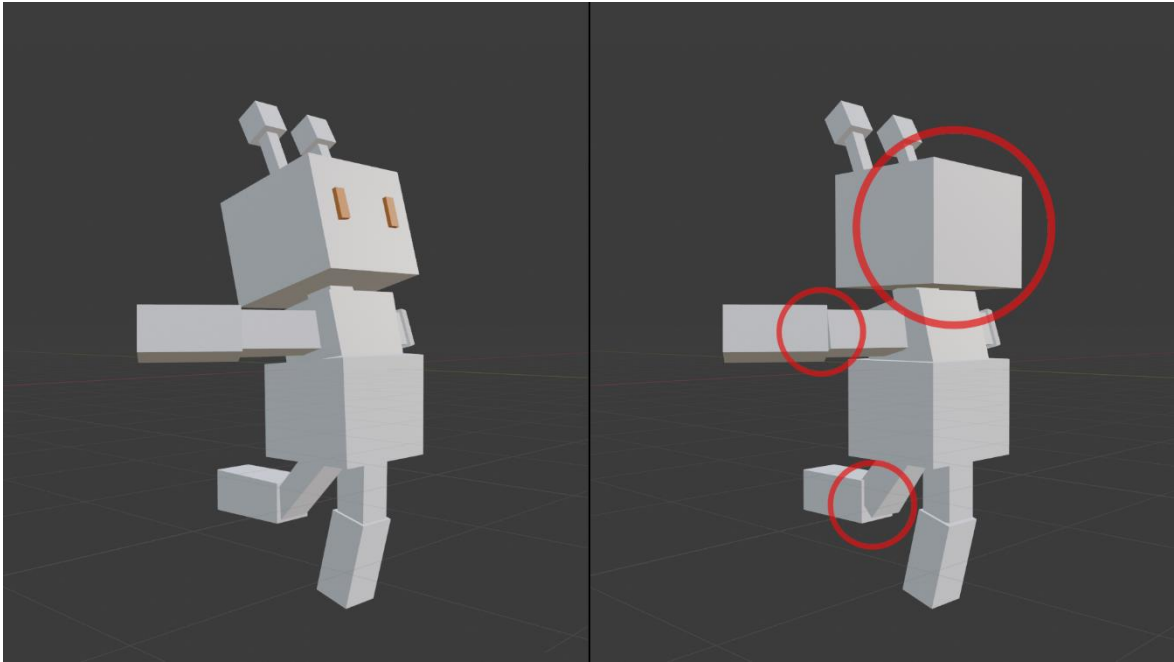
Kuva 13. Nokkosperhosen animointia

Yksittäisten luiden avulla voidaan liikutella mallin tarvittavia osia kolmen eri translaation avulla: sijainnin, rotaation ja skaalan. Nämä kolme perustoimintoa ovat avain mallin animaation luontiin, joka tuo sille elävyyden tuntua. (Blain 2015, 174).

Skeletonin luihin tehdyt animaatiot nähdään aikajanalla oransseina palloina. Niitä kutsutaan nimeltään keyframe, jotka kertovat mihin aikajanan kohtaan tai kohtiin on tehty animaatiota. (Blain 2015, 414–415). Näiden lisäksi voidaan nähdä, onko yksittäisessä keyframessa muokattu sijaintia, rotaatiota tai/ja skaalaa ja kuinka kauan se kestää. Keyframeja on helppo kopioida ja liittää uuteen kohtaan, jos animaatiossa sisältää paljon toistoa.

Mallin luomisvaihe tulee vaikuttamaan sen animaatioon. Yhdestä objektista koostuva mallin animaatio pysyy kasassa todennäköisemmin kuin useammasta objektista koostuva. Tämä johtuu siitä, että yhdestä objektista koostuvan mallin osat ovat kiinni toisissaan polygonien avulla ja tämä pitää sen koossa. Se tekee animoinnista helpompaa. Vastaavasti useammasta objektista koostuva mallin objektit eivät ole kiinni toisissaan samalla tavalla kuin yhdestä objektista muodostuva. Se voi pahimmassa tapauksessa rikkoo mallin osiin, kun sitä aletaan animoimaan.

Alla olevasta kuvasta 14 näkyy kaksi samannäköistä hahmoa, mutta vasemmanpuolimmainen on luotu yhdestä objektista ja oikeanpuolimmainen useammasta. Molemmat ovat samankokoisia, omaavat identtiset skeletonin ja asennon. Kuvasta 14 huomataan, kuinka paljon hahmon luontitapa vaikuttaa paljon sen animaatioon.



Kuva 14. Esimerkki robottihahmojen asennosta

Useasta objektista koostuva mallia voidaan kuitenkin käyttää animaatiota hyödyksi vähän animaatiota kaipaavissa asioissa. Esimerkiksi perhosen siipien liikkuminen on parempi toteuttaa siipien koostuvan eri objekteista kuin ollessaan kiinni perhosessa, koska animaatiovaiheessa siivet tulisivat venyttämään perhosen kehoa huomattavasti. Tämä tekisi animaatiosta luonnottoman näköisen, mikä tulisi pilaamaan mallin animaation kokonaan.

5 Käytännön osuus

5.1 Toimeksiantaja

Opinnäytetyön toimeksiantajana oli Tekoälypaja, joka on toiminut vuoden 2019 syksystä lähtien. Tekoälypajan tavoitteena on soveltaa ja rakentaa tekoälyä, joka kuuluu LAB-ammattikorkeakoulun hankkeeseen nimeltä Tekoälypaja. (LAB University of Applied Sciences). Siten se on osana LUT-konsernia.

LUT-konserni koostuu LUT-yliopistosta ja LAB-ammattikorkeakoulusta. Sen tarkoituksena on tehdä korkeakoulukeskittymästä kansainvälinen, joka näkyy korkeakouluissa yhteistyönä eri yritysten kanssa. (LUT University 2017).

LAB-ammattikorkeakouluun kuuluu kaksi kampusta, jotka sijaitsevat Lahdessa ja Lappeenrannassa. Kampukset ovat olleet vuodesta 2020 lähtien samaa ammattikorkeakoulua. (LUT University 2019).

Tekoälypaja on tekoälyn tutkimiseen ja kehittämiseen liittyvä työpaja, jonka tavoitteena on edistää terveyden ja hyvinvoinnin sekä kiertotalousalan pk-yrityksille sensori- ja tekoälyteknologian avulla. Vastaavaa avointa innovaatioalustaa ei vielä ole Lahden alueella, joka keskittyy tekoälyteknologioiden soveltamiseen. (LAB University of Applied Sciences).

5.2 Työn tavoitteet

Työn tarkoituksena oli tehdä low poly -malleja Tekoälypajan VR-peliprojektiin, jotka tullaan toteuttamaan Blenderillä. Blenderin avulla saadaan toteutettua yhteensopivat mallit VR-peliin, minkä Tekoälypaja tulee toteuttamaan Unity-pelimoottorilla.

VR-pelissä pelaaja pääsee low poly -maailmaa tekemään eri aktiviteettejä kuten jousiamuntaa ja tammenterhojen poimintaa. Nämä aktiviteetit sijaitsevat low poly -maailman luonnossa, joka jäljittelee todellisen maailman luonnon ympäristöä. Aktiviteettien lisäksi pelissä pääsee myös ihaillemaan pelimaailman luontoa. VR-pelin tarkoituksena on saamaan suuret ikäluokat liikkumaan pelissä olevien aktiviteettien avulla.

Mallien piti liittyä jollain tapaan luontoon ja retkeilyyn, jotka sopivat VR-peliprojektin teemaan. Niiden piti myös olla myös yhteensopivia jo olemassa olevien mallinnuksien kanssa, jotka ovat toteutettu low poly -tyylillä. Niiden värimaailma koostui yksittäisistä väreistä, joten uudet mallit eivät saisi sisältää liikaa yksityiskohtia. Liian tarkat yksityiskohdat tulisi rikkoamaan illuusion VR-pelin low poly -maailmasta.

Ulkonäön lisäksi osalle mallille tuli toteuttaa animaatio, jotka jäljittelisivät luonnollista liikettä tehden enemmän eloa VR-peliin. Näitä olisi muun muassa perhosien lentämisanimaatio ja kalojen uimisanimaatio.

5.3 Toteutus

Työn tavoitteiden ollessa selvät, alkoi niiden toteutus. Mallien luontiin oli varattu kolme viikkoa, koska harjoittelujaksoa oli vain enää sen verran jäljellä. Aloitettaessa mallien tekoa sain listan VR-peliprojektiin tarvittavista eri malleista. Jokaisessa mallissa oli kerrottu, tarvitsiko se animaatiota ja minkälaista sen piti olla. Ne oli myös jaettu kahteen eri kategoriaan: kiireellisiin ja vapaavalintaisiin.

Kiireelliset mallit olivat sellaisia, jotka tarvittiin mahdollisimman pian VR-peliin ja olivat jotenkin osana vuorovaikutteista pelikokemusta. Näitä malleja olivat muun muassa kori ja tammenterho, joiden avulla saatiin tehtyä tammenterhojen poiminta-aktiviteetti VR-peliin. Kuvassa 15 näkyy tammenterhon poiminta-aktiviteetistä.



Kuva 15. Kuvakaappaus tammenterhojen poiminta-aktiviteetistä (Tuomisto 2020)

Vapaavalintaiset mallit olivat puolestaan vaan osana VR-pelin maailmaa ilman vuorovaikutusta ja kyseiset mallit toteutettiin vasta kiireellisten jälkeen. Näitä oli paljon enemmän kuin kiireellisiä ja siten niitä tehtiin niin paljon, kun ehti tekemään. Tähän syynä oli aikaraja, mikä

rajoitti kaikkien mallien toteuttamista ajoissa. Kuvassa 16 näkyy pari vapaavalinnaista mallia, jotka toteutettiin lyhyellä aikataululla.



Kuva 16. Puussa tarkkaileva pöllö (Tuomisto 2020)

Mallintamisessa sai käyttää mitä tahansa mallinnusohjelmaa eli sitä ei ollut rajattu tietyn ohjelman käyttämiseen. Niinpä low poly -mallit toteutettiin Blenderin 2.83.0-versiossa, josta oli eniten kokemusta. Toteutushetkellä Blender 2.83.0 oli uusin versio mallinnusohjelmasta.

Low poly -mallien teossa käytettiin box modeling -mallintamistekniikkaa, jolla oli kätevä luoda mallit. Sen avulla onnistui mallintamaan sekä yksinkertaiset että monimutkaiset geometriaa omaavat mallit ja niiden apuna tarvittaessa käytettiin myös loop cutia, extrudea ja subdivisionia. Symmetriset mallien mallintamisessa käytettiin mirror-toimintoa.

Kaikille low poly -malleille lisättiin väri joko värimateriaalina tai teksturointina. Yksivärisille ja monivärisille malleille lisättiin suoraan värimateriaalit materials-välilehdestä. Näin kyseisille malleille ei tarvinnut tehdä erillistä teksturointia. Värimateriaaleja käytettiin muun muassa laavun ja laiturin teossa.

Teksturointia eli yksityiskohtia tarvitsevat mallien teksturointipohjan UV Map luotiin Smart UV Projectilla. Tämä oli tekohetkellä henkilökohtaisesti käytetyin tapa luoda UV Map, koska sen luonnista muulla tavalla ei ollut kokemusta. Teksturointi ja niiden prosessit tehtiin vain niihin malleihin, jotka tarvitsivat yksityiskohtia. Nämä jaettiin kahteen kategoriaan:

yksinkertaisiin ja monimutkaisiin. Molemmat kategoriassa olevat teksturoinnit toteutettiin eri ohjelmilla.

Yksinkertaisien yksityiskohtia tarvitsevat mallit toteutettiin Blenderin Texture Paint -välilehdellä. Sillä pohjaväriin luonti UV Mapiin onnistui helposti bucket-työkalun avulla, jolloin se sai yksivärisen pohjan. Tämän jälkeen siihen pystyttiin lisäämään eri värejä brush-työkalun avulla joko maalamalla suoraan mallin pintaan tai teksturointiin.

Monimutkaisemmat yksityiskohdat puolestaan toteutettiin PaintTool SAI -piirustusohjelmalla, koska ohjelma sisälsi enemmän työkaluja teksturoinnin piirtämiseen kuin Blenderin Texture Paintissa. Kyseisellä ohjelmalla oli myös helppo toteuttaa malleihin teksturoinnit, kunhan mallin UV Map ensiksi erotettiin teksturointipohjaväristä omalla pohjavälillä. Ilman värierottelua ei olisi pystytty erottamaan UV Mapia teksturointipohjasta. PaintTool SAI:ta käytettiin muun muassa pöllön ja perhosien teksturoinnissa. Perhosten lopullisen ulkonäön näkee kuvassa 17.



Kuva 17. Erilaisia perhosia VR-pelissä (Tuomisto 2020)

Animointi ja sen vaiheet toteutettiin myös Blenderissä. Mallien skeletonit luotiin yhdestä luusta lähtien ja niiden määrä riippui mallin geometriasta. Jokaiselle animointia tarvitsevalla mallille animointi toteutettiin Blenderin Animation-välilehdellä, jossa oli tarvittavat työkalut animointiin. Animaatioiden luonnissa liikuteltiin mallin yksittäisiä luita tehden sille

animaatiota. Low poly -mallien animointia oli muun muassa perhosilla lentämisanimaatiota ja kaloilla uimisanimaatiota. Kuvassa 18 näkyy pelissä olevia kaloja lammessa.



Kuva 18. Kalojen uiskentelua (Tuomisto 2020)

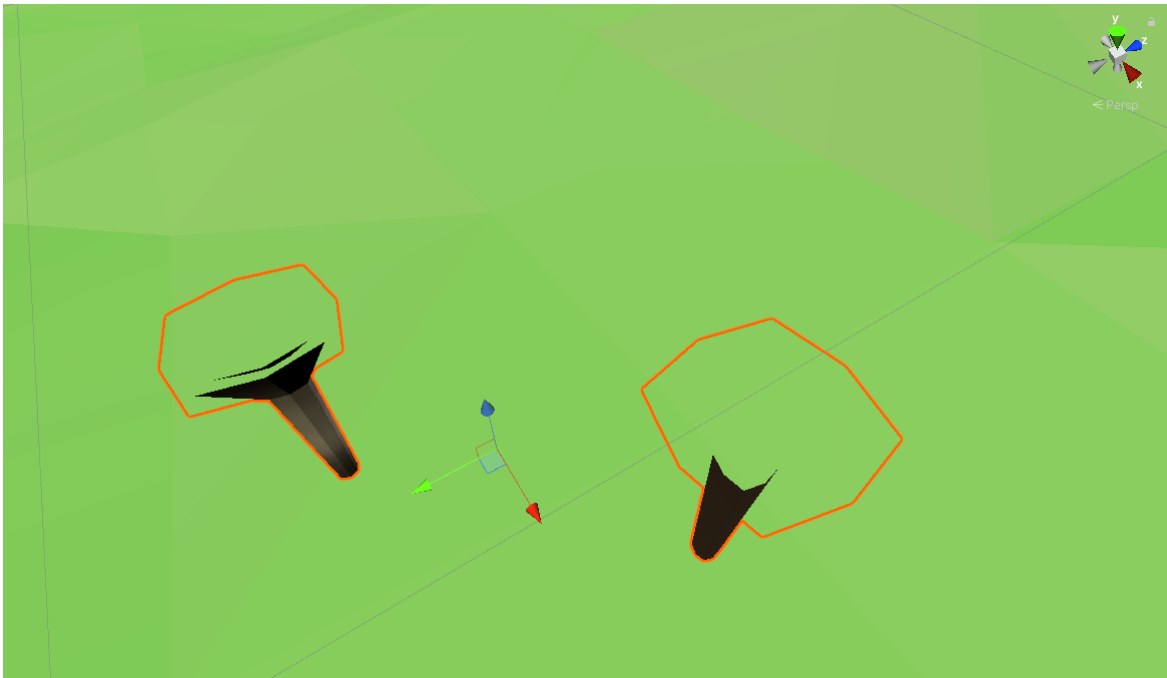
Lopuksi kaikki mallit tarkastettiin, että ne olivat helposti tunnistettavissa oikean elämän asioihin ja eläimiin. Myös teksturointia tai/ja animaatiota omaavat mallit tarkistettiin, että tekstuurit sopisivat hyvin mallien pintaan ja animaatiot toimisivat halutulla tavalla.

5.4 Ongelmat ja ratkaisut

Kuitenkaan projektin aikana ongelmilta ei voitu välttyä, vaikka low poly -mallien luonti onnistui helposti. Ongelmia esiintyi parissa mallissa: puun yhdestä lehtipuskasta puuttui puolet, ja mustatorvisienestä oli kadonnut hattuosa kokonaan. Nämä ongelmat näkyivät vain Unityn puolella, kun ne importattiin VR-peliin. Blenderissä kuitenkin mallit näkyivät ja toimivat oikein ilman mitään ongelmia.

Aluksi epäily low poly -mallien ongelmaan oli Unityssä, koska kyseinen ongelma näkyi vain siellä. Kuitenkaan Unityn puolella ei voinut olla ongelmaa, koska muut importatut mallit toimivat normaalisti VR-pelissä eli pelkkä mallien importaus ei voinut aiheuttaa tiettyjen mallien polygonien katoamista.

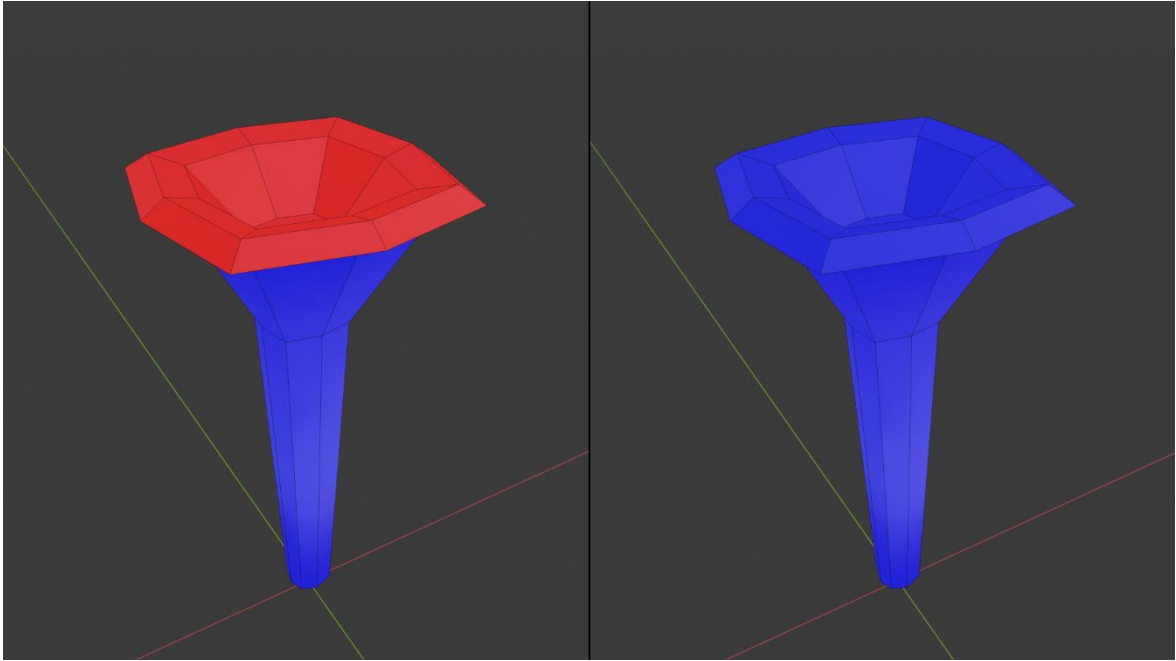
Ratkaisu ongelman syyhyn oli aluksi vaikea löytää, kun ei tiedetty missä ongelman alkuperä sijaitisi ja millä sanoilla ongelmaa kuvailisi. Etsittyä jonkin aikaa löytyi ongelmaan ratkaisu. Ongelma oli tullut Blenderin puolelta ja se jostain syystä oli ns. kääntänyt mallien tietyt polygonit väärinpäin. Tämä teki niistä rikkiäisen näköisiä. Kyseinen ongelma näkyi Unityssä kuva 19 lailla.



Kuva 19. Mustatorvisienen näkyminen Unityssä (Tuomisto 2020)

Ongelmaan oli kuitenkin yksinkertainen ratkaisu. Blenderissä pystyy Face Orientation -nimisessä moodissa katsomaan, jos mallissa olisi poikkeamia. Ns. terve malli näkyisi Face Orientation -moodissa kokonaan sinisenä joka puolelta. Vastaavasti ongelmakohdat näkyisivät punaisena.

Käytettyään Face Orientation -moodia ongelmia tuottavissa malleissa, havaittiin niiden ongelmakohdat. Malleista valittiin vain punaiset facet eli polygonit ja ne käännettiin takaisin sinisiksi näppäinkomennolla Alt + N > Flip. Tämä korjasi mallien ongelman heti ja ne näkyivät normaalisti VR-pelissä muiden mallien lailla. Alapuolella olevassa kuva 20 näkyy, miten ongelma näkyi Face Orientation -moodissa ja miltä mustatorvisieni näytti korjaamisen jälkeen.



Kuva 20. Korjaamaton ja korjattu versio mustatorvisienestä Face Orientation -moodissa

Yhteenvetona Face Orientationin avulla saatiin näkyville malleissa olevat virheet ja siten ne saatiin korjattua. Vaikka Blenderin päästä tuli kyseinen ongelma malleihin, silti tulee jäämään ikuisesti mysteeriksi miksi Blender ylipäätänsä oli päättänyt kääntää kyseiset polygonit väärinpäin tietyistä malleista.

6 Yhteenveto

Lopputuloksena saatiin toimivat low poly -mallit Tekoälypajan VR-peliprojektiin pienistä ongelmista huolimatta. Ne sopivat yhteen VR-peliprojektin aikaisempien mallien kanssa.

Työnantaja oli todella tyytyväinen tehtyihin low poly -malleihin ja siitä, että sain kiireellisemmät mallit tehtyä nopealla aikataululla valmiiksi. Työnantaja oli myös kiitollinen siitä, että jaksoin tehdä vapaavalintaisia malleja heidän peliprojektiinsa kiireellisten lisäksi. Työnantaja painoittikin, että kunhan saisin edes kiireelliset mallit valmiiksi ajoissa, niin olisivat tyytyväisiä saamiin malleihin.

Henkilökohtaisella tasolla olin myös todella tyytyväinen tehtyihin low poly -malleihin teksturointia ja animaatioita myöden, koska niiden luominen onnistui haluamallani tavalla. Myös niiden tekeminen oli mielekästä hommaa. Olisin kuitenkin tehnyt mallien UV Unwrapin toisella tavalla.

UV Unwrap onnistui mallin tekovaiheessa kätevästi Smart UV Projectin avulla, mutta nykyään tekisin sen Unwrapillä. Unwrapin avulla olisi saanut UV Mapista selkeämmän näköisemmän, kun olisi voinut itse päättää mallin saumojen rajat. Mallien tekohetkellä en tiennyt paljoa teksturoinnin teosta kuten miten tehdään saumoja malliin ja miten se vaikuttaa UV Mapiin. Silloin tiesin vain Smart UV Project -tavasta, jota olin käyttänyt paljon ennen VR-peliprojektia.

Oppimisen kannalta mallien toteutus oli todella hyödyllinen. Pääsin käyttämään mallintamistaitojani ja kehittää sitä lisää VR-peliprojektin avulla. Olen vapaa-ajollani mallintanut Blenderillä jonkin verran ja oli unelma päästä käyttämään mallinnustaitojani muuhunkin kuin pelkästään harrastusmielessä.

VR-peliprojektiin voisi kehitellä lisää eri malleja, joita kaikkia en kerennyt toteuttamaan yksin harjoittelun aikarajan tullessaan vastaan. Muuten olisin mielellään mallintanut kaikki halutut mallit VR-peliin, jos siihen olisi ollut enemmän aikaa.

Lähteet

Blain, J. M. 2015. The Complete Guide to Blender Graphics: Computer Modeling & Animation. 2. painos. Cleveland: CRC Press.

Blender. About. Blender. Viitattu 10.9.2020. Saatavissa: <https://www.blender.org/about/>

Blender 2.79 Manual. Mapping Types. Blender. Viitattu 17.9.2020. Saatavissa: https://docs.blender.org/manual/en/2.79/editors/uv_image/uv/editing/unwrapping/mapping_types.html

Blender 2.81 Manual. Rigify. Blender. Viitattu 17.9.2020. Saatavissa: <https://docs.blender.org/manual/en/2.81/addons/rigging/rigify.html>

Blender 2.90 Manual. Image or Movie. Blender. Viitattu 17.9.2020. Saatavissa: https://docs.blender.org/manual/en/latest/render/materials/legacy_textures/types/image_movie.html

Denham, T. 2020. What is UV Mapping & Unwrapping? Concept Art Empire. Viitattu 9.9.2020. Saatavissa: <https://conceptartempire.com/uv-mapping-unwrapping/>

LAB University of Applied Sciences. Tekoälypaja. LAB University of Applied Sciences. Viitattu 1.10.2020. Saatavissa: <https://lab.fi/fi/projekti/tekoalypaja>

LUT University. 2017. LUT-konserni – mitä se tarkoittaa käytännössä? LUT University. Viitattu 9.10.2020. Saatavissa: <https://blogs.lut.fi/abitiimi/lut-konserni-mita-se-tarkoittaa-kaytannossa/>

LUT University. 2019. Lahden ammattikorkeakoulun tukipalvelut siirtyneet LUT-yliopistoon. LUT University. Viitattu 9.10.2020. Saatavissa: https://www.lut.fi/uutiset/-/asset_publisher/h33vOeufOQWn/content/lahden-ammattikorkeakoulun-tukipalvelut-siirtyneet-lut-yliopistoon

Munn, R; Derakhshani, D. 2008. Introducing 3ds Max 2008. John Wiley & Sons, Incorporated. Viitattu 11.9.2020. Saatavissa: <https://ebookcentral-proquest-com.ezproxy.saimia.fi/lib/lab-ebooks/detail.action?pq-origsite=primo&docID=331664>

Petty, J. 2020. What is 3D Rigging for Animation & Character Design? Concept Art Empire. Viitattu 28.9.2020. Saatavissa: <https://conceptartempire.com/what-is-rigging/>

Rector, E. 2019. What is 3D Modeling and Design? A Beginners Guide to 3D. MarketScale. Viitattu 16.9.2020. Saatavissa: <https://marketscale.com/industries/building-management/what-is-3d-modeling-and-design-a-beginners-guide-to-3d/>

Selin, E. 10 Different types of 3D modeling techniques. Artistic Render. Viitattu 16.9.2020. Saatavissa: <https://artisticrender.com/10-different-types-of-3d-modeling-techniques/>

Slick, J. 2019. 7 Common Modeling Techniques for Film and Games. Lifewire. Viitattu 9.10.2020. Saatavissa: <https://www.lifewire.com/common-modeling-techniques-for-film-1953>

Treehouse. 2015. Understanding Normal Maps. Treehouse. Viitattu 1.10.2020. Saatavissa: <https://blog.teamtreehouse.com/understanding-normal-maps>

Turbosquid 3D Resources. Tris, Quads & N-Gons. Turbosquid 3D Resources. Viitattu 5.10.2020. Saatavissa: <https://resources.turbosquid.com/training/modeling/tris-quads-n-gons/>

Unity. 2020a. Quick guide to the Unity Asset Store. Unity. Viitattu 15.9.2020. Saatavissa: <https://unity3d.com/quick-guide-to-unity-asset-store>

Unity. 2020b. Coding in C# in Unity for beginners. Unity. Viitattu 15.9.2020. Saatavissa: <https://unity3d.com/learning-c-sharp-in-unity-for-beginners>

Unity Documentation. 2018. Creating and Using Scripts. Unity Documentation. Viitattu 15.9.2020. Saatavissa: <https://docs.unity3d.com/Manual/CreatingAndUsingScripts.html>

Unity Store. 2020. Choose the plan that is right for you. Unity. Viitattu 15.9.2020. Saatavissa: <https://store.unity.com/compare-plans>

Kuvalähteet

Blain J.M. 2015. Figure 15.1 (muunneltu). Viitattu 19.10.2020. Cleveland: CRC Press.

Tuomisto, J. 2020. Kuvat 15–19. Viitattu 9.10.2020.